CrimeStat® IV

VERSION 4.0

A Spatial Statistics Program for the Analysis of Crime Incident Locations

Ned Levine & Associates

Houston, TX

The National Institute of Justice

Washington, DC

June 2013

Table of Contents

Major Chapter Headings

Part I: Program Overview

Part II: Spatial Description

Part V: Spatial Modeling II

Acknowledgments

CrimeStat IV (version 4.0) was developed under the direction of Dr. Ned Levine of ***Ned Levine & Associates***, Houston, TX with Grant 2005-IJ-CX-K037 from the Office of Science and Technology***, National Institute of Justice*** (NIJ), Washington, DC. The developer would like to thank the many individuals who contributed to this program over the years since its inception:

1. Ms. Haiyan Teng of Houston, TX, the primary programmer for versions 2.0 through 4.0. Her high level of programming competence and mathematical expertise was essential for the successful completion of the crime travel demand routines, the Bayesian Journey to Crime routine, the Head-Bang routine, and the Markov Chain Monte Carlo regression routines as well as ensuring that all new routines were properly integrated into the main program. She is a co-author of three chapters.

2. Mr. Ron Wilson, formerly project manager at the Mapping and Analysis for Public Safety Program (MAPS) at NIJ and currently a researcher at the U.S. Department of Housing and Urban Development, who supported the project through much of this development and provided valuable feedback on the new routines and their utility. He also has been instrumental in pushing for the development of CrimeStat libraries that will be released separately later this year.

3. Mr. Joel Hunt, current project manager at the Mapping and Analysis for Public Safety Program (MAPS) at NIJ. He took over administrative management of the project towards for NIJ towards the end of the development but has been supportive throughout.

4. Professor Richard Block of Loyola University in Chicago has contributed as a methodological and criminal justice advisor to the project since early in its development. He has played a critical role in conducting quality control tests and is the author of one chapter and the co-author of five chapters.

5. Dr. Shaw-pin Miaou of Transmidas Consulting Services in College Station, TX provided detailed instructions for building the MCMC Poisson-Gamma and Poisson-Gamma-CAR regression models. Ms. Haiyan Teng and Dr. Li Sheng converted the instructions into C++ and wrote numerical libraries for it.

6. Dr. Dominique Lord of Texas A & M University in College Station, TX provided detailed technical help on the Poisson and binomial maximum likelihood and

MCMC methods. He is also the co-author of six chapters on the regression module.

7. Dr. Byung-Jung Park of the Korea Transport Institute in Goyang, South Korea provided mathematical clarification of the models and also has developed alternative dispersion and spatial autoregressive functions. He is also the co-author of five chapters.

8. Professor Wim Bernasco of the Netherlands Institute for the Study of Crime and Law Enforcement and the Department of Spatial Economics, Faculty of Economics and Business Administration, VU University Amsterdam, Netherlands, for designing the discrete choice module and is the co-author of chapters 21 and 22.

9. Professor Wil Gorr of Carnegie-Mellon University in Pittsburgh, PA, for designing the time series forecasting module and is the co-author of chapters 23 and 24.

10. Dr. Andreas Olligschlaeger of TruNorth Data Systems, Inc. of Baden, PA for programming the time series forecasting module. He is the co-author of chapters 23 and 24.

11. Professor Shashi Shekhar of the University of Minnesota in Minneapolis for supervising the conversion of CrimeStat II routines into libraries.

12. Mr. Ian Cahill of Cahill Software, Edmonton, Alberta for providing OLS, Poisson, negative binomial, and multinomial regression maximum likelihood code, based on his MLE++ software package. http://www.magma.ca/~cahill. The code forms the engine of the maximum likelihood routines though additional capabilities have been added. He is a co-author on two chapters.

13. Dr. Srinivas Geedipally of the Texas Transportation Institute, Dallas, TX, for developing the MCMC version of the normal distribution along with the spatial autocorrelation variants. He is a co-author on three chapters.

14. Dr. Li Sheng, of Houston, TX helped Ms. Teng in programming the MCMC routines. In particular, he created a numerical library that allowed the algorithms to run much faster. He is a co-author on three chapters.

15. Dr. David Wong of George Mason University for providing advice on the Getis-Ord “G” statistic.

16. Mr. Daniel Helms of Scytale Consulting, Reston, VA, served as a criminal justice advisor to the project and played an important role in testing the crime travel demand routines that was developed in version 3.

17. Mr. Long Doan of *Doan Consulting*, Falls Church, VA was the original programmer for the project. Mr. Doan’s brilliance in programming was essential to the development of the initial program.

18. Professor Luc Anselin of the University of Illinois at Urbana-Champaign provided technical advice and documentation on the regression models used in the crime travel demand model that was developed in version 3.

19. Professor Peter Stopher of the University of Sidney in Australia provided technical advice on the crime travel demand model developed in version 3.

20. Professor Luc Anselin of Arizona State University, Tempe, AZ, provided technical advice on the OLS and Poisson regression models.

21. Mr. Phil Canter, formerly of the *Baltimore County Police Department*, Towson, MD, has been with the project since its inception. For this round, he provided support and data for analysis.

22. Ms. Sandra Wortham of *Wortham Design*, Wilmington, DE designed the graphical icons used in the program.

23. The GNU project library for providing F-test and t-test code. http://www.gnu.org.

24. Dr. Carolyn Rebecca Block of the Illinois Criminal Justice Information Authority for providing the STAC routine.

25. All the other individuals from the MAPS unit who have supported the project in earlier stages: for the third version, Ms. Debra Stoe; for the second version, Ms. Elizabeth Groff of the Institute of Law and Justice, Mr. Eric Jefferis of the University of Akron, and Professor Robert Langworthy of the University of Central Florida, Orlando, FL; and, for the first version, Ms. Cindy Mamalian and Dr. Nancy LaVigne of the Urban Institute.

26. Ms. Patsy Lee of the Greater Manchester Police Department, Manchester, England, for providing crime data on Manchester.

27. To Dr. Alan Robertson and Mr. Barry Fosberg of the Houston Police Department for providing crime data on Houston;

28. To the individuals of the Baltimore Metropolitan Council who provided network and other data on both Baltimore County and the City of Baltimore, in particular Jacqueline Zee, Matt de Rouville, and Gene Bandy. Thanks also to Alan Clark of the Houston-Galveston Area Council for making available data on Houston motor vehicle crashes.

29. To individuals who have provided detailed feedback and information for this and previous versions of *CrimeStat*: Professor Eric Renshaw of the University of Strathclyde in Glasgow, Mr. John DeVoe of Siebel Systems, Professor Jim LeBeau of Southern Illinois University, Mr. Bryan Hill of the Glendale (Arizona) Police Department, Professor Karl Kim of the University of Hawaii, Mr. Luben Dimov of Louisiana State University, Mr. Weijie Zhou of the Houston-Galveston Area Council, and Mr. Martin Hittleman of Valley Community College in Los Angeles.

30. To the individuals who provided example applications for the manual: Renato Assunção, Cláudio Beato, Bráulio Silva of the Federal University of Minas Gerais in Belo Horizonte, Brazil; Daniel Bibel of the Massachusetts State Police; Gilberto Câmara, Silvana Amaral, Antônio Miguel V. Monteiro, and José A. Quintanilha of the Instituto Nacional de Pesquisas Espaciais in Brazil; Spencer Chainey of InfoTech Enterprises Europe in London, England; Richard Crepeau of Appalachian State University; Jaishankar Karuppannan of the University of Madras in Chepauk, India; Yongmei Lu of Southwest Texas State University; David McGrath of the Johnstown Castle Research Centre in Wexford, Ireland; Nathalie Pavy and Jean Bousquet of Laval University of Quebec; Dietrich Oberwittler and Marc Wiesenhütter of the Max Planck Institute for Foreign and International Criminal Law in Freiburg, Germany; Derek Paulsen of Appalachian State University; Gaston Pezzuchi of the Buenos Aires Province Police Force; Mike Saweda of the University of Ottawa; Takahito Shimada of the National Police Agency in Chiba, Japan; Daisy Smith and Steph Winstanley of the Greater Manchester Police Department; Brent Snook, Paul Taylor & Craig Bennell of the University of Liverpool, England; Matthew Stone of the California Department of Health Services, Chaosheng Zhang of the National University of Ireland in

Galway, Ireland; Marta A. Guerra of the Centers for Disease Control and Prevention; Richard Hoskins of the State of Washington Department of Health; Tom Reynolds of the University of Texas School of Public Health along with Luc Anselin, Richard Block, Carolyn Block, Phil Canter, Long Doan, Daniel Helms, Jim LeBeau, Ron Wilson and Bryan Hill mentioned above.

31. To the dozens of individuals who provided feedback and suggestions for improving the program. They are, unfortunately, too numerous to list.

32. Finally, this program is dedicated to my wife, Dr. C. Elizabeth Castro, for being so patient and supportive throughout this long process. She is an inspiration to me for this whole effort.

Disclaimer and License Agreement

This project was supported by Grant 2005-IJ-CX-K037 awarded by the National Institute of Justice, Office of Justice Programs, US Department of Justice and built on earlier National Institute of Justice grants. Points of view in this document are those of the author and do not necessarily represent the official position or policies of the US Department of Justice.

 It can be distributed freely for educational or research purposes, but cannot be re-sold. It must be cited correctly in any publication or report that uses results from the program. The correct citation is:

> Ned Levine, *CrimeStat IV: A Spatial Statistics Program for the Analysis of Crime Incident Locations (version 4.0)*. Ned Levine & Associates, Houston, TX, and the National Institute of Justice, Washington, DC, June 2013.

implied, including but not limited to the warranties of merchantability and fitness for a particular purpose. In no event will Ned Levine and Associates, the United States Government or any of their respective employees be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the software or documentation. Neither Ned Levine and Associates, the United States Government, nor their respective employees are responsible for any costs including, but not limited to, those incurred as a result of lost profits or revenue, loss of time or use of software, loss of data, the costs of recovering such software or data, the cost of substitute software, or other similar costs. Any actions taken or documents printed as a result of using this software and its accompanying documentation remain the responsibility of the user.

Any questions about the use of this program should be directed to either:

Dr. Ned Levine
Ned Levine & Associates
Houston, TX
crimestat@nedlevine.com

Mr. Joel Hunt
Mapping and Analysis for Public Safety Program
National Institute of Justice
U. S. Department of Justice
810 7th St, NW

CrimeStat IV

Part I: Program Overview

Chapter 1:

Introduction to *CrimeStat IV*

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Chapter 1:
Introduction to *CrimeStat IV*

CrimeStat® is a spatial statistics package that can analyze crime incident location data. Its purpose is to provide a variety of tools for the spatial analysis of crime incidents or other point locations. It is a stand-alone *Windows* program that can interface with most desktop geographic information systems (GIS). It is designed to operate with large crime incident data sets collected by metropolitan police departments. However, it can be used for other types of applications involving point locations, such as the location of arrests, motor vehicle crashes, emergency medical service pickups, or facilities (e.g., police stations).

Uses of Spatial Statistics in Crime Analysis

Most GIS packages, such as *MapInfo*®, *ArcGIS*®, and *Maptitude*®, have very sophisticated data base operations (Pitney Bowes, 2012; ESRI, 2012a; Caliper, 2012). They have limited statistical methods, however, though this has been slowly changing over time as the programs have added various statistical functions. For most purposes, GIS can provide great utility for crime analysis, allowing the plotting of different incident locations and the ability to select subsets of the data (e.g., incidents by precinct, incidents by time of day). Most crime analysts visually inspect incident maps and, based on their experience, draw conclusions about shifts over time, 'hot spots' and other patterns suggested by the data.

There are times, however, when a more quantitative approach is needed. For example, an analyst wishing to examine patterns of streets robberies over time will need indices which document how the robberies may have shifted. For a neighborhood showing an apparent sudden increase in auto thefts, there needs to be a quantitative standard to define the 'typical' level of auto thefts. In assigning police cars to patrol particular major arteries, the center of minimum travel needs to be identified in order to maximize response time to calls for service. For research, as well, quantification is important. In examining correlates of burglaries, for example, a researcher needs to determine the exposure level, namely how many residences or commercial buildings exist in a community in order to establish a level of burglary risk. Or a precinct station may want to target areas for which there is a high concentration of incidents occurring within a short time ('hot spots'). While some of these analyses can be conducted with GIS queries, quantification can allow a more precise identification and the ability to compare different types of incidents. In short, there are many uses for quantitative analysis for which a statistical program becomes important.

The *CrimeStat IV* Spatial Statistics Program

CrimeStat is a program designed to provide statistical summaries and models of crime incident data. The program provides crime analysts and researchers with a wide range of spatial statistical procedures that can be linked to a GIS. The procedures vary from the simple to some very sophisticated 'cutting edge' routines. The reasoning is that different audiences vary in their needs and requirements. The program should be of benefit to different organizations. For many crime analysts, simple descriptions of the spatial distribution will be sufficient with the aim being practical intervention over a short time period. For these persons, many of the techniques provided in *CrimeStat* will be unnecessary.

For other analysts, statistical tools can supplement a much larger GIS effort, such as the sophisticated crime analysis system built by the Baltimore County Police Department (BCPD, 2012). For other researchers, even more demanding techniques may be needed to detect the underlying spatial structure as a means for formulating a temporal-spatial theory. A pattern in and of itself has little meaning unless it is linked to some framework. The ability to quantify relationships with a large amount of data can address problems that previously were avoided and can be a first step in developing an explanatory framework or interventionist strategy. *CrimeStat* attempts to address both types of needs by providing statistics in a 'toolbox' framework. We recognize that today's exotic statistical techniques may become tomorrow's practical diagnostics and want the program to be useful for many years.

Input and Output

CrimeStat is a full-featured *Windows* program using a graphical interface with database and expanded statistical functions. It can read files in various formats - *dBase*®, which is a common file format in desktop GIS programs, Excel (both 'xls' and 'xlsx' formats), *ArcGIS* Shape (shp) files, *MapInfo* data (dat) files, and files conforming to the ODBC standard, such as Lotus 1-2-3, and Microsoft Access (Microsoft, 2010). In addition, many other GIS packages, such as *Maptitude*® can read 'dbf', 'shp', 'bna' or 'mif' files.

Output includes both displayed tables, which can be printed as text or copied to a word processing program, and graphical output. *CrimeStat* can write graphical objects to the *ArcGis*®, *MapInfo*®, *Maptitude* GIS programs, *Surfer*® *10, ArcGIS Spatial Analyst*© programs, and to those that can read Ascii grid files (e.g., *Vertical Mapper*®; Rockware, 2012; Golden Software, 2012; ESRI, 2012b).

Statistical Routines

CrimeStat IV includes statistics routines for both statistical description and modeling. These are divided into six general statistical categories with more than 80 individual routines:

Data Setup

Primary file

Input file with X/Y coordinates
Define coordinate system
Define data units

Secondary file

Input second file with X/Y coordinates as baseline
Define coordinate system
Define data units

Reference file

Create reference grid
Use existing reference grid

Type of distance measurement

Use direct distance
Use indirect distance
Use network distance

Spatial Description

Spatial distribution

Mean center
Standard distance deviation
Standard deviational ellipse
Median center
Center of minimum distance
Directional mean and variance
Convex Hull

Spatial Autocorrelation

Moran's "I" spatial autocorrelation index
Geary's "C" spatial autocorrelation index
Adjusted Geary's "C" spatial autocorrelation index
Getis-Ord Global "G" spatial autocorrelation index with simulation of credible intervals
Moran Correlogram with simulation of credible intervals
Geary Correlogram with simulation of credible intervals
Getis-Ord Correlogram with simulation of credible intervals

Distance analysis I

Nearest neighbor analysis
Ripley's "K" statistic
Assign primary points to secondary points

Distance Analysis II

Within primary file distance matrix
Between primary file and secondary file distance matrix
Between primary file and grid distance matrix
Between secondary file and grid distance matrix

Hot Spot Analysis

Hot spot analysis I

Mode
Fuzzy mode
Nearest neighbor hierarchical clustering with simulation of credible intervals
Risk-adjusted nearest neighbor hierarchical clustering with simulation of credible intervals

Hot spot analysis II

Spatial and temporal analysis of crime routine (STAC) with simulation of credible intervals
K-mean clustering

Hot spot analysis of Zones

Anselin's local Moran test with simulation of credible intervals
Getis-Ord local "G" test with simulation of credible intervals
Zonal nearest neighbor hierarchical clustering with simulation of credible intervals

Risk-adjusted zonal nearest neighbor hierarchical clustering with simulation of credible intervals

Spatial Modeling I

Interpolation I

Single variable kernel density interpolation
Dual variable kernel density interpolation

Interpolation II

Head-Bang analysis
Interpolated Head-Bang analysis

Space-time analysis

Knox index
Mantel index
Correlated walk model for analysis and prediction

Journey-to-crime analysis

Calibrate Journey-to-crime function
Journey-to-crime estimation
Draw crime trips

Bayesian Journey-to-crime analysis

Diagnostics for Journey-to-crime methods
Estimate likely origin of a serial offender

Spatial Modeling II

Regression I

MLE OLS and Poisson regression models
MCMC Poisson and Logit regression models
MCMC Poisson and Logit exposure regression models
MCMC spatial Poisson and Logit regression models
MCMC spatial Poisson and Logit exposure regression models

Regression II

Using OLS regression models to make predictions
Using Poisson spatial regression models to make predictions

Discrete Choice I

Create dataset for conditional logit model
Estimate multinomial logit model
Estimate conditional logit model

Discrete Choice II

Using multinomial logit model to make predictions
Using conditional logit model to make predictions

Time Series Forecasting

Time Series Data
Extrapolative Time Series Forecasting
Classical Decomposition: Seasonality
The Detection Problem

Crime Travel Demand

Trip Generation

Skewness diagnostics
Calibrate model
Make prediction
Balance predicted origins & destinations

Trip Distribution

Calculate observed origin-destination trips
Calibrate impedance function
Calibrate origin-destination model
Apply predicted origin-destination model
Compare observed and predicted origin-destination trip lengths

Mode Split

Calculate mode split for trips

Network Assignment

Check for one-way streets
Create a transit network from primary file
Network assignment of trips to travel network

Many of these routines allow variations yielding an even larger number of statistics to be calculated. Two features of the program should be noted. First, and foremost, *CrimeStat* is a

program that specializes in the analysis of point locations. Over the years, many statistical tools have been developed for analyzing point locations. Many of these have either not been implemented as computer programs or were collected together as part of a specialized statistical system. They have been typically unavailable to crime analysts and the major statistical packages (e.g., *SAS®*, *SPSS™*, *Systat®*) do not include these routines. Consequently, we have collected those that are most appropriate for crime analysis and detection and organized them into a single package with a common graphical interface. They represent a wide variety of tools that can be used for crime analysis. *CrimeStat* can also analyze zonal data by treating them as 'pseudo' points. For example, the centroid of a census tract can be treated as a point and a value associated with the tract (e.g., its population) can be treated as an Intensity value (see chapter 3).

Second, *CrimeStat* includes a variety of modeling tools for analyzing multivariate relationships. These include spatial regression routines for analyzing skewed distributions along with spatial autocorrelation, discrete choice modeling routines for modeling unique decisions, time series forecasting routines for analyzing sudden changes in the level of incidents in a zone, and a crime travel demand module for analyzing crime patterns over an entire metropolitan area.

Program Requirements

Required Hardware and Operating System

CrimeStat IV was developed for the *Windows 7* and *Windows 8* operating systems, though it will also work with the *Windows 2000*, or *Windows XP* operating systems; it is not hardware dependent so that any processor that can run *Windows 7* or *Windows 8* will suffice. Some of the routines can also run on *Windows XP* and earlier Windows operating systems. However, the program was not designed around nor fully tested for those operating systems. It is highly recommended that the program be run on a more current version of *Windows*.

While it can run on a relatively slow computer (e.g., 250 MHz clock speed) with limited RAM (e.g., 64 MB), it will run much better on a 2.6 GHz computer (or faster) with more than 2 GB of RAM. In general, the faster the processor used and the more RAM, the quicker the program will run. The program is very intensive with respect to calculations. Some of the statistics produce very large matrices (e.g., the trip distribution routines in the Crime Travel Demand module). Depending on the size of the data files that will be processed, there may be hundreds of millions of calculations on any one run. It is critical, therefore, that the computer be fast and have sufficient amounts of RAM.

Available RAM Limits the Size of Files

For most of the simple statistics, a reasonably fast computer will be adequate. However,

the Markov Chain Monte Carlo (MCMC) regression routines, the discrete choice module, the temporal modeling module, and several of the trip distribution routines will push the limits of most computer systems. For example, the 32 bit Windows operating system has a maximum addressable limit of 4 GB (i.e., 4 billion bytes) of RAM (actual and virtual). With a trip distribution matrix, there are *M x N* cells where *M* is the number of rows (origins) and *N* is the number of columns (destinations). With 8 bytes (64 bits) being assigned to a number in a cell (including the decimal and decimal places), practically the maximum matrix that could be loaded into memory would be about 22,000 x 22,000. One would never be able to use this amount since a lot of RAM will be taken up by the program and operating system. Nevertheless, using the calculation, the storage space required to save such a matrix will limit the size of the database, aside from taking a very long time to be calculated. In short, the size of the files that can be processed will depend on the particular routines being run.

With a 64 bit operating system (e.g., Windows 7 or Windows 8), the theoretical maximum for addressable memory is 192 GB. Again, with 8 bytes per cell, the available RAM would allow a maximum square matrix of about 154,000 x 154,000 cells. Clearly, for large data sets, a 64 bit Windows operating system is preferable.

Multi-threading

CrimeStat is a multi-threaded application written to take advantage of multiple processors if the hardware and operating system support multiple processors. The program is designed to be multi-threading which means that it will take advantage of multiple processors (called 'cores') using *Windows 8, Windows 7, and Windows Vista* operating systems. These operating systems will support up to 64 core processors while *Windows Server 2008* R2 supports up to 256 processors. Earlier versions of Windows (e.g., Windows 2000) supported two core processors. Thus, if there are two processors and *Windows 7* is the operating system, *CrimeStat* will calculate routines in about half the time. If there are four processors and *Windows Server 2008* is the operating system, *CrimeStat* will calculate routines in about a quarter of the time. The multiples are not exact since processing time must be allocated for input of data and output of tables. Also, some of the routines (the Markov Chain Monte Carlo regression models, the temporal modeling module) are sequential so that the advantages of multi-threading will not play much of a part.

For small data sets, this feature is not important as most runs will be very quick. However, for large data sets (e.g., 3000 cases or larger), the speed of calculations become important. For example, on a 1.6 GHz single-processor *Pentium M®* computer with 1 GB of RAM running *Windows XP Professional*, it took about 4 minutes to complete a nearest neighbor analysis on 14,853 cases involving the calculating of distance from every point to every other point and identifying the 100 nearest neighbors. On a 2.4 GHz dual-processor *Intel® Core™ 2*

computer with 4 GB of RAM running a 64-bit *Windows 7*, it took about 50 seconds to complete the same task. On a 2.4 GHz quad-processor *Intel® Core™ i7-2760QM* processor with 16 GB of RAM and running a 64-bit *Windows 7*, the task took 16 seconds, more than three times faster than the 'Core 2' processor and 60 times faster than the 'Pentium' processor. The larger the file that is being processed, the more critical becomes the calculating efficiency of the computer.

If a police department is expecting to run large data sets, it would benefit them to purchase fast multiple-processor computers with lots of RAM and fast hard disks to speed calculating times. The evolution of new processors is moving in this direction anyway so that multi-processor computers have become the norm.

Required Software

CrimeStat needs a Windows environment to operate. The program was designed for a *Windows 7* operating system so it is better optimized for that system. In particular, *Windows 7* and *Windows 8* have two features that allows *CrimeStat* to run more efficiently. First, they are multi-threading operating systems and can utilize multiple processors, as mentioned above. Second, they address memory in a more efficient way, as a large flat block. The 64 bit version of Windows 7 or Windows 8 in particular, will handle larger data blocks (called *words*) than the older 32 bit versions of Windows 7 and earlier operating systems.

CrimeStat is a stand-alone program. Hence, it does not require any other program other than a Windows operating system. However, to be maximally useful, there should be an accompanying GIS program. While point data can be obtained from a non-GIS system (e.g., census files include lat/lon coordinates for the centroid of census units), the use of the GIS to assign the coordinates is almost necessary. Further, many of the outputs of *CrimeStat* are for GIS programs. For example, to view an ellipse of a hot spot or to view a three dimensional interpolation produced by *CrimeStat* will require an appropriate GIS package.

Installing the Program

CrimeStat comes compressed in a zipped file called *CrimeStat.zip*. To install the program, it is necessary to have a compression program that recognizes the 'zip' format:

1. Create a directory using *Windows Explorer* and copy the file to that directory.

2. Double click on the file name in *Explorer*. When the name *CrimeStat.zip* is visible in the dialog box name field, double click the name with the left mouse button and point the extraction to the directory that you defined. *CrimeStat* will be installed in that directory.

3. The program help menu can also access the manual if the chapters of the manual are kept in the same directory as the program.

Adding an Item to the Start Menu

To add *CrimeStat* to the start menu:

1. Click on the *Start* button in Windows followed by *Settings* then *Taskbar*. Click on *Start Menu Programs* followed by *Add*.

2. In the dialog box, click on *Browse*, point to the directory where *CrimeStat* resides, and click on its name followed by *Open*. When the name *CrimeStat* is in the dialog box name field, click on the *Next* button.

3. Double-click on the folder to which *CrimeStat* is to be assigned.

4. Finally, type a name for *CrimeStat* (e.g., CrimeStat) followed by *Finish*.

Adding an Icon to the Desktop

To add *CrimeStat* to the desktop:

1. Double-click on *My Computer*.

2. Double-click on the drive in which *CrimeStat* resides followed by the directory that it is in (it may be several levels down).

3. Click once on the name *CrimeStat* with the left button and then hold down the right mouse button.

4. While holding the right mouse button, scroll to *Create Shortcut*.

5. The name *Shortcut to CrimeStat* will be placed at the end of the list of files.

6. Highlight the name by clicking on it once. Hold the left mouse button down and drag this name on to the desktop.

7. You can rename it *CrimeStat* by clicking on its icon with the right mouse button followed by *Rename*.

8. Alternatively, you can use *Windows Explorer* to create a shortcut and then drag the shortcut to the desktop.

Installing the Sample Data Sets

There are eight sample data sets that can be used to run the program, also in 'zip' format plus notes defining the variables. Since the data are ***simulated***, they should not be used for real applications.[1] They are provided to allow a user to become familiar with the program quickly. Many of these data sets have *Read Me* files that explain their data structure. However, ultimately, the value of the program must be tested on real data, rather than simulated data.

1. **General Sample Data.zip**. The data are simulated incident points from Baltimore City and Baltimore County in Maryland.

   A. *Incident.dbf* - A simulated data set of 1061 incidents (e.g., robberies) in Baltimore County and the City of Baltimore

   B. *Baltpop.dbf* - The 1990 population, area and population density of 1349 block groups in Baltimore County and the City of Baltimore

2. **Jtc Sample Data.zip**. There are three files of simulated data for use with the Journey-to-crime routine (Chapter 13). The data should be stored in the same directory:

   A. *JtcTest1.dbf* - A simulated data set of 2000 robberies in Baltimore County that can be used for calibrating a travel demand function. Each record has a crime location and a residence location of the offender.

   B. *JtcTest2.dbf* - A simulated data set of 2500 burglaries in Baltimore County that can be used for calibrating a travel demand function. Each record has a crime location and a residence location of the offender.

[1] The data were simulated by a random number generator following the distribution of several types of crime incidents. Because the data were selected by a random generator, the points do not necessarily fall on streets or even stay within the boundaries of the jurisdiction. Their purpose is to provide a simple data set so users can become familiar with the program and should not be used for actual research.

C. *Serial1.dbf* - A simulated data set of the location of seven incidents committed by a single serial offender. To become familiar with the journey to crime routine, they can be treated as either robberies or burglaries.

3. **Bayesian Jtc Sample Data.zip**. There are six files of simulated data for use with the Bayesian Journey-to-crime routine (Chapter 14). The data should be stored in the same directory:

A. Bayesian_calibration_file.dbf – A simulated data set of 963 crimes committed by 88 serial offenders. Each record has an offender ID, the UCR code, and the crime location and residence location of the offender.

B. Observed_OD_Distribution.dbf – A simulated matrix of crime trips from 533 origin zones in Baltimore County (MD) and the City of Baltimore (MD) to 325 destination zones in Baltimore County. Each record includes the location of the origin zone, the location of the destination zone, and the number of crime 'trips' for each combination.

C. Jtcfull.txt – A journey-to-crime calibration file that can be used to estimate the travel distance of offenders from each origin zone to each destination zone.

D. S14A.dbf – the crime locations of an offender who committed 14 offences before being caught. Each record includes the UCR code and the crime location and residence location of the offender.

E. TS15A.dbf – the crime locations of an offender who committed 15 offences before being caught. Each record includes the UCR code, the date, the time, and the crime location and residence location of the offender.

F. Test_Bayesian_Jtc_routine.param – a *CrimeStat* parameters file for loading these data into *CrimeStat* to run the routine.

4. **Correlated Walk Analysis Sample Data.zip**. These are three files of simulated data for use with the Correlated Walk Analysis routine (Chapter 12):

A. *PredictableOffender1.dbf* - A simulated data set for an algorithmic offender who committed 13 incidents.

B. *PredictableOffender2.dbf* -A simulated data set for an algorithmic offender who committed 12 incidents.

C. *RealOffender1.dbf* - A data set for a real offender who committed 12 incidents - 10 larceny thefts, 1 robbery and 1 burglary.

D. *RealOffender2.dbf* - A data set for a real offender who committed 15 incidents - 10 larceny thefts, 2 assaults, 2 burglaries and 1 robbery.

5. **Crime Travel Demand Sample Data.zip**. There are 13 files of data, CrimeStat parameter files, and a spreadsheet file for modeling travel behavior in Baltimore County, Md. They are examples used in the crime travel demand module (Chapters 25-32):

A. *Crime Travel Demand read me.pdf* - a file that explains the three data sets and their fields and describes the eight parameter files.

B. *BCOrigins.dbf* - a data set on 532 origin zones in both Baltimore County and the City of Baltimore from the late 1990s. There are data on crimes originating from each zone and demographic, economic and land use variables associated with those zones.

C. *BCDestinations.dbf* - a data set of 325 destination zones in Baltimore County only. There are data on crimes occurring in each zone and demographic, economic and land use variables associated with those zones.

D. *ObservedODTrips.dbf* - the actual trip distribution indicating the number of trips from each origin zone to each destination zone.

E. *Trip generation origin model.param* - Runs trip generation model using the Poisson regression for the origin zones affecting Baltimore County.

F. *Trip generation destination model.param* – Runs trip generation model for Baltimore County destinations.

G. *Make predicted origins.param* – Applies modeled coefficients for the origin model to the same data set from which it was modeled. Then the routine adds in external trips.

H. *Balance Origins and Destinations.param* – Balances the number of trips by origin and by destination. In the example, the number of predicted destinations are held constant.

I. *Calibrate Origin-Destination Model Coefficients.param* – Using the predicted origins and predicted destinations from the trip generation stage, estimates coefficients for distributing trips from origin zones to destination zones.

J. *Apply Origin-Destination Model.param* – Inputs the predicted origins and predicted destinations from the trip generation stage as well as the modeled coefficients from **H** above. Outputs predicted trips for each origin-destination zone combination. For the graphic display, outputs top 200 trips.

K. *Compare Observed and Predicted Trip Lengths.param* – Inputs observed (actual) and predicted trip distribution and compares them by trip lengths. Calculates coincidence ratio and then compares the top 200 origin-destination links.

L. *Mode Split Model.param* – Inputs predicted origins, predicted destinations, and predicted trips along with estimates of the mode split function (see Excel spreadsheet below). Splits trips by origin-destination pair into specific travel modes. The output is both a table of origin-destination trips by mode as well as five *ArcGis* shape files representing zone-to-zone trips by mode.

M. *Mode split impedance defaults.xls* - An Excel spreadsheet for estimating the coefficients of the mode split stage. This should be used in establishing the parameters for the mode split routine.

6. **Mode Split Impedance Defaults.xls**. A spreadsheet for estimating the parameters of the mode split impedance function (Chapter 30).

7. **Discrete Choice Modeling Sample Data.zip**. There are two files for running the multinomial logit model (Chapters 21 and 22):

A. *HoustonWeaponUse.dbf* - A data set of weapon use during robberies in Houston ,TX. Each record contains an offender ID, a randomly assigned

crime location, the type of weapon used during the robbery (WEAPON) and 11 predictive variables. See the attachment to Chapter 22 for details.

B. *Run MNL model of Houston robberies.param* – A *CrimeStat* parameters file for loading the multinomial logit model from these data.

C. *TheHagueBurglars.dbf* – A file of 548 cleared burglaries from The Hague, Netherlands. The file contains information on the characteristics of the burglars and neighborhood identifiers. This will be combined with the data set on neighborhoods in The Hague.

D. *TheHagueNeighborhoods.dbf* – a file of 89 neighborhoods in The Hague for the years 1996-2001.The data set is for use an alternatives files in creating a data set for the conditional logit model. This will be combined with the data set on burglars in The Hague.

E. *TheHagueNeighborhoodsXBurglars.dbf* – this file is the result of matching the file *TheHagueBurglars.dbf* with *TheHagueNeighborhoods.dbf* using the 'Create dataset for conditional discrete choice model' routine under Discrete Choice I module. It is used to validate the results of combing TheHagueBurglars.dbf with TheHagueNeighborhoods.dbf files.

8. **Time Series Forecasting Sample Data.zip**. There are two files for running the exponential smoothing and prediction routines (Chapters 23 and 24):

A. *Weekly crimes by Tract.dbf* – A data set of weekly robberies by 140 census tracts in Pittsburgh, PA, between 1990 and 1999.

B. *Monthly crimes by Tract.dbf* – A data set of monthly robberies by 140 census tracts in Pittsburgh, PA, between 1990 and 1999.

Again, to repeat, many of these data are simulated. Though they are based on actual cases, the X and Y coordinates have been randomly assigned so that the real locations of the crime location or the offender's residence are hidden. They should be used only to learn how to run individual routines.

To install any of these sample data files, it is necessary to have a compression program that recognizes the 'zip' format:

1. Create a data directory using *Windows Explorer* and copy the files to that directory.

2. In *Windows Explorer*, double-click on its name and then follow the instructions.

Step-by-Step Instructions

This manual will go through the program step-by-step to address how it can be used by a crime mapping/analysis unit within a police department. Chapter 2 provides a quick guide for all the data definition and program routines and Chapter **3** provides detailed instructions on setting up data to run with *CrimeStat*. The statistical routines are described in parts II, III, IV, V, and VI. Part II presents a number of statistics for spatial description. Part III presents hot spot analysis techniques for both points and zones. Part IV presents a number of statistics for spatial modeling (called Spatial Modeling I) while part V presents multivariate tools for spatial modeling (called Spatial Modeling II). Finally, part VI presents a crime travel demand module. The different statistics are presented and detailed examples of each technique are shown.

Options

There is an option tab that allows the saving and loading of program parameters and the setting of colors for each of main headings: Data setup, Spatial description, Hot Spot Analysis, Spatial modeling I, Spatial Modeling II, and Crime Travel Demand. One can also output simulated data during the simulation runs; this will be explained in the appropriate section.

Short Applications

The manual also includes a number of applications conducted by other researchers and analysts. These are presented as one page sidebars at the end of each of the chapters. Most of these are from criminal justice. But, applications from other fields have also been included. The aim is to show the diversity of applications that researchers and analysts have used with the various routines in *CrimeStat*.

Online Help

In addition, there is on-line help for the program. There is a *Help* button that can be

pushed to access all the help items. In addition, the program has context-sensitive help. On any page or routine, clicking on the *Help* button at the bottom of the screen will pop up an appropriate help item. The on-line help can also access the program manual. For this to be available, be sure to store the chapters of the manual in the *same directory* as the program.

Accessing the Help Menu in Windows

CrimeStat IV works with the Windows Vista, Windows 7, Windows Server 2008 R2, and Windows 8 operating systems. However, these operating systems do not include the help menu file that was available in previous versions of Windows (WinHlp32.exe) and clicking on the CrimeStat help button may not work. *If* you have this problem, Microsoft has developed a special file that allows help menus to be viewed. It will be necessary to obtain the file and install it. The URL is found at:

http://support.microsoft.com/kb/917607

Select the file that is appropriate for your operating system and follow the instructions on the page. Additional diagnostic information is also provided on the page.

References

BCPD (2012). *Baltimore County Police Crime Statistics and More*. Baltimore County Police Department: Towson, MD. http://www.baltimorecountymd.gov/Agencies/police/crime/index.html

Caliper (2012). *Maptitude: Geographic Information Systems*. Caliper: Newton, MA. http://www.caliper.com/maptovu.htm.

ESRI (2012a). *ArcGIS: Mapping & Analysis for Understanding Our World*. Environmental Systems Research Institute: Redlands, CA. http://www.esri.com/software/arcgis/index.html

ESRI (2012b). *ArcGIS Spatial Analyst*. Environmental Systems Research Institute: Redlands, CA. http://www.esri.com/software/arcgis/extensions/spatialanalyst/.

Golden Software (2012). *Surfer® 10*. Golden Software: Golden, CO. http://www.goldensoftware.com/products/surfer/surfer.shtml

Microsoft (2010). *Word 2010*. Microsoft: Redmond, WA. http://office.microsoft.com/en-us/word/.

Pitney Bowes (2012). *MapInfo Professional*. Pitney Bowes Software, Inc: Troy, NY. http://www.pbinsight.com/products/location-intelligence/applications/mapping-analytical/mapinfo-professional/.

Rockware (2012). *Vertical Mapper*. Rockware, Inc: Golden, CO.

Chapter 2:

Quickguide to *CrimeStat IV*

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 2:
Quickguide to *CrimeStat IV*

Introduction

The following are brief instructions for the use of *CrimeStat® IV* and parallels the online help menus in the program. Because there are a large number of routines in *CrimeStat*, this quickguide is very long. Detailed instructions on individual routines should be obtained from Chapters 3-32 in the documentation.

CrimeStat has five basic groupings in 27 program tabs and one option tab. Each tab lists routines, options and parameters:

Data setup

1. Primary File
2. Secondary File
3. Reference File
4. Measurement Parameters

Spatial description

5. Spatial Distribution
6. Spatial Autocorrelation
7. Distance Analysis I
8. Distance Analysis II

Hot spot analysis

9. Hot Spot Analysis I
10. Hot Spot Analysis II
11. Hot Spot Analysis of Zones

Spatial modeling I

12. Interpolation I
13. Interpolation II
14. Space-time Analysis

15. Journey-to-crime
16. Bayesian Journey-to-crime

Spatial modeling II

17. Regression I
18. Regression II
19. Discrete Choice I
20. Discrete Choice II
21. Time Series Forecasting

Crime Travel Demand

22. Project Directory
23. Trip Generation
24. Trip Distribution
25. Mode Split
26. Network Assignment
27. File Worksheet

Options

28. Saving parameters, colors and options

Throughout this chapter, figures 2.1-2.28 show the 28 tab screens with examples of data input and routine selection.

I. Data Setup

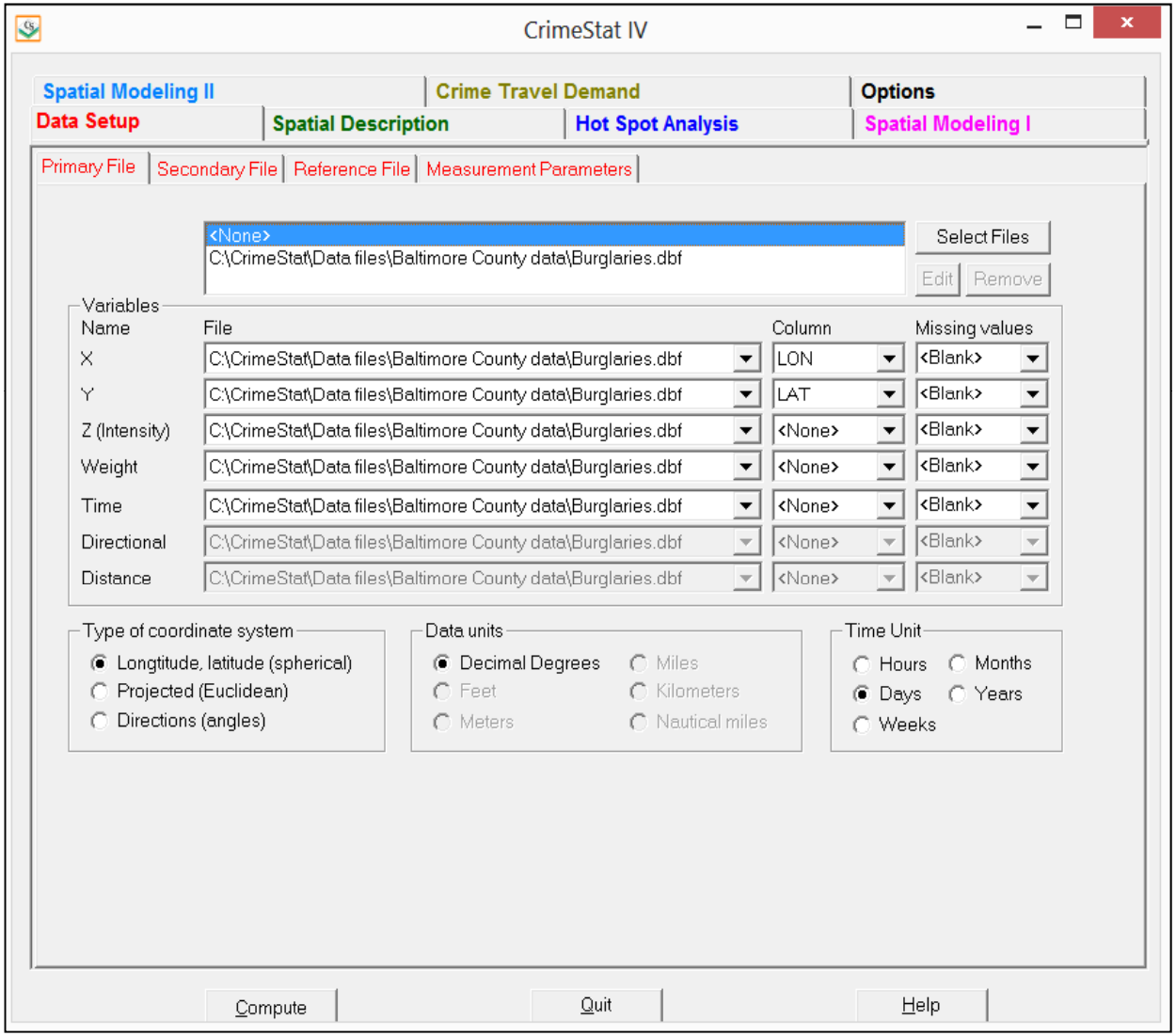
The data setup section involves defining the data set and variables for a primary file (required) and a secondary file (optional), identifying a reference grid (required for several routines), and defining measurement parameters (required for several routines).

Primary File

A primary file is required for *CrimeStat*. It is a point file with X and Y coordinates. For example, the primary file could be the location of street robberies, each of which have an

Figure 2.1:

Primary File Setup



associated X and Y coordinates. Also, there can be associated weights or intensities, though these are optional. Also, there can be time references, though these are optional. For example, if the points are the locations of police stations, then the intensity variable could be the number of calls for service at each police station while the weighting variable could be service zones. More than one file can be selected. The time references are used in the space-time analysis routines are defined in terms of hours, days, weeks, months, or years.

Select Files

Select the primary file. *CrimeStat* reads dbase 'dbf', ArcGIS point 'shp' and ASCII files. Select the type of file to be selected. Use the browse button to search for a particular file name. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns. Note that there is a utility that will convert an Excel 'xls' or 'xlsx' to a 'dbf' file on the Options tab.

Variables

Define the file that contains the X and Y coordinates. If there are weights or intensities being used, define the file that contains these variables. Certain statistics (e.g., spatial autocorrelation, local Moran) require intensity values and most other statistics can use intensity values. Most other statistics can use weights. It is possible to have both an intensity variable and a weighting variable, though the user should be cautious in doing this to avoid 'double weighting'. If a time variable is used, it must be an integer or real number (e.g., 1, 36892). Do not use formatted dates (e.g., 01/01/2001, October 1, 2001). Convert these to real numbers before using the space-time analysis routines.

Columns

Select the variables for the X and Y coordinates respectively (e.g., Lon, Lat, Xcoord, Ycoord.) If weights or intensities are being used, select the appropriate variable names. If a time variable is used, select the appropriate variable name.

Missing Values

Identify whether there are any missing values. By default, *CrimeStat* will ignore records with blank values in any of the eligible fields or records with non-numeric values (e.g., alphanumeric characters, , *). Blanks will always be excluded unless the user selects ***<none>***. There are 8 possible options:

1. <*blank*> fields are automatically excluded. This is the default
2. <*none*> indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0
3. *0* is excluded
4. *–1* is excluded
5. *0 and –1* indicates that both 0 and -1 will be excluded
6. *0, -1 and 9999* indicates that all three values (0, -1, 9999) will be excluded
7. *Any* other numerical value can be treated as a missing value by typing it (e.g., 99)
8. *Multiple* numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99)

Directional

If the file contains directional coordinates (angles), define the file name and variable name (column) that contains the directional measurements. If directional coordinates are being used, there can be an optional distance variable for the measurement. Define the file name and variable name (column) that contains the distance variable.

Time Units

Define the units for the time variable and are defined in terms of hours, days, weeks, months, or years. Time is only used for the primary file. The default value is days. Note, only integer or real numbers can be used (e.g., 1, 36892). Do not use formatted dates (e.g., 01/01/2001, October 1, 2001). Convert these to real or integer numbers before using the space-time analysis routines.

Type of Coordinate System and Data Units

Select the type of coordinate system. If the coordinates are longitudes and latitudes, then a spherical system is being used and data units will automatically be decimal degrees. If the coordinate system is projected (e.g., State Plane, Universal Transverse Mercator – UTM), then data units could be feet (e.g., State Plane), meters (e.g., UTM.), miles, kilometers, or nautical miles**.** If the coordinate system is directional, then the coordinates are angles and the data units box will be blanked out. For directions, an additional distance variable can be used. This measures the distance of the incident from an origin location; the units are undefined.

Note: if a projected coordinate system is used, but the coordinate system is defined as longitude/latitude (spherical), an error message will appear that says "Found invalid data at row 1 of the primary data set!". Change the coordinate system to Projected (Euclidean).

Secondary File

A secondary data file is optional. It is also a point file with X and Y coordinates. It is usually used in comparison with the primary file. There can be weights or intensities variables associated, though these are optional. For example, if the primary file is the location of motor vehicle thefts, the secondary file could be the centroid of census block groups that have the population of the block group as the intensity (or weight) variable. In this case, one could compare the distribution of motor vehicle thefts with the distribution of population in, for example, the Ripley's "K" routine or the dual kernel density estimation routine. More than one file can be selected. Time units are not used in the secondary file.

Select Files

Select the secondary file. *CrimeStat* reads dbase 'dbf', ArcGIS point 'shp' and ASCII files. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns. Note that there is a utility that will convert an Excel 'xls' or 'xlsx' to a 'dbf' file on the Options tab.

Variables

Define the file that contains the X and Y coordinates. If weights or intensities are being used, define the file that contains these variables. Certain statistics (e.g., spatial autocorrelation, local Moran) require intensity values and most other statistics can use intensity values. Most other statistics can use weights. It is possible to have both an intensity variable and a weighting variable, though the user should be cautious in doing this to avoid 'double weighting'. Time units are not used in the secondary file.

Columns

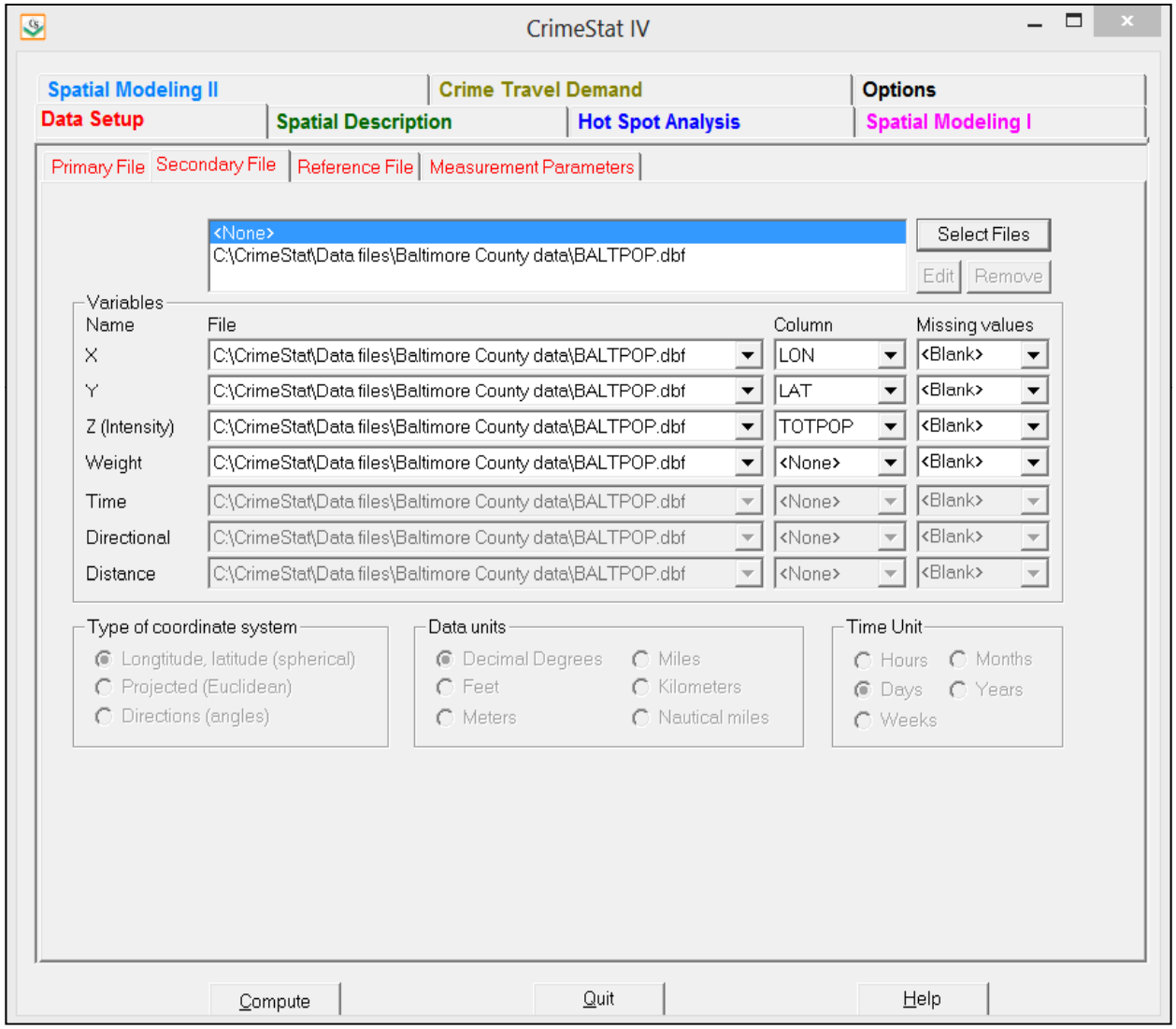
Select the variables for the X and Y coordinates respectively (e.g., Lon, Lat, Xcoord, Ycoord.) If there are weights or intensities being used, select the appropriate variable names. Time units are not used in the secondary file.

Missing Values

Identify whether there are any missing values. By default, *CrimeStat* will ignore records with blank values in any of the eligible fields or records with non-numeric values

Figure 2.2:

Secondary File Setup



(e.g., *, alphanumeric characters , *). Blanks will always be excluded unless the user selects ***<none>***. There are 8 possible options:

1. <*blank*> fields are automatically excluded. This is the default
2. <*none*> indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0
3. *0* is excluded
4. *–1* is excluded
5. *0 and –1* indicates that both 0 and -1 will be excluded
6. *0, -1 and 9999* indicates that all three values (0, -1, 9999) will be excluded
7. *Any* other numerical value can be treated as a missing value by typing it (e.g., 99)
8. *Multiple* numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99)

Type of Coordinate System and Data Units

The secondary file must have the same coordinate system and data units as the primary file. This selection will be blanked out, indicating that the secondary file carries the same definition as the primary file. Directional coordinates (angles) are not allowed for the secondary file nor are time variables.

Reference File

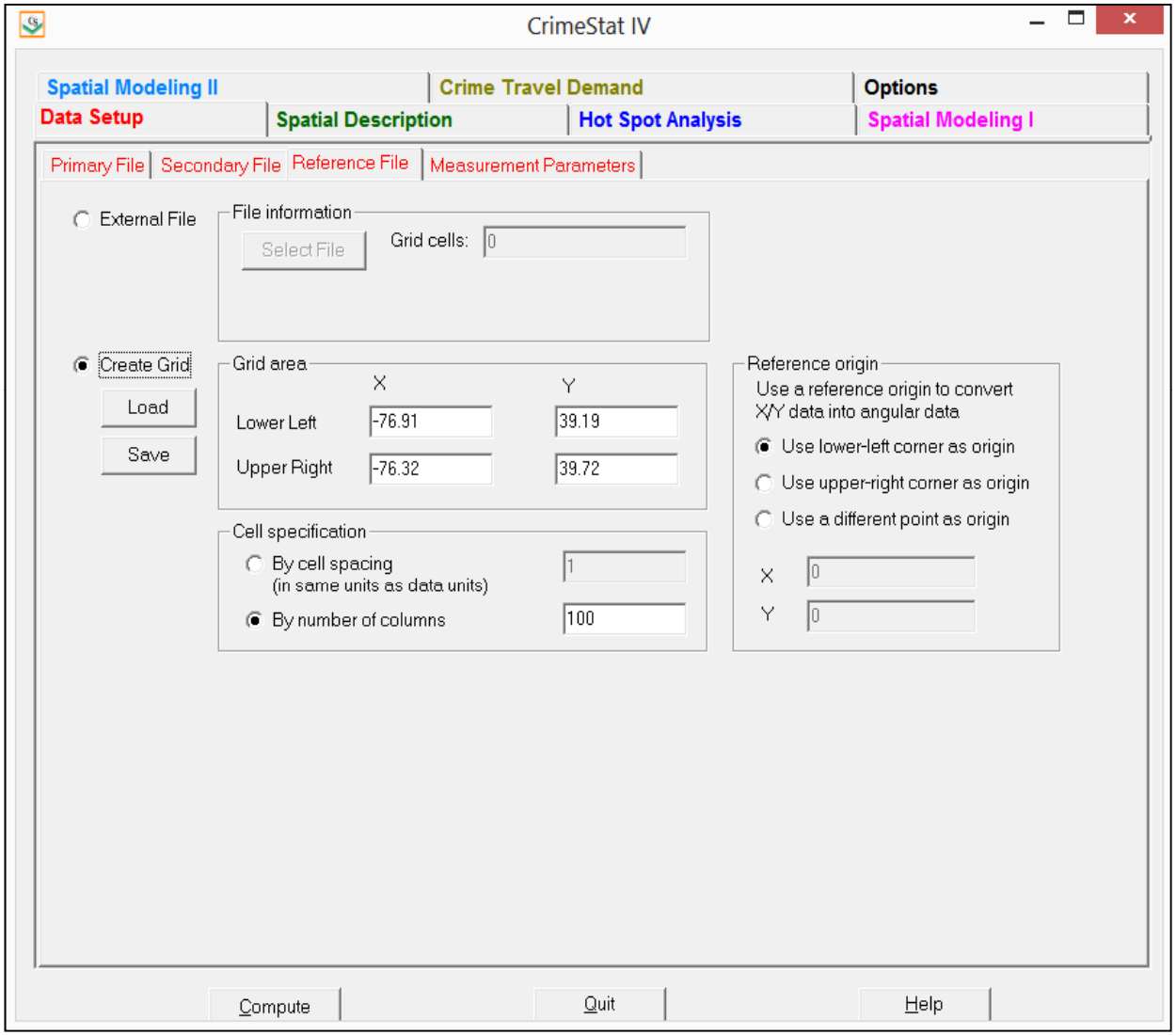
For referencing the study area, there is a reference grid, a reference origin, and an area. The reference file is used in the risk-adjusted nearest neighbor hierarchical clustering routine, journey-to-crime estimation and in the single and dual variable kernel density estimation routines. The file can be an external file that is input or can be generated by *CrimeStat*. It is usually, though not always, a grid which is overlaid on the study area. The reference origin is used in the directional mean routine. The file can be an external file that is input or can be generated by *CrimeStat*. The area is that of the study region.

Create Reference Grid

If allowing *CrimeStat* to generate a true grid, click on 'generated' and then input the lower left and upper right X and Y coordinates of a rectangle placed over the study area. Cells can be defined either by cell size, in the same coordinates and data units as the primary file, or by the number of columns in the grid (the default). In addition, a reference origin can be defined for the directional mean routine. The reference grid can be saved and re-used. Click on 'Save' and enter a file name. To use an already saved file, click on 'Load' and the file name.

Figure 2.3:

Reference File Setup



The coordinates are saved in the registry, but can be re-saved in any directory. With the Load screen open, click on 'Save to file' and then enter a directory and a file name. The default file extension is 'ref.

External Reference File

If an external file that stores the coordinates of each grid cell is to be used, select the name of the reference file. *CrimeStat* reads dbase 'dbf', ArcGIS point 'shp' and ASCII files. Select the type of file to be selected. Use the browse button to search for the file. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns. A reference file that is read into *CrimeStat* need not be a true grid (a matrix with *k* columns and *l* rows.) However, a reference file that is read in can only be output to *Surfer for Windows* since the other output formats – *ArcGIS*, *MapInfo*, , *ArcGIS Spatial Analyst*, and ASCII grid require the reference file to be a true grid.

Reference Origin

A reference origin can be defined for the directional mean routine. The reference origin can be assigned to:

1. Use the lower-left corner defined by the minimum X and Y values. This is the default
2. Use the upper-right corner defined by the maximum X and Y values
3. Use a different origin point. With the later, the user must define the origin

Measurement Parameters

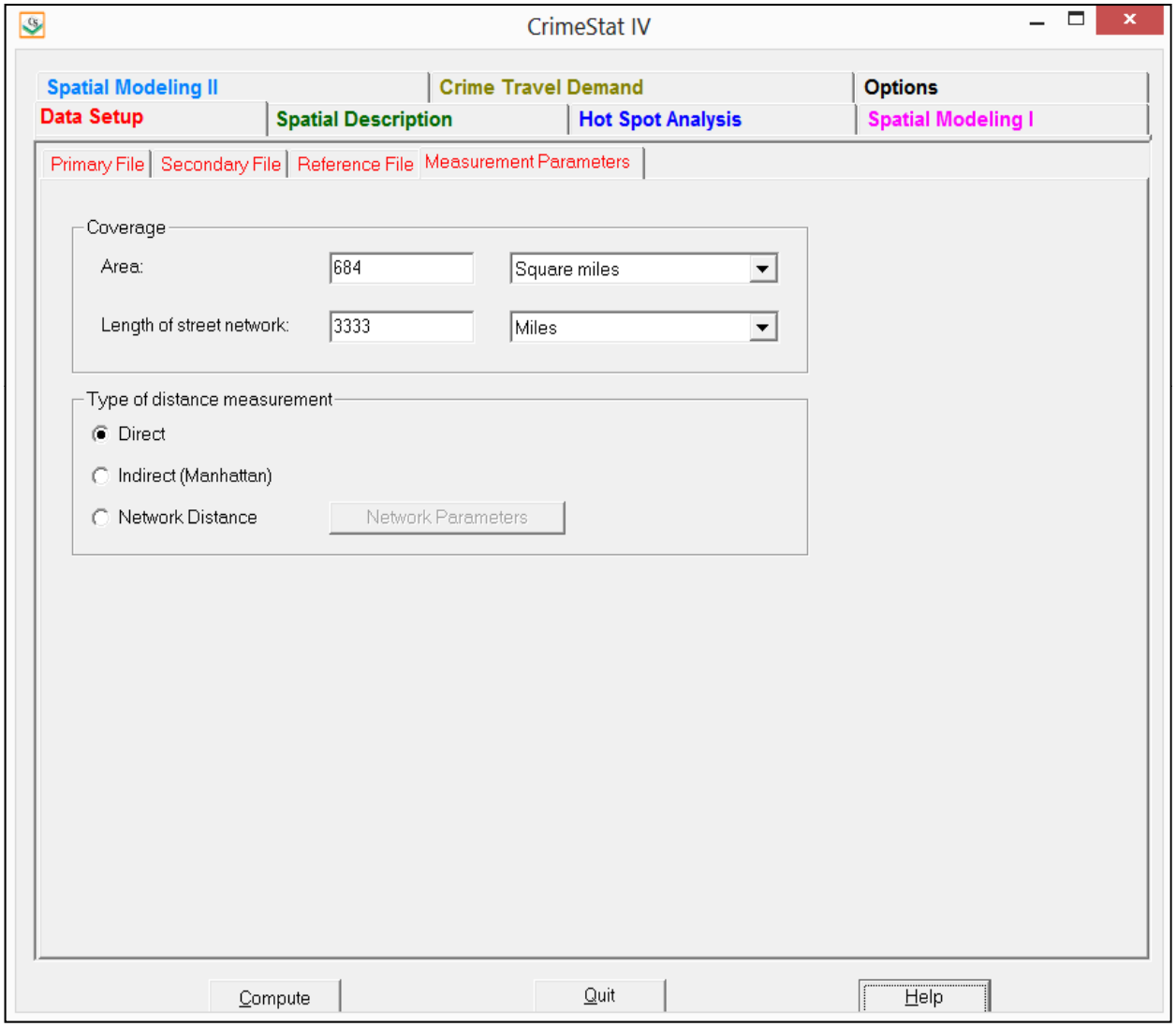
The measurement parameters page defines the measurement units of the coverage and the type of distance measurement to be used. There are three components that are defined:

Area

First, define the geographical area of the study area in area units (square miles, square nautical miles, square feet, square kilometers, square meters.) Irrespective of the data units that are defined for the primary file, *CrimeStat* can convert to various area measurement units. These units are used in the nearest neighbor, Ripley's "K", nearest neighbor hierarchical clustering, risk-adjusted nearest neighbor hierarchical clustering, Stac, and K-means clustering routines.

Figure 2.4:

Measurement Parameters Setup



If no area units are defined, then *CrimeStat* will define a rectangle by the minimum and maximum X and Y coordinates.

Length of Street Network

Second, define the total length of the street network within the study area or an appropriate comparison network (e.g., freeway system) in distance units (miles, nautical miles, feet, kilometers, or meters). The length of the street network is used in the linear nearest neighbor routine. Irrespective of the data units that are defined for the primary file, *CrimeStat* can convert to distance measurement units. The distance units should be in the same metric as the area units (e.g., miles and square miles/meters and square meters.)

Type of Distance Measurement

Third, define how distances are to be calculated. There are three choices:

1. Direct distance
2. Indirect (Manhattan) distance
3. Network distance

Direct

If direct distances are used, each distance is calculated as the shortest distance between two points. If the coordinates are spherical (i.e., latitude, longitude), then the shortest direct distance is a 'Great Circle' arc on a sphere. If the coordinates are projected, then the shortest direct distance is a straight line on a Euclidean plane.

Indirect

If indirect distances are used, each distance is calculated as the shortest distance between two points on a grid, that is with distance being constrained to the horizontal or vertical directions (i.e., not diagonal.) This is sometimes called 'Manhattan' metric. If the coordinates are spherical (i.e., latitude, longitude), then the shortest indirect distance is a modified right angle on a spherical right triangle. If the coordinates are projected, then the shortest indirect distance is the right angle of a right triangle on a two-dimensional plane

Network distance

If network distances are used, each distance is calculated as the shortest path between two points using the network. Alternatives to distance can be used including speed, travel time, or travel cost. Click on 'Network parameters' and identify a network file.

Type of network

Network files can *bi-directiona*l (e.g., a TIGER file) or *single directional* (e.g., a transportation modeling file). In a bi-directional file, travel can be in either direction. In a single directional file, travel is only in one direction. Specify the type of network to be used.

Network input file

The network file can either be a shape file (line, polyline, or polylineZ file) or another file, either a dBase IV 'dbf', ArcGIS 'shp' or ASCII file . The default is a shape file. If the file is a shape file, the routine will know the locations of the nodes. For a dBase IV or other file, the X and Y coordinate variables of the end nodes must be defined. These are called the "From" node and the "End" node. An optional weight variable is allowed for all types of file0073. The routine identifies nodes and segments and finds the shortest path. If there are one-way streets in a bi-directional file, the flag fields for the "From" and "To" nodes should be defined.

Network weight field

Normally, each segment in the network is not weighted. In this case, the routine calculates the shortest distance between two points using the distance of each segment. However, each segment can be weighted by travel time, speed or travel costs. If travel time is used for weighting the segment, the routine calculates the shortest time for any route between two points. If speed is used for weighting the segment, the routine converts this into travel time by dividing the distance by the speed. Finally, if travel cost is used for weighting the segment, the routine calculates the route with the smallest total travel cost. Specify the weighting field to be used and be sure to indicate the measurement units (distance, speed, travel time, or travel cost) at the bottom of the page. If there is no weighting field assigned, then the routine will calculate using distance.

From one-way flag and To one-way flag

One-way segments can be identified in a bi-directional file by a 'flag' field (it is not necessary in a single directional file). The 'flag' is a field for the end nodes of the segment with

values of ‘0’ and ‘1’. A ‘0’ indicates that travel can pass through that node in either direction whereas a ‘1’ indicates that travel can only pass from the other node of the same segment (i.e., travel cannot occur from another segment that is connected to the node). The default assumption is for travel to be allowed through each node (i.e., there is a ‘0’ assumed for each node). For each one-way street, specify the flags for each end node. A ‘0’ allows travel from any connecting segments whereas a ‘1’ only allows travel from the other node of the same segment. Flag fields that are blank are assumed to allow travel to pass in either direction.

FromNode ID and ToNode ID

If the network is single directional, there are individual segments for each direction. Two-way streets have two segments, one for each direction. On the other hand, one-way streets have only one segment. The FromNode ID and the ToNode ID identify from which end of the segment travel should occur. If no FromNode ID and ToNode ID is defined, the routine will chose the first segment of a pair that it finds, whether travel is in the right or wrong direction. To identify correctly travel direction, define the FromNode and ToNode ID fields.

Network coordinate system

The type of coordinate system for the network is assumed to be the same as for the primary file.

Segment measurement unit

By default, the shortest path is in terms of distance. However, each segment can be weighted by travel time, travel speed, or travel cost.

1. For travel time, the units are minutes, hours, or unspecified cost units.
2. For speed, the units are miles per hour and kilometers per hour. In the case of speed as a weighting variable, it is automatically converted into travel time by dividing the distance of the segment by the speed, keeping units constant.
3. For travel cost, the units are undefined and the routine identifies routes by those with the smallest total cost.

II. Spatial Description

The spatial description section calculates spatial description, spatial autocorrelation, distance analysis, and hot spot statistics. The distance analysis and hot spot analysis statistics are on two separate tabs each.

Spatial Distribution

Spatial distribution provides statistics that describe the overall spatial distribution. These are sometimes called centrographic, global, or first-order spatial statistics. There are six routines for describing the spatial distribution. An intensity variable and a weighting variable can be used for the first five routines, though it is not required. An intensity variable *is* required for the two spatial autocorrelation routines; a weighting variable can also be used for the spatial autocorrelation indices. All outputs can be saved as text files. Some outputs can be saved as graphical objects for import into desktop GIS programs.

Mean Center and Standard Distance (Mcsd)

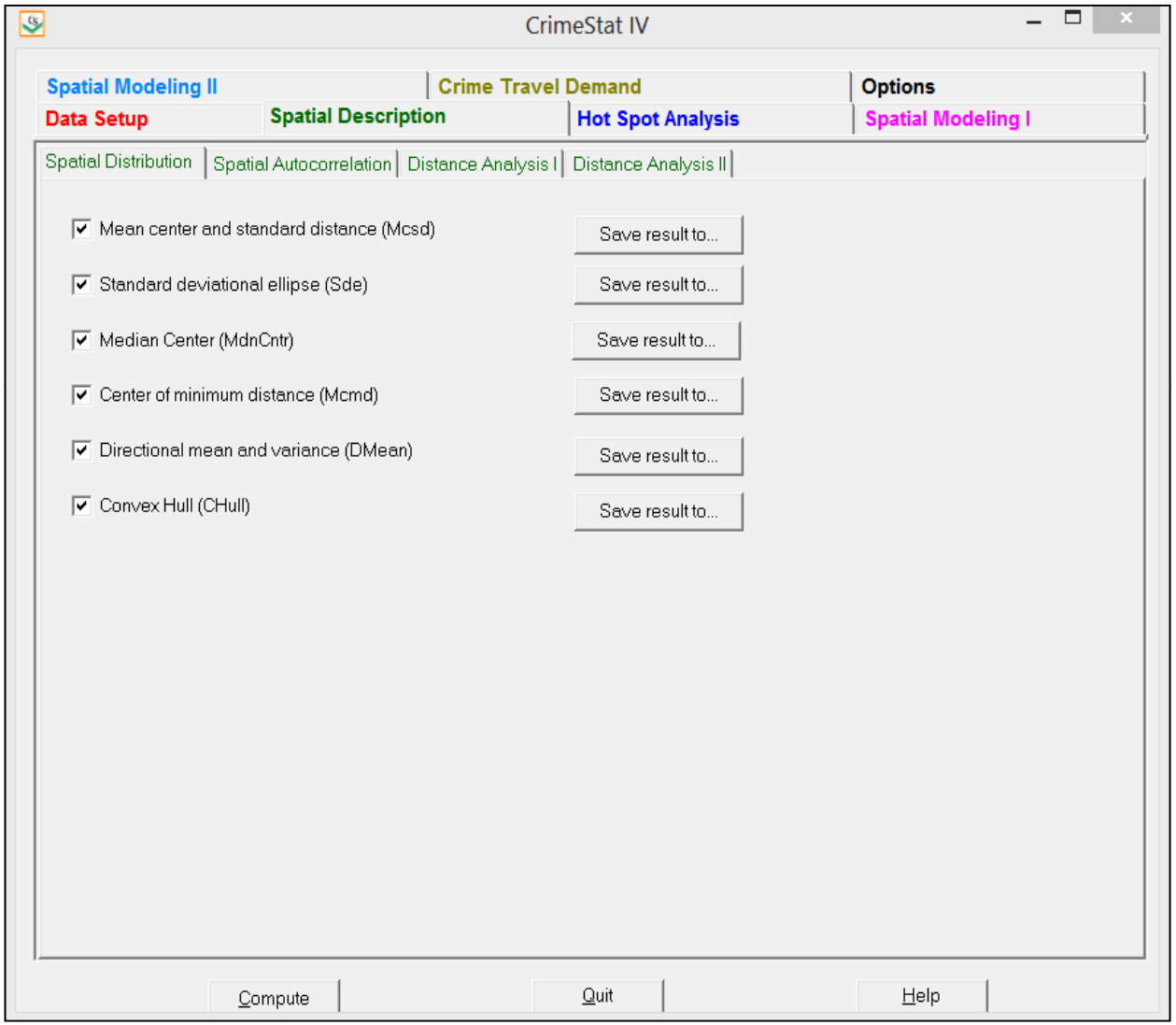
The mean center and standard distance define the arithmetic mean location and the degree of dispersion of the distribution. The Mcsd routine calculates 9 statistics:

1. The sample size
2. The minimum X and Y values
3. The maximum X and Y values
4. The X and Y coordinates of the mean center
5. The standard deviation of the X and Y coordinates
6. The X and Y coordinates of the geometric mean
7. The X and Y coordinates of the harmonic mean
8. The standard distance deviation, in meters, feet and miles. This is the standard deviation of the distance of each point from the mean center.
9. The circle area defined by the standard distance deviation, in square meters, square feet and square miles.

The tabular output can be printed and the mean center (mean X, mean Y), the geometric mean, the harmonic mean, the standard deviations of the X and Y coordinates, and the standard distance deviation can be output as graphical objects to ArcGIS ‘shp’, MapInfo ‘mif’, and *Google Earth* ‘kml’ (for spherical coordinates only) formats. A file name should be provided.

Figure 2.5:

Spatial Distribution Statistics



For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. If the coordinate system is spherical (longitude/latitude), then the file can be saved as a *Google Earth* 'kml' output.

The mean center is output as a point (MC<*file name*>.) The geometric mean is output as a point (GM<*file name*>.) The harmonic mean is output as a point (HM<file name>.) The standard deviation of both the X and Y coordinates is output as a rectangle (XYD<file name>.) The standard distance deviation is output as a circle (SDD<*file name*>.)

Standard Deviational Ellipse (Sde)

The standard deviational ellipse defines both the dispersion and the direction (orientation) of that dispersion. The Sde routine calculates 9 statistics:

1. The sample size
2. The clockwise angle of Y-axis rotation in degrees
3. The ratio of the long to the short axis after rotation
4. The standard deviation along the new X and Y axes in meters, feet and miles
5. The X and Y axes lengths in meters, feet and miles
6. The area of the ellipse defined by these axes in square meters, square feet and square miles
7. The standard deviation along the X and Y axes in meters, feet and miles for a 2X standard deviational ellipse
8. The X and Y axes lengths in meters, feet and miles for a 2X standard deviational ellipse
9. The area of the 2X ellipse defined by these axes in square meters, square feet and square miles.

The tabular output can be printed and the 1X and 2X standard deviational ellipses can be output as graphical objects to ArcGIS 'shp', MapInfo 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. A file name should be provided. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The 1X standard deviational ellipse is output as an ellipse (SDE<*file name*>.) The 2X standard deviational ellipse is output as an ellipse with axes that are twice as large as the 1X standard deviational ellipse (2SDE<*file name*>.)

Median Center (MdnCntr)

The median center is the point at which the median of the X coordinates intersects the median of the Y coordinates. The MdnCntr routine outputs 3 statistics:

1. The sample size
2. The median value of the X coordinate
3. The median value of the Y coordinate

The tabular output can be printed and the median center can be output as a graphical object to ArcGIS 'shp', MapInfo 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. A file name should be provided. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. The median center is output as a point (MndCntr<file name>.)

Center of Minimum Distance (Mcmd)

The center of minimum distance defines the point at which the distance to all other points is at a minimum. Unfortunately, it is sometimes also called the 'median center', but not to be confused with median center that is the intersection of the median of X and the median of Y (see above). The Mcmd routine outputs 5 statistics:

1. The sample size
2. The mean of the X and Y coordinates
3. The number of iterations required to identify a center of minimum distance
4. The degree of error (tolerance) for stopping the iterations
5. The X and Y coordinates which define the center of minimum distance

The tabular output can be printed and the center of minimum distance can be output as a graphical object to ArcGIS 'shp', MapInfo 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. A file name should be provided. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory

as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. The center of minimum distance is output as a point (Mcmd<file name>).

Directional Mean and Variance (Dmean)

The angular mean and variance are properties of angular measurements. The angular mean is an angle defined as a bearing from true North: 0 degrees. The directional variance is a relative indicator varying from 0 (no variance) to 1 (maximal variance.) Both the angular mean and the directional variance can be calculated either through angular (directional) coordinates or through X and Y coordinates.

Output with directional coordinates

If the primary file cases are directional coordinates (bearings/angles from 0 to 360 degrees), the angular mean is calculated directly from the angles. An optional distance variable can be included. In this case, the directional mean routine will output five statistics:

1. The sample size
2. The unweighted mean angle
3. The weighted mean angle
4. The unweighted circular variance
5. The weighted circular variance

Output with X and Y coordinates

On the other hand, if the primary file incidents are defined in X and Y coordinates, the angles are defined relative to the reference origin (see Reference file) and the angular mean is converted into an equation. In this case, the directional mean routine will output nine statistics:

1. The sample size
2. The unweighted mean angle
3. The weighted mean angle
4. The unweighted circular variance
5. The weighted circular variance
6. The mean distance
7. The intersection of the mean angle and the mean distance (directional mean)
8. The X and Y coordinates for the triangulated mean
9. The X and Y coordinates for the weighted triangulated mean

The directional mean and triangulated mean can be saved as an *ArcGIS 'shp', MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. A file name should be provided. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The unweighted directional mean - the intersection of the mean angle and the mean distance is output with the prefix 'Dm' while the unweighted triangulated mean location is output with a 'Tm' prefix. The weighted triangulated mean is output with a 'TmWt' prefix. The tabular output can be printed.

Convex Hull (Chull)

The convex hull draws a polygon around the outer points of the distribution. It is useful for viewing the shape of the distribution. The routine outputs three statistics:

1. The sample size
2. The number of points in the convex hull
3. The X and Y coordinates for each of the points in the convex hull

The convex hull can be saved as an *ArcGIS 'shp', MapInfo 'mif',* , various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files with a 'Chull' prefix. For *MapInfo* 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. The convex hull is output as a graphical object with no attributes associated with it (i.e., only a polygon that defines the convex hull).

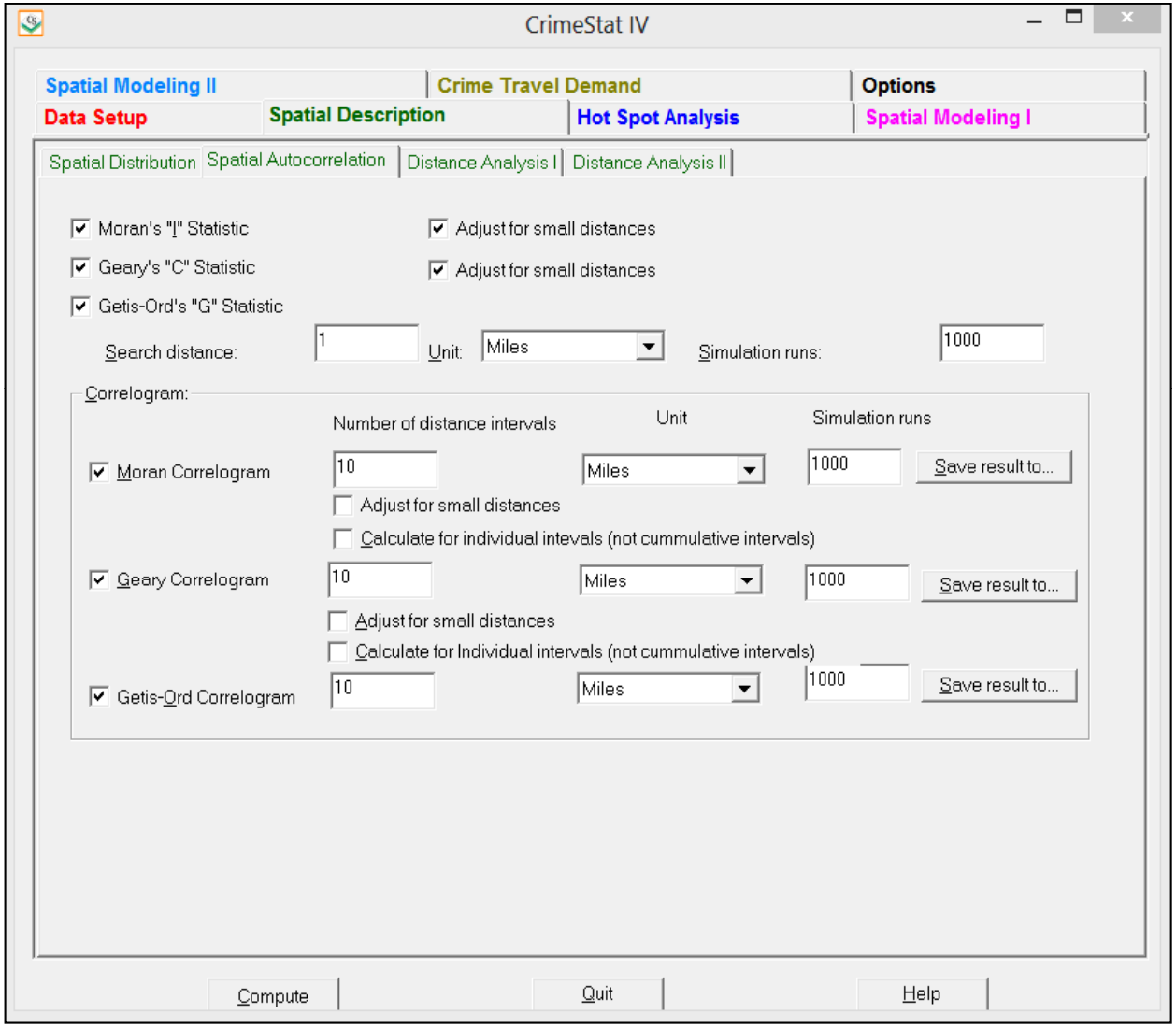
III. Spatial Autocorrelation

Spatial Autocorrelation Indices

Spatial autocorrelation indices identify whether point locations are spatially related, either clustered or dispersed. These indices would typically be applied to zonal data where an attribute value can be assigned to each zone. Six spatial autocorrelation indices are calculated. All **require** an intensity variable in the Primary File.

Figure 2.6:

Spatial Autocorrelation Statistics



Moran's "I"(MoranI)

Moran's "I" statistic is the classic indicator of spatial autocorrelation. It is an index of co-variation between different point locations and is similar to a product moment correlation coefficient, typically varying from –1 to +1 (though these are not absolute limits). A positive value indicates that there is positive spatial autocorrelation, that in general zones are nearby other zones with similar values (either high or low) while a negative value indicates negative spatial autocorrelation, that in general zones are nearby other zones with different values (either high values next to zones with low values, or the opposite). The "I" value is calculated with the intensity variable specified on the Primary File page.

Adjust for small distances

If this box is checked, small distances are adjusted so that the maximum weighting is 1. This ensures that "I" won't become excessively large for points that are very close together. The default value is no adjustment.

Moran's "I" Output

The Moran's "I" routine calculates 6 statistics:

1. The sample size
2. Moran's "I"
3. The spatially random (expected) "I"
4. The standard error of "I"
5. A significance test of "I" under the assumption of normality (Z-test)
6. A significance test of "I" under the assumption of randomization (Z-test)

Values of "I" greater than the expected I indicate clustering while values of "I" less than the expected I indicate dispersion. The significance test indicates whether these differences are greater than what would be expected by chance. The tabular output can be printed.

Geary's "C" (GearyC)

Geary's "C" statistic is an alternative indicator of spatial autocorrelation. It is an index of paired comparisons between different point locations and typically varies from 0 (similar values) to 2 (dissimilar values.) Theoretically, a value of +1 indicates spatial independence, that the values of one zone are unrelated to the values of nearby zones. Values less than +1 indicate positive spatial autocorrelation (zones have values similar to their neighbors) while values greater

than +1 indicate negative spatial autocorrelation (zones have values different to their neighbors). The “C” value is calculated with the intensity variable specified on the Primary File page

Adjust for small distances

If this box is checked, small distances are adjusted so that the maximum weighting is 1. This ensures that “C” won't become excessively large or excessively small for points that are close together. The default value is no adjustment.

Geary “C” Output

The Geary’s “C” routine calculates 8 statistics:

1. The sample size
2. Geary's "C"'
3. Adjusted “C” (1-“C”)
4. The spatial random (expected) "C"
5. The standard error of "C"
6. A significance test of "C" under the assumption of normality (Z-test)
7. The one-tail probability level
8. The two-tail probability level

Values of “C” that are less than the expected “C” indicate clustering while values of “C” that are greater than the expected “C” indicate dispersion. The significance test indicates whether these differences are greater than what would be expected by chance. The tabular output can be printed.

The adjusted “C” converts the statistic so that it varies between +1 and -1 and is similar to a Moran’s “I”. A positive value of the adjusted “C” indicates positive spatial autocorrelation while a negative value indicates negative spatial autocorrelation.

Getis-Ord General G (Getis-OrdG)

The Getis-Ord “G” statistic is an index of spatial autocorrelation for values of a variable that fall within a specified distance of each other (search distance). When compared to an expected value of G under the assumption of no spatial association, the statistic has the advantage over other global spatial autocorrelation measures (Moran, Geary) in that it can distinguish between ‘hot spots’ and ‘cold spots’. The “G” value is calculated with the intensity variable specified on the Primary File page and with respect to a specified search distance (user defined).

By itself, the G statistic is not very meaningful. The “G” value varies from 0 to 1 since it indicates the interaction of pairs of zones that are within the search distance relative to the interaction of all pairs of zones. As the search distance increases, this statistic will automatically approach 1.0. Consequently, G is compared to an expected value of G under the assumption of no significant spatial association.

Further, under the assumption that G is normally distributed, a Z-test can be constructed that tests for the significance of the actual G. A positive Z-value indicates spatial clustering of high values more than what would be expected under chance (hot spots) while a negative Z-value indicates spatial clustering of low values more than what would be expected under chance (cold spots). A “G” value around 0 typically indicates either no spatial autocorrelation at all or that the number of hot spots more or less balances the number of cold spots. The statistic requires an intensity variable in the primary file.

Search distance

The user must specify a search distance for the test and indicate the distance units (miles, nautical miles, feet, kilometers, or meters).

Getis-Ord “G” Output

The Getis-Ord “G” routine calculates 8 statistics:

1. The sample size
2. Getis-Ord “G”
3. The spatially random (expected) "G"
4. The difference between “G” and the expected “G”
5. The standard error of "G"
6. A Z-test of "G" under the assumption of normality (Z-test)
7. The one-tail probability level
8. The two-tail probability level

Simulation of confidence intervals

Since the Getis-Ord “G” statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a permutation type Monte Carlo simulation can be run to estimate approximate confidence intervals around the “G” value. Specify the number of simulations to be run (e.g., 100, 1000, 10000). In addition to the above statistics, a simulation includes the following statistics:

9. The minimum “G” value
10. The maximum “G” value
11. The 0.5 percentile of “G”
12. The 2.5 percentile of “G”
13. The 5 percentile of “G”
14. The 10 percentile of “G”
15. The 90 percentile of “G”
16. The 95 percentile of “G”
17. The 97.5 percentile of “G”
18. The 99.5 percentile of “G”

The four pairs of percentiles (10 and 90; 5 and 95; 2.5 and 97.5; 0.5 and 99.5) create approximate 80%, 90%, 95% and 99% confidence intervals respectively. The tabular results can be printed or saved to a text file.

Moran Correlogram

The Moran Correlogram calculates the Moran’s “I” index for different distance intervals/bins (not adjusted for small distances). The “I” value typically varies between -1 and +1 though these are not absolute limits. An “I” value of 0 indicates no spatial autocorrelation. An “I” value greater than 0 indicates positive spatial autocorrelation (zones have values similar to their neighbors) while an “I” value less than 0 indicates negative spatial autocorrelation (zones have values different from their neighbors).

The Moran Correlogram calculates these “I” values as a function of distance. The user can select any number of distance intervals. The default is 10 distance intervals. The “I” value for each distance interval is calculated with the intensity variable specified on the Primary File page.

Adjust for small distances

If the item is checked, small distances are adjusted so that the maximum weighting is 1. This ensures that the “I” values for individual distances won't become excessively large or excessively small for points that are close together. The default value is no adjustment.

Calculate for individual intervals

By default, the Moran Correlogram routine calculates the “I” values for the cumulative distance from 0 to the end of the interval. If the user checks the box to ‘Calculate for individual intervals’, then the “I” values for only those pairs of points that fall within the interval are

calculated. This can be useful for checking the spatial autocorrelation for a specific interval or checking whether some distances don't have sufficient numbers of points (in which case the "I" value will be unreliable).

Simulation of confidence intervals

Since the Moran "I" statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a permutation type Monte Carlo simulation can be run to estimate approximate confidence intervals around the "I" values for each distance interval. Specify the number of simulations to be run (e.g., 100, 1000, 10000).

Moran Correlogram Output

The output includes:

1. The sample size
2. The maximum distance
3. The bin (interval) number
4. The midpoint of the distance bin
5. The "I" value for the distance bin

and if a simulation is run:

6. The minimum "I" value for the distance bin
7. The maximum "I" value for the distance bin
8. The 0.5 percentile of "I" for the distance bin
9. The 2.5 percentile of "I" for the distance bin
10. The 97.5 percentile of "I" for the distance bin
11. The 99.5 percentile of "I" for the distance bin

The two pairs of percentiles (2.5 and 97.5; 0.5 and 99.5) create approximate 95% and 99% confidence interval of "I" for each distance bin. The minimum and maximum "I" values create an *envelope*. However, unless a large number of simulations are run, the actual "I" value for any bin may fall outside the envelope. The tabular results can be printed, saved to a text file or saved as a 'dbf' file (MoranCorr<file name> with the file name being provided by the user.

Graphing the "I" values by distance

A graph is produced that shows the "I" value on the Y-axis by the distance bin on the X-axis. Click on the "Graph" button. If a simulation is run, the 2.5 and 97.5 percentiles of the simulated "I" values are also shown on the graph. The graph displays the reduction in spatial autocorrelation with distance. The graph is useful for selecting the type of kernel in the Single- and Dual-kernel interpolation routines when the primary variable is weighted. For a presentation quality graph, however, the output file should be brought into Excel or another graphics program in order to display the change in "I" values and label the axes properly.

Geary Correlogram

The Geary Correlogram calculates the Geary "C" index for different distance intervals/bins (not adjusted for small distances). The "C" value typically varies between 0 and 2 though these are not absolute limits. A "C" value of 1 indicates no spatial autocorrelation. A value of "C" less than 1 indicates positive spatial autocorrelation (zones have values similar to their neighbors) while a value of "C" greater than 1 indicates negative spatial autocorrelation (zones have values different from their neighbors). The user can select any number of distance intervals. The default is 10 distance intervals. The "C" value for each distance interval is calculated with the intensity variable specified on the Primary File page.

Adjust for small distances

If the item is checked, small distances are adjusted so that the maximum weighting is 1 This ensures that the "C" values for individual distances won't become excessively large or excessively small for points that are close together. The default value is no adjustment.

Calculate for individual intervals

By default, the Geary Correlogram routine calculates the "C" values for the cumulative distance from 0 to the end of the interval. If the user checks the box to 'Calculate for individual intervals', then the "C" values for only those pairs of points that fall within the interval are calculated. This can be useful for checking whether points separated by particular distances are clustered or whether there are unreliable "C" values for particular distance intervals.

Simulation of confidence intervals

Since the Geary "C" statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a permutation type Monte Carlo simulation can be run to

estimate approximate confidence intervals around the “C” values for each distance interval. Specify the number of simulations to be run (e.g., 100, 1000, 10000).

Geary Correlogram Output

The output includes:

1. The sample size
2. The maximum distance
3. The bin (interval) number
4. The midpoint of the distance bin
5. The “C” value for the distance bin
6. The Adjusted “C” value for the distance bin

and if a simulation is run:

7. The minimum “C” value for the distance bin
8. The maximum “C” value for the distance bin
9. The 0.5 percentile of “C” for the distance bin
10. The 2.5 percentile of “C” for the distance bin
11. The 97.5 percentile of “C” for the distance bin
12. The 99.5 percentile of “C” for the distance bin.

The two pairs of percentiles (2.5 and 97.5; 0.5 and 99.5) create an approximate 95% and 99% confidence interval. The minimum and maximum “C” values create an *envelope*. However, unless a large number of simulations are run, the actual “C” value for any bin may fall outside the envelope. The tabular results can be printed, saved to a text file or saved as a ‘dbf’ file (GearyCorr<file name> with the file name being provided by the user.

Graphing the “C” values by distance

A graph can be shown with the “C” value on the Y-axis by the distance bin on the X-axis. Click on the “Graph” button. If a simulation is run, the 2.5 and 97.5 percentiles of the simulated “C” values are also shown on the graph. The graph displays the reduction in spatial autocorrelation with distance. The graph is useful for selecting the type of kernel in the single- and dual-kernel interpolation routines when the primary variable is weighted. For a presentation quality graph, however, the output file should be brought into Excel or another graphics program in order to display the change in “C” values and label the axes properly.

Getis-Ord Correlogram

The Getis-Ord Correlogram calculates the Getis-Ord "G" index for different distance intervals/bins. The user can select any number of distance intervals. The default is 10 distance intervals. The statistic requires an intensity variable in the primary file.

Simulation of confidence intervals

Since the Getis-Ord "G" statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a permutation type Monte Carlo simulation can be run to estimate approximate confidence intervals around the "G" values for each distance interval. Specify the number of simulations to be run (e.g., 100, 1000, 10000).

Getis-Ord Correlogram Output

The output includes:

1. The sample size
2. The maximum distance
3. The bin (interval) number
4. The midpoint of the distance bin
5. The "G" value for the distance bin
6. The expected "G" value for the distance bin

and if a simulation is run:

7. The minimum "G" value for the distance bin
8. The maximum "G" value for the distance bin
9. The 0.5 percentile of "G" for the distance bin
10. The 2.5 percentile of "G" for the distance bin
11. The 97.5 percentile of "G" for the distance bin
12. The 99.5 percentile of "G" for the distance bin

The two pairs of percentiles (2.5 and 97.5; 0.5 and 99.5) create an approximate 95% and 99% confidence interval. The minimum and maximum "G" values create an *envelope*. However, unless a large number of simulations are run, the actual "G" value for any bin may fall outside the envelope. The tabular results can be printed, saved to a text file or saved as a 'dbf' file (Getis-OrdCorr<file name> with the file name being provided by the user.

Graphing the "G" values by distance

A graph can be shown that shows the "G" and Expected "G" values on the Y-axis by the distance bin on the X-axis. Click on the "Graph" button. If a simulation is run, the 2.5 and 97.5 percentiles of the simulated "G" values are also shown on the graph along with the "G"; the Expected "G" is not shown in this case. The graph displays the reduction in spatial autocorrelation with distance. Note that the "G" and expected "G" approach 1.0 as the search distance increases, that is as the pairs included within the search distance approximate the number of pairs in the entire data set. The graph is useful for selecting the type of kernel in the single- and dual-kernel interpolation routines when the primary variable is weighted. For a presentation quality graph, however, the output file should be brought into Excel or another graphics program in order to display the change in "G" values and label the axes properly.

Distance Analysis I

Distance analysis provides statistics about the distances between point locations. It is useful for identifying the degree of clustering of points. It is sometimes called second-order analysis. The distance routines are divided into two pages: Distance Analysis I and Distance Analysis II. On the first page, there are four routines for describing properties of the distances.

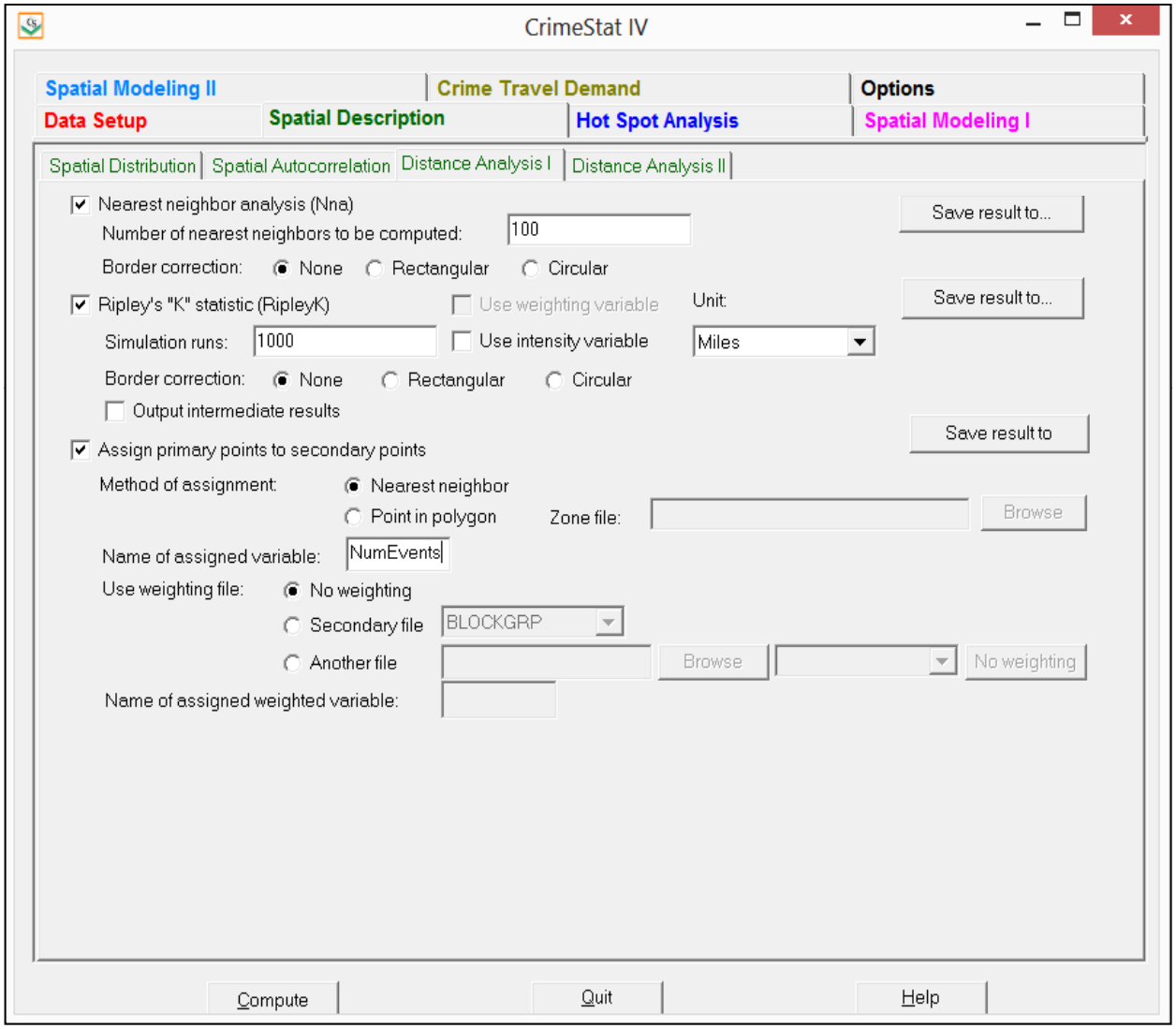
Nearest Neighbor Analysis (Nna)

The nearest neighbor index provides an approximation about whether points are more clustered or dispersed than would be expected on the basis of chance. It compares the average distance of the nearest other point (nearest neighbor) with a spatially random expected distance by dividing the empirical average nearest neighbor distance by the expected random distance (the nearest neighbor index.) The nearest neighbor routine requires that the geographical area be entered on the Measurement Parameters page and that direct distances be used. The NNA routine calculates 10 statistics:

1. The sample size
2. The mean nearest neighbor distance in meters, feet and miles
3. The standard deviation of the nearest neighbor distance in meters, feet and miles
4. The minimum distance in meters, feet and miles
5. The maximum distance in meters, feet and miles
6. The mean random distance (for both the maximum bounding rectangle and the user input area, if provided
7. The mean dispersed distance in meters, feet and miles (for both the maximum bounding rectangle and the user input area, if provided)

Figure 2.7:

Distance Analysis I Statistics



8. The nearest neighbor index (for both the maximum bounding rectangle and the user input area, if provided)
9. The standard error of the nearest neighbor index (for both the maximum bounding rectangle and the user input area, if provided)
10. A significance test of the nearest neighbor index (Z-test)
11. The p-values associated with a one tail and two tail significance test

The tabular results can be printed, saved to a text file or saved as a 'dbf' file. For the latter, specify a file name in the "Save result to" in the dialogue box.

K-*order nearest neighbors*

The K-nearest neighbor index compares the average distance to the K^{th} nearest other point with a spatially random expected distance. The user can specify the number of K-nearest neighbors to be calculated, if more than one is to be calculated. *CrimeStat* will calculate 3 statistics for each order specified:

1. The mean nearest neighbor distance in meters for the order
2. The expected nearest neighbor distance in meters for the order
3. The nearest neighbor index for the order

The NNA routine will use the user-defined area unless none is provided in which case it will use the maximum bounding rectangle. The tabular results can be printed, saved to a text file or output as a 'dbf' file. For the latter, specify a file name in the "Save result to" dialogue box.

Edge correction of nearest neighbors

The nearest neighbor analysis does not adjust for underestimation for incidents near the boundary of the study area. It is possible that there are nearest neighbors outside the boundary that are closer than the measured nearest neighbor. The nearest neighbor analysis has three edge correction options:

1. No adjustment – this is the default;
2. Adjustment that assumes the study area is a rectangle; and
3. Adjustment that assumes the study area is a circle. The rectangular and circular edge corrections adjust the nearest neighbor distances of points near the border. If a point is closer to the border (of either a rectangle or a circle) than to the measured nearest neighbor distance, then the distance to the border is taken as the adjusted nearest neighbor distance.

Linear Nearest Neighbor Analysis

The linear nearest neighbor index provides an approximation about whether points are more clustered or dispersed along road segments than would be expected on the basis of chance. It is used with **indirect** (Manhattan) distances and requires the input of the total length of a road network on the measurement parameters page (see Measurement Parameters.) That is, if indirect distances are checked on the measurement parameters page, then the linear nearest neighbor will be calculated. The linear nearest neighbor index is the ratio of the empirical average linear nearest neighbor distance to the expected linear random distance. The NNA routine outputs 10 statistics for the linear nearest neighbor index:

1. The sample size;
2. The mean linear nearest neighbor distance in meters, feet and miles
3. The minimum distance between points along a grid network
4. The maximum distance between points along a grid network
5. The mean random linear distance
6. The linear nearest neighbor index
7. The standard deviation of the linear nearest neighbor distance in meters, feet and miles
8. The standard error of the linear nearest neighbor index
9. A t-test of the difference between the empirical and expected linear nearest neighbor distance
10. The p-values associated with a one tail and two tail significance test

Linear K-order nearest neighbors

NNA can calculate K-nearest linear neighbors and compare this distance the average linear distance to the K^{th} nearest other point with a spatially random expected distance. The user can specify the number of K-nearest linear neighbors to be calculated, if more than one are to be calculated. *CrimeStat* will calculate 3 statistics for each order specified:

1. The mean linear nearest neighbor distance in meters for the order
2. The expected linear nearest neighbor distance in meters for the order
3. The linear nearest neighbor index for the order

Edge correction of linear nearest neighbors

The nearest neighbor analysis does not adjust for underestimation for incidents near the boundary of the study area. It is possible that there are nearest neighbors outside the boundary

that are closer than the measured nearest neighbor. The nearest neighbor analysis has three edge correction options: 1) No adjustment – this is the default; 2) Adjustment that assumes the study area is a rectangle; and 3) Adjustment that assumes the study area is a circle. The rectangular and circular edge corrections adjust the nearest neighbor distances of points near the border. If a point is closer to the border (of either a rectangle or a circle) than to the measured nearest neighbor distance, then the distance to the border is taken as the adjusted nearest neighbor distance.

Ripley's "K" Statistic (RipleyK)

Ripley's "K" statistic compares the number of points within any distance to an expected number for a spatially random distribution. The empirical count is transformed into a square root function, called L (see documentation for more details). The RipleyK routine calculates 6 statistics:

1. The sample size
2. The maximum distance in meters, feet and miles
3. 100 distance bins
4. The distance for each bin
5. The transformed statistic, L(t), for each distance bin
6. The expected random L under complete spatial randomness, L(csr)

The tabular results can be printed, saved to a text file, or saved as a 'dbf' file. For the latter, specify a file name in the "Save result to" in the dialogue box.

Simulating confidence intervals

A Monte Carlo simulation can be run to evaluate an approximate confidence interval around the L statistic. The user specifies the number of simulation runs and the L statistic is calculated for randomly assigned data. The random output is sorted and percentiles are calculated. Values of L that are greater than any particular percentile indicate more concentration while values of L less than any particular percentile indicate more dispersion. L is calculated for each of 100 distance intervals (bins.) Eight percentiles are identified for these statistics:

1. The minimum for the spatially random L value
2. The maximum for the spatially random L value
3. The 0.5 percentile for the spatially random L value
4. The 2.5 percentile for the spatially random L value

5. The 95 percentile for the spatially random L value
6. The 97.5 percentile for the spatially random L value
7. The 99 percentile for the spatially random L value
8. The 99.5 percentile for the spatially random L value

Confidence intervals can be estimated from these percentiles. The two most commonly used ones are the 95% (defined by the 2.5 and 97.5 percentiles) and the 99% (defined by the 0.5 and 99.5 percentiles). The simulated data that is used can be viewed by checking the 'Dump simulation data' box on the Options tab.

Edge correction of Ripley's K statistic

The default setting for the Ripley's "K" statistic does not adjust for underestimation for incidents near the boundary of the study area. However, it is possible that there are points outside the study area boundary that are closer than the search radius of the circle used to enumerate the "K" statistic. The Ripley's "K" statistic has three edge correction options: 1) No adjustment – this is the default; 2) Adjustment that assumes the study area is a rectangle; and 3) Adjustment that assumes the study area is a circle. The rectangular and circular edge corrections adjust the Ripley's "K" statistic for points near the border. If the distance of a point to the border (of either a rectangle or a circle) is smaller than to the radius of the circle used to enumerate the "K" statistics, then the point is weighted inversely proportional to the area of the search radius that is within the border.

Output Intermediate Results

There is a box labeled “Output intermediate results”. If checked, a separate dbf file will be output that lists the intermediate calculations. The file will be called “RipleyTempOutput.dbf”. There are five output fields:

1. The point number (POINT), starting at 0 (for the first point) and proceeding to N–1 (for the Nth point)
2. The search radius in meters (SEARCHRADI)
3. The count of the number of *other* points that are within the search radius (COUNT)
4. The weight assigned (WEIGHT)
5. The count times the weight (CTIMESW)

Assign Primary Points to Secondary Points

This routine will assign each primary point to a secondary point and then will sum by the number of primary points assigned to each secondary point. It is useful for adding up the number of primary points that are close to each secondary point. For example, in the crime travel demand module, this routine can assign incidents to zones as the module uses zonal totals. The result is a count of primary points associated with each secondary point. It is also possible to sum different variables sequentially. For example, in the crime travel demand module, both the number of crimes originating in each zone and the number of crimes occurring in each zone are needed. This can be accomplished in two runs. First, sum the incidents defined by the origin coordinates to each zone (secondary file). Second, sum the incidents defined by the destination coordinates to each zone (also secondary file). The result would be two columns, one showing the number of origins in each secondary file zone and the second showing the number of destinations in each secondary file zone.

There are two methods for assigning the primary points to the secondary.

Nearest neighbor assignment

This routine assigns each primary point to the secondary point to which it is closest. If there are two or more secondary points that are exactly equal, the assignment goes to the first one on the list.

Point-in-polygon assignment

This routine assigns each primary point to the secondary point for which it falls within its polygon (zone). A zone (polygon) shape file must be provided and the routine checks which secondary zone each primary point falls within.

Zone file for point-in-polygon assignment

If point-in-polygon assignment is used, a zonal file must be provided. This is a polygon file that defines the zones to which the primary points are assigned. The zone file should be the same as the secondary file (see Secondary file). For each point in the primary file, the routine identifies which polygon (zone) it belongs to and then sums the number of points per polygon.

Name of assigned variable

Whether nearest neighbor or point-in-polygon assignment is used, specify the name of the summed variable. The default name is FREQ.

Use weighting file

The primary file records can be weighted by another file. This would be useful for correcting the totals from the primary file. For example, if the primary file were robbery incidents from an arrest record, the sum of this variable (i.e. the total number of robberies) may produce a biased distribution over the secondary file zones because the primary file was not a random sample of all incidents (e.g., if it came from an arrest record where the distribution of robbery arrests is not the same as the distribution of all robbery incidents).

The secondary file or another file can be used to adjust the summed total. The weighting variable should have a field that identifies the ratio of the true to the measured count for each zone. A value of 1 indicates that the summed value for a zone is equal to the true value; hence no adjustment is needed. A value greater than 1 indicates that the summed value needs to be adjusted upward to equal the true value. A value less than 1 indicates that the summed value needs to be adjusted downward to equal the true value.

If another file is to be used for weighting, indicate whether it is the secondary file or, if another file, the name of the other file.

Name of assigned weighted variable

For a weighted sum, specify the name of the variable. The default will be ADJFREQ.

Save result to

For both routines, the output is a 'dbf' file. Define the file name. Note: be careful about using the same name as the secondary file as the saved file will have the new variable. It is best to give it a new name.

A new variable will be added to this file that gives the number of primary points in each secondary file zone and, if weighting is used, a secondary variable will be added which has the adjusted frequency.

Output intermediate results

If the label "Output intermediate results" is checked, a separate dbf file will be output that lists the intermediate calculations. The file will be called "RipleyTempOutput.dbf". There are five output fields:

1. The point number (POINT), starting at 0 (for the first point) and proceeding to N–1 (for the Nth point)
2. The search radius in meters (SEARCHRADI)
3. The count of the number of *other* points that are within the search radius (COUNT)
4. The weight assigned (WEIGHT)
5. The count times the weight (CTIMESW)

Distance Analysis II

On the second Distance Analysis page, there are four routines that calculate distance matrices:

Distance Matrices

1. From each primary point to every other primary point
2. From each primary point to each secondary point
3. From each primary point to the centroid of each reference file grid cell. This requires a reference file to be defined or used.
4. From each secondary point to the centroid of each reference file grid cell. This requires a reference file to be defined or used

CrimeStat can calculate distances between points for a single file or distances between points for two different files. These matrices are useful for examining the frequency of different distances or for providing distances for another program. Because the output files are usually very large, only text output is allowed. This can then be read into a database or large statistical program for processing. Keep in mind that there may be storage problems for large matrices.

Within File Point-to-Point (Matrix)

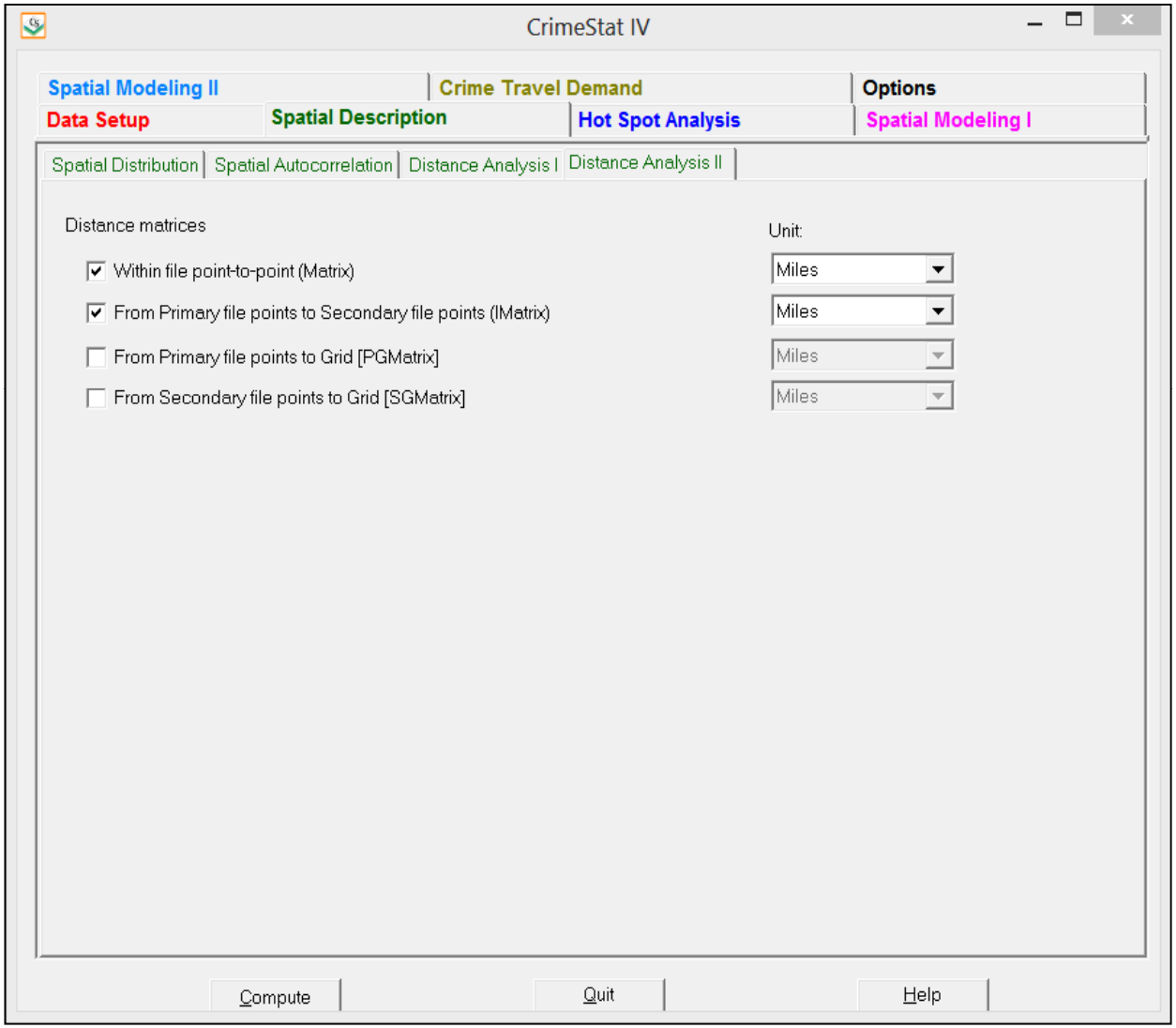
This routine outputs the distance between each point in the primary file to every other point in a specified distance unit (miles, nautical miles, feet, kilometers, or meters). The Matrix output can be saved to a text file.

From Primary File Points to Secondary File Points (IMatrix)

This routine outputs the distance between each point in the primary file to each point in the secondary file in a specified distance unit (miles, nautical miles, feet, kilometers, or meters). The IMatrix output can be saved to a text file.

Figure 2.8:

Distance Analysis II Statistics



From Primary File Points to Grid (PGMatrix)

This routine outputs the distance between each point in the primary file to the centroid of each cell in the reference grid. A reference has to be defined or provided on the Reference file page. Again, the distance units must be specified (miles, nautical miles, feet, kilometers, or meters). The output can be saved to a text file.

From Secondary File Points to Grid (SGMatrix)

This routine outputs the distance between each point in the secondary file to the centroid of each cell in the reference grid. A reference has to be defined or provided on the Reference file page. Again, the distance units must be specified (miles, nautical miles, feet, kilometers, or meters). The output can be saved to a text file.

III. Hot Spot Analysis

Hot spot (or cluster) analysis identifies groups of incidents that are clustered together. It is a method of second-order analysis that identifies the cluster membership of points. There are a number of different hot spot analysis routines in *CrimeStat*. They are organized on three program tabs: Hot Spot analysis I, Hot Spot analysis II, and Hot Spot Analysis of Zones.

Hot Spot Analysis I

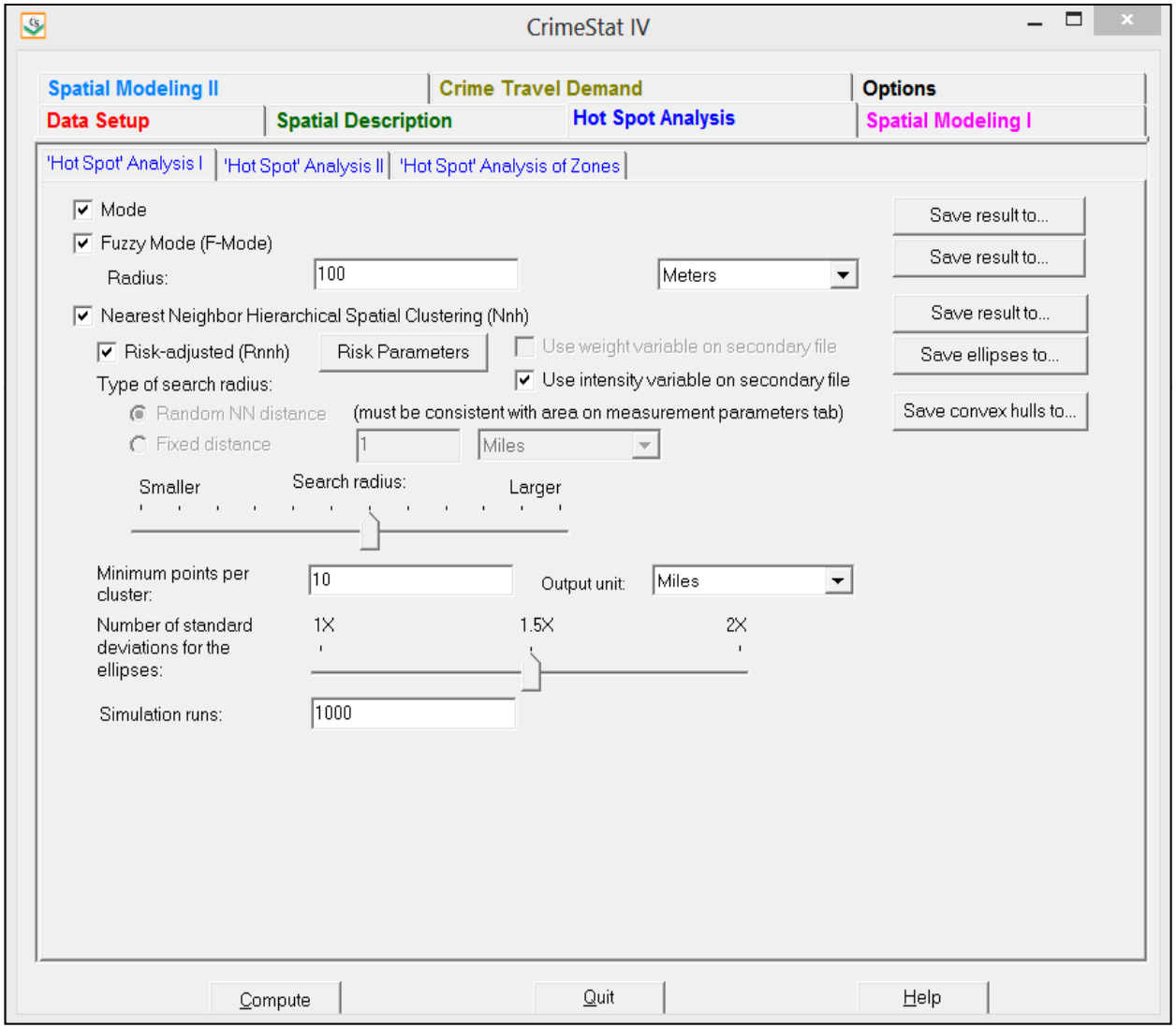
Hot spot (or cluster) analysis identifies groups of incidents that are clustered together. It is a method of second-order analysis that identifies the cluster membership of points. On the Hot Spot Analysis I page, there are four statistics that can be used to identify hot spots: 1) the mode; 2) the fuzzy mode; 3) Nearest neighbor hierarchical spatial clustering; and 4) Risk-adjusted nearest neighbor hierarchical spatial clustering.

Mode

The mode calculates the frequency of incidents for each unique location, defined by an X and Y coordinate. It will output a list of all unique locations and their X and Y coordinates and the number of incidents occurring at each, ranked in decreasing order from most frequent to least frequent. It will also list their rank order from 1 to the last unique location. The data can be output to a 'dbf' file. For the latter, specify a file name in the “Save result to” in the dialogue box.

Figure 2.9:

Hot Spot Analysis I



Fuzzy Mode

The fuzzy mode calculates the frequency of incidents for each unique location within a small, user-specified distance. The user must specify the search radius and the units for the radius (miles, nautical miles, feet, kilometers, or meters). Distances should be small (e.g., less than 0.25 miles). The routine will identify each unique location, defined by its X and Y coordinates, and will calculate the number of incidents that fall within the search radius. It will output a list of all unique locations and their X and Y coordinates and the number of incidents occurring at each, ranked in decreasing order from most frequent to least frequent. It will also list their rank order from 1 to the last unique location. The data can be output to a 'dbf' file.

Nearest Neighbor Hierarchical Spatial Clustering (Nnh)

The nearest neighbor hierarchical spatial clustering routine is a constant-distance clustering routine that groups points together on the basis of spatial proximity. The user defines a threshold distance and the minimum number of points that are required for each cluster, and an output size for displaying the clusters with ellipses. The routine identifies first-order clusters, representing groups of points that are closer together than the threshold distance and in which there is at least the minimum number of points specified by the user. Clustering is hierarchical in that the first-order clusters are treated as separate points to be clustered into second-order clusters, and the second-order clusters are treated as separate points to be clustered into third-order clusters, and so on. Higher-order clusters will be identified only if the distances between their centers are closer than the new threshold distance.

Threshold distance

The threshold distance is the search radius around a pair of points. For each pair of points, the routine determines whether they are closer together than the search radius. There are two ways to determine a threshold distance:

Random nearest neighbor distance

First, the search distance is chosen by the random nearest neighbor distance. The default value is 0.1 (i.e., fewer than 10% of the pairs could be expected to be as close or closer by chance.) Pairs of points that are closer together than the threshold distance are grouped together whereas pairs of points that are greater than the threshold distance are ignored. The smaller the threshold distance, the smaller the significance level that is selected and the fewer pairs will be selected. On the other hand, choosing a larger threshold distance (and, consequently, a higher

significance level) will usually lead to more pairs being selected. However, the more pairs that are selected, the greater the likelihood that clusters could be chance groupings.

The slide bar is used to adjust the significance level. Move the slide bar to the left to choose a smaller threshold distance and to the right to choose a larger threshold distance.

Fixed distance

Second, a fixed distance can be selected. The default is 1 mile. In this case, the search radius uses the fixed distance and the slide bar is inoperative.

Minimum number of points

The minimum number of points required for each cluster allows the user to specify a minimum number of points for each cluster. The default is 10 points. Third, the output size for the clusters can be adjusted by the second slide bar. These are the number of standard deviations defined by the ellipse, from one standard deviation (the default value) to three standard deviations. Typically, one standard deviation will cover about 65% of the cases whereas three standard deviations will cover more than 99% of the cases.

The tabular results can be printed, saved to a text file, or output as a 'dbf' file. The graphical results can be output as either ellipses or as convex hulls (or both) to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Separate file names must be selected for the ellipse output and for the convex hull output. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Tabular output

The routine outputs six results for each cluster that is calculated:

1. The hierarchical order and the cluster number
2. The mean center of the cluster (Mean X and Mean Y)
3. The standard deviational ellipse of the cluster (the rotation and the lengths of the X and Y axes)
4. The number of points in the cluster
5. The area of the cluster

6. The density of the cluster (points divided by area)

Ellipse output

The results can be output graphically as an ellipse to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. A file name should be provided. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

First and higher-order ellipses will be output as separate objects. The prefix will be 'NNH1' for the first-order ellipses, 'NNH2' for the second-order ellipses, and 'NNH3' for the third-order ellipses. Higher-order ellipses will only index the number.

Output size for ellipses

The cluster output size can be adjusted by the lower slide bar. This specifies the number of ellipse standard deviations to be calculated for each cluster: one standard deviation (1X - the default value), one and a half standard deviations (1.5X), or two standard deviations (2X). The default value is one standard deviation. Typically, one standard deviation will cover more than half the cases whereas two standard deviations will cover more than 99% of the cases, though the exact percentage will depend on the distribution. Slide the bar to select the number of standard deviations for the ellipses. The output file is saved as Nnh<number><file name> with the file name being provided by the user. The number is the order of the clustering (i.e., 1, 2…).

Restrictions on the number of clusters can be placed by defining a minimum number of points that are required. The default is 10. If there are too few points allowed, then there will be many very small clusters. By increasing the number of required points, the number of clusters will be reduced.

Convex hull cluster output

The clusters can also be output as convex hulls to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Specify a file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The name will be output with a ‘CNNH1’ prefix for the first-order clusters, a ‘CNNH2’ prefix for the second-order clusters, and a ‘CNNH3’ prefix for the third-order clusters. Higher-order clusters will index only the number.

Simulating confidence intervals

A Monte Carlo simulation can be run to estimate the approximate confidence intervals around first-order Nnh clusters; second- and higher-order clusters are not simulated since their structure depends on first-order clusters. The user specifies the number of simulation runs and the Nnh clustering is calculated for randomly assigned data. The random output is sorted and percentiles are calculated. The output includes the number of first-order clusters, the area, the number of points, and the density. Twelve percentiles are identified for these statistics:

1. The minimum for the spatially random Nnh simulations
2. The maximum for the spatially random Nnh simulations
3. The 0.5 percentile for the spatially random Nnh simulations
4. The 1 percentile for the spatially random Nnh simulations
5. The 2.5 percentile for the spatially random Nnh simulations
6. The 5 percentile for the spatially random Nnh simulations
7. The 10 percentile for the spatially random Nnh simulations
8. The 90 percentile for the spatially random Nnh simulations
9. The 95 percentile for the spatially random Nnh simulations
10. The 97.5 percentile for the spatially random Nnh simulations
11. The 99 percentile for the spatially random Nnh simulations
12. The 99.5 percentile for the spatially random Nnh simulations

Confidence intervals can be estimated from these percentiles. The two most commonly used ones are the 95% (defined by the 2.5 and 97.5 percentiles) and the 99% (defined by the 0.5 and 99.5 percentiles). The simulated data that is used can be viewed by checking the 'Dump simulation data' box on the Options tab.

Risk-Adjusted Nearest Neighbor Hierarchical Spatial Clustering (Rnnh)

The risk-adjusted nearest neighbor hierarchical spatial clustering routine groups points together on the basis of spatial proximity, but the grouping is adjusted according to the distribution of a baseline variable. The routine requires both a primary file (e.g., robberies) and a secondary file (e.g., population). For the secondary variable, if an intensity or weight variable is to be used, it should be specified.

The user selects a threshold probability for grouping a *pair* of points together by chance and the minimum number of points that are required for each cluster, and an output size for displaying the clusters with ellipses. In addition, a kernel density model for the secondary variable must be specified. The threshold distance is determined by the threshold probability and the grid cell density produced by the kernel density estimate of the secondary variable. Thus, in areas with high density of the secondary variable, the threshold distance is smaller than in areas with low density of the secondary variable.

The routine identifies first-order clusters, representing groups of points that are closer together than the threshold distance and in which there is at least the minimum number of points specified by the user. Clustering is hierarchical in that the first-order clusters are treated as separate points to be clustered into second-order clusters, and the second-order clusters are treated as separate points to be clustered into third-order clusters, and so on. Higher-order clusters will be identified only if the distance between their centers are closer than the new threshold distance.

The tabular results can be printed, saved to a text file, or output as a 'dbf' file. The graphical results can be output as either ellipses or as convex hulls (or both) to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Separate file names must be selected for the ellipse output and for the convex hull output. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Threshold distance

The threshold distance is the confidence interval around a random expected distance for a *pair* of points (called *credible interval*). However, unlike the Nnh routine where the threshold distance is constant throughout the study area, the threshold distance for the Rnnh routine is adjusted inversely proportional to the distribution of the secondary (baseline) variable. In areas with a high density of the secondary variable, the threshold distance will be small whereas in areas with a low density of the secondary variable, the threshold distance will be large. The default threshold probability is 0.1 (i.e., fewer than 10% of the pairs could be expected to be as close or closer by chance.) Pairs of points that are closer together than the threshold distance are grouped together whereas pairs of points that are greater than the threshold distance are ignored. The smaller the significance level that is selected, the smaller will be the threshold distance with, usually, fewer pairs being selected. On the other hand, choosing a higher significance level, the larger the threshold distance and, usually, the more pairs will be selected. However, the higher the significance level chosen, the greater the likelihood that clusters could be chance groupings.

Move the slide bar to the left to choose a smaller threshold distance and to the right to choose a larger threshold distance.

Risk parameters

A density estimate of the secondary variable must be calculated to adjust the threshold distance of the primary variable. This is done through kernel density estimation. The risk parameters tab defines this model. The secondary variable is automatically assumed to be the 'at risk' (baseline) variable. The user specifies a method of interpolation (normal, uniform, quartic, triangular, and negative exponential kernels) and the choice of bandwidth (fixed interval or adaptive interval). If an adaptive interval is used, the minimum sample size for the band width (search radius) must be specified. If a fixed interval is used, the size of the interval (radius) must be specified along with the measurement units (miles, nautical miles, feet, kilometers, or meters). Finally, the units of the output density must be specified (squared miles, squared nautical miles, squared feet, squared kilometers, squared meters).

The routine overlays a 50 x 50 grid on the study area and calculates a kernel density estimate of the secondary variable. The density is then re-scaled to equal the sample size of the primary variable. For each grid cell, a cell-specific threshold distance is calculated for grouping a pair of points together by chance. The threshold probability selected by the user is applied to this cell-specific threshold distance to produce a threshold distance that corresponds to the cell-specific confidence interval. Pairs of points that are closer than the cell-specific threshold distance are selected for first-order clustering.

Use of intensity variable

If an intensity variable has been used in the secondary file, the intensity box should be checked.

Minimum number of points

The minimum number of points required for each cluster allows the user to specify a minimum number of points for each cluster. The default is 10 points. Third, the output size for the clusters can be adjusted by the second slide bar. These are the number of standard deviations defined by the ellipse, from one standard deviation (the default value) to three standard deviations. Typically, one standard deviation will cover about 65% of the cases whereas three standard deviations will cover more than 99% of the cases.

Tabular output

The routine outputs six results for each cluster that is calculated:

1. The hierarchical order and the cluster number
2. The mean center of the cluster (Mean X and Mean Y)
3. The standard deviational ellipse of the cluster (the rotation and the lengths of the X and Y axes)
4. The number of points in the cluster
5. The area of the cluster
6. The density of the cluster (points divided by area)

Ellipse output

The results can be output graphically as an ellipse to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. A file name should be provided. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

First- and higher-order ellipses will be output as separate objects. The prefix will be 'RNNH1' for the first-order ellipses, 'RNNH2' for the second-order ellipses, and 'RNNH3' for the third-order ellipses. Higher-order ellipses will only index the number.

Output size for ellipses

The cluster output size can be adjusted by the lower slide bar. This specifies the number of ellipse standard deviations to be calculated for each cluster: one standard deviation (1X - the default value), one and a half standard deviations (1.5X), or two standard deviations (2X). The default value is one standard deviation. Typically, one standard deviation will cover more than half the cases whereas two standard deviations will cover more than 99% of the cases, though the exact percentage will depend on the distribution. Slide the bar to select the number of standard deviations for the ellipses. The output file is saved as Rnnh<number><file name> with the file name being provided by the user. The number is the order of the clustering (i.e., 1, 2…).

Restrictions on the number of clusters can be placed by defining a minimum number of points that are required. The default is 10. If there are too few points allowed, then there will

be many very small clusters. By increasing the number of required points, the number of clusters will be reduced.

Convex hull cluster output

The clusters can also be output as convex hulls to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Specify a file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

First- and higher-order clusters will be output as separate objects. The clusters will have a 'CRNNH1' prefix for the first-order clusters, a 'CRNNH2' prefix for the second-order clusters, and a 'CRNNH3' prefix for the third-order clusters. Higher-order clusters will index only the number.

Simulating confidence intervals

A Monte Carlo simulation can be run to estimate the approximate confidence intervals around first-order Rnnh clusters; second- and higher-order clusters are not simulated since their structure depends on first-order clusters. The user specifies the number of simulation runs and the Rnnh clustering is calculated for randomly assigned data. The random output is sorted and percentiles are calculated. The output includes the number of first-order clusters, the area, the number of points, and the density.

Twelve percentiles are identified for these statistics:

1. The minimum for the spatially random Rnnh simulations
2. The maximum for the spatially random Rnnh simulations
3. The 0.5 percentile for the spatially random Rnnh simulations
4. The 1 percentile for the spatially random Rnnh simulations
5. The 2.5 percentile for the spatially random Rnnh simulations
6. The 5 percentile for the spatially random Rnnh simulations
7. The 10 percentile for the spatially random Rnnh simulations
8. The 90 percentile for the spatially random Rnnh simulations
9. The 95 percentile for the spatially random Rnnh simulations
10. The 97.5 percentile for the spatially random Rnnh simulations
11. The 99 percentile for the spatially random Rnnh simulations

12. The 99.5 percentile for the spatially random Rnnh simulations

Confidence intervals can be estimated from these percentiles. The two most commonly used ones are the 95% (defined by the 2.5 and 97.5 percentiles) and the 99% (defined by the 0.5 and 99.5 percentiles). The simulated data that is used can be viewed by checking the 'Dump simulation data' box on the Options tab.

Hot Spot Analysis II

On the Hot Spot Analysis II page, there are two statistics that can be used to identify hot spots: 1) STAC; and 2) K-means clustering.

Spatial and Temporal Analysis of Crime (STAC)

The Spatial and Temporal Analysis of Crime (STAC) routine is a variable-distance clustering routine. It initially groups points together on the basis of a constant search radius, but then combines clusters that overlap. On the STAC Parameters tab, define a search radius, the minimum number of points that are required for each cluster, and an output size for displaying the clusters with ellipses.

The tabular results can be printed, saved to a text file, or output as a 'dbf' file. The graphical results can be output as either ellipses or as convex hulls (or both) to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Separate file names must be selected for the ellipse output and for the convex hull output. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

STAC parameters

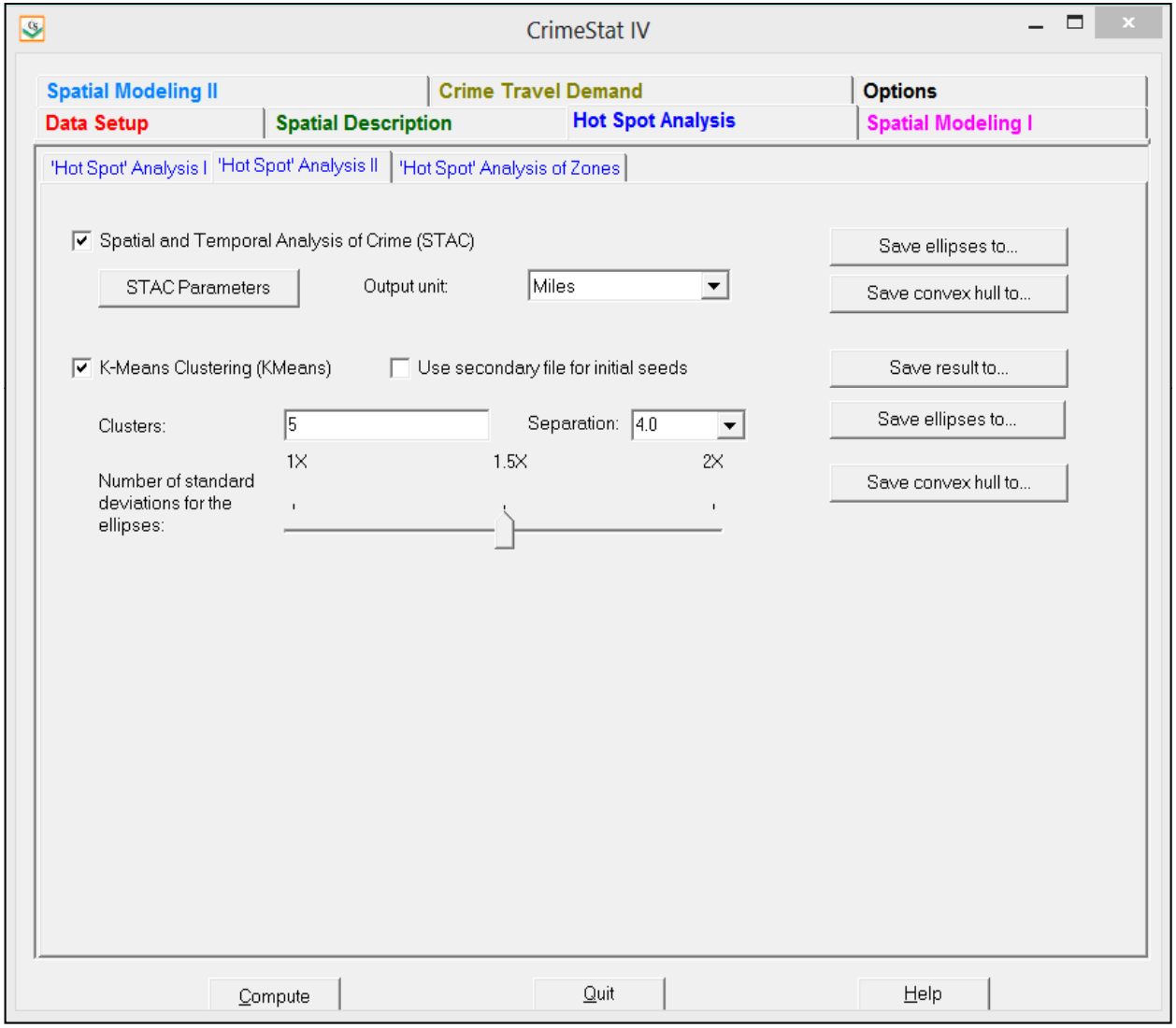
The STAC parameters tab allows the selection of a search radius, the minimum number of points, the scan type, the boundary definition, the number of simulation runs, and the output size of the STAC ellipses.

Search radius

The search radius is the distance within the STAC routine searches. The default is 0.5 miles. A 20 x 20 grid is overlaid on the study area. At each intersection of a row and a

Figure 2.10:

Hot Spot Analysis II



column, the routine counts all points that are closer than the search radius. Overlapping circles are combined to form variable-size clusters. The smaller the search radius that is selected, the fewer points will be selected. On the other hand, choosing a larger search area, the more points will be selected. However, the larger the search area, the greater the likelihood that clusters could be chance groupings. On the STAC Parameters tab, type the search radius into the box and specify the measurement units (miles, nautical miles, feet, kilometers, or meters).

Scan type

The scan type is the type of grid overlaid on the study area. There are two choices: rectangular (default) and triangular.

Boundary

The study area boundaries can be defined from the data set or the reference grid.

Minimum number of points

The minimum number of points required for each cluster allows the user to specify a minimum number of points for each cluster. The default is 5 points. If there are too few points allowed, then there will be many very small clusters. By increasing the number of required points, the number of clusters will be reduced. On the STAC Parameters tab, type the minimum number of points each cluster is required to have.

Tabular output

The routine outputs eight results for each cluster that is calculated:

1. The hierarchical order and the cluster number
2. The mean center of the cluster (Mean X and Mean Y)
3. The standard deviational ellipse of the cluster (the rotation and the lengths of the X and Y axes)
4. The number of points in the cluster
5. The area of the cluster
6. The density of the cluster (cluster points divided by area)
7. The number of points in the ellipse
8. The density of the ellipse (ellipse points divided by area)

Ellipse output

The results can be output graphically as an ellipse to *ArcGIS* 'shp', *MapInfo 'mif', various ASCII formats*, or *Google Earth* 'kml' (if the coordinates are spherical) files. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. The ellipses will be output as combined objects. The prefix will be 'ST'.

Output size for ellipses

The cluster output size can be adjusted by the lower slide bar This specifies the number of ellipse standard deviations to be calculated for each cluster: one standard deviation (1X - the default value), one and a half standard deviations (1.5X), or two standard deviations (2X). The default value is one standard deviation. Typically, one standard deviation will cover more than half the cases whereas two standard deviations will cover more than 99% of the cases, though the exact percentage will depend on the distribution. The output file is saved as St<file name> with the file name being provided by the user. On the STAC Parameters tab, slide the bar to select the number of standard deviations for the ellipses.

Convex hull cluster output

The clusters can also be output as convex hulls to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Specify a file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. The name will be output with a 'CST' prefix.

Simulating confidence intervals

A Monte Carlo simulation can be run to estimate the approximate confidence intervals around the STAC clusters. The user specifies the number of simulation runs and the STAC clustering is calculated for randomly assigned data. The random output is sorted and percentiles are calculated. The output includes the number of clusters, the area, the number of points, and the density. Fifteen percentiles are identified for these statistics:

1. The minimum for the spatially random STAC simulations
2. The minimum for the spatially random STAC simulations

3. The minimum for the spatially random STAC simulations
4. The minimum for the spatially random STAC simulations
5. The maximum for the spatially random STAC simulations
6. The 0.5 percentile for the spatially random STAC simulations
7. The 1 percentile for the spatially random STAC simulations
8. The 2.5 percentile for the spatially random STAC simulations
9. The 5 percentile for the spatially random STAC simulations
10. The 10 percentile for the spatially random STAC simulations
11. The 90 percentile for the spatially random STAC simulation
12. The 95 percentile for the spatially random STAC simulations
13. The 97.5 percentile for the spatially random STAC simulations
14. The 99 percentile for the spatially random STAC simulations
15. The 99.5 percentile for the spatially random STAC simulations

Confidence intervals can be estimated from these percentiles. The two most commonly used ones are the 95% (defined by the 2.5 and 97.5 percentiles) and the 99% (defined by the 0.5 and 99.5 percentiles). The simulated data that is used can be viewed by checking the 'Dump simulation data' box on the Options tab.

K-Means Clustering (Kmeans)

The K-means clustering routine is a procedure for partitioning all the points into K groups in which K is a number assigned by the user. The routine finds K seed locations in which points are assigned to the nearest cluster. The default K is 5. If K is small, the clusters will typically cover larger areas.

The tabular results can be printed, saved to a text file, or output as a 'dbf' file. The graphical results can be output as either ellipses or as convex hulls (or both) to *ArcGIS* 'shp', *MapInfo 'mif'*, various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Separate file names must be selected for the ellipse output and for the convex hull output. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Initial cluster locations

The routine starts with an initial guess (seed) for the K locations and then conducts local optimization. The user can modify the location of the initial clusters in two ways, which are not mutually exclusive:

Separation

1. The separation between the initial clusters can be increased or decreased. There is a separation scale with pre-defined values from 1 to 10; the default is 4. The user can type in any number, however (e.g., 15). Increasing the number increases the separation between the initial cluster locations while decreasing the number decreases the separation.

Initial seed locations

2. The user can define the initial seed locations and the number of clusters, K, with a secondary file. The routine takes K from the number of points in the secondary file and takes the X/Y coordinates of the points as the initial seed locations.

Tabular output

The routine outputs seven characteristics for each cluster that is calculated:

1. The cluster ID
2. The center of minimum distance of the cluster (Mean X and Mean Y)
3. The standard deviational ellipse of the cluster (the rotation and the lengths of the X and Y axes)
4. The area of the cluster
5. The sum of squares in distances between the center of minimum distance of the cluster and each point that is part of the cluster
6. The mean squared error of the distances between the center of minimum distance of the cluster and each point that is part of the cluster
7. The number of points in the cluster

Ellipse output

The results can be output graphically as an ellipse to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Specify a file

name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. The ellipses will be output as separate objects with a 'KM' prefix.

Output size for ellipses

For both methods, the cluster output size can be adjusted with the lower slide bar. This specifies the number of ellipse standard deviations to be calculated for each cluster: one standard deviation (1X - the default value), one and a half standard deviations (1.5X), or two standard deviations (2X). The default value is one standard deviation. Typically, one standard deviation will cover more than half the cases whereas two standard deviations will cover more than 99% of the cases, though the exact percentage will depend on the distribution. Slide the bar to select the number of standard deviations for the ellipses. The output file is saved as Km<file name> with the file name being provided by the user.

Convex hull cluster output

The clusters can also be output as convex hulls to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or *Google Earth* 'kml' (if the coordinates are spherical) files. Specify a file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. The convex hulls will be output as separate objects with a 'CKM' prefix.

Hot Spot Analysis of Zones

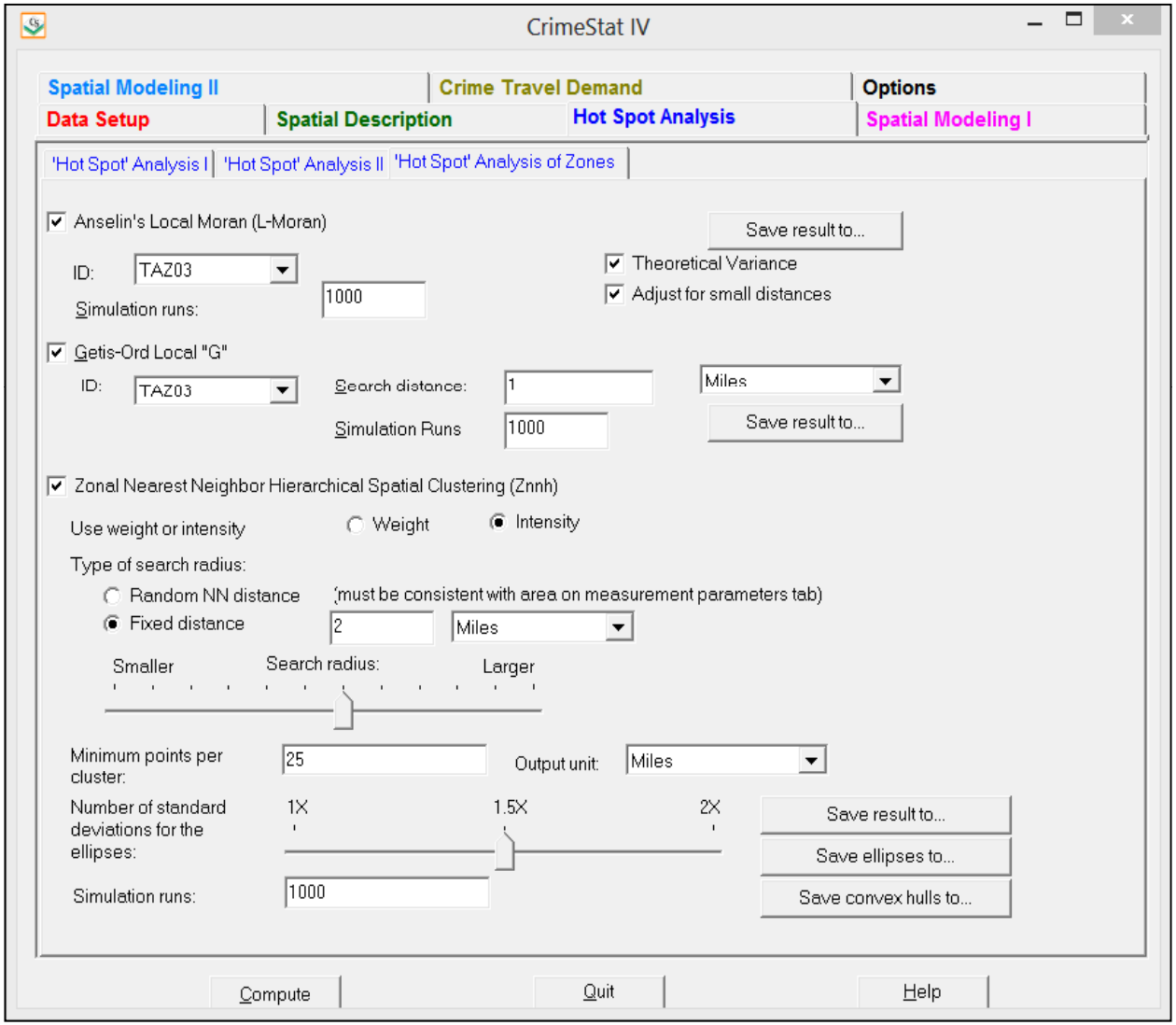
The Hot Spot Analysis of Zones section includes clustering statistics for zonal data. These include 1) Anselin's local Moran; 2) the Getis-Ord local "G", and 3) the zonal nearest neighbor hierarchical clustering algorithm.

Anselin's Local Moran (L-Moran)

Anselin's Local Moran statistic applies the Moran's "I" statistic to individual points (or zones) to assess whether particular points/zones are spatially related to the nearby points (or zones). The statistic requires an intensity variable in the primary file. Unlike the global Moran's "I" statistic, the local Moran is applied to each individual zone. The index points to

Figure 2.11:

Hot Spot Analysis of Zones



clustering or dispersion relative to the local neighborhood. Zones with high "I" values have an intensity value that is higher than their neighbors while zones with low "I" values have intensity values lower than their neighbors. The output can be printed or output as a 'dbf' file.

ID field

The user should indicate a field for the ID of each point (or zone). This ID will be saved with the output and can then be linked with the input file (Primary File) for mapping.

Theoretical variance

If checked, the routine will calculate the theoretical variance of the "I" value for each zone (see documentation for details).

Adjust for small distances

If checked, small distances are adjusted so that the maximum weighting is no higher than 1. This ensures that the local "I" won't become excessively large for points that are grouped together. The default setting is no adjustment.

Simulation of confidence intervals

A Monte Carlo simulation can be run to estimate approximate confidence intervals around the "I" value for each zone. Note, a simulation may take time to run especially if the data set is large or if a large number of simulation runs are requested. Specify the number of simulations to be run (e.g., 100, 1000, 10000).

Output

The output is for each zone and includes:

1. The sample size
2. The ID for the zone
3. The X coordinate for the zone
4. The Y coordinate for the zone
5. The "I" for the zone
6. The expected "I" for the zone

and if the theoretical variance is checked:

7. The theoretical variance of the “I” for the zone
8. A Z-test of the “I” under the assumption of normality

and if a simulation is run:

9. The 0.5 percentile of “I” for the zone
10. The 2.5 percentile of “I” for the zone
11. The 97.5 percentile of “I” for the zone
12. The 99.5 percentile of “I” for the zone

The two pairs of percentiles (2.5 and 97.5; 0.5 and 99.5) create approximate 95% and 9% confidence interval of “I” for each zone. The tabular results can be printed, saved to a text file or saved as a ‘dbf’ file (LMoranCorr<file name> with the file name being provided by the user.

The ‘dbf’ output file can then be linked to the input ‘dbf’ file by using the ID field as a matching variable. This would be done if the user wants to map the “I” variable, the Z-test, or those zones for which the “I” value is either higher than the 97.5 or 99.5 percentiles or lower than the 2.5 or 0.5 percentiles of the simulation results.

Getis-Ord Local “G” (L-Getis-Ord)

The Getis-Ord “G” statistic is an index of spatial autocorrelation for values of a variable that fall within a specified distance of each other. When compared to an expected value of G under the assumption of no spatial association, it has the advantage over other global spatial autocorrelation measures (Moran, Geary) in that it can distinguish between hot spots and cold spots. The “G” value is calculated with the intensity variable specified on the Primary File page and with respect to a specified search distance (defined by the user).

The Getis-Ord Local “G” statistic applies the Getis-Ord "G" statistic to individual zones to assess whether particular zones are spatially related to the nearby ones (‘neighbors’). Unlike the global Getis-Ord “G”, the Getis-Ord Local “G” is applied to each individual zone.

By itself, the G statistic for an individual zone is not very meaningful. The “G” value varies from 0 to 1 since it indicates the interaction of pairs of zones that are within the search distance relative to the interaction of all pairs of zones. As the search distance increases, this statistic will automatically approach 1.0. Consequently, G is compared to an expected value of G under the assumption of no significant spatial association.

Further, under the assumption that G is normally distributed, a Z-test can be constructed that tests for the significance of the actual G. A positive Z-value indicates spatial clustering of high values more than what would be expected under chance (hot spots) while a negative Z-value indicates spatial clustering of low values more than what would be expected under chance (cold spots). A “G” value around 0 indicates no spatial autocorrelation.

ID field

The user should indicate a field for the ID of each point (or zone). This ID will be saved with the output and can then be linked with the input file (Primary File) for mapping.

Search distance

The user must specify a search distance for the test and indicate the distance units (miles, nautical miles, feet, kilometers, or meters).

Simulation of confidence intervals

Since the Getis-Ord “G” statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a permutation type Monte Carlo simulation can be run to estimate approximate confidence intervals around the “G” value. Specify the number of simulations to be run (e.g., 100, 1000, 10000).

Output

The output is for each zone and includes:

1. The sample size
2. The ID for the zone
3. The X coordinate for the zone
4. The Y coordinate for the zone
5. The “G” for the zone
6. The expected “G” for the zone
7. The standard deviation of “G” for the zone
8. A Z-test of "G" under the assumption of normality for the zone

and if a simulation is run:

9. The 0.5 percentile of “G” for the zone

10. The 2.5 percentile of "G" for the zone
11. The 97.5 percentile of "G" for the zone
12. The 99.5 percentile of "G" for the zone

The two pairs of percentiles (2.5 and 97.5; 0.5 and 99.5) create approximate 95% and 99% confidence interval of "G" for each zone. The tabular results can be printed, saved to a text file or saved as a 'dbf' file (LGetis-OrdCorr<file name> with the file name being provided by the user.

The 'dbf' output file can then be linked to the input 'dbf' file by using the ID field as a matching variable. This would be done if the user wants to map the "G" variable, the expected "G" or those zones for which the "G" value is either higher than the 97.5 or 99.5 percentiles or lower than the 2.5 or 0.5 percentiles of the simulation results.

Zonal Nearest Neighbor Hierarchical Clustering (Znnh)

The zonal nearest neighbor hierarchical spatial clustering routine applies the nearest neighbor hierarchical clustering algorithm. The point-based Nnh is a constant-distance clustering routine that groups points together on the basis of spatial proximity. A threshold distance is defined and the minimum number of points that are required for each cluster specified. The output can be displayed with ellipses or convex hulls.

On the other hand, in the zonal Nnh (Znnh), the algorithm is adjusted to allow *weighting* of each zone, usually applied to a single point within the zone (e.g., a centroid). Thus, if the 'point' is a centroid of a zone, then the weighting is an attribute assigned to that centroid (e.g., population, employment, median household income). Clusters are groups of adjacent zones that have much higher weights than non-clustered zones.

The routine requires a primary file (e.g., robberies) that is weighted with the weight or intensity variable (see Primary File). On the Znnh routine, the user defines a weighting variable, a threshold distance and the minimum number of zones that are required for each cluster, and an output size for displaying the clusters with ellipses or convex hulls.

The routine identifies first-order clusters that represent groups of zones that are closer together than the threshold distance, that have the highest weights, and in which there is at least the minimum number of zones specified by the user (the minimum is 3 zones). Clustering is hierarchical in that the first-order clusters are treated as separate zones to be clustered into second-order clusters, and the second-order clusters are treated as separate zones to be clustered into third-order clusters, and so on. Higher-order clusters will be identified only if the distances between their centers are closer than the new threshold distance.

Weighting variable

Each zone must be weighted by a variable. This can be either the intensity variable or the weight variable defined on the Primary File page (but not both). The user specifies whether the intensity or the variable variable is to be used. The default is Intensity.

Threshold distance

The threshold distance is the search radius around a zone centroid. For each pair of zones, the routine determines whether they are closer together than the search radius. There are two ways to determine a threshold distance:

Random nearest neighbor distance

First, the search distance is chosen by the random nearest neighbor distance. The default value is 0.1 (i.e., fewer than 10% of the pairs could be expected to be as close or closer by chance.) Pairs of zones that are closer together than the threshold distance are grouped together whereas pairs of zones that are greater than the threshold distance are ignored. The smaller the threshold distance and the smaller the significance level that is selected, then the fewer numbers of paired zones will be selected. On the other hand, choosing a larger threshold distance (and, consequently, a higher significance level) will usually lead to more pairs being selected. However, the more pairs that are selected, the greater the likelihood that clusters could be chance groupings.

The slide bar is used to adjust the significance level. Move the slide bar to the left to choose a smaller threshold distance and to the right to choose a larger threshold distance.

Fixed distance

Second, a fixed distance can be selected. The default is 1 mile. In this case, the search radius uses the fixed distance and the slide bar is inoperative.

Minimum number of zones

The minimum number of zones required for each cluster allows the user to specify a minimum number of zones for each cluster. The default is 10 zones and the minimum is 3. Third, the output size for the clusters can be adjusted by the second slide bar. These are the number of standard deviations defined by the ellipse, from one standard deviation (the default value) to three

standard deviations. Typically, one standard deviation will cover about 65% of the cases whereas three standard deviations will cover more than 99% of the cases.

The tabular results can be printed, saved to a text file, or output as a 'dbf' file. The graphical results can be output as either ellipses or as convex hulls (or both) to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or Google Earth 'kml' (if the coordinate system is spherical) files. Separate file names must be selected for the ellipse output and for the convex hull output. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Simulating confidence intervals

A Monte Carlo simulation can be run to estimate the approximate confidence intervals around first-order Nnh clusters; second- and higher-order clusters are not simulated since their structure depends on first-order clusters. The user specifies the number of simulation runs and the Nnh clustering is calculated for randomly assigned data. The random output is sorted and percentiles are calculated. The output includes the number of first-order clusters, the area, the number of zones, and the density.

Confidence intervals can be estimated from these percentiles. The two most commonly used ones are the 95% (defined by the 2.5 and 97.5 percentiles) and the 99% (defined by the 0.5 and 99.5 percentiles). The simulated data that is used can be viewed by checking the 'Dump simulation data' box on the Options tab.

Type of graphical output

The type of graphical output is specified, either standard deviational ellipses or convex hulls around the zones identified in each cluster. If the output is to be ellipses, then the output size for the clusters can be adjusted by the second slide bar. These are the number of standard deviations defined by the ellipse, from one standard deviation (the default value) to three standard deviations. Typically, one standard deviation will cover about 50-60% of the zones (and a higher percentage of the total of the weighting variable) whereas three standard deviations will cover more than 99% of the zones. On the other hand, if the output is to be convex hulls, the routine outputs a convex hull for each identified cluster.

Ellipse output

The results can be output graphically as an ellipse to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or Google Earth 'kml' (if the coordinate system is spherical) files. A file name should be provided. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

First and higher-order ellipses will be output as separate objects. The prefix will be 'NNH1' for the first-order ellipses, 'NNH2' for the second-order ellipses, and 'NNH3' for the third-order ellipses. Higher-order ellipses will only index the number.

Output size for ellipses

The cluster output size can be adjusted by the lower slide bar. This specifies the number of ellipse standard deviations to be calculated for each cluster: one standard deviation (1X - the default value), one and a half standard deviations (1.5X), or two standard deviations (2X). The default value is one standard deviation. Typically, one standard deviation will cover more than half the zones in a cluster whereas two standard deviations will cover more than 99% of the zones in a cluster, though the exact percentage will depend on the distribution. Slide the bar to select the number of standard deviations for the ellipses. The output file is saved as Znnh<number><file name> with the file name being provided by the user. The number is the order of the clustering (i.e., 1, 2…).

Restrictions on the number of clusters can be placed by defining a minimum number of zones that are required. The default is 10 and the minimum is 3. If there are too few zones allowed, then there will be many very small clusters. By increasing the number of required zones, the number of clusters will be reduced.

Convex hull cluster output

The clusters can also be output as convex hulls to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or Google Earth 'kml' (if the coordinate system is spherical) files. Specify a file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The name will be output with a 'CNNH1' prefix for the first-order clusters, a 'CNNH2' prefix for the second-order clusters, and a 'CNNH3' prefix for the third-order clusters. Higher-order clusters will index only the number.

Note that ellipses may extend beyond the zones that are clustered together and may also leave out zones that are part of the cluster. Ellipses are abstractions and, while good for visualization, are not precise. Convex hulls are more precise since they define only those zones that are part of the cluster.

Tabular output

The routine outputs six results for each cluster that is calculated:

1. The hierarchical order and the cluster number
2. The mean center of the cluster (Mean X and Mean Y)
3. The standard deviational ellipse of the cluster (the rotation and the lengths of the X and Y axes)
4. The number of zones in the cluster
5. The area of the cluster
6. The density of the cluster (the total weight of the zones divided by area)

and if a simulation is run:

7. The minimum for the spatially random Znnh simulations:
8. The maximum for the spatially random Znnh simulations
9. The 0.5 percentile for the spatially random Znnh simulations
10. The 1 percentile for the spatially random Znnh simulations
11. The 2.5 percentile for the spatially random Znnh simulations
12. The 5 percentile for the spatially random Znnh simulations
13. The 10 percentile for the spatially random Znnh simulations
14. The 90 percentile for the spatially random Znnh simulations
15. The 95 percentile for the spatially random Znnh simulations
16. The 97.5 percentile for the spatially random Znnh simulations
17. The 99 percentile for the spatially random Znnh simulations
18. The 99.5 percentile for the spatially random Znnh simulations

IV. Spatial Modeling I

The first spatial modeling section conducts kernel density estimation, Head Bang statistics, space-time analysis, journey-to-crime calibration and estimation, and Bayesian journey-to-crime diagnostics and estimation. The spatial modeling section is made up of five distinct tabs: Interpolation I, Interpolation II, Space-time analysis, Journey-to-crime estimation, and Bayesian Journey-to-crime estimation.

Interpolation I

The interpolation I tab allows estimates of point density using the kernel density smoothing method. There are two types of kernel density smoothing, one applied to a single distribution of points and the other that compares two different distributions. Each type has variations on the method that can be selected. Both types require a reference file that is overlaid on the study area (see Reference file.) The kernels are placed over each point and the distance between each reference cell and each point are evaluated by the kernel function. The individual kernel estimates for each cell are summed to produce an overall estimate of density for that cell. The intensity and weighting variables can be used in the kernel estimate. The densities can be converted into probabilities.

Single Kernel Density Estimate (KernelDensity)

The single kernel density routine estimates the density of points for a single distribution by overlaying a symmetrical surface over each point, evaluating the distance from the point to each reference cell by the kernel function, and summing the evaluations at each reference cell.

File to be interpolated

The estimate can be applied to either the primary file (see Primary File) or a secondary file (see Secondary file.) Select which file is to be interpolated. The default is the Primary.

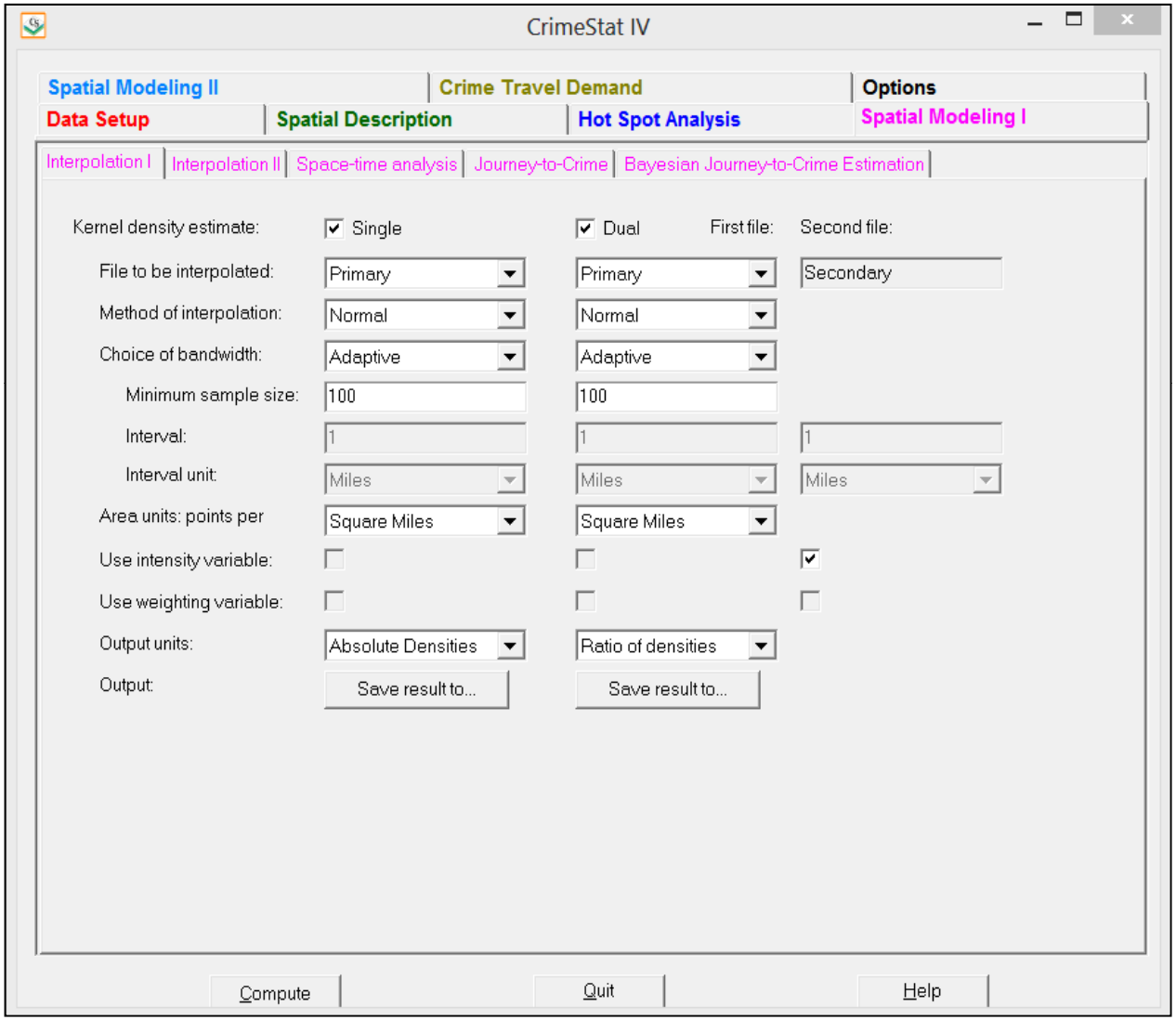
Method of interpolation

There are five types of kernel distributions that can be used to estimate point density:

1. The **normal** kernel overlays a three-dimensional normal distribution over each point that then extends over the area defined by the reference file. This is the default kernel function.

Figure 2.12:

Interpolation I Statistics



2. The **uniform** kernel overlays a uniform function over each point that only extends for a limited distance.

3. The **quartic** kernel overlays a quartic function over each point that only extends for a limited distance.

4. The **triangular** kernel overlays a three-dimensional triangle over each point that only extends for a limited distance.

5. The **negative exponential** kernel overlays a three dimensional negative exponential function over each point that only extends for a limited distance

The methods produce similar results though the normal is generally smoother for any given bandwidth.

Choice of bandwidth

The kernels are applied to a limited search distance, called 'bandwidth'. For the normal kernel, bandwidth is the standard deviation of the normal distribution. For the uniform, quartic, triangular and negative exponential kernels, bandwidth is the radius of a circle defined by the surface. For all types, larger bandwidth will produce smoother density estimates and both adaptive and fixed bandwidth intervals can be selected.

Adaptive bandwidth

An adaptive bandwidth distance is identified by the minimum number of other points found within a circle drawn around a single point. A circle is placed around each point, in turn, and the radius is increased until the minimum sample size is reached. Thus, each point has a different bandwidth interval. This is the default bandwidth setting. The user can modify the minimum sample size. The default is 100 points.

Fixed bandwidth

A fixed bandwidth distance is a fixed interval for each point. The user must define the interval and the distance units by which it is calculated (miles, nautical miles, feet, kilometers, or meters).

Output (areal) units

Specify the areal density units as points per square mile, per squared nautical miles, per square feet, per square kilometers, or per square meters. The default is points per square mile.

Use intensity variable

If an intensity variable is being interpolated, then this box should be checked.

Use weighting variable

If a weighting variable is being used in the interpolation, then this box should be checked.

Calculate densities or probabilities

The density estimate for each cell can be calculated in one of three ways:

1. **Absolute densities**. This is the number of points per grid cell and is scaled so that the sum of all grid cells equals the sample size. This is the default.

2. **Relative densities**. For each grid cell, this is the absolute density divided by the grid cell area and is expressed in the output units (e.g., points per square mile)

3. **Probabilities**. This is the proportion of all incidents that occur in the grid cell. The sum of all grid cells equals a probability of 1

Select whether absolute densities, relative densities, or probabilities are to be output for each cell. The default is absolute densities.

Output

The results can be output as a *Surfer for Windows* file (for both an external or generated reference file) or as an *ArcGIS* 'shp', *MapInfo* 'mif', *ArcGIS Spatial Analyst* 'asc', or ASCII grid 'grd' file (only if the reference file is generated by *CrimeStat*). For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The output file is saved as K<file name> with the file name being provided by the user.

Dual Kernel Density Estimate (DualKernel)

The dual kernel density routine compares two different distributions involving the primary and secondary files. A 'first' file and 'second' file need to be defined. The comparison allows the ratio of the first file divided by the second file, the logarithm of the ratio of the first file divided by the second file, the difference between the first file and second file (i.e., first file – second file), or the sum of the first file and the second file.

File to be interpolated

Identify which file is to be the 'first file' (primary or secondary) and which is to be the 'second file (primary or secondary.) The default is Primary for the first file and Secondary for the second file.

Method of interpolation

There are five types of kernel distributions that can be used to estimate point density:

1. The **normal** kernel overlays a three-dimensional normal distribution over each point that then extends over the area defined by the reference file. This is the default kernel function.

2. The **uniform** kernel overlays a uniform function over each point that only extends for a limited distance.

3. The **quartic** kernel overlays a quartic function over each point that only extends for a limited distance.

4. The **triangular** kernel overlays a three-dimensional triangle over each point that only extends for a limited distance.

5. The **negative exponential** kernel overlays a three dimensional negative exponential function over each point that only extends for a limited distance

The methods produce similar results though the normal is generally smoother for any given bandwidth.

Choice of bandwidth

The kernels are applied to a limited search distance, called 'bandwidth'. For the normal kernel, bandwidth is the standard deviation of the normal distribution. For the uniform, quartic, triangular and negative exponential kernels, bandwidth is the radius of a circle defined by the surface. For all types, larger bandwidth will produce smoother density estimates and both adaptive and fixed bandwidth intervals can be selected.

Adaptive bandwidth

An adaptive bandwidth distance is identified by the minimum number of other points found within a circle drawn around a single point. A circle is placed around each point, in turn, and the radius is increased until the minimum sample size is reached. Thus, each point has a different bandwidth interval. This is the default bandwidth setting. The user can modify the minimum sample size. The default is 100 points.

Fixed bandwidth

A fixed bandwidth distance is a fixed interval for each point. The user must define the interval and the distance units by which it is calculated (miles, nautical miles, feet, kilometers, or meters). The default is one mile.

Variable bandwidth

A variable bandwidth allows separate fixed intervals for both the first and second files. For each, the user must define the interval and the distance units by which it is calculated (miles, nautical miles, feet, kilometers, or meters). The default is one mile for both the first and second files.

Output (areal) units

Specify the areal density units as points per square mile, per squared nautical miles, per square feet, per square kilometers, or per square meters. The default is points per square mile.

Use intensity variable

For the first and second files separately, check the appropriate box if an intensity variable is being interpolated.

Use weighting variable

For the first and second files separately, check the appropriate box if a weighting variable is being used in the interpolation.

Calculate densities or probabilities

The density estimate for each cell can be calculated in one of six ways:

1. Ratio of densities - this is the ratio of the density for the first file divided by the density of the second file
2. Log ratio of densities - this is the natural logarithm of the ratio of the density for the first file divided by the density of the second file.
3. Absolute difference in densities - this is the difference between the absolute density of the first file and the absolute density of the second file. It is the *net* difference. The densities of each file are scaled so that the sum of the grid cells equals the sample size.
4. Relative difference in densities - this is the difference between the relative density of the first file and the relative density of the second file. It is the *relative* difference. The cell densities of each file are divided by the grid cell area to produce a measure of relative density in the specified output units (e.g., points per square mile). The relative density of the second file is then subtracted from the relative density of the first file.
5. Absolute sum of densities - this is the sum of the absolute density of the first file and the absolute density of the second file. It is the *net* sum. The densities of each file are scaled so that the sum of the grid cells equals the sample size.
6. Relative sum of densities - this is the sum of the relative density of the first file and the relative density of the second file. It is the *relative* sumThe cell densities of each file are divided by the grid cell area to produce a measure of relative density in the specified output units (e.g., points per square mile). The relative density of the second file is then added to the relative density of the first file.

Select whether the ratio of densities, the log ratio of densities, the absolute difference in densities, the relative difference in densities, the absolute sum of densities, or the relative sum of densities are to be output for each cell. The default is the ratio of densities.

Output

The results can be output as a *Surfer for Windows* file (for both an external or generated reference file) or as an *ArcGIS* 'shp', *MapInfo* 'mif', *ArcGIS Spatial Analyst* 'asc', or ASCII grid 'grd' file (only if the reference file is generated by *CrimeStat*). For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The output file is saved as *DK<file name>* with the file name being provided by the user.

Interpolation II

The interpolation II tab allows the implementation of the Head Bang statistic for zonal data and its interpolation to a grid.

Head Bang

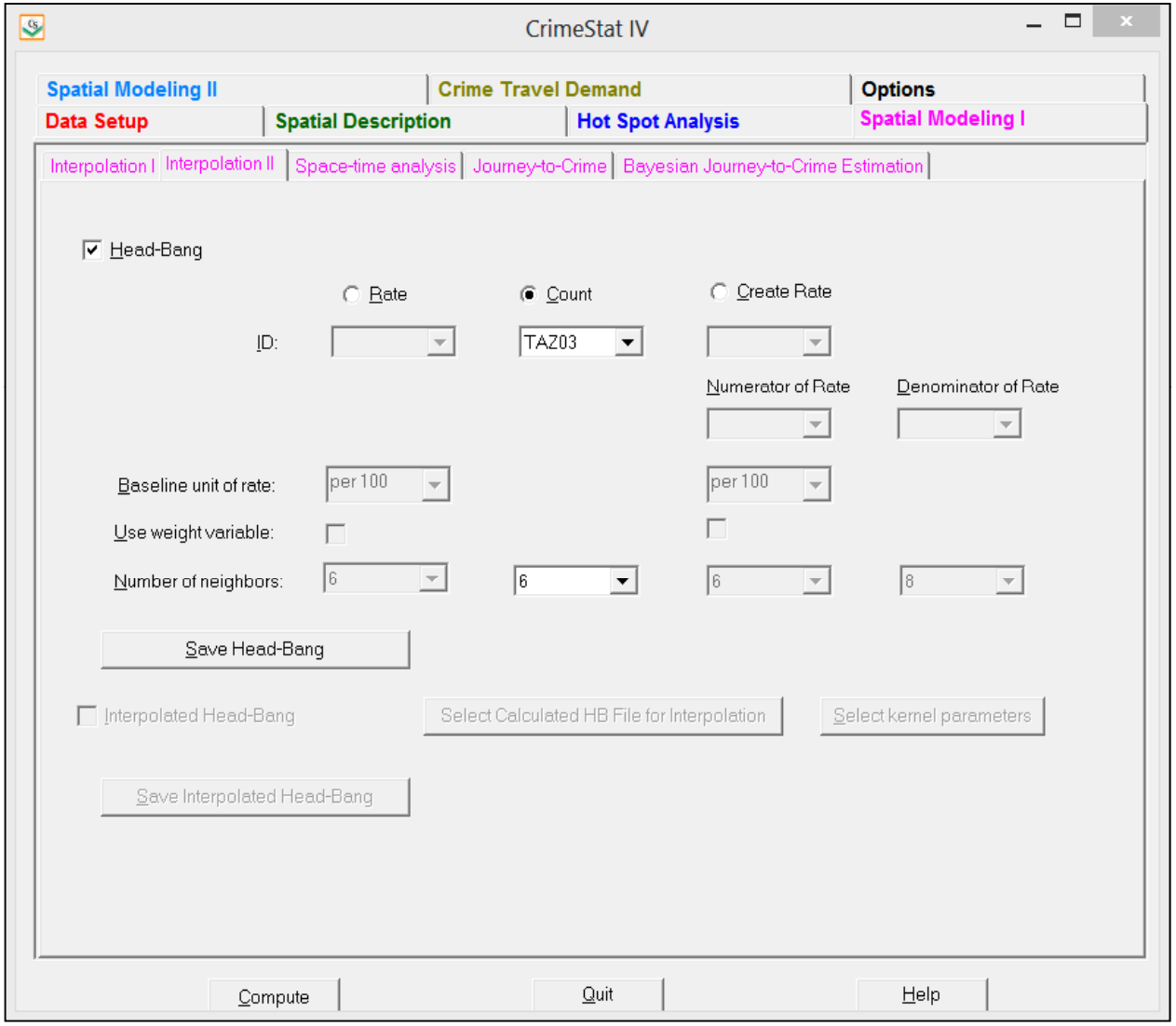
The Head Bang statistic is a weighted two-dimensional smoothing algorithm that is applied to zonal data. It is useful for eliminating extreme values in a distribution and adjusting the values of zones to be similar to their neighbors. The statistic requires an intensity variable in the primary file. The value of the intensity variable for each zone is compared to its neighbors with the number of neighbors defined by the user. The intensity values of the neighbors are rank-ordered and then divided into two equal-sized groups, high and low. The median of the high group of neighbors and the median of the low group of neighbors are calculated. The intensity value of the zone is then compared to these two medians. If it falls between the two medians, then the zone keeps its intensity value. If its value is higher than the high median, then the zone takes the high median as its value unless it has a weighting which is greater than its neighbors. If its value is lower than the low median, then the zone takes the low median as its value unless it has a weighting which is greater than its neighbors.

Type of Variable to be Smoothed

The user must specify whether the variable to be smoothed is a rate variable, a volume variable, or two variables that are to be combined into a rate.

Figure 2.13:

Interpolation II Statistics



Rate variable

If the variable to be smoothed is a rate variable, the variable that is smoothed must be defined in the Z(Intensity) field on the Primary File. Also, a weight variable should be chosen and should be defined in the Weight field on the Primary File.

ID field

The ID field that identifies zones must be defined.

Baseline unit for rate

The rate is an index of one variable relative to another variable, the baseline. Specify the unit that the rate is expressed by powers of 10. The range is from 1 (absolute rate) to 1 per 1,000,000. The default is 1 per 100 (or percentages).

Use weight variable

The rate can (and probably should) be weighted by an additional weight variable specified on the Primary File page. Check the 'Use weight variable' box to weight the rate. Otherwise, the weight is 1. A typical weight variable would be the population size of the zone.

Number of neighbors

The user must also specify the number of neighbors to be used for the comparison. The number of neighbors can run from 4 through 40. The default is 6. If the number of neighbors selected is even, the routine divides the data set into two equal-sized groups. If the number of neighbors selected is odd, then the middle zone is used in calculating both the low median and the high median.

Volume variable

If the variable to be smoothed is a volume variable, the variable that is smoothed must be defined in the Z(Intensity) field on the Primary File.

ID field

The ID field that identifies zones must be defined.

Number of neighbors

The user must also specify the number of neighbors to be used for the comparison. The number of neighbors can run from 4 through 40. The default is 6. If the number of neighbors selected is even, the routine divides the data set into two equal-sized groups. If the number of neighbors selected is odd, then the middle zone is used in calculating both the low median and the high median.

Create rate

Unlike the rate and volume calculations, the user must specify which two variables (fields) must be related to create a rate. One of these is to be defined in the *numerator of the rate* box and one in the *denominator of the rate* box. For example, if the data include number of robberies as one field in the data set and population as another field, then the number of robberies would identified as the numerator of the rate while population would be identified as the denominator of the rate. Both variables should be volumes.

Also, a weight variable should be chosen and should be defined in the Weight field on the Primary File. The weight is applied to the created rate after it is calculated. A typical weight variable would be the population size of the zone.

ID field

The ID field that identifies zones must be defined.

Baseline unit for rate

The rate is an index of one variable relative to another variable, the baseline. The result of the division of the numerator by the denominator will then be multiplied by the base unit of the baseline. Specify the unit that the rate is expressed by powers of 10. The range is from 1 (absolute rate) to 1,000,000 (resulting in an index of 1:1,000,000). The default is 100 (resulting in an index of 1:100, or percentages).

Use weight variable

The rate can (and probably should) be weighted by an additional weight variable specified on the Primary File page. Check the 'Use weight variable' box to weight the rate. Otherwise, the weight is 1. A typical weight variable would be the population size of the zone.

Number of neighbors

The user must also specify the number of neighbors to be used for the comparison. The number of neighbors can run from 4 through 40. The default is 6. If the number of neighbors selected is even, the routine divides the data set into two equal-sized groups. If the number of neighbors selected is odd, then the middle zone is used in calculating both the low median and the high median.

Output for each zone

The output is for each zone and includes:

1. The ID field
2. The X coordinate
3. The Y coordinate
4. The smoothed intensity variable (Z_MEDIAN)
5. The weight of the zone (WEIGHT). The default is 1.0.

Select output file

The tabular results can be printed, saved to a text file or saved as a 'dbf' file. For saving to a 'dbf' file, specify a file name in the "Save result to" in the dialogue box.

1. If the routine is run on a volume, then the file is saved as VolHB<file name> with the file name being provided by the user.

2. If the routine is run on a rate, then the file is saved as RateHB<file name> with the file name being provided by the user.

3. If the routine is run with a rate being created from two variables in the file, then the file is saved as CRateHB< file name> with the file name being provided by the user.

The 'dbf' file can then be linked to the input 'dbf' file by using the ID field as a matching variable. This would be done if the user wants to map the smoothed variable.

Interpolated Head Bang (IHB)

The Head Bang calculations can be interpolated to a grid. If the user checks this box, then the routine will also interpolate the calculations to a grid using kernel density estimation. An output file from the Head Bang routine is required. Also, a reference file is required to be defined on the Reference File page.

Essentially, the routine takes a Head Bang output and interpolates it to a grid using a kernel density function. The same results can be obtained by inputting the Head Bang output on the Primary File page and using the single kernel density routine on the Interpolations I page. However, there is no intensity variable in the Interpolated Head Bang because the intensity has already been incorporated in the Head Bang output. Also, there is no weighting of the Head Bang estimate.

The user must then define the parameters of the interpolation.

Method of interpolation

There are five types of kernel distributions that can be used to interpolate the Head Bang to the grid:

1. The **normal** kernel overlays a three-dimensional normal distribution over each point that then extends over the area defined by the reference file. This is the default kernel function.

2. The **uniform** kernel overlays a uniform function over each point that only extends for a limited distance.

3. The **quartic** kernel overlays a quartic function over each point that only extends for a limited distance.

4. The **triangular** kernel overlays a three-dimensional triangle over each point that only extends for a limited distance

5. The **negative exponential** kernel overlays a three dimensional negative exponential function over each point that only extends for a limited distance.

The different kernel functions produce similar results though the normal is generally smoother for any given bandwidth.

Choice of bandwidth

The kernels are applied to a limited search distance, called 'bandwidth'. For the normal kernel, bandwidth is the standard deviation of the normal distribution. For the uniform, quartic, triangular and negative exponential kernels, bandwidth is the radius of a circle defined by the surface. For all types, larger bandwidth will produce smoother density estimates and both adaptive and fixed bandwidth intervals can be selected.

Adaptive bandwidth

An adaptive bandwidth distance is identified by the minimum number of other points found within a circle drawn around a single point. A circle is placed around each point, in turn, and the radius is increased until the minimum sample size is reached. Thus, each point has a different bandwidth interval. This is the default bandwidth setting. The user can modify the minimum sample size. The default is 100 points.

Fixed bandwidth

A fixed bandwidth distance is a fixed interval for each point. The user must define the interval and the distance units by which it is calculated (miles, nautical miles, feet, kilometers, or meters).

Output (areal) units

Specify the areal density units as points per square mile, per squared nautical miles, per square feet, per square kilometers, or per square meters. The default is points per square mile.

Calculate densities or probabilities

The density estimate for each cell can be calculated in one of three ways:

1. **Absolute densities**. This is the number of points per grid cell and is scaled so that the sum of all grid cells equals the sample size. This is the default.

2. **Relative densities**. For each grid cell, this is the absolute density divided by the grid cell area and is expressed in the output units (e.g., points per square mile)
3. **Probabilities**. This is the proportion of all incidents that occur in the grid cell. The sum of all grid cells equals a probability of 1

Select whether absolute densities, relative densities, or probabilities are to be output for each cell. The default is absolute densities.

Output

The results can be output as an *ArcGIS* 'shp', *MapInfo* 'mif', *ArcGIS Spatial Analyst* 'asc', *Surfer for Windows* file (for both an external or generated reference file), or as or ASCII grid 'grd' file (only if the reference file is generated by *CrimeStat*). For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The output file is saved as IHB<file name> with the file name being provided by the user.

Space-time Analysis

The space-time analysis tab allows the analysis of the interaction between space and time. There are four routines. First, there is the Knox index that shows the simple binomial relationship between events occurring in space and in time. Second, there is the Mantel index that shows the correlation between closeness in space and closeness in time. Third, there is a spatial-temporal moving average that calculates a mean center for a temporal span. Fourth, there is a Correlated Walk Analysis that diagnoses the spatial and temporal sequencing of incidents committed by a serial offender.

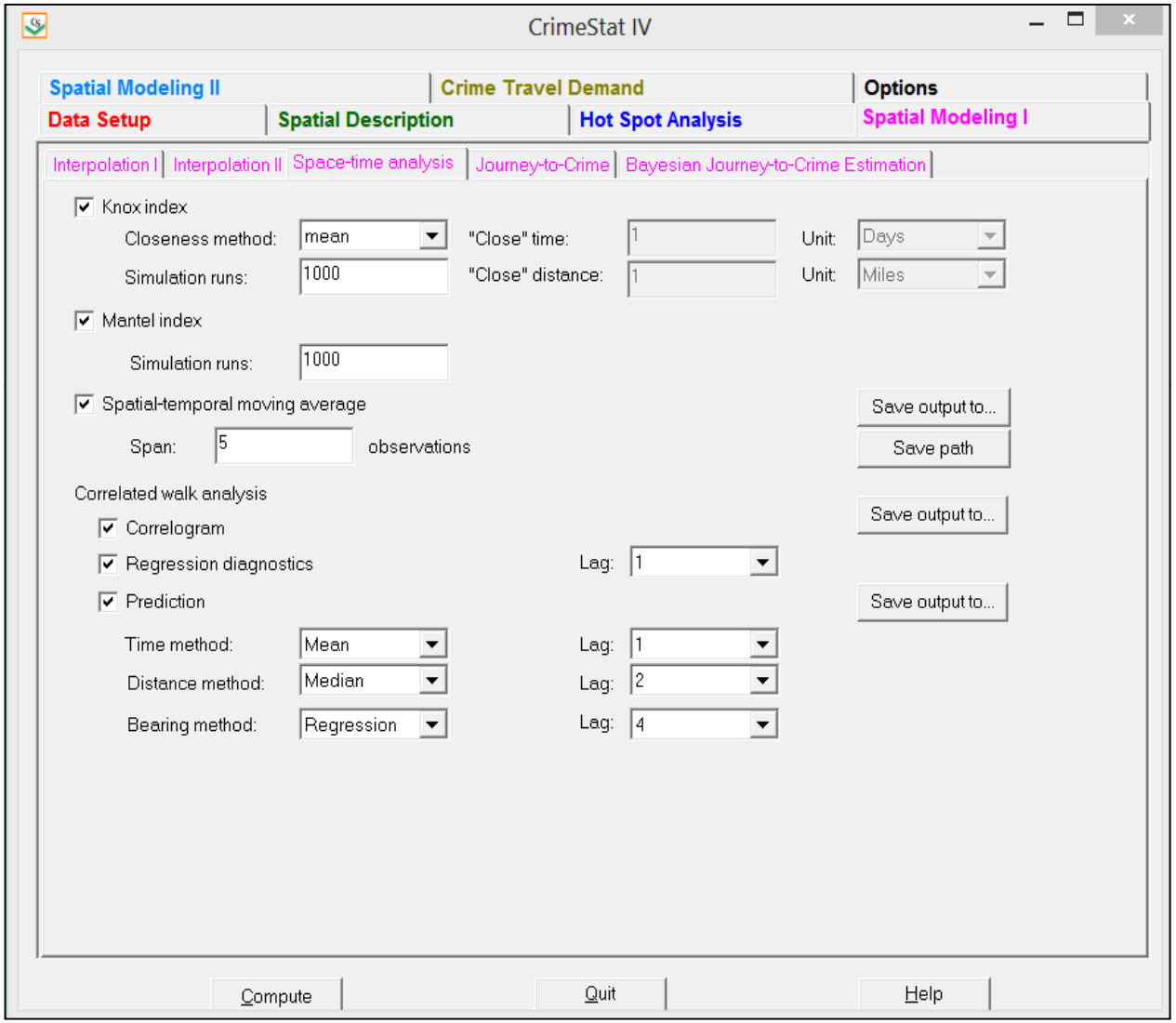
For each of these routines, time **must** be defined by an integer or real variable, and **not** by a formatted date. For example, 3 days, 2.1 weeks, 4.3 months, or the number of days from January 1, 1900 (e.g., 37174) are all eligible time values. 'November 1, 2001', '07/30/01' or '19th October, 2001' are not eligible values. Convert all formatted dates into a real number. Time units must be consistent across all observations (i.e., all values are hours or days or weeks or months or years, but not two or more these units). If these conditions are violated, *CrimeStat* will calculate results, but they won't be correct.

Knox Index

The Knox index is an index showing the relationship between 'closeness in time' and 'closeness in distance'. Pairs of events are compared in distance and in time and are represented

Figure 2.14:

Space-Time Analysis



as a 2 x 2 table. If there is a relationship, it would normally be positive, that is events that are close together in space (i.e., in distance) are also occurring in a short time span. There are three methods for defining closeness in time or in distance:

1. Mean. That is, events that are closer together than the mean time interval or are closer together than the mean distance are defined as 'Close' whereas events that are father together than mean time interval or are farther together than the mean distance are defined as 'Not close'. This is the default.

2. Median. That is, events that are closer together than the median time interval or are closer together than the median distance are defined as 'Close' whereas events that are father together than median time interval or are farther together than the median distance are defined as 'Not close'.

3. User defined. The user can specify any value for distinguishing 'Close' and 'Not close' for either time or distance.

The output includes a 2 x 2 table of the distribution of pairs categorized as 'Close' or 'Not close' in time and in distance. Note, that since pairs of events are being compared, there are N*(N-1)/2 pairs in a data set where N is the number of events. The output also includes a table of the expected of the distribution of pairs on the assumption that events in time are space are independent of each other. Finally, the output includes a chi-square test of the differences between the observed and expected distributions. Note, that since pairs are being compared, independence of observations is not true and a usual p-value associated with the chi-square test cannot be properly calculated.

Simulating confidence intervals

A Monte Carlo simulation can be run to estimate the approximate Type I error probability levels for the Knox index. The user specifies the number of simulation runs. Data are randomly assigned and the chi-square value for the Knox index is calculated for each run. The random output is sorted and percentiles are calculated. Twelve percentiles are identified for this index:

1. The minimum for the spatially random Knox chi-square
2. The maximum for the spatially random Knox chi-square
3. The 0.5 percentile for the spatially random Knox chi-square
4. The 1 percentiles for the spatially random Knox chi-square
5. The 2.5 percentile for the spatially random Knox chi-square
6. The 5 percentile for the spatially random Knox chi-square

7. The 10 percentile for the spatially random Knox chi-square
8. The 90 percentile for the spatially random Knox chi-square
9. The 95 percentile for the spatially random Knox chi-square
10. The 97.5 percentile for the spatially random Knox chi-square
11. The 99 percentile for the spatially random Knox chi-square
12. The 99.5 percentile for the spatially random Knox chi-square

Confidence intervals can be estimated from these percentiles. The two most commonly used ones are the 95% (defined by the 2.5 and 97.5 percentiles) and the 99% (defined by the 0.5 and 99.5 percentiles). The simulated data that is used can be viewed by checking the 'Dump simulation data' box on the Options tab.

Mantel Index

The Mantel index is the correlation between closeness in time and closeness in distance across pairs. Each pair of events is compared for the time interval and the distance between them. If there is a positive relationship between closeness in time and closeness in space (distance), then there should be a sizeable positive correlation between the two measures. Note, that since pairs of events are being compared, there are N*(N-1)/2 pairs in the data set where N is the number of events.

Simulating confidence intervals

A Monte Carlo simulation can be run to estimate the approximate confidence intervals around the Mantel correlation. The user specifies the number of simulation runs and the Mantel index is calculated for randomly assigned data. The random output is sorted and percentiles are calculated. Twelve percentiles are identified for this index:

1. The minimum for the spatially random Mantel index
2. The maximum for the spatially random Mantel index
3. The 0.5 percentile for the spatially random Mantel index
4. The 1 percentiles for the spatially random Mantel index
5. The 2.5 percentile for the spatially random Mantel index
6. The 5 percentile for the spatially random Mantel index
7. The 10 percentile for the spatially random Mantel index
8. The 90 percentile for the spatially random Mantel index
9. The 95 percentile for the spatially random Mantel index
10. The 97.5 percentile for the spatially random Mantel index
11. The 99 percentile for the spatially random Mantel index

12. The 99.5 percentile for the spatially random Mantel index

Confidence intervals can be estimated from these percentiles. The two most commonly used ones are the 95% (defined by the 2.5 and 97.5 percentiles) and the 99% (defined by the 0.5 and 99.5 percentiles). The simulated data that is used can be viewed by checking the 'Dump simulation data' box on the Options tab.

Spatial-Temporal Moving Average

This routine calculates the mean center as it changes over the sequence of the events. The routine sorts the incidents in the order in which they occur. The user defines a *span* of sequential incidents; the default is five observations. The routine places a window covering the span over the incidents and calculates the mean center (the mean X coordinate and the mean Y coordinate). It then moves the window one observation. Approximations are made at the beginning and end observations for the sequence. The result is a set of mean centers ordered from the first through last observations. This statistic is useful for identifying whether the central location for a set of incidents (perhaps committed by a serial offender) has moved over time.

There are four outputs for this routine:

1. The sample size
2. The number of observations making up the span
3. The span number
4. The X and Y coordinates for each span window.

The tabular results are output as a dBase ‘dbf’ file. 0020A line showing the sequential output cal also be output as an *ArcGIS* ‘shp’, *MapInfo* ‘mif’ or ‘bna’ ASCII formats. For MapInfo ‘mif’ format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected. The object will be output with a “STMA” prefix.

Correlated Walk Analysis

The Correlated Walk Analysis (CWA) analyzes the sequential movements of a serial offender and makes predictions about the time and location of the next event. Sequential movements are analyzed in terms of three parameters: Time difference between events (e.g., the number of days between two consecutive events), Distance between events – the distance between

two consecutive events, and Bearing (direction) between events – the angular direction between two consecutive events in degrees (from 0 to 360).

There are three CWA routines for analyzing sequential events:

1. Correlogram (CWA-C)
2. Regression diagnostics (CWA-D)
3. Prediction (CWA-P)

Correlated Walk Analysis Correlogram (CWA-C)

The correlogram presents the lagged correlations between events for time difference, distance, and bearing (direction). The lags are the sequential comparisons. A lag of 0 is the sequence compared with itself; by definition, the correlation is 1.0. A lag of 1 is the sequence compared with the previous sequence. A lag of 2 is the sequence compared with two previous sequences. A lag of 3 is the sequence compared with three previous sequences, and so forth. In total, comparisons are made up to seven previous sequences (a lag of 7).

Typically, for time difference, distance and location separately, the lag with the highest correlation is the strongest. However, with each consecutive lag, the sample size decreases by one and a high correlation associated with a high lag comparison can be unreliable if the sample size is small. Consequently, the adjusted correlogram discounts the correlations by the number of lags.

The CWA correlogram is output as a dBase 'dbf' file.

Correlated Walk Analysis Regression Diagnostics (CWA-D)

The regression diagnostics presents the regression statistics for different lag models. The lag must be specified; the default is a lag of 1 (the sequential events compared with the previous events). Three regression models are run for time difference, direction, and bearing. The output includes statistics for:

1. The sample size
2. The distance and time units
3. The lag of the model (from 1 to 7)
4. The multiple R (correlation) between the lags
5. The squared multiple R (i.e., R-squared)
6. The standard error of estimate for the regression

7. The coefficient, standard error, t-value, and probability value (two-tail) for the constant.
8. The coefficient, standard error, t-value, and probability value (two-tail) for the coefficient.
9. The analysis of variance for the regression model, including the sum-of-squares and the mean-square error for the regression model and the residual (error), the F-test of the regression mean-square error divided by the residual mean-square error, and the probability level for the F-test.

In general, the model with the lowest standard error of estimate (and, consequently, highest multiple R) is best. However, with a small sample size, the model can be unreliable. Further, with each consecutive lag, the sample size decreases by one and a high multiple R associated with a high lag comparison can be unreliable if the sample size is small.

Correlated Walk Analysis Prediction (CWA-P)

The prediction routine allows the prediction of a next event, in time, distance, and direction. For each parameter – time difference, distance, and bearing, there are three models that can be used:

1. The mean difference (i.e., use mean time difference, mean distance, mean bearing)

2. The median difference (i.e., use median time difference, median distance, median bearing)

3. The regression model (i.e., use the estimated regression coefficient and intercept)

For each of these, a different lag comparison can be used, from 1 to 7. The lag defines the sequence from which the prediction is made. Thus, for a lag of 1, the interval from the next-to-last to the last event is used as a reference (i.e., between events N-1 and N); for a lag of 2, the interval from the third-to-last to the next-to-last event is used as a reference (i.e.,between events N-2 and N-1); and so forth. The particular model selected is then added to the reference sequence.

Example 1: with a lag of 1 and the use of the mean difference, the mean time difference is added to the time of the last event, the mean distance is added to the location of the last event, and the mean bearing is added to the location of the last event.

Example 2: with a lag of 2 and the use of the regression model, the predicted time difference is added to the time of the next-to-last event; the predicted distance is added to the location of the next-to-last event and the prediction bearing is added to the location of the last event. Note: if the regression model is used, the lag for distance and bearing must be the same.

Example 3: with a lag of 1 for time, a lag of 2 for distance and the use of the mean distance, and a lag of 3 for bearing and the use of the median bearing, the predicted time difference is added to the last event, the mean distance is added to the location of the next-to-last event, and the median bearing is added to the location of the third-from-last event.

Tabular output

The tabular output includes:

1. The method used for time, distance, and bearing
2. The lag used for time, distance, and bearing
3. The predicted time difference
4. The predicted distance
5. The predicted bearing
6. The final predicted time
7. The X-coordinate of the final predicted location
8. The Y-coordinate of the final predicted location

Graphical output

If the user specifies an output file name, there are five graphical objects that are output as an *ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats:

1. The sequence of incidents from the first to the last. This object has a prefix of 'Events' before the file name provided by the user.
2. The predicted location of the next event. This is the event after the last in the input sequence. This object has a prefix of 'Preddest' before the file name.
3. The predicted path between the last event in the sequence and the expected next event. This object has a prefix of 'Pw' before the file name.
4. The center of minimum distance for the sequence of events. This is the single best measure of the likely origin location of the offender

5. The expected path between the center of minimum distance and the predicted location of the next event. This is a guess about the likely origin and likely destination for a next event by the offender.

For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Journey to Crime Estimation (Jtc)

The journey to crime (Jtc) routine estimates the likelihood that a serial offender lives at any location within the study area. Both a primary file and a reference file are required. The locations of the serial crimes are defined in the primary file while all locations within the study area are identified in the reference file. The Jtc routine can use two different travel distance functions: 1) An already-calibrated distance function; and 2) A mathematical formula. Either direct or indirect (Manhattan) distances can be used though the default is direct (see Measurement parameters.)

Calibrate Journey to Crime Function

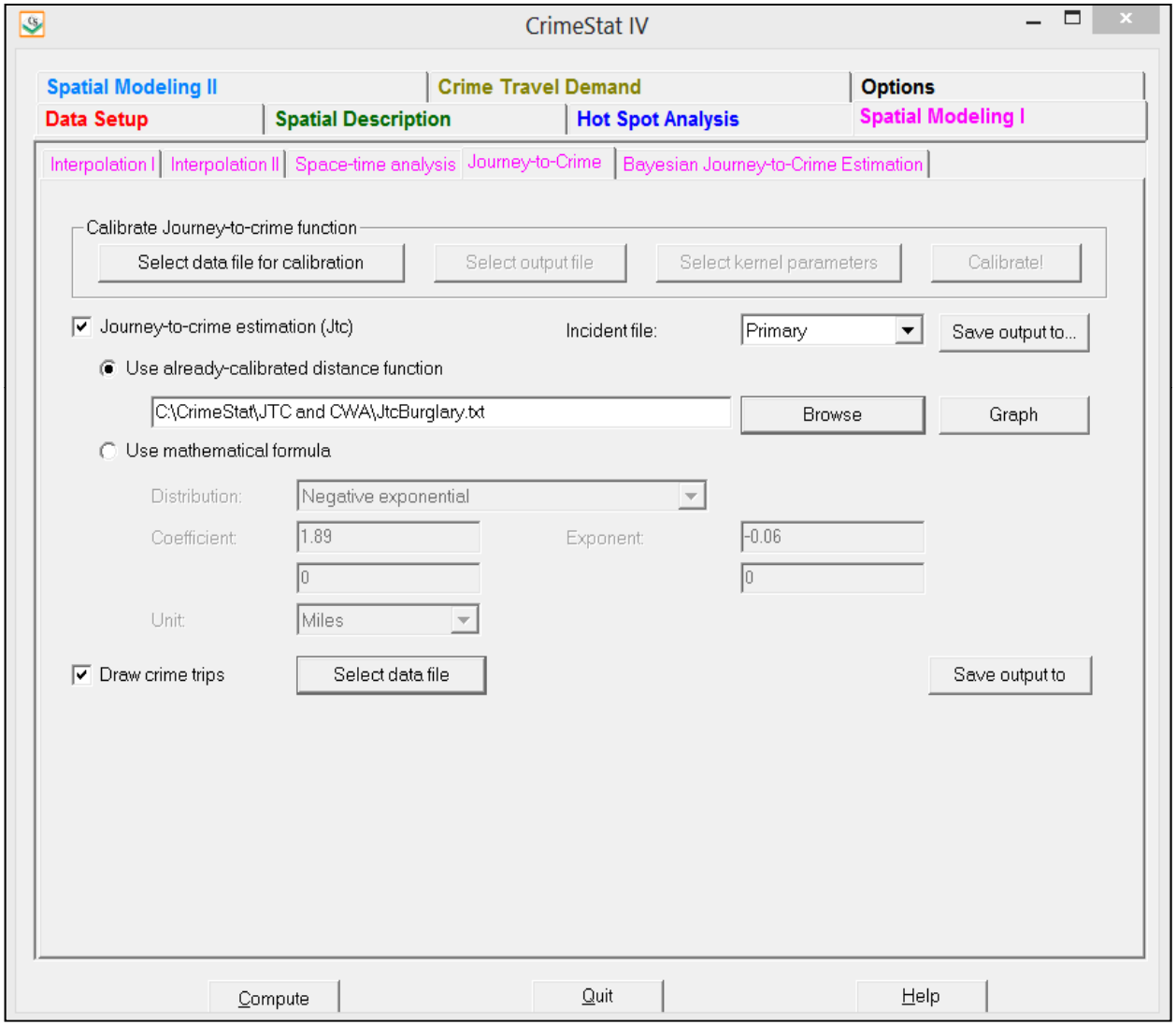
This routine calibrates a journey to crime distance function for use in the estimation routine. A file is input which has a set of incidents (records) that includes both the X and Y coordinates for the location of the offender's residence (origin) and the X and Y coordinates for the location of the incident that the offender committed (destination.) The routine estimates a travel distance function (trip lengths) using a one-dimensional kernel density method. For each record, the distance between the origin location and the destination location is calculated and is represented on a distance scale. The maximum distance is calculated and divided into a number of intervals; the default is 100 equal sized intervals, but the user can modify this. For each distance (point) calculated, a one-dimensional kernel is overlaid. For each distance interval, the values of all kernels are summed to produce a smooth function of journey to crime distance. The results are saved to a file that can be used in the journey to crime estimation routine.

Select data file for calibration

Select the file that has the X and Y coordinates for the origin and destination locations. *CrimeStat* reads dbase 'dbf', ArcGIS point 'shp' and ASCII files. Select the tab and specify the type of file to be selected. Use the browse button to search for the file. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns.

Figure 2.15:

Journey-to-crime Analysis



Variables

Define the file which contains the X and Y coordinates for both the origin (residence) and destination (crime) locations

Columns

Select the variables for the X and Y coordinates respectively for *both* the origin and destination locations (e.g., Lon, Lat, HomeX, HomeY, IncidentX, IncidentY.) Both locations must be defined for the routine to work.

Missing values

Identify whether there are any missing values for these four fields (X and Y coordinates for both origin and destination locations). By default, *CrimeStat* will ignore records with blank values in any of the eligible fields or records with non-numeric values (e.g., alphanumeric characters, , *). Blanks will always be excluded unless the user selects **<*none*>**. There are 8 possible options:

1. <*blank*> fields are automatically excluded. This is the default
2. <*none*> indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0
3. *0* is excluded
4. *–1* is excluded
5. *0 and –1* indicates that both 0 and -1 will be excluded
6. *0, -1 and 9999* indicates that all three values (0, -1, 9999) will be excluded
7. *Any* other numerical value can be treated as a missing value by typing it (e.g., 99)
8. *Multiple* numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99)

Type of coordinate system and data units

Select the type of coordinate system. If the coordinates are in longitudes and latitudes, then a spherical system is being used and data units will automatically be decimal degrees. If the coordinate system is projected (e.g., State Plane, Universal Transverse Mercator – UTM), then data units could be either in feet (e.g., State Plane) or meters (e.g., UTM.) Directional coordinates are not allowed for this routine.

Select kernel parameters

There are five parameters that must be defined.

Method of interpolation

There are five types of kernel distributions that can be used to estimate the distance decay density of the trip lengths:

1. The **normal** kernel overlays a three-dimensional normal distribution over each point that then extends over the area defined by the reference file. This is the default kernel function.

2. The **uniform** kernel overlays a uniform function over each point that only extends for a limited distance.

3. The **quartic** kernel overlays a quartic function over each point that only extends for a limited distance.

4. The **triangular** kernel overlays a three-dimensional triangle over each point that only extends for a limited distance.

5. The **negative exponential** kernel overlays a three dimensional negative exponential function over each point that only extends for a limited distance

The methods produce similar results though the normal is generally smoother for any given bandwidth.

Choice of bandwidth

The kernels are applied to a limited search distance, called 'bandwidth'. For the normal kernel, bandwidth is the standard deviation of the normal distribution. For the uniform, quartic, triangular and negative exponential kernels, bandwidth is the radius of a circle defined by the surface. For all types, larger bandwidth will produce smoother density estimates and both adaptive and fixed bandwidth intervals can be selected.

Fixed bandwidth

A fixed bandwidth distance is a fixed interval for each point. The user must define the interval, the interval size, and the distance units by which it is calculated (miles, nautical miles, feet, kilometers, or meters). The default bandwidth setting is fixed with intervals of 0.25 miles each. The interval size can be changed.

Adaptive bandwidth

An adaptive bandwidth distance is identified by the minimum number of other points found within a symmetrical band drawn around a single point. A symmetrical band is placed over each distance point, in turn, and the width is increased until the minimum sample size is reached. Thus, each point has a different bandwidth size. The user can modify the minimum sample size. The default for the adaptive bandwidth is 100 points.

Specify interpolation bins

The interpolation bins are defined in one of two ways:

1. By the number of bins. The maximum distance calculated is divided by the number of specified bins.This is the default with 100 bins. The user can change the number of bins

2. By the distance between bins. The user can specify a bin width in miles, nautical miles, feet, kilometers, and meters

Output (areal) units

Specify the areal density units as points per mile, nautical mile, foot, kilometer, or meter. The default is points per mile.

Calculate densities or probabilities

The density estimate for each cell can be calculated in one of three ways:

1. **Absolute densities**. This is the number of points per grid cell and is scaled so that the sum of all grid cells equals the sample size. This is the default.

2. **Relative densities**. For each grid cell, this is the absolute density divided by the grid cell area and is expressed in the output units (e.g., points per square mile)

3. **Probabilities**. This is the proportion of all incidents that occur in the grid cell. The sum of all grid cells equals a probability of 1.

Select whether absolute densities, relative densities, or probabilities are to be output for each cell. The default is absolute densities.

Select output file

The output *must* be saved to a file. *CrimeStat* can save the calibration output to either a dbase 'dbf' or ASCII text 'txt' file.

Calibrate!

Click on 'Calibrate!' to run the routine. The output is saved to the specified file upon clicking on 'Close'. The output file is saved as JtcCalib<file name> with the file name being provided by the user.

Graph of journey to crime travel function

Click on 'View graph' to see the journey crime travel distance function (journey to crime likelihood by distance.) The screen view can be printed by clicking on 'Print'. For a better quality graph, however, the output should be imported into a graphics package.

Journey to Crime Estimation (Jtc)

The journey to crime (Jtc) routine estimates the likelihood that a serial offender lives at any location within the study area. Both a primary file and a reference file are required. The locations of the serial crimes are defined in the primary file while all locations within the study area are identified in the reference file. The Jtc routine can use two different travel distance functions: 1) An already-calibrated distance function; and 2) A mathematical formula.

Use an already-calibrated distance function

If a travel distance function has already been calibrated (see 'Calibrate journey to crime function'), the file can be directly input into the Jtc routine.

Input

The user selects the name of the already-calibrated travel distance function. *CrimeStat* reads dbase 'dbf'', ArcGIS 'shp' and ASCII text files.

Output

The Jtc routine calculates a relative likelihood estimate for each cell of the reference file. Higher values indicate higher relative likelihoods. The results can be output as a *Surfer for Windows* file (for both an external or generated reference file) or as an *ArcGIS* 'shp', *MapInfo* 'mif', *ArcGIS Spatial Analyst* 'asc', or ASCII grid 'grd' file (only if the reference file is generated by *CrimeStat)*. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The output file is saved as Jtc<file name> with the file name being provided by the user.

Use a mathematical formula

A mathematical formula can be used instead of a calibrated distance function. To do this, it is necessary to specify the type of distribution. There are five mathematical models that can be selected:

1. Negative exponential
2. Normal
3. Lognormal
4. Linear
5. Truncated negative exponential

The normal is the default. For each mathematical model, two or three different parameters must be defined:

1. For the negative exponential, the coefficient and exponent
2. For the normal distribution, the mean distance, standard deviation and coefficient
3. For the lognormal distribution, the mean distance, standard deviation and coefficient
4. For the linear distribution, an intercept and slope
5. For the truncated negative exponential, a peak distance, peak likelihood, intercept, and exponent

Output

The Jtc estimation routine calculates a relative likelihood estimate for each cell of the reference file. Higher values indicate higher relative likelihoods. The results can be output as a *Surfer for Windows* file (for both an external or generated reference file) or as an *ArcGIS* 'shp', *MapInfo* 'mif', *ArcGIS Spatial Analyst* 'asc', or ASCII grid 'grd' file (only if the reference file is generated by *CrimeStat*). For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The output file is saved as Jtc<file name> with the file name being provided by the user.

Draw Crime Trips

This routine is a utility for both the Journey to crime routine and the Trip Distribution routine (in the Crime Travel Demand module). If given a file with origins and destinations, the routine will draw a line between the origin and destination for each record. It is useful for examining the actual trip links made by an offender.

Select data file

Select the file that has the X and Y coordinates for the origin and destination locations. *CrimeStat* reads dbase 'dbf', ArcGIS point 'shp' and ASCII files. Select the tab and specify the type of file to be selected. Use the browse button to search for the file. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns.

Variables

Define the file which contains the X and Y coordinates for both the origin (residence) and destination (crime) locations

Columns

Select the variables for the X and Y coordinates respectively for *both* the origin and destination locations (e.g., Lon, Lat, HomeX, HomeY, IncidentX, IncidentY.) Both locations must be defined for the routine to work.

Type of coordinate system and data units

Select the type of coordinate system. If the coordinates are in longitudes and latitudes, then a spherical system is being used and data units will automatically be decimal degrees. If the coordinate system is projected (e.g., State Plane, Universal Transverse Mercator – UTM), then data units could be either in feet (e.g., State Plane) or meters (e.g., UTM.) Directional coordinates are not allowed for this routine.

Save output to

The graphical results can be output as lines in *ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Bayesian Journey to Crime Estimation (Jtc)

The Bayesian Journey-to-crime module (BJtc) is a tool for estimating the likely residence location of a serial offender. It is an extension of the Journey-to-crime routine (Jtc) which uses a travel distance function to make guesses about the likely residence location. The Bayesian Journey to crime routine estimates the likelihood that a serial offender lives at any location within the study area using two pieces of information: 1) the distribution of incidents committed by the offender; and 2) the distribution of origins by other offenders who committed crimes in the same location as the offender, based on an origin-destination matrix.

A travel distance function is applied to the distribution of incidents to produce one estimate of the likely origin of the offender while an origin-destination matrix is used to produce another estimate of the likely origin of the offender based on the origins of other offenders who committed crimes in the same locations. Both estimates can be combined in several ways to produce a joint estimate of the likely origin of the offender.

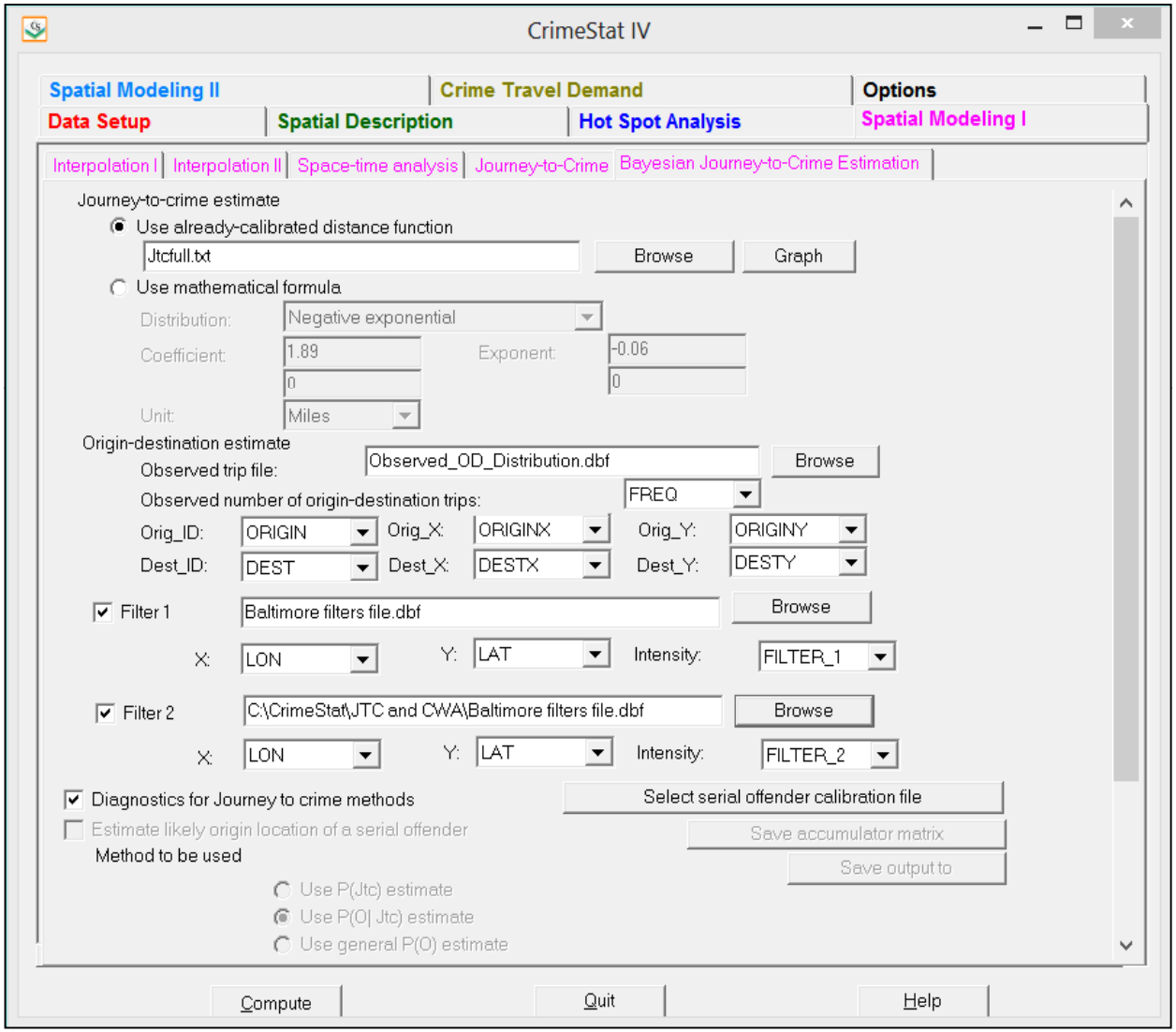
There are two routines in the Bayesian Journey to Crime module:

1. Diagnostics for comparing different journey-to-crime methods; and

2. A routine for estimating the likely origin of a serial offender using a selected journey-to-crime method.

Figure 2.16:

Bayesian Journey-to-crime Analysis



The routines are applications of Bayes Theorem to Journey to Crime estimation.

Bayes Theorem

Bayes Theorem is defined as:

$$P(B|A) = \frac{P(B)*P(A|B)}{P(A)} \quad (2.1)$$

where P(B|A) is the probability of event B given event A (the conditional probability of B given A), P(B) is the simple probability of event B, P(A|B) is the probability of event A given event B (the conditional probability of A given B), and P(A) is the probability of event A.

Bayesian Inference

In the statistical interpretation of Bayes Theorem, the probabilities are estimates of a random variable. Let θ be a parameter of interest and let X be some data. Thus, Bayes Theorem can be expressed as:

$$P(\theta|X) = \frac{P(\theta)*P(X|\theta)}{P(X)} \quad (2.2)$$

where P(θ|X) is the posterior probability of θ given the data, *X*, and P(θ) is the probability that θ has a certain distribution and is often called the *prior probability*. P(X|θ) is the probability that the data would be obtained given that θ is true and is often called the *likelihood function* (i.e., it is the likelihood that the data will be obtained given the distribution of θ). Finally, P(X) is the marginal probability of the data, the probability of obtaining the data under all possible scenarios; essentially, it is the data.

The equation can be rephrased in logical terms:

$$\text{The posterior probability that } \theta \text{ is true given the data, X} = \frac{\text{Likelihood of obtaining the data given } \theta \text{ is true} * \text{Prior probability of } \theta}{\text{Marginal probability of X}} \quad (2.3)$$

In other words, this formulation allows an estimate of the probability of a particular parameter, θ, to be updated given new information. Since θ is the prior probability of an event, given some new data, X, Bayes Theorem can be used to update the estimate of θ. The prior

probability of θ can come from prior studies, an assumption of no difference between any of the conditions affecting θ, or an assumed mathematical distribution. The likelihood function can also come from empirical studies or an assumed mathematical function. Irrespective of how these are interpreted, the result is an estimate of the parameter, θ, given the evidence, X. This is called the *posterior probability* (or posterior distribution).

Application of Bayesian inference to Journey to Crime Estimation

Applying Bayesian inference to journey to crime estimation, there are three different estimates of where an offender lives:

1. An estimate of the residence location of a single offender based on the location of the incidents that this person committed and an assumed travel distance function, P(Jtc).

2. An estimate of the residence location of a single offender based on a general distribution of all offenders, irrespective of any particular destinations for incidents, P(O). Essentially, this is the distribution of origins irrespective of the destinations.

3. An estimate of the residence location of a single offender based on the distribution of offenders given the distribution of incidents committed by the single offender, P(O|Jtc).

The Bayesian formula can now be approximated by:

$$P(Jtc|O) \approx \frac{P(O|Jtc)*P(Jtc)}{P(O)} \quad (2.4)$$

where P(Jtc|O) is the probability that a particular serial offender lives at any one location given both an estimate of where the offender lives given a travel distance function and an estimate of where an offender lives given the distribution of origins by other offenders who committed crimes in the same locations. The numerator expresses this relationship and is called the *Bayesian product term*. Since obtaining the probability of the data under all scenarios is virtually impossible to estimate, the equation is an approximation, relating this product term to the distribution of all offenders, P(O). This is called *Bayesian risk*.

The Bayesian Journey to Crime Estimation Module

The Bayesian Journey-to-crime estimation module is made up of two routines, one for diagnosing which Journey-to-crime method is best and one for applying that method to a particular serial offender.

Data Preparation for Bayesian Journey to Crime Estimation

There are three sets of data that are required and one optional data set. The three required ones are:

1. The incidents committed by a single offender for which an estimate will be made of where that individual lives

2. A journey-to-crime function that estimates the likelihood of an offender committing crimes at a certain distance (or travel time if a network is used)

3. An origin-destination matrix

The fourth, optional data set is a diagnostics file of multiple known serial offenders for which both their residence and crime locations are known.

Both a primary file and a reference file are also required. For the Bayesian Jtc Diagnostics routine, any point file can be used as the primary file. For the Bayesian Jtc Estimation routine, the primary file should be the locations of the crimes committed by the single serial offender for whom the estimate is being obtained. The reference file also needs to be defined and should include all locations where crimes have been committed (see Reference File).

Serial offender data

For each serial offender for whom an estimate will be made of where that person lives, the data set should include the location of the incidents committed by the offender. The data are set up as a series of records in which each record represents a single event. On each data set, there are X and Y coordinates identifying the location of the incidents this person has committed.

Journey-to-crime travel function

The Journey-to-crime function is an estimate of the likelihood of an offender traveling a certain distance. Typically, it represents a frequency distribution of distances traveled, though it

could be a frequency distribution of travel times if a network was used to calibrate the function with the Journey to crime estimation routine (see Journey to crime estimation). It can come from an a priori assumption about travel distances, prior research, or a calibration data set of offenders who have already been caught. The "Calibrate journey-to-crime function" routine (on the Journey-to-crime page under Spatial modeling) can be used to estimate this function.

The BJtc routine can use two different travel distance functions: 1) An already-calibrated distance function; and 2) A mathematical formula. Either direct or indirect (Manhattan) distances can be used though the default is direct (see Measurement parameters.)

Origin-destination matrix

The origin-destination matrix relates the number of offenders who commit crimes in one of N zones who live (originate) in one of M zones. It can be created from the "Calculate observed origin-destination trips" routine (on the 'Describe origin-destination trips' page under the Trip distribution module of the Crime Travel Demand model).

Diagnostics file for Bayesian Jtc routine

The aim of the diagnostics file is to provide information to the analyst about which of several parameters (to be described below) are best at guessing where an offender lives. The assumption is that if a particular parameter was best with the K offenders in a diagnostics file in which the residence location was known, then the same parameter will also be best for a serial offender for whom the residence location is not known.

How many serial offenders are needed to make up a diagnostics file? There is no simple answer to this. Clearly, the more, the better since the aim is to identify which parameter is most sensitive with a certain level of precision and accuracy. Certainly, a minimum of 10 would be necessary. But, more would certainly be more accurate. Further, the offender records used in the diagnostics file should be similar in other dimensions to the offender that is being tracked. However, this may be impractical.

Once the data sets have been collected, they need to be placed in an *appended* file, with one serial offender on top of another. Each record has to represent a single incident. Further, the records have to be arranged sequentially with all the records for a single offender being grouped together. The routine automatically sorts the data by the offender ID. But, to be sure that the result is consistent, the data should be prepared in this way.

Regarding the fields in each record, at the minimum there is a need for an ID field, and the X and Y coordinates of both the crime location and the residence location. The ID field is any string variable.

Diagnostics for Journey to Crime Methods

The following applies to the "diagnostics" routine only.

Data Input

The user inputs the five required data sets and two optional data sets.

1. Any primary file with an X and Y location. A suggestion is to use one of the files for the serial offender, but this is not essential
2. A grid that will be overlaid on the study area. Use the Reference File under Data setup to define the X and Y coordinates of the lower-left and upper-right corners of the grid as well as the number of columns
3. A journey-to-crime function that estimates the likelihood of an offender committing crimes at a certain distance (or travel time if a network is used)
4. An origin-destination matrix
5. The diagnostics file of known serial offenders in which both their residence and crime locations are known
6. (Optional) A data set that includes a filter variable (see below)
7. (Optional) A data set that includes a second filter variable (see below)

Methods Tested

The "diagnostics" routine compares seven methods for estimating the likely location of a serial offender:

1. The Journey-to-crime distance method, P(Jtc).
2. The general crime distribution based on the origin-destination matrix, P(O). Essentially, this is the distribution of origins irrespective of the destinations.
3. The distribution of origins based only on the incidents committed by the serial offender, P(O|Jtc).
4. The product of the Journey-to-crime estimate (1 above) and the distribution of origins based only on the incidents committed by the serial offender (3 above), P(Jtc)*P(O|Jtc). This is the numerator of the Bayesian function, the product of the prior probability times the likelihood estimate.

5. The simple average of the Journey-to-crime estimate (1 above) and the distribution of origins based only on the distribution of incidents committed by the serial offender (3 above), P(Jtc) + P(O|Jtc). This is an alternative to the product term (4 above).
6. The Bayesian risk estimate as indicated in the discussion above (method 4 above divided by method 2 above), P(Bayes risk).
7. The center of minimum distance, Cmd. Previous research has indicated that the center of minimum of distance produces the least error in minimizing the distance between where the method predicts the most likely location for the offender and where the offender actually lives.

Interpolated Grid

With each serial offender, in turn, and with each method, the routine overlays a grid over the study area. The grid is defined by the Reference File parameters (see Data setup). The routine then interpolates each input data set into a probability estimate for each grid cell with the sum of the cells equaling 1.0 (within three decimal places). The manner in which the interpolation is done varies by the method:

1. For the Journey-to-crime method, P(Jtc), the routine interpolates the selected distance function to each grid cell to produce a density estimate. The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

2. For the general crime distribution method, P(O), the routine sums up the incidents by each origin zone from the origin-destination matrix and interpolates that using the normal distribution method of the single kernel density routine (see Kernel Density Interpolation). The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

3. For the distribution of origins based only on the incidents committed by the serial offender, from the origin-destination matrix the routine identifies the zone in which the incidents occur and reads only those origins associated with those destination zones. Multiple incidents committed in the same origin zone are counted multiple times. The routine then uses the single kernel density routine to interpolate the distribution to the grid (see Kernel Density Interpolation). The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

4. For the product of the Journey-to-crime estimate and the distribution of origins based only on the incidents committed by the serial offender, the routine multiples the

probability estimate obtained in 1 above by the probability estimate obtained in 3 above. The product density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

5. For the simple average of the Journey-to-crime estimate and the distribution of origins based only on the incidents committed by the serial offender, the routine adds the probability estimate obtained in 1 above to the probability estimate obtained in 3 above and divides by two. The average density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

6. For the Bayesian risk estimate, the routine takes the product estimate (4 above) and divides it by the general crime distribution estimate (2 above). The resulting density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

7. Finally, for the center of minimum distance estimate, the routine calculates the center of minimum distance for each serial offender in the "diagnostics" file and calculates the distance between this statistic and the location where the offender is actually residing. This is used only for the distance error comparisons.

Note in all of the probability estimates (excluding 7), the cells are converted to probabilities prior to any multiplication or division. The results are then re-scaled so that the resulting grid is a probability (i.e., all cells sum to 1.0).

Additional Filtering

A filter is a probability matrix that is applied to the estimate but is not conditioned on the existing variables in the model. For example, an opportunity matrix that was independent of the distribution of offences by a single serial offender or the origins of other offenders who committed crimes in the same locations could be applied as an alternative (equation 14.14):

$$P(Jtc|O) \propto P(Jtc) * P(O|Jtc) * P(A) \quad (2.5)$$

In this case, P(A) is an independent matrix. Another filter that could be applied is residential land use. The vast majority of offenders are going to live in residential areas. Thus, a residential land use filter estimates the probability of a residential land use for every cell, P(A), could be applied to screen out cells that are not residential, such as

$$P(Jtc|O) \propto P(Jtc) * P(O|Jtc) * P(A) \quad (2.6)$$

In this way, additional information can be integrated into the methodology to improve the accuracy and precision of the estimates. Clearly, having additional variables be conditioned upon existing variables in the model would be ideal since that would fit the true Bayesian approach. But, even if independent filters were brought in, the model could be improved.

Defining up to two filters

The Bayesian Journey-to-crime routine allows the addition of up to two filters, called **F1** and **F2**. If one filter variable is defined as a data set, then F1 will be applied to the probability components. If two filter variables are defined as data sets, then both F1 and F2 will be applied simultaneously to the probability components.

Output of Routine

For each offender in the "diagnostics" file, the routine calculates three different statistics for each of the methods:

1. The estimated **probability** in the cell where the offender actually lives. It does this by, first, identifying the grid cell in which the offender lives (i.e., the grid cell where the offender's residence X and Y coordinate is found) and, second, by noting the probability associated with that grid cell. The higher the probability, the better the estimate.

2. The **percentile** of all grid cells in the entire grid that have to be searched to find the cell where the offender lives based on the probability estimate from 1, ranked from those with the highest probability to the lowest. Obviously, this percentile will vary by how large a reference grid is used (e.g., with a very large reference grid, the percentile where the offender actually lives will be small whereas with a small reference grid, the percentile will be larger). But, since the purpose is to compare methods, the actual percentage should be treated as a relative index. The result is sorted from low to high so that the smaller the percentile, the better. For example, a percentile of 1% indicates that the probability estimate for the cell where the offender lives is within the top 1% of all grid cells. Conversely, a percentile of 30% indicates that the probability estimate for the cell where the offender lives in within the top 30% of all grid cell.

3. The **distance** between the cell with the highest probability and the cell where the offender lives. The smaller the distance between the cell with the highest probability and the cell where the offender lives, the better.

Output matrices

The "diagnostics" routine outputs two separate matrices. The probability estimates (numbers 1 and 2 above) are presented in a separate matrix from the distance estimates (number 3 above). The user can save the total output as a text file or can copy and paste each of the two output matrices into a spreadsheet separately. We recommend the copying-and-pasting method into a spreadsheet as it will be difficult to line up differing column widths for the two matrices and summary tables at the bottom of each.

Summary statistics

The "diagnostics" routine will also provide summary information at the bottom of each matrix. For the probability matrix, these include:

1. The mean (probability or percentile)
2. The median (probability or percentile)
3. The standard deviation (probability or percentile)
4. The number of times the Jtc estimate produces the highest probability
5. The number of times the O|Jtc estimate produces the highest probability
6. The number of times the O estimate produces the highest probability
7. The number of times the "product" term estimate produces the highest probability
8. The number of times the Bayesian risk estimate produces the highest probability
9. If filter variable F1 has been defined:
    A. The number of times the Jtc*F1 estimate produces the highest probability
    B. The number of times the O|Jtc*F1 estimate produces the highest probability
    C. The number of times the "Product"*F1 estimate produces the highest probability
    D. The number of times the Bayesian risk*F1 estimate produces the highest probability
10. If both filter variable F1 and filter variable F2 have been defined:
    A. The number of times the Jtc*F1*F2 estimate produces the highest probability
    B. The number of times the O|Jtc*F1*F2 estimate produces the highest probability
    C. The number of times the "Product"*F1*F2 estimate produces the highest probability
    D. The number of times the Bayesian risk*F1*F2 estimate produces the highest probability

For the distance matrix, these include:

1. The mean distance
2. The median distance
3. The standard deviation distance
4. The number of times the Jtc estimate produces the closest distance
5. The number of times the O|Jtc estimate produces the closest distance
6. The number of times the O estimate produces the closest distance
7. The number of times the "product" term estimate produces the closest distance
8. The number of times the Bayesian risk estimate produces the closest distance
9. The number of times the center of minimum distance (CMD) produces the closest distance
10. *If* filter variable F1 has been defined:
    A. The number of times the Jtc*F1 estimate produces the closest distance
    B. The number of times the O|Jtc*F1 estimate produces the closest distance
    C. The number of times the "Product"*F1 estimate produces the closest distance
    D. The number of times the Bayesian risk*F1 estimate produces the closest distance
11. *If* both filter variable F1 and filter variable F2 have been defined:
    A. The number of times the Jtc*F1*F2 estimate produces the closest distance
    B. The number of times the O|Jtc*F1*F2 estimate produces the closest distance
    C. The number of times the "Product"*F1*F2 estimate produces the closest distance
    D. The number of times the Bayesian risk*F1*F2 estimate produces the closest distance

These statistics, especially the summary statistics, should indicate which of the methods produces the best accuracy, defined in terms of highest probability (for the probability matrix) and closest distance (for the distance matrix), and efficiency, defined in terms of the smallest search area to locate the serial offender.

Estimate Likely Origin Location of a Serial Offender

The following applies to the Bayesian Jtc "Estimate likely origin of a serial offender" routine. Once the "diagnostic" routine has been run and a preferred method selected, the next routine allows the application of that method to a single serial offender.

Data Input

The user inputs the three required data sets and a reference file grid. The two filter variables can also be applied, but are optional

1. The incidents committed by a single offender that we're interested in catching. This must be the primary file.
2. A journey-to-crime function that estimates the likelihood of an offender committing crimes at a certain distance (or travel time if a network is used).
3. An origin-destination matrix.
4. The reference file also needs to be defined and should include all locations where crimes have been committed (see Reference File).
5. (Optional) A data set that includes a filter variable (see above).
6. (Optional) A data set that includes a second filter variable (see above).

Methods Tested

The Bayesian Jtc "Estimate" routine interpolates the incidents committed by the serial offender to a grid, allowing the user to estimate where the offender is liable to live. There are 13 different methods for estimating the likely location of a serial offender that can be used, depending on whether filter variables are used or not. However, the user has to choose only <u>one</u> of these:

1. The Journey-to-crime distance method, P(Jtc).

2. The general crime distribution based on the origin-destination matrix, P(O). Essentially, this is the distribution of origins irrespective of the destinations.

3. The distribution of origins based only on the incidents committed by the serial offender, P(O|Jtc).
4. The product of the Journey-to-crime estimate (<u>1</u> above) and the distribution of origins based only on the incidents committed by the serial offender (<u>3</u> above), P(Jtc)*P(O|Jtc). This is the numerator of the Bayesian function discussed above, the product of the prior probability times the likelihood estimate.

5. The weighted average of the Journey-to-crime estimate (<u>1</u> above) and the distribution of origins based only on the distribution of incidents committed by the serial offender (<u>3</u> above), P(Jtc) + P(O|Jtc). This is an alternative to the product term (<u>4</u> above). The user must select weights for each of the two estimates such

that the sum of the weights equal 1.0. The default weights are 0.5 for each estimate.

6. The Bayesian risk estimate as indicated above (method 4 above divided by method 2 above), P(Bayes risk).
7. *If* one filtering variable, F1, has been used:

   A. P(Jtc)*F1
   B. P(O|Jtc)*F1
   C. "Product"*F1
   D. Bayesian risk*F1

8. *If* two filtering variables have been used:

   A. P(Jtc)*F1*F2
   B. P(O|Jtc)*F1*F2
   C. "Product"*F1*F2
   D. Bayesian risk*F1*F2

Interpolated Grid

For the estimation method that is selected, the routine overlays a grid on the study area. The grid is defined by the reference file parameters (see Reference File). The routine then interpolates the input data set (the primary file) into a probability estimate for each grid cell with the sum of the cells equaling 1.0 (within three decimal places). The manner in which the interpolation is done varies by the method chosen:

1. For the Journey to crime method, P(Jtc), the routine interpolates the selected distance function to each grid cell to produce a density estimate. The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0;
2. For the general crime distribution method, P(O), the routine sums up the incidents by each origin zone and interpolates that using the normal distribution method of the single kernel density routine (see Kernel Density Interpolation). The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

3. For the distribution of origins based only on the incident committed by the serial offender, the routine identifies the zone in which the incident occurs and reads only those origins associated with those destination zones in the origin-destination matrix. Multiple incidents committed in the same origin zone are counted multiple

times. The routine then uses the single kernel density routine to interpolate the distribution to the grid (see Kernel Density Interpolation). The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

4. For the product of the Journey-to-crime estimate and the distribution of origins based only on the incidents committed by the serial offender, the routine multiples the probability estimate obtained in 1 above by the probability estimate obtained in 3 above. The product density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

5. For the Bayesian risk estimate, the routine takes the product estimate (4 above) and divides it by the general crime distribution estimate (2 above). The resulting densities are converted to probabilities so that the sum of the grid cells equals 1.0.

6. If one or two filter variables are used, each filter variable is interpolated to the reference grid and then converted into probabilities. The filter probability grid is then multiplied by the P(Jtc), P(O|Jtc), "Product" or Bayesian risk grids to produce a filtered grid.

Note that in all estimates, the cells are converted to probabilities prior to any multiplication or division. The results are then re-scaled so that the resulting grid is a probability (i.e., all cells sum to 1.0).

Output of Routine

Once the method has been selected, the routine interpolates the data to the grid cell and outputs it as a 'shp', 'mif/mid', or Ascii file for display in a GIS program. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The tabular output shows the probability values for each cell in the matrix and also indicates which grid cell has the highest probability estimate.

Accumulator Matrix

There is also an intermediate output, called the *accumulator matrix*, which the user can save. This lists the number of origins identified in each origin zone for the specific pattern of

incidents committed by the offender, prior to the interpolation to grid cells. That is, in reading the origin-destination file, the routine first identifies which zone each incident committed by the offender falls within. Second, it reads the origin-destination matrix and identifies which origin zones are associated with incidents committed in the particular destination zones. Finally, it sums up the number of origins by zone ID associated with the incident distribution of the offender. This can be useful for examining the distribution of origins by zones prior to interpolating these to the grid.

V. Spatial Modeling II

The second spatial modeling section conducts regression modeling of a dependent variable, either binomial, unconstrained, or a count variable. It also includes a module for modeling discrete (nominal) choices. There are five sets of routines in the section: 1) Regression I for modeling multivariate predictors of a continuous or binary variable; 2) Regression II for making predictions on a new data set based on a regression model; 3) Discrete choice I for modeling discrete decisions; 4) Discrete choice II for making predictions on a new data set based on a discrete choice model; and 5) Temporal modeling for predicting the expected number of counts of an incident variable by zones and for detecting when the actual number exceeds a threshold prediction.

Regression Modeling I

The aim of a regression model is to estimate a functional relationship between a dependent variable and one or more independent variables. In the current version, 18 possible regression models are available with several options for each of these:

MLE Normal (OLS)
MCMC Normal
MCMC Normal-CAR
MCMC Normal-SAR
MLE Poisson
MLE Poisson with linear dispersion correction (NB1)
MLE Poisson-Gamma (NB2)
MCMC Poisson-Gamma (NB2)
MCMC Poisson-Gamma-CAR
MCMC Poisson-Gamma-SAR
MCMC Poisson-Lognormal
MCMC Poisson-Lognormal-CAR

MCMC Poisson-Lognormal-SAR
MLE Binomial Logit
MLE Binomial Probit
MCMC Binomial Logit
MCMC Binomial Logit-CAR
MCMC Binomial Logit-SAR

In addition, each of the 12 MCMC models can be run with an exposure (offset) variable used to define the population 'at risk' allowing a total of 30 possible regression models to be run.

There are two pages in the module. The Regression I page allows the testing of a model while the Regression II page allows a prediction to be made based on an already-estimated model. Also, since the Regression I module and Trip Generation module in the Crime Travel Demand Model duplicate regression functions, only one of these can be run at a time.

Input Data set

The data set for the regression module can be the Primary file or another file. If it is the Primary file, then it must be a spatial data file with and X and Y coordinates defined on each record. If it is another file, click on 'Other' and then identify the file. Only 'dbf' or 'txt' files are allowed.

Dependent Variable

To start loading the module, click on the 'Calibrate model' tab. A list of variables from the Primary File is displayed. There is a box for defining the dependent variable. The user must choose one dependent variable.

Independent Variables

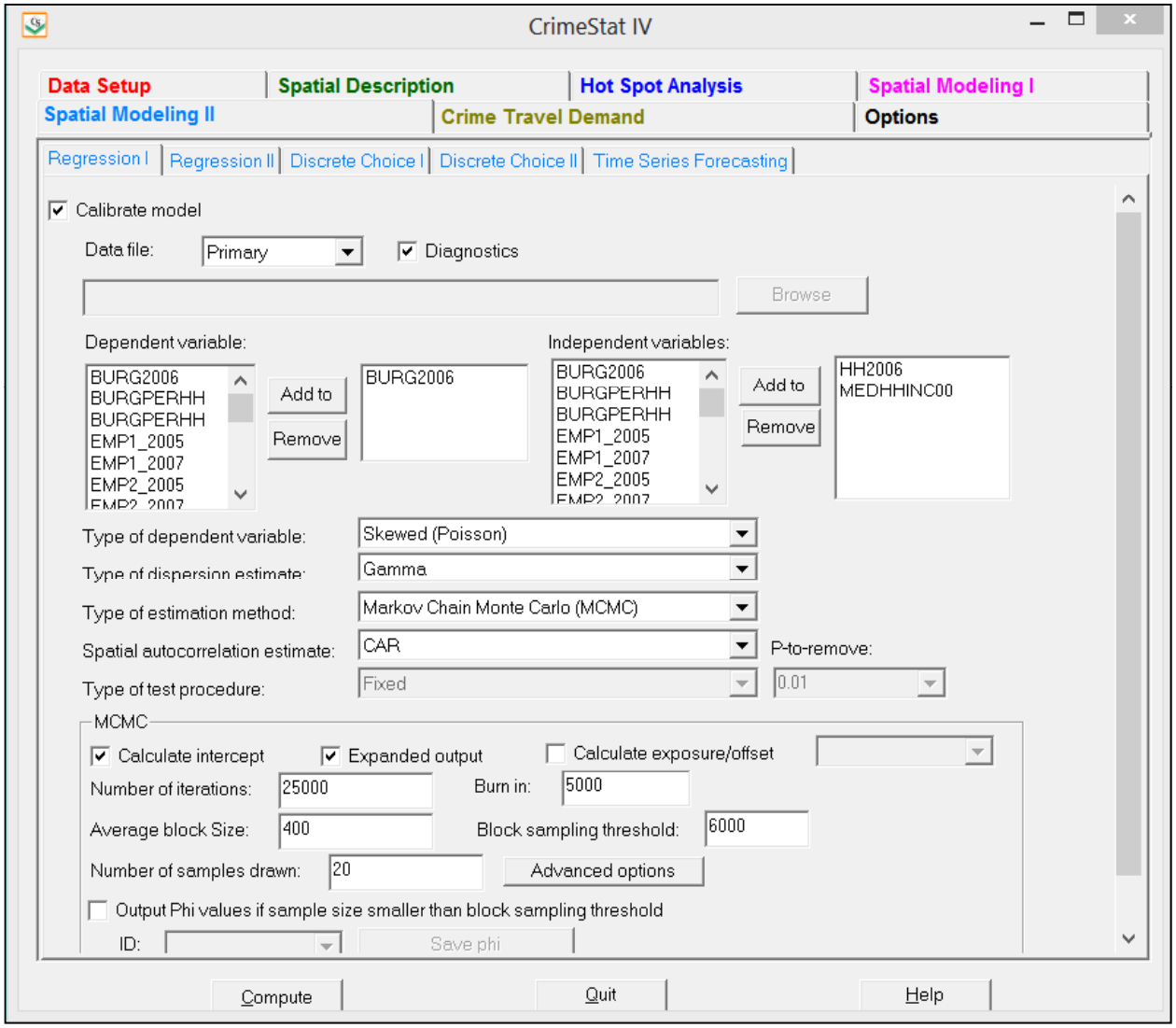
There is a box for defining the independent variables. The user must choose one or more independent variables. There is no limit to the number. The variables are output in the same order as specified in the dialogue so a user should consider how these are to be displayed.

Model decisions

There are five decisions that must be made for each regression model.

Figure 2.17:

Regression Modeling I



Type of Dependent Variable

The first model decision is the type of dependent variable: Skewed (Poisson), Normal/OLS, Binomial Probit, or Binomial Logit/Logistic. The default is a Poisson.

Type of Dispersion Estimate

The second model decision is the type of dispersion estimate to be used. The choices are Gamma, Poisson, Poisson with linear correction, and Lognormal. The default is Gamma. For the MLE and MCMC Normal (OLS) models, the dispersion is automatically normal. For the binomial logit or binomial probit, the dispersion is automatically binomial.

Type of Estimation Method

The third model decision is the type of estimation method to be used: Maximum Likelihood (MLE) or Markov Chain Monte Carlo (MCMC). The default is MLE.

Spatial Autocorrelation Regression Model

If the user accepts an MCMC algorithm, then a fourth decision is whether to run a spatial autocorrelation estimate along with it (a Conditional Autoregressive function – CAR, or a Simultaneous Autoregressive function - SAR). The MCMC Normal, MCMC Poisson-Gamma, MCMC Poisson-Lognormal, and MCMC Logit functions can be run with a spatial autocorrelation parameter. Note that the CAR model runs quite quickly whereas the SAR model runs very slowly. Unless the data set is small or a SAR model is absolutely essential, we recommend using a CAR function for the spatial regression models.

Type of Test Procedure

The fifth, and last model decision, is whether to run a fixed model or a backward elimination *stepwise* procedure (only with the MLE models). A fixed model includes all selected independent variables in the regression whereas a backward elimination model starts with all selected variables in the model but proceeds to drop variables that fail the P-to-remove test, one at a time. Any variable that has a significance level in excess of the P-to-remove value is dropped from the equation.

Specify whether a fixed model (all selected independent variables are used in the regression) or a backward elimination stepwise model is used. The default is a fixed model. If a backward elimination stepwise model is selected, choose the P-to-remove value (default is .01).

MCMC Model Choices

If the user chooses the MCMC algorithm, then eight *additional* decisions have to be made.

Number of Iterations

The first MCMC decision is the number of iterations. The default is 25,000. The number should be sufficient to produce reliable estimates of the parameters. Check the MC Error/Standard deviation ratio and the G-R statistic after the run to be sure most parameters are below 1.05 and 1.20 respectively. If not, increase the number of iterations and 'burn in' iterations.

'Burn in' iterations

The second MCMC decision is the number of initial iterations that will be dropped from the final distribution (the 'burn in' period). The default is 5,000. The number of 'burn in' iterations should be sufficient for the algorithm to reach an equilibrium state and produce reliable estimates of the parameters. Check the MC Error/Standard deviation ratio and the G-R statistic after the run to be sure most parameters are below 1.05 and 1.20 respectively. If not, increase the number of iterations and 'burn in' iterations.

Block Sampling Threshold

The third MCMC decision is whether to run all the records through the MCMC algorithm or whether to draw block samples. The algorithm will be run on all records unless the number of records exceeds number specified in the block sampling threshold. The default threshold is 6000 records. To run all the records through the MCMC algorithm, change this value to be greater than the number of records in the database. Note that calculation time will increase substantially if all records in a large database are run through the algorithm.

Average Block Size

The fourth MCMC decision is the number of records to be drawn for each block sample if the total number of records is greater than the block sampling threshold. The default is 400 records per block sample. Note that this is an average. Actual samples will vary in size. The output will display the expected sample size and the average sample size that was drawn.

Number of Samples Drawn

The fifth MCMC decision is the number of samples to be drawn if the total number of records is greater than the block sampling threshold. The default is 25 block samples. Typically, 20-30 block samples will achieve stable model results.

Calculate Intercept

The sixth MCMC decision is whether to run a model with or without an intercept (constant). The default is with an intercept estimated. To run the model without the intercept, uncheck the 'Calculate intercept' box.

Calculate Exposure/Offset

The seventh MCMC decision is whether to run a risk model. If the model is a risk or rate model, then an exposure (offset) variable needs to be defined. The exposure (offset) choice is available for the MCMC Poisson-Gamma, MCMC Poisson-Lognormal, and MCMC Binomial Logit models plus their spatial autocorrelation options. It is not available for the MCMC Normal or MCMC Normal-CAR/SAR models. Check the 'Calculate exposure/offset' box and identify the variable that will be used as the exposure variable. The coefficient for this variable will automatically be 1.0.

Advanced Options

The eighth, and last, MCMC decision is the prior values used for the different parameters being estimated. The MCMC algorithm requires an initial estimate for each parameter. There is a dialogue of advanced options for the MCMC algorithm by which they can be changed.

Initial Parameters Values

For the beta coefficients (including the intercept), the default values are 0. These are displayed as a blank screen for the Beta box. However, other prior estimates of the beta coefficients can be substituted for the assumed 0 coefficients. To do this, all independent variable coefficients plus the intercept (if used) must be listed in the order in which they appear in the model and must be separated by commas. Do not include the beta coefficients for the spatial autocorrelation term (if used) or the error (Taupsi) term.

Taupsi (error term)

The output of the MCMC always includes an error term, called *Taupsi* (τ_ψ). This is an exponent of the error term, $e^{\tau\psi}$, which together is called the *dispersion parameter*. The default value for Taupsi is 1.0. The user can substitute an alternative value.

Rho and Tauphi

The spatial autocorrelation component is made up of three separate sub-components, called Rho, Tauphi, and Alpha and are additive. Rho is roughly a global component that applies to the entire data set. Tauphi is roughly a neighborhood component that applies to a sub-set of the data. Alpha is essentially a localized effect. The default initial values for Rho and Tauphi are 0.5 and 1 respectively. The user can substitute alternative values for these parameters.

Alpha

Alpha is the exponent for the distance decay function in the spatial model. Essentially, the distance decay function defines the weight to be applied to the values of nearby records. The weight can be defined by one of three mathematical functions. First, the weight can be defined by a negative exponential function.

Second, the weight can be defined by a restricted negative exponential with the negative exponential operating up to the specified search distance, whereupon the weight becomes 0 for greater distances.

Third, the weight can be defined as a uniform value for all other observations within a specified search distance. This is a *contiguity* (or adjacency) measure. Essentially, all other observations have an equal weight within the search distance and 0 if they have a greater distance.

For the negative exponential and restricted negative exponential functions, substitute the selected value for alpha in the alpha box and for the restricted negative exponential and uniform functions, specify the search distance and distance units. The default is a negative exponential with an alpha of -1.0 in miles.

Value for 0 distance between records

The advanced options dialogue has a parameter for the minimum distance to be assumed between different records. If two records have the same X and Y coordinates (which could happen if the records are individual events, for example), then the distance between these records

will be 0. This could cause unusual calculations in estimating spatial effects. Instead, it is more reliable to assume a slight difference in distance between all records. The default is 0.005 miles but the user can modify this (including substituting 0 for the minimal distance).

Output

The output depends on whether an MLE or an MCMC model has been run.

Maximum Likelihood (MLE) Model Output

The MLE routines (Normal/OLS, Poisson, Poisson with linear correction, MLE Poisson-Gamma, Binomial Probit, and Binomial Logit/Logistic) produce a standard output that includes summary statistics and estimates for the individual coefficients.

MLE Summary Statistics

The summary statistics include:

Information about the model

1. The data file
2. The dependent variable
3. The number of records
4. The residual degrees of freedom (N – number of parameters estimated)
5. The type of regression model (Normal/OLS, Poisson, Poisson with linear correction, Poisson-Gamma, Binomial Logit, Binomial Probit)
6. The method of estimation (MLE)

Likelihood statistics

7. Log-likelihood estimate, which is a negative number. For a set number of independent variables, the more negative the log-likelihood the better.
8. Log-likelihood per case. This divides the log-likelihood by the sample size (N). This indicates the average contribution to the log-likelihood of each observation. The more negative, the better.
9. Akaike Information Criterion (AIC) adjusts the log-likelihood for the degrees of freedom. The smaller the AIC, the better.

10. AIC per case. This divides the AIC statistic by the sample size (N). This indicates the average contribution to the AIC of each observation. The smaller, the better.
11. Bayesian Information Criterion (BIC), sometimes known as the Schwartz Criterion (SC), adjusts the log-likelihood for the degrees of freedom. The smaller the BIC, the better.
12. BIC per case. This divides the BIC/SC statistic by the sample size (N). This indicates the average contribution to the BIC/SC of each observation. The smaller, the better.
13. Deviance compares the log-likelihood of the model to the log-likelihood of a model that fits the data perfectly. A smaller deviance is better.
14. The probability value of the deviance based on a Chi-square test with N-K-1 degrees of freedom where K is the number of independent variables.
15. Pearson Chi-square is a test of how closely the predicted model fits the data. A smaller Chi-square is better since it indicates the model fits the data better.
16. The probability value of the Pearson Chi-square based on a Chi-square test with N-K-1 degrees of freedom where K is the number of independent variables.

Model error estimates

17. Mean Absolute Deviation (MAD). For a set number of independent variables, a smaller MAD is better.
18. Quartiles for the Mean Absolute Deviation. For any one quartile, smaller is better.
19. Mean Squared Predictive Error (MSPE). For a set number of independent variables, a smaller MSPE is better.
20. Quartiles for the Mean Squared Predictive Error. For any one quartile, smaller is better.
21. Squared multiple R (for Normal/OLS models only). This is the percentage of the dependent variable accounted for by the independent variables.
22. Adjusted squared multiple R (for Normal/OLS models only). This is the squared multiple R adjusted for degrees of freedom.

Dispersion tests

23. Adjusted deviance. This is a measure of the difference between the observed and predicted values (the residual error) adjusted for degrees of freedom. The smaller the adjusted deviance, the better. A value greater than 1 indicates over-dispersion.

24. Probability of adjusted deviance. This is the probability associated with the adjusted deviance test with 1 degree of freedom.
25. Adjusted Pearson Chi-square. This is the Pearson Chi-square adjusted for degrees of freedom. The smaller the Pearson Chi-square, the better. A value greater than 1 indicates over-dispersion.
26. Probability of Adjusted Pearson Chi-square. This is the probability associated with the Pearson Chi-square test with 1 degree of freedom.
27. Dispersion multiplier. This is the ratio of the expected variance to the expected mean. For a set number of independent variables, the smaller the dispersion multiplier, the better. For example, in a pure Poisson distribution, the dispersion should be 1.0. In practice, a ratio greater than 10 indicates that there is too much variation that is unaccounted for in the model. Either add more variables or change the functional form of the model.
28. Z-test for dispersion multiplier (Poisson models only). This is a test for whether the dispersion parameter is significantly greater than that assumed by the Poisson model. It is a test of over-dispersion.
29. P-value for Z-test of dispersion parameter (Poisson models only). This is the one-tail probability level associated with the Z-test.
30. Inverse dispersion multiplier. For a set number of independent variables, a larger inverse dispersion multiplier is better. A ratio close to 1.0 is considered good.

MLE Individual Coefficient Statistics

For the individual coefficients, the following are output:

31. The coefficient. This is the estimated value of the coefficient from the maximum likelihood estimate.
32. Standard Error. This is the estimated standard error from the maximum likelihood estimate.
33. Pseudo-tolerance. This is the tolerance value based on a linear prediction of the variable by the other independent variables. See equation Up. 2.18.
34. Z-value. This is asymptotic Z-test that is defined based on the coefficient and standard error. It is defined as Coefficient/Standard Error.
35. p-value. This is the two-tail probability level associated with the Z-test.

Markov Chain Monte Carlo (MCMC) Model Output

The MCMC routines (Poisson-Gamma, Poisson-Gamma-CAR/SAR, Poisson-Lognormal, Poisson-Lognormal-CAR/SAR, Binomial Logit, Binomial Logit-CAR/SAR) produce a standard

output and an optional expanded output**.** The standard output includes summary statistics and estimates for the individual coefficients.

MCMC Summary Statistics

The summary statistics include:

Information about the model

1. The dependent variable
2. The number of records
3. The sample number. This is only output when the block sampling method is used.
4. The number of cases for the sample. This is only output when the block sampling method is used.
5. Date and time for sample. This is only output when the block sampling method is used
6. The residual degrees of freedom (N – number of parameters estimated)
7. The type of regression model (Normal, Normal-CAR/SAR, Poisson-Gamma, Poisson-Gamma-CAR/SAR, Poisson-Lognormal, Poisson-Lognormal-CAR/SAR, Binomial Logit, Binomial Logit-CAR/SAR)
8. The method of estimation
9. The number of iterations
10. The 'burn in' period
11. The block size is the expected number of records selected for each block sample. The actual number may vary.
12. The number of samples drawn. This is output when the block sampling method used.
13. The average block size. This is output when the block sampling method used.
14. The type of distance decay function used. This is output for models that use CAR or SAR spatial autocorrelation functions.
15. Condition number for the distance matrix. If the condition number is large, then the model may not have properly converged. This is output for the Poisson-Gamma-CAR model only.
16. Condition number for the inverse distance matrix. If the condition number is large, then the model may not have properly converged. This is output for the Poisson-Gamma-CAR/SAR or Poisson-Lognormal-CAR/SAR models only.

Likelihood statistics

17. Log-likelihood estimate, which is a negative number. For a set number of independent variables, the smaller the log-likelihood (i.e., the most negative) the better.
18. Log-likelihood per case. This divides the log-likelihood by the sample size (N). This indicates the average contribution to the log-likelihood of each observation. The more negative, the better.
19. Deviance Information Criterion (DIC) for Poisson models only. This adjusts the log-likelihood for the effective degrees of freedom. The smaller the DIC, the better.
20. Akaike Information Criterion (AIC) adjusts the log-likelihood for the degrees of freedom. The smaller the AIC, the better.
21. AIC per case. This divides the AIC statistic by the sample size (N). This indicates the average contribution to the AIC of each observation. The smaller, the better.
22. Bayesian Information Criterion (BIC), sometimes known as the Schwartz Criterion (SC), adjusts the log-likelihood for the degrees of freedom. The smaller the BIC, the better.
23. BIC per case. This divides the BIC/SC statistic by the sample size (N). This indicates the average contribution to the BIC/SC of each observation. The smaller, the better.
24. Deviance compares the log-likelihood of the model to the log-likelihood of a model that fits the data perfectly. A smaller deviance is better.
25. The probability value of the deviance based on a Chi-square test with N-K-1 degrees of freedom where *K* is the number of independent variables.
26. Pearson Chi-square is a test of how closely the predicted model fits the data. A smaller Chi-square is better since it indicates the model fits the data well
27. The probability value of the Pearson Chi-square based on a Chi-square test with N-K-1 degrees of freedom where *K* is the number of independent variables.

Model error estimates

28. Mean Absolute Deviation (MAD). For a set number of independent variables, a smaller MAD is better.
29. Quartiles for the Mean Absolute Deviation. For any one quartile, smaller is better.
30. Mean Squared Predictive Error (MSPE). For a set number of independent variables, a smaller MSPE is better.

31. Quartiles for the Mean Squared Predictive Error. For any one quartile, smaller is better.

Dispersion tests

32. Adjusted deviance. This is a measure of the difference between the observed and predicted values (the residual error) adjusted for degrees of freedom. The smaller the adjusted deviance, the better. A value greater than 1 indicates over-dispersion.
33. The probability value of the adjusted deviance based on a Chi-square test with 1 degree of freedom.
34. Adjusted Pearson Chi-square. This is the Pearson Chi-square adjusted for degrees of freedom. The smaller the Pearson Chi-square, the better. A value greater than 1 indicates over-dispersion.
35. The probability value of the adjusted Pearson Chi-square based on a Chi-square test with 1 degree of freedom.
36. Dispersion multiplier. This is the ratio of the expected variance to the expected mean. For a set number of independent variables, the smaller the dispersion multiplier, the better. In a pure Poisson distribution, the dispersion should be 1.0. In practice, a ratio greater than 10 indicates that there is too much variation that is unaccounted for in the model. Either add more variables or change the functional form of the model.
37. Inverse dispersion multiplier. For a set number of independent variables, a larger inverse dispersion multiplier is better. A ratio close to 1.0 is considered good.

MCMC Individual Coefficients Statistics

For the individual coefficients, the following are output:

38. The mean coefficient. This is the mean parameter value for the *N-k* iterations where *k* is the 'burn in' samples that are discarded. With the MCMC block sampling method, this is the mean of the mean coefficients for all block samples.
39. The standard deviation of the coefficient. This is an estimate of the standard error of the parameter for the *N-k* iterations where *k* is the 'burn in' samples that are discarded. With the MCMC block sampling method, this is the mean of the standard deviations for all block samples.
40. t-value. This is the t-value based on the mean coefficient and the standard deviation. It is defined by Mean/Std.
41. p-value. This is the two-tail probability level associated with the t-test.

42. Adjusted standard deviation (Adj. Std). The block sampling method will produce substantial variation in the mean standard deviation, which is used to estimate the standard error. Consequently, the standard error will be too large. An approximation is made by multiplying the estimated standard deviation by

$$\sqrt{\frac{\bar{n}}{N}}$$

where $\bar{n}$ is the average sample size of the block samples and *N* is the number of records. If no block samples are taken, then this statistic is not calculated.

43. Adjusted t-value. This is the t-value based on the mean coefficient and the adjusted standard deviation. It is defined by Mean/Adj_Std. If no block samples are taken, then this statistic is not calculated.

44. Adjusted p-value. This is the two-tail probability level associated with the adjusted t-value. If no block samples are taken, then this statistic is not calculated.

45. MC error is a Monte Carlo simulation error. It is a comparison of the means of *m* individual chains relative to the mean of the entire chain. By itself, it has little meaning.

46. MC error/Std is the MC error divided by the standard deviation. If this ratio is less than .05, then it is a good indicator that the posterior distribution has converged.

47. G-R stat is the Gelman-Rubin statistic which compares the variance of *m* individual chains relative to the variance of the entire chain. If the G-R statistic is under 1.2, then the posterior distribution is commonly considered to have converged.

48. Spatial autocorrelation term (Phi) for CAR/SAR models only. This is the estimate of the fixed effect spatial autocorrelation effect. It is made up of three components: a global component (Rho); a local component (Tauphi); and a local neighborhood component (Alpha, which is defined by the user).

49. The log of the error in the model (Taupsi). This is an estimate of the unexplained variance remaining. Taupsi is the exponent of the dispersion multiplier, $e^{\tau\psi}$. For any fixed number of independent variables, the smaller the Taupsi, the better.

Expanded Output (MCMC only)

If the expanded output box is selected, additional information on the percentiles from the MCMC sample are displayed. If the block sampling method is used, the percentiles are the means of all block samples. The percentiles are:

50. 2.5^{th} percentile

51. 5th percentile
52. 10th percentile
53. 25th percentile
54. 50th percentile (median)
55. 75TH percentile
56. 90th percentile
57. 95th percentile
58. 97.5th percentile

The percentiles can be used to construct confidence intervals around the mean estimates or to provide a non-parametric estimate of significance as an alternative to the estimated t-value in the standard output. For example, the 2.5th and 97.5th percentiles provide approximate 95 percent confidence intervals around the mean coefficient while the 0.5th and 99.5th percentiles provide approximate 99 percent confidence intervals.

The percentiles will be output for all estimated parameters including the intercept, each individual predictor variable, the spatial effects variable (Phi), the estimated components of the spatial effects (Rho and Tauphi), and the overall error term (Taupsi).

Output Phi Values (CAR/SAR models only)

For CAR or SAR models only, the individual Phi values can be output. This will occur if the sample size is smaller than the block sampling threshold. Check the 'Output Phi value if sample size smaller than block sampling threshold' box. An ID variable must be identified and a DBF output file defined.

Multicollinearity Among the Independent Variables in the Regression Model

A major consideration in any regression model is that the independent variables are statistically independent. Non-independence is called *Multicollinearity*. Non-independence means that there is overlap in prediction among two or more independent variables. This can lead to uncertainty in interpreting coefficients as well as an unstable model that may not hold in the future. Generally, it is a good idea to reduce Multicollinearity as much as possible.

A tolerance test is given for each coefficient. This is defined as 1 – the R-square of the independent variable predicted by the remaining independent variables in the equation using an Ordinary Least Squares model. It is an indicator of how much the other independent variables in a model account for the variance of any particular independent variable. Since the method uses the Ordinary Least Squares methods, it is an approximate (pseudo) test for the functions estimated

by maximum likelihood. A message is displayed that indicates probable or possible Multicollinearity. If there is substantial Multicollinearity (indicated by low tolerance values), it is a good idea is to drop one of the multicolinear independent variables and re-run the model. However, each of the coefficients should be inspected carefully before accepting a final model.

Graph of Residual Errors

While the output page is open, clicking on the graph button will display a graph of the residual errors (on the Y axis) against the predicted values (on the X axis). Only residual errors that vary between -200 and +200 are shown to allow most of the errors to be displayed.

Save Output

The predicted values and the residual errors can be output to a DBF file with a RegOut<*root name*> with the root name being provided by the user. The output includes all the variables in the input data set plus two new ones: 1) the predicted values of the dependent variable for each observation (with the field name PREDICTED); and 2) the residual error values, representing the difference between the actual /observed values for each observation and the predicted values (with the field name RESIDUAL). The file can be imported into a spreadsheet or graphics program and the errors plotted against the predicted dependent variable.

Save Estimated Coefficients

The individual coefficients can be output to a DBF file with a RegCoeff <*root name*> with the root name being provided by the user. This file can be used in the 'Make Prediction' routine under Regression II.

Diagnostic Tests

The regression module has a set of diagnostic tests for evaluating the characteristics of the data and the most appropriate model to use. There is a diagnostics box on the Regression I page.

Diagnostics are provided on:

1. The minimum and maximum values for the dependent and independent variables
2. Skewness in the dependent variable
3. Spatial autocorrelation in the dependent variable
4. Estimated values for the distance decay parameter – alpha, for use in the CAR or SAR models

5. Multicolinarity among the independent variables

Minimum and Maximum Values for the Variables

First, the minimum and maximum values of both the dependent and independent variables are listed. A user should look for ineligible values (e.g., -1) as well as variables that have a very high range. The MLE routines are sensitive to variables with very large ranges.

Skewness Tests

Skewness in the dependent variable can distort a linear model by allowing high values to be underestimated while allowing low values to be overestimated and a Poisson-type model is preferred over the linear for highly skewed variables.

The diagnostics utility tests for skewness using two different measures: 1) the "***g"*** statistic, and 2) the ratio of the simple variance to the simple mean. Either significant "g" scores or variance-to-mean ratios greater than about 2:1 should make the user cautious about using a linear model. If either measure indicates skewness, *CrimeStat* prints out a message indicating the dependent variable appears to be skewed and that a Poisson or Poisson-Gamma model should be used.

Testing for Spatial Autocorrelation in the Dependent Variable

The third type of test in the diagnostics utilities is the Moran's "I" coefficient for spatial autocorrelation. If the "I" is significant, *CrimeStat* outputs a message indicating that there is definite spatial autocorrelation in the dependent variable and that it needs to be accounted for, either by a proxy variable or by estimating a CAR or SAR model.

Estimating the Value of Alpha for CAR or SAR Models

The fourth type of diagnostic test is an estimate of a plausible value for the distance decay function, α, in CAR or SAR models. Three values of alpha are given in different distance units, one associated with a weight of 0.9 (a very steep distance decay), one associated with a weight of 0.75 (a moderate distance decay), and one associated with a weight of 0.5 (a shallow distance decay). Users should run the Moran Correlogram and examine the graph of the drop off in spatial autocorrelation to assess what type of decay function most likely exists. The user should choose an alpha value that best represents the distance decay and should define the distance units for it.

Multicollinearity Test

The fifth type of diagnostic test is for Multicollinearity among the independent predictors. The pseudo-tolerance test is presented for each independent variable. This is defined as $1-R^2$ for the other independent variables in the equation. Each independent variable should have a high tolerance (0.90 or higher). *CrimeStat* prints out an error message if tolerance is not high.

Regression Modeling II

The Regression II module allows the user to apply a model to another data set and make a prediction. The 'Make prediction' routine allows the application of coefficients to a data set. There are two types of models that are fitted – linear and Poisson. For both types of model, the coefficients file must include information on the intercept and each of the coefficients. The user reads in the saved coefficient file and matches the variables to those in the new data set based on the order of the coefficients file.

If the model had estimated a general spatial effect from a CAR or SAR model, then the general Phi will have been saved with the coefficient files. If the model had estimated specific spatial effects from a CAR or SAR model, then the specific Phi values will have been saved in a separate Phi coefficients file. In the latter case, the user must read in the Phi coefficients file along with the general coefficient file.

Data File

The data set for the regression module can be the Primary file or another file. If it is the Primary file, then it must have X and Y coordinates defined on each record. If it is another file, click on 'Other' and then identify the file. Only 'dbf' or 'txt' files are allowed.

Saved Coefficients File

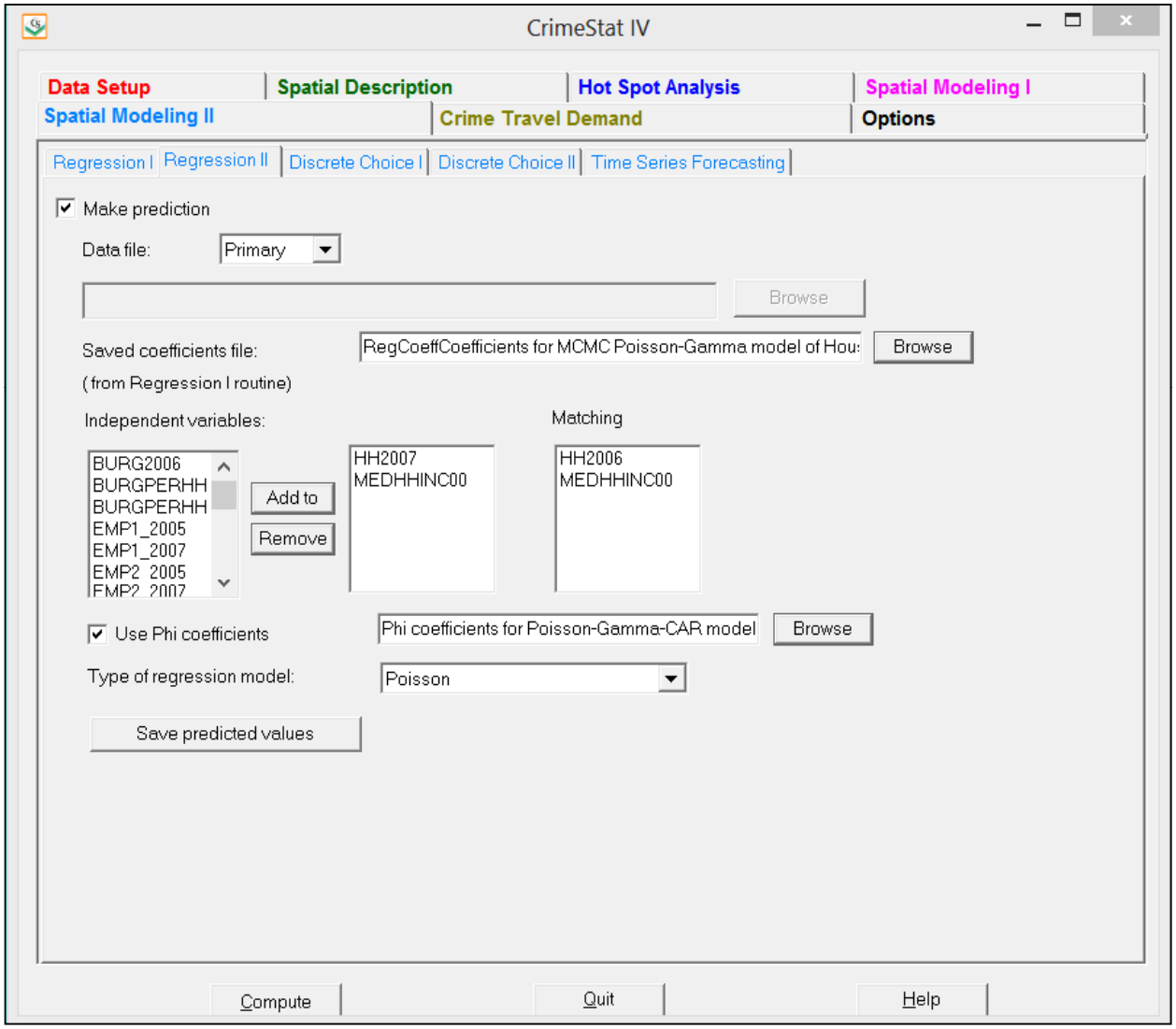
In order to make a prediction, a model must have already been calibrated and the coefficients saved in a coefficients file. Point to the directory where the coefficients file has been saved and identify it.

Matching Independent Variables

The independent variables that were used in the calibrated coefficients file will be listing in the matching column. Select corresponding variables from the input data file. The items should be listed in the same order as in the matching column.

Figure 2.18:

Regression Modeling II



Use Phi coefficients

If the saved coefficients file was from a model that was a spatial regression, the saved Phi coefficients can be also applied to the new data set. The number of Phi coefficients must match the number of records in the input data file, however. For example, this would be appropriate when a model is calibrated on zones which do not change over time. Therefore, the Phi coefficients estimated for the zones in one time period could be applied to the same zones to make a prediction for a later time period.

Point to the directory where the Phi coefficients have been saved and identify the file.

Output

The screen output provides predictions of the value of the dependent variable in the same order as in the input data set.

Save Predicted Values

The predicted values and the residual errors can be output to a DBF file with a RegMakePred<*root name*> with the root name being provided by the user. A column called PREDICTED will be added that contains the predicted value of the dependent variable.

Discrete Choice Modeling I

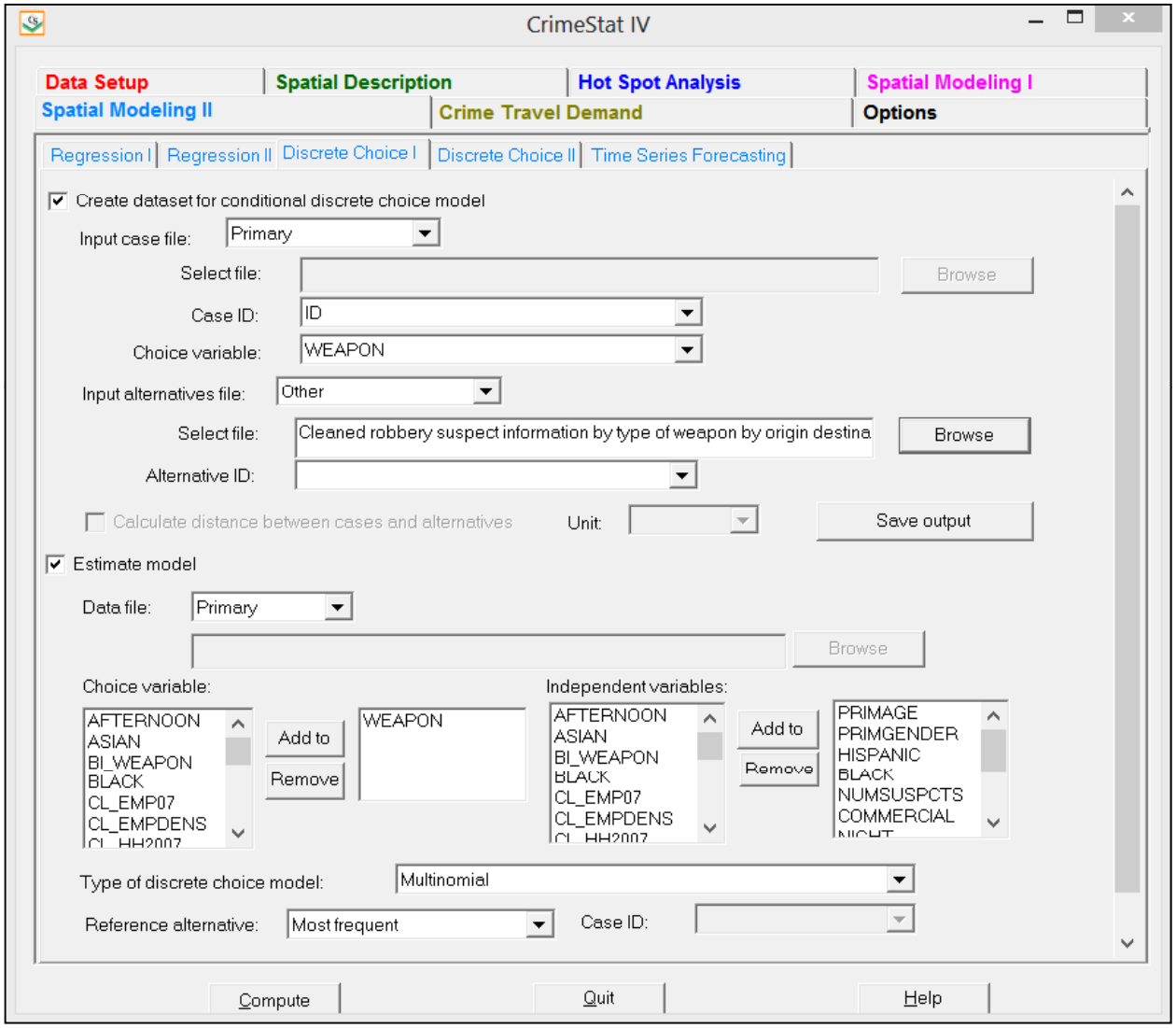
The aim of the discrete choice I module is to estimate a functional relationship between a discrete (nominal) dependent variable and one or more independent variables. It is a statistical method that is derived from utility theory, i.e. random utility maximization (RUM) theory. A 'decision maker' (e.g., an offender committing a crime) is faced with a set of alternatives, labeled 1 through *J*, from which s/he has to select exactly one.

The probability that an alternative will be chosen is a function of its observed and unobserved utility to the decision maker. The observed utility is a function of known variables and can be expressed as a linear combination of the independent variables. The unobserved utility is the random error component of the model. The estimated probability is the exponentiated observed utility of a specific alternative, *J*, divided by the sum of the exponentiated observed utilities of all available alternatives.

There are two general forms of the discrete choice model, multinomial logit and conditional logit. The *multinomial logit* model estimates the probability that a specific

Figure 2.19:

Discrete Choice Modeling I



alternative, 1 to *J*, as a function of characteristics of the decision makers, either personal characteristics (e.g., age, gender, ethnicity) or environmental characteristics (e.g., the median household income of the block in which the decision maker lives). The probability that any one alternative is chosen is estimated as a function of these characteristics. Per variable (characteristic), there is one parameter estimated for every alternative, one of which is the reference alternative in which the coefficients are automatically set to 0.

The multinomial logit model is most appropriate when the outcome of the choice is expected to depend mostly on characteristics of the decision maker (and not on observed characteristics of the alternatives) and when there are only a limited number of alternatives available (e.g., 5 weapon choices). The *conditional logit* model is a more general model and estimates the probability of a set of alternatives, 1 to *J*, as a function of characteristics of the alternatives themselves, possibly in interaction with characteristics of the decision maker. The conditional logit model is most appropriate when the outcome of the choice is expected to depend mostly on the characteristics of the alternatives, and can handle a large number of alternatives. However, the analysis file becomes very large. There is a single parameter estimated for every characteristic of the alternative.

Although the multinomial and the conditional logit are based on a single underlying statistical model, their estimation requires different data structures. In the multinomial logit model, the data contain a single record for every decision maker, and a single dependent (nominal) variable that indicates which alternative (*1..J*) was chosen. Thus, if there are *N* decision makers, there are *N* records and at least one varible indicates which alternative was chosen. The file structure is thus similar to that used in the regression module.

In the conditional logit model, for each decision maker there is a record for every choice that this decision maker is faced. Thus, if there are *N* decision makers and *J* alternatives available to every decision maker, then the data set has *N*J* records, one for every alternative faced by the decision maker. In this case, the alternative that was selected has to be indicated by a dichotmous (dummy) variable (1 for chosen and 0 for not chosen).

Create Data set for Conditional Logit Model

This routine is optional. It simplifies the task of creating a database for use in the conditional logit model. It matches a *case* database with a alternatives data base, producing the cross join of both databases. The ***case*** database is the database for the multinomial logit model. It will thus have the individual records of the decision makers – offenders, individuals, organizations. It will include at least one variable indicating the alternative that the decision maker selected (e.g., type of crime committed, the type of weapon used, the location where the

crime was committed) as well as characteristics of the individuals or characteristics associated with the individuals (e.g., age, gender, ethnicity, median household income of the zone where the decision maker lives time of event, day of week of event).

The ***alternatives*** database, on the other hand, lists the individual alternatives that were available (e.g., all the locations where a crime could be committed, all the different types of weapons that were used by different offenders) as well as attributes associated with the alternatives themselves (e.g., median household income or number of employees working at the locations, or characteristics associated with each type of weapon).

The joined has one record per alternative for each case. Thus, if there are *N* individuals faced with *J* choices, then the matching routine will create *N*J* records. It should be noted that the matching assigns every characteristic associated with a choice to every case associated with a decision maker. A field, called CHOSEN, is automatically added to every record. This field has the value 1 for alternatives that were chosen and 0 for alternatives that were not chosen. The Chosen field should thus sum to N (i.e., only one record per decision maker should have a selected alternative). Also, as an option, and only if both the individuals and the alternatives have geographic coordinates, a second field called DISTANCE will be added that calculates the distance from each case record to each alternative record. The user must specify which distance units are to be used (miles, kilometers, meters, feet, or nautical miles).

For example, if both the case database and the alternatives database contain X and Y coordinates, then it is possible to calculate the distance between every decision maker and every choice. In most situations, locations at shorter distances are more likely to be chosen.

The routine cannot calculate other interactions associated with a specific alternative and particular decision maker, and such interactions must be added to the data outside CrimeStat. Interactions between variables in the data can be calculated. For example, to test whether increasing distance makes alternatives less attractive for juvenile offenders but not for adult offenders, an interaction DISTANCE x AGE can be calculated. Other interactions require additional information, for example if location choice is what is modeled, one may want to add a variable indicating, for each alternative location, how many prior offences the offender has committed before in that alternative location. In these cases the external file is constructed by the user, and the step “Create data set for conditional discrete choice model” is skipped.

Input Case File

The case data set for the Discrete Choice I module can be the Primary file or another file. If it is the Primary file, then it must have X and Y coordinates defined on each record. If it is

another file, then there are no coordinates defined. Click on 'Other' and then identify the file. Only 'dbf' or 'txt' files are allowed. To avoid confusion, the user must verify that no variable/field in the input case file has the same name as any variable in the Input Alternatives File (see below).

Case ID

Select the Case ID. The Input Case File must have a Case ID, a variable that uniquely identifies cases in the Input Case File.

Choice variable

Select the Choice Variable. The Input Case File must contain a variable (field) that identifies alternative chosen by the decision maker. For example, if the choice is about the type of weapon used, then the Choice Variable indicates whether it was a gun, a knife, strong arm, and so forth. Or, if the choice is the census tract in which a crime was perpetrated, then the Choice Variable identifies the census tract where the incident occurred.

Input Alternatives File

The alternatives data set for the Discrete Choice I module can be the Primary file or another file. If it is the Primary file, then it is a spatial file and must have X and Y coordinates on each record. If it is another file, then there are no coordinates defined. Click on 'Other' and then identify the file. Only 'dbf' or 'txt' files are allowed. To avoid confusion, the user must verify that no variable in the input alternative file has the same name as any variable in the Input Case File.

Alternatives ID

Select the Alternatives ID. The Alternatives File must have an Alternative ID, a variable that uniquely identifies records in file. The coding of the Alternative ID variable must exactly match the coding of the Choice Variable in the Input Case File. Be careful about ID names. If the ID names are the same, the name will appear twice in the file with the first use representing the case file and the second use representing the alternatives file. The names reflect the link between each case ID and each alternatives ID. It will be better to use different names to confusion.

Calculate Distance between Cases to Alternatives

There is an optional box that allows the routine to calculate the distance from each case record to each alternative record. If checked, the routine will calculate the distance. This only applies if both the case file and the alternatives file are either the Primary file or Secondary. The user must specify the distance units to be used in the calculation (in miles, kilometers, feet, meters, or nautical miles). The box is checked by default. The saved filed will have a new field called DIST. For example, if the X/Y coordinates for an offender's home address are coded in the Input Case File while the coordinates for census tract are recorded in the Input Alternatives File, then the distances from the offender's home to each alternative census tract will be calculated.

Save Output

The matched Input Case and Input Alternatives file is saved as a new file in 'dbf' format, that can subsequently be used to estimate a conditional (but not multinomial) logit model, as described below under 'Estimating a conditional logit model'. The user should define the name of the file and point to the directory where it is saved. The output includes all fields from the case file and all fields from the choice file, and optionally a field DIST containing calculated distances. There will be *J* records for each of the *N* cases. There will be an automatically added field called CHOSEN that takes the value '1' for the choice that was selected and '0' for choices that were not selected.

Note that because the joined data base can be very large, before you start creating a data set for conditional discrete choice model, be careful to include in the alternatives and choice files only variables that you are likely to use in your analysis, and to format them to be as small as possible.

Estimate Model

The Estimate Model routine will estimate a discrete choice model, either the multinomial logit or the conditional logit.

Estimating a Multinomial Logit Model

The *multinomial logit* model is used when there is one record per decision maker with a choice having been made by the decision maker. The model estimates the effect of each independent variable on the probability of each distinct alternative. The data are structured so that there is one record per decision maker with the choice variable indicating which alternative

was chosen. The data set is similar to that of the regression model in that there is one record per decision maker.

The model then estimates the effects of the independent variables on the probability of each alternative. By definition, one of the alternatives (by default the most frequently chosen alternative, otherwise to be chosen by the user) is the reference alternative to which the other alternatives are compared.

The multinomial logit model is always estimated with a constant. This type of model is appropriate when values of the predictor variables only vary across cases (decision-makers), not across alternatives.

Estimating a Conditional Logit Model

The *conditional logit* model, on the other hand, is used when the values of the predictor variables vary across alternatives. In that case, there is one record per alternative per decision maker. That is, the decision maker is faced with *J* alternatives but chooses only one. The database must indicate which of the *J* alternatives was selected and the model estimates the effect of each independent variable on choosing an alternative. There is a record for every alternative faced by the decision maker. The parameter estimates indicate the effects of the independent variables on the likelihood that the alternative is selected.

Typically, if there are *N* decision makers and *J* alternatives, then there will be normally *N x J* records. It is possible for a particular decision maker to have fewer than *J* alternatives. The model will still work.

Data File

The data set for the model can be either the Primary file or another file (the Secondary file is not available). If the Primary file is used, the coordinate system and distance units are the same as were defined on the Primary file page.

Select file for other discrete choice file

If the discrete choice file is another file than the Primary file, the user must browse and identify the file.

Choice Variable

A list of variables from the discrete choice file is displayed. There is a box for defining

the choice variable. The user must select one choice variable. . For the conditional logit model, on the other hand, the variable contains a set of 1's (for selected alternatives) or 0's (for alternatives that were not selected). If the data set was constructed with the *CrimeStat* 'Create data set for conditional discrete choice model' routine, then the field CHOSEN should be used.

Note that the field that is added for the choice variable (whether CHOSEN or another variable) is inspected for unique values. If the data set is large, it may take awhile to filter through those values. Eventually, though, the variable will be added to the choice variable dialogue.

Independent Variables

There is a box for defining the independent variables. The user must choose one or more independent variables. There is no limit to the number. The variables are output in the same order as specified in the dialogue so a user should consider how these are to be displayed. The order in which the variables are entered does not affect the estimated parameters.

Type of Discrete Choice Model

The type of discrete choice model to be estimated must be specified. The choices are *Multinomial* (logit) or *Conditional* (logit). The default model is the Conditional logit. NOTE: the file used for a Multinomial Logit model is different than the file used for a Conditional Logit model. With the file used in the Multinomial Logit model, there is one record per case with the choice specified on the record. With the file used in the Conditional Logit model, there is one record per alternative with *J* records per case (where *J* is the number of alternatives). Be sure to use the correct file type. The routine ***assumes*** that the data are ***consistent*** with the type of model chosen. For a multinomial logit model, the routine will treat each record as a separate decision maker and will estimate a model for each choice less the reference choice. For a conditional logit model, the routine will treat each record as one of *J* choices (where *J* is defined by the user – see below) and will estimate a single model for the decision.

The user needs to be very careful that the correct data set is used with the appropriate model because the routine can estimate its equations with either of these data sets. That is, if the data set is appropriate for the multinomial logit model but the user specifies a conditional logit model, the routine will estimate a single equation treating multiples of *J* records as a single decision maker. Similarly, if the data set is appropriate for a conditional logit model but the user specifies a multinomial logit model, the routine will treat each record as if it were a separate decision maker and will estimate one equation for each choice that it finds in the choice variable. The results in both these cases will be meaningless since the there is a mismatch between the data

set and the type of model selected. In short, the user should be aware of this.

Reference alternative *(multinomial logit model only)*

For the multinomial logit model, the user should specify which choice is to be used as the reference. The constant and the coefficients for the reference choice will automatically be 0. The user should specify a particular choice from the list of available alternatives or select the most frequently used alternative as the reference choice. Keep in mind that the coefficients will change depending on which alternative is selected as the reference choice since a comparison is always relative. This will affect the interpretation of the coefficients though not the estimated probabilities.

For the conditional logit model, however, there is no reference choice. Therefore, this field will be blanked out when the type of discrete choice model is conditional.

Case ID ***(conditional logit model only)***

When a conditional logit model is estimated, each case contributes multiple records to the data file (as many as there are alternatives). In order for *CrimeStat* to know which records belong to the same case (decision maker), the user must specify a Case ID variable, i.e. a variable that uniquely identifies cases (decision makers). If the data set was created with the *CrimeStat* 'Create Data set for Conditional Logit Model' routine, the variable is the Case ID variable specified in that routine.

Output for the Discrete Choice Model

The output includes both summary statistics and individual variable coefficients estimates. The output will vary between the multinomial logit and conditional logit models.

Discrete Choice Model Summary Statistics

The summary statistics include:

Information about the model

1. Date and time
2. The data file
3. The dependent (choice) variable
4. The number of records

5. The degrees of freedom
6. The type of choice model (multinomial discrete or conditional discrete)
7. Number of alternatives
8. The method of estimation (MLE – maximum likelihood estimation, only in this version).

Discrete choice model likelihood statistics

9. Log-likelihood estimate, which is a negative number. For a set number of independent variables, the smaller the log-likelihood (i.e., the most negative) the better.
10. Log-likelihood per case. Smaller (more negative) values are better. This is useful when comparing a similar model but with different numbers of records.
11. Akaike Information Criterion (AIC) adjusts the log-likelihood for the degrees of freedom. The smaller the AIC, the better.
12. AIC per case. Smaller values are better.
13. Bayesian Information Criterion (BIC), sometimes known as the Schwartz Criterion (SC), adjusts the log-likelihood for the degrees of freedom. The smaller the BIC, the better.
14. BIC per case. Smaller values are better.
15. Mean Absolute Deviation (MAD). For a set number of independent variables, a smaller MAD is better.
16. Mean Squared Predictive Error (MSPE). For a set number of independent variables, a smaller MSPE is better.

Discrete Choice Individual Coefficients Statistics

There is a different coefficient output for the multinomial logit model than for the conditional logit model. The multinomial logit model will output constants ***and*** individual coefficients for each of *J*-1 alternatives (where *J* is the total number of alternatives). The constant and coefficients for the reference alternative are automatically defined as zero (0). For example, if there are four alternatives, then three sets of equations will be output, one for each of the *J-1* (4-1=3) alternatives.

The coefficients are always relative to the reference alternative. Therefore, a positive coefficient indicates that the independent variable contributes more for that alternative than for the reference alternative while a negative coefficient indicates that the independent variable contributes less for that choice than for the reference choice. The significance test of the

coefficient indicates whether the difference is statistically significant or not compared to the reference alternative. Note that the multinomial logit model *always* has a constant.

On the other hand, the conditional logit model will output a single set of individual coefficients with ***no*** constant. There is no reference choice and the coefficients are relative to not choosing a particular alternative (i.e., having a value of 0 for CHOSEN).

For the individual coefficients, the following are output for each independent variable:

1. The coefficient.
2. The standard error of the coefficient.
3. t-value.
4. p-value. This is the two-tail probability level associated with the t-test.
5. Odds ratio. This is the exponentiation of the coefficient (i.e., e^{β}). It indicates the relative odds of that variable affecting the choice relative to the reference choice (multinomial logit model) or relative to 0 (conditional logit model).

Average predicted probability

For the conditional logit model only, an additional table is output that indicates the average predicted probability of the model for those cases that were selected (i.e., in which CHOSEN=1), for those cases that were not selected (i.e., in which CHOSEN=0), and for all cases. The number of records associated with each category and the standard deviation are given.

Multicollinearity Among Independent Variables in the Discrete Choice Model

A major consideration in any regression model (including discrete choice) is that the independent variables are statistically independent. Non-independence is called *Multicollinearity* and means that there is overlap in prediction among two or more independent variables. This can lead to uncertainty in interpreting coefficients as well as to an unstable model that may not hold in the future. Generally, it is a good idea to reduce Multicollinearity as much as possible.

A tolerance test is given for each coefficient. This is defined as 1 – the R-square of the independent variable predicted by the remaining independent variables in the equation using an Ordinary Least Squares model. It is an indicator of how much the remaining variables in a model account for the variance of any particular independent variable. Since the method uses the Ordinary Least Squares (OLS) methods, it is an approximate (pseudo) test for the discrete choice routines. OLS assumes normality and constant residual errors. However, many

independent variables are not normally distributed (e.g., income, distance traveled, number of persons living in poverty).

Consequently, the use of OLS to test for Multicollinearity is exact only when the independent variable being examined for tolerance is normally distributed; otherwise, it is an approximate test. Nevertheless, it is useful indicator of multicollinearity. If the tolerance is low, that definitely indicates that there is multicollinearity. On the other hand, a high tolerance level does not necessarily indicate that there is little multicollinearity. From the test, a guidance message is displayed that indicates probable or possible Multicollinearity. If there is substantial Multicollinearity (indicated by low tolerance values), it is a good idea is to drop one of the multicolinear independent variables and re-run the model.

Save Output

The output from the discrete choice model can be saved.

Saved Multinomial Logit Output

For the multinomial logit model, the output is a 'dbf' file that includes all the input variables along with the estimated probability for each choice and the residual error for each choice (the observed choice, 1 or 0, minus the predicted probability). The probability and residual error is presented for each of the *J* alternatives. These are labeled with a 'P_ ' for probability and 'R_' for residual error. The different alternatives are indicated by a subscript from 0 (for the reference choice) through *J-1* (for the other alternatives) in the same order in which they are listed in Reference Choice dialogue (excluding the reference choice itself). For example, P_Choice0 is the estimated probability for choice 0 (the reference choice) while R_Choice3 is the estimated residual error for choice 3 (the third one listed in the list under Reference Choice excluding the reference choice itself).

Saved Conditional Logit Output

For the conditional logit model, the output is a 'dbf' file and includes all the input variables along with the estimated probability and the residual error for the case. For each case ID, there will be only one record that was chosen. Further, since the conditional logit model produces only one equation, there is only one probability and one residual error. The probability is labeled PREDPROB and the residual error is labeled RESID. The residual error can be used to compare different models. The MAD and MSPE statistics (discussed above) summarize the residual errors. But, a user might want to plot the residuals against one of the independent

Save Estimated Coefficients

The coefficients from either the multinomial logit or the conditional logit models can be saved for use with other data sets. Specify a directory where the coefficients file is to be saved and provide a root name. The saved coefficients file for the multinomial logit model will have a DCCoeffMNL prefix while the saved coefficients file for the conditional logit model will have a DCCoeffCNL prefix before the user defined root name.

Discrete Choice Modeling II

The Discrete Choice II module allows the user to apply the estimated coefficients from a discrete choice model to another data set (or a subset of the same data set) and calculate predicted probabilities, for either the multinomial logit or the conditional logit model. The 'Make prediction' routine allows the application of coefficients to a data set. The saved coefficients are applied to similar independent variables and to corresponding values of the choice variable to produce an estimated probability of an alternative.

Make Prediction

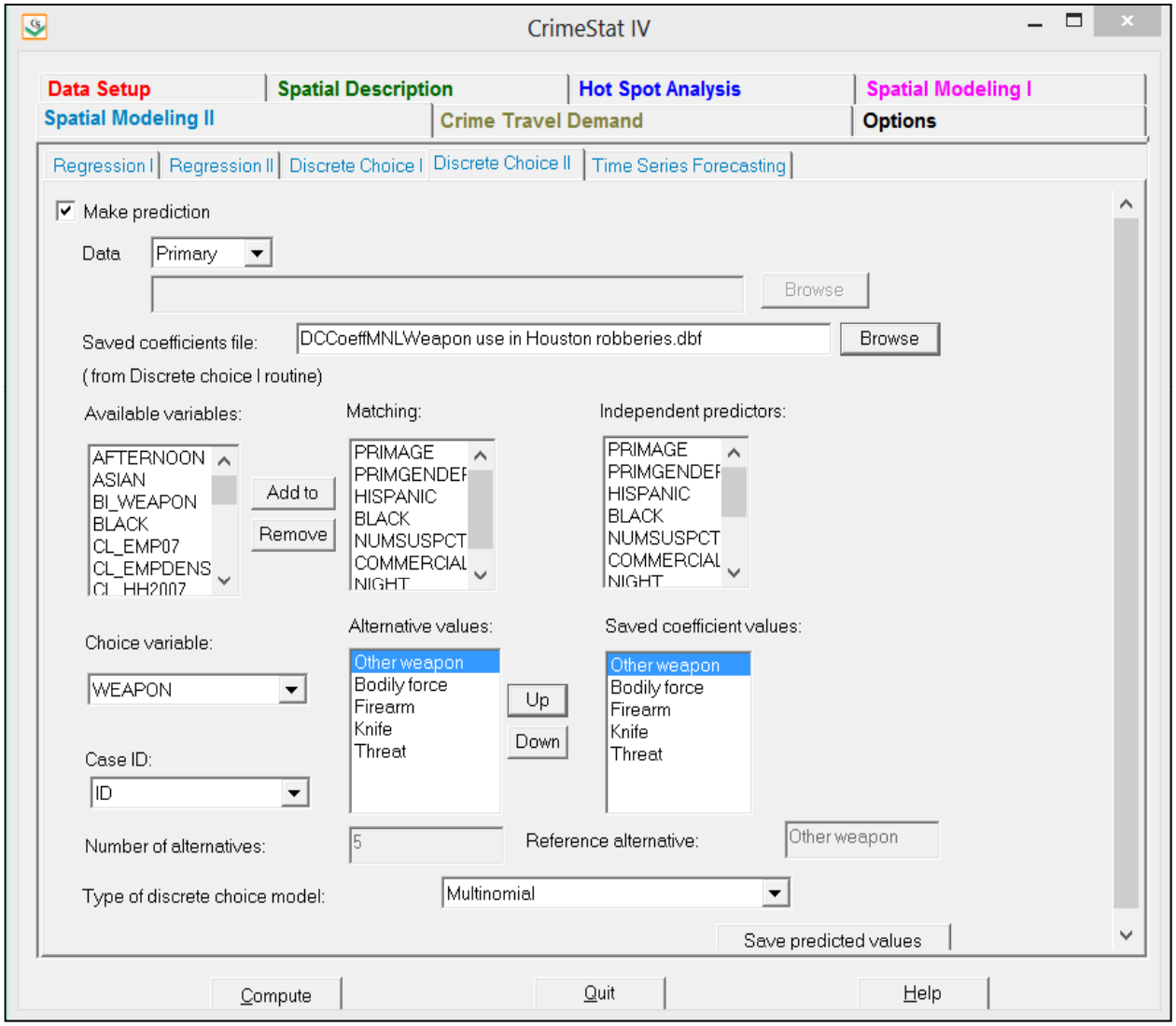
There are two types of models that can be fitted – multinomial logit or conditional logit. For both types of model, the coefficients file must include information on each of the coefficients. In addition, the coefficients model for the multinomial must include the value of the constant. The user reads in the saved coefficient file and matches the variables to those in the new data set based on the order of the coefficients file.

Discrete Choice Data File

The new data set can be either the Primary file or another file. If another file is being used, point to the directory where it is stored and identify it. The structure of the file for which a prediction is made must be the same as that from which the model was initially calibrated. That is, for a multinomial logit prediction, there must be a file with one record per decision maker and which includes and ID and each of the independent variables used in the prediction. For a

Figure 2.20:

Discrete Choice Modeling II



conditional logit prediction, there must be a joined file with a record for every combination of case and alternative.

Discrete Choice Saved Coefficients File

In order to make a prediction, a model must have already been estimated and the coefficients saved in a coefficients file. Point to the directory where the coefficients file has been saved and identify it.

Available Variables

The box labeled 'Available variables' will list all the fields on the input data set.

Independent Predictors

The independent variables that were used in the estimated coefficients file will be listed in the right column. They will be in the same order as was estimated in the calibration file.

Matching variables

Select corresponding variables from the input data file for the middle column. The items should be listed in the same order as in the 'independent predictors' column. They should be similar variables in content but need not have the names as in the original data file.

Alternative Values (multinomial logit only)

The values of the choice variables from the input file will be displayed in the middle column. The order should match the values in the adjacent saved coefficients file column. The 'Up' and 'Down' buttons can be used to re-order the values to be sure they are matched exactly.

Saved coefficient values *(multinomial logit model only)*

The values of the saved coefficients file will be displayed in the right column. Additional values can be added with the "Add to" button and existing values can be removed with the "Remove" button. It is essential that the values in the middle column match ***exactly*** their corresponding values in the right column.

Reference alternative *(multinomial logit only)*

The reference alternative value is displayed. If it is not correct, type in the correct value to be used or, better yet, re-estimate the original model. This field will be blanked out for the conditional logit model since it is not appropriate.

Discrete Choice Prediction Output

The screen output provides predictions of the value of the dependent variable in the same order as in the input data set. For the multinomial logit model, the predictions are labeled as CHOICE0 (for the reference choice), CHOICE1, CHOICE2, and so forth, in the same order as in the input data set. For each alternative, these predictions represent the probability that this alternative is chosen, given the values of the predictor variables.

For the conditional logit model, the prediction is applied to each available alternative. The screen output presents the predictions in matrix format with the case ID listed on the vertical axis and the choices listed on the horizontal axis (labeled CHOICE0, CHOICE1, CHOICE2, and so forth, in the same order as in the input data set).

Save Predicted Values for Discrete Choice Prediction

The predicted values and the residual errors can be output to a DBF file with a DCMakePredMNL<*root name*> for the multinomial logit and DCMakePredCNL<*root name*> for the conditional logit with the root name being provided by the user. The output files differ between the multinomial and conditional logit models.

Multinomial Logit Prediction Output

For the multinomial logit prediction, there is probability produced for each of the *J* alternatives. The probabilities are labeled P_CHOICE0 (for the reference choice), P_CHOICE1, P_CHOICE2, and so forth in the same order as in the Choice Values dialogue (with the exception of the reference choice which is always defined as P_CHOICE0). The probabilities will sum to 1.0 for all alternatives (within rounding-off error).

Conditional Logit Prediction Output

For the conditional logit prediction, there is a single probability output which is applied to the particular record. Since the data for the conditional logit model has a single record for each choice faced by the decision maker, the probability applies to that choice. The probabilities will sum to 1.0 for all alternatives (within rounding-off error). The column is labeled PREDPROB.

Time Series Forecasting

The Time Series Forecasting module is designed for the forecasting of crime or other counts by specific geographical areas (districts) and the detection of unusual levels of activity above-and-beyond the forecast. The methods are useful for tactical deployment of police resources but can be used by other fields where the monitoring of events by time is a regular part of their procedures. The module has a single interface page. It requires a user to specify an input file – either the Primary File or another file, identify variables in the file used for forecasting, select a seasonality adjustment, specify an exponential smoothing model, turn on the Trigg Tracking Signal, define Trigg parameter values, and save the output.

Input File

This is the file with the data for the Time Series Forecasting module. The data set for the regression module can be the Primary file or another file. If it is the Primary file, then it must have X and Y coordinates defined on each record. If it is another file, click on 'Other' and then identify the file. Only 'dbf' or 'txt' files are allowed.

Each record represents a unique combination of an area unit and a season number. A minimum of three years worth of data is required. For example, if there are 20 districts and monthly counts of the number of events over three years, then there will be 720 records (20 districts x 12 months x 3 years).

Areal Unit

The areal unit is the name or identifier for the district of the incident being forecasted. The name can be alphanumeric or numeric.

Year

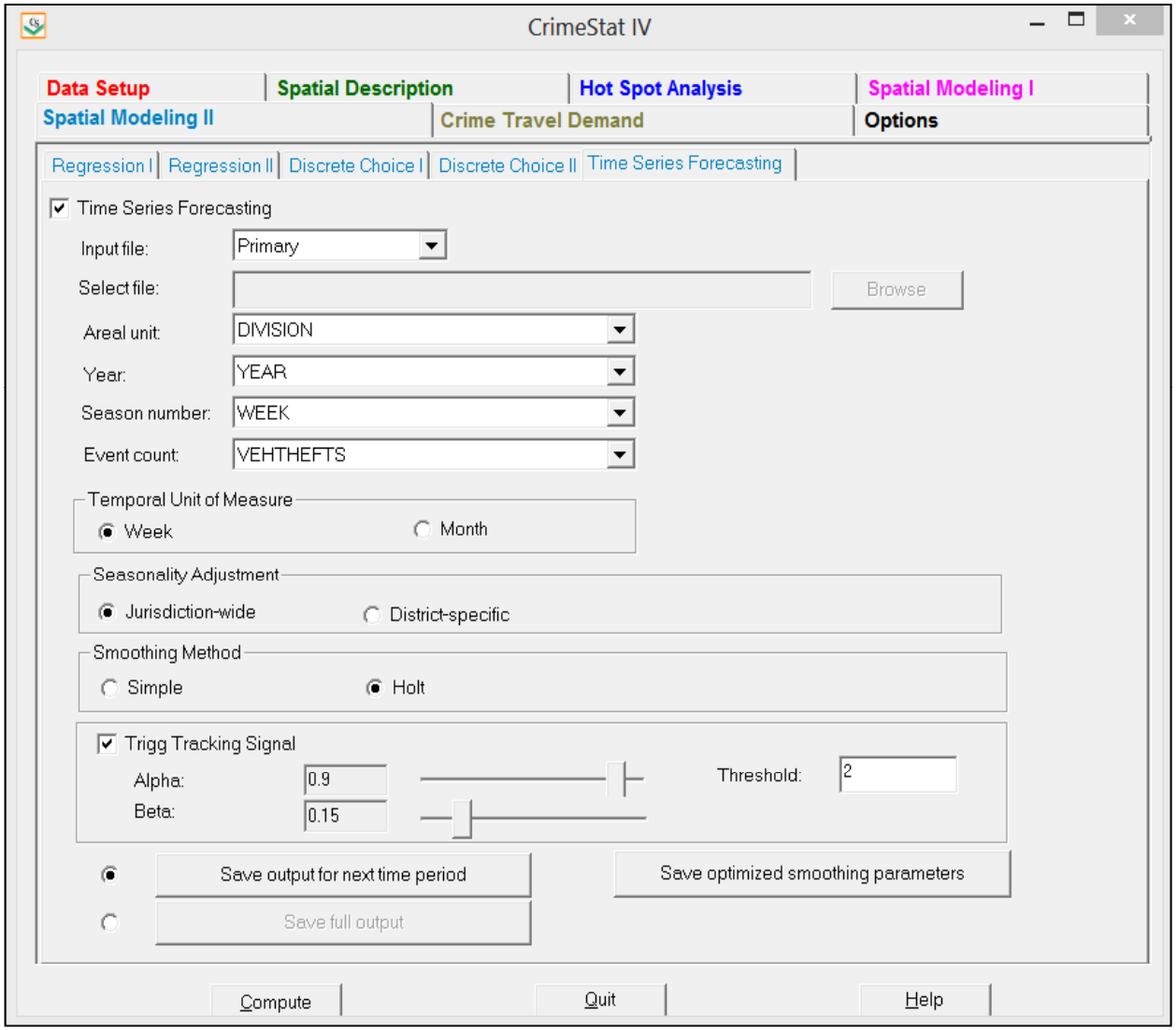
The year is the calendar year such as 2012 of each data record. This must be recorded. As mentioned above, there must be at least three years of data. This is a numeric variable.

Season Number

This is the season number. A season is the unique temporal identifier. With this module, only months or weeks are allowed. Thus, the season number is 1 through 12 for months and 1 through 52 for weeks. Note that there cannot be partial weeks. Since a year has 365 or

Figure 2.21:

Time Series Forecasting



366 days, there are 1 or 2 extra days left over. These must be assigned to either the first week of the next year or the last week of the current year.

Event Count

This is the count of the number of events for a given areal unit, year, and time period.

Temporal Unit of Measure

This field defines the type of season used, either week or month.

Seasonality Adjustment

The seasonality adjustment is the adjustment made for each time observation for seasonal patterns such as when, for example, crime is low in February and high in July relative to the time series trend line. The routine uses either the data from the entire jurisdiction (e.g., the entire city) - jurisdiction-wide, and applies this to each district or it uses individual data from each district so that each gets its own unique seasonal pattern - district-specific.

Smoothing Method

The smoothing method provides a more reliable estimate of the expected number of events based on past trends. The routine provides two alternative models, simple smoothing or Holt exponential smoothing. Simple smoothing assumes that there is no trend and that future values will follow past values. Holt smoothing adds a trend line into the expected number of future events. The models have smoothing parameters which CrimeStat automatically chooses by minimizing one-step-ahead forecast errors.

Trigg Tracking Signal

The Trigg Tracking Signal provides a test statistic for unusual activity in the number of events. If the absolute value of the signal exceeds a pre-specified threshold value, then there is a "signal trip" meaning that it is likely that there is an unusual change in events. The signal has three parameters with default values provided, alpha, beta and the threshold value.

Alpha and beta are parameters that vary between 0 and 1. An alpha of 0.9 makes the tracking signal very reactive to current data on the anticipation of changes in a time series pattern. A value of beta of 0.15 smoothes the measure of spread used to standardize the Trigg

signal and retains some history. Cohen, Garman, and Gorr (2009) found that these are the best performing parameter values. However, the user can experiment.

Alpha

Alpha is a smoothing parameter that varies between 0 and 1. An alpha of 0.9 (the default value) makes the tracking signal very reactive to current data on the anticipation of changes in a time series pattern. Note that "Alpha" is the same parameter as used in simple exponential smoothing for forecasting, but here is used to smooth the Trigg tracking signal instead of crime counts. Decreasing the parameter alpha below 0.9 will reduce the importance of more recent events.

Beta

Beta is a smoothing parameter that varies between 0 and 1. A value of beta of 0.15 (the default value) smooths the measure of spread used to standardize the Trigg signal and retains a good amount of history while allowing estimates to drift and follow changing spread in the data. Increasing beta above 0.15 will smooth the data more and will reduce the Trigg more towards the mean.

Threshold

The threshold is the value of the Trigg Tracking Signal that indicates whether the expected number of events will be greater than what is normally expected ("business-as-usual"). The default threshold of 1.5 is somewhat liberal in the sense that it will signal more periods of unusual activity. However, most police organizations would rather respond to more expected events even if the increased activity does not materialize (i.e., are false positives) than not respond and have events blow up. To use more conservative values, try 1.75 or 2.0 to get fewer signal trips.

Output

There are three types of output – full, one-step ahead, and the optimized smoothing parameters. The first two outputs produce the following calculated values:

1. DE_SEASON is the number of events per period (EVENTCOUNT) divided by the seasonal factor for the current observation's season (December) and, thus, is a de-seasonalized count of events. To calculate the seasonal factor for each record divide EVENTCOUNT by DE_SEASON.

2. SMTH_LEVEL is the smoothed estimate for the current observation (e.g., December 2012).

3. When using the Holt smoothing method, there is one additional estimated parameter. SMTH_SLOPE is the change in estimated crime for each step ahead. If, for example, you need the forecasts for February 2013 and your current time period is December 2012, you add two times SMTH_SLOPE to SMTH_LEVEL because February 2013 is two steps ahead of December 2012.

4. SQ_ERROR is the squared forecast error of the current observation from the forecast made for it from the previous period (e.g., November 2012 if the current period is December 2012).

5. TRIGG is the value of the Trigg Tracking Signal for the current observation.

6. SIGNALTRIP indicates whether the Trigg level was higher than the threshold. If it was, this field will have a **1** to indicate that the Trigg value was greater than or equal to the threshold selected and the detected change is an increase, a **-1** if the Trigg value is greater than or equal to the threshold but the detected change was a decrease, and a **0** otherwise.

7. FORECAST is the one-step-ahead forecast, for the next observation in time. For example, if the the current period is December 2012, then one-step ahead forecast is for January 2013. For a January 2013 forecast and simple exponential smoothing it is SMTH_LEVEL for December 2012 multiplied by the seasonal factor for January 2013. For January 2013 and Holt smoothing it is the sum of SMTH_LEVEL and SMTH_SLOPE times the seasonal factor for January 2013.

Save Full Output

The full output includes all input fields plus the calculated values. If the user clicks the Save full output button and then clicks the Save full output button, a save output window opens. Select dBase 'DBF' for the Save output to field, browse to the folder of your choice, and type a file name. Both the input data and the one-step ahead forecast are output to the screen and to a 'dbf' file. The file will be saved with a "TS_F" prefix before the defined file name.

Save Output for Next Time Period

The next time period output includes only the calculated fields for both the screen and saved file. The word "next" refers to the forecast made for the next time period, while the Trigg tracking signal evaluates the current period. Again, in the dialog for saving the output file, type the .dbf extension in the chosen file name. The file is saved as a 'dbf' file with a "TS_C" (for 'current') prefix.

Save Optimized Smoothing Parameters

The third type of output shows the results of the optimization process for exponential smoothing. This provides information on the parameters used to optimize the smoothing for each district. Define the file name and it will be saved as an ASCII text file with a 'txt' extension. The output fields are:

1. Optimum Alpha is the smoothing parameter value for *level* of a time series that minimizes the one-step-ahead forecast sum of squared errors.

2. Optimum Gamma is the smoothing parameter value for *time trend slope* of a time series that minimizes the one-step-ahead forecast sum of squared errors.

3. SSE is the resulting optimal sum of squared errors for the time series.

It is valuable to review the optimal parameters to see which areas have stable versus dynamic time series. Note that for the Trigg calculation, we want a large alpha to detect large changes in the number of recent events. That is why the default value of alpha is 0.9. However, for forecasting, we want a low alpha in order to smooth the data to produce a stable forecast.

VI. Crime Travel Demand Modeling

The crime travel demand module is a sequential model of crime travel by zone over a metropolitan area. Crime incidents are allocated to zones, both by the location where the crime occurred (destinations) and the location where the offender started (origins). A crime trip is defined as a crime event that originates at one location and ends at another location; the two locations can be the same. For each zone, the number of crimes originating in the zone and the number of crimes ending (occurring) in that zone are enumerated. Thus, the model is for count (or volumes), not rates. Other zonal data must be obtained to be used as predictor variables of the origin and destination counts.

The model is made up four sequential steps, each of which can involve smaller steps:

1. Trip generation – separate models are developed for predicting the number of crimes originating or ending in each zone. There are, therefore, two models. One is a model of the predicted number of crime trips that originate in each zone while the other is a model of the predicted number of crime trips that end in each zone. The number of origin zones can be greater than the number of destination zones.

2. Trip distribution – A model is developed for the number of crimes originating in each zone that go to each destination zone. The result is a prediction of the number of crimes originating in each zone that end in each zone (trip links).

3. Mode split – A model is developed that splits the number of predicted trips from each origin zone to each destination zone by travel mode (e.g., walking, bicycle, driving, bus, train). Thus, each zone-to-zone trip link is separated into different travel modes.

4. Network assignment – A model is developed for the route taken for each crime trip link (whether for all modes or by separate modes). Thus, the shortest path through a network is determined. Different travel modes will have different routes since bus and train, in particular, must use a separate network.

Crime Travel Demand Data Preparation

In order to run the crime travel demand module, particular data must be obtained and prepared. These involve:

1. A zonal framework that will be used for the modeling. In general, it is best to select the smallest zone size for which data can be obtained (e.g., block groups, census tracts, traffic analysis zones). However, it is often difficult to obtain data for the smallest units (e.g., blocks, grid cells). The larger the zone size, the more there will be intra-zonal trips and the greater the error in the model. Thus, the user must balance the need for small zones with the availability of data. Since crimes can occur outside a study area, the number of origin zones can be (and probably should be) greater than the number of destination zones. However, each destination zone should be included within the origin zone collection. Typically, there will be separate data sets for the origin zones and for the destination zones.

2. Data on crime origins and crime destinations are obtained (usually from arrest records) and are allocated to zones. The incidents are then summed by zone to produce a count. The "Assign primary points to secondary points" routine (under Distance analysis) can be used for this purpose. Thus, each origin zone has a count of the number of crimes originating in that zone and each destination zone has separate counts of the number of crimes originating in that zone and the number of crimes occurring (ending) in that zone. Crimes can be sub-divided into types (e.g., robbery, burglary, vehicle theft).

3. Additional data for the zones are obtained. These would include population (or households), sub-populations (e.g., age groups, race/ethnic groups), income levels, poverty levels, employment (retail and non-retail), land use, particular types of land use (e.g., drug locations, markets, parking lots), policing variables (e.g., personnel deployment, beat frequency), intervention variables (e.g., drug treatment centers), and other variables. It's important that all variables included must cover all zones for either the origin data set or the destination data set. For example, if poverty is used a variable in the origin model, then all origin zones must have an enumeration of poverty. Similarly, if retail employment is used as a variable in the destination model, then all destination zones must have an enumeration of retail employment.

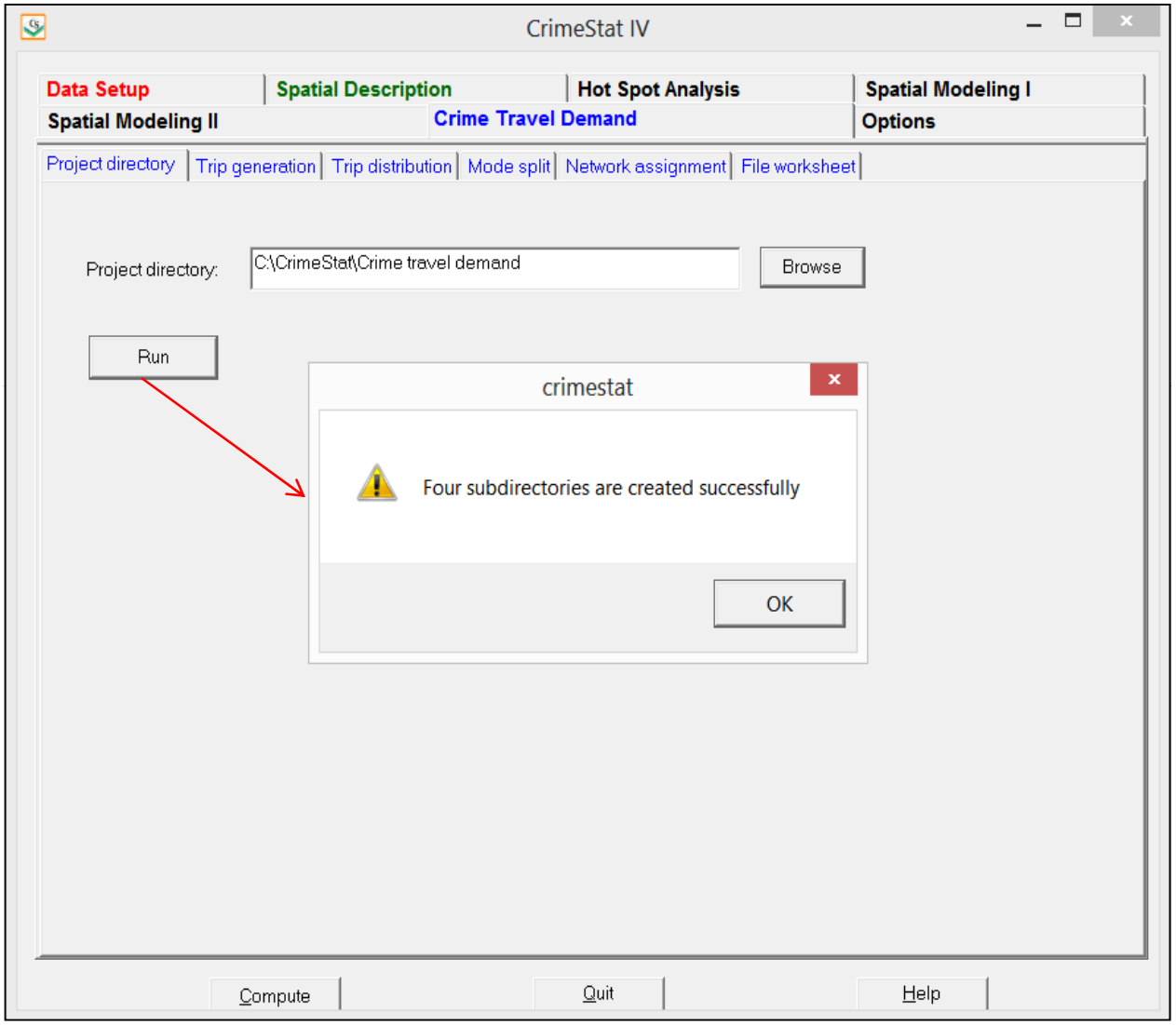
4. Data on dummy variables and special generators are also obtained. Dummy variables would be a proxy for a condition that does or does not exist. Zones that have the condition are assigned a '1' whereas zones that do not have the condition are assigned a '0'. For example, if a freeway cross a zone, then a freeway dummy variable would assign '1' to that zone (and all others that the freeway crossed) whereas all other zones received a '0' for this variable. A special generator is a land use that attracts trips (e.g., a stadium, a railroad station). All zones that have the special generator are assigned a value whereas all other zones receive a '0'; the value can either be a dummy variable (i.e., a '1') or the actual count if that can be obtained (e.g., the number of patrons at a football stadium event).

Project Directory

The Crime Travel Demand module is a complex model that involves many different files. Because of this, we recommend that the separate steps in the model be stored in separate directories under a main project directory. While the user can save any file to any directory

Figure 2.22:

Crime Travel Demand Project Directory



within the module, keeping the inputs and output files in separate directories can make it easier to identify files as well as examine files that have already been used at some later time.

Project Directory Utility

The project directory utility allows the creation of a master directory for a project and four separate sub-directories under the master directory that correspond to the four modeling stages.

The user puts in the name of a project in the dialogue box and points it to a particular drive and directory location (depending on the number of drives available to the user). For example, a project directory might be called "Robberies 2003" or "Bank robberies 2005". The utility then creates this directory if it does not already exist and creates four sub-directories underneath the project directory:

1. Trip generation
2. Trip distribution
3. Mode split
4. Network assignment

The user can then save the different output files into the appropriate directories. Further, for each sequential step in the crime travel demand model, the user can easily find the output file from the previous step which would become input file for the next step.

Trip Generation

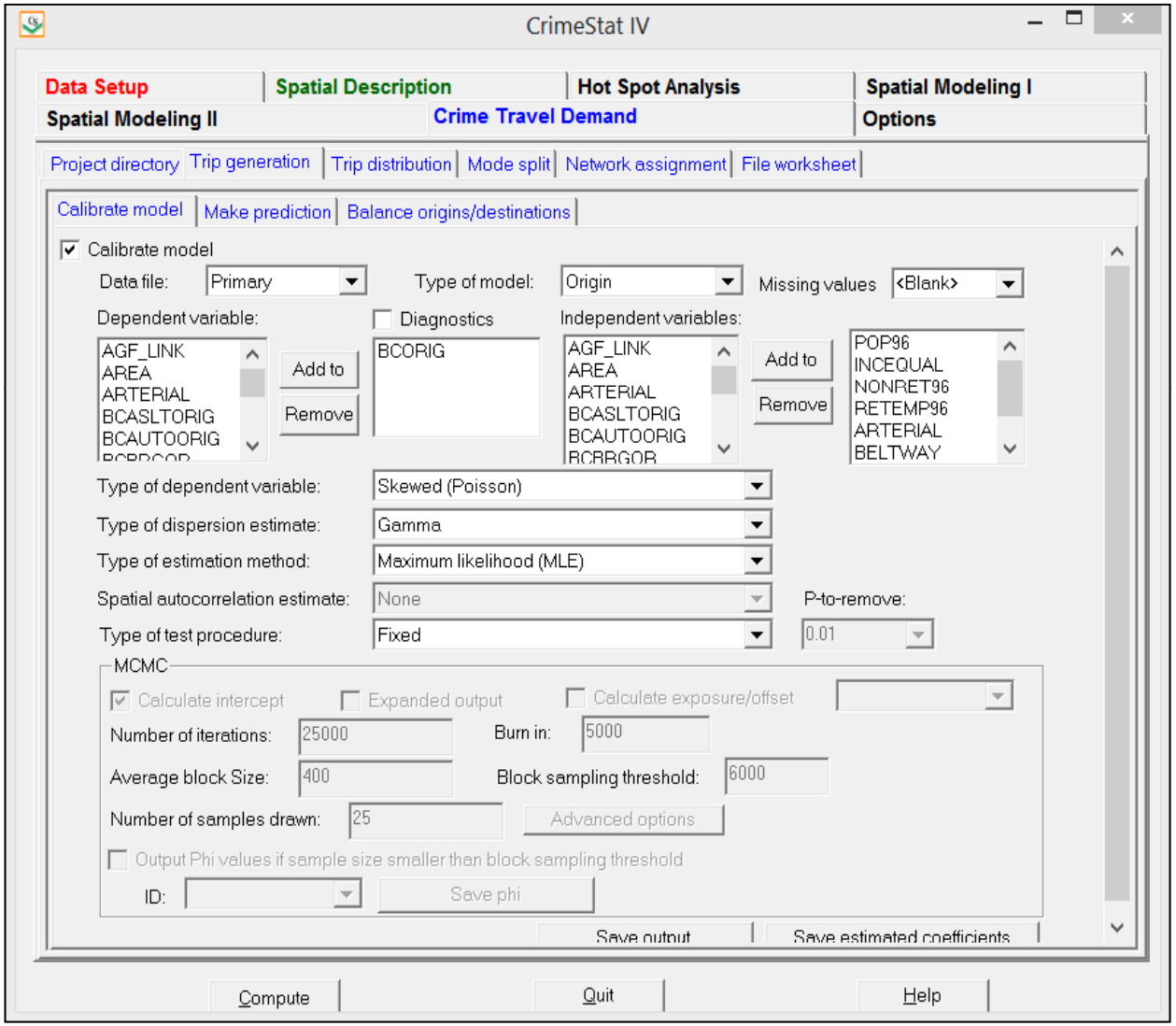
Trip generation involves the development of separate models for predicting the number of crimes originating in each zone and the number of crimes occurring (ending) in each zone. There are three steps to the trip generation:

1. **Calibrate model.** A step that calibrates the model against known data using regression techniques. The result is a prediction of the number of trips either originating in a zone (the origin model) or the number of trips ending in a zone (the destination model).

2. **Make prediction**. A step that applies the calibrated model to a data set and also allows the addition of trips from outside the study area (external trips).

Figure 2.23:

Trip Generation Modeling



1. **Balance predicted origins & destinations**. A step that ensures that the number of predicted origins equals the number of predicted destinations. Since a trip involves an origin and a destination, it is essential that the number of origins equal the number of destinations.

Calibrate Trip Generation Model

This step involves calibrating a regression model against the zonal data. Two separate models are developed, one for trip origins and one for trip destinations. The dependent variable is the number of crimes originating in a zone (for the trip origin model) or the number of crimes ending in a zone (for the trip destination model). The independent variables are zonal variables that may predict the number of origins or destinations.

In the current version, 13 possible regression models are available with several options for each of these:

MLE Normal (OLS)
MCMC Normal
MCMC Normal-CAR
MCMC Normal-SAR
MLE Poisson
MLE Poisson with linear dispersion correction (NB1)
MLE Poisson-Gamma (NB2)
MCMC Poisson-Gamma (NB2)
MCMC Poisson-Gamma-CAR
MCMC Poisson-Gamma-SAR
MCMC Poisson-Lognormal
MCMC Poisson-Lognormal-CAR
MCMC Poisson-Lognormal-SAR

Since the Regression I module and Trip Generation module duplicate most of the regression functions, only one of these can be run at a time.

Input Data set

The data set for the trip generation must be the Primary File data set. The coordinate system and distance units are also the same.

Dependent Variable

To start loading the module, click on the 'Calibrate model' tab. A list of variables from the Primary File is displayed. There is a box for defining the dependent variable. The user must choose one dependent variable.

Independent Variables

There is a box for defining the independent variables. The user must choose one or more independent variables. There is no limit to the number. The variables are output in the same order as specified in the dialogue so a user should consider how these are to be displayed.

Model decisions

There are five decisions that must be made for each regression model.

Type of Dependent Variable

The first model decision is the type of dependent variable The first model decision is the type of dependent variable: Skewed (Poisson) or Normal (OLS). The default is a Poisson.

Type of Dispersion Estimate

The second model decision is the type of dispersion estimate to be used. The choices are Gamma, Poisson, Poisson with linear correction, Normal (automatically defined for the Normal model), or lognormal. The default is Gamma.

Type of Estimation Method

The third model decision is the type of estimation method to be used: Maximum Likelihood (MLE) or Markov Chain Monte Carlo (MCMC). The default is MLE.

Spatial Autocorrelation Regression Model

If the user accepts an MCMC algorithm, then a fourth decision is whether to run a spatial autocorrelation estimate along with it. This can only be run if the dependent variable is Poisson and MCMC has been chosen as the type of estimation method. The spatial autocorrelation choices are Conditional Autoregressive (CAR) or Simultaneous Autoregression (SAR).

Type of Test Procedure

The fifth, and last model decision, is whether to run a fixed model or a backward elimination *stepwise* procedure (only with an MLE model). A fixed model includes all selected independent variables in the regression whereas a backward elimination model starts with all selected variables in the model but proceeds to drop variables that fail the P-to-remove test, one at a time. Any variable that has a significance level in excess of the P-to-remove value is dropped from the equation.

Specify whether a fixed model (all selected independent variables are used in the regression) or a backward elimination stepwise model is used. The default is a fixed model. If a backward elimination stepwise model is selected, choose the P-to-remove value (default is .01).

MCMC Model Choices

If the user chooses the MCMC algorithm, then eight additional decisions have to be made.

Number of Iterations

The first MCMC decision is the number of iterations to be run. The default is 25,000. The number should be sufficient to produce reliable estimates of the parameters. Check the MC Error/Standard deviation ratio and the G-R statistic after the run to be sure most parameters are below 1.05 and 1.20 respectively. If not, increase the number of iterations and 'burn in' iterations.

'Burn in' iterations

The second MCMC decision is the number of initial iterations that will be dropped from the final distribution (the 'burn in' period). The default is 5,000. The number of 'burn in' iterations should be sufficient for the algorithm to reach an equilibrium state and produce reliable estimates of the parameters. Check the MC Error/Standard deviation ratio and the G-R statistic after the run to be sure most parameters are below 1.05 and 1.20 respectively. If not, increase the number of iterations and 'burn in' iterations.

Block Sampling Threshold

The third MCMC decision is whether to run all the records through the MCMC algorithm or whether to draw block samples. The algorithm will be run on all records unless the number of records exceeds the block sampling threshold. The default threshold is 6000 records. To run all the records through the MCMC algorithm, change this value to be greater than the number of

records in the database. Note that calculation time will increase substantially if all records in a large database are run through the algorithm.

Average Block Size

The fourth MCMC decision is the number of records to be drawn for each block sample if the total number of records is greater than the block sampling threshold. The default is 400 records per block sample. Note that this is an average. Actual samples will vary in size. The output will display the expected sample size and the average sample size that was drawn.

Number of Samples Drawn

The fifth MCMC decision is the number of samples to be drawn if the total number of records is greater than the block sampling threshold. The default is 25 block samples. Typically, 20-30 block samples will achieve stable model results.

Calculate Intercept

The sixth MCMC decision is whether to run a model with or without an intercept (constant). The default is with an intercept estimated. To run the model without the intercept, uncheck the 'Calculate intercept' box.

Calculate Exposure/Offset

The seventh MCMC decision is whether to run a risk model. If the model is a risk or rate model, then an exposure (offset) variable needs to be defined. Check the 'Calculate exposure/offset' box and identify the variable that will be used as the exposure variable. The coefficient for this variable will automatically be 1.0.

Advanced Options

The eighth MCMC decision is the prior values used for the different parameters being estimated. The MCMC algorithm requires an initial estimate for each parameter. There is a dialogue of advanced options for the MCMC algorithm by which they can be changed.

Initial Parameters Values

For the beta coefficients (including the intercept), the default values are 0. These are displayed as a blank screen for the Beta box. However, other prior estimates of the beta

coefficients can be substituted for the assumed 0 coefficients. To do this, all independent variable coefficients plus the intercept (if used) must be listed in the order in which they appear in the model and must be separated by commas. Do not include the beta coefficients for the spatial autocorrelation term (if used) or the error (Taupsi) term.

Taupsi (error term)

The output of the MCMC always includes an error term, called *Taupsi* (τ_ψ). This is an exponent of the error term, $e^{\tau\psi}$, which together is called the *dispersion parameter*. The default value for Taupsi is 1.0. The user can substitute an alternative value.

Rho and Tauphi

The spatial autocorrelation component is made up of three separate sub-components, called Rho, Tauphi, and Alpha and are additive. Rho is roughly a global component that applies to the entire data set. Tauphi is roughly a neighborhood component that applies to a sub-set of the data. Alpha is essentially a localized effect. The default initial values for Rho and Tauphi are 0.5 and 1 respectively. The user can substitute alternative values for these parameters.

Alpha

Alpha is the exponent for the distance decay function in the spatial model. Essentially, the distance decay function defines the weight to be applied to the values of nearby records. The weight can be defined by one of three mathematical functions. First, the weight can be defined by a negative exponential function.

Second, the weight can be defined by a restricted negative exponential with the negative exponential operating up to the specified search distance, whereupon the weight becomes 0 for greater distances.

Third, the weight can be defined as a uniform value for all other observations within a specified search distance. This is a *contiguity* (or adjacency) measure. Essentially, all other observations have an equal weight within the search distance and 0 if they are greater than the search distance.

For the negative exponential and restricted negative exponential functions, substitute the selected value for alpha in the alpha box and for the restricted negative exponential and uniform functions, specify the search distance and distance units. The default is a negative exponential with an alpha of -1.0 in miles.

Value for 0 distance between records

The advanced options dialogue has a parameter for the minimum distance to be assumed between different records. If two records have the same X and Y coordinates (which could happen if the records are individual events, for example), then the distance between these records will be 0. This could cause unusual calculations in estimating spatial effects. Instead, it is more reliable to assume a slight difference in distance between all records. The default is 0.005 miles but the user can modify this (including substituting 0 for the minimal distance).

Output

The output depends on whether an MLE or an MCMC model has been run.

Maximum Likelihood (MLE) Model Output

The MLE routines (Normal/OLS, Poisson, Poisson with linear correction, MLE Poisson-Gamma, Binomial Probit, MLE Binomial Logit) produce a standard output that includes summary statistics and estimates for the individual coefficients.

MLE Summary Statistics

The summary statistics include:

Information about the model

1. The dependent variable
2. The number of records
3. The degrees of freedom (N – number of parameters estimated)
4. The type of regression model (Normal/OLS, Poisson, Poisson with linear correction, Poisson-Gamma, Binomial Probit, Binomial Logit)
5. The method of estimation (MLE)

Likelihood statistics

6. Log-likelihood estimate, which is a negative number. For a set number of independent variables, the more negative the log-likelihood the better.
7. Akaike Information Criterion (AIC) adjusts the log-likelihood for the degrees of freedom. The smaller the AIC, the better.

8. Bayesian Information Criterion (BIC), sometimes known as the Schwartz Criterion (SC), adjusts the log-likelihood for the degrees of freedom. The smaller the BIC, the better.
9. Deviance compares the log-likelihood of the model to the log-likelihood of a model that fits the data perfectly. A smaller deviance is better.
10. The probability value of the deviance based on a Chi-square with k-1 degrees of freedom.
11. Pearson Chi-square is a test of how closely the predicted model fits the data. A smaller Chi-square is better since it indicates the model fits the data well.

Model error estimates

12. Mean Absolute Deviation (MAD). For a set number of independent variables, a smaller MAD is better.
13. Quartiles for the Mean Absolute Deviation. For any one quartile, smaller is better.
14. Mean Squared Predictive Error (MSPE). For a set number of independent variables, a smaller MSPE is better.
15. Quartiles for the Mean Squared Predictive Error. For any one quartile, smaller is better.
16. Squared multiple R (for linear model only). This is the percentage of the dependent variable accounted for by the independent variables.
17. Adjusted squared multiple R (for linear model only). This is the squared multiple R adjusted for degrees of freedom.

Over-dispersion tests

18. Adjusted deviance. This is a measure of the difference between the observed and predicted values (the residual error) adjusted for degrees of freedom. The smaller the adjusted deviance, the better. A value greater than 1 indicates over-dispersion.
19. Adjusted Pearson Chi-square. This is the Pearson Chi-square adjusted for degrees of freedom. The smaller the Pearson Chi-square, the better. A value greater than 1 indicates over-dispersion.
20. Dispersion multiplier. This is the ratio of the expected variance to the expected mean. For a set number of independent variables, the smaller the dispersion multiplier, the better. For example, in a pure Poisson distribution, the dispersion should be 1.0. In practice, a ratio greater than 10 indicates that there is too much

variation that is unaccounted for in the model. Either add more variables or change the functional form of the model

21. Inverse dispersion multiplier. For a set number of independent variables, a larger inverse dispersion multiplier is better. A ratio close to 1.0 is considered good.

MLE Individual Coefficient Statistics

For the individual coefficients, the following are output:

22. The coefficient. This is the estimated value of the coefficient from the maximum likelihood estimate.
23. Standard Error. This is the estimated standard error from the maximum likelihood estimate.
24. Pseudo-tolerance. This is the tolerance value based on a linear prediction of the variable by the other independent variables. See equation Up. 2.18.
25. Z-value. This is asymptotic Z-test that is defined based on the coefficient and standard error. It is defined as Coefficient/Standard Error.
26. p-value. This is the two-tail probability level associated with the Z-test.

Markov Chain Monte Carlo (MCMC) Model Output

The MCMC routines (Poisson-Gamma, Poisson-Gamma-CAR/SAR, Poisson-Lognormal, Poisson-Lognormal-CAR/SAR, Binomial Logit, Binomial Logit-CAR/SAR) produce a standard output and an optional expanded output**.** The standard output includes summary statistics and estimates for the individual coefficients.

MCMC Summary Statistics

The summary statistics include:

Information about the model

1. The dependent variable
2. The number of records
3. The sample number. This is only output when the block sampling method is used.
4. The number of cases for the sample. This is only output when the block sampling method is used.

5. Date and time for sample. This is only output when the block sampling method is used
6. The degrees of freedom (N – number of parameters estimated)
7. The type of regression model (Poisson-Gamma, Poisson-Gamma-CAR/SAR, Poisson-Lognormal, Poisson-Lognormal-CAR/SAR, Binomial Logit, Binomial Logit-CAR/SAR)
8. The method of estimation
9. The number of iterations
10. The 'burn in' period
11. The distance decay function used. This is output for CAR/SAR models only.
12. The block size is the expected number of records selected for each block sample. The actual number may vary.
13. The number of samples drawn, output when the block sampling method used.
14. The average block size. This is output when the block sampling method used.
15. The type of distance decay function. This is output for CAR/SAR models only.
16. Condition number for the distance matrix. If the condition number is large, then the model may not have properly converged. This is output for CAR/SAR models only.
17. Condition number for the inverse distance matrix. If the condition number is large, then the model may not have properly converged. This is output for CAR/SAR models only.

Likelihood statistics

18. Log-likelihood estimate, which is a negative number. For a set number of independent variables, the smaller the log-likelihood (i.e., the most negative) the better.
19. Deviance Information Criterion (DIC) adjusts the log-likelihood for the effective degrees of freedom. The smaller the DIC, the better.
20. Akaike Information Criterion (AIC) adjusts the log-likelihood for the degrees of freedom. The smaller the AIC, the better.
21. Bayesian Information Criterion (BIC), sometimes known as the Schwartz Criterion (SC), adjusts the log-likelihood for the degrees of freedom. The smaller the BIC, the better.
22. Deviance compares the log-likelihood of the model to the log-likelihood of a model that fits the data perfectly. A smaller deviance is better.
23. The probability value of the deviance based on a Chi-square with k-1 degrees of freedom.

24. Pearson Chi-square is a test of how closely the predicted model fits the data. A smaller Chi-square is better since it indicates the model fits the data well.

Model error estimates

25. Mean Absolute Deviation (MAD). For a set number of independent variables, a smaller MAD is better.
26. Quartiles for the Mean Absolute Deviation. For any one quartile, smaller is better.
27. Mean Squared Predictive Error (MSPE). For a set number of independent variables, a smaller MSPE is better.
28. Quartiles for the Mean Squared Predictive Error. For any one quartile, smaller is better.

Over-dispersion tests

29. Adjusted deviance. This is a measure of the difference between the observed and predicted values (the residual error) adjusted for degrees of freedom. The smaller the adjusted deviance, the better. A value greater than 1 indicates over-dispersion.
30. Adjusted Pearson Chi-square. This is the Pearson Chi-square adjusted for degrees of freedom. The smaller the Pearson Chi-square, the better. A value greater than 1 indicates over-dispersion.
31. Dispersion multiplier. This is the ratio of the expected variance to the expected mean. For a set number of independent variables, the smaller the dispersion multiplier, the better. In a pure Poisson distribution, the dispersion should be 1.0. In practice, a ratio greater than 10 indicates that there is too much variation that is unaccounted for in the model. Either add more variables or change the functional form of the model.
32. Inverse dispersion multiplier. For a set number of independent variables, a larger inverse dispersion multiplier is better. A ratio close to 1.0 is considered good.

MCMC Individual Coefficients Statistics

For the individual coefficients, the following are output:

33. The mean coefficient. This is the mean parameter value for the *N-k* iterations where *k* is the 'burn in' samples that are discarded. With the MCMC block sampling method, this is the mean of the mean coefficients for all block samples.
34. The standard deviation of the coefficient. This is an estimate of the standard error of the parameter for the *N-k* iterations where *k* is the 'burn in' samples that are

discarded. With the MCMC block sampling method, this is the mean of the standard deviations for all block samples.

35. t-value. This is the t-value based on the mean coefficient and the standard deviation. It is defined by Mean/Std.
36. p-value. This is the two-tail probability level associated with the t-test.
37. Adjusted standard deviation (Adj. Std). The block sampling method will produce substantial variation in the mean standard deviation, which is used to estimate the standard error. Consequently, the standard error will be too large. An approximation is made by multiplying the estimated standard deviation by $\sqrt{\frac{\bar{n}}{N}}$ where $\bar{n}$ is the average sample size of the block samples and N is the number of records. If no block samples are taken, then this statistic is not calculated.
38. Adjusted t-value. This is the t-value based on the mean coefficient and the adjusted standard deviation. It is defined by Mean/Adj_Std. If no block samples are taken, then this statistic is not calculated.
39. Adjusted p-value. This is the two-tail probability level associated with the adjusted t-value. If no block samples are taken, then this statistic is not calculated.
40. MC error is a Monte Carlo simulation error. It is a comparison of the means of m individual chains relative to the mean of the entire chain. By itself, it has little meaning.
41. MC error/Std is the MC error divided by the standard deviation. If this ratio is less than .05, then it is a good indicator that the posterior distribution has converged.
42. G-R stat is the Gelman-Rubin statistic which compares the variance of m individual chains relative to the variance of the entire chain. If the G-R statistic is under 1.2, then the posterior distribution is commonly considered to have converged.
43. Spatial autocorrelation term (Phi) for Poisson-Gamma-CAR models only. This is the estimate of the fixed effect spatial autocorrelation effect. It is made up of three components: a global component (Rho); a local component (Tauphi); and a local neighborhood component (Alpha, which is defined by the user).

Expanded Output (MCMC only)

If the expanded output box is selected, additional information on the percentiles from the MCMC sample are displayed. If the block sampling method is used, the percentiles are the means of all block samples. The percentiles are:

44. 2.5th percentile
45. 5th percentile
46. 10th percentile
47. 25th percentile
48. 50th percentile (median)
49. 75TH percentile
50. 90th percentile
51. 95th percentile
52. 97.5th percentile

The percentiles can be used to construct confidence intervals around the mean estimates or to provide a non-parametric estimate of significance as an alternative to the estimated t-value in the standard output. For example, the 2.5th and 97.5th percentiles provide approximate 95 percent confidence intervals around the mean coefficient while the 0.5th and 99.5th percentiles provide approximate 99 percent confidence intervals.

The percentiles will be output for all estimated parameters including the intercept, each individual predictor variable, the spatial effects variable (Phi), the estimated components of the spatial effects (Rho and Tauphi), and the overall error term (Taupsi).

Output Phi Values (CAR/SAR models only)

For CAR or SAR models only, the individual Phi values can be output. This will occur if the sample size is smaller than the block sampling threshold. Check the 'Output Phi value if sample size smaller than block sampling threshold' box. An ID variable must be identified and a DBF output file defined.

Multicollinearity Among the Independent Variables

A major consideration in any regression model is that the independent variables are statistically independent. Non-independence is called *Multicollinearity*. Non-independence means that there is overlap in prediction among two or more independent variables. This can lead to uncertainty in interpreting coefficients as well as an unstable model that may not hold in the future. Generally, it is a good idea to reduce Multicollinearity as much as possible. A tolerance test is given for each coefficient. This is defined as 1 – the R-square of the independent variable predicted by the remaining independent variables in the equation using an Ordinary Least Squares model. It is an indicator of how much the other independent variables in the equation account for the variance of any particular independent variable. Since the method uses the Ordinary Least Squares methods, it is an approximate (pseudo) test for the Poisson

regression routines. A message is displayed that indicates probable or possible Multicollinearity. A good idea is to drop one of the multicolinear independent variables and re-run the model. However, each of the coefficients should be inspected carefully before accepting a final model.

Graph of Residual Errors

While the output page is open, clicking on the graph button will display a graph of the residual errors (on the Y axis) against the predicted values (on the X axis). Only residual errors that vary between -200 and +200 are shown to allow most of the errors to be displayed.

Save Output

The predicted values and the residual errors can be output to a DBF file with a TripGenOut<*root name*> with the root name being provided by the user. The output includes all the variables in the input data set plus two new ones: 1) the predicted values of the dependent variable for each observation (with the field name PREDICTED); and 2) the residual error values, representing the difference between the actual /observed values for each observation and the predicted values (with the field name RESIDUAL). The file can be imported into a spreadsheet or graphics program and the errors plotted against the predicted dependent variable.

Save Estimated Coefficients

The individual coefficients can be output to a DBF file with a TripGenCoeff<*root name*> with the root name being provided by the user. This file can be used in the 'Make Prediction' routine of the Trip Generation module.

Diagnostic Tests

The regression module has a set of diagnostic tests for evaluating the characteristics of the data and the most appropriate model to use. There is a diagnostics box on the 'Calibrate model' page.

Diagnostics are provided on:

1. The minimum and maximum values for the dependent and independent variables
2. Skewness in the dependent variable
3. Spatial autocorrelation in the dependent variable

4. Estimated values for the distance decay parameter – alpha, for use in CAR/SAR models
5. Multicolinarity among the independent variables

Minimum and Maximum Values for the Variables

First, the minimum and maximum values of both the dependent and independent variables are listed. A user should look for ineligible values (e.g., -1) as well as variables that have a very high range. The MLE routines are sensitive to variables with very large ranges.

Skewness Tests

Skewness in the dependent variable can distort a linear model by allowing high values to be underestimated while allowing low values to be overestimated and a Poisson-type model is preferred over the linear for highly skewed variables.

The diagnostics utility tests for skewness using two different measures: 1) the "***g"*** statistic, and 2) the ratio of the simple variance to the simple mean. Either significant "g" scores or variance-to-mean ratios greater than about 2:1 should make the user cautious about using a linear model. If either measure indicates skewness, *CrimeStat* prints out a message indicating the dependent variable appears to be skewed and that a Poisson-based model should be used.

Testing for Spatial Autocorrelation in the Dependent Variable

The third type of test in the diagnostics utilities is the Moran's "I" coefficient for spatial autocorrelation. If the "I" is significant, *CrimeStat* outputs a message indicating that there is definite spatial autocorrelation in the dependent variable and that it needs to be accounted for, either by a proxy variable or by estimating a CAR or SAR model.

Estimating the Value of Alpha for CAR or SAR Models

The fourth type of diagnostic test is an estimate of a plausible value for the distance decay function, α, in CAR or SAR models. Three values of alpha are given in different distance units, one associated with a weight of 0.9 (a very steep distance decay), one associated with a weight of 0.75 (a moderate distance decay), and one associated with a weight of 0.5 (a shallow distance decay). Users should run the Moran Correlogram and examine the graph of the drop off in spatial autocorrelation to assess what type of decay function most likely exists. The user should choose an alpha value that best represents the distance decay and should define the distance units for it.

Multicollinearity Test

The fifth type of diagnostic test is for Multicollinearity among the independent predictors. The tolerance test is presented for each independent variable. This is defined as $1-R^2$ for the other independent variables in the equation. Each independent variable should have a high tolerance (0.90 or higher). *CrimeStat* prints out an error message if tolerance is not high.

Make Trip Generation Prediction

This routine applies an already-calibrated regression model to a data set. This would be useful for several reasons: 1) if external trips are to be added to the model (which is normally preferred); 2) if the model is applied to another data set; and 3) if variations on the coefficients are being tested with the same data set. The model will need to be calibrated first (see Calibrate trip generation model) and the coefficients saved as a parameters file. The coefficient parameter file is then re-loaded and applied to the data.

Data Input File

The data file is input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Type of Model

Specify whether the model is for origins or destinations. This will be printed out on the output header.

Trip Generation Parameters (coefficients) File

This is the saved coefficient parameter file. It is DBF with a TRIPGENCOEFF prefix. Load the file by clicking on the Browse button and finding the file. Once loaded, the variable names of the saved coefficients are displayed in the "Matching parameters" box.

Independent Variables

Select independent variables from the list of variables in the data file. Up to 15 variables can be selected.

Matching Parameters

The selected independent variables need to be matched to the saved variables in the trip generation parameters file **in the same order**. Add the appropriate variables one by one in the order in which they are listed in the matching parameters box. It is essential that the order by the same otherwise the coefficients will be applied to the wrong variables.

If the model had estimated a general spatial effect from a CAR or SAR model, then the general Phi will have been saved with the coefficient files. If the model had estimated specific spatial effects from a CAR or SAR model, then the specific Phi values will have been saved in a separate Phi coefficients file. In the latter case, the user must read in the Phi coefficients file along with the general coefficient file.

Missing Values

Specify any missing value codes for the variables. Blank records will automatically be considered as missing. If any of the selected dependent or independent variables have missing values, those records will be excluded from the analysis.

Add External Trips

External trips are trips that start outside the modeled study area. Because they are crimes that originate outside the study area, they were not included in the zones used for the origin model. Therefore, they have to be independently estimated and added to the origin zone total to make the number of origins equal to the number of destinations. Click on the "Add external trips" button to enable this feature.

Number of external trips

Add the number of external trips to the box. This number will be added as an extra origin zone (the External zone).

Origin ID

Specify the origin ID variable in the data file. The external trips will be added as an extra origin zone, called the "External" zone. Note: all destination ID's should be in the origin zone file and must have the same names. This is necessary for subsequent modeling stages.

Type of Regression Model

Specify the type of regression model to be used. The default is a Poisson regression and the other alternative is a Linear (Ordinary Least Squares) regression.

Save Predicted Values

The output is saved as a 'dbf' file under a different file name with a TripGenMakePred<*root name*> with the root name being provided by the user. The output includes all the variables in the input data set plus the predicted values of the dependent variable for each observation (with the name PREDICTED. In addition, *if* external trips were added, then there is a new record with the name EXTERNAL listed in the Origin ID column. This record lists the added trips in the PREDICTED column and zeros (0) for all other numeric fields.

Output

The tabular output includes summary information about file and lists the predicted values for each input zone.

Balance Origins and Destinations

Since, by definition, a 'trip' has an origin and a destination, the number of predicted origins must equal the number of predicted destinations. Because of slight differences in the data sets of the origin model and the destination model, it is possible that the total number of predicted origins (including any external trips – see Make trip generation prediction) may not equal the total number of predicted destinations. This step, therefore, is essential guarantee that this condition will be true. The routine adjusts either the number of predicted origins or the number of predicted destinations so that the condition holds. The trip distribution routines will not work unless the number of predicted origins equals the number of predicted destinations (within a very small rounding-off error).

Predicted origin file

Specify the name of the predicted origin file by clicking on the Browse button and locating the file.

Origin variable

Specify the name of the variable for the predicted origins (e.g., PREDICTED).

Predicted destination file

Specify the name of the predicted destination file by clicking on the Browse button and locating the file.

Destination variable

Specify the name of the variable for the predicted origins (e.g., PREDICTED).

Balancing method

Specify whether origins or destinations are to be held constant. The default is 'Hold destinations constant'.

Save predicted origin/destination file

The output is saved as a 'dbf' file under a different file name. The output includes all the variables in the input data set plus the adjusted values of the predicted values of the dependent variable for each observation. If destinations are held constant, the adjusted variable name for the predicted trips is ADJORIGIN. If origins are held constant, the adjusted variable name for the predicted trips is ADJDEST.

Output

The tabular output includes file summary information plus information about the number of origins and destinations before and after balancing. In addition, the predicted values of the dependent variable are displayed.

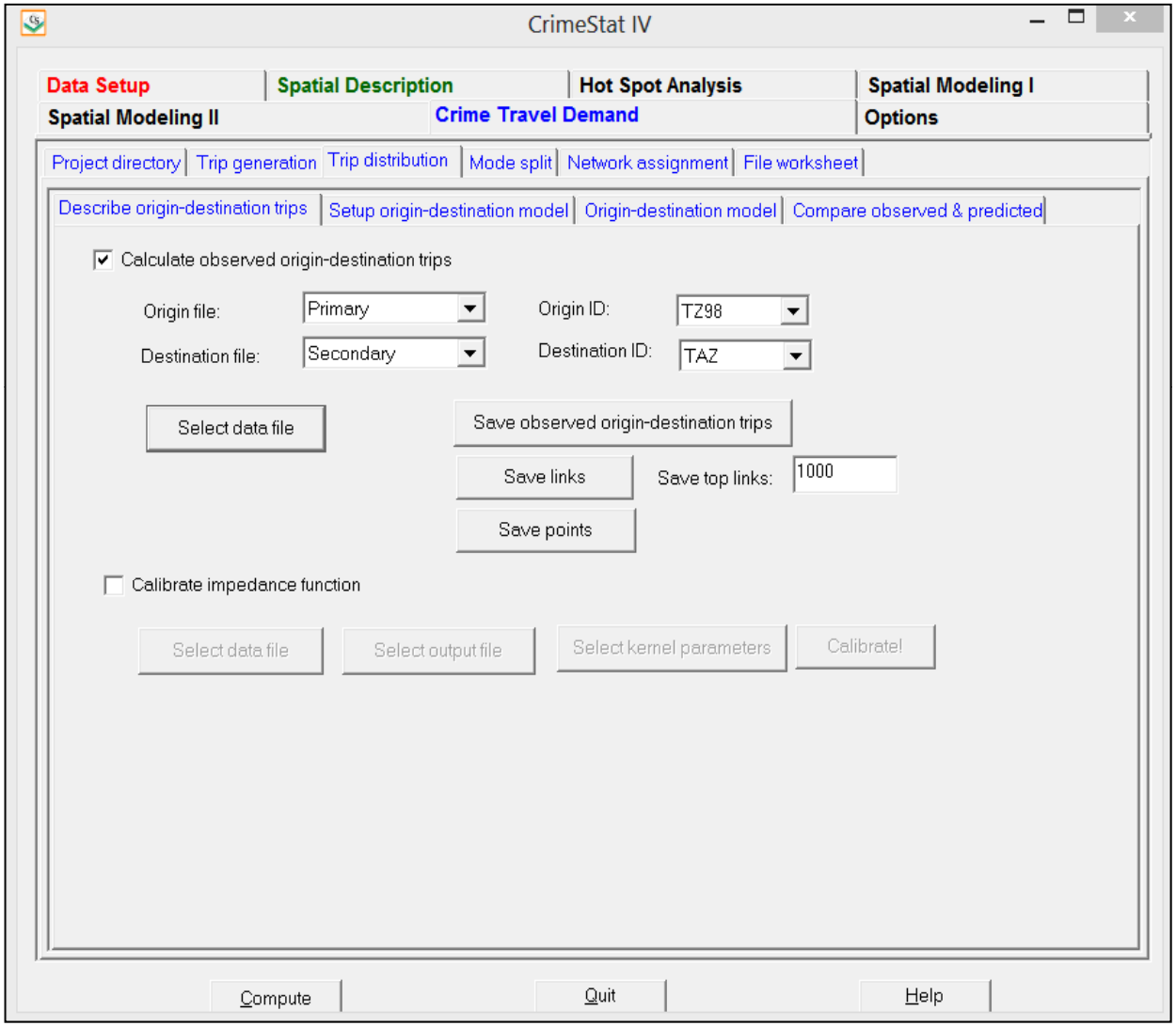
Trip Distribution

Trip distribution involves the estimation of the number of trips that travel from each origin zone (including the 'external' zone) to each destination zone. The estimation is based on a gravity-type model. The determining variables are the number of predicted origins, the number of predicted destinations, the impedance (or cost) of travel between the origin zone, coefficients for the origins and destinations, and exponents of the origins and destinations.

The user inputs the number of predicted origins and predicted destinations and specifies an impedance model (which can be mathematical or calibrated from an existing data set). In addition,

Figure 2.24:

Trip Distribution Modeling



the user specifies exponents for the origin and destination values. The model iteratively estimates the coefficients. In addition, the routine can calculate the actual (observed) trip distribution with an existing data set that lists individual origin and destination locations. Finally, a comparison between the observed distribution and that predicted by the model can be made.

Describe Origin-Destination trips

An empirical description of the actual trip distribution matrix can be made if there is a data set that includes individual origin and destination locations. The user defines the origin location and the destination location for each record and a set of zones from which to compare the individual origins and destinations. The routine matches up each origin location with the nearest zone, each destination location with the nearest zone, and calculates the number of trips from each origin zone to each destination zone. This is an *observed* distribution of trips by zone.

Calculate Observed Origin-Destination trips

Check if an empirical origin-destination trip distribution is to be calculated.

Origin file

The origin file is a list of origin zones with a single point representing the zone (e.g., the centroid). There can be more origin zones than destination zones, but **all** destination zones must be included among the origin zone list. The origin file must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Origin ID

Specify the origin ID variable in the data file (e.g., CensusTract, Block, TAZ). **Note**: all destination ID's should be in the origin zone file and must have the same names.

Destination file

The destination file is a list of destination zones with a single point representing the zone (e.g., the centroid). It must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Destination ID

Specify the destination ID variable in the data file (e.g., CensusTract, Block, TAZ). Note: the ID's used for the destination file zones must be the same as in the origin file.

Select data file

The data set must have individual origin and destination locations. Each record must have the X/Y coordinates of an origin location and the X/Y coordinates of a destination location. For example, an arrest file might list individual incidents with each incident having a crime location (the destination) and a residence or arrest location (the origin). Select the file that has the X and Y coordinates for the origin and destination locations. *CrimeStat* reads dbase 'dbf'', ArcGIS 'shp' and ASCII text files. Select the tab and specify the type of file to be selected. Use the browse button to search for the file. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns.

Variables

Define the file which contains the X and Y coordinates for both the origin (residence) and destination (crime) locations.

Columns

Select the variables for the X and Y coordinates respectively for *both* the origin and destination locations (e.g., Lon, Lat, HomeX, HomeY, IncidentX, IncidentY.) Both locations must be defined for the routine to work.

Missing values

Identify whether there are missing values for these four fields (X and Y coordinates for both origin and destination locations). By default, *CrimeStat* will ignore records with blank values in any eligible field or records with non-numeric values (e.g., alphanumeric characters, , *). Blanks will always be excluded unless the user selects **<*none*>**. There are 8 possible options:

1. <*blank*> fields are automatically excluded. This is the default
2. <*none*> indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0
3. *0* is excluded

4. *–1* is excluded
5. *0 and –1* indicates that both 0 and -1 will be excluded
6. *0, -1 and 9999* indicates that all three values (0, -1, 9999) will be excluded
7. *Any* other numerical value can be treated as a missing value by typing it (e.g., 99)
8. *Multiple* numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99).

Type of coordinate system and data units

The coordinate system and data units are listed for information. If the coordinates are in longitudes and latitudes, then a spherical system is being used and data units will automatically be decimal degrees. If the coordinate system is projected (e.g., State Plane, Universal Transverse Mercator – UTM), then data units could be either in feet (e.g., State Plane) or meters (e.g., UTM.)

Table output

The entire origin-destination matrix is output as a table to the screen including summary file information and:

1. The origin zone (ORIGIN)
2. The destination zone (DEST)
3. The number of observed trips (FREQ)

Save observed origin-destination trips

If specified, the full origin-destination output is saved as a 'dbf' file named by the user.

File output

The file output includes:

1. The origin zone (ORIGIN)
2. The destination zone (DEST)
3. The X coordinate for the origin zone (ORIGINX)
4. The Y coordinate for the origin zone (ORIGINY)
5. The X coordinate for the destination zone (DESTX)
6. The Y coordinate for the destination zone (DESTY)
7. The number of trips (FREQ)

Note: each record is a unique origin-destination combination and there are M x N records where M is the number of origin zones (including the external zone) and N is the number of destination zones.

Save links

The top observed origin-destination trip links can be saved as separate **line** objects for use in a GIS. Specify the output file format (*ArcGIS* 'shp', *MapInfo* 'mif' or ASCII) and the file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Save top links

Because the output file is very large (number of origin zones x number of destination zones), the user can select a sub-set of zone combinations with the most observed trips. Indicating the top K links will narrow the number down to the most important ones. The default is the top 100 origin-destination combinations. Each output object is a line from the origin zone to the destination zone with an ODT prefix. The prefix is placed before the output file name. The line graphical output for each object includes:

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (ODT)
3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of observed trips for that combination (FREQ)
10. The distance between the origin zone and the destination zone.

Save points

Intra-zonal trips (trips in which the origin and destination are the same zone) can be output as separate **point** objects as an *ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in

the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Again, the top *K* points are output (default=100). Each output object is a point representing an intra-zonal trip with an ODTPOINTS prefix. The prefix is placed before the output file name. The point graphical output for each object includes:

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (POINTSODT)
3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of observed trips for that combination (FREQ)

Calibrate Impedance Function

This function allows the calibration of an approximate travel impedance function based on actual trip distributions. It is used to describe the travel distance of an actual sample (the calibration sample). A file is input which has a set of incidents (records) that includes both the X and Y coordinates for the location of the offender's residence (origin) and the X and Y coordinates for the location of the incident that the offender committed (destination.) The routine estimates a travel distance function using a one-dimensional kernel density method. For each record, the distance between the origin location and the destination location is calculated and is represented on a distance scale. The maximum distance is calculated and divided into a number of intervals; the default is 100 equal sized intervals, but the user can modify this. For each distance (point) calculated, a one-dimensional kernel is overlaid. For each distance interval, the values of all kernels are summed to produce a smooth function of travel impedance. The results are saved to a file that can be used origin-destination model. Note, however, that this is an empirical distribution and represents the combination of origins, destinations, and costs. It is not necessarily a good description of the impedance (cost) function by itself. Many of the mathematical functions produce a better fit than the empirical impedance function.

Select data file for calibration

Select the file that has the X and Y coordinates for the origin and destination locations. *CrimeStat* reads dbase 'dbf', ArcGIS ‘shp’ and ASCII files. Select the tab and select the type of

file to be selected. Use the browse button to search for the file. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns.

Variables

Define the file which contains the X and Y coordinates for both the origin (residence) and destination (crime) locations

Columns

Select the variables for the X and Y coordinates respectively for *both* the origin and destination locations (e.g., Lon, Lat, HomeX, HomeY, IncidentX, IncidentY.) Both locations must be defined for the routine to work.

Missing values

Identify whether there are any missing values for these four fields (X and Y coordinates for both origin and destination locations). By default, *CrimeStat* will ignore records with blank values in any of the eligible fields or records with non-numeric values (e.g., alphanumeric characters, , *). Blanks will always be excluded unless the user selects ***<none>***. There are 8 possible options:

1. <*blank*> fields are automatically excluded. This is the default
2. <*none*> indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0
3. *0* is excluded
4. *–1* is excluded
5. *0 and –1* indicates that both 0 and -1 will be excluded
6. *0, -1 and 9999* indicates that all three values (0, -1, 9999) will be excluded
7. *Any* other numerical value can be treated as a missing value by typing it (e.g., 99)
8. *Multiple* numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99)

Type of coordinate system and data units

Select the type of coordinate system. If the coordinates are in longitudes and latitudes, then a spherical system is being used and data units will automatically be decimal degrees. If the coordinate system is projected (e.g., State Plane, Universal Transverse Mercator – UTM), then

data units could be either in feet (e.g., State Plane) or meters (e.g., UTM.) Directional coordinates are not allowed for this routine.

Select kernel parameters

There are five parameters that must be defined.

Method of interpolation

There are five types of kernel distributions that can be used to estimate point density:

1. The **normal** kernel overlays a three-dimensional normal distribution over each point that then extends over the area defined by the reference file. This is the default kernel function.

2. The **uniform** kernel overlays a uniform function over each point that only extends for a limited distance.

3. The **quartic** kernel overlays a quartic function over each point that only extends for a limited distance.

4. The **triangular** kernel overlays a three-dimensional triangle over each point that only extends for a limited distance.
5. The **negative exponential** kernel overlays a three dimensional negative exponential function over each point that only extends for a limited distance

The methods produce similar results though the normal is generally smoother for any given bandwidth.

Choice of bandwidth

The kernels are applied to a limited search distance, called 'bandwidth'. For the normal kernel, bandwidth is the standard deviation of the normal distribution. For the uniform, quartic, triangular and negative exponential kernels, bandwidth is the radius of a circle defined by the surface. For all types, larger bandwidth will produce smoother density estimates and both adaptive and fixed bandwidth intervals can be selected.

Fixed bandwidth

A fixed bandwidth distance is a fixed interval for each point. The user must define the interval, the interval size, and the distance units by which it is calculated (miles, nautical miles, feet, kilometers, or meters). The default bandwidth setting is fixed with intervals of 0.25 miles each. The interval size can be changed.

Adaptive bandwidth

An adaptive bandwidth distance is identified by the minimum number of other points found within a symmetrical band drawn around a single point. A symmetrical band is placed over each distance point, in turn, and the width is increased until the minimum sample size is reached. Thus, each point has a different bandwidth size. The user can modify the minimum sample size. The default for the adaptive bandwidth is 100 points.

Specify interpolation bins

The interpolation bins are defined in one of two ways:

1. By the number of bins. The maximum distance calculated is divided by the number of specified bins. The default is 100 bins. The user can change the number of bins.

2. By the distance between bins. The user can specify a bin width in miles, nautical miles, feet, kilometers, and meters.

Output (areal) units

Specify the areal density units as points per mile, nautical mile, foot, kilometer, or meter. The default is points per mile.

Calculate densities or probabilities

The density estimate for each cell can be calculated in one of three ways:

1. **Absolute densities.** This is the number of points per grid cell and is scaled so that the sum of all grid cells equals the sample size.
2. **Relative densities**. For each grid cell, this is the absolute density divided by the grid cell area and is expressed in the areal output units (e.g., points per square mile)

3. **Probabilities**. This is the proportion of all incidents that occur in the grid cell. The sum of all grid cells equals a probability of 1. Unlike the Jtc calibration routine, this is the default. In most cases, a user would want a proportional (probability) distribution as the relative differences in impedance for different costs are what is of interest.

Select whether absolute densities, relative densities, or probabilities are to be output for each cell. The default is probabilities.

Select output file

The output *must* be saved to a file. *CrimeStat* can save the calibration output to either a dbase 'dbf' or ASCII text 'txt' file.

Calibrate!

Click on 'Calibrate!' to run the routine. The output is saved to the specified file upon clicking on 'Close'.

Graphing the travel impedance function

Click on 'View graph' to see the travel impedance function. The screen view can be printed by clicking on 'Print'. For a better quality graph, however, the output should be imported into a graphics package.

Setup Origin-Destination Model

The page is for the setup of the origin-destination model. All the relevant files, models and exponents are input on the page.

Predicted origin file

The predicted origin file is a file that lists the origin zones with a single point representing the zone (e.g., the centroid) and also includes the predicted number of crimes by origin zone. The file must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Origin variable

Specify the name of the variable for the predicted origins (e.g., PREDICTED, ADJORIGINS).

Origin ID

Specify the origin ID variable in the data file (e.g., CensusTract, Block, TAZ). Note: all destination IDs should be in the origin zone file and must have the same names.

Predicted destination file

The predicted destination file is a list of destination zones with a single point representing the zone (e.g., the centroid) and also includes the predicted number of crimes by destination zone. It must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Destination variable

Specify the name of the variable for the predicted destination (e.g., PREDICTED, ADJDEST).

Destination ID

Specify the destination ID variable in the data file (e.g., CensusTract, Block, TAZ). Note: the ID's used for the destination file zones must be the same as in the origin file.

Exponents

The exponents are power terms for the predicted origins and destinations and indicate the relative strength of those variables. For example, compared to an exponent of 1.0 (the default), an exponent greater than 1.0 will strengthen that variable (origins or destinations) while an exponent less than 1.0 will weaken that variable. They can be considered 'fine tuning' adjustments.

Origins

Specify the exponent for the predicted origins. The default is 1.0.

Destinations

Specify the exponent for the predicted origins. The default is 1.0.

Impedance function

The trip distribution routine can use two different travel distance functions: 1) An already-calibrated distance function; and 2) A mathematical formula. The default is a mathematical formula.

Use an already-calibrated distance function

If a travel distance function has already been calibrated (see 'Calibrate impedance function' under trip distribution), the file can be directly input into the routine. The user selects the name of the already-calibrated travel distance function. *CrimeStat* reads dbase 'dbf', ArcGIS 'shp' and ASCII files.

Use a mathematical formula

A mathematical formula can be used instead of a calibrated distance function. To do this, it is necessary to specify the type of distribution. There are five mathematical models that can be selected:

1. Negative exponential
2. Normal
3. Lognormal
4. Linear
5. Truncated negative exponential

The lognormal is the default. For each mathematical model, two or three different parameters must be defined:

1. For the negative exponential, the coefficient and exponent
2. For the normal distribution, the mean distance, standard deviation and coefficient
3. For lognormal distribution, the mean distance, standard deviation and coefficient
4. For the linear distribution, an intercept and slope
5. For the truncated negative exponential, a peak distance, peak likelihood, intercept, and exponent.

Measurement unit

The routine can calculate impedance in four ways, by:

1. Distance (miles, nautical miles, feet, kilometers, or meters)
2. Travel time (minutes, hours)
3. Speed (miles per hour, kilometers per hour)
4. General travel costs (unspecified units).

These must be setup under Network distance on the Measurement Parameters page. Specify the appropriate units. In the Network Parameters dialogue, specify the measurement units. The default is distance in miles.

Assumed impedance for external zones

For trips originating outside the study area (external trips), specify the amount and the units that will be assumed for these trips. The default is 25 miles.

Assumed impedance for intra-zonal trips

For trips originating and ending in the same zone (intra-zonal trips), specify the amount and the units that will be assumed for these trips. The default is 0.25 miles.

Minimum number of trips per cell

The parameter allows a minimum number of predicted trips for each origin-destination combination (cell). It will return a zero (0) if the predicted number is less than the minimum. This can be adjusted to avoid many cells with very small numbers of predicted trips. Care must be taken, though, as this can alter the overall distribution. The default minimum is 0.05 trips per cell.

Model constraints

In calibrating a model, the routine must constrain either the origins or the destinations (single constraint) or constrain both the origins and the destinations (double constraint). In the latter case, it is an iterative solution. The default is to constrain destinations as it is assumed that the destinations totals (the number of crimes occurring in each zone) are probably more correct than the number of crimes originating in each zone. . Specify the type of constraint for the model.

Constrain origins

If constrain origins is selected, the total number of trips from each origin zone will be held constant.

Constrain destinations

If constrain destinations is selected, the total number of trips from each destination zone will be held constant.

Constrain both origins and destinations

If constrain both origins and destinations is selected, the routine iteratively works out a balance between the number of origins and the number of destinations.

Origin-Destination Model

The trip distribution (origin-destination) model is implemented in two steps. First, the coefficients are calculated according to the exponents and impedance functions specified on the setup page. Second, the coefficients and exponents are applied to the predicted origins and destinations resulting in a predicted trip distribution. Because these two steps are iterative, they cannot be run simultaneously.

Calibrate origin-destination model

Check the 'Calibrate origin-destination model' box to run the calibration model.

Save modeled coefficients (parameters)

The modeled coefficients are saved as a 'dbf' file. Specify a file name.

Apply predicted origin-destination model

Check the 'Apply predicted origin-destination model' box to run the trip distribution prediction.

Modeled coefficients file

Load the modeled coefficients file saved in the 'Calibrate origin-destination model' stage.

Assumed coordinates for external zone

In order to model trips from the 'external' zone (trips from outside the study area), specify coordinates for this zone. These coordinates will be used in drawing lines from the predicted origins to the predicted destinations. There are four choices:

1. Mean center (the mean X and mean Y of all origin file points are taken). This is the default.
2. Lower-left corner (the minimum X and minimum Y values of all origin file points are taken).
3. Upper-right corner (the maximum X and maximum Y values of all origin file points are taken).
4. Use coordinates (user-defined coordinates). Indicate the X and Y coordinates that are to be used.

Table output

The table output includes summary file information and (with default names):

1. The origin zone (ORIGIN)
2. The destination zone (DEST)
3. The number of predicted trips (PREDTRIPS)

Save predicted origin-destination trips

Define the output file. The output is saved as a 'dbf' file with the file name specified by the user.

File output

The file output includes (with default names):

1. The origin zone (ORIGIN)
2. The destination zone (DEST)
3. The X coordinate for the origin zone (ORIGINX)
4. The Y coordinate for the origin zone (ORIGINY)
5. The X coordinate for the destination zone (DESTX)
6. The Y coordinate for the destination zone (DESTY)
7. The number of predicted trips (PREDTRIPS)

Note: each record is a unique origin-destination combination and there are M x N records where M is the number of origin zones (including the external zone) and N is the number of destination zones.

Save links

The top predicted origin-destination trip links can be saved as separate **line** objects for use in a GIS. Specify the output file format (*ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats) and the file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Save top links

Because the output file is very large (number of origin zones x number of destination zones), the user can select a sub-set of zone combinations with the most predicted trips. Indicating the top K links will narrow the number down to the most important ones. The default is the top 100 origin-destination combinations. Each output object is a line from the origin zone to the destination zone with an ODT prefix. The prefix is placed before the output file name. The graphical output includes (with default names):

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (ODT)
3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of predicted trips for that combination (PREDTRIPS)
10. The distance between the origin zone and the destination zone.

Save points

Intra-zonal trips (trips in which the origin and destination are the same zone) can be output as separate **point** objects as an *ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in

the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Again, the top K points are output (default=100). Each output object is a point representing an intra-zonal trip with an ODTPOINTS prefix. The prefix is placed before the output file name. The graphical output for each includes (with default names):

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (POINTSODT)
3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of predicted trips for that combination (PREDTRIPS)

Compare Observed & Predicted Origin-Destination Trip Lengths

The predicted trip distribution model can be compared with the observed (actual) trip distribution. Since there are many cells for this comparison (M origins x N destinations), a comparison is usually conducted for the trip length distributions. Each origin-destination link (whether the observed distribution or that predicted by the model) is converted into a trip length. The maximum distance between an origin and a destination is then divided into K bins (intervals), where K can be defined by the user; the default is 25. The two distributions are compared with two statistics: 1) the coincidence ratio (essentially a positive correlation index that varies between 0 and 1 with 0 representing little coincidence and 1 representing perfect coincidence) and 2) the Komolgorov-Smirnov two-sample test (a test of the difference between the cumulative proportions of the observed and predicted distributions). There is also a graph that compares the two distributions.

Observed trip file

Select the observed trip distribution file by clicking on the Browse button.

Observed number of origin-destination trips

Specify the variable for the observed number of trips. The default name is FREQ.

Orig_ID

Specify the ID name for the origin zone. The default name is ORIGIN. Note: the origin ID's should be the same as in the predicted file in order to compare the top links.

Orig_X

Specify the name for the X coordinate of the origin zone. The default name is ORIGINX.

Orig_Y

Specify the name for the Y coordinate of the origin zone. The default name is ORIGINY.

Dest_ID

Specify the ID name for the destination zone. The default name is DEST. Note: the destination ID's should be the same as in the predicted file in order to compare the top links.

Dest_X

Specify the name for the X coordinate of the destination zone. The default name is DESTX.

Dest_Y

Specify the name for the Y coordinate of the destination zone. The default name is DESTY.

Predicted trip file

Select the predicted trip distribution file by clicking on the Browse button and finding the file.

Predicted number of origin-destination trips

Specify the variable for the predicted number of trips. The default name is PREDTRIPS

Orig_ID

Specify the ID name for the origin zone. The default name is ORIGIN. Note: the origin ID's should be the same as in the observed file in order to compare the top links.

Orig_X

Specify the name for the X coordinate of the origin zone. The default name is ORIGINX.

Orig_Y

Specify the name for the Y coordinate of the origin zone. The default name is ORIGINY.

Dest_ID

Specify the ID name for the destination zone. The default name is DEST. Note: the destination ID's should be the same as in the observed file in order to compare the top links.

Dest_X

Specify the name for the X coordinate of the destination zone. The default name is DESTX.

Dest_Y

Specify the name for the Y coordinate of the destination zone. The default name is DESTY.

Select bins

Specify how the bins (intervals) will be defined. There are two choices. One is to select a fixed number of bins. The other is to select a constant interval.

Fixed number

This sets a fixed number of bins. An interval is defined by the maximum distance between zone divided by the number of bins. The default number of bins is 25. Specify the number of bins.

Constant interval

This defines an interval of a specific size. If selected, the units must also be chosen. The default is 0.25 miles. Other distance units are nautical miles, feet, kilometers, and meters. Specify the interval size.

Save comparison

The output is saved as a 'dbf' file with the file name specified by the user.

Table output

The table output includes summary information and:

1. The number of trips in the observed origin-destination file
2. The number of trips in the predicted origin-destination file
3. The number of intra-zonal trips in the observed origin-destination file
4. The number of intra-zonal trips in the predicted origin-destination file
5. The number of inter-zonal trips in the observed origin-destination file
6. The number of inter-zonal trips in the predicted origin-destination file
7. The average observed trip length
8. The average predicted trip length
9. The median observed trip length
10. The median predicted trip length
11. The Coincidence Ratio (an indicator of congruence varying from 0 to 1)
12. The D value for the Komolgorov-Smirnov two-sample test
13. The critical D value for the Komolgorov-Smirnov two-sample test
14. The p-value associated with the D value of Komolgorov-Smirnov two-sample test relative to the critical D value.

and for each bin:

15. The bin number
16. The bin distance
17. The observed proportion
18. The predicted proportion

File output

The saved file includes (with default names):

1. The bin number (BIN)
2. The bin distance (BINDIST)
3. The observed proportion (OBSERVPROP)
4. The predicted proportion (PREDPROP)

Graph of observed and predicted trip lengths

While the output page is open, clicking on the graph button will display a graph of the observed and predicted trip length proportions on the Y-axis by the trip length distance on the X-axis.

Compare Top Links

As an alternative to a comparison of trip lengths for the observed and predicted distributions, the top links can be compared with a pseudo-Chi square test. Since the top links have the most trips, the Chi square distribution can be used for comparison. However, because the rest of the distribution is not being used, significance tests are invalid.

The statistic compares the number of trips for the top links in the observed distribution with the number of trips for the same links in the predicted model. The routine calulates a Chi square value.

The statistic is useful for comparing different models. The *lower* the Chi square value, the better the fit between the predicted model and the observed for the top links. The aim is to find the model that gives the lowest possible Chi square value.

Note: in order to use this routine, the origin and destination ID's ***must*** be the same for both the observed and predicted trip files.

Click the box and specify the number of links to be compared. The default value is 100. The output includes:

1. The number of links that are compared

and for each trip pair in order of the number of trips:

2. The zone ID of the origin zone (FromZone)
3. The zone ID of the destination zone (ToZone)
4. The observed (actual) number of trips
5. The predicted number of trips.

At the bottom of the page is a Chi-square test of the difference between the observed and predicted number of trips for the top links. Since not all trips have been included in this distribution, no significance test is conducted. The aim should be to find the model with the lowest Chi-square value.

Optimizing the Fit Between the Observed and Predicted Links

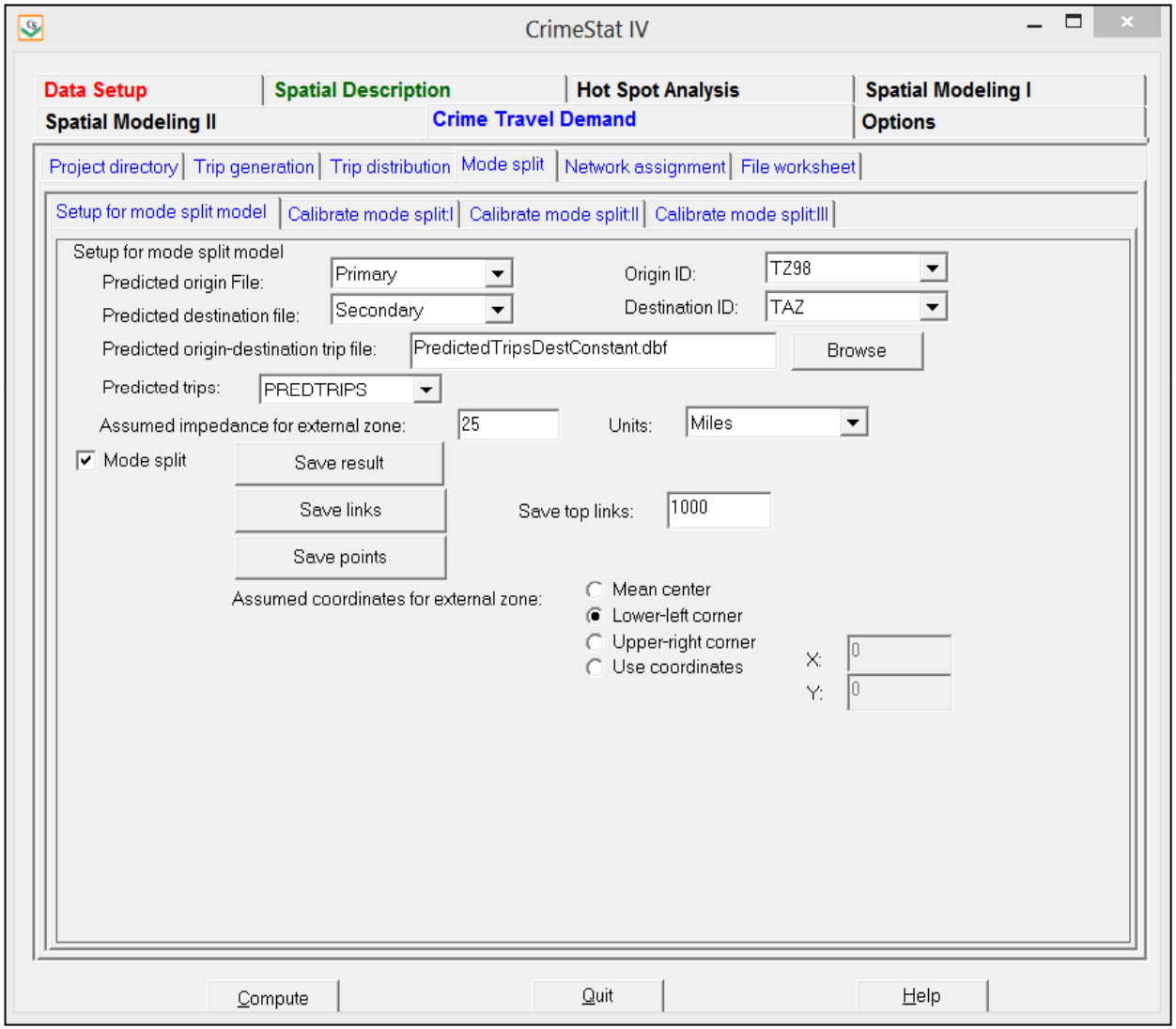
Ideally, the best model would fulfill three comparison tests. First, the number of intra-zonal tests (and, by implication, the number of inter-zonal trips) in the predicted trip distribution would be identical to the number of intra-zonal trips in the observed distribution. Second, the overall model would have a high coincidence ratio and a non-significant Komolgorov-Smirnov test for the trip length comparison. Third, the Chi square value for the top links would be the lowest possible. In practice, an optimal model may have to balance these three criteria, producing a good match in the number of intra-zonal trips, a reasonably low Chi square value for the top links, and a reasonably high coincidence ratio for the trip length comparison. There may not be a single, optimal model.

Mode Split

Mode split involves separating the predicted trips by link (i.e., the trips from any one origin zone, A, to any one destination zone, B) into distinct travel modes (e.g., walk, bicycle, drive, bus, train). The basis of the separation is an aggregate relative impedance function. This is, essentially, the 'cost' of traveling by any one mode relative to all modes, whether cost is defined in terms of distance, travel time, or generalized costs. The model can be determined by either an empirically-derived impedance function or a mathematical function. The empirically-derived impedance function would come from a calibration data set whereas the mathematical function is selected on the basis of either previous experience or other studies. The separate impedance functions can be constrained to a network in order to prevent trips from being allocated that are nearly impossible (e.g., train trips where there are no train lines and bus trips where there are no bus routes).

Figure 2.25:

Mode Split Modeling



The steps of the routine are as follows. First, the user inputs a file of predicted trips (i.e., the number of predicted trips from every origin zone to every destination zone). Second, the user defines which travel modes are to be modeled. Up to five separate modes are allowed.

Third, the user sets up an impedance model for **each** travel mode. Any of the impedance models can be constrained to a particular network (e.g., bus mode constrained to a bus network; train mode constrained to a train network). This would normally be desired even for modes where travel in any direction is possible (e.g., walk, bicycle, drive modes). Fourth, and finally, after all impedance models have been defined, the routine is run and splits the predicted trips into the defined modes on the basis of the relative impedance of each mode to all impedances.

Setup for Mode Split Model

This page defines the predicted trip file and the output file. It also allows a definition of where external trips are assumed to come from.

Predicted origin file

The predicted origin file is a file that lists the origin zones with a single point representing the zone (e.g., the centroid) and also includes the predicted number of crimes by origin zone. The file must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Origin variable

Specify the name of the variable for the predicted origins (e.g., PREDICTED, ADJORIGINS).

Origin ID

Specify the origin ID variable in the data file (e.g., CensusTract, Block, TAZ). Note: all destination ID's should be in the origin zone file and must have the same names.

Predicted destination file

The predicted destination file is a list of destination zones with a single point representing the zone (e.g., the centroid) and also includes the predicted number of crimes by destination zone. It must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Destination variable

Specify the name of the variable for the predicted destination (e.g., PREDICTED, ADJDEST).

Destination ID

Specify the destination ID variable in the data file (e.g., CensusTract, Block, TAZ). Note: the ID's used for the destination file zones must be the same as in the origin file.

Predicted origin-destination trip file

The predicted origin-destination trip file lists the predicted number of trips from every origin zone to every destination zone. On the mode split setup page, select the predicted trip file (i.e., the predicted origin-destination trip file by clicking on the 'Browse' button.

Predicted trips

Specify the variable for the predicted number of trips. The default name is PREDTRIPS

Assumed impedance for external zone

In order to model trips from the 'external zone' (trips from outside the study area), specify an impedance to be assumed. The default is 25 miles.

Assumed coordinates for external zone

In order to model trips from the 'external' zone (trips from outside the study area), specify coordinates for this zone. These coordinates will be used in drawing lines from the predicted origins to the predicted destinations. There are four choices:

1. Mean center (the mean X and mean Y of all origin file points are taken). This is the default.
2. Lower-left corner (the minimum X and minimum Y values of all origin file points are taken).
3. Upper-right corner (the maximum X and maximum Y values of all origin file points are taken).
4. Use coordinates (user-defined coordinates). Indicate the X and Y coordinates that are to be used.

Run Mode Split

Check the "Mode split" box to enable the routine. It will run when the "Compute" button is clicked.

Mode Split Output

There are three types of output for the mode split routine.

1. The zone-to-zone trip file for **each** mode separately can be output as a dbf file.
2. The most frequent inter-zonal (i.e., trips between different zones) trips for **each** mode separately can be output as polylines.
3. The most frequent intra-zonal (i.e., trips within the same zone) trips for **each** mode separately can be output as points.

Output file name (save result)

Define the output file name by clicking on 'Save result'. The output will be saved as a 'dbf' file with the file name specified by the user. For **each** mode, the prefix 'TMode' will be prefaced before the file. For example, if the name provided by the user is "robberies.dbf" and if there are three travel modes modeled, then there will be three output files (TMode1robberies.dbf; TMode2robberies.dbf; TMode3robberies.dbf).

File output

The file output includes:

1. The origin zone (ORIGIN)
2. The destination zone (DEST)
3. The X coordinate for the origin zone (ORIGINX)
4. The Y coordinate for the origin zone (ORIGINY)
5. The X coordinate for the destination zone (DESTX)
6. The Y coordinate for the destination zone (DESTY)
7. The number of predicted trips (PREDTRIPS)

Note: each record is a unique origin-destination combination and there are M x N records where M is the number of origin zones (including the external zone) and N is the number of destination zones.

Save links

The top predicted origin-destination trip links can be saved as separate **line** objects for use in a GIS. Specify the output file format (*ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats) and the file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

For **each** mode, the prefix 'TripMode' will be prefaced before the file. For example, if the name provided by the user is "robberies" and if there are three travel modes modeled, then there will be three graphical output files (TripMode1robberies.shp/mif; TripMode2robberies.shp/mif; TripMode3robberies.shp/mif).

Save top links

Because the output file is very large (number of origin zones x number of destination zones), the user can select a sub-set of zone combinations with the most predicted trips. Indicating the top K links will narrow the number down to the most important ones. The default is the top 100 origin-destination combinations. Each output object is a line from the origin zone to the destination zone with a TripMode prefix where '' is the mode number. The prefix is placed before the output file name. The graphical output includes:

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (ODT)
3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of predicted trips for that combination (PREDTRIPS)
10. The distance between the origin zone and the destination zone.

Save points

Intra-zonal trips (trips in which the origin and destination are the same zone) can be output as separate **point** objects as an *ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the

projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Again, the top K points are output (default=100). Each output object is a point representing an intra-zonal trip with a TripModePoints prefix where ‘’ is the mode number. The prefix is placed before the output file name. For example, if the name provided by the user is “robberies” and if there are three travel modes modeled, then there will be three graphical output files (TripModePoints1robberies.shp/mif; TripModePoints2robberies.shp/mif; TripModePoints3robberies.shp/mif).

The graphical output for each includes:

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (POINTSODT)
3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of predicted trips for that combination (PREDTRIPS)

Calibrate Mode Split: I-III

For each mode (up to five), the impedance parameters have to be set. There are three pages for this:

1. “Calibrate mode split: I” covers modes 1 and 2.
2. “Calibrate mode split: II’ covers modes 3 and 4.
3. “Calibrate mode split: III” covers mode 5.

For each mode, the user should indicate whether the mode is to be used, the name to be used for the mode, whether a default impedance will be calculated directly or if it should be constrained to a network, and the specific impedance model used. If any mode is not used, then it will not be part of the calculations. Use only those modes that are relevant, but, also, be sure not to leave out any important ones.

The following instructions apply to each of the five modes.

Mode

Check the box if the mode is to be used.

Label

Put in a label for the mode. Default names are provided (walk, bicycle, drive, bus, train), but the user is not required to use those.

Impedance constraint

The impedance will be calculated either directly or is constrained to a network. The default impedance is defined with the type of distance measurement specified on the Measurement Parameters page (under Data setup). On the other hand, if the impedance is to be constrained to a network, then the network has to be defined.

Default

The default impedance is that specified on the Measurement parameters page. If direct distance is the default distance (on the measurement parameters page), then all impedances are calculated as a direct distance. If indirect distance is the default, then all impedances are calculated as indirect (Manhattan) distance. If network distance is the default, then all impedances are calculated using the specified network and its parameters; travel impedance will automatically be constrained to the network under this condition.

Constrain to network

An impedance calculation should be constrained to a network where there are limited choices. For example, a bus trip requires a bus route; if a particular zone is not near an existing bus route, then a direct distance calculation will be misleading since it will probably underestimate true distance. Similarly, for a train trip, there needs to be an existing train route. Even for walking, bicycling and driving trips, an existing network might produce a more realistic travel impedance than simply assuming a direct travel path. If the impedance calculation is to be constrained to a network, then the network must be defined.

Check the 'Constrain to network' box and click on the 'Parameters' button. The network file can be either a shape **line** or **polyline** file (the default) or another file, either dBase IV 'dbf'

or ASCII. If the file is a shape file, the routine will know the locations of the nodes. All the user needs to do is identify a weighting variable, if used.

For a dBase IV or other file, the X and Y coordinate variables of the end nodes must be defined. These are called the "From" node and the "End" node, though there is no particular order. An optional weight variable is allowed for both a shape or dbf file. The routine identifies nodes and segments and finds the shortest path. By default, the shortest path is in terms of distance though each segment can be weighted by travel time, travel speed, or generalized cost; in the latter case, the units are minutes, hours, or unspecified cost units.

Note: using network distance for distance calculations can be a very slow process (i.e., taking many hours or even up to several days for calculating a large matrix).

Minimum absolute impedance

If the mode is constrained to a network, an additional constraint is needed to ensure realistic allocations of trips. This is the minimum absolute impedance between zones. The default is 2 miles. For any zone pair (an origin zone and a destination zone) that is closer together (in distance, time interval, or cost) than the minimum specified, no trips will be allocated to that mode. This constraint is to prevent unrealistic trips being assigned to intra-zonal trips or trips between nearby zones. *CrimeStat* uses three impedances for a constrained network: 1) the impedance from the origin zone to the nearest node on the network (e.g., nearest rail station); b) the impedance along the network to the node nearest to the destination; and c:) the impedance from that node to the destination zone. Since most impedance functions for a mode constrained to a network will have the highest likelihood some distance from the origin, it's possible that the mode would be assigned to, essentially, very short trips (e.g., the distance from an origin zone to a rail network and then back again might be modeled as a high likelihood of a train trip even though such a trip is very unlikely).

For each mode that is constrained to a network, specify the minimum absolute impedance. The units will be the same as that specified by the measurement units. The default is 2 miles. If the units are distance, then trips will only be allocated to those zone pairs that are equal to or greater in distance than the minimum specified. If the units are travel time or speed, then trips will only be allocated to those zone pairs that are farther apart than the distance that would be traveled in that time at 30 miles per hour. If the units are cost, then the routine calculates the average cost per mile along the network and only allocates trips to those zone pairs that are farther apart than the distance that would be traveled at that average cost.

Impedance function

The model split routine can use two different travel distance functions: 1) An already-calibrated distance function; and 2) A mathematical formula. The default is a mathematical formula.

Use an already-calibrated distance function

If a travel distance function for the specific mode has already been calibrated (see 'Calibrate impedance function' under trip distribution), the file can be directly input into the routine. That routine can be used to calibrate a function if there are data on origins and destinations for individual travel modes.

The user selects the name of the already-calibrated travel distance function. *CrimeStat* reads dbase 'dbf', ArcGIS 'shp', and ASCII files.

Use a mathematical formula

A mathematical formula can be used instead of a calibrated distance function. To do this, it is necessary to specify the type of distribution. There are five mathematical models that can be selected:

1. Negative exponential – the default
2. Normal distribution
3. Lognormal distribution
4. Linear distribution
5. Truncated negative exponential

For each mathematical model, two or three different parameters must be defined:

1. For the negative exponential, the coefficient and exponent. This is the default and default values are provided.
2. For the normal distribution, the mean distance, standard deviation and coefficient.
3. For lognormal distribution, the mean distance, standard deviation and coefficient.
4. For the linear distribution, an intercept and slope.
5. For the truncated negative exponential, a peak distance, peak likelihood, intercept, and exponent.

Segment measurement unit

The routine can calculate impedance in four ways, by:

1. Distance (miles, nautical miles, feet, kilometers, or meters)
2. Travel time (minutes, hours)
3. Speed (miles per hour, kilometers per hour)
4. General travel costs (unspecified units).

Specify the appropriate units. The default is distance in miles.

Network Assignment

Network assignment involves assigning predicted trips (either all trips or by separate travel modes) to a particular route on a network. That is, for every origin-destination trip link, a particular route is found along a network (roadway, transit). The routine does this using a shortest path algorithm. The user must provide the network with its parameters. The routine allows the definition of one-way streets in order to produce a more realistic representation. In the current version, the assignment routine works on one predicted trip file at a time.

Predicted Origin-Destination file

The predicted origin-destination trip file is a file that lists the predicted number of trips from every origin zone to every destination zone. Select the predicted trip file (i.e., the predicted origin-destination trip file) by clicking on the 'Browse' button.

Origin ID

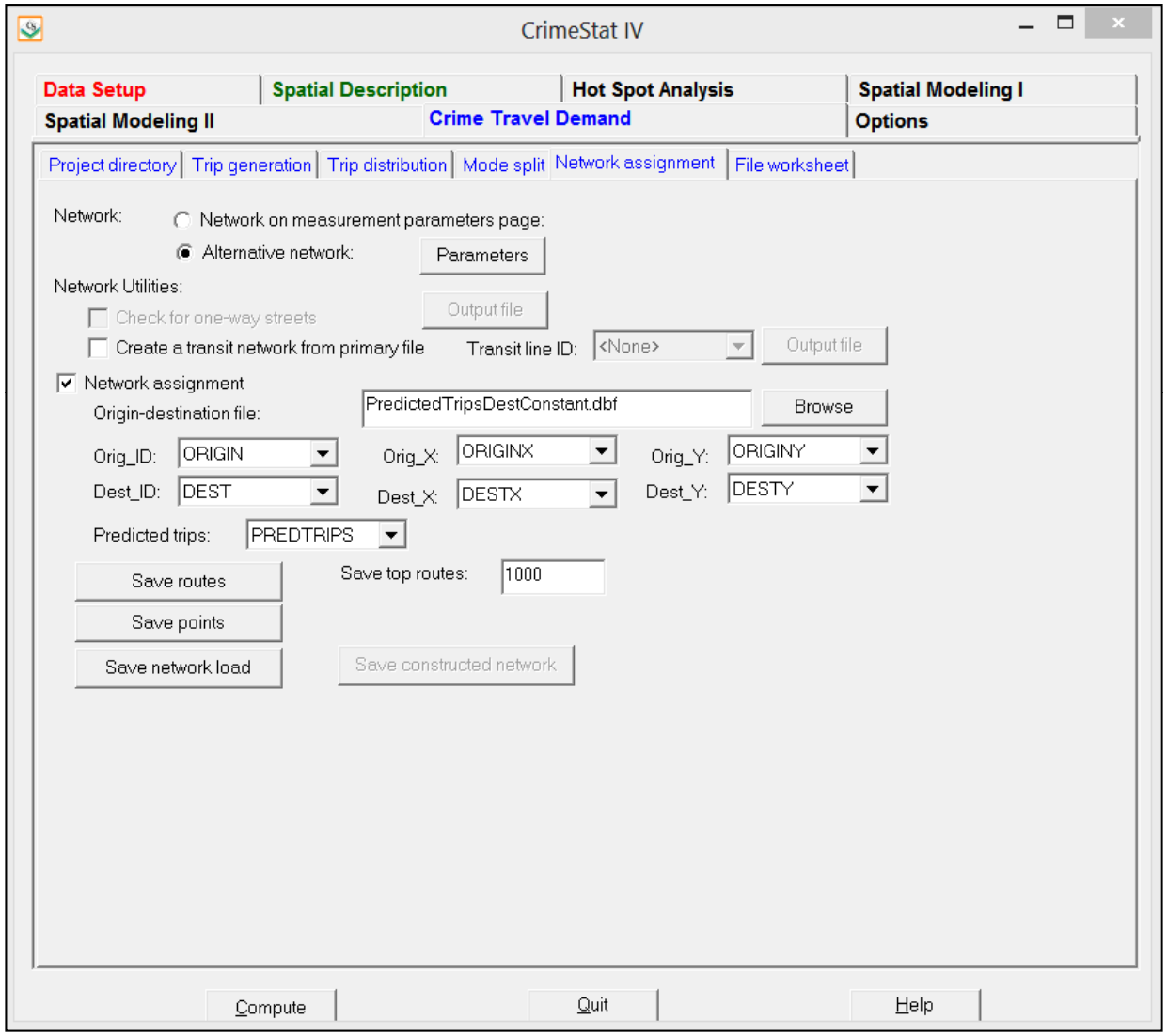
Specify the origin zone ID variable in the data file. The default name is ORIGIN.

Origin_X

Specify the name of the variable for the X coordinate of the origin zone. The default name is ORIGINX.

Figure 2.26:

Network Assignment Modeling



Origin_Y

Specify the name of the variable for the Y coordinate of the origin zone. The default name is ORIGINY.

Destination ID

Specify the destination zone ID variable in the data file. The default name is DEST.

Destination_X

Specify the name of the variable for the X coordinate of the destination zone. The default name is DESTX.

Destination_Y

Specify the name of the variable for the Y coordinate of the destination zone. The default name is DESTY.

Predicted trips

Specify the variable for the predicted number of trips. The default name is PREDTRIPS

Network Used

The network assignment routine requires a network from which the shortest path from every origin zone to every destination zone can be computed. To run this routine, check the 'Network assignment' box at the top of the page.

The user must specify the network that is to be used. There are two choices.

1. If a network was defined on the Measurement parameters page (Data setup), that network can be used to calculate the shortest path.
2. Whether a network has been defined on the Measurement parameters page or not, an alternative network can be selected. This will take priority if a network has been defined on both pages.

Network on m*easurement parameters* page

Check the 'Network on Measurement parameters page' box to use that network. All the parameters will have been defined for that setup (see Measurement parameters page).

Alternative network

If an alternative network is to be used, it must be defined. Check the 'Alternative network' box and click on the 'Parameters' button.

Note: if a network is also used on the Measurement Parameters page, then it must be defined there as well. CrimeStat will check whether that file exists; if it does not, the routine will stop and an error message will be issued. Therefore, if an alternative network is used, the user should probably change the distance measurement on the Measurement Parameters page to direct or indirect distance.

Type of network

Network files can *bi-directiona*l (e.g., a TIGER file) or *single directional* (e.g., a transportation modeling file). In a bi-directional file, travel can be in either direction. In a single directional file, travel is only in one direction. Specify the type of network to be used.

Network input file

The network file can either be a shape file (line, polyline, or polylineZ file) or another file, either dBase IV 'dbf' or ASCII. The default is a shape file. If the file is a shape file, the routine will know the locations of the nodes. For a dBase IV or other file, the X and Y coordinate variables of the end nodes must be defined. These are called the "From" node and the "End" node. An optional weight variable is allowed for both a shape or dbf file. The routine identifies nodes and segments and finds the shortest path. If there are one-way streets in a bi-directional file, the flag fields for the "From" and "To" nodes should be defined.

Network weight field

Normally, each segment in the network is not weighted. In this case, the routine calculates the shortest distance between two points using the distance of each segment. However, each segment can be weighted by travel time, speed or travel costs. If travel time is used for weighting the segment, the routine calculates the shortest time for any route between two points. If speed is used for weighting the segment, the routine converts this into travel time by

dividing the distance by the speed. Finally, if travel cost is used for weighting the segment, the routine calculates the route with the smallest total travel cost. Specify the weighting field to be used and be sure to indicate the measurement units (distance, speed, travel time, or travel cost) at the bottom of the page. If there is no weighting field assigned, then the routine will calculate the path using distance.

From one-way flag and To one-way flag

One-way segments can be identified in a bi-directional file by a 'flag' field (it is not necessary in a single directional file). The 'flag' is a field for the end nodes of the segment with values of '0' and '1'. A '0' indicates that travel can pass through that node in either direction whereas a '1' indicates that travel can only pass from the other node of the same segment (i.e., travel cannot occur from another segment that is connected to the node). The default assumption is for travel to be allowed through each node (i.e., there is a '0' assumed for each node). There is a 'From one-way flag' field and a 'To one-way flag' field. For each one-way street, specify the flags for each end node. A '0' allows travel from any connecting segments whereas a '1' only allows travel from the other node of the same segment. Flag fields that are blank are assumed to allow travel to pass in either direction.

FromNode ID and ToNode ID

If the network is single directional, there are individual segments for each direction. Typically, two-way streets have two segments, one for each direction. On the other hand, one-way streets have only one segment. The FromNode ID and the ToNode ID identify from which end of the segment travel should occur. If no FromNode ID and ToNode ID is defined, the routine will chose the first segment of a pair that it finds, whether travel is in the right or wrong direction. To identify correctly travel direction, define the FromNode and ToNode ID fields.

Network coordinate system

The type of coordinate system for the network file is the same as for the primary file.

Segment measurement unit

By default, the shortest path is in terms of distance. However, each segment can be weighted by travel time, travel speed, or travel cost.

1. For travel time, the units are minutes, hours, or unspecified cost units.

2. For speed, the units are miles per hour and kilometers per hour. In the case of speed as a weighting variable, it is automatically converted into travel time by dividing the distance of the segment by the speed, keeping units constant.
3. For travel cost, the units need to be defined in terms of cost per unit distance (e.g., per mile, per kilometer). The routine will then identify routes by those with the smallest total cost.

Network Utilities

There are two network utilities that can be used.

Check for one-way streets

First, there is a routine that will identify one-way streets *if* the network is single directional. In a single directional file, one-way streets do not have a reciprocal pair (i.e., a segment traveling in the opposite direction). This is indicated by a reciprocal pair of ID's for the "From" and "To" nodes. If checked, the routine identifies those segments that do not have reciprocal node ID's. The network is saved with a new field called "**Oneway**". One-way segments are assigned a value of '1' value and two-way segments are assigned a value of '0'. The output is saved as an *ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Create a transit network from primary file

Second, there is a routine that will create a network from the primary file. This is useful for creating a transit network from a collection of bus stops (bus network) or rail stations (rail network). If checked, the routine will read the primary file and will draw lines from one point to another *in the order* in which the points appear in the primary file. Note, it is essential to order the points in the same order in which the network should be drawn (otherwise, an illogical network will be obtained). It is easy to do this in a spreadsheet program.

Transit Line ID

The routine can handle multiple lines, for example different rail lines or bus routes (e.g., Line A, Line B, Route 1, Route 2). In the primary file, the points must be grouped by lines, however, and must be classified by an ID field. Within each group, the points must be arranged

in order of occurrence; the routine will draw a lines from one point to another in that order. In the Transit Line ID field, indicate which variable is the classification variable.

The output is saved as an *ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Network Output

There are three types of output for the network assignment routine. First, the most frequent inter-zonal (i.e., trips between different zones) routes can be output as polylines. Second, the most frequent intra-zonal (i.e., trips within the same zone) routines can be output as points. Third, the entire network can be output in terms of the total number of trips that occur on each segment (network load).

Save routes

The shortest routes can be saved as separate **polyline** objects for use in a GIS. Specify the output file format (*ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats) and the file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Save top routes

Because the output file is very large (number of origin zones x number of destination zones), the user can select a zone-to-zone route with the most predicted trips. The default is the top 100 origin-destination combinations. Each output object is a line from the origin zone to the destination zone with a Route prefix. The prefix is placed before the output file name. The graphical output includes:

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (ROUTE)
3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)

6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of trips on that particular route (FREQ)
10. The distance between the origin zone and the destination zone (DIST).

Save points

Intra-zonal trips (trips in which the origin and destination are the same zone) can be output as separate **point** objects as an *ArcGIS* 'shp', *MapInfo* 'mif' or various ASCII formats. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Again, the top K points are output (default=100). Each output object is a point representing an intra-zonal trip with a RoutePoints. The prefix is placed before the output file name.

The graphical output for each includes:

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (ROUTEPoints)
3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of trips on that particular route (FREQ)
10. The distance between the origin zone and the destination zone (DIST).

Save network load

It is also possible to save the total network *load* as an *ArcGIS* 'shp', *MapInfo* 'mif' or ASCII file. This is the total number of trips on each segment of the network. The routine takes every origin zone to destination zone combination and sums the number of trips that occur on each segment of the network. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo

system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

Click on the "Save output network" box and specify a file name for the output.

Crime Travel Demand Case Studies

Chapters 31 and 32 present two case studies on crime travel demand, one by Richard Block and one by Dan Helms.

File Worksheet

The file worksheet allows the saving of names for the files in the crime travel demand module. Because there are a large number of files used (many used in multiple routines), saving the names will make it easier to keep track of the files. The file worksheet is not required for use in the crime travel demand module. But we do recommend using it remember the names of files in a particular travel demand model. There are five worksheets for keeping track of the different routines.

File Worksheet 1

This worksheet keeps track of the files used in the trip generation step. These include:

Trip generation

Calibrate model
Make prediction
Balance origins with destinations

File Worksheet 2

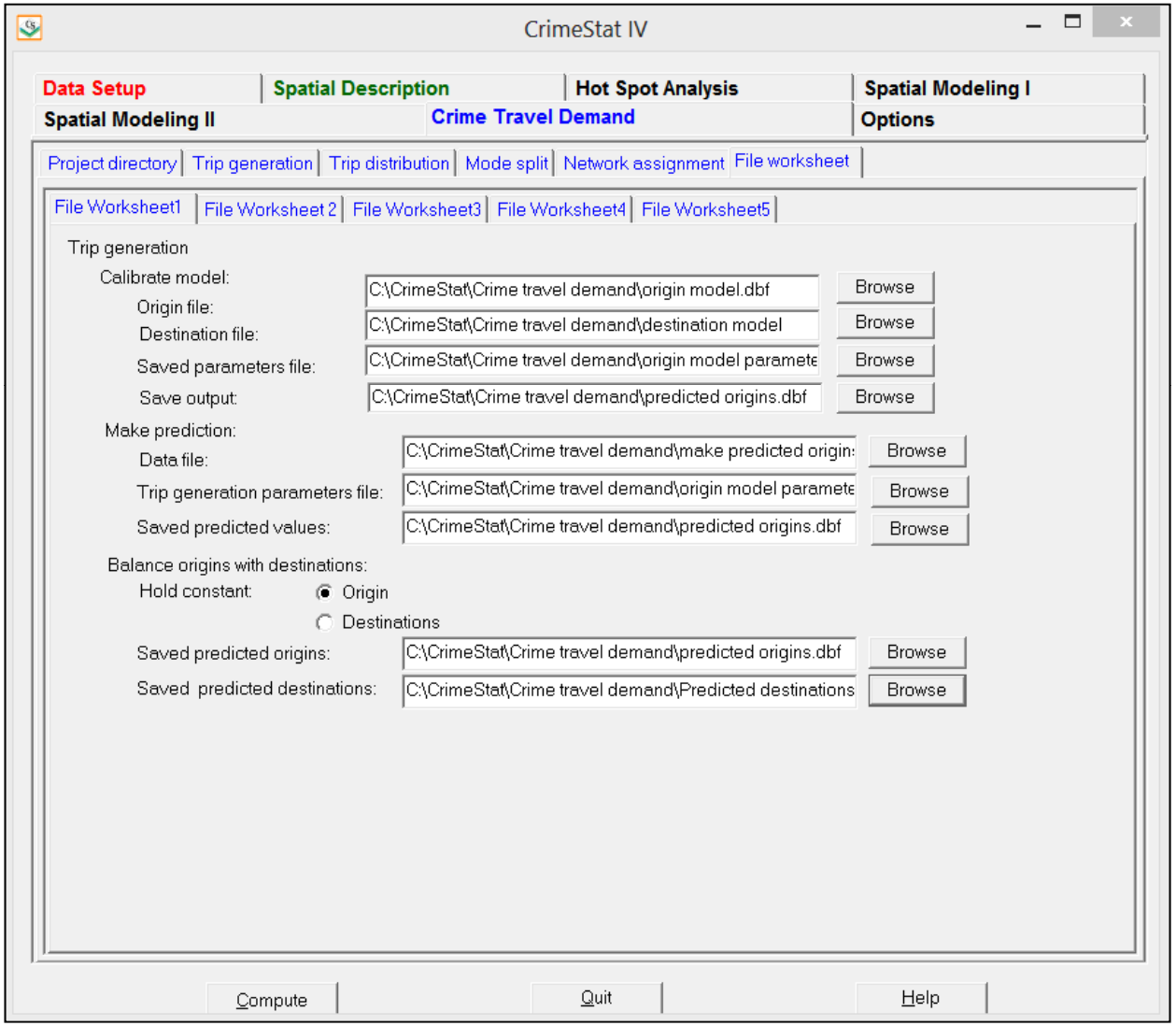
This worksheet keeps track of some used in the trip distribution step, in particular the observed trip distribution and trip distribution model setup. These include:

Trip distribution

Describe origin-destination trips
Setup origin-destination model

Figure 2.27:

Crime Travel Demand File Worksheet



File Worksheet 3

This worksheet also keeps track files used in the trip distribution step, in particular the trip distribution model and the comparison between the observed and predicted trip length distributions. These include:

Origin-destination model
Compare observed and predicted origin-destination trip lengths

File Worksheet 4

This worksheet keeps track of the files used in the mode split step, including the mode split setup and modes 1-3. These include:

Mode split

Setup for mode split
Modes modeled
Modes 1-3

File Worksheet 5

This worksheet keeps track of the remaining files used in the mode split step (modes 4-5) as well as network assignment routine. These include:

Mode split (continued)

Modes modeled
Modes 4-5

Network assignment

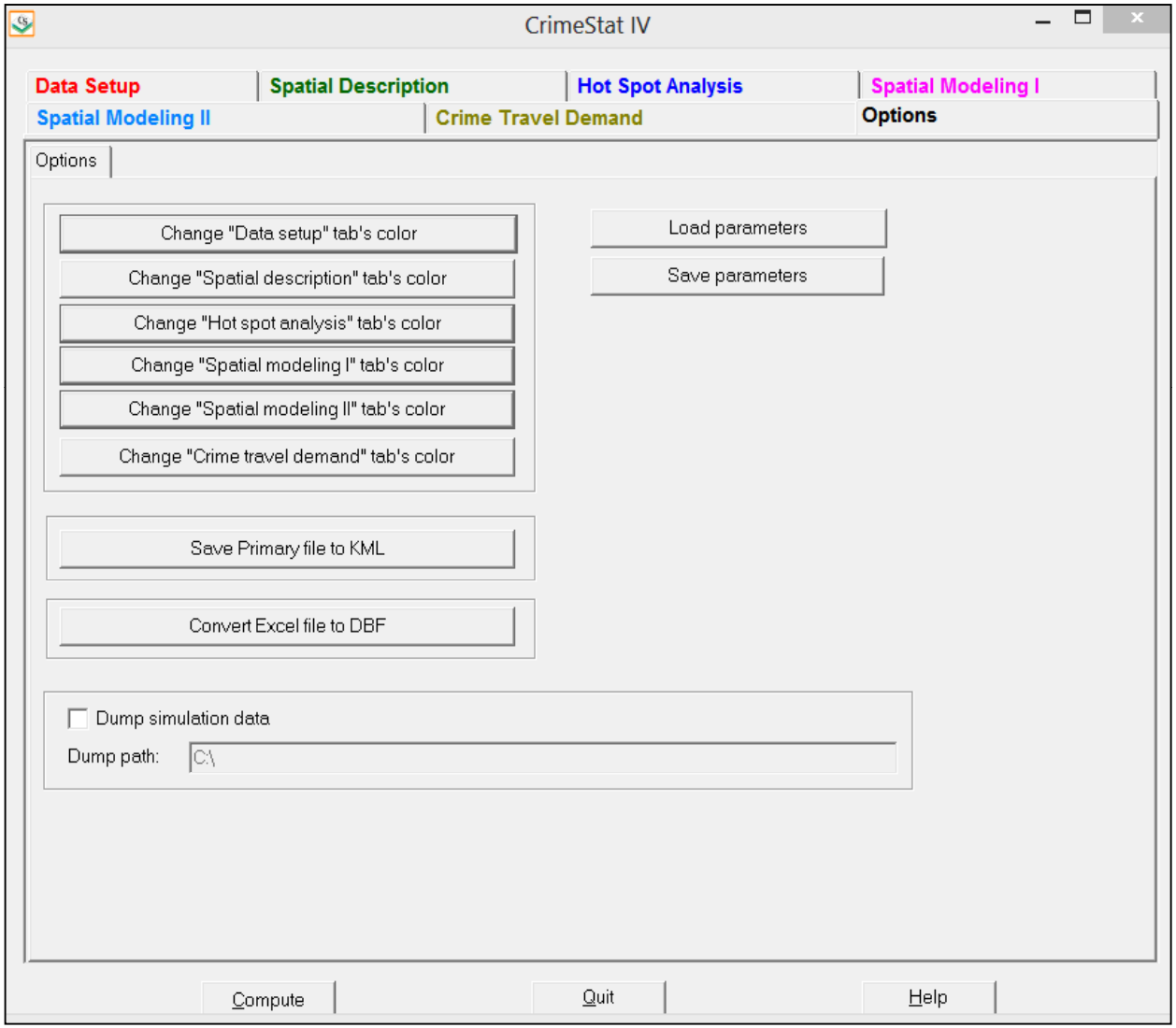
VII. Options

The Options page includes six features that can improve the usability of CrimeStat.

1. Colors for tabs. The user can select one of tens of thousands of colors for each of the major tabs. The Options tab remains black.

Figure 2.28:

CrimeStat Options



2. There is a utility for saving the Primary file to *Google Earth* 'kml' files ***if*** the coordinate system is spherical (longitude and latitude). *Google Earth* only accepts universal, spherical coordinates so that this option is not available if the data are projected. Many of the routines in *CrimeStat* can save objects as 'kml' if the coordinate system is spherical. This utility allows the primary file to be also converted for display in *Google Earth*.

3. There is a utility for converting Excel 'xls' and 'xlsx' files to 'dbf'. Excel is a very common format for data storage. However, *CrimeStat* was designed around 'dbf' files. The utility allows Excel spreadsheets to be quickly converted to 'dbf'. Note that only single sheet (page) Excel files can be converted (not multi-sheet files).

   A. Click on the utility and then find the file to be converted.

   B. Define the output name

   C. Click 'O.K.' and the file will be converted to a 'dbf' file.

4. The user can specify a directory for dumping simulation files (the default is none).

5. The user can *save CrimeStat* parameters in a parameter 'param' file. Only top level parameters can be saved, however. The parameters selected on dialogues that open (e.g., Advanced options) cannot be saved.

6. The user can *load* a *CrimeStat* parameter file. Again, Only top level parameters can be loaded.

Chapter 3:

Entering Data into *CrimeStat IV*

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 3:

Entering Data into *CrimeStat IV*

Organization of Program into Tabs

The graphical user interface of *CrimeStat* is a tabbed form (Figure 3.1). These are divided into six general statistical categories plus an options tab with more than 80 individual routines:

Data Setup

Primary file

Input file with X/Y coordinates
Define coordinate system
Define data units

Secondary file

Input second file with X/Y coordinates as baseline
Define coordinate system
Define data units

Reference file

Create reference grid
Use existing reference grid

Type of distance measurement

Use direct distance
Use indirect distance
Use network distance

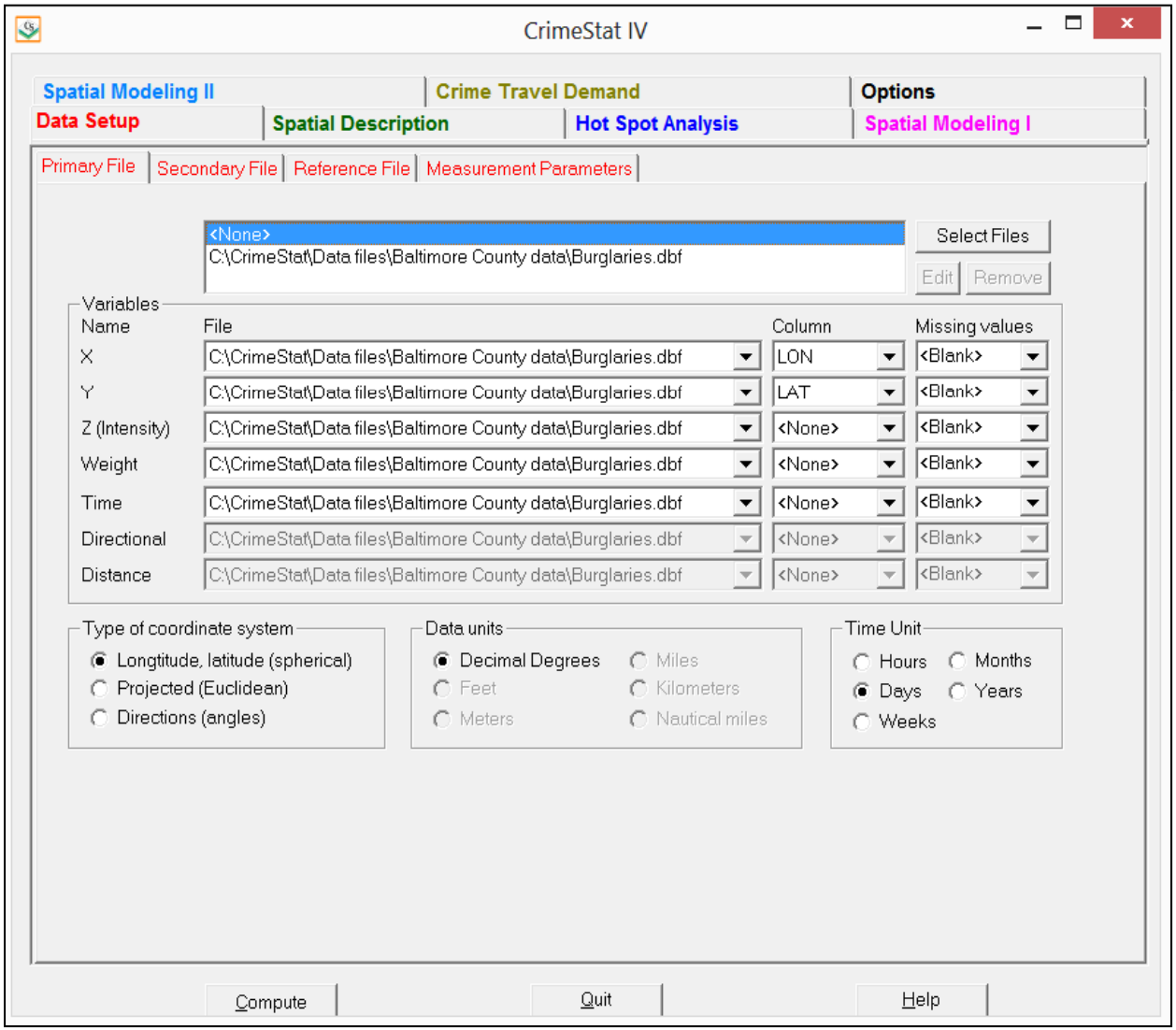
Spatial Description

Spatial distribution

Mean center
Standard distance deviation
Standard deviational ellipse
Median center

Figure 3.1:

CrimeStat User Interface



Center of minimum distance
Directional mean and variance
Convex Hull

Spatial Autocorrelation

Moran's "I" spatial autocorrelation index
Geary's "C" spatial autocorrelation index
Adjusted Geary's "C" spatial autocorrelation index
Getis-Ord Global "G" spatial autocorrelation index with simulation of credible intervals
Moran Correlogram with simulation of credible intervals
Geary Correlogram with simulation of credible intervals
Getis-Ord Correlogram with simulation of credible intervals

Distance analysis I

Nearest neighbor analysis
Ripley's "K" statistic
Assign primary points to secondary points

Distance Analysis II

Within primary file distance matrix
Between primary file and secondary file distance matrix
Between primary file and grid distance matrix
Between secondary file and grid distance matrix

Hot Spot Analysis

Hot spot analysis I

Mode
Fuzzy mode
Nearest neighbor hierarchical clustering with simulation of credible intervals
Risk-adjusted nearest neighbor hierarchical clustering with simulation of credible intervals

Hot spot analysis II

Spatial and temporal analysis of crime routine (STAC) with simulation of credible intervals
K-mean clustering

Hot spot analysis of Zones

Anselin's local Moran test with simulation of credible intervals
Getis-Ord local "G" test with simulation of credible intervals
Zonal nearest neighbor hierarchical clustering with simulation of credible intervals
Risk-adjusted zonal nearest neighbor hierarchical clustering with simulation of credible intervals

Spatial Modeling I

Interpolation I

Single variable kernel density interpolation
Dual variable kernel density interpolation

Interpolation II

Head-Bang analysis
Interpolated Head-Bang analysis

Space-time analysis

Knox index
Mantel index
Correlated walk model for analysis and prediction

Journey-to-crime analysis

Calibrate Journey-to-crime function
Journey-to-crime estimation
Draw crime trips

Bayesian Journey-to-crime analysis

Diagnostics for Journey-to-crime methods
Estimate likely origin of a serial offender

Spatial Modeling II

Regression I

MLE Normal (OLS) and Poisson regression models
MCMC Normal, Poisson, and Logit regression models
MCMC Normal, Poisson, and Logit exposure regression models

MCMC spatial Normal, Poisson, and Logit regression models
MCMC spatial Poisson and Logit exposure regression models

Regression II

Using OLS regression models to make predictions
Using Poisson spatial regression models to make predictions

Discrete Choice I

Create dataset for conditional logit model
Estimate multinomial logit model
Estimate conditional logit model

Discrete Choice II

Using multinomial logit model to make predictions
Using conditional logit model to make predictions

Time Series Forecasting

Exponential smoothing
Exponential smoothing forecast
Trigg Tracking Mechanism

Crime Travel Demand

Trip Generation

Skewness diagnostics
Calibrate model
Make prediction
Balance predicted origins & destinations

Trip Distribution

Calculate observed origin-destination trips
Calibrate impedance function
Calibrate origin-destination model
Apply predicted origin-destination model
Compare observed and predicted origin-destination trip lengths

Mode Split

Calculate mode split for trips

Network Assignment

Check for one-way streets
Create a transit network from primary file
Network assignment of trips to travel network

Required Data

CrimeStat can input data in one of three formats - ASCII, *dbase III/IV* 'dbf' and, *ArcGIS*® point shape files 'shp',. The default is 'dbf'. It is essential that the files have X and Y coordinates as part of their structure. The program assumes that the assigned X and Y coordinates are correct.

The default is 'dbf. This is an older format but is well structured for numerical analysis. With ASCII formats, the columns have to be defined (see below). Finally, only *point* shape files can be read by *CrimeStat*.[1]

If you read an *ArcGIS*® point shape file, the incident's X and Y coordinates are automatically added as the first fields in the primary file by *CrimeStat*. *CrimeStat* also can read in a secondary file which also must have X and Y coordinates included as separate fields. For several of the modules (regression, discrete choice, time series forecasting), a non-spatial file (without coordinates) can be read, but in general most routines require coordinates.

Excel to dbf Conversion Utility

Since Excel is a very common file format, *CrimeStat* has a utility for converting Excel 'xls' and 'xlsx' files into 'dbf' files. The utility is located on the options page. Click on the button 'Convert Excel file to DBF' and then locate the Excel file. Then, choose a name for the output file and click on 'O.K.'. A copy of the file will be made in 'dbf' format, which is the standard format for *CrimeStat*.

There are a couple of issues for which users should be aware.

1. First, the utility only will work with single sheet Excel files. Any file that has more than one sheet will not be converted.

[1] *CrimeStat* cannot read polygon shape files nor multi-point shape files.

2. Second, the utility interprets the first row as the field names. For every label it sees, it will identify that column as a field or variable. If there are any columns that have blanks in the first row, then that column will not be converted.

3. Third, the utility interprets the variable it finds in the first non-blank record as the type of variable (numeric or alphanumeric). Be sure that *all* columns are consistent in the type of variable.

4. Fourth, there are limits to the number of columns that can be converted (256) and the width of each column (20 characters). Error messages will be displayed if the Excel file exceeds these limits.

5. Fifth, and finally, users should *clean* datasets thoroughly before trying to convert them to dbf files. Eliminate unnecessary fields and fields with many blank records (the results will be unreliable if a high percentage of the records have blank values). Be careful about fields that will be converted to unreadable characters. Many software packages introduce formatting characters into fields. When these are converted, they produce strange characters and are unreadable. If this happens, delete the field before converting.

In short, as with any statistical package, a clean dataset is essential for providing useful information as well as allowing *CrimeStat* to work properly with the data.

Coordinates

CrimeStat analyzes point data, defined geographically by X and Y coordinates. These X/Y coordinates represent a single location where either an incident occurred (e.g., a burglary) or where a person, building or other object can be represented as a single point. A point will have X and Y coordinates in a spherical or Cartesian system. In a spherical coordinate system, each point can be defined by longitude (for X) and latitude (for Y). In a projected coordinate system, such as State Plane or UTM, each X and Y is defined by feet or meters from an arbitrary reference origin. *CrimeStat* can handle both spherical and projected points. For some uses, coordinates can be polar, that is defined as angles from an arbitrary reference vector, usually direct north.[2] One of the routines in the program calculates the angular mean and variance of a collection of angles.

---

[2] The spherical 'lat/lon' system is, of course, one type of polar coordinate system. But, it is a polar coordinate system with particular restrictions. Latitudes are angles up to 90^0, north or south of the Equator. Longitudes are angles from 0^0 to 180^0, east and west of the Greenwich Meridian. In the usual polar coordinate system, angles can vary from 0^0 to 360^0.

Point data can be obtained from a number of sources. The most frequent would be the various incident data bases stored by a police department, which could include calls for service, crime reports, or closed cases. Other sources of incident data can include secondary data from other agencies (e.g., hospital records, emergency medical service records, locations of businesses) or even sampled data (Levine & Wachs, 1986a; 1986b). There are also point data from media sources such as radio and televisions, and potentially from Internet sources.

To read projected coordinates into *CrimeStat*, the user does not need to define the particular projection (other than to indicate that the coordinates are projected). *ArcGIS*® will output the objects in the projected units so that they can be read directly into that program or into *ArcGIS*®. However, to output calculated objects to *MapInfo*® requires the definition of a specific projection used (see endnote *i*) or the use of the Universal Translator in *MapInfo*® (see Dick Block attachment at the end of the chapter).

Intensities and Weights

For some uses, points can have *intensity* values or *weights*. These are optional inputs in *CrimeStat*. An *intensity* is a value assigned to a point location aside from the X/Y coordinates. It is another variable, typically denoted as a Z-value. For example, if the point location is the location of a police station, then the intensity could be the number of calls for service over a month at that station. Or, for census geography, if the point is the centroid of a census tract, then the intensity could be the population of that census tract. In other words, an intensity is a variable assigned to a particular location.

Some of the routines in *CrimeStat* require an intensity value (e.g., the spatial autocorrelation indices) and others can utilize a point location with an intensity value assigned (e.g., kernel density interpolation). If no intensity value is assigned, the routines which require it cannot be run while the routines which can utilize it will assume that the intensity is 1 (i.e., that all points have equal intensity).

A *weight* occurs when different point locations are to receive differential statistical treatment. For example, if a police department has designated different areas for service, for example 'urban' and 'rural', a value can be assigned for each of these areas (e.g., '1' for urban and '2' for rural). Many of the routines in *CrimeStat* will use the weights in the calculations. Weights would be useful if different zones are to be evaluated on the basis of another variable.

Longitudes are angles from 0^0 to 180^0, east and west of the Greenwich Meridian. In the usual polar coordinate system, angles can vary from 0^0 to 360^0.

For example, suppose a police department has divided its service area into urban and rural. In the rural part, there are twice as many patrol officers assigned per capita than in the urban areas; the higher population densities in the urban areas are assumed to compensate for the longer travel distances in the rural areas. Let us assume that all crimes occurring in the rural areas receive a weight of 2 while those in the urban area receive a weight of 1. The police department wants to estimate the density of household burglaries relative to the population using the dual kernel density function (see Chapter 10). But, to reflect the differential assignment of police officers, the analysts use the service area as a weight. The result would be a per capita estimate of burglary density (i.e., burglaries per person), but weighted by the service area. It would provide an estimate of burglary risk adjusted for differential service in rural and urban areas. In most cases, there will no weights, in which case, all points are assumed to have an equal weight of '1'.

It is possible to have both intensities and weights, although this would be rare. For example, if the X and Y coordinates are the centroids of census tracts, a third variable - the total population of each census tract could be an intensity. There could also be an weighting based on service area. In calculating the Moran's "I" spatial autocorrelation index, the total population is used as an intensity while the service area is used as a weight. In this case, *CrimeStat* calculates a weighted Moran's I spatial autocorrelation.

But the use of both an intensity variable *and* a weight would be less common. For most of the statistics, a variable could be used as *either* a weight or intensity, and the results will be the same. However, be careful in assigning the same variable as both intensity and a weight. In such instances, cases may end up being weighted twice, which will produce distorted results.[3]

Time Measures

CrimeStat includes several routines for analyzing spatial characteristics in relation to time. Many serial crime incidents occur in a short period of time. For example, a group of car thieves may steal cars from a neighborhood over a very short period of time, for example a few days. Thus, there is often an interaction between a concentrated spatial pattern of events

[3] An alternative way to thinking about intensities and weights is to treat both as two different weights - weight #1 and weight #2. For example, weight #1 could be the population in a surrounding zone while weight #2 could be the employment in that same zone. Thus, incidents (e.g., burglaries) could be weighted both by the surrounding population and the surrounding employment. The analogy with double weights is not quite correct since several of the statistics (Moran's I, Geary's C and Local Moran) use only intensity, but not a weight. The distinction between intensities and weights is historical, relating to the manner in which the statistics have been derived.

occurring in a short time period. Because of this, police departments routinely collect information on the time of the event, the day and time.

There are three routines which analyze spatial concentration in relation to time: the Knox index, the Mantel index, and a correlated walk model. But for using any of these routines, the user has to define time in a consistent manner. Both the primary and secondary files can allow a time variable. However, these have to be defined in a *consistent* manner for all records in a file. There are five time periods that are allowed:

Hour
Day (default)
Week
Month
Year

The default is 'day'. That is, the program will assume that any time variable is in days, either an arbitrary number of days (e.g., days from January 1st) or the number of days from January 1, 1900, which is the default time reference for most computer systems. If the time unit is not in days, the user needs to indicate the appropriate unit.

There is also an entire module for analyzing temporal changes by zone and detecting emerging incidents (Spatial Modeling II). For these routines, the temporal and geographical identifiers are coded into the structure of the data set and do not have to be separately defined. See Chapters 23 and 24 for details.

Missing Value Codes

Unfortunately, data is frequently messy. In most police departments, the crime incident data base is being continually updated, daily and, perhaps, hourly. At any one time, many of the records will not have been geocoded or will have been incompletely geocoded.

Blank records

CrimeStat allows the inclusion of codes for missing values, that is, values of eligible fields that are not complete or are not correct. These codes are applied to the fields defined on the primary or secondary data sets (X, Y, weight, intensity). Automatically, *CrimeStat* will exclude records with blank fields or with fields having any non-numeric value (e.g., alphanumeric characters, #, *) for the eligible fields. The statistics will be calculated only on

those records which have eligible numerical values. Fields for other variables in the data base that are not defined in the primary and secondary data sets will be ignored.

Other missing value codes

In addition to blank and non-numeric values, *CrimeStat* can exclude any other value that has been used for a missing values code (e.g., 0, -1, 99). That is, if the program encounters a field with a missing value code, it will exclude that record from the calculations. Next to the X, Y, weight, and intensity fields on both the primary and secondary files is a missing values code box. The default has been set to blank. That is, if *CrimeStat* finds no information in a field, it will ignore that record. However, there are eight options that can be selected:

1. ***<blank>*** fields are automatically excluded. This is the default;
2. ***<none>*** indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0;
3. ***0*** is excluded;
4. ***-1*** is excluded;
5. ***0 and -1*** indicates that both 0 and -1 will be excluded;
6. ***0, -1 and 9999*** indicates that all three values (0, -1, 9999) will be excluded;
7. ***Any*** other numerical value can be treated as a missing value by typing it (e.g., 99); and
8. ***Multiple*** numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99).

It is important for users to understand their data sets prior to using *CrimeStat*. If the data are 'clean', that is all X/Y fields are populated with correct values as are all weight/intensity fields (if used), then the program will have no problems running routines. On the other hand, in large administrative data bases, such as in most police departments, there will be many records that are incomplete or have missing values codes (e.g., 0). Unless *CrimeStat* is told what are the missing value codes, with the exception of blank or non-numeric values it will include them in the calculations. For example, some data base programs put a 0 for an X or Y field which has not been geocoded. *CrimeStat* does not know that the 0 is a missing value and will use it in calculations since 0 is a perfectly good number. It is important that users either clean their data thoroughly or define the missing value codes completely for the primary and secondary files.

Primary File

The *Primary File* is required to run the program and provides the coordinates of points of incidents. On the primary file tab, first click on *Select Files*. A dialog box appears that allows

the selection of three file formats for the primary file (Figure 3.2). For each of the file formats, the user must define two characteristics - the type of file (ASCII, '.dbf', or '.shp') and the name of the file. There is a browse window that allows the user to find the file.

In developing this program, we have targeted it towards users of *ArcGIS®*, *MapInfo®* and other GIS programs (e.g., Maptitude®). These GIS programs either store their attribute data in *dBase III/IV/V* format in a file with a 'dbf' extension (e.g., precinct1.dbf) or can read and write directly 'dbf' files. Many other GIS programs, however, also can read 'dbf' files. For *ArcGIS®* and *MapInfo®*, the X and Y coordinates which define crime incident points are not directly part of the 'dbf' file, but instead exist on the geographic file.

Input File Formats

Dbf

In *CrimeStat*, the default file format is'dbf'. These are files that have rows as records and columns as fields/variables. There is a limit of 256 fields. Since *CrimeStat* works with numeric data, user should minimize or even eliminate alphanumeric fields since these can take up a lot of space on the hard disk. The one exception is the need for an ID field for many of the routines.

Another consideration is the size of each field. Some programs create 'dbf' files with 64 or more decimal places. These files end up being very large and take a long time to process. The additional precision with 64 decimals is completely non-essential. A user would be advised to reduce the number of decimal places. Usually, no more than 12-15 decimals places are sufficient for a high degree of calculation accuracy.

Shp

In *ArcGIS®* the coordinates are stored on the 'shp' file, not the 'dbf' file. *CrimeStat* can read directly a 'shp' file so the 'dbf' file is not required to have the X and Y coordinates.

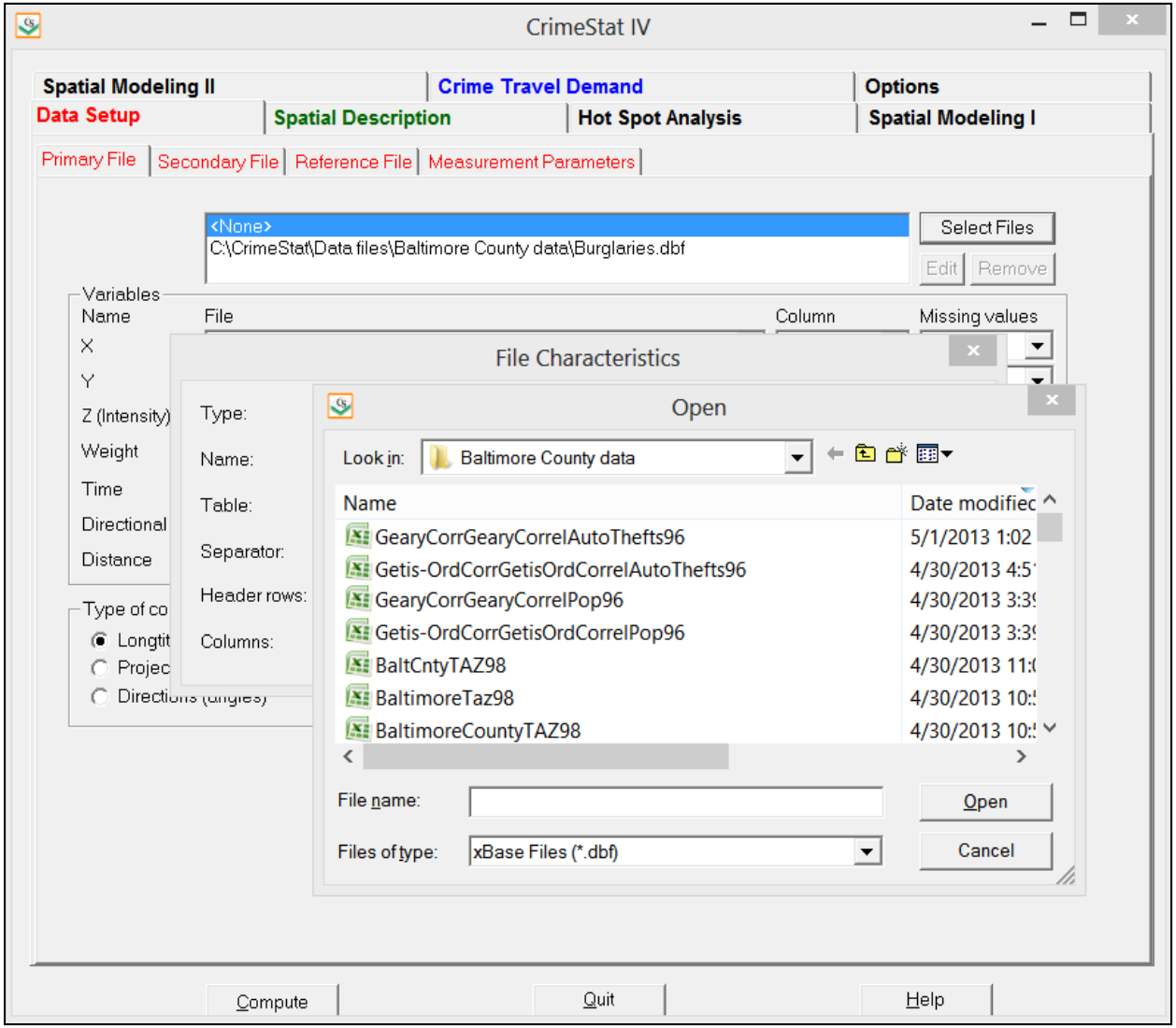
ASCII

For an ASCII file, however, three additional characteristics must be defined. The first is the type of character used to separate (delimit) variables in the file. There are four possibilities:[4]

[4] Note that in an ASCII file, a tab *looks like* it is separated by spaces. However, the underlying ASCII code is different and *CrimeStat* will treat these characteristics differently. That is, if the separator is a tab but the user indicates that it is a space, *CrimeStat* will not properly read the data.

Figure 3.2:

File Format Selection



Space (one or more, the default)
Comma delimited
Semicolon
Tab

The second characteristic is the number of rows which have labels on them (*Header Rows*). Some ASCII files will have rows that label the names of the variables. The user should indicate the number if this is the case otherwise *CrimeStat* will produce an error code. The default is 0, that is, the program assumes that there are no headers unless instructed otherwise. To change this, the user should insert the cursor in the appropriate cell, backspace to erase the default number and type in the correct number.

The third characteristic of an ASCII file that must be defined is the number of variables (columns or fields) in the file. With spherical or projected coordinates, there will be at least two variables (the X and Y coordinate) and there may be more if other variables are included in the file. However, with directional coordinates (see below), there may be only one. *CrimeStat* assumes that the number of columns in the ASCII file is two unless instructed otherwise. The user should insert the cursor in the appropriate cell, backspace to erase the default number and type in the correct number. After defining the file type and name, the user should click on *OK*.

Identifying Variables

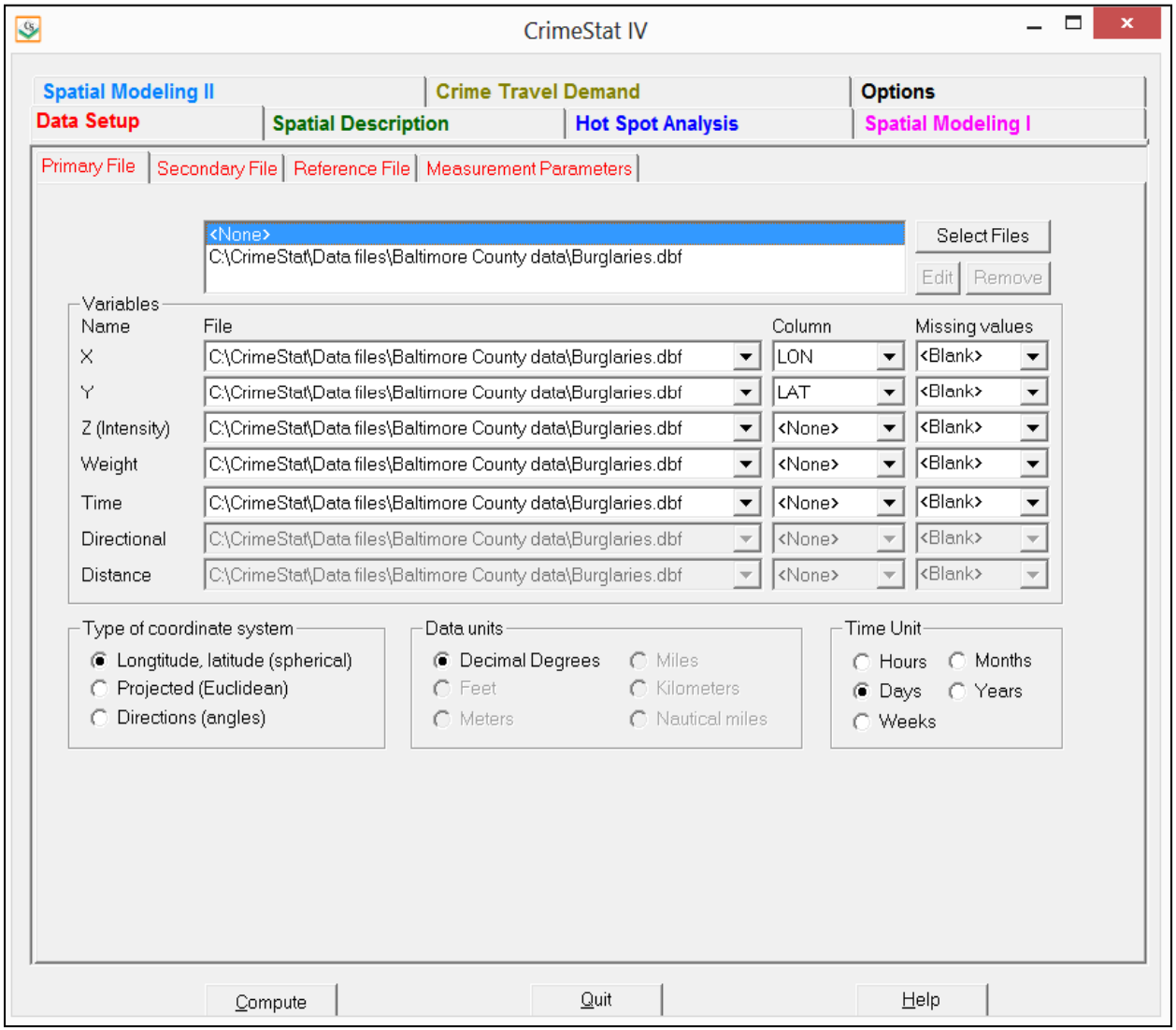
After defining a file and its format, either 'dbf', 'shp' or ASCII, it is necessary to identify the variables. Two variables are required and two are optional. The required variables are the X and Y coordinates. The user should indicate the file name that contains the coordinates by clicking on the drop down menu and highlighting the correct name. After having identified which file contains the X and Y coordinates, it is necessary to identify the variable name. Click on the drop down menu under *Column* and highlight the name of the variable for the X and Y coordinates respectively.[5] Figure 3.3 shows a correct defining of file and variable names for the primary file.

Multiple files can be entered on the primary file tab. However, only one can be utilized at a time. In theory, one can have separate files containing the X and Y coordinates, though in practice this will rarely occur.

[5] Hint: If you type the first letter of the name (e.g., 'L' for longitude), then the program will find the first name that begins with that letter). Typing the letter again will find the second name, and so forth.

Figure 3.3:

Primary File Definition



Weight Variable

Sometimes, a point location is weighted. As mentioned above, weights are used when points represents areas and the areas are statistically treated differently. For most of the statistics, *CrimeStat* can weight the statistics during the calculation (e.g., the weighted mean center, the weighted nearest neighbor index).

By default, *CrimeStat* assigns a weight of 1 to each point. If the user does not define a weight variable, then the program assumes that each point has equal weight (i.e., 1). On the other hand, if there are weights, then the weight variable should be defined on the primary file screen and its name listed.

Intensity Variable

Similarly, a point location can have an intensity assigned to it. Most of the statistics in *CrimeStat* can use an intensity variable and some statistics require it (Moran's I, Geary's C and Local Moran). If no intensity is defined, *CrimeStat* will not calculate statistics requiring an intensity variable and, in statistics where an intensity is optional (e.g., interpolation), will assume a default intensity of 1. On the other hand, if there is an intensity variable, then this should be defined on the primary file screen and its variable name identified.

In general, be very careful about using ***both*** an intensity variable ***and*** a weighting variable. Use both only when there are separate weights and intensities. Most of the routines can use both intensities and weighting and may, consequently, double-weight cases. Figure 3.4 shows a primary file screen with an intensity variable defined.

Time Variable

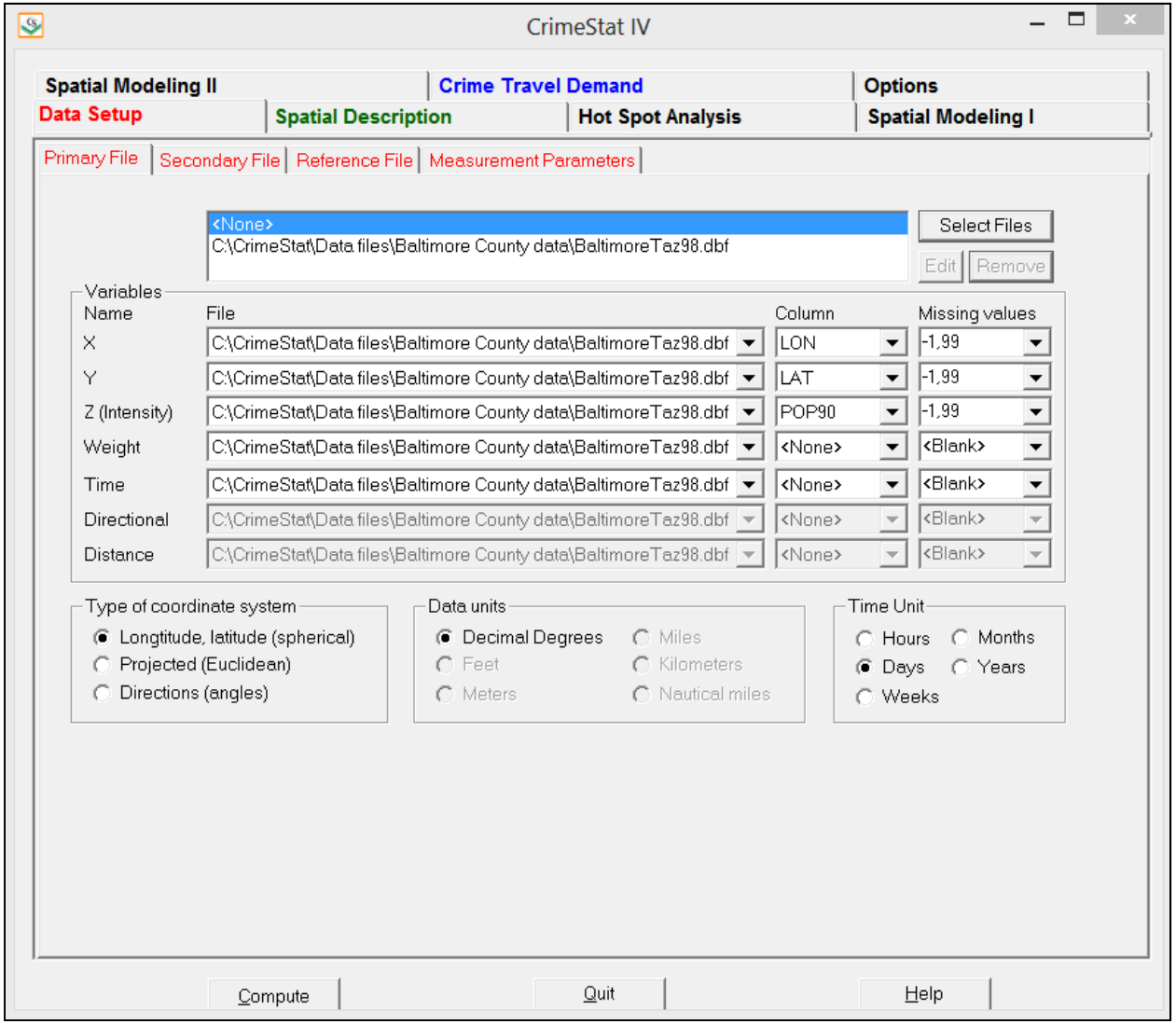
Finally, a time variable can be defined for use in the special Space-time analysis tools under Spatial modeling. *CrimeStat* allows five different time references:

Hours
Days
Weeks
Months
Years

The default is 'days' but the user can choose one of four other time periods. However, the program assumes that all records are consistently defined (i.e., all records use the same time

Figure 3.4:

Primary File with Intensity Variable Defined



unit). For example, if some records are in days but others are in hours, the program will not know that there is an inconsistency and will treat each record as if it was the same time unit. It is important, therefore, to ensure that all records are consistent in the way that time is defined. Figure 3.5 illustrates the defining of a time variable on the primary file page.

Note that the time series forecasting module (under Spatial Modeling II) requires that time be coded into the structure of the data rather than defined as a separate variable. See Chapters 23 and 24 for details.

Coordinate System

In addition to the primary file name and variable assignment, it is necessary to identify the type of coordinate system used and the units of measurement. *CrimeStat* recognizes three coordinate systems:

Spherical coordinates (longitude and latitude)

This is a universal coordinate system that measures location by angles from reference points on Earth. The units are longitude (X coordinate) and latitude (Y coordinate).

Projected coordinates

Projected coordinates are arbitrary coordinates based on a particular projection of the earth to a flat plane. They have an arbitrary origin (the place where X=0 and Y=0) and are almost always defined in units of feet or meters (see endnote *ii*).

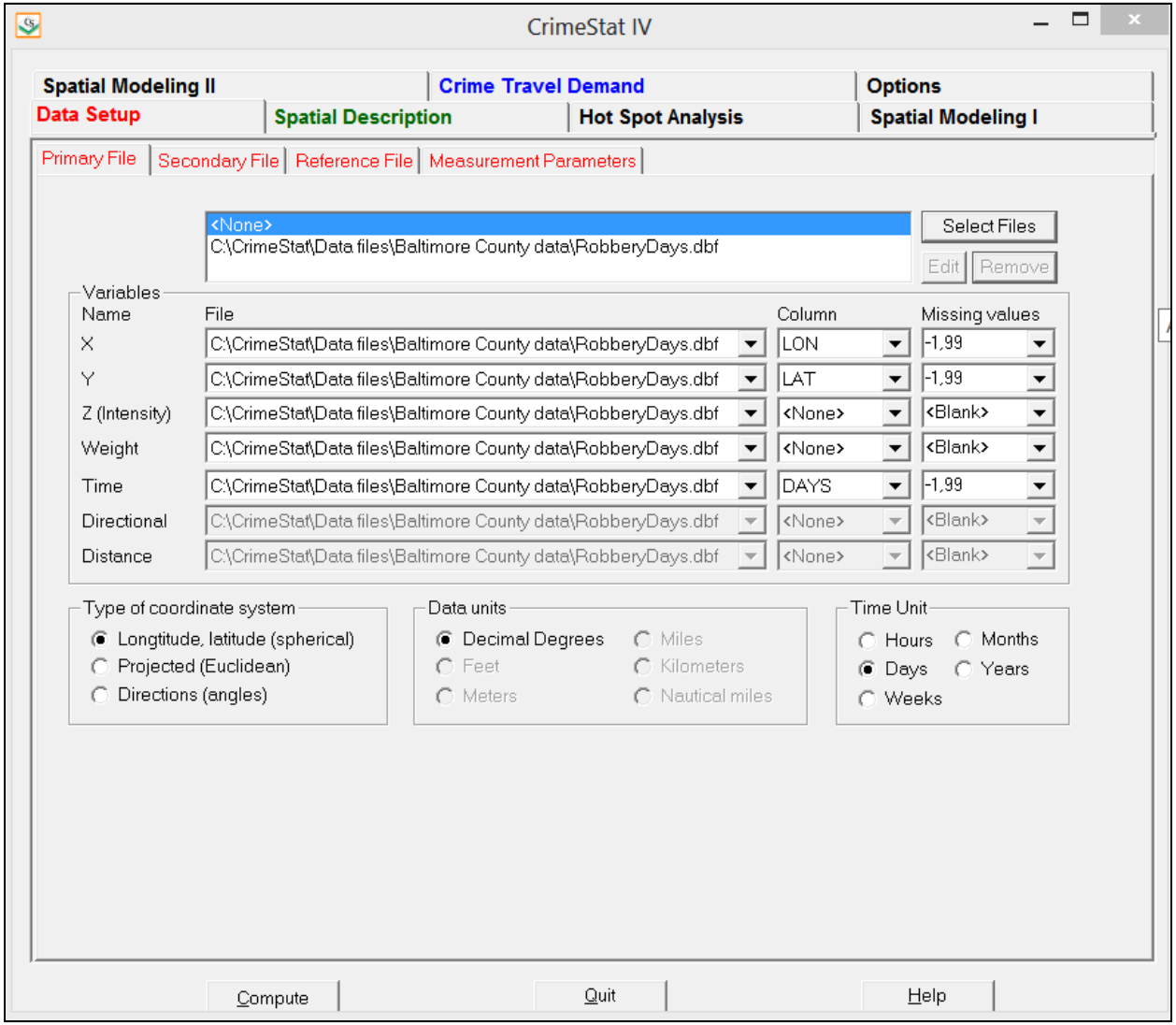
CrimeStat can work with either spherical or projected coordinates. On the primary file tab, the user indicates which coordinate system is being used. If the coordinate system is spherical, then units are automatically assumed to be latitude and longitude in decimal degrees. If the coordinate system is projected, then it is necessary to specify whether the measurement units are feet or meters.

Directional coordinates

For some uses, a polar coordinate system can be used. Point locations are defined by angles from an arbitrary reference line, usually true north and vary between 0^0 and 360^0 in a clockwise rotation. All locations are measured as an angular deviation from the reference point and with distance being measured from a central location. *CrimeStat* has the ability to read in

Figure 3.5:

Time Variable Definition



angles for use in calculating the angular mean and variance. In addition, if directional coordinates are used, an optional distance variable for each measurement can be used.

If the file contains directional coordinates (angles), define the file name and variable name (column) that contains the directional measurements. If used, define the file name and variable name (column) that contains the distance variable. Figure 3.6 shows the primary file definition using directions.

Secondary File

CrimeStat also allows for the inputting of a secondary file. For example, the primary file could be locations where motor vehicles were stolen while the secondary file could be the location where stolen vehicles were recovered. Alternatively, the primary file could be burglary locations while the secondary file could be police stations.

CrimeStat can construct two different types of indices with a secondary file. First, it can calculate the distance from every primary file point to every secondary file point. For example, this might be useful in assessing where to place police cars in order to minimize travel distance in response to calls for service.

Second, *CrimeStat* can utilize both primary and secondary files in estimating a three-dimensional density surface (see Chapter 10). For example, if the primary file are residential burglaries and the secondary file contains the centroids of census block groups with the population within each block group assigned as an intensity variable, then *CrimeStat* can estimate the density of burglaries relative to the density of population (i.e., burglary risk).

The secondary file can also be '.dbf', '.shp' or ASCII. As with a primary file, there must be an X and Y variable defined, but it must be in the same coordinate system and data units as the primary file. The secondary file can also have weights and intensities assigned, but not a time variable.. Figure 3.7 shows the inputting of an ASCII file for the secondary data set while Figure 3.8 shows a correct definition of the secondary file.

Reference File

Several of the routines in *CrimeStat* generalize the point data to all locations in the study area, in particular the one-variable and two-variable density interpolation routines (Chapter 10), the risk-adjusted nearest neighbor hierarchical clustering routine (Chapter 7), the zonal risk-adjusted nearest neighbor hierarchical clustering routine (Chapter 9), the journey-to-crime

Figure 3.6:

File Definition with Angles (Directions)

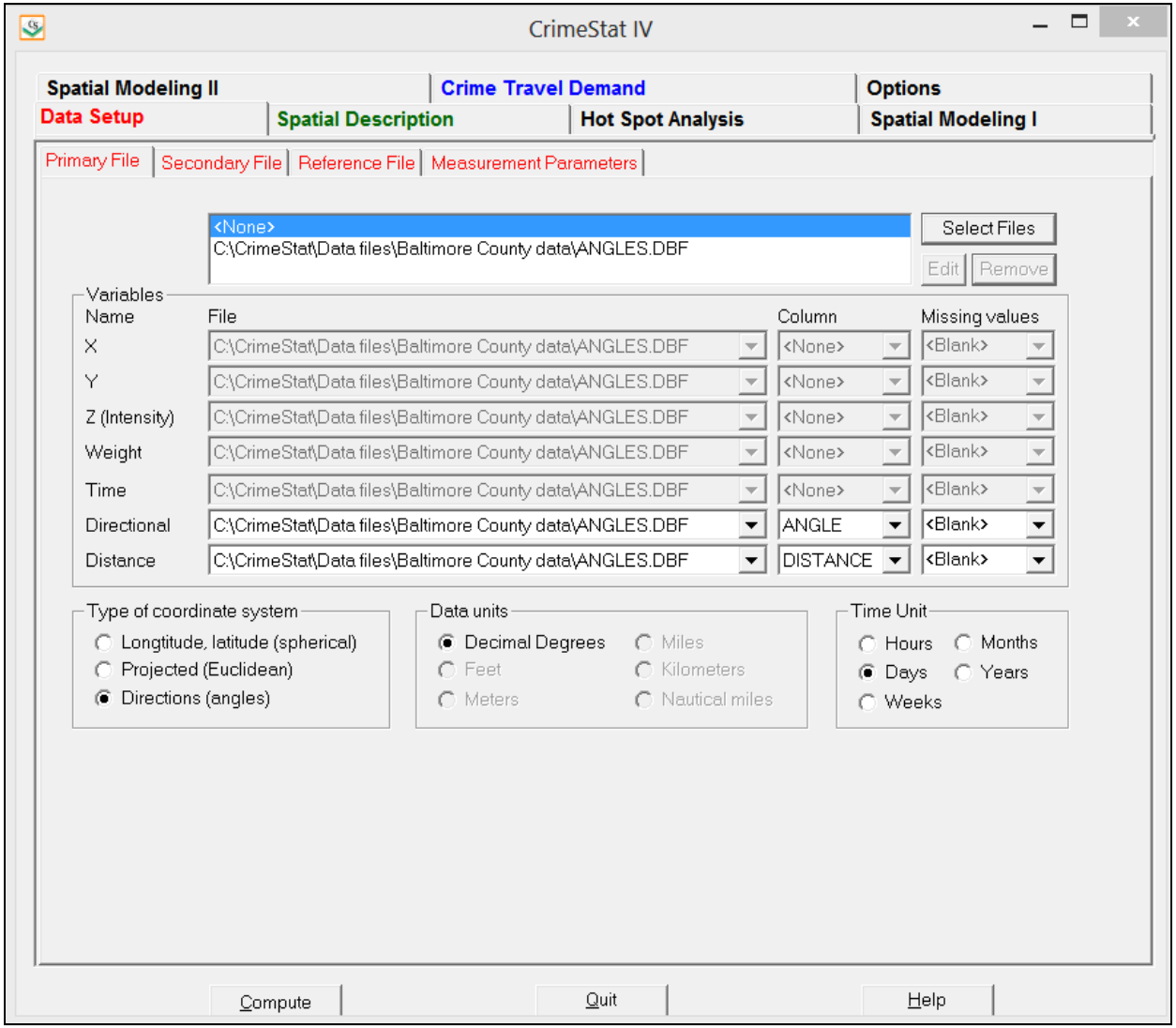


Figure 3.7:

Ascii File Selection of Secondary File

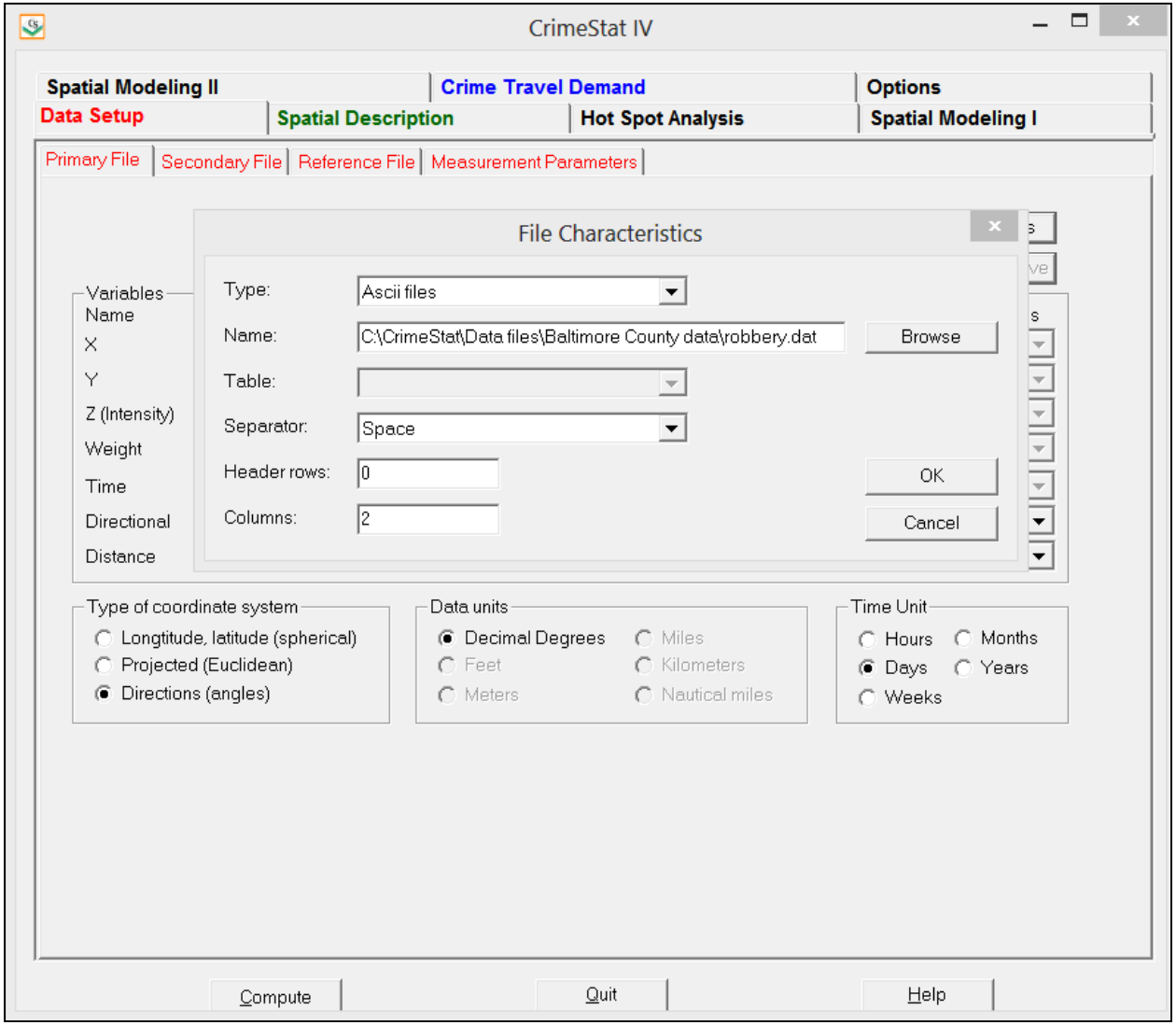
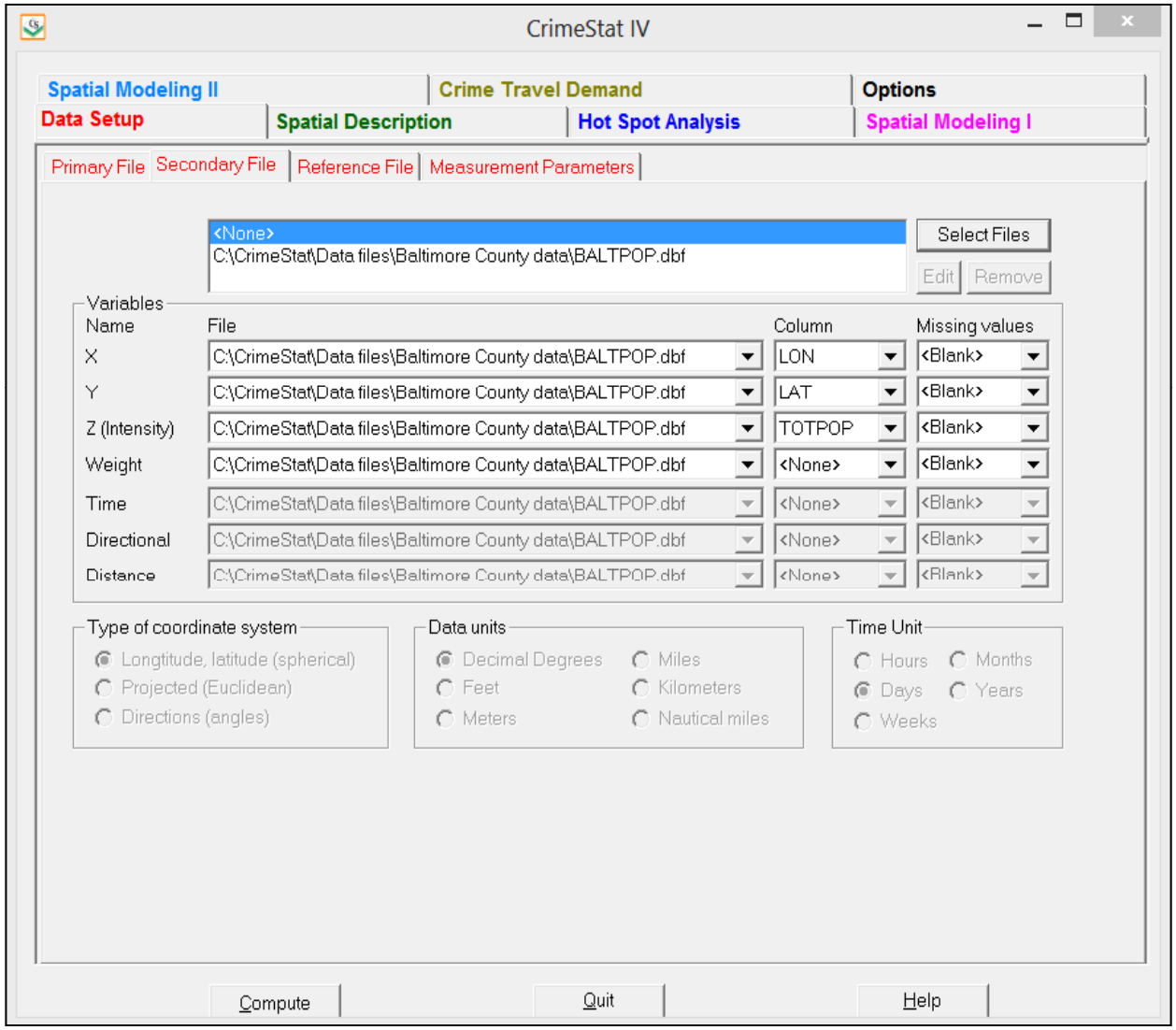


Figure 3.8:

Secondary File Definition



estimation routine (Chapter 13) and the Bayesian journey-to-crime estimation routine (Chapter 14). The generalization uses a reference file placed over the study area. The STAC program also uses a reference file for searching (Chapter 8).

Typically, the reference file is a rectangular grid file (true grid), that is a rectangle with cells defined by columns and rows; each grid cell is a rectangle and column-row combinations are used. It is possible to use a non-rectangular grid file under special circumstances (e.g., a grid with water, mountains or other jurisdictions removed), but a rectangular grid would be used in most cases. *CrimeStat* can create a grid file directly or can read in an external grid file. Figure 3.9 shows a grid placed over both the County of Baltimore and the City of Baltimore.

Creating a Reference Grid

CrimeStat can also create a true grid. There are two steps:

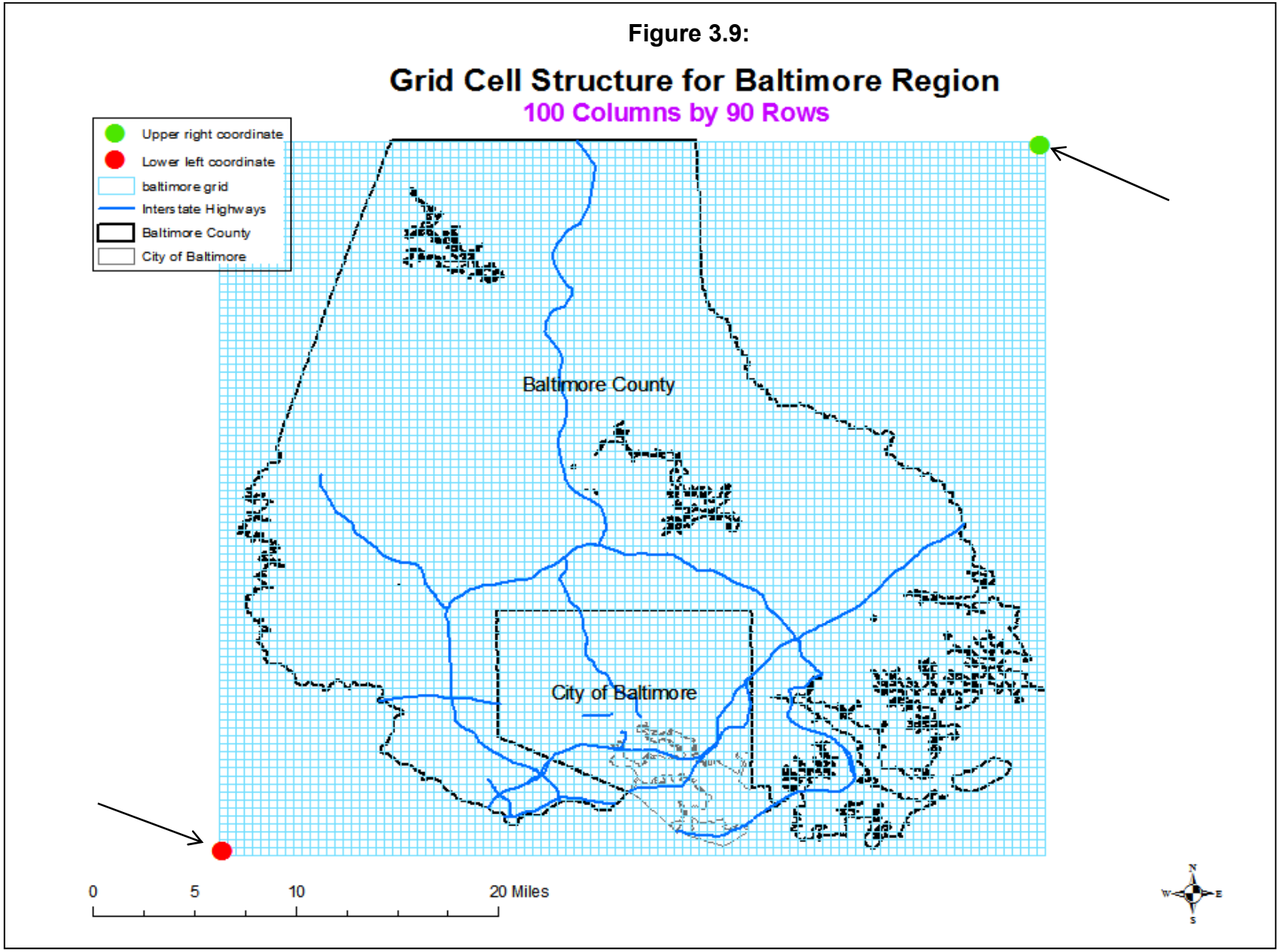
1. The user selects *Create Grid* from the Reference File tab and inputs the X and Y coordinates of the lower-left and upper-right coordinates of the grid. These coordinates must be the same as for the primary file.

Thus, if the primary file uses spherical (lat/lon) coordinates, then the grid file coordinates must also us lat/lon. Conversely, if the primary file coordinates are projected, then the grid file coordinates must also be projected with the same measurement units (feet or meters). The lower-left and upper-right coordinates are those from a grid which covers the geographical area. A user should identify these with a GIS program or from a properly indexed map.

2. The user selects whether the grid is to be created by cell spacing or by the number of columns.

With *By cell spacing*, the size of the cell is defined by its horizontal width, in the same units as the measurement units of the primary file. This would be used to maintain a certain size of spacing for a cell. For example, if the coordinate system is spherical and the lower-left coordinates are -76.90 and 39.20 degrees and the upper-right coordinates are -76.32 and 39.73 degrees (a grid which overlaps Baltimore City and Baltimore County), then the horizontal distance - the difference in the two longitudes (0.58 degrees) must be divided into appropriate sized intervals. At this latitude, the difference in longitudes is 34.02 miles. If a user wanted cell spacing of 0.01 degrees, then this would be entered and *CrimeStat* will calculate 59 columns (cells) in the horizontal direction, one for each interval of 0.01 and one for the fractional remainder. If the coordinate system is projected, then similar calculations would be made using the projected units (feet or meters).

Figure 3.9:


Grid Cell Structure for Baltimore Region
100 Columns by 90 Rows
Upper right coordinate
Lower left coordinate
baltimore grid
Interstate Highways
Baltimore County
City of Baltimore
Baltimore County
City of Baltimore
0
5
10
20 Miles
N
W
E
S

Probably an easier way to specify the grid is to indicate the number of columns. By checking *By number of columns*, the user defines the number of columns to be calculated. *CrimeStat* will automatically calculate the cell spacing needed and will calculate the required number of rows. For example, using the same coordinates as above, if a user wanted half mile squares for the cells, then they would need approximately 68 cells in the horizontal direction since 34.02 miles divided by 0.5 mile squares equals about 68 cells. Figure 3.10 shows a correctly defined reference file where *CrimeStat* creates the reference grid with the number of columns being defined; in the example, 100 columns are requested.

Saving a Reference File

The user can save the lower-left and upper-right coordinates of a defined reference grid and the number of columns. Type **S**ave <filename>. The coordinates and column sizes will be saved in the system registry. To load an already defined reference file, type **L**oad and then check the appropriate filename, followed by clicking on 'Load'.

In addition, the user can save the reference parameters to an external file. To do this, it has to be already saved in the system registry. Type **L**oad and then check the appropriate filename, followed by clicking on 'Save to File". Define the directory and file name and click 'Save'. The file will be saved with an 'ref' extension (e.g., BaltimoreCounty.ref).

External Grid File

Many GIS programs can create uniform grids that cover a geographical area. As with the primary and secondary files, these need to be converted to either '.dbf', ASCII, or '.shp' format. To use an existing grid file created in a GIS or another program, the user clicks on *From File* on the Reference File tab and selects the file.

There are three characteristics that should be identified for an existing grid file:

1. The name of the file. The user selects the file from a dialog box similar to the primary file.

2. If the existing reference file is a true grid, the *True Grid* box should be checked.

3. If the reference file is a true grid, the number of columns should be entered. *CrimeStat* will automatically count the number of records in the file and place it in the *Cells* box. When the number of columns is entered, *CrimeStat* will automatically calculate the number of rows.

Figure 3.10:

Create Reference Grid Setup

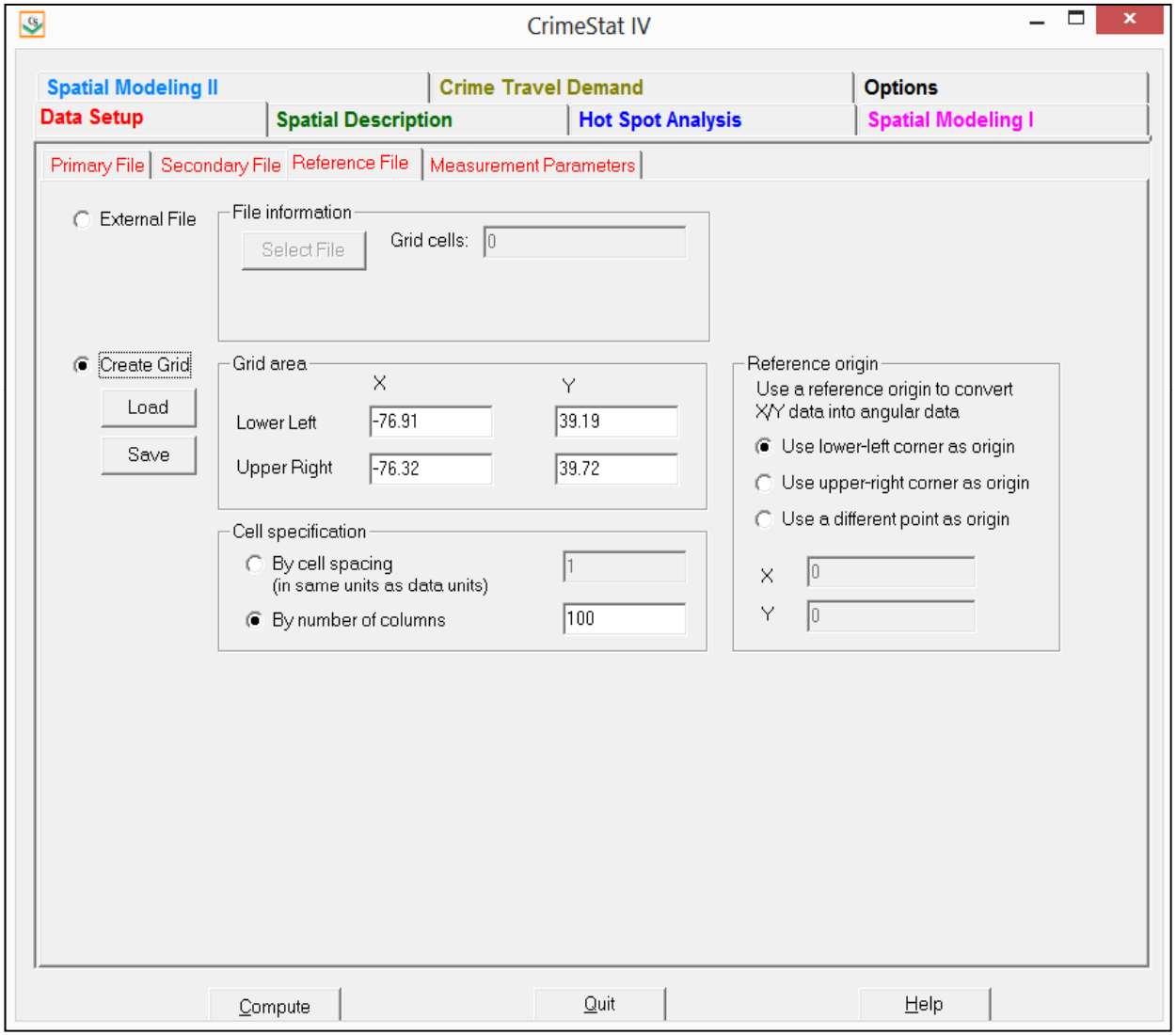


Figure 3.11 shows a correctly defined reference file using an existing grid file. One must be careful in using a file which is not a grid. *CrimeStat* can output the results of the interpolation routines in several GIS formats - *Surfer for Windows, ArcGIS Spatial Analyst®, ArcGIS®, MapInfo®* and several Ascii formats. Of these, only the output to *Surfer for Windows®* will allow the reference to be a shape other than a true grid. For the interpolation outputs of *ArcGIS Spatial Analyst®, ArcGIS®, MapInfo®* and the Ascii formats, the reference file must be a true grid.

Use of Reference File

A reference grid can be very useful. First, a number of the routines use it for either interpolation (single and dual kernel routines; nearest neighbor hierarchical clustering routine; journey-to-crime; Bayesian journey-to-crime) or keying a search radius (STAC). Second, a grid produced by *CrimeStat* can be used as a separate layer in a GIS program in order to reference other data that is displayed, aside from statistical calculations. Historically, many map uses are referenced to a grid in order to produce a systematic inventory (e.g., parcel maps; tax assessor maps; U.S. Geological Survey 7.5" 'quad' maps). In short, it is a routine with multiple purposes.

Measurement Parameters

The final properties that complete data definition are the measurement parameters. On the Measurement Parameters tab, the user defines the geographical area and the length of street network for the study area, and indicates whether direct, indirect or network distance is to be used for calculations. Figure 3.12 shows the measurement parameters tab page.

Area of Study Region

In calculating distances between points for two of the statistics - the nearest neighbor index (Nna) and the Ripley 'K' index (RipleyK), and for using the nearest neighbor hierarchical clustering (Nnh) routine, the STAC routine, or the zonal nearest neighbor hierarchical clustering (Znnh) routine, the area for which the points fall within needs to be defined (the study area). The user indicates the area of the geographical coverage and the measurement units that distances are calculated (feet, meters, miles, nautical miles, kilometers). Unlike the data units for the coordinate system, which must be consistent, CrimeStat can calculate distances in any of these units. In some cases, analysis will be conducted on a subset of the study area, rather than the entire area. For each analysis, the user should identify the area of the subset for which distance statistics are to be calculated.

Figure 3.11:

Reference File Definition With An External File

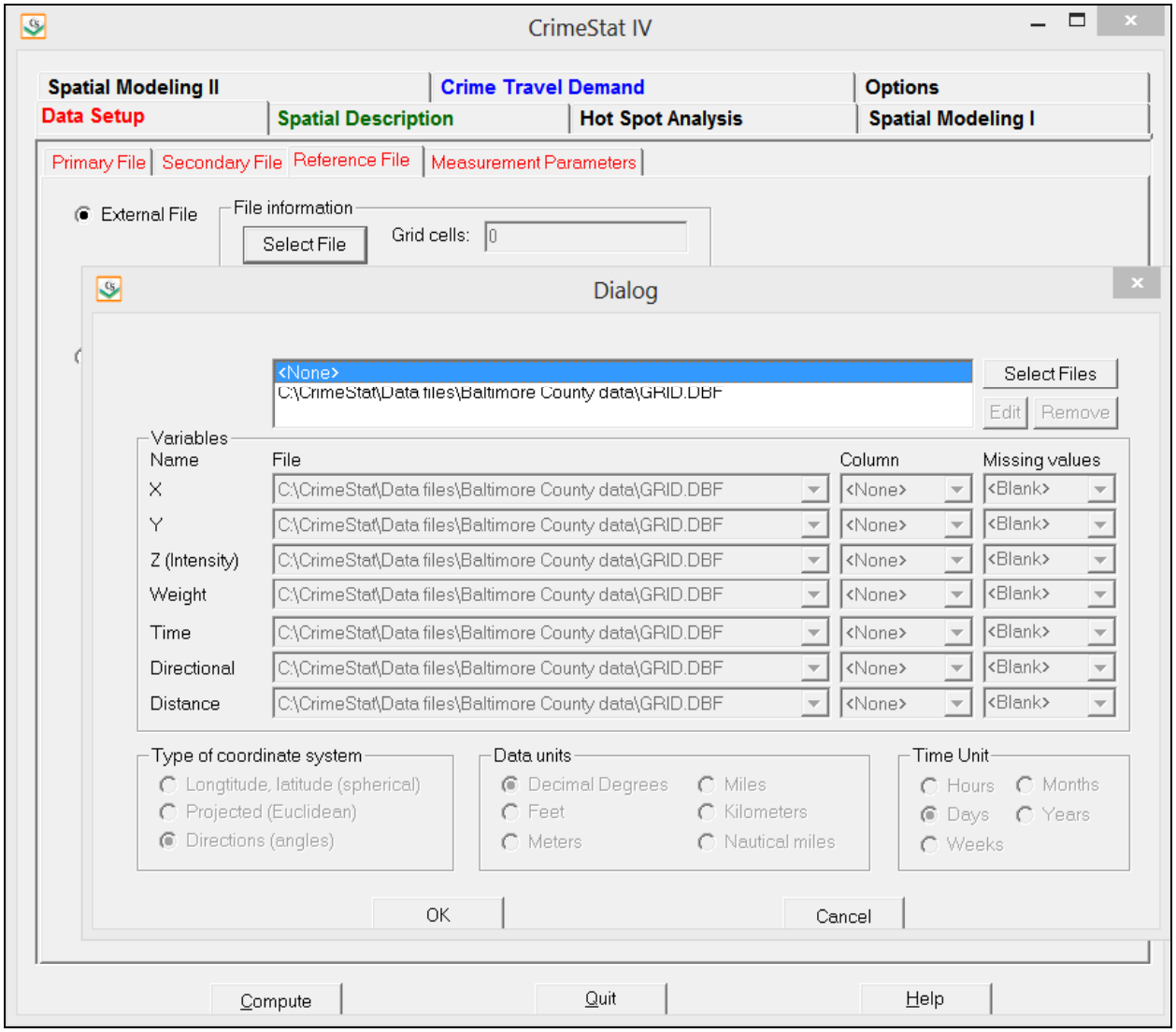
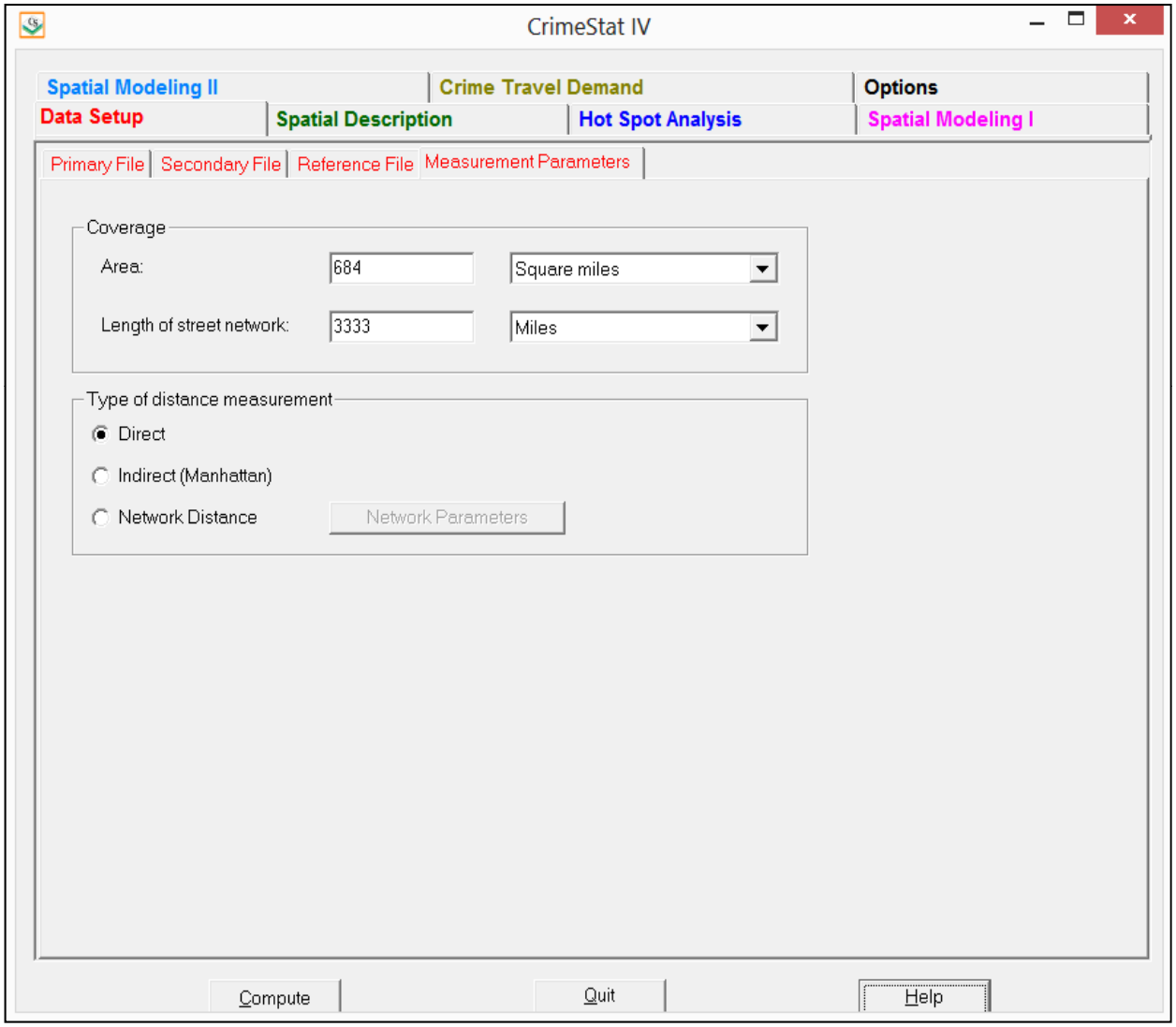


Figure 3.12:

Measurement Parameters Page



Length of Street Network

In addition, the linear nearest neighbor statistic uses the total length of the street network as a baseline for comparison (see Chapter 6). If this statistic is to be used, the total length of the street network should be defined. Most GIS programs can sum the total length of the street network. Again, if subsets of the study are used, the user should indicate the appropriate length of street network for the subset so that the comparison is appropriate.

Type of Distance Measurement

Direct distance

CrimeStat can calculate distance in three different ways: direct, indirect, and network distances. Direct distance is the shortest distance between two points. On a flat plane with a projected coordinate system, the shortest distance between two points is a straight line. However, with a spherical coordinate system, the shortest distance between two points is a Great Circle line. Depending on the coordinate system used, CrimeStat will calculate Great Circle distances using spherical geometry for spherical coordinates and Euclidean distances for projected coordinates. The drawings in Figure 3.13 illustrate direct distance with a projected and spherical coordinate system. The shortest distance between point A and point B is either a straight line (projected) or a Great Circle (spherical). For details see McDonnell, 1979 (chapter 1) or Snyder, 1987 (pp. 29-33).

Indirect distance

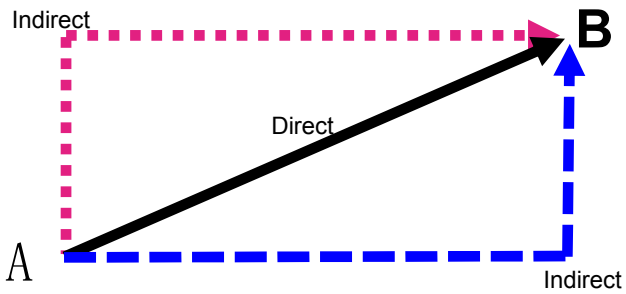
Indirect distance is an approximation to travel on a rectangular road network. This is frequently called *Manhattan* distance, referring to the grid-like structure of Manhattan. Many cities, but certainly not all, lay out their streets in grids. The degree to which this is true varies. Older cities will not usually have grid structures whereas newer cities tend to use grid layouts more. Of course, no real city is a perfect grid, though some come close (e.g., Salt Lake City). Distance measured over a street network is always longer than a direct line or arc. In a perfect grid, travel can only occur in a horizontal or vertical direction so that the distance traveled is the sum of the horizontal and vertical street lengths that have been traversed (i.e., one cannot cut diagonally across a block). Distance is measured as the sum of horizontal and vertical distances traveled between two points.

Indirect distance approximates actual travel in a city where streets are arranged in grid pattern. In this case, indirect distance would be a more appropriate distance measurement than

Figure 3.13:

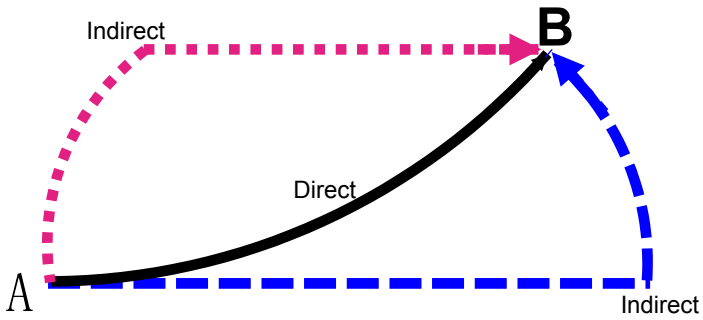
Direct and Indirect Distances

Two-dimensional Projected Geometry: Euclidean distance



A-B distance ('dotted route')
=
A-B distance('dashed route')

Three-dimensional Spherical Geometry: Great Circle distance



A-B distance ('dotted route')
<
A-B distance ('dashed route')

direct distance. Also, there is a linear nearest neighbor index that measures the distribution of point locations in relation to the street network rather than the geographical area and uses indirect distance. This will be discussed in Chapter 6. In this case, the use of indirect distance would be preferable than direct distance (see endnote *iii*).[6]

Network distance

Network distance is travel on an actual network. The network can be roads, a transit system, rail lines, or even bicycle paths. Travel is constrained to the network which usually will make it longer than direct distance measurement. However, the advantage is that travel is measured along the available routes rather than as an abstract 'straight line' or 'grid'. Another advantage of network distance is that the network can be weighted by travel time, travel speed or travel cost. Thus, it is possible to measure approximate travel time or travel cost through the network and not just distance. It is generally recognized that travel time is a more realistic dimension than distance since it will vary by time of day. For example, it generally takes a lot longer to travel any distance in an urban area during the peak evening 'rush hours' (4-7 PM) than at, say, 3 AM in the morning. Distance is always invariant whereas travel time varies.

An even more realistic dimension is travel cost. Trips over a metropolitan area are governed by a number of variables aside from travel time - vehicle operating costs, parking costs and, even, likely risks (e.g., likelihood of being caught). For an offender who is traveling, those other cost factors may be as important as the actual time it takes in determining whether to make a crime trip. In Chapter 29, there is a discussion of travel costs in the context of travel decisions.

---

[6] With a projected coordinate system, indirect distances can be measured by perpendicular horizontal or vertical lines on a flat plane because all direct paths between two points have equal distances. For example in Figure 3.13, whether the distance is measured from point A north to the Y-coordinate of point B and then eastward until point B is reached or, alternatively, from point A eastward to the X-coordinate of point B, then northward until point B is reached, the distances will be the same. One of the advantages of a Manhattan geometry is that travel distances that are pointed towards the final direction) are equal.

With a spherical coordinate system, however, Manhattan distances are not equal with different routes. Because the distance between two points at the same latitude decreases with increasing latitude (north or south) from the equator, the path between two points will differ on the route with Manhattan rules. In Figure 3.13, for example, it is a longer distance to travel from point A eastward to the longitude of point B, before traveling north to point B than to travel northward from point A to the same latitude as point B before traveling eastward to point B. Consequently, *CrimeStat* modifies the Manhattan rules for a spherical coordinate system by calculating both routes between two points and averaging them. This is called a *Modified Spherical Manhattan Distance.*

There are two major disadvantages in using network distance, however. First, there are errors in networks. For example, a network may not have incorporated all new roads or converted roads. Thus, the network algorithm will not choose a particular route when, in fact, it actually exists and people use it. It is critical that networks be updated to ensure accuracy. See Chapter 26 for a discussion of network errors and the need to thoroughly clean them.

Second, it can take a long time to calculate distance along a network. The shortest path algorithm that is used must explore many alternative routines, a time consuming process. For simple statistics, this is not liable to be a problem. But, for some of the more complicated matrix operations (e.g., the distance from every point to every other point), calculation time increases exponentially with the number of cases. For any complex calculation, it becomes impractical to have to wait a long time just for a little extra precision. In short, it may not be worth the trouble.

Distance Calculations

Distances in CrimeStat are calculated with the following formulas:

Direct, Projected Coordinate System

Distance is measured as the hypotenuse of a right triangle in Euclidean geometry:

$$d_{AB} = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2} \quad (3.1)$$

where d_{AB} is the distance between two points, A and B, X_A and X_B are the X-coordinates for points A and B in a projected coordinate system, Y_A and Y_B are the Y-coordinates for points A and B in a projected coordinate system.

Direct, Spherical Coordinate System

Distance is measured as the Great Circle distance between two points. All latitudes (φ) and longitudes (λ) are first converted into radians using:

$$Radians\ for\ latitude(\varphi) = \frac{2\pi\varphi}{360} \quad (3.2)$$

$$Radians\ for\ longitude(\lambda) = \frac{2\pi\lambda}{360} \quad (3.3)$$

Then, the distance between the two points is determined from:

$$d_{AB} = 2Arcsin\{Sin^2\left[\frac{(\varphi_B - \varphi_A)}{2}\right] + Cos\varphi_A Cos\varphi_B Sin^2[\frac{(\lambda_B - \lambda_A)}{2}]^{\frac{1}{2}}\} \quad (3.4)$$

with all angles being defined in radians where d_{AB} is the distance between two points, A and B, φ_A and φ_B are the latitudes of points A and B, and λ_A and λ_B are the longitudes of points A and B (Snyder, 1987, p. 30, 5-3a).

Indirect, Projected Coordinate System

Distance is measured as the sides of a right triangle using Euclidean geometry. For each segment:

$$d_{AB} = |X_A - X_B| + |Y_A - Y_B| \quad (3.5)$$

where d_{AB} is the distance between two points, A and B, X_A and X_B are the X-coordinates for points A and B in a projected coordinate system, and Y_A and Y_B are the Y-coordinates for points A and B in a projected coordinate system. Note the absolute value of the difference is taken for each term. Then, the total distance is the sum of the distance of individual segments.

Indirect, Spherical Coordinate System

Distance is measured by the average of summed Great Circle distances of two routes, one in the east-west direction followed by a north-south direction and the other in the north-south direction followed by an east-west direction:

$$d_{AB} = \frac{d1_{AB} + d2_{AB}}{2} \quad (3.6)$$

where d_{AB} is the distance between two points, A and B, $d1_{AB}$ is the distance from the two points traversing initially from an east-west direction and then from a north-south direction, and $d2_{AB}$ is the distance from the two point traversing initially from a north-south direction and then from an east-west direction. Because of the curvature of the earth, the two distances - $d1_{AB}$ and $d2_{AB,}$ will not be the same. The average of the two is taken as indirect, spherical distance.

Network Distance

Network distance is calculated with a shortest path algorithm. Chapters 26 and 30 provide more information on networks and how distance is calculated on them. A short summary will be

given here. In general, distance is calculated by a shortest path algorithm. In a *shortest path* for a single trip (from a single origin to a single destination), the route with the lowest overall *impedance* is selected. Impedance can be defined in terms of distance, travel time, speed, or generalized cost.

There are a number of shortest path algorithms that have been developed (Sedgewick, 2002). They differ in terms of whether they are breadth-first (i.e., search all possibilities) or depth-first (i.e., go straight to the target) algorithms and whether they examine a one-to-many relationship (i.e., from a single origin node to many nodes) or a many-to-many relationship (all pairs from each node to every other node).

The algorithm that is most commonly used for shortest path analysis of moderate-sized data sets (up to a million cases) is called *A**, which is pronounced "A-star" (Nilsson, 1980; Stout, 2000; Rabin 2000a, 2000b; Sedgewick, 2002). It is a one-to-many algorithm but is an improvement over another commonly-used algorithm called *Dijkstra* (Dijkstra, 1959). Therefore, I will start first by describing the Dijkstra algorithm before explaining the A* algorithm.

Dijkstra algorithm

The Dijkstra algorithm is a one-to-many search strategy in which a shortest path from a single node to all other nodes is calculated. The routine is a breadth-first algorithm in that it searches all possible paths, but it builds the path one segment at a time. Starting from an origin location (node), it identifies the node that is nearest to it **and** which has not already been identified on the shortest path. After each node has been identified to be on the shortest path, it is removed from the search possibilities. The algorithm proceeds until the shortest path to all nodes has been determined.

The algorithm can also be structured to find the shortest path between a particular origin node and a particular destination node. In this case, it will quit once the destination node has been identified on the shortest path. The algorithm can also be structured to find the shortest path from each origin node to each destination node. It does this one path at a time (e.g., it finds the shortest path from node A to all other nodes; then it finds the shortest path from node B to all other nodes; and so forth).

A Algorithm*

The biggest problem with the Dijkstra algorithm is that it searches the path to every single node. If the purpose were to find the shortest path from a single node to all other nodes, then this would produce the best solution. However, with a matrix of distances from one set of

points to another set of points (an origin-destination matrix), we really want to know the distance between a pair of nodes (one origin and one destination). Consequently, the Dikjstra algorithm is very, very slow compared to what we need. It would be a lot quicker if we could find the distance from each origin-destination pair one at a time, but quit the algorithm as soon as that distance has been determined.

This is where the A* algorithm comes in. A* was developed within the artificial intelligence research area as a means for developing a *heuristic* rule for solving a problem (Nilsson, 1980). In this case, the heuristic rule is the remaining distance from a solved node to the final destination. That is, at every step in the Dijkstra routine, an estimate is made of the remaining distance from each possible choice to the final destination. The node that is chosen for the shortest path is that which has the least total *combined* distance from the previously determined node to the final goal. Thus, for any step, if d_{i1} is the distance to a node, i, which has not already been put on the shortest path and d_{i2} is an estimate of the distance from that node to the final destination, the estimated total distance for that node is:

$$d_i = d_{i1} + d_{i2} \tag{3.7}$$

Of all the nodes that could be chosen, the node, i, which has the shortest total distance is selected next for the shortest path. There are two caveats to this statement. First, the node, i, cannot have already been selected for the shortest path; this is just re-stating the rules by which we search for nodes that have not yet been put on the shortest path list. Second, the estimate of the remaining distance to the final destination must be less than or equal to the actual distance to the final destination. In other words, the estimated distance, d_{i2}, cannot be an overestimate (Nilsson, 1980). However, the closer the estimated distance is to the real distance, the more efficient will be the search.

How then do we determine a reasonable estimate for d_{i2}? The answer is a straight line from the possible node to the final destination since the shortest distance between two points is a straight line (or, on a sphere, a Great Circle distance since the shortest distance between two points is an arc). If we simply calculate the straight-line from the node that we are exploring to the final node, then the heuristic will work. The effect of this simplifying heuristic is to cut down substantially on the number of nodes that have to be searched. As with the Dijkstra algorithm, A* can be applied to multiple origins. It does it one origin-destination combination at a time.

As mentioned, Chapters 26 and 30 discuss in more detail networks and how shortest path is calculated in them.

Saving Parameters

All data setup parameters can be saved. In the Options tab, there is a 'Save parameters' button. The parameter file must be saved with a 'param' extension. To re-load a saved parameters file, use the 'Load parameters' button.

Statistical Routines and Output

Statistical routines are selected from five groupings of statistics:

4. Spatial Description
5. Hot Spot Analysis
6. Spatial Modeling I
7. Spatial Modeling II
8. Crime Travel Demand

The user selects the routines and inputs any parameters, if required. Clicking on the 'Compute' button will run all the routines that have been selected. Since *CrimeStat* is multi-threaded, different routines run in separate threads and may finish at different times. When a routine is finished, a 'Finished' message will be displayed at the bottom of the screen.

Virtually all the routines output to either GIS packages or to standard 'dbf' files which can be read by spreadsheet, data base, and graphics programs. While each output table can be printed as an Ascii file to a printer, it is recommended that the user output the results in 'dbf' and read it into a program that has better output capabilities. For example, the nearest neighbor and Ripley's K routines output columns can be saved as standard 'dbf' files which can be read by spreadsheet programs such as Excel® or Lotus 1-2-3®. The spreadsheet data, in turn, can be imported into most graphics programs, such as PowerPoint® or Freelance Graphics®, for creating better quality graphics. For 'cut-and-paste' operations, user can copy portions of the output tables and paste them into word processing programs. One should see *CrimeStat* as a collection of specialized statistical routines that can produce output for other programs, rather than as a full-blown package.

A Tutorial with a Sample Data Set

Let us run through the data setup and running of several routines with one of the sample data sets that were provided (SampleData.zip). Unzipping this file reveals two files called *Incident.dbf* and *BaltPop.dbf*. The incident file is a collection of incident locations that have

been randomly simulated while the other file includes the 1990 population of census block groups in the Baltimore region.[7] Both files have locations coded in spherical (longitude-latitude) coordinates. The X/Y coordinates for the incident file is the location where the incident (crime) occurred. The X/Y coordinates for the block groups is the centroid location.

1. Start the *CrimeStat* program by either double-clicking on the *CrimeStat* icon on the desktop (if installed) or else opening Windows Explorer and locating the directory where *CrimeStat* is stored and double-clicking on the file called *crimestat.exe*.

2. Once the program splash page closes, the user will be looking at the **Data Setup** page with the Primary File page open.

3. Click on 'Select Files' followed by 'Browse'. Locate the file called Incident.dbf and click on 'Open' followed by 'OK'.

4. The file name will now be listed for the X, Y, Z (intensity), Weight, and Time fields. This variable, however, only has three fields - ID, Lon, Lat, indicating an record number, the longitude and latitude of the incident location.

5. Identify the appropriate fields under the Column heading by clicking on the cell and scrolling down to the appropriate name. For the X variable, the relevant name is Lon and for the Y variable, the relevant name is Lat (i.e., that is the names used for coordinates in this file. However, the variables will not always be simply named). For this example, there are no intensity, weight or time variables.

6. Under Type of Coordinate System, be sure that 'Longitude/latitude (spherical)' is checked since this data set use spherical coordinates.

7. Because the coordinate system is spherical, the data units are automatically decimal degrees. If they were projected, one would have to choose the particular units - feet, meters, miles, kilometers, or nautical miles. This finishes the setup for the primary file.

8. Next, Click on the Secondary File tab.

[7] Note: the incident locations have had random coordinates assigned so this file should not be used for research.

9. Again, click on select files, locate and open the *BaltPop.dbf* file. This is a file of census block groups. You are going to treat each block group as a '*pseudo-point*', that is, as a single point which represents the block group. That point is the centroid of the block group. The population will be treated as residing exactly at that point.[8]

10. Once loaded, this file has six variables: Blockgroup, lon, lat, area, density, and Totpop.

11. Define the particular variables. For this file, the X variable is Lon and the Y variable is Lat. Also, define a Z (intensity) variable with Totpop. Note, that you could also assign this name to the Weight variable. Whether the population variable is assigned to the Intensity or Weight variable does not matter to the calculation. However, do not assign this name to both the intensity and the weight (i.e., only use one). This finishes the setup for the secondary variable.

12. Click on the Reference File tab. For these data, you will define a rectangle that covers the study area by identifying the X and Y coordinates for the lower-left corner of the rectangle and the upper-right corner of the rectangles. The following coordinates will work (Table 3.1):

Table 3.1:
Coordinates for Corners of Sample Data Set

	X	Y
Lower-left corner	-76.91	39.19
Upper-right corner	-76.32	39.72

[8] The population does not live at the centroid, of course, unless the block group is a single building. But by treating the block group as a pseudo-point, we can analyze the population (or any other characteristic of the block group).

13. You will also need to tell the program how many columns you want it to calculate. The default value of 100 is fine. If you want it finer, type in a larger number. If you want it cruder, type in a smaller number. This finishes the Reference File setup.

14. Clock on the Measurement Parameters tab. There are three parameters that have to be defined.

    A. For many routines, an area estimate is needed. For this sample set, 684 square miles works.

    B. For the linear nearest neighbor statistic only, the program needs the total length of the street network. In this data, the total street length of the Tiger Files for Baltimore City and Baltimore County is 4868.9 miles.

    C. Finally, the type of distance measurement has to be defined, direct or indirect. For this example, use direct measurement.

15. The data setup is now finished. If you want to re-use this data setup, click on the Options page and 'Save parameters'. Define a file name and be sure to give it a 'param' extension (e.g., SampleData.param). The next time you want to run this data set, all you'll need to do is click on the **Options** page, click on 'Load parameters', and click on the name of the parameters file that you saved.

16. You are now ready to run some statistics. For this example, you will run only four statistics.

17. First, click on the **Spatial Description** page and then click on the Spatial Distribution tab.

    A. Check the Mean center and standard distance (Mcsd) box. Then, click on the 'Save result to' button and identify which GIS program you are writing to (ArcGIS® 'shp'; several Ascii formats; MapInfo® 'MIF) and give it a name (e.g., SampleData).

    B. Also, check the Standard deviational ellipse (Sde) box and, similarly, choose a file output with a name. You can use the same name (e.g., SampleData). *CrimeStat* will assign a ***unique*** prefix to each graphical object.

18. Second, click on the **Hot Spot Analysis** tab followed by the 'Hot Spot' Analysis I sub-heading. Then, check the Nearest Neighbor Hierarchical Clustering (Nnh) box. For this example, keep the default search radius, minimum points per cluster, and number of standard deviations for the ellipses. Also, click on 'Save ellipses to', select a GIS file output, and give it a name. Again, you can use the same name as with the other statistics.

19. Third, click on the **Spatial Modeling I** page and then the 'Interpolation I' tab. Check the dual kernel density interpolation box. This routine will interpolate the incident distribution (primary file) relative to the population distribution (secondary file). For this example, keep the default kernel parameters (these are explained in more detail in Chapter 10). Because the secondary variable is weighted by population (defined as the 'Intensity' variable) be sure to check the 'Use intensity' variable box towards the bottom of the page. This ensures that the dual kernel routine will interpolate the population variable that you assigned when you set up the secondary file.

20. You are now ready to run the statistics. Click on the 'Compute' button. The routine will run until all four routines that you selected are finished; the time will depend on the speed of your computer.

21. Each of the outputs is displayed on a separate results tab. You can print any of these results by clicking on 'Save to text file' (one at a time).

22. You can also display the graphical objects created by the routine in your GIS. Click on 'Close' to close the results window. Then, bring up your GIS and find the objects created by this run. There will be a number of graphical objects associated with the mean center routine (having prefixes of Mc, Xyd, Sdd, Gm, and Hm; see Chapter 4 for details). There will be two graphical objects associated with the nearest neighbor clustering routine (with prefixes of Nnh1 and Nnh2). Finally, there will be a grid object created by the dual kernel routine with a Dk prefix. You can load these objects into a GIS and display them along with the data file. For the dual kernel grid, you will need to graph the variable called "Z" to see the pattern.

23. For example, Figure 3.14 shows an *ArcGIS*® map of 1996 vehicle thefts in Baltimore City and Baltimore County along with the standard deviational ellipse of the vehicle thefts, calculated with *CrimeStat*. *CrimeStat* outputs the ellipse as a shape file, which is then brought directly into *ArcGIS*®. A similar output could

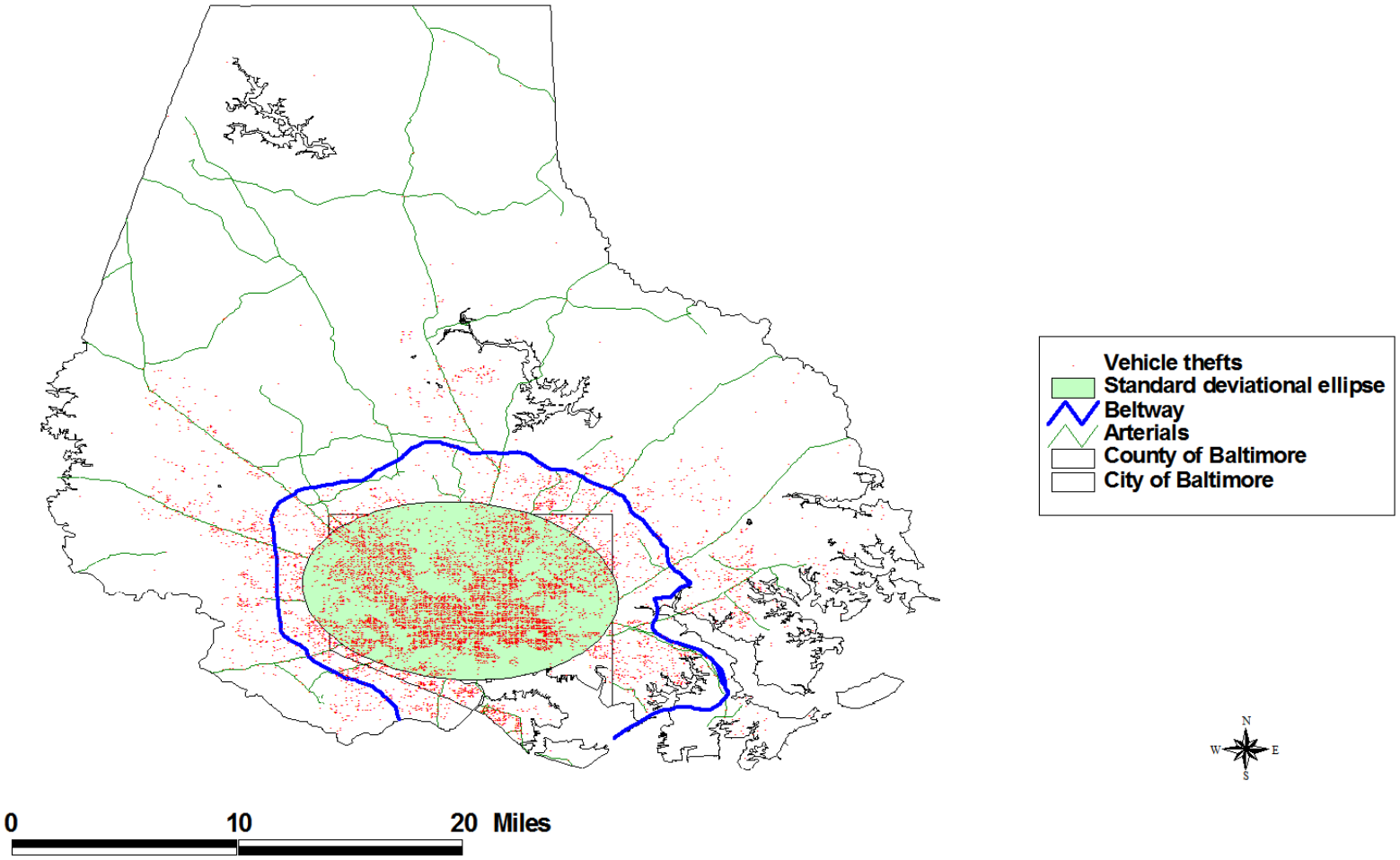
have been done for *MapInfo®*. Most of the statistics in *CrimeStat* have similar visual representations that can be displayed in a GIS program.

24. When you are finished with *CrimeStat*, click on 'Quit' to exit the program.

This finishes the quick tutorial. *CrimeStat* is very easy to set up and to run. In the next chapters, the focus will be on the statistics in the program starting with the analysis of spatial distributions.

Figure 3.14:

Baltimore Vehicle Thefts: 1996
Location of Incidents and Standard Deviational Ellipse



References

Committee on Map Projections (1986). *Which Map is Best*, American Congress on Surveying and Mapping, Falls Church, VA., 1986.

Dijkstra, E. W. (1959). A note on two problems in connection with graphs, *Numerische Mathematik*, 1, 269-271.

Greenhood, D. (1964). *Mapping*. The University of Chicago Press: Chicago.

Levine, N. & Wachs, M. (1986a). Bus Crime in Los Angeles: I - Measuring The Incidence. *Transportation Research*. 20 (4), 273-284.

Levine, N. & Wachs, M. (1986b). Bus Crime in Los Angeles: II - Victims and Public Impact. *Transportation Research*. 20 (4), 285-293.

McDonnell, P. W. Jr. (1979). *Introduction to Map Projections.* New York: Marcel Dekker, Inc.

Maling, D. H. (1973). *Coordinate Systems and Map Projections* (1973). George Philip & Sons, London.

Nilsson, N. J. (1980). *Principles of Artificial Intelligence*. Morgan Kaufmann Publishers, Inc.: Los Altos, CA.

Rabin, S. (2000a). A* aesthetic optimizations. In DeLoura, M., *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 264-271.

Rabin, S. (2000b). A* speed optimizations. In DeLoura, M., *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 272-287.

Robinson, A. H., Sale, R. D., Morrison, J. L. & Muehrcke, P. C. (1984). *Elements of Cartography* (5th edition). J. Wiley & Sons: New York.

Sedgewick, R. (2002). *Algorithms in C++: Part 5 Graph Algorithms* (3rd edition). Addison-Wesley: Boston.

Snyder, J. P. (1987). *Map Projections - A Working Manual*. U.S. Geological Survey Professional Paper 1395. U. S. Government Printing Office: Washington, DC.

References (continued)

Snyder, J. P. & Voxland, P. M. (1989). *An Album of Map Projections*. U.S. Geological Survey Professional Paper 1453. U. S. Government Printing Office: Washington, DC.

Stout, B. (2000). The basics of A* for path planning. In DeLoura, M.. *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 254-263.

Endnotes

i. Some *MapInfo* users in Europe have found difficulty in directly reading MIF/MID files from *CrimeStat* and converting them to the particular national coordinate system (e.g., British National Grid, French National Geographic Institute). For example, in the United Kingdom, Pete Jones of the North Wales Police Department has developed a way around this problem. He writes

"To save the result as a *MapInfo* (.mif) format the following is required:

MIF Options
Name of Projection: Earth Projection
Projection Number: 8
Datum Number 79

Before importing the .mif table into *MapInfo* you need to edit it. Open the .mif file with a text editor. You need to change the following line:

CoordSys Earth Projection 8, 79

Change it to:

CoordSys Earth Projection 8, 79, 7, -2, 49, 0.9996012717, 400000, -100000

Now save the .mif file. You can now import the file into *MapInfo*."

In France, J. Marc Zaninetti of the University of Orléans figured out how to import graphical objects into *MapInfo* using the French coordinate system. He writes

"First convert with *MapInfo* your map to the international European Latitude/Longitude ED87 projection system.

Second, produce the X and Y coordinates and export the data table in Dbase.

Third, with *CrimeStat II*, modify the Save Output parameter in order to change the origin of the projection. By default, the MIF Options are the following:

Name of projection:	Earth projection
Projection number:	1 (Latitude longitude)
Datum number:	33 (international GRS80 origin 0°E, 0°N)

The European norm ED87 has the Datum number 108, so you have to change only this parameter. The new options are the following :

Name of projection:	Earth projection
Projection number:	1 (Latitude longitude)
Datum number:	108 (European data ED87).

Finally, you can now import the MIF output tables directly into your *MapInfo* maps."

iv. Because the Earth is curved, any two dimensional representation produces distortion. The spherical latitude/longitude system (called 'lat/lon' for short) is a universal coordinate system. It is universal because it utilizes the spherical nature of the Earth and each location has a unique set of coordinates. Most other coordinate systems are projected because they are portrayed on a two-dimensional flat plane. Strictly speaking, spherical coordinates - longitudes and latitudes, are not X and Y coordinates since the world is round. However, by convention, they are often referred to as X and Y coordinates, particularly if a small section of the Earth is projected on a flat plane (a computer screen or a printed map).

Projections differ in how they 'flatten' or *project* a sphere onto a two dimensional plane. Typically, there are four properties of maps which cannot all be maintained in any two dimensional representation:

Shape - maintaining correct shape of a land body

Area - if the space represented on a map covers the same area throughout the map, it is called an equal-area map. The proportionality is maintained.

Distance - the distance between two points is in constant scale (i.e., the scale does not change)

Direction - the direction from a point towards another point is true.

Any projection creates one or more types of distortion and particular projections are chosen in order to have accuracy in one or two of these properties. Different projections portray different types of information. Most projections assume that the Earth is a sphere, a situation that is not completely true. The Earth's diameter at the equator is slightly greater than the distance between the poles (Snyder, 1987). The circumference of the Earth between the Poles is about 24,860 miles on a meridian; the circumference at the Equator is about 75 miles more.

There is an infinite number of projections. However, only a couple dozen have been used in practice (Greenhood, 1964; Snyder, 1987; Snyder & Voxland, 1989). They are based on projections of the sphere onto a cylinder, cone or flat plane. In the United States, several common coordinate systems are used. Theoretically, the projection and the coordinate system can be distinguished (i.e., a particular projection could use one of several coordinate systems, e.g. meters or feet). However, in practice, particular projections use common coordinates. Among the most common in use in the United States are:

A. Mercator - The *Mercator* is an early projection, and one of the most famous, which is used for world maps. The projection is done on a cylinder, which is vertically centered on a meridian, but touching a parallel. The globe is projected on the cylinder as if light is emanating from the center of the globe while the Earth turns. The meridians cut the equator at equal intervals. However, they maintain parallel lines, unlike the globe where they converge at the poles. The longitudes are stretched with increasing latitude (in both north and south directions) up until the 80^{th} parallel. The effect is that shape is approximately correct and direction is true. Distance, however, is distorted. For example, on a Mercator map, Greenland appears as big as the United States, which it is not. Distances can be measured in any units for a Mercator though usually they are measured

in miles or kilometers.

B. Transverse Mercator - If the Mercator is rotated 90^0 so that the cylinder is centered on a parallel, rather than a meridian, it is called a *Transverse Mercator*. The cylinder is projected as being horizontal but is touching a meridian. The Transverse Mercator is divided into narrow north-south zones in order to reduce distortion. The meridian that the cylinder is touching is called the *Central Meridian* of the zone. Distances are accurate within a limited distance from the central meridian. Thus, the boundaries of zones are selected in order to maintain reasonable distance accuracy. In the U.S., many states use the Transverse Mercator as the basis for their state plane coordinate system including Arizona, Hawaii, Illinois, and New York.

C. Universal Transverse Mercator (UTM) - In 1936, the International Union of Geodesy and Geophysics established a standard use of the Transverse Mercator, called the *Universal Transverse Mercator* (or UTM). In order to reduce distortion, the globe is divided into 60 zones, 6 degrees of longitude wide. For latitude, each zone is divided further into strips of 8 degrees latitude, from 84^o N to 80^o S. Within each band, there is a central meridian which, in theory, would be geodetically true. But, to reduce distortion across the area covered by each zone, scale along the central meridian is reduced to 0.9996. This produces two parallel lines of zero distortion approximately 180 km away from the central meridian. Scale at the boundary of the zone is approximately 1.0003 at U.S. latitudes. Coordinates are expressed in meters. By convention, the origin is the lower left corner of the zone. From the origin, *Eastings* are displacements eastward and from the origin, *Northings* are displacements northward. The central meridian is given an Easting of 500,000 meters. The Northing for the equator varies depends on the hemisphere. For the northern hemisphere, the equator has a Northing of 0 meters. For the southern hemisphere, the Equator has a Northing of 10,000,000 meters. The UTM system was adopted by the U.S. Army in 1947 and has been adopted by many national and international mapping agencies. Distances are always measured in meters in UTM.

D. Oblique Mercator - There are a number of cylindrical projections which are neither centered on a meridian (as in the Mercator) or on a parallel (as in the Transverse Mercator). These are called *Oblique Mercator* projections because the cylinder is centered on a line which is oblique to parallels or meridians. In the U.S., the *Hotine Oblique Mercator* is used for Alaska.

E. Lambert Conformal Conic - The *Lambert Conformal Conic* is a projection made on a cone, rather than a cylinder. Lambert's conformal projection centers the cone over a central location (usually the North Pole) and the cone 'cuts' through the globe at parallels chosen to be standards. Within those standards, shapes are true and meridians are straight. Outside those standards, parallels are spaced at increasing intervals the further north or south they go to reduce distance distortion. The projection is the basis of many state plane coordinate systems, including California, Connecticut, Maryland, Michigan, and Virginia.

F. Alber's Equal-Area - Another projection on a cone is the *Albers Equal-Area* except that parallels are spaced at decreasing intervals the further north or south they are placed from the standard parallels. The map is an equal-area projection and scale is true in the east-west direction.

G. State Plane Coordinates - Every state in the United States has an official coordinate system, called

the *State Plane Coordinate System*. Each state is divided into one or more zones and a particular projection is used for each zone. With the exception of Alaska, which uses the Hotine Oblique Mercator for one of its eight zones, all state plane coordinate systems use either the Transverse Mercator or the Lambert Conformal Conic. Each state's shape determines which projection is chosen to represent that state. Typically, states extending in a north-south direction use Transverse Mercator projections while states extending in an east-west direction use Lambert Conformal Conic projections. But, there are exceptions, such as California which uses the Lambert. Projections are chosen to minimize distortion over the state. Several states use both projections (Florida, New York) and Alaska uses all three. Distances are measured in feet.

See Snyder (1987) and Snyder and Voxland (1989) for more details on these and other projections including the mathematical transformations used in the various projections. Other good references are Maling (1973), Robinson, Sale, Morrison and Muehrcke (1984) and the Committee on Map Projections (1986).

iii. With a projected coordinate system, indirect distances can be measured by perpendicular horizontal or vertical lines on a flat plane because all direct paths between two points have equal distances. For example in Figure 3.13, whether the distance is measured from point A north to the Y-coordinate of point B and then eastward until point B is reached or, alternatively, from point A eastward to the X-coordinate of point B, then northward until point B is reached, the distances will be the same. One of the advantages of a Manhattan geometry is that travel distances that are direct (i.e., that are pointed towards the final direction) are equal.

With a spherical coordinate system, however, Manhattan distances are not equal with different routes. Because the distance between two points at the same latitude decreases with increasing latitude (north or south) from the equator, the path between two points will differ on the route with Manhattan rules. In Figure 3.13, for example, it is a longer distance to travel from point A eastward to the longitude of point B, before traveling north to point B than to travel northward from point A to the same latitude as point B before traveling eastward to point B. Consequently, *CrimeStat* modifies the Manhattan rules for a spherical coordinate system by calculating both routes between two points and averaging them. This is called a *Modified Spherical Manhattan Distance.*

Linking *CrimeStat IV* to *MapInfo*®

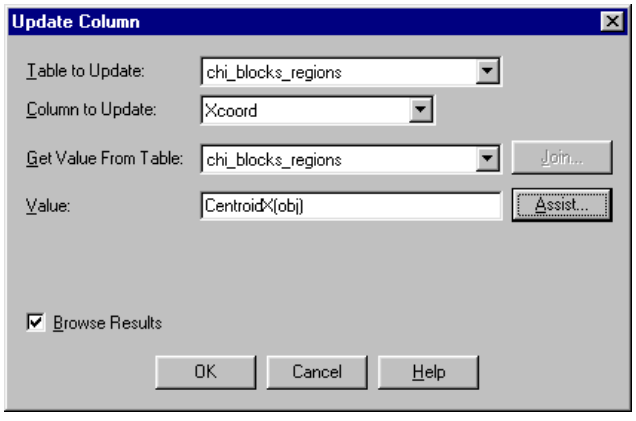
Richard Block
Professor of Sociology and Criminal Justice
Loyola University of Chicago

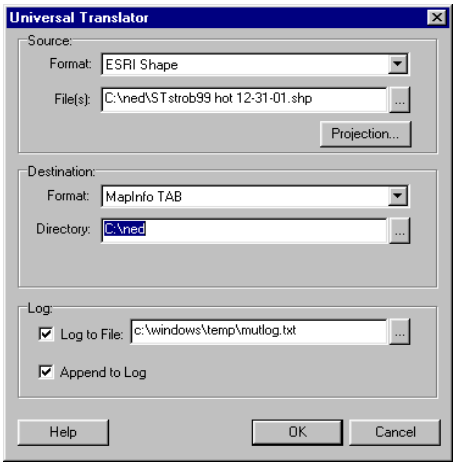
MapInfo® point 'dat' files can be inputted to *CrimeStat* as primary or secondary files. However, x and y coordinates need to be added to the file. If the point data are in latitude/longitude, this is easily done with a free extension, *Table Geography*, available through the Directions Magazine website as part of the KGM utilities at: http://www.directionsmag.com/tools/Default.asp?a=file&ID=11 . Add this extension to your MapInfo toolbox. Click on the tool. You will first be asked for a table to add coordinates. The program automatically adds columns for longitude and latitude.

If you are using another projection, you will need to add and update columns to your file. To do this, add columns for x and y coordinates to your table (Table–>Maintenance–>Table Structure–>Add Field) in an appropriate numeric format for your projection. As shown in left figure, update these new columns with the coordinates (Table–>update column). Choose the data file and column that you want to update. Next, click assist and then functions. Choose *centroidx* to update the horizontal field and *centroidy* to update the vertical field. Within *CrimeStat*, identify the file type as *MapInfo* 'dat'.

For some *CrimeStat* require a reference file. These are identified by the lower-left and upper-right coordinates of a rectangle. To derive these coordinates, make the top map (cosmetic) layer editable. Draw a rectangle identifying the study area. Select the rectangle. Convert it to a region (objects-→ convert to region). Double click on the rectangle, and the appropriate coordinates and area of the rectangle will appear.

Several *CrimeStat* routines output geographic features that can be added as a layer in *MapInfo*. To output these graphics, first designate an output file. If you are working in longitude/latitude, choose a *MapInfo* 'mif 'file as output. In *MapInfo*, import the mif file (Table–>Import), and open the file as a layer in your map. For any other projection, output to an *ESRI* shape file and use the Universal Translator tool (right figure) to import your file (Tools--->Universal Translator). Choose *ESRI* shape and the file that you designated in CrimeStat. Next, choose the appropriate projection. Identify the destination format–chose *MapInfo tab* and, finally, identify the directory for storage of the file. The table can then be opened as a layer on your map. *CrimeStat* graphic output is brought into *MapInfo* as regions and has all the functionality of a regions layer. Figure 7.6 includes STAC and single kernel density output.





CrimeStat IV

Part II: Spatial Description

Chapter 4:

Centrographic Statistics

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 4:
Centrographic Statistics

In this chapter, the spatial distribution of crime incidents will be discussed. The statistics that are used in describing the spatial distribution of crime incidents will be explained and will be illustrated with examples from *CrimeStat® III*. For the examples, crime incident data from Baltimore County and Baltimore City will be used. Figure 4.1 shows the user interface for the spatial distribution statistics in *CrimeStat*. For each of these, the statistics will first be presented followed by examples of their use in crime analysis.

Centrographic Statistics

The most basic type of descriptors for the spatial distribution of crime incidents are *centrographic statistics*. These are indices which estimate basic parameters about the distribution (Lefever, 1926; Furfey, 1927; Bachi, 1957; Neft, 1962, Hultquist, Brown and Holmes, 1971; Ebdon, 1988). They include:

1. Mean center
2. Median center
3. Center of minimum distance
4. Standard deviation of X and Y coordinates
5. Standard distance deviation
6. Standard deviational ellipse

They are called centrographic in that they are two dimensional correlates to the basic statistical moments of a single-variable distribution - mean, standard deviation, skewness, and kurtosis (see Bachi, 1957). They have been applied to crime analysis by Stephenson (1980) and by Langworthy and Jefferis (1998). Because two dimensions add complexity not seen in one dimension, these statistical moments have been modified to be appropriate. Figure 4.2 shows how the centrographic statistics are selected in *CrimeStat.*

Mean Center

The simplest descriptor of a distribution is the ***mean center***. This is merely the mean of the X and Y coordinates. It is sometimes called a *center of gravity* in that it represents the point in a distribution where all other points are balanced if they existed on a plane and the mean center was a fulcrum (Ebdon, 1988; Burt and Barber, 1996).

Figure 4.1:

Spatial Distribution Screen

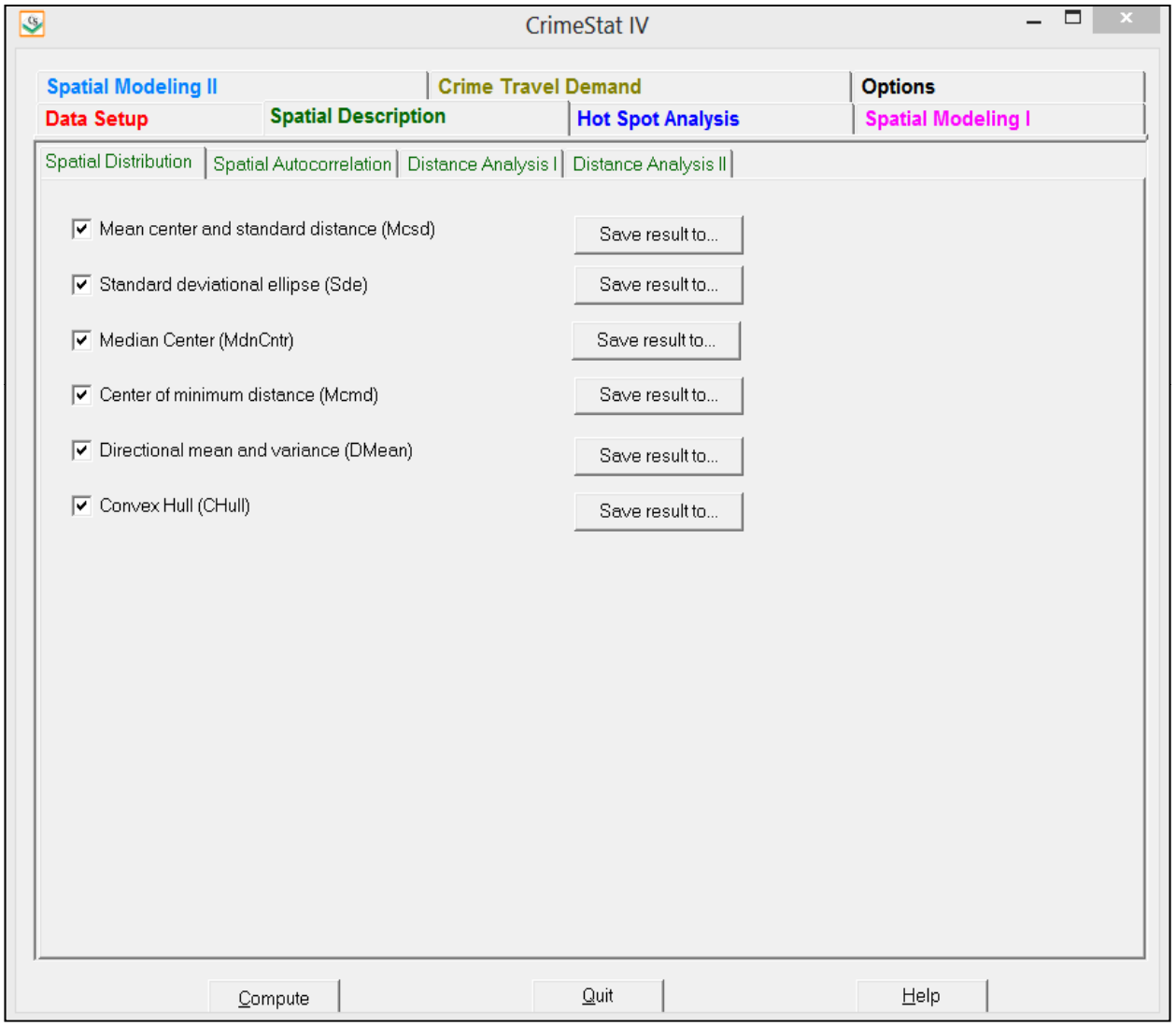
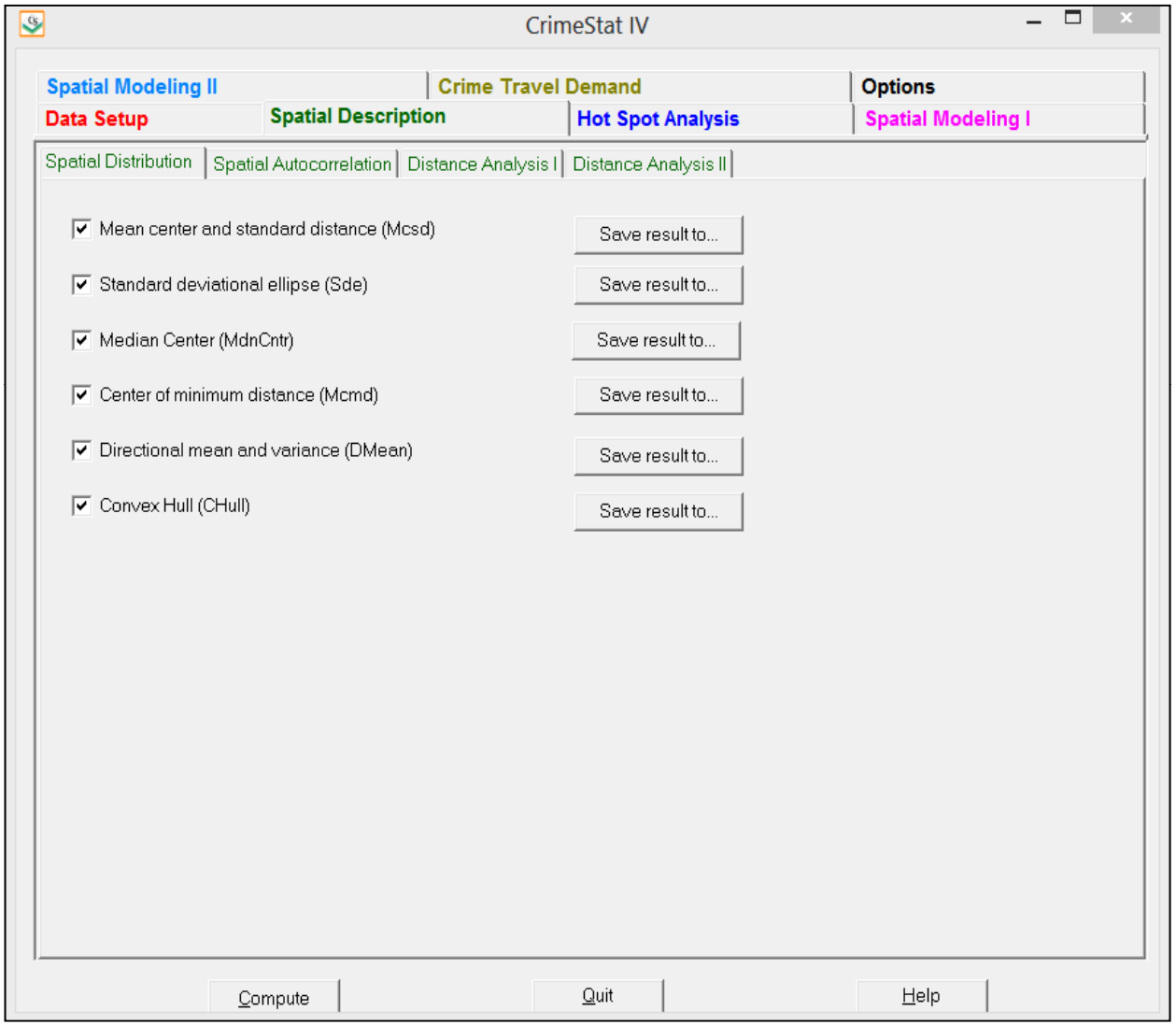


Figure 4.2:

Selecting Centrographic Statistics



For a single variable, the mean is the point at which the sum of all differences between the mean and all other points is zero. Unfortunately, for two variables, such as the location of crime incidents, the mean center is not necessarily the point at which the sum of all distances to all other points is minimized. That property is attributed to the center of minimum distance (see below). However, the mean center can be thought of as a point where both the sum of all differences between the mean X coordinate and all other X coordinates is zero and the sum of all differences between the mean Y coordinate and all other Y coordinates is zero.

The formula for the mean center is:

$$\bar{X} = \sum_{i=1}^{N} \frac{X_i}{N} \tag{4.1}$$

$$\bar{Y} = \sum_{i=1}^{N} \frac{Y_i}{N} \tag{4.2}$$

where X_i and Y_i are the coordinates of individual locations and N is the total number of points. To take a simple example, the mean center for burglaries in Baltimore County has spherical coordinates of longitude -76.608482, latitude 39.348368 and for robberies longitude -76.620838, latitude 39.334816. Figure 4.3 illustrates these two mean centers.

Weighted Mean Center

A ***weighted mean center*** is produced by weighting each coordinate by another variable, W_i. For example, if the coordinates are the centroids of census tracts, then the weight could be the population within the census tract. The weights have to be a positive number greater than or equal to 1. The numerator is the sum of the product of the variable and the weight while the denominator is the sum of weights,

$$\bar{X} = \frac{\sum_{i=1}^{N} W_i X_i}{\sum_{i=1}^{N} W_i} \tag{4.3}$$

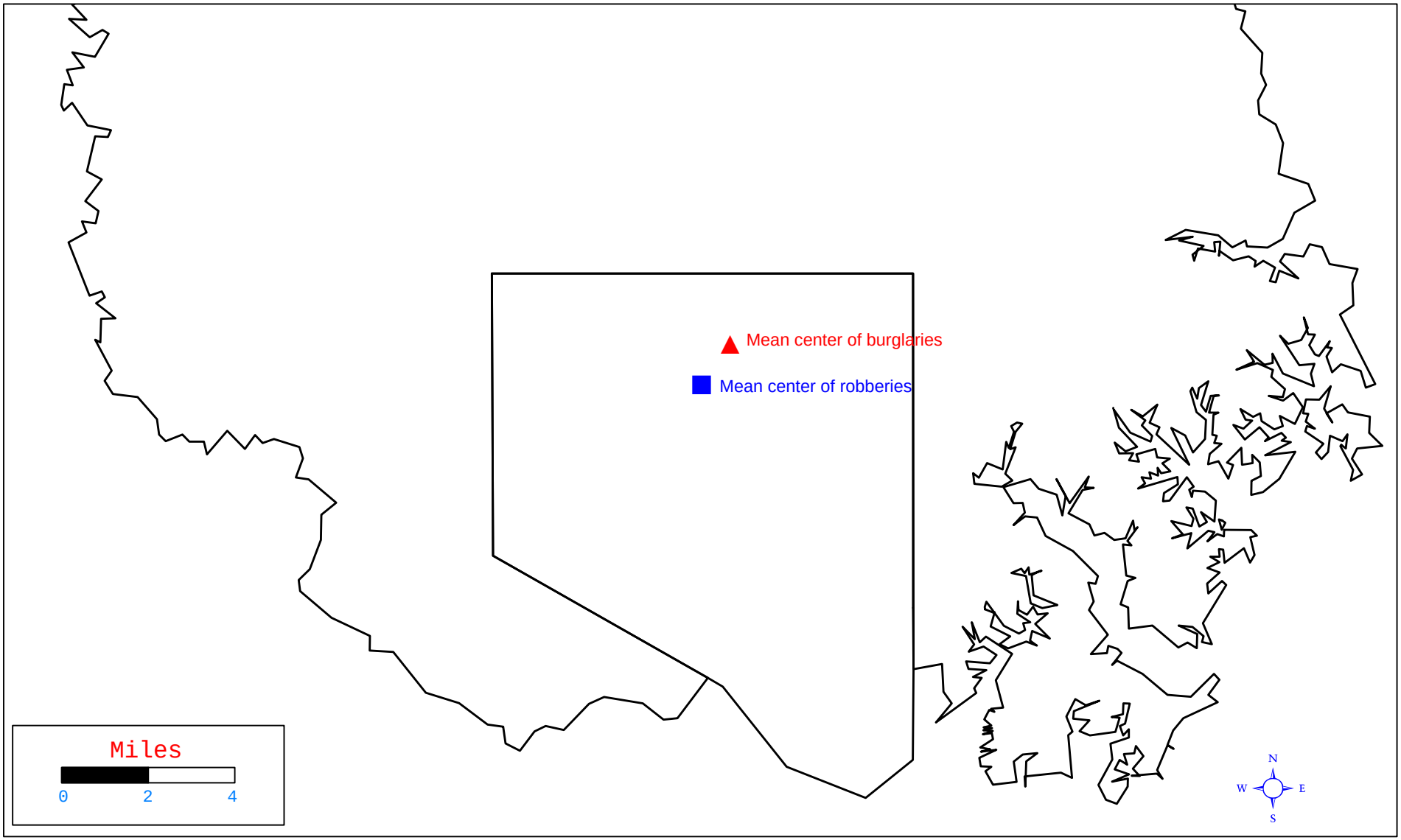
$$\bar{Y} = \frac{\sum_{i=1}^{N} W_i Y_i}{\sum_{i=1}^{N} W_i} \tag{4.4}$$

where W_i is the weight of observation i and X_i and Y_i are as defined in equations 4.1 and 4.2.

The advantage of a weighted mean center is that points associated with areas can have the characteristics of the areas included. For example, if the coordinates are the centroids of census

Figure 4.3: Burglary and Robbery in Baltimore County

Comparison of Mean Centers



tracts, then the weight of each centroid could be the population within the census tract. This will produce a different center of gravity than the unweighted center of all census tracts.

CrimeStat allows the mean to be weighted by either the weighting variable or by the intensity variable. Users should be careful, however, not to weight the mean with both the weighting and intensity variable unless there is an explicit distinction being made between weights and intensities.

To take an example, in the six jurisdictions making up the metropolitan Baltimore area (Baltimore City, and Baltimore, Carroll, Harford, Howard and Anne Arundel counties), the mean center of all census block groups is longitude -76.619121, latitude 39.304344. This would be an *unweighted* mean center of the block groups. On the other hand, the mean center of the 1990 population for the Baltimore metropolitan area had coordinates of longitude -76.625186 and latitude 39.304186, a position slightly southwest of the unweighted mean center. Weighting the block groups by median household income produces a mean center which is still more southwest. Figure 4.4 illustrates these three mean centers.

Weighted mean centers can be useful because they describe spatial differentiation in the metropolitan area and factors that may correlate with crime distributions. Another example is the weighted mean centers of different ethnic groups in the Baltimore metropolitan area (figure 4.5). The mean center of the White population is almost identical to the unweighted mean center. On the other hand, the mean center of the African-American/Black population is southwest of this and the mean center of the Hispanic/Latino population is considerably south of that for the White population. In other words, different ethnic groups tend to live in different parts of the Baltimore metropolitan area. Whether this has any impact on crime distributions is an empirical question.

When the *Mcsd* box is checked, *CrimeStat* will run the routine. *CrimeStat* has a status bar that indicates how much of the routine has been run (Figure 4.6).[1] The results of these statistics are shown in the *Mcsd* output table (figure 4.7).

1 Hint. There are 40 bars indicated in the status bar while a routine is running. For long runs, users can estimate the calculation time by timing how long it takes for two bars to be displayed and then multiply by 20.

Figure 4.4: Center of Baltimore Metropolitan Population

Mean Center of Block Groups Weighted By Selected Variables

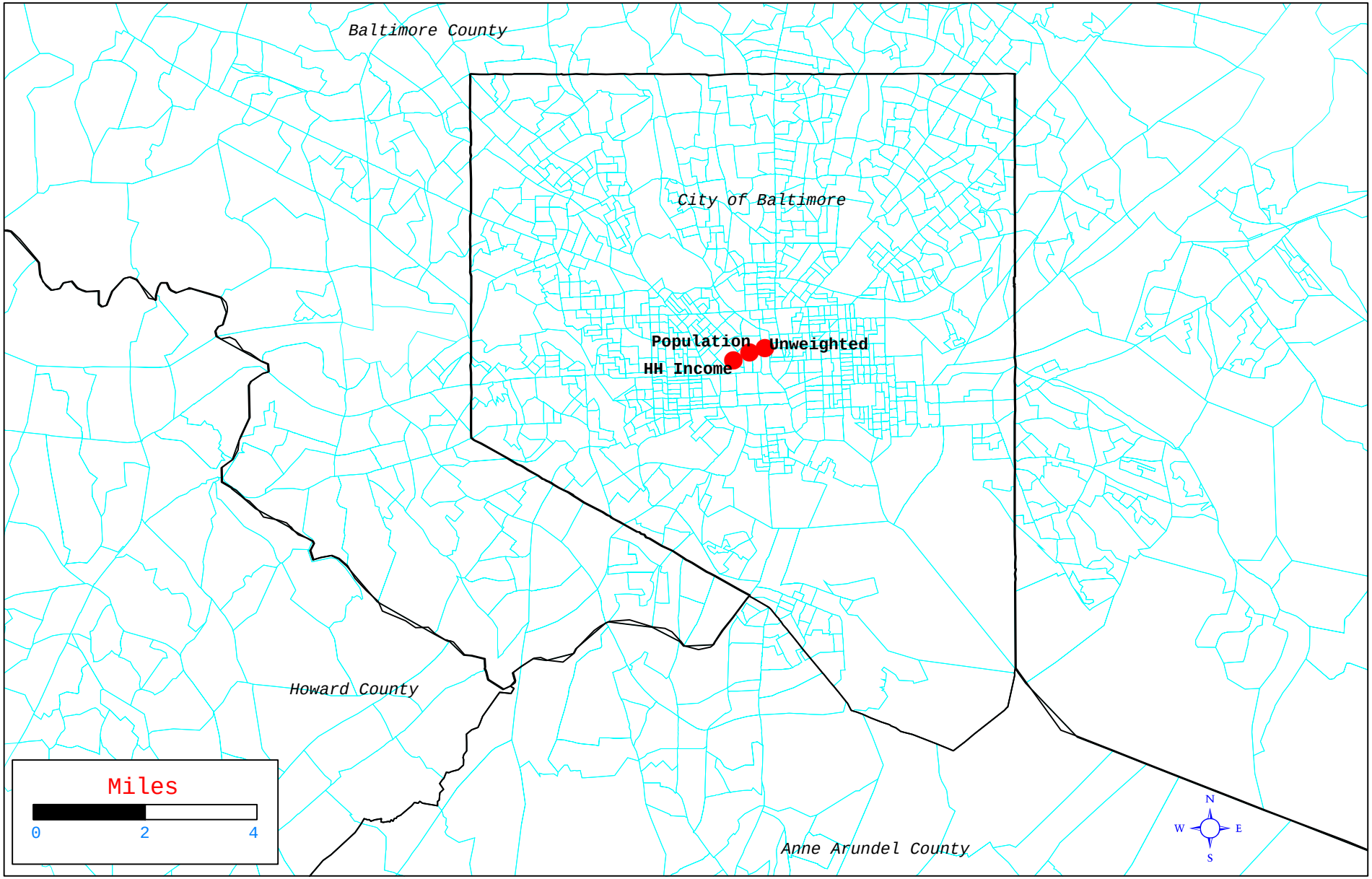


Figure 4.5: Center of Baltimore Metropolitan Population

Mean Center of Block Groups Weighted By Selected Variables

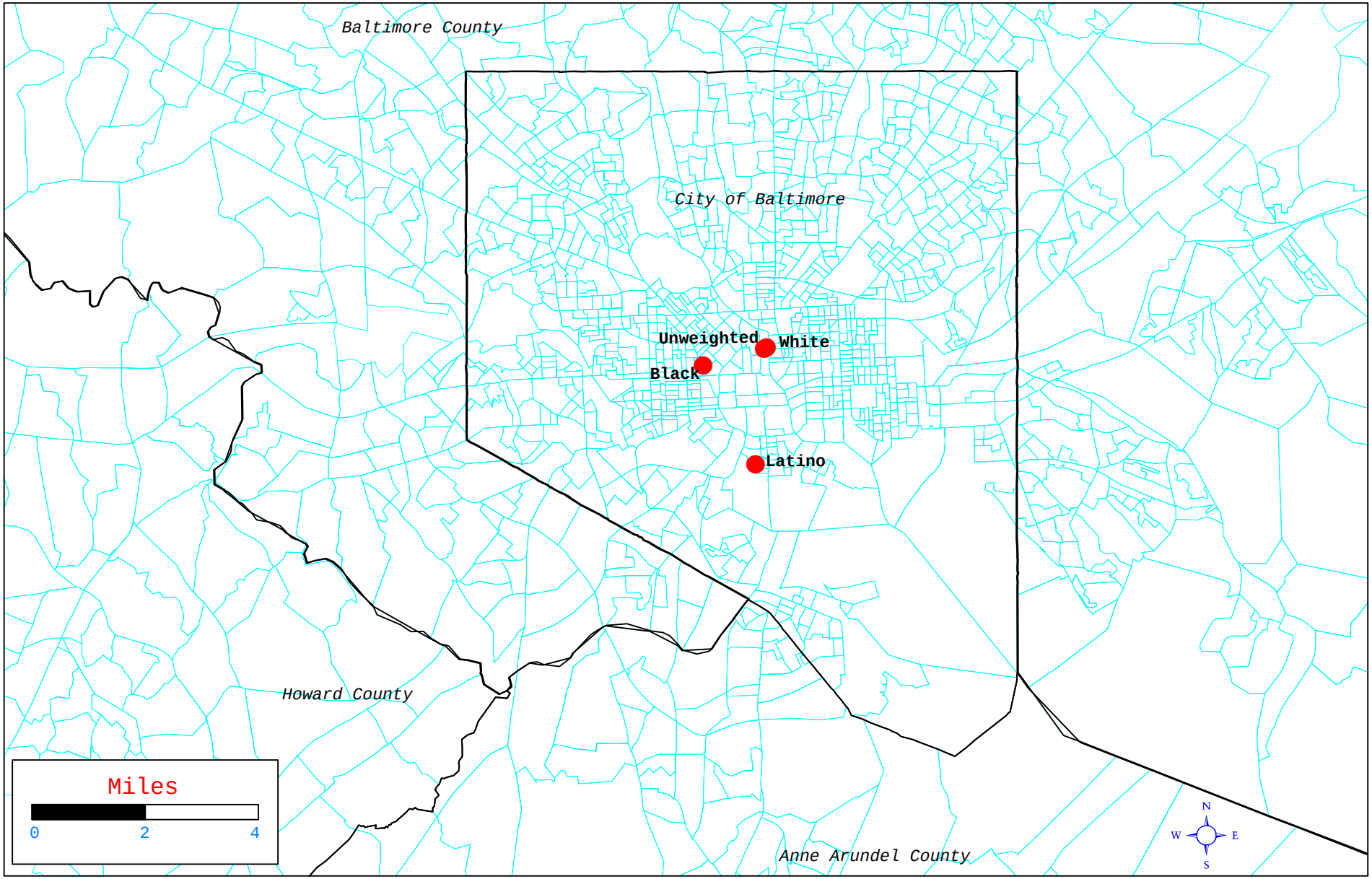


Figure 4.6:

CrimeStat Calculating a Routine

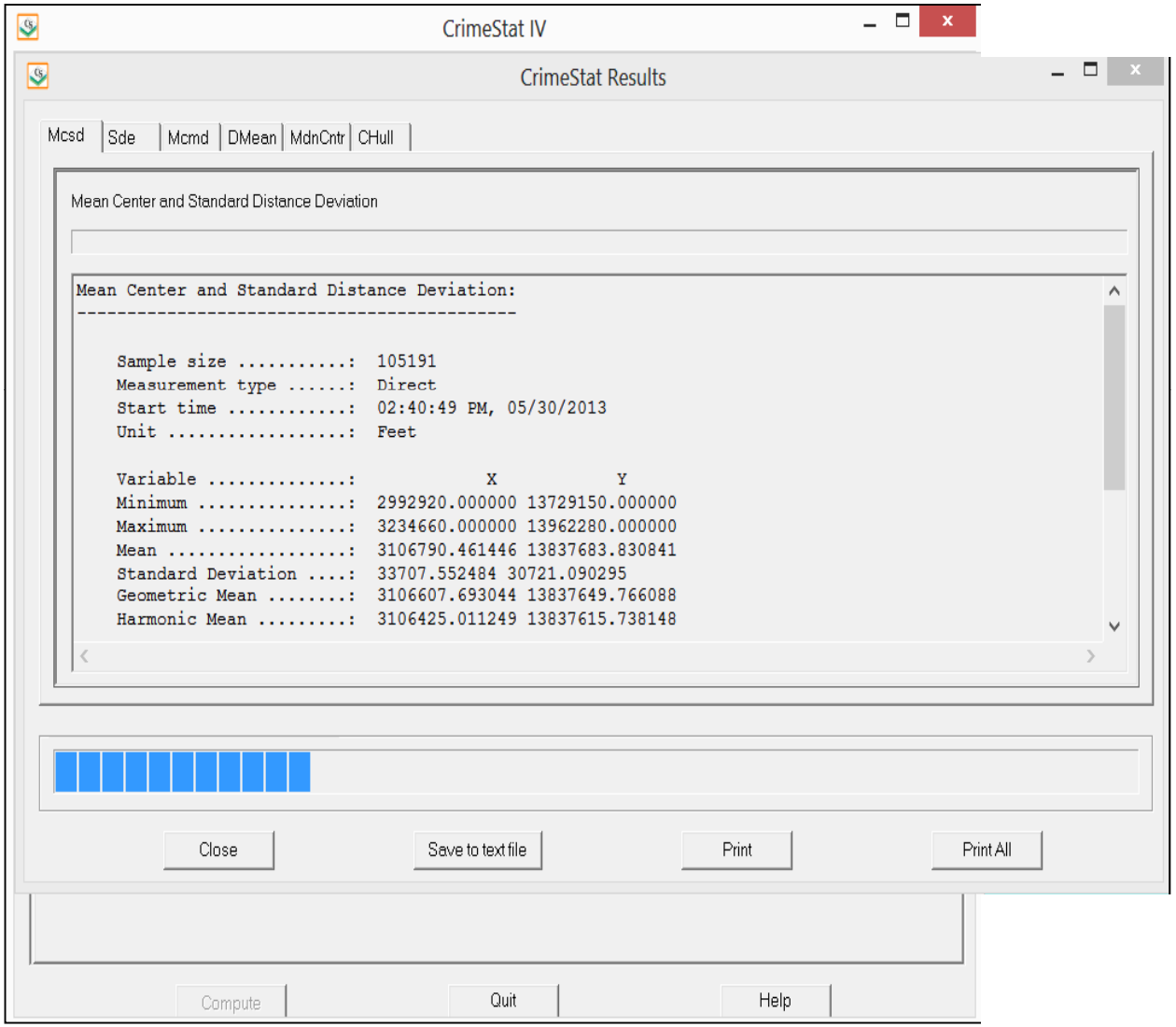


Figure 4.7:

Mean Center and Standard Distance Deviation Output

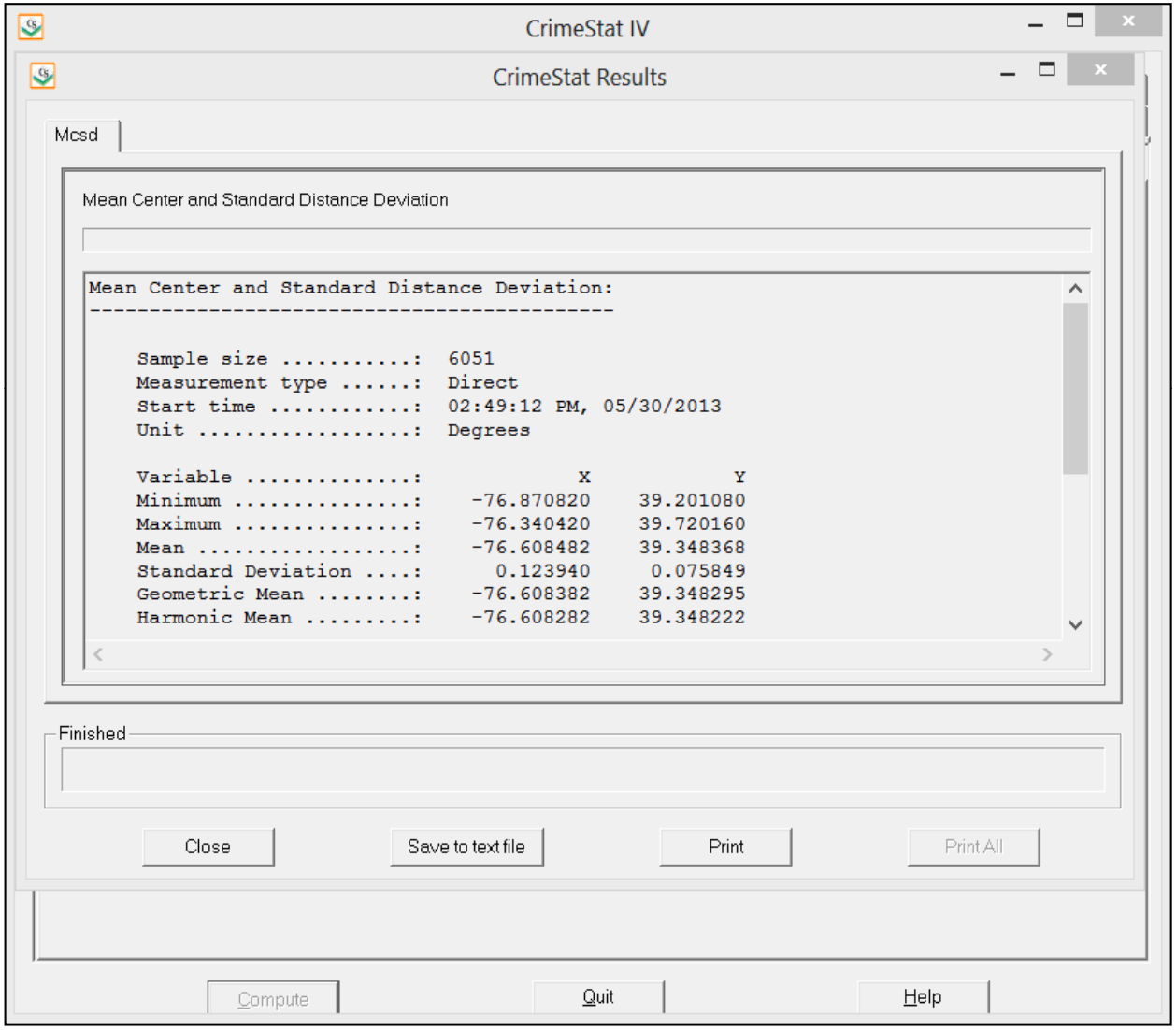
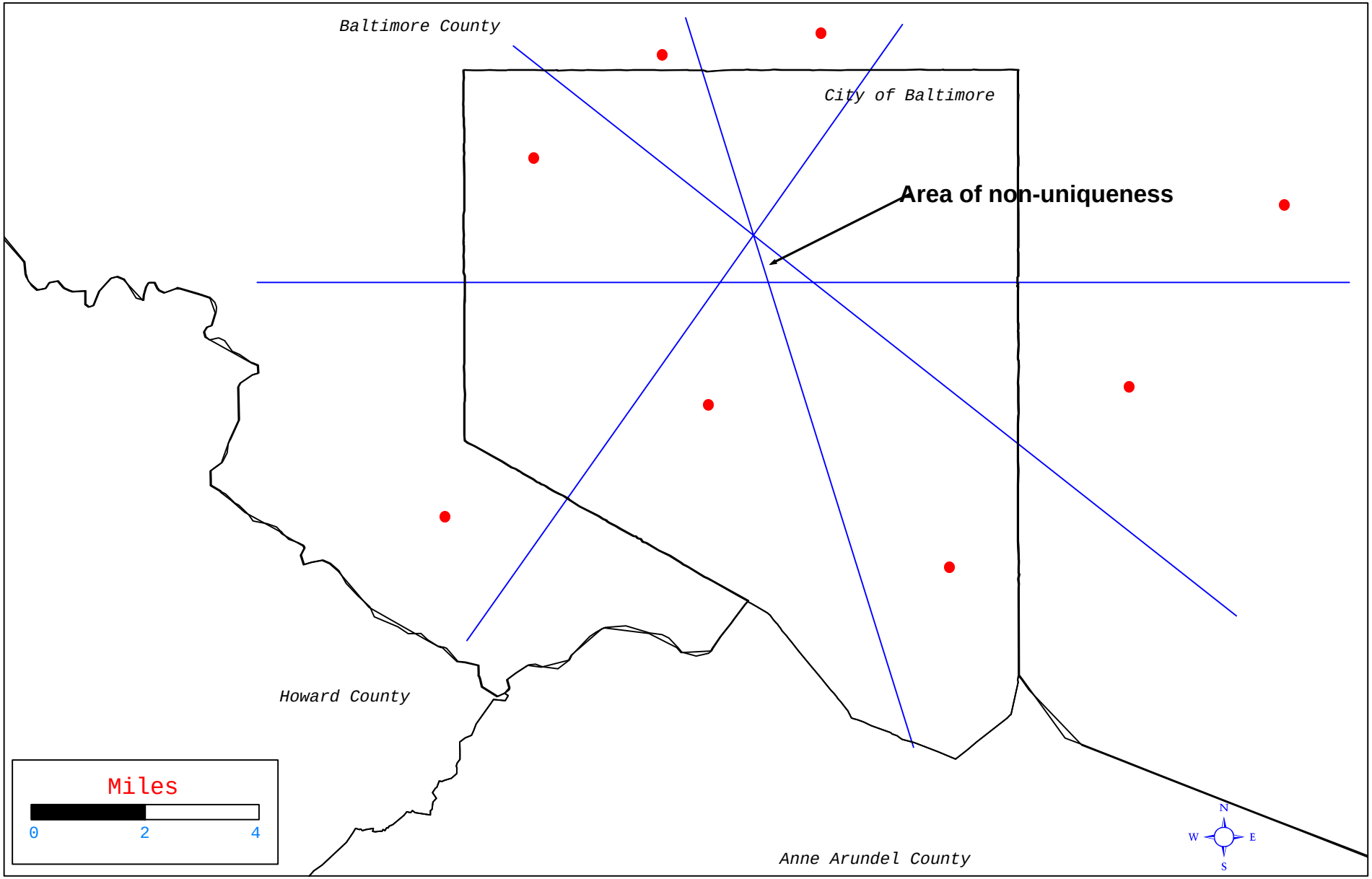


Figure 4.8: Non-Uniqueness of a Median Center

Lines Splitting Incident Locations Into Two Halves



Median Center

The **median center** is the intersection between the median of the X coordinate and the median of the Y coordinate. The concept is simple. However, it is not strictly a median. For a single variable, such as median household income, the median is that point at which 50% of the cases fall below and 50% fall above. On a two dimensional plane, however, there is not a single median because the location of a median is defined by the way that the axes are drawn.

For example, in figure 4.8, there are eight incident points shown. Four lines have been drawn which divide these eight points into two groups of four each. However, the four lines do not identify an exact location for a median. Instead, there is an area of non-uniqueness in which any part of it could be considered the 'median center'. This violates one of the basic properties of a statistic is that it be a unique value.

Nevertheless, as long as the axes are not rotated, the median center can be a useful statistic. The *CrimeStat* routine outputs three statistics:

1. The sample size
2. The median of X
3. The median of Y

The tabular output can be printed and the median center can be output as a graphical object to *ArcGIS* 'shp', *MapInfo* 'mif', *Google Earth* 'kml, or various Ascii files. A root name should be provided. The median center is output as a point (MdnCntr<root name>).

Center of Minimum Distance

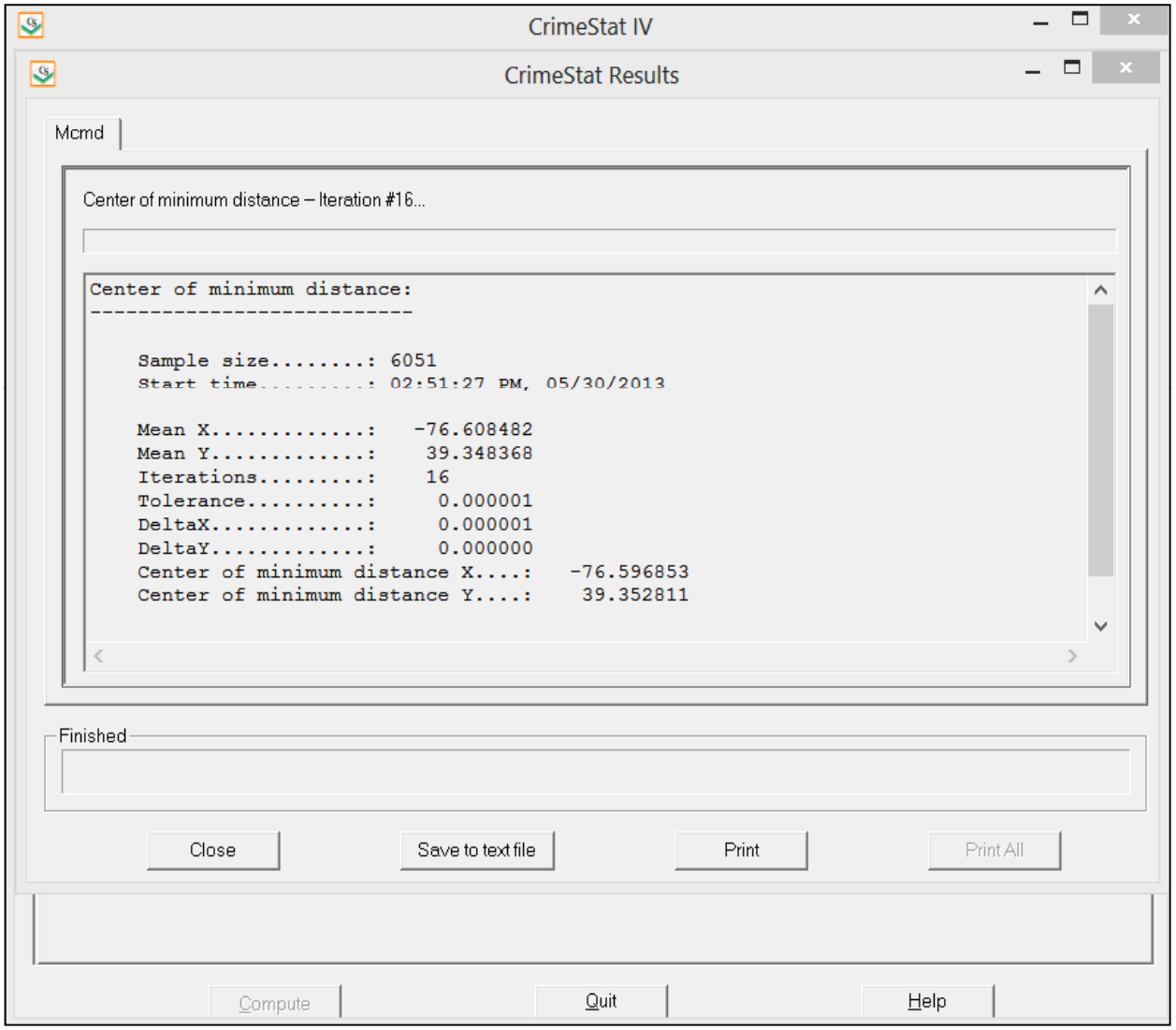
Another centrographic statistic is the **center of minimum distance**. Unfortunately, this statistic is sometimes also called the *median center*, which can make it confusing since the above statistic has the same name. Nevertheless, unlike the median center above, the center of minimum distance is a unique statistic in that it defines the point at which the sum of the distance to all other points is the smallest (Burt and Barber, 1996). It is defined as:

$$Center\ of\ minimum\ distance = CMD = \sum_{i=1}^{N} d_{iC} = min \qquad (4.5)$$

where d_{ic} is the distance between a single point, i, and C, the center of minimum distance (with an X and Y coordinate). Unfortunately, there is not a formula that can calculate this location.

Figure 4.9:

Center of Minimum Distance Output



Instead, an iterative algorithm is used that approximates this location (Kuhn and Kuenne, 1962; Burt and Barber, 1996; see endnote *i*). Depending on whether the coordinates are spherical or projected, *CrimeStat* will calculate distance as either Great Circle (spherical) or Euclidean (projected), as discussed in the previous chapter. The results are shown in the *Mcmd* output table (figure 4.9).

The importance of the center of minimum distance is that it is a location where distance to all the defining incidents is the smallest. Since *CrimeStat* only measures distances as either direct or indirect, actual travel time is not being calculated. But in many jurisdictions, the minimum distance to all points is a good approximation to the point where travel distances are minimized. For example, in a police precinct, a patrol car could be stationed at the center of minimum distance to allow it to respond quickly to calls for service.

For example, figure 4.10 maps the center of minimum distance for 1996 auto thefts in both Baltimore City and Baltimore County and compares this to both the mean center and the median center statistic. As seen, both the center of minimum distance and the median center are south of the mean center, indicating that there are slightly more incidents in the southern part of the metropolitan area than in the northern part. However, the difference in these three statistics is very small, especially the median center and the center of minimum distance.

Standard Deviations of the X and Y Coordinates

In addition to the mean center and center of minimum distance, *CrimeStat* will calculate various measures of spatial distribution, which describe the dispersion, orientation, and shape of the distribution of a variable (Hammond & McCullogh 1978; Ebdon 1988). The simplest of these is the raw **standard deviations of the X and Y coordinates**, respectively. The formulas used are the standard ones found in most elementary statistics books:

$$s_x = \sqrt{\sum_{i=1}^{N} \frac{(X_i - \bar{X})}{N-1}} \tag{4.6}$$

$$s_y = \sqrt{\sum_{i=1}^{N} \frac{(Y_i - \bar{y})}{N-1}} \tag{4.7}$$

where X_i and Y_i are the X and Y coordinates for individual points, $\bar{X}$ and $\bar{Y}$ are the means of X and Y respectively, and N is the total number of points. Note that 1 is subtracted from the number of points to produce an unbiased estimate of the standard deviation.

Figure 4.10: 1996 Metropolitan Baltimore Auto Thefts

Mean Center and Center of Minimum Distance for 1996 Auto Thefts

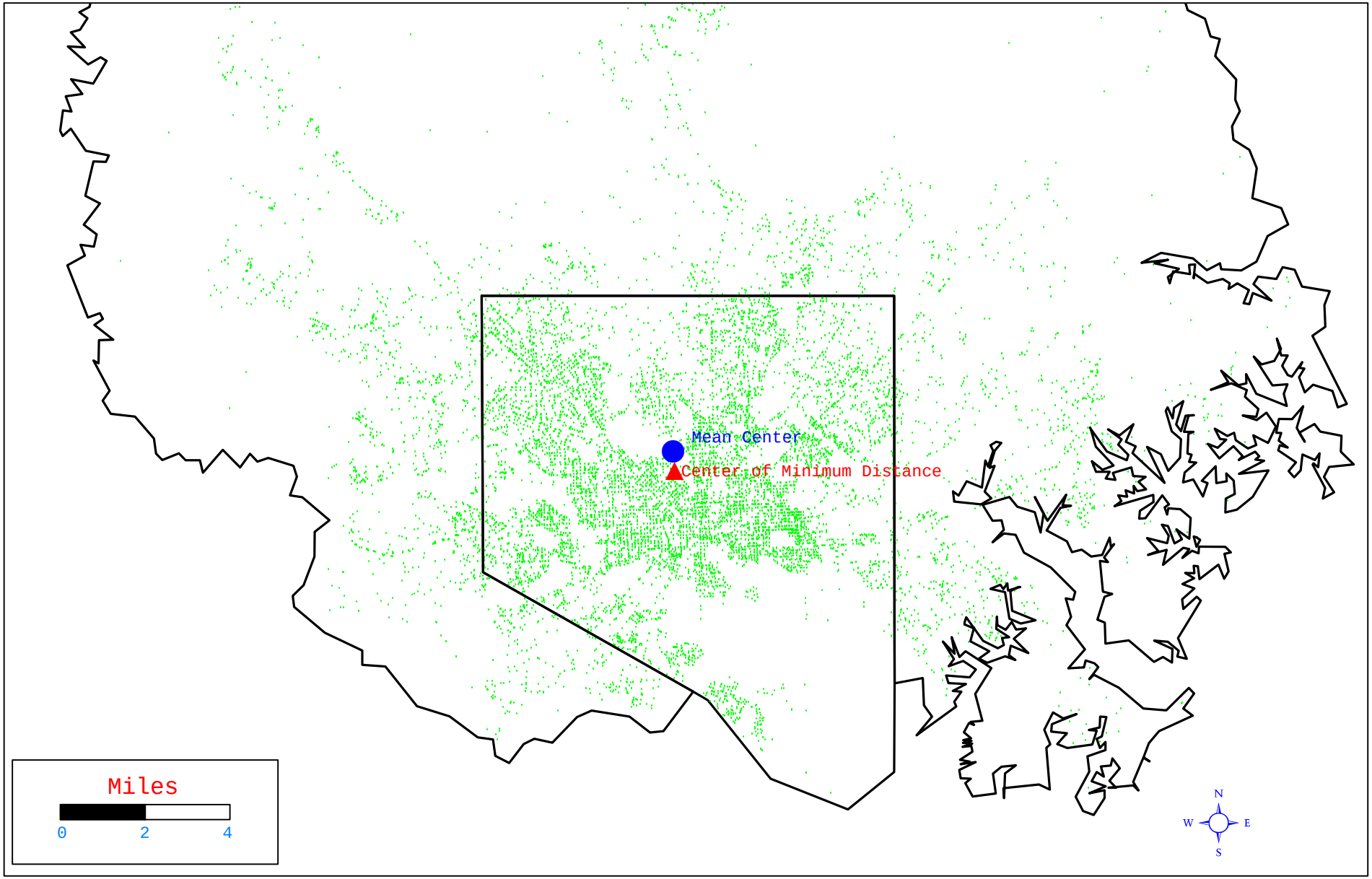


Figure 4.11 shows the standard deviation of the coordinates for auto thefts and represents this as a rectangle. As seen, the distribution of auto thefts spreads more in an east-west direction than in a north-south direction.

Standard Distance Deviation

While the standard deviation of the X and Y coordinates provides some information about the dispersion of the incidents, there are two problems with it. First, it does not provide a single summary statistic of the dispersion in the locations and is actually two separate statistics, the dispersion in X and the dispersion in Y. Second, it provides measurement in the units of the coordinate system. Thus, if spherical coordinates are being used, then the units will be decimal degrees. On the other hand, if projected coordinates are being used, then units will be in feet or meters or some other metric.

A measure which overcomes these problems is the **standard distance deviation** (or *standard distance*, for short). This is the standard deviation of the *distance* of each point from the mean center and is expressed in measurement units (feet, meters, miles). It is the two-dimensional equivalent of a standard deviation.

The formula for it is:

$$SDD = \sqrt{\sum_{i=1}^{N} \frac{(d_{iMC})^2}{N-2}} \tag{4.8}$$

where d_{iMC} is the distance between each point, i, and the mean center and N is the total number of points. Note that 2 is subtracted from the number of points to produce an unbiased estimate of standard distance since there are two constants from which this distance is measured (mean of X, mean of Y).[2]

The standard distance can be represented as a single vector rather than two vectors as with the standard deviation of the X and Y coordinates. Figure 4.12 shows the mean center and standard distance deviation of both robberies and burglaries for 1996 in Baltimore County

---

2 With a weight for an observation, w_i, the squared distance is weighted and the formula becomes:

$$s_{XY} = \sqrt{\frac{\sum w_i d_{imc}^2}{(\sum w_i)-2}}$$

where d_{iMC} is the distance from the point to the mean center. Both summations are over all points, N.

Figure 4.11: 1996 Metropolitan Baltimore Auto Thefts

Mean Center and Standard Deviations of X and Y Coordinates

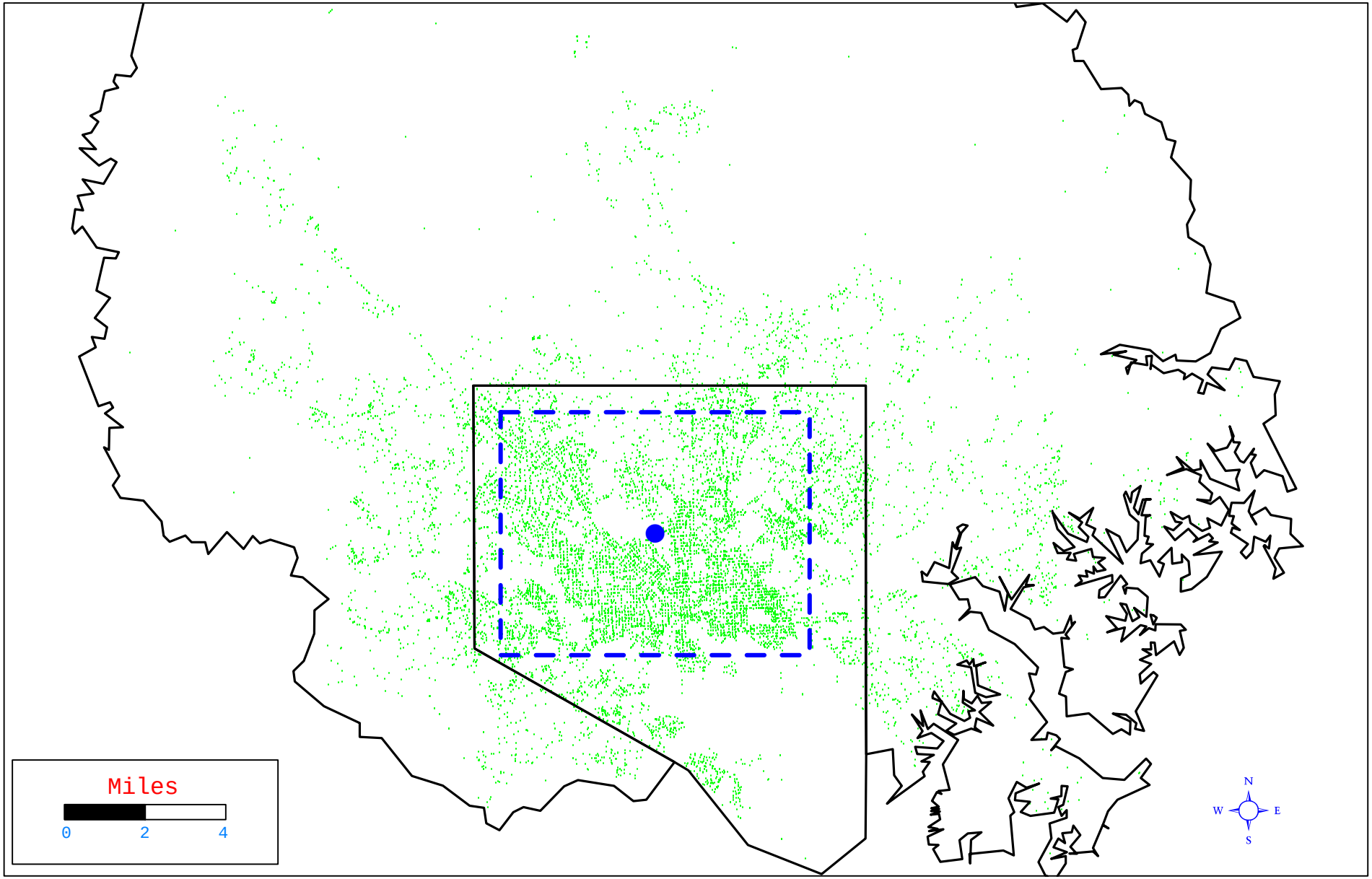
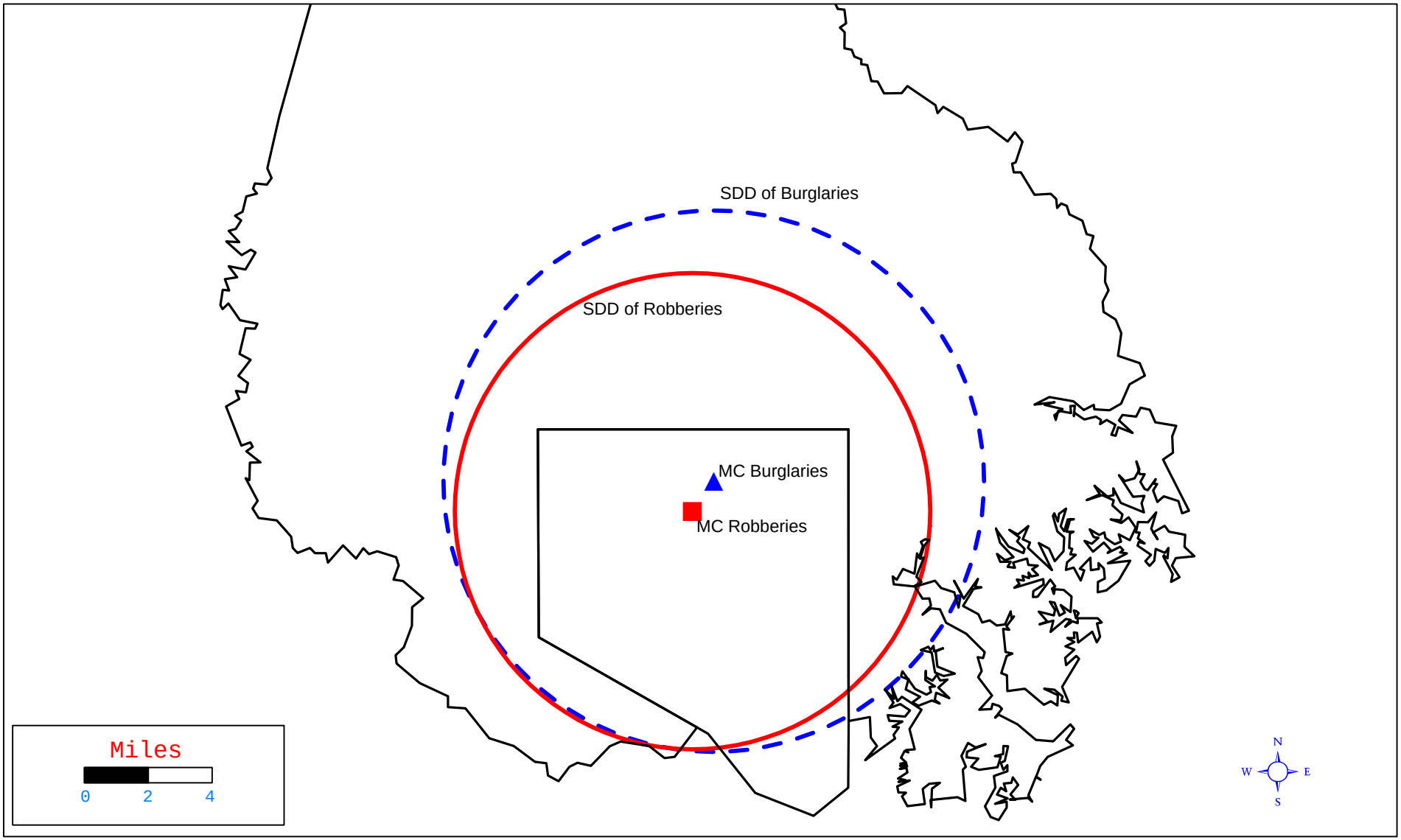


Figure 4.12: 1996 Baltimore County Burglaries and Robberies

Comparison of Mean Centers and Standard Distance Deviations



represented as circles. It is clear that the spatial distributions of these two types of crime vary with robberies being slightly more concentrated.

Standard Deviational Ellipse

The standard distance deviation is a good single measure of the dispersion of the incidents around the mean center. However, with two dimensions, distributions are frequently skewed in one direction or another (a condition called *anisotropy*).

Instead, there is another statistic that gives dispersion in two dimensions, the **standard deviational ellipse** (or *ellipse*, for short; Ebdon, 1988; Cromley, 1992). The standard deviational ellipse is derived from the bivariate distribution (Furfey, 1927; Neft, 1962; Bachhi, 1957) and is defined by:

$$Bivariate\ distribution = \sqrt{\frac{(s_X^2+s_Y^2)}{2}} \tag{4.9}$$

The two standard deviations, in the X and Y directions, are orthogonal to each other and define an ellipse. Ebdon (1988) rotates the X and Y axis so that the sum of squares of distances between points and axes are minimized. By convention, it is shown as an ellipse.

Aside from the mean X and mean Y, the formulas for these statistics are as follows (the observation subscript, i, has been dropped from the summation sign):

1. The Y-axis is rotated *clockwise* through an angle, θ, where

$$\theta = arctan\left\{\frac{\left[\sum(X_i-\bar{\mathrm{X}})^2-\sum(\mathrm{Y_i}-\bar{\mathrm{y}})^2\right]+\sqrt{\left[\{(\sum(X_i-\bar{\mathrm{X}})^2-\sum(\mathrm{Y_i}-\bar{\mathrm{y}})^2\}^2+4\{\sum\left(\mathrm{X_i}-\bar{\mathrm{X}}\right)(\mathrm{Y_i}-\bar{\mathrm{y}})^2\}\right]}}{2\sum(X_i-\bar{\mathrm{X}})(\mathrm{Y_i}-\bar{\mathrm{y}})}\right\} \tag{4.10}$$

where all summations are for i=1 to N (Ebdon, 1988).

2. Two standard deviations are calculated, one along the transposed X-axis and one along the transposed Y-axis:

$$s_X = \sqrt{\frac{\sum[\left(x_i-\bar{\mathrm{X}}\right)\cos\theta-\left(Y_i-\bar{\mathrm{y}}\right)\sin\theta]^2}{N-2}} \tag{4.11}$$

$$s_Y = \sqrt{\frac{\sum[\left(x_i - \bar{\mathrm{x}}\right)\sin\theta + \left(Y_i - \bar{\mathrm{y}}\right)\cos\theta]^2}{N-2}} \tag{4.12}$$

where $\bar{X}$ and $\bar{Y}$ are the means of X and Y respectively, θ is the angle (in radians), and N is the number of points. Note, again, that 2 is subtracted from the number of points in both denominators to produce an unbiased estimate of the standard deviational ellipse since there are two constants from which the distance along each axis is measured ($\bar{X}, \bar{Y}$; see endnote ii).

3. The X-axis and Y-axis of the ellipse are defined by:

$$Length_X = 2s_X \tag{4.13}$$

$$Length_Y = 2s_Y \tag{4.14}$$

4. The area of the ellipse is:

$$A = \pi s_X s_X \tag{4.15}$$

Figure 4.13 shows the output of the ellipse routine and figure 4.14 maps the standard deviational ellipse of auto thefts in Baltimore City and Baltimore County for 1996.

Geometric Mean

The mean center routine (Mcsd) includes two additional means. First, there is the **geometric mean**, which is a mean associated with the mean of the logarithms. It is defined as:

$$Geometric\ Mean\ of\ X = GM(X) = \prod_{j=1}^{N} (X_i^{W_i})^{\frac{1}{\Sigma W_i}} \tag{4.16}$$

$$Geometric\ Mean\ of\ Y = GM(Y) = \prod_{j=1}^{N} (Y_i^{W_i})^{\frac{1}{\Sigma W_i}} \tag{4.17}$$

where Π is the product term of each point value, i (i.e., the values of X or Y are multiplied times each other), W_i is the weight used (default=1), and N is the sample size (Everitt, 2011). The weights must be defined on the Primary File page, either in the Weights field or in the Intensity field (but not both together).

Figure 4.13:

Standard Deviational Ellipse Output

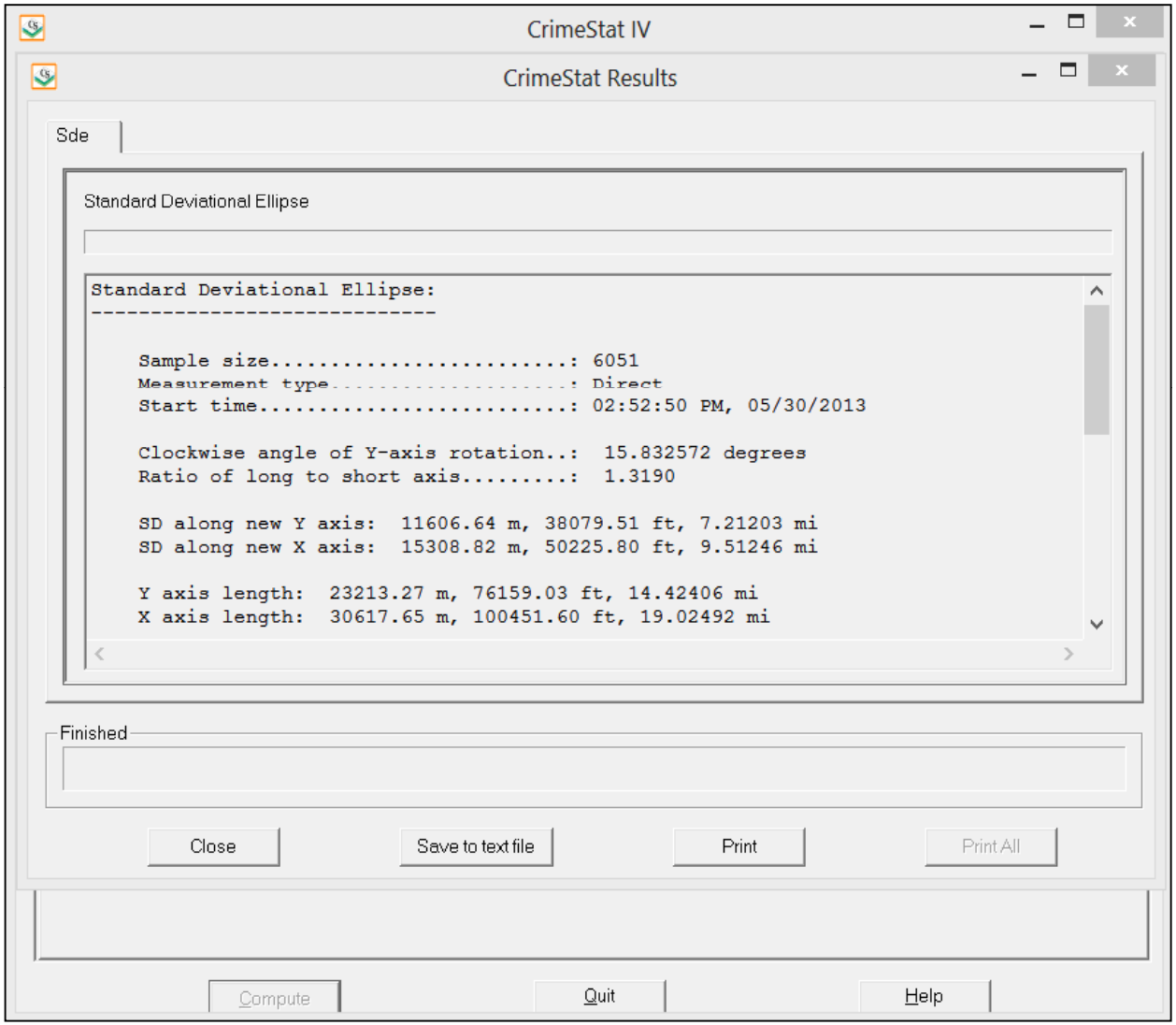
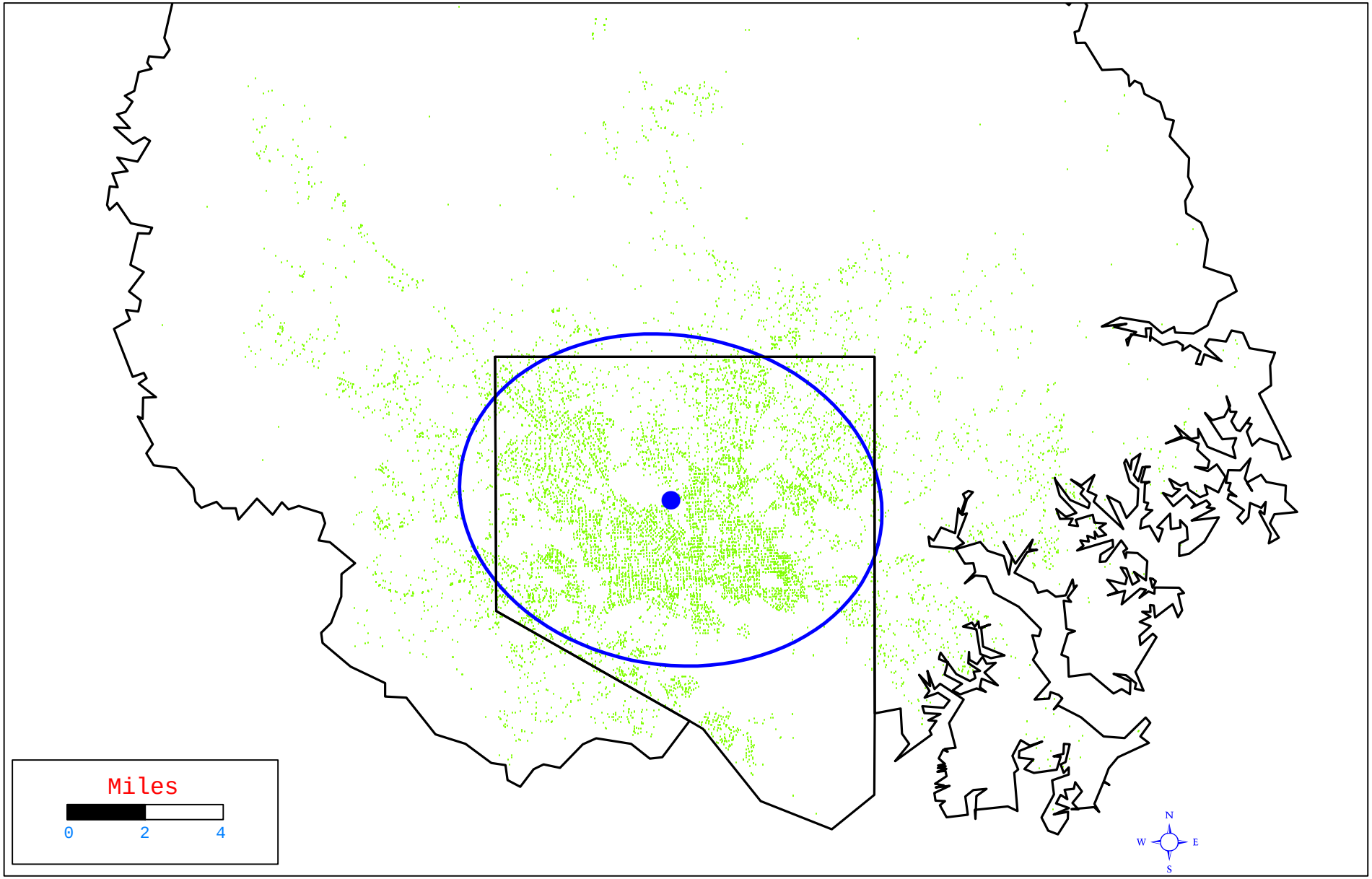


Figure 4.14: 1996 Metropolitan Baltimore Auto Thefts

Mean Center and Standard Deviational Ellipse



The equation can be evaluated by logarithms:

$$Ln[GM(X)] = \frac{1}{\sum W_i}[W_1 Ln(X_1) + W_2 Ln(X_2) + \ldots + W_N Ln(X_N)]$$

$$= \frac{\sum[W_i Ln(X_i)]}{\sum W_i} \quad (4.18)$$

$$Ln[GM(Y)] = \frac{1}{\sum W_i}[W_1 Ln(Y_1) + W_2 Ln(Y_2) + \ldots + W_N Ln(Y_N)] = \frac{\sum[W_i Ln(Y_i)]}{\sum W_i} \quad (4.19)$$

$$GM(X) = e^{Ln[GM(X)]} \quad (4.20)$$

$$GM(Y) = e^{Ln[GM(Y)]} \quad (4.21)$$

The geometric mean is the anti-log of the mean of the logarithms. If weights are used, then the logarithm of each X or Y value is weighted and the sum of the weighted logarithms are divided by the sum of the weights. If weights are not used, then the default weight is 1 and the sum of the weights will equal the sample size. The geometric mean is output as part of the Mcsd routine and has a 'Gm' prefix before the user defined name.

Uses

The geometric mean is used when units are multiplied by each other (e.g., robberies increase by 5% one year, 3% the next, and 4% the next; Wikipedia, 2007a). One cannot take the simple mean because there is a cumulative change in the units. In most cases, this is not relevant to point (incident) locations since the coordinates of each incident are independent and are not multiplied by each other. However, the geometric mean can be useful because it converts all X and Y coordinates into logarithms and, thus, has the effect of discounting extreme values.

Harmonic Mean

The **harmonic mean** is a lso a mean which discounts extreme va lues, but is calculated differently. It is defined as (Wikipedia, 2007b):

$$Harmonic\ mean\ of\ X = HM(X) = \frac{\sum W_i}{\sum(W_i/X_i)} \quad (4.22)$$

$$Harmonic\ mean\ of\ Y = HM(Y) = \frac{\sum W_i}{\sum(W_i/Y_i)} \quad (4.23)$$

where W_i is the weight used (default=1), X_i and Y_i are the X and Y values, and N is the sample size. The weights have to be defined on the Primary File page, either in the Weights field or in the Intensity field (but not both together).

The harmonic mean of X and Y is the inverse of the mean of the inverse of X and Y respectively (i.e., take the inverse; take the mean of the inverse; and invert the mean of the inverse). If weights are used, then each X or Y value is weighted by its inverse while the numerator is the sum of the weights. If weights are not used, then the default weight is 1 and the sum of weights will equal the sample size. The harmonic mean is output as part of the Mcsd routine and has a 'Hm' prefix before the user-defined name.

Uses

Typically, harmonic means are used in calculating the average of rates, or quantities whose values are changing over time (Wikipedia, 2007b). For example, in calculating the average speed over multiple segments of equal length (see chapter 30 on Network Assignment), the harmonic mean should be used, not the arithmetic mean. If there are two adjacent road segments, each one mile in length and if a car travels over the first segment 20 miles per hour (mph) but over the second segment at 40 mph, the average speed is not 30 mph (the arithmetic mean), but 26.7 mph (the harmonic mean). The car takes 3 minutes to travel the first segment (60 minutes per hour times 1 mile divided by 20 mph) and 1.5 minutes to travel the second segment (60 minutes per hour times 1 mile divided by 40 mph). Thus, the total time to travel the two miles is 4.5 minutes and the average speed is 26.7 mph.

Again, for point (incident) locations, the harmonic mean would normally not be relevant since the coordinates of each of the incidents are independent. However, since the harmonic mean is weighted more heavily by the smaller values, it can be useful to discount cases which have outlying coordinates.

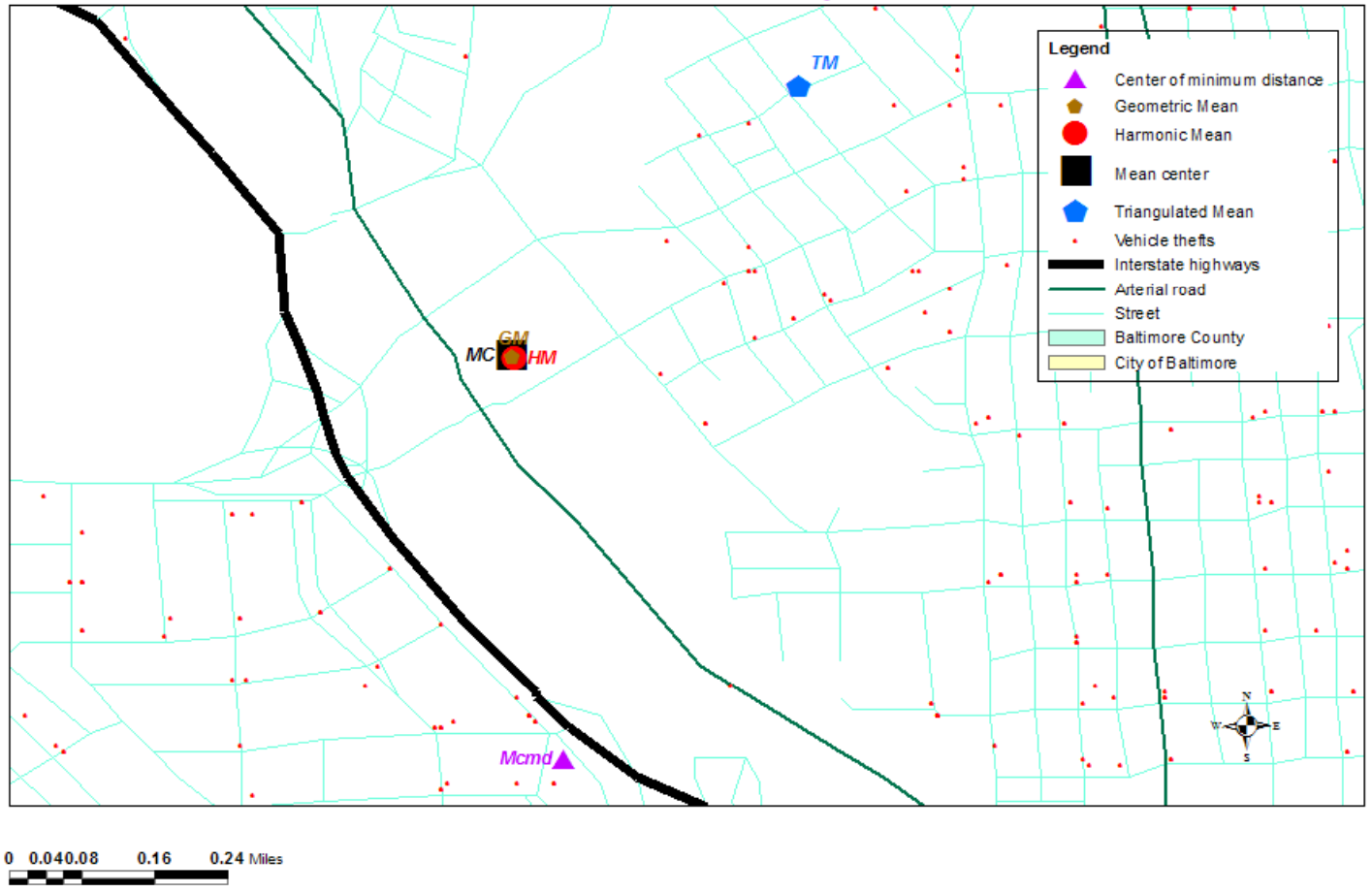
In other words, the harmonic mean of X and Y respectively is the inverse of the mean of the inverse of X and Y respectively (i.e., take the inverse; take the mean of the inverse; and invert the mean of the inverse). If weights are used, then each X or Y value is weighted and the numerator is the sum of the weights. If weights are not used, then the sum of the weights will equal the sample size. The harmonic mean is output as part of the Mcsd routine and has a 'Hm' prefix before the user defined name.

The geometric and harmonic means are discounted means that 'hug' the center of the distribution. They differ from the mean center when there is a very skewed distribution.

Figure 4.15:

1996 Baltimore Metropolitan Vehicle Thefts

5 Mean Centers Compared



To contrast the different means, figure 4.15 below shows five different means for Baltimore County motor vehicle thefts:

1. Mean center;
2. Center of minimum distance;
3. Geometric mean;
4. Harmonic mean; and
5. Triangulated mean (discussed below)

In the example, the mean center, geometric mean, and harmonic mean fall very close to each other; however, they will not always be so. The center of minimum distance approximates the geographical center of the distribution. The triangulated mean is defined by the angularity and distance from the lower-left and upper-right corners of the data set (see below).

Centrographic descriptors can be very powerful tools for examining spatial patterns. They are a first step in any spatial analysis, but an important one. The above example illustrates how they can be a basis for decision-making, even with small samples. A couple of other examples can be illustrated.

Average Density

The **average density** is the number of incidents divided by the area. It is a measure of the average number of events per unit of area; it is sometimes called the *intensity*. If the area is defined on the measurement parameters page, the routine uses that value for area; otherwise, it takes the rectangular area defined by the minimum and maximum X and Y values (the bounding rectangle).

Output Files

Calculating the Statistics

Once the statistics have been selected, the user clicks on *Compute* to run the routine. The results are shown in a results table.

Tabular Output

For each of these statistics, *CrimeStat* produces tabular output. In *CrimeStat*, all tables are labeled by symbols, for example Mcsd for the mean center and standard distance deviation or Mcmd for the center of minimum distance. All tables present the sample size.

Graphical Objects

The six centrographic statistics can be output as graphical objects. The mean center and center of minimum distance are output as single points. The standard deviation of the X and Y coordinates is output as a rectangle. The standard distance deviation is output as a circle and the standard deviational ellipse is output as an ellipse.

CrimeStat currently supports graphical outputs to *ArcGIS* 'shp', *MapInfo* 'mif', *Google Earth* 'kml, or various Ascii files. Before running the calculation, the user should select the desired output files and specify a root name (e.g., Precinct1Burglaries). Figure 4.16 shows a dialog box for outputting a shape file to *ArcGIS*. For *MapInfo* output only, the user has to also indicate the name of the projection, the projection number and the datum number. These can be found in the *MapInfo* users guide. By default, *CrimeStat* will use the standard parameters for a spherical coordinate system (Earth projection, projection number 1, and datum number 33). If a user requires a different coordinate system, the appropriate values should be typed into the space. Figure 4.17 shows the selection of the *MapInfo* coordinate parameters.

If requested, the output files are saved in the specified directory under the specified (root) name. For each statistic, *CrimeStat* will add prefix letters to the root name.

MC<*root*> for the mean center
MdnCntr<root> for the median center
Mcmd<*root*> for center of minimum distance
XYD<*root*> for the standard deviation of the X and Y coordinates
SDD<*root*> for the standard distance deviation
SDE<*root*> for the standard deviational ellipse.

The '.shp' files can be read directly into *ArcGIS* as themes. The 'kml files can be read directly into Google Earth. The '.mif' files have to be imported into *MapInfo*.[3]

3 In *MapInfo*, the comm`and is Table Import <*MapInfo Interchange file*> though it is a lot easier to use the *MapInfo* Universal Translator.

Figure 4.16:

Outputting a Shape File to *ArcView/ArcGIS*

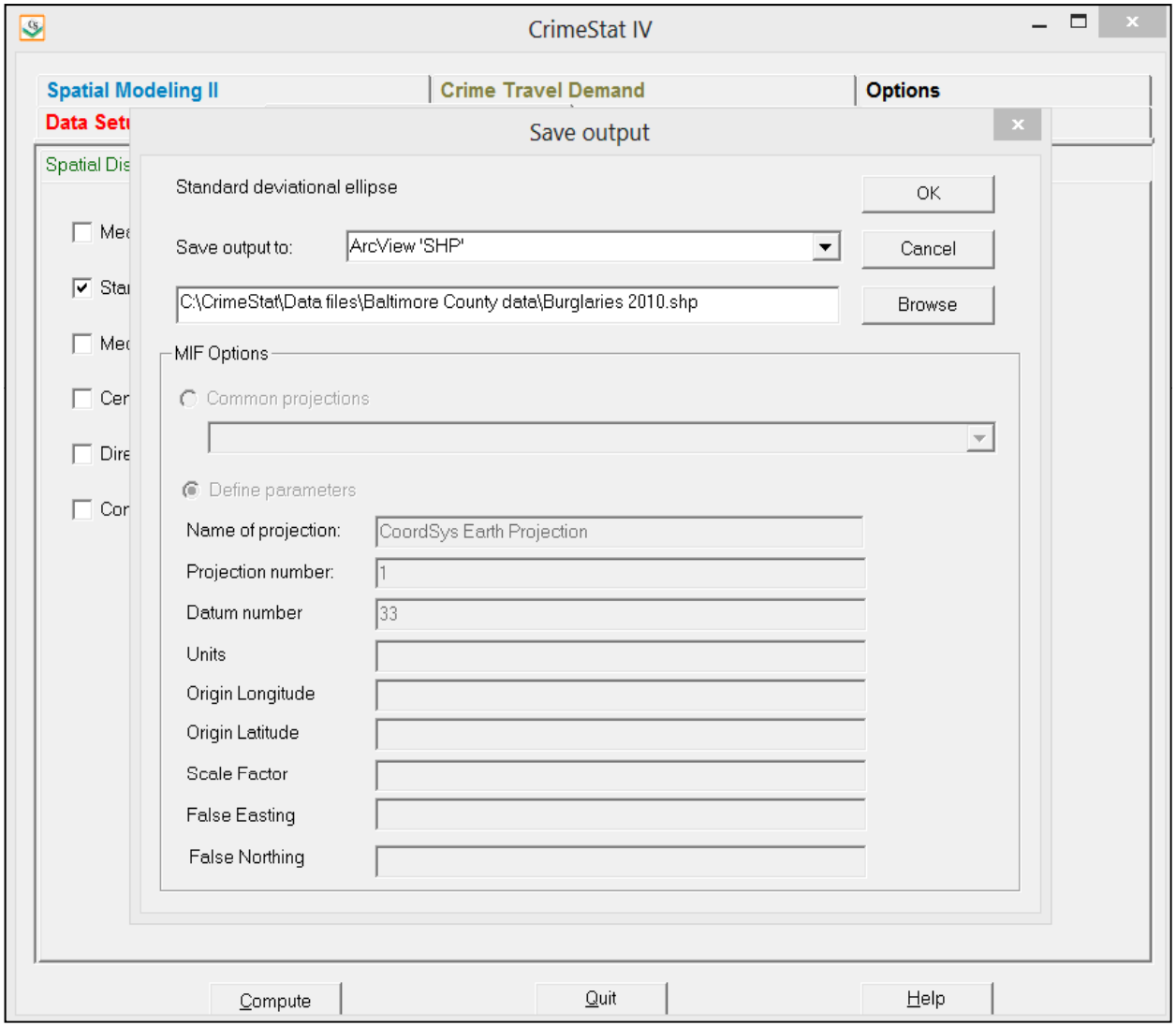
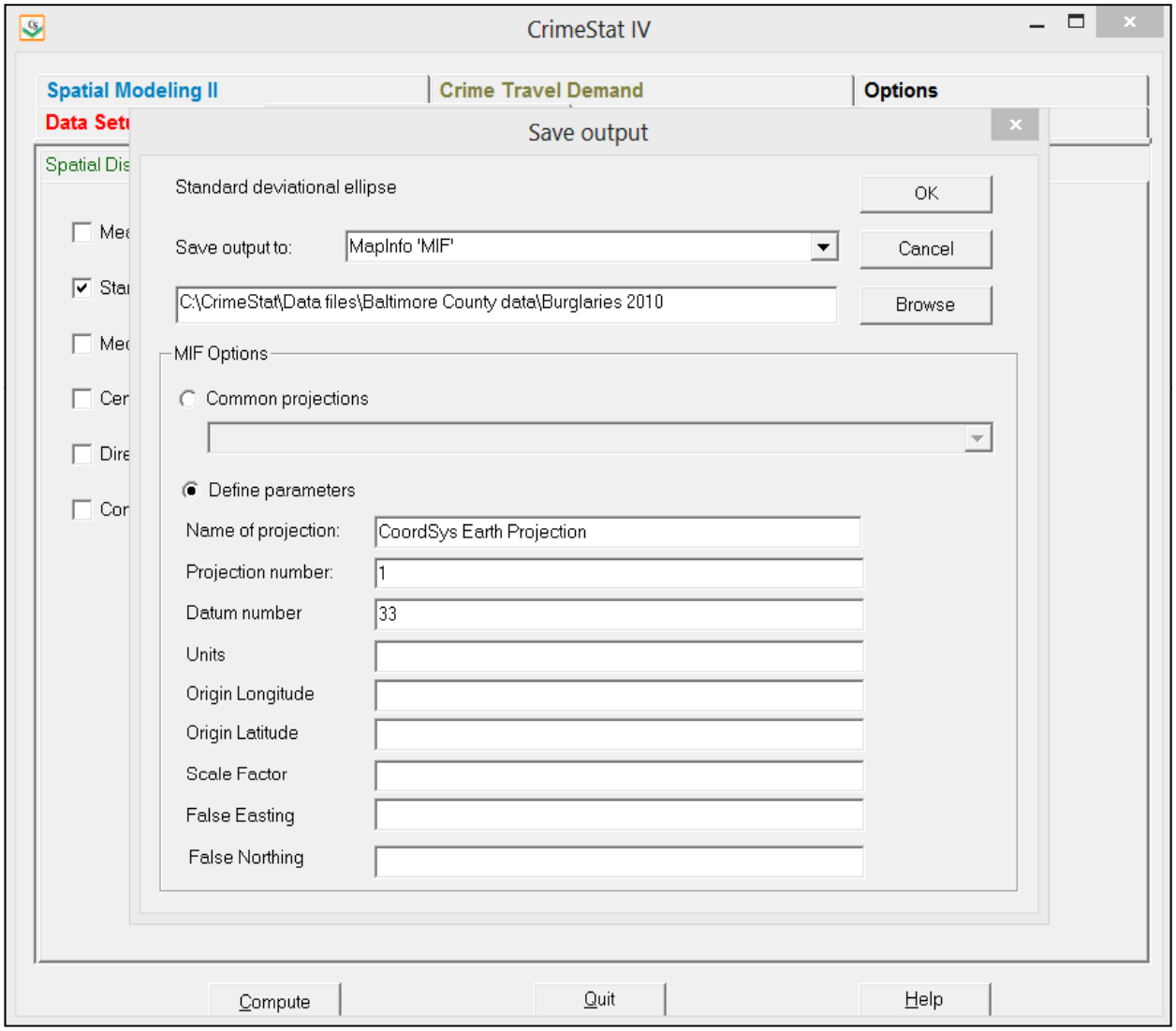


Figure 4.17:

MapInfo Output Options



Statistical Testing

While the current version of *CrimeStat* does not conduct statistical tests that compare two distributions, it is possible to conduct such tests. Appendix A presents a discussion of the statistical tests that can be used. Instead, the discussion here will focus on using the outputs of the routines without formal testing.

Decision-making Without Formal Tests

Formal significance testing has the advantage of providing a consistent inference about whether the difference in two distributions is likely or unlikely to be due to chance. Almost all formal tests compare the distribution of a statistic with that of a random distribution. However, police departments frequently have to make decisions based on small samples, in which case the formal tests are less useful than they would with larger samples. Still, the centrographic statistics calculated in *CrimeStat* can be useful and can help a police department make decision even in the absence of formal tests.

Examples of Centrographic Statistics

Example 1: June and July Auto Thefts in Precinct 11

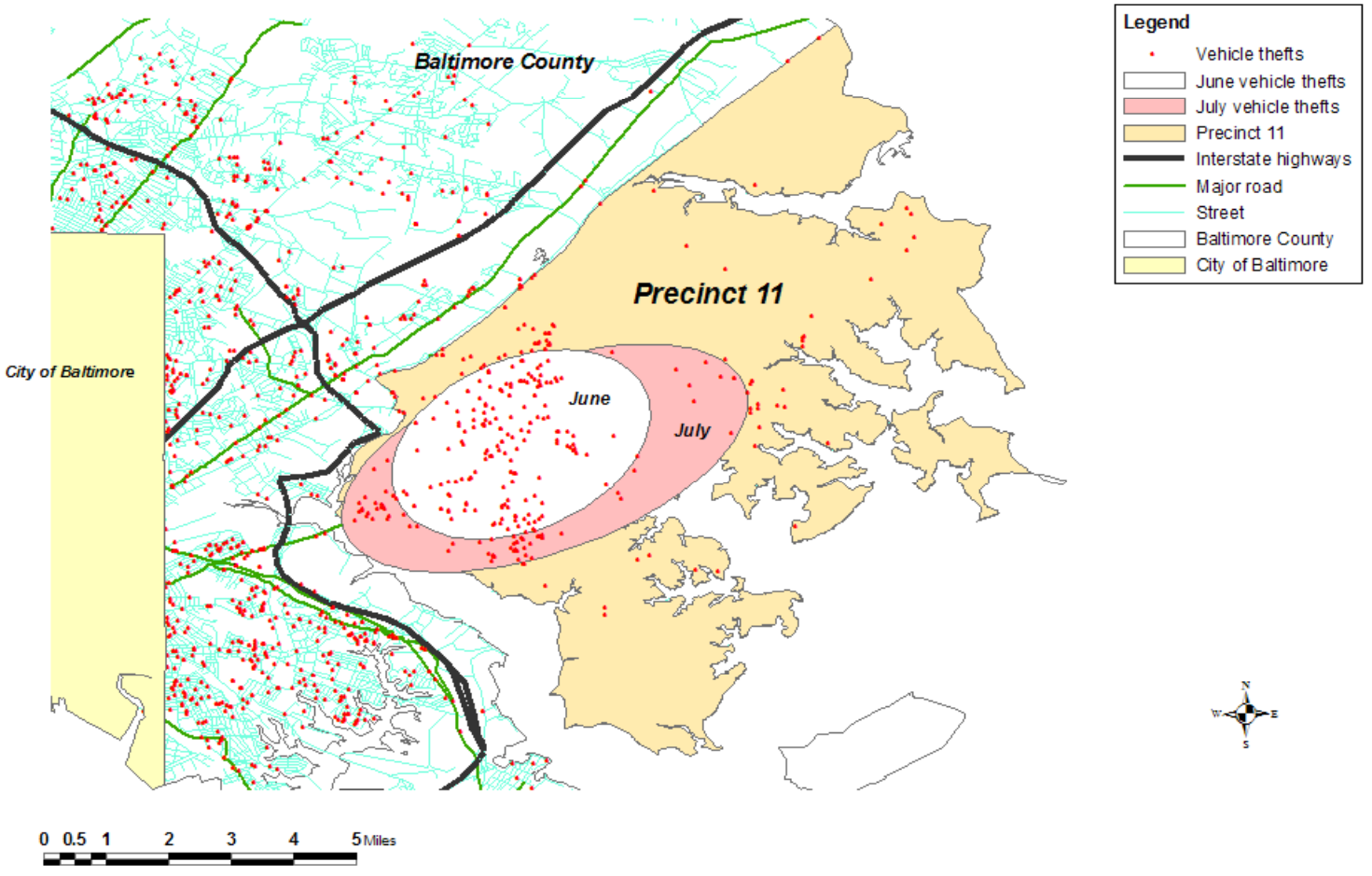
We want to illustrate the use of these statistics to make decisions with two examples. The first is a comparison of crimes in small geographical areas. In most metropolitan areas, most analysts will concentrate on particular sub-areas of the jurisdiction, rather than on the jurisdiction itself. In Baltimore County, for instance, analysis is done both for the jurisdiction as a whole as well as by individual precincts.

Below in Figure 4.18 are the standard deviational ellipses for 1996 auto thefts for June and July in Precinct 11 of Baltimore County. As can be seen, there was a spatial shift that occurred between June and July of that year, the result most probably of increased vacation travel to the Chesapeake Bay. While the comparison is very simple, involving looking at the graphical object created by *CrimeStat*, such a month to month comparison can be useful for police departments because it points to a shift in incident patterns, allowing the police department to reorient their patrol units.

Figure 4.18:

Vehicle Theft Change in Precinct 11

Standard Deviational Ellipses for June and July 1996



Example 2: Serial Burglaries in Baltimore City and Baltimore County

The second example illustrates a rash of burglaries that occurred on both sides of the border of Baltimore City and Baltimore County. On one hand there were ten residential burglaries that occurred on the western edge of the City/County border within a short time period of each other and, on the other hand, there were 13 commercial burglaries that occurred in the central part of the metropolitan areas. Both police departments suspected that these two sets were the work of a serial burglar (or group of burglars). What they were not sure about was whether the two sets of burglaries were done by the same individuals or by different individuals.

The number of incidents involved are too small for significance testing; only one of the parameters tested was significant and that could easily be due to chance. However, the police do have to make a guess about the possible perpetrator even with limited information. Let's use *CrimeStat* to try and make a decision about the distributions.

Figure 4.19 illustrates these distributions. The thirteen commercial burglaries are shown as squares while the ten residential burglaries are shown as triangles. Figure 4.20 plots the mean centers of the two distributions. They are close to each other, but not identical. An initial hunch

would suggest that the robberies are committed by two perpetrators (or groups of perpetrators), but the mean centers are not different enough to truly confirm this expectation.

Similarly, Figure 4.21 plots the center of minimum distance. Again, there is a difference in the distribution, but it is not great enough to truly rule out the single perpetrator theory. Figure 4.22 plots the raw standard deviations, expressed as a rectangle by *CrimeStat*. The dispersion of incidents overlaps to a sizeable extent and the area defined by the rectangle is approximately the same. In other words, the search area of the perpetrator or perpetrators is approximately the same. This might argue for a single perpetrator, rather than two. Figure 4.23 shows the standard distance deviation of the two sets of incidents. Again, there is sizeable overlap and the search radiuses are approximately the same.

Only with the standard deviational ellipse, however, is there a fundamental difference between the two distributions (figure 4.24). The pattern of commercial robberies is falling along a northeast-southwest orientation while that for residential robberies along a northwest-southeast axis. In other words, when the orientation of the incidents is examined, as defined by the standard deviational ellipse, there are two completely opposite patterns. Unless this difference can be explained by an obvious factor (e.g., the distribution of commercial establishments), it is probable that the two sets of robberies were committed by two different perpetrators (or groups of perpetrators).

Figure 4.19:

Profiling Serial Burglars

Incident Distribution of Two Serial Burglars

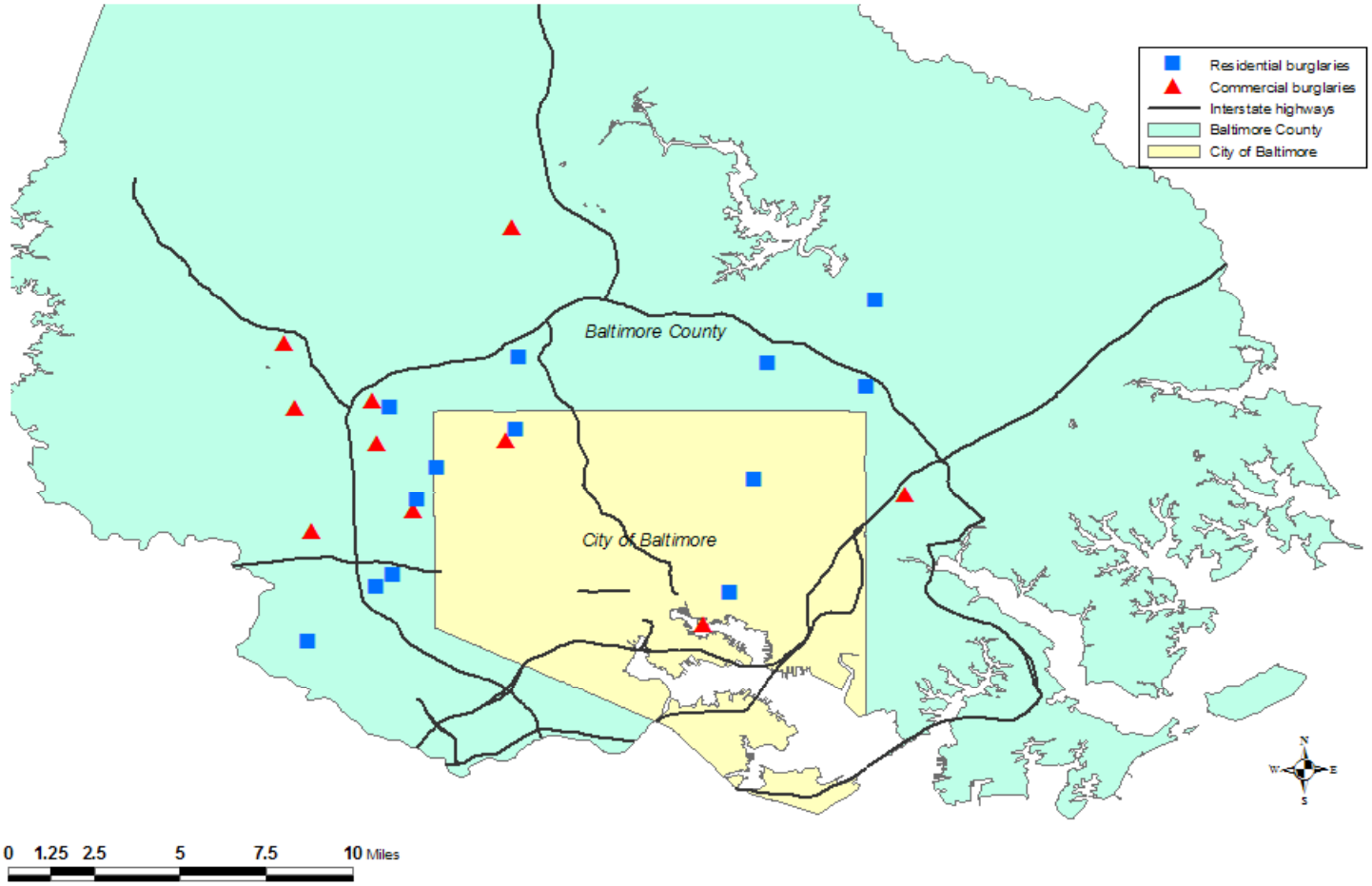


Figure 4.20:

Profiling Serial Burglars

Mean Center for Two Serial Burglars

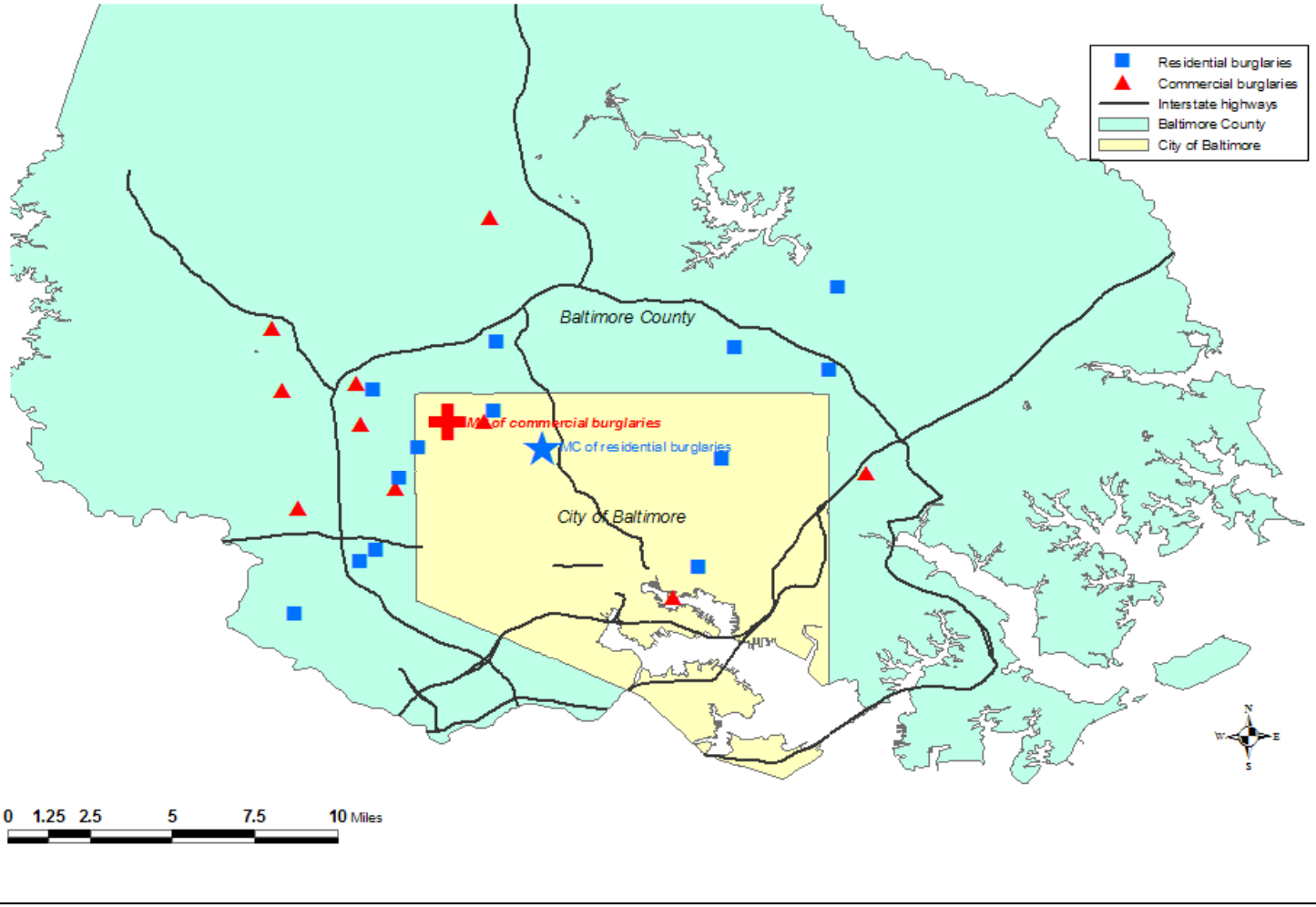


Figure 4.21:

Profiling Serial Burglars

Center of Minimum Distance for Two Serial Burglars

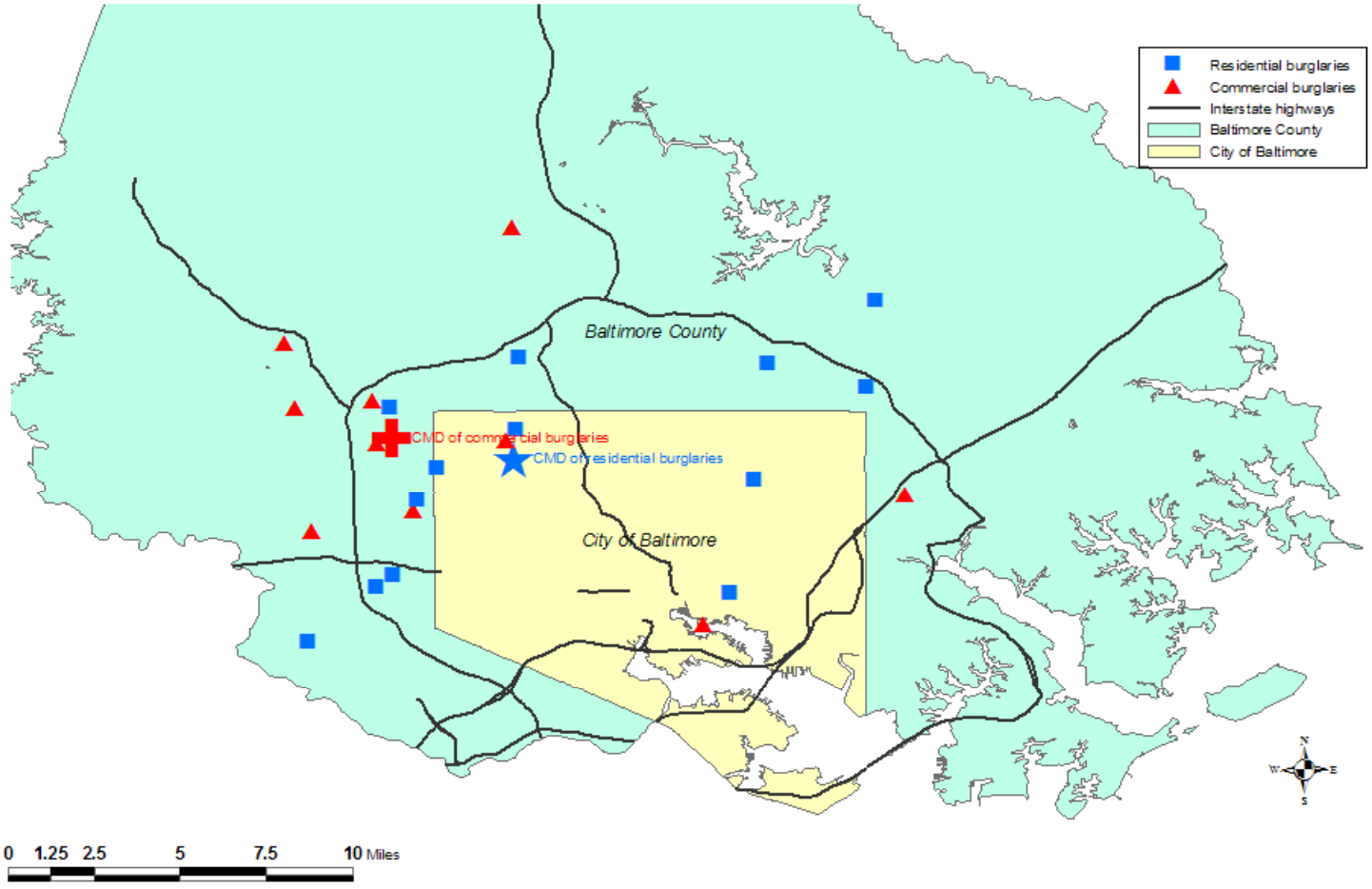


Figure 4.22:

Profiling Serial Burglars

Standard Deviations of Incidents for Two Serial Burglars

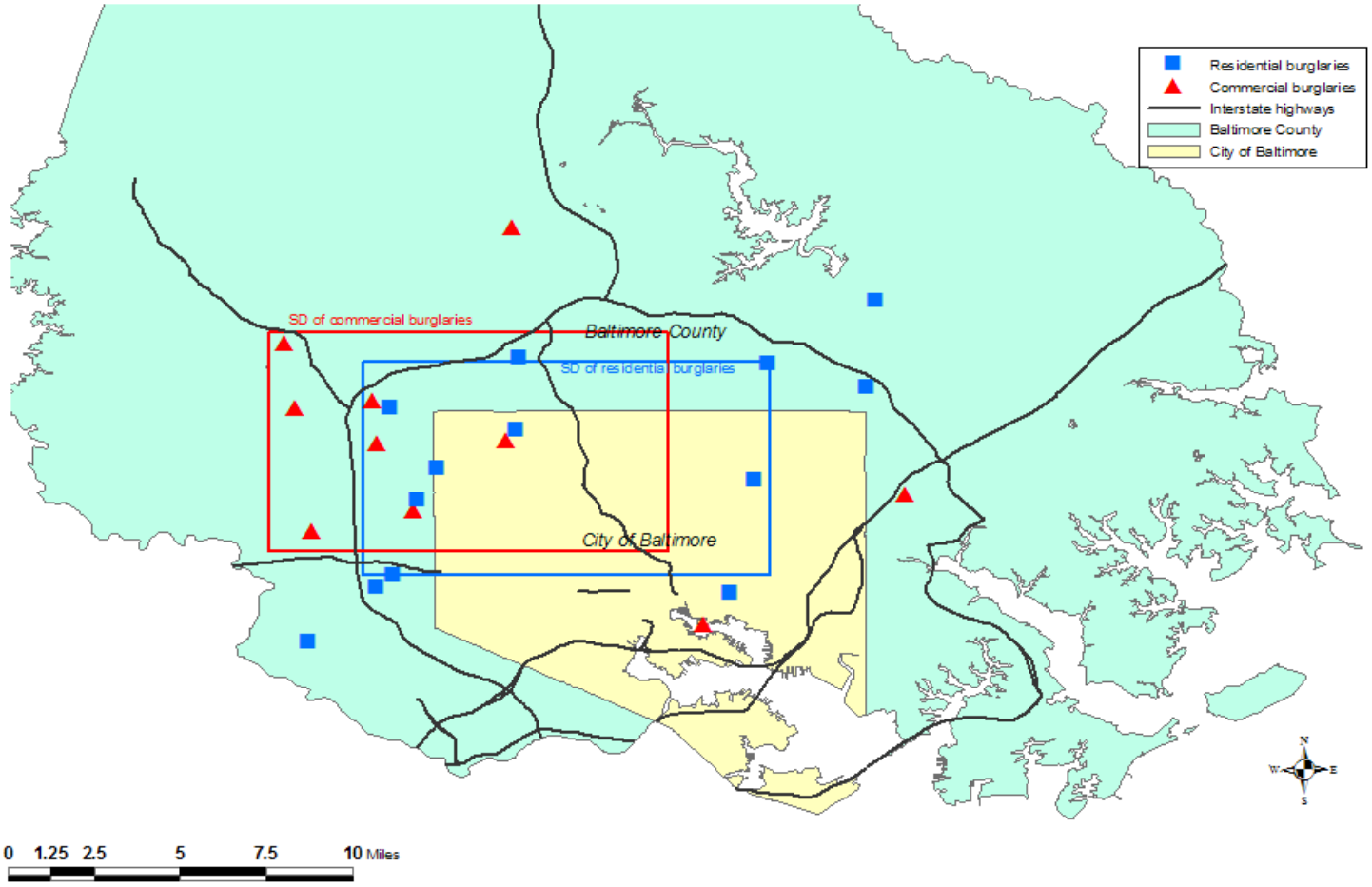


Figure 4.23:

Profiling Serial Burglars

Standard Distance Deviation of Incidents for Two Serial Burglars

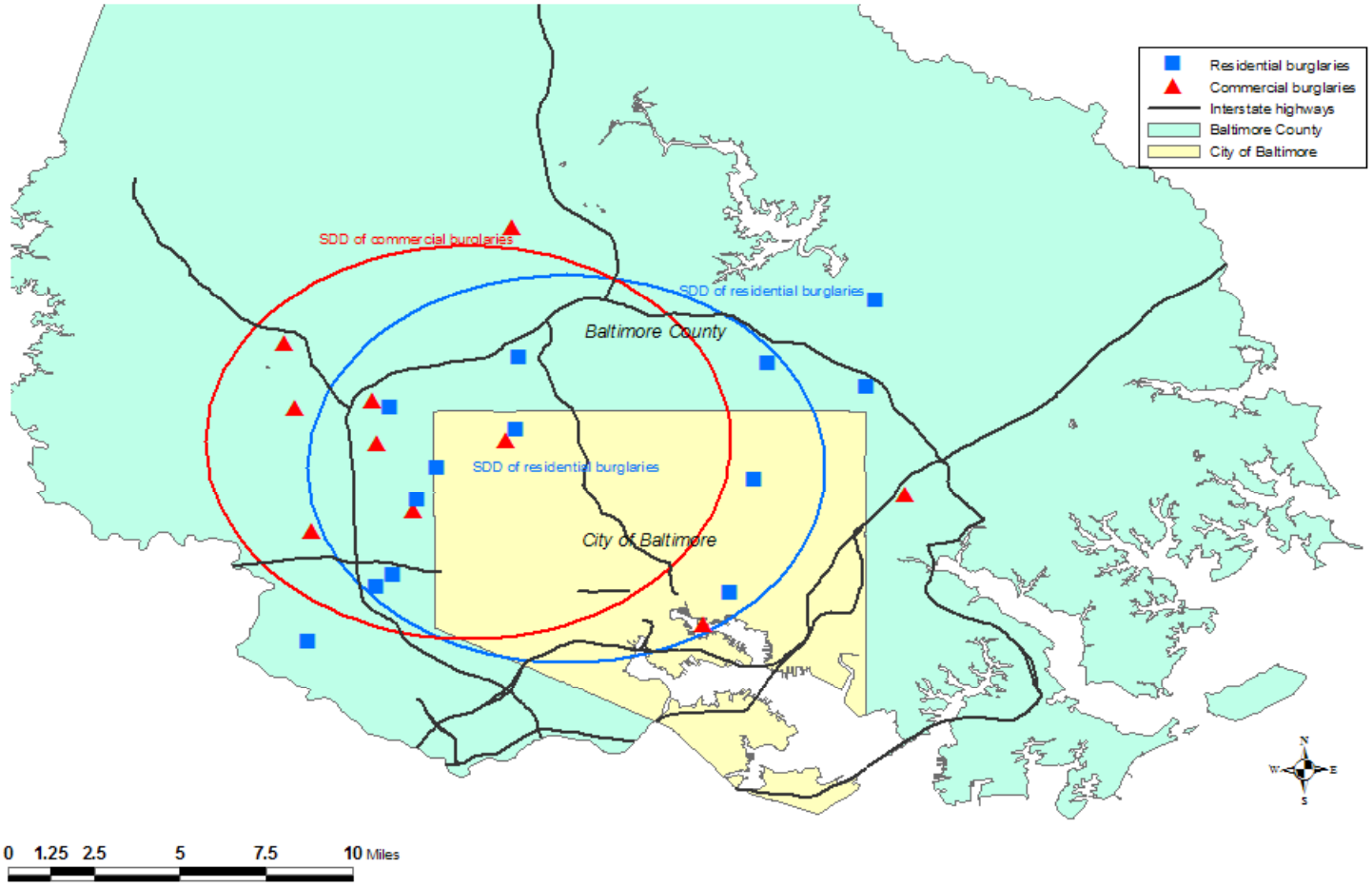
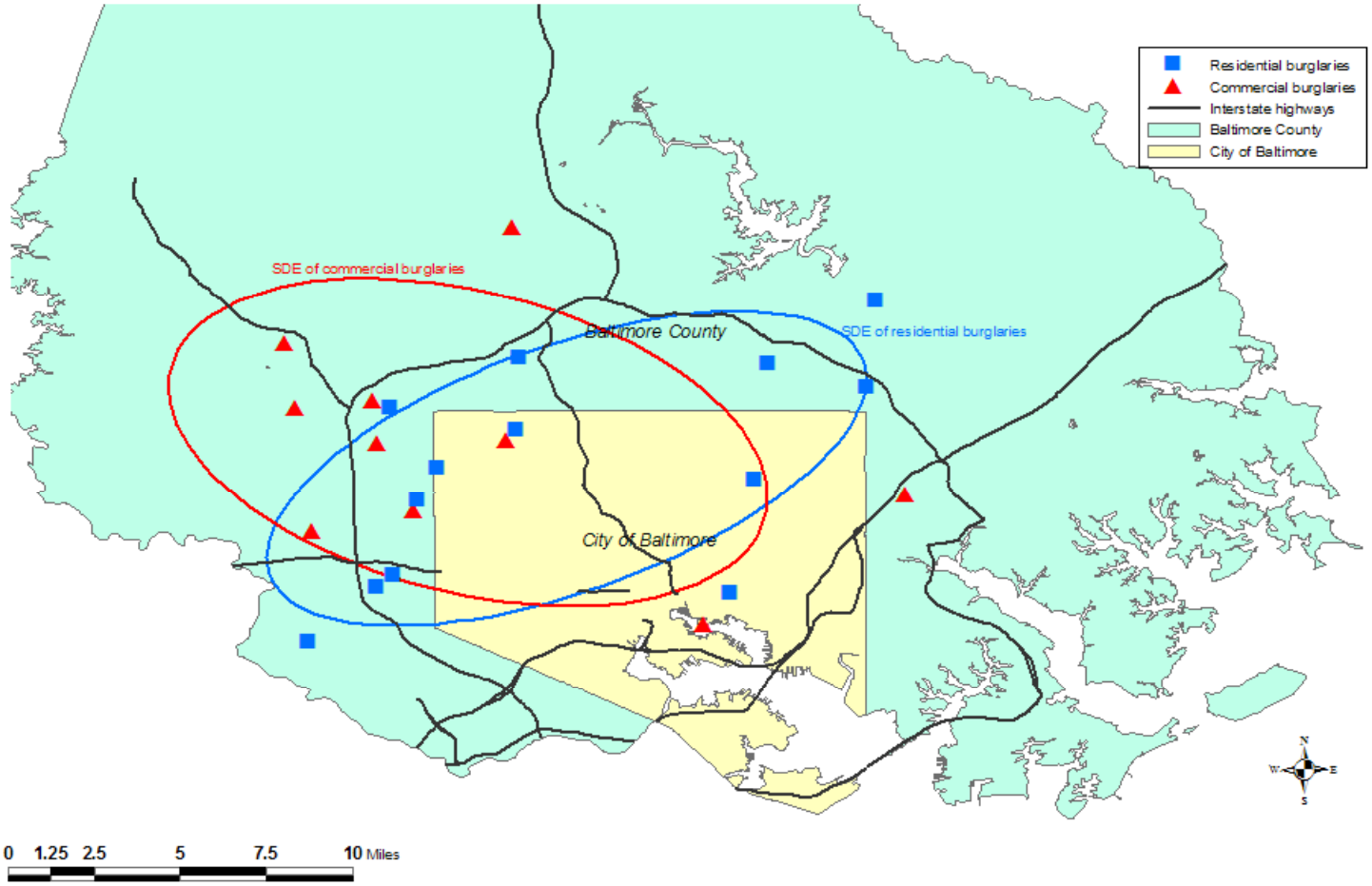


Figure 4.24:

Profiling Serial Burglars

Standard Deviational Ellipse of Incidents for Two Serial Burglars



Directional Mean and Variance

Centrographic statistics utilize the coordinates of a point, defined as an X and Y value on either a spherical or projected/Cartesian coordinate system. There is another type of metric that can be used for identifying incident locations, namely a **polar coordinate** system. A *vector* is a line with direction and length. In this system, there is a reference vector (usually 0^0 due North) and all locations are defined by angular deviations from this reference vector. By convention, angles are defined as deviations from 0^0, clockwise through 360^0. Note the measurement scale is a circle which returns back on itself (i.e. 0^0 is also 360^0). Point locations can be represented as vectors on a polar coordinate system.

With such a system, ordinary statistics cannot be used. For example, if there are five points which on the northern side of the polar coordinate system and are defined by their angular deviations as 0^0, 10^0, 15^0, 345^0, and 350^0 from the reference vector (moving clockwise from due North), the statistical mean will produce an erroneous estimate of 144^0. This vector would be southeast and will lie in an opposite direction from the distribution of points.

Instead, statistics have to be calculated by trigonometric functions. The input for such a system is a set of vectors, defined as angular deviations from the reference vector and a distance vector. Both the angle and the distance vector are defined with respect to an origin. The routine can calculate angles directly or can convert all X and Y coordinates into angles with a bearing from an origin. For reading angles directly, the input is a set of vectors, defined as angular deviations from the reference vector. *CrimeStat* calculates the mean direction and the circular variance of a series of points defined by their angles. On the primary file screen, the user must select Direction (angles) as the coordinate system.

If the angles are to be calculated from X/Y coordinates, the user must define an origin location. On the reference file page, the user can select among three origin points:

1. The lower-left corner of the data set (the minimum X and Y values). This is the default setting.

2. The upper-right corner of the data set (the maximum X and Y values); and

3. A user-defined point.

Users should be careful about choosing a particular location for an origin, either lower-left, upper-right or user-defined. If there is a point at that origin, *CrimeStat* will drop that case since any calculations for a point with zero distance are indeterminate. Users should check that

there is no point at the desired origin. If there is, then the origin should be adjusted slightly so that no point falls at that location (e.g., taking slightly smaller X and Y values for the lower-left corner or slightly larger X and Y values for the upper right corner).

The routine converts all X and Y points into an angular deviation from true North relative to the specified origin and a distance from the origin. The bearing is calculated with different formulae depending on the quadrant that the point falls within.

First Quadrant

With the lower-left corner as the origin, all angles are in the first quadrant. The clockwise angle, θ_i, is calculated by:

$$\theta = \arctan\left[\frac{Abs(X_i - X_O}{Abs(Y_i - Y_O)}\right] \tag{4.24}$$

where X_i is the X-value of the point, Y_i is the Y-value of the point, X_O is the X-value of the origin, and Y_O is the Y-value of the origin.

The angle, θ_i, is in radians and can be converted to polar coordinate degrees using:

$$\theta_{degrees} = \theta_{radians}\frac{180}{\pi} \tag{4.25}$$

Third Quadrant

With the upper-right corner as the origin, all angles are in the third quadrant. The clockwise angle, θ_i, is calculated by:

$$\theta_{radians} = \pi + \arctan\left[\frac{Abs(X_i - X_O)}{Abs(Y_i - Y_O)}\right] \tag{4.26}$$

where the angle, θ_i, is again in radians. Since there are 2π radians in a circle, π radians is 180^0. Again, the angle in radians can be converted into degrees with formula 4.25 above.

Second and Fourth Quadrants

When the origin is user-defined, each point must be evaluated as to which quadrant it is in. The second and fourth quadrants define the clockwise angle, θ_i, differently:

Second quadrant

$$\theta_{radians} = 0.5\pi + \arctan\left[\frac{Abs(Y_i - Y_O)}{Abs(X_i - X_O)}\right] \quad (4.27)$$

Fourth quadrant

$$\theta_{radians} = 1.5\pi + \arctan\left[\frac{Abs(Y_i - Y_O)}{Abs(X_i - X_O)}\right] \quad (4.28)$$

Once all X/Y coordinates are converted into angles, the mean angle is calculated.

Mean Angle

With either angular input or conversion from X/Y coordinates, the *Mean Angle* is the resultant of all individual vectors (i.e., points defined by their angles from the reference vector). It is an angle that summarizes the mean direction. Graphically, a *resultant* is the sum of all vectors and can be shown by laying each vector end to end. Statistically, it is defined as

$$Mean\ angle = \bar{\theta}_{radians} = Abs\left\{\arctan\left[\frac{\sum d_i sin\theta}{\sum d_i cos\theta}\right]\right\} \quad (4.29)$$

where the summation of sines and cosines is over the total number of points, i, defined by their angles, θ_i. Each angle, θ_i, can be weighted by the length of the vector, d_i. In an unweighted angle, d_i is assumed to be of equal length, 1. The absolute value of the ratio of the sum of the weighted sines to the sum of the weighted cosines is taken. All angles are in radians. In determining the mean angle, the quadrant of the resultant must be identified:

1. If $\Sigma(\sin\theta_i)>0$ and $\Sigma(\cos\theta_i)>0$, then $\overline{\theta}$ can be used directly as the mean angle.

2. If $\Sigma(\sin\theta_i)>0$ and $\Sigma(\cos\theta_i)<0$, then $\overline{\theta}$ is $\pi/2 + \theta$.

3. If $\Sigma(\sin\theta_i)<0$ and $\Sigma(\cos\theta_i)<0$, then $\overline{\theta}$ is $\pi + \theta$.

4. If $\Sigma(\sin\theta_i)<0$ and $\Sigma(\cos\theta_i)>0$, then $\overline{\theta}$ is $1.5\pi + \theta$.

Formulas 4.26, 4.27, 4.28 and 4.29 above are then used to convert the directional mean back to an X/Y coordinate, depending on which quadrant it falls within.

Circular Variance

The dispersion (or variance) of the angles are also defined by trigonometric functions. The unstandardized variance, R, is sometimes called the *sample resultant length* since it is the resultant of all vectors (angles):

$$R = \sqrt{[\sum(d_i \, sin\theta_i)^2] + [\sum(d_i cos\theta_i)^2]} \tag{4.30}$$

where d_i is the length of vector, i, with an angle (bearing) for the vector of θ_i. For the unweighted sample resultant, d_i is 1.

Because R increases with sample size, it is standardized by dividing by N to produce a *mean resultant length*:

$$\bar{R} = \frac{R}{N} \tag{4.31}$$

where N is the number points (sample size).

Finally, the average distance from the origin, D, is calculated and the *circular variance* is calculated by:

$$Circular\ variance = \frac{1}{D}\left(D - \frac{R}{N}\right) = \frac{D-\bar{R}}{D} = 1 - \frac{\bar{R}}{D} \tag{4.32}$$

This is the standardized variance which varies from 0 (no variability) to 1 (maximum variability). The details of the derivations can be found in Burt and Barber (1996) and Gaile and Barber (1980).

Mean Distance

The mean distance, $\bar{d}$, is calculated directly from the X and Y coordinates. It is identified in relation to the defined origin.

Directional Mean

The directional mean is calculated as the intersection of the mean angle and the mean distance. It is not a unique position since distance and angularity are independent dimensions. Thus, the directional mean calculated using the minimum X and minimum Y location as the reference origin (the 'lower left corner') will yield a different location from the directional mean

calculated using the maximum X and maximum Y location as the origin (the 'upper right corner'). There is a weighted and unweighted directional mean. Though *CrimeStat* calculates the location, users should be aware of the non-uniqueness of the location. The unweighted directional mean can be output with a 'Dm' prefix. The weighted directional mean is not output.

Triangulated Mean

The triangulated mean is defined as the intersection of the two vectors, one from the lower-left corner of the study area (the minimum X and Y values) and the other from the upper-right corner of the study area (the maximum X and Y values). It is calculated by estimating mean angles from each origin (lower left and upper right corners), translating these into equations, and finding the point at which these equations intersect (by setting the two functions equal to each other).

Directional Mean Output

The directional mean routine outputs nine statistics:

1. The sample size;
2. The unweighted mean angle;
3. The weighted mean angle;
4. The unweighted circular variance;
5. The weighted circular variance;
6. The mean distance;
7. The intersection of the mean angle and the mean distance;
8. The X and Y coordinates for the triangulated mean; and
9. The X and Y coordinates for the weighted triangulated mean.

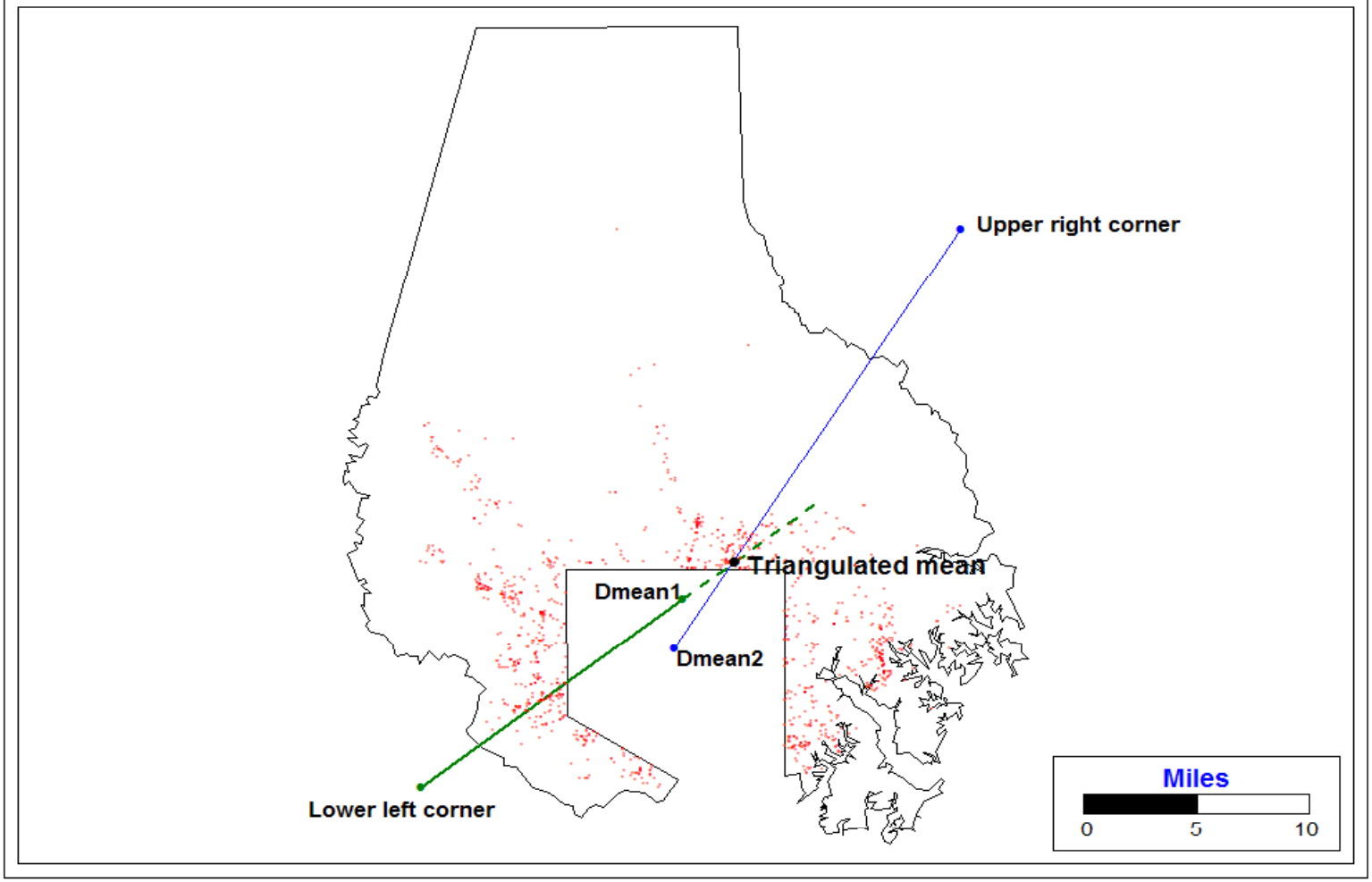
The directional mean and triangulated mean can be saved as *ArcGIS* 'shp', *MapInfo* 'mif', *Google Earth* 'kml, or various Ascii files. The unweighted directional mean - the intersection of the mean angle and the mean distance, is output with the prefix 'Dm' while the unweighted triangulated mean location is output with a 'Tm' prefix. The weighted triangulated mean is output with a 'TmWt' prefix. See the example below.

Figure 4.25 shows the unweighted triangular mean for 1996 Baltimore County robberies and compares it to the two directional means calculated using the lower-left corner (Dmean1) and the upper-right corner (Dmean2) respectively as origins. As can be seen, the two directional means fall at different locations. Lines have been drawn from each origin point to their

Figure 4.25:

Triangulated Mean for Baltimore County Robberies

Defined by the Intersection of Two Mean Angles



respective directional means and are extended until they intersect. As seen, the triangulated mean falls at the location where the two vectors (i.e., mean angles) intersect.

Because the triangulated mean is calculated with vector geometry, it will not necessarily capture the central tendency of a distribution. Asymmetrical distributions can cause it to be placed in peripheral locations. On the other hand, if the distribution is relatively balanced in each direction, it can capture the center of orientation perhaps better than other means, as figure 4.25 shows. Appendix A includes a discussion of how to formally test the mean direction between two different distributions.

Convex Hull

The convex hull is a boundary drawn around the distribution of points. It is a relatively simple concept, at least on the surface. Intuitively, it represents a polygon that circumscribes all the points in the distribution such that no point lies outside of the polygon.

The complexity comes because there are different ways to define a convex hull. The most basic algorithm is the *Graham scan* (Graham, 1972). Starting with one point known to be on the convex hull, typically the point with the lowest X coordinate, the algorithm sorts the remaining points in angular order around this in a counterclockwise manner. If the angle formed by the next point and the last edge is less than 180 degrees, then that point is added to the hull. If the angle is greater than 180 degrees, then the chain of nodes starting from the last edge must be deleted. The routine proceeds until the hull closes back on itself (de Berg, van Kreveld, Overmans, and Schwarzkopf, 2000).

Many alternative algorithms have been proposed. Among these are the 'gift wrap' (Chand and Kapur, 1970; Skiena, 1997), the Quick Hull, the "Divide and conquer" (Preparata and Hong, 1977), and the incremental (Kallay, 1984) algorithms. Even more complexity has been introduced by the mathematics of fractals where an almost infinite number of borders could be defined (Lam and De Cola, 1993). In most implementations, though, a simplified algorithm is used to produce the convex hull.

CrimeStat implements the 'gift wrap' algorithm. Starting with the point with the lowest Y coordinate, A, it searches for another point, B, such that all other points lie to the left of the line AB. It then finds another point, C, such that all remaining points lie to the left of the line BC. It continues in this way until it reaches the original point A again. It is like 'wrapping a gift' around the outside of the points.

The routine outputs three statistics:

1. The sample size;
2. The number of points in the convex hull
3. The X and Y coordinates for each of the points in the convex hull

The convex hull can be saved as *ArcGIS* 'shp', *MapInfo* 'mif', *Google Earth* 'kml, or various Ascii files with a 'Chull' prefix.

Figure 4.26 shows the convex hull of Baltimore County robberies for 1996. As seen, the hull occupies a relatively smaller part of Baltimore County. Figure 4.27, on the other hand shows the convex hull of 1996 Baltimore County burglaries. As seen, the convex hull of the burglaries cover a much larger area than for the robberies.

Uses and Limitations of the Convex Hull

A convex hull can be useful for displaying the geographical extent of a distribution. Simple comparisons, such as in Figures 4.26 and 4.27, can show whether one distribution has a greater extent than another. Further, as we shall see, a convex hull can be useful for describing the geographical spread of a crime hot spot, essentially indicating where the crimes are distributed.

On the other hand, a convex hull is vulnerable to extreme values. If one incident is isolated, the hull will of necessity be large. The mean center, too, is influenced by extreme values but not to the same extent since it averages the location of all points. The convex hull, on the other hand, is defined by the most extreme points. A comparison of different crime types or the same crime type for different years using the convex hull may only show the variability of the extreme values, rather than any central property of the distribution. Therefore, caution must be used in interpreting the meaning of a hull.

Figure 4.26:

Convex Hull of Baltimore County Robberies: 1996

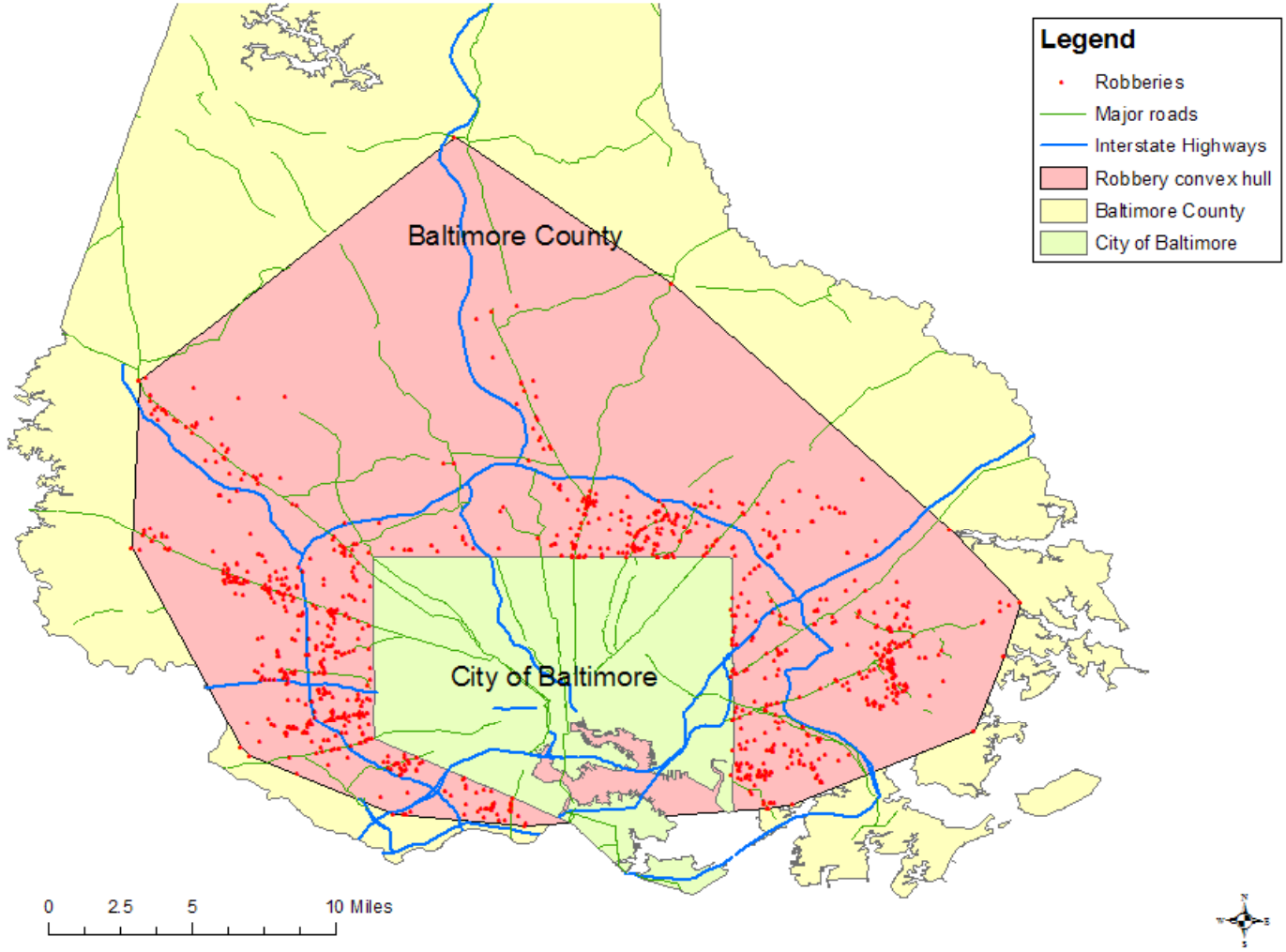
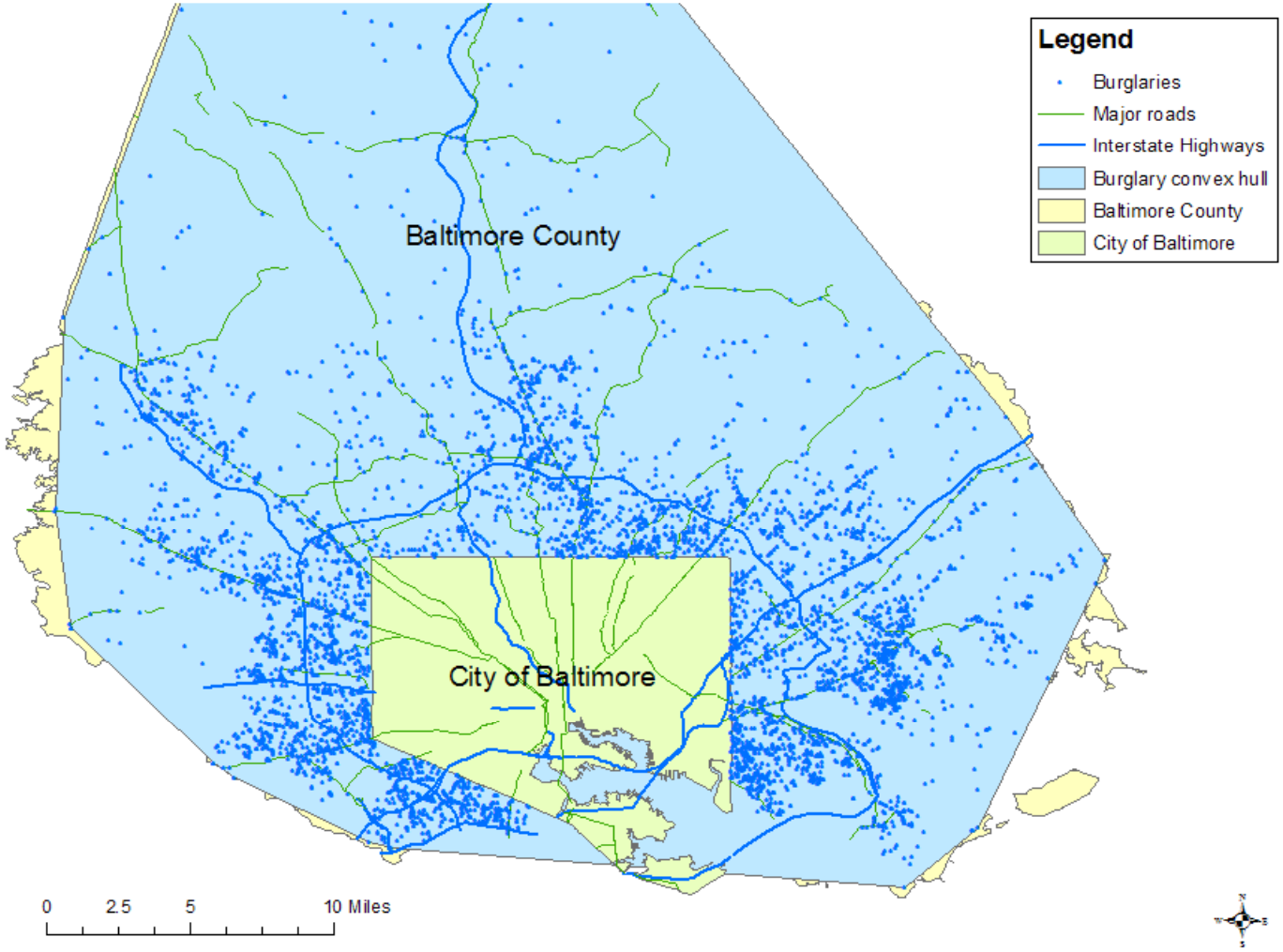


Figure 4.27:

Convex Hull of Baltimore County Burglaries: 1996



References

Bachi, R. (1957). *Statistical Analysis of Geographical Series*. Central Bureau of Statistics, Kaplan School, Hebrew University: Jerusalem.

Burt, J. E. & Barber, G. M. (1996). *Elementary Statistics for Geographers* (second edition). The Guilford Press: New York.

Chand, D & Kapur, S. (1970). An algorithm for convex polytopes. *J. ACM*, 17, 78-86.

Cromley, R. G. (1992). *Digital Cartography*. Prentice Hall: Englewood Cliffs, NJ.

de Berg, M.; van Kreveld, M.; Overmans, M.; & Schwarzkopf, O. (2000). Convex hulls: mixing things. In *Computational Geometry: Algorithms and Applications*, 2nd rev. ed. Springer-Verlag: Berlin, 235-250.

Ebdon, D. (1988). *Statistics in Geography* (second edition with corrections). Blackwell: Oxford.

Everitt, B. S. (2011). *Cluster Analysis* (5th edition). J. Wiley: London.

Furfey, P. H. (1927). A note on Lefever's 'Standard deviational ellipse'. *American Journal of Sociology*. XXIII, 94-98.

Gaile, G. L. & Burt, J. E. (1980). *Directional Statistics*. Concepts and Techniques in Modern Geography No. 25. Institute of British Geographers, Norwich, England: Geo Books.

Graham, R (1972). An efficient algorithm for determining the convex hull of a finite planar point set. *Info. Proc. Letters*, 1, 132-133.

Hammond, R., & McCullagh, P. (1978). *Quantitative Techniques in Geography: An Introduction*. Second Edition. Clarendon Press: Oxford, England.

Hultquist, J., Brown, L. & Holmes, J. (1971). Centro: a program for centrographic measures. Discussion paper no. 21, Department of Geography, Ohio State University: Columbus, OH.

Kallay, M. (1984). The complexity of incremental convex hull algorithms in Rd, *Info. Proc. Letters* 19, 197.

References (continued)

Kuhn, H. W. & Kuenne, R. E. (1962). An efficient algorithm for the numerical solution of the generalized Weber problem in spatial economics, *Journal of Regional Science* 4, 21-33.

Lam, N. Siu-ngan & De Cola, L. (1993). *Fractals in Geography*. The Blackburn Press: Caldwell, NJ.

Langworthy, R. H. & Jefferis, E. (1998). The utility of standard deviational ellipses for project evaluation. Discussion paper, National Institute of Justice: Washington, DC.

Lefever, D. (1926). Measuring geographic concentration by means of the standard deviational ellipse. *American Journal of Sociology*, 32(1): 88-94.

Neft, D. S. (1962). *Statistical Analysis for Areal Distributions*. Ph.D. dissertation, Columbia University: New York.

Preparata, F. & Hong, S. J. (1977). Convex hulls of finite sets of points in two and three dimensions, *Comm. ACM,* 20, 87-93.

Skiena, S. S. (1997). Convex hull. §8.6.2 in *The Algorithm Design Manual*. Springer-Verlag: New York, 351-354.

Stephenson, L. (1980). Centrographic analysis of crime. In D. George-Abeyie & K. Harries (eds), *Crime, A Spatial Perspective*, Columbia University Press: New York.

Endnotes

i. *CrimeStat*'s implementation of the Kuhn and Kuenne algorithm is as follows (from Burt and Barber, 1996, 112-113):

1. Let t be the number of the iteration. For the first iteration only (i.e., t=1) the weighted mean center is taken as the initial estimate of the median location, X_t and Y_t.

2. Calculate the distance from each point, i, to the current estimate of the median location, d_{ict}, where i is a single point and ct is the current estimate of the median location during iteration t.

   a. If the coordinates are spherical, then Great Circle distances are used.

   b. If the coordinates are projected, then Euclidean distances are used.

3. Weight each case by a weight, W_i, and calculate:

   $$K_{it} = W_i e^{-d_{ict}}$$

   where **e** is the base of the natural logarithm(2.7183..).

   a. If no weights are defined in the primary file, W_i is assumed to be 1.

   b. If weights are defined in the primary file, W_i takes their values.

   Note that as the distance, d_{ict}, approaches 0, then $e^{-d_{ict}}$ becomes 1.

4. Calculate a new estimate of the center of minimum distance from:

   $$X^{t+1} = \frac{\sum K_{it} X_i}{\sum K_{it}} \qquad \text{for i=1…n}$$

   $$Y^{t+1} = \frac{\sum K_{it} Y_i}{\sum K_{it}} \qquad \text{for i=1…n}$$

   where X_i and Y_i are the coordinates of point i (either lat/lon for spherical or feet or meters for projected).

5. Check to see how much change has occurred since the last iteration

$$\text{Abs}|X^{t+1} - X^t| \leq 0.000001$$

$$\text{Abs}|Y^{t+1} - Y^t| \leq 0.000001$$

a. If either the X or Y coordinates have changed by greater than 0.000001 between iterations, substitute X^{t+1} for X^t and Y^{t+1} for Y^t and repeat steps B through D.

b. If *both* the change in X and the change in Y is less than or equal to 0.000001, then the estimated X_t and Y_t coordinates are taken as the center of median distance.

ii. Formulas for the new axes provided by Ebdon (1988) and Cromley (1992) yield standard deviational ellipses that are too small, for two different reasons. First, they produce transformed axes that are too small. If the distribution of points is random and even in all directions, ideally the standard deviational ellipse should be equal to the standard distance deviation, since $S_x = S_y$. The formula used here has this property. Since the formula for the standard distance deviation is (equation 4.8):

$$SDD = \sqrt{\left[\frac{\Sigma(X_i-\bar{X})^2+(Y_i-\bar{y})^2}{N-2}\right]}$$

If $S_x = S_y$, then $\Sigma(X_i - X)^2 = \Sigma(Y_i - Y)^2$, therefore:

$$SDD = \sqrt{2\frac{\Sigma(X_i-\bar{X})^2}{N-2}}$$

Similarly, the formulas for the transformed axes are (4.9, 4.10):

$$s_X = \sqrt{2\frac{\Sigma[\left(X_i-\bar{X}\right)\cos\theta-(Y_i-\bar{y})\sin\theta]^2}{N-2}}$$

$$s_Y = \sqrt{2\frac{\Sigma[\left(X_i-\bar{X}\right)\sin\theta+(Y_i-\bar{y})\cos\theta]^2}{N-2}}$$

However, if $S_x = S_y$, then $\theta = 0$, $\cos \theta = 1$, $\text{Sin } \theta = 0$ and, therefore:

$$s_X = s_Y = \sqrt{2 \frac{\sum (X_i - \bar{X})^2}{N-2}}$$

which is the same as for the standard distance deviation (SDD) under the same conditions. The formulas used by Ebdon (1988) and Cromley (1992) produce axes which are $\sqrt{2}$ times too small.

The second problem with the Ebdon and Cromley formulas is that they do not correct for degrees of freedom and, hence, produce too small a standard deviational ellipse. Since there are two constants in each equation, $\bar{X}$ and $\bar{Y}$, then there are only N-2 degrees of freedom. The cumulative effect of using transformed axes that are too small and not correcting for degrees of freedom yields a much smaller ellipse than that used here.

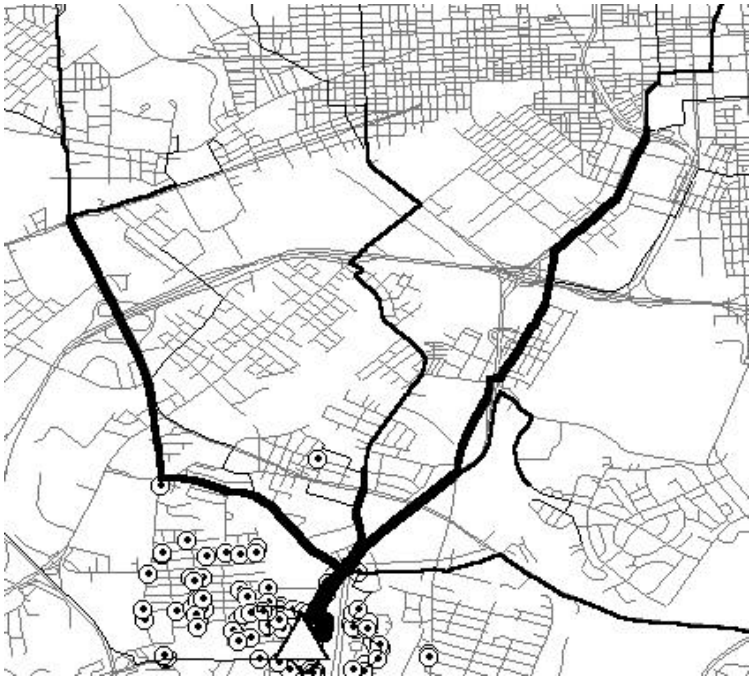
Attachments

Using Spatial Measures of Central Tendency with Network Analyst to Identify Routes Used by Motor Vehicle Thieves

Philip R. Canter
Baltimore County Police Department
Towson, Maryland

Motor vehicle thefts have been steadily declining countywide over the last 5 years, but one police precinct in southwest Baltimore County was experiencing significant increases over several months. Cases were concentrated in several communities, but directed deployment and saturated patrols had minimal impact. In addition to increasing patrols in target communities, the precinct commander was interested in deploying police on roads possibly used by motor vehicle thieves. Police analysts had addresses for theft and recovery locations; it was a matter of using the existing highway network to connect the two locations.

To avoid analyzing dozens of paired locations, analysts decided to set up a database using one location representing the origin of motor vehicle thefts for a particular community. The origin was computed using *CrimeStat*'s median center for motor vehicle theft locations reported for a particular community. The median center is the position of minimum average travel and is less affected by extreme locations compared to the arithmetic mean center. The database consisted of the median center paired with a recovery location. Using Network Analyst, a least-effort route was computed for cases reported by community. A count was assigned to each link along a roadway identified by Network Analyst. Analysts used the count to thematically weight links in ArcView. The precinct commander deployed resources along these routes with orders to stop suspicious vehicles. This operation resulted in 27 arrests, and a reduction in motor vehicle thefts.

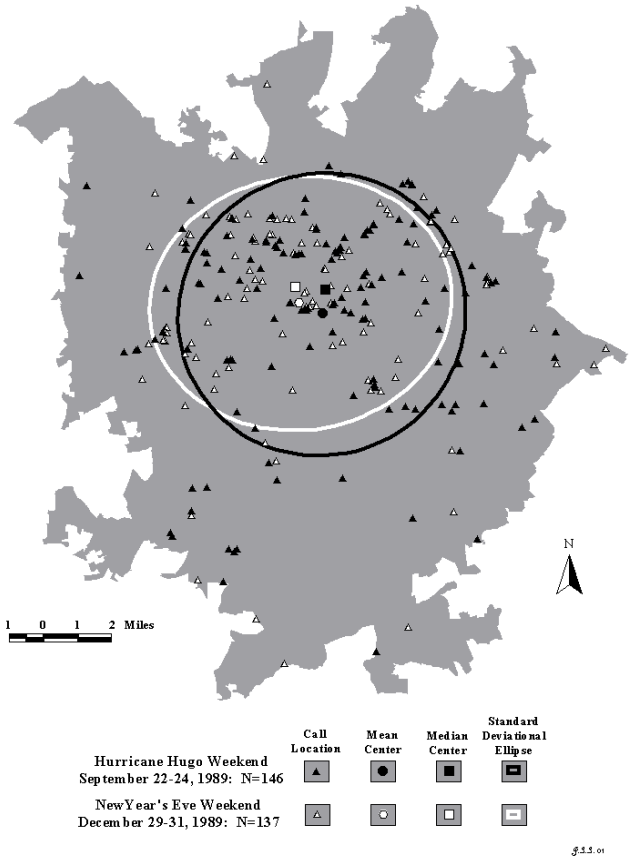


Centrographic Analysis
Man With A Gun Calls For Service
Charlotte, N.C., 1989

James L. LeBeau
Administration of Justice
Southern Illinois University – Carbondale

Hurricane Hugo arrived on Friday, September 22, 1989 in Charlotte, North Carolina. That weekend experienced the highest counts of *Man With A Gun* calls for service for the year. The locations of the calls during the Hugo Weekend are compared with the following New Year's Eve weekend.

CrimeStat was used to compare the two weekends. Compared to the New Year's Eve weekend: 1) Hugo's mean and median centers are more easterly; 2) Hugo's ellipse is larger and more circular; and 3) Hugo's ellipse shifts more to the east and southeast. The abrupt spatial change of *Man With A Gun* calls during a natural disaster might indicate more instances of defensive gun use for protection of property.



Chapter 5:

Spatial Autocorrelation Statistics

Ned Levine[1]
Ned Levine & Associates
Houston, TX

[1] The author would like to thank Dr. David Wong for help with the Getis-Ord 'G' and local Getis-Ord statistics.

Table of Contents

Table of Contents (continued)

Chapter 5:

Spatial Autocorrelation Statistics

This chapter discusses statistics for describing spatial autocorrelation that are applicable to zonal data. A good grasp of basic statistics is a requirement for reading this chapter. Figure 5.1 shows the Spatial Autocorrelation page within the Spatial Description section. This includes global tests of spatial autocorrelation for zone data or point data in which an attribute can be associated with the coordinates. The section includes six tests for global spatial autocorrelation:

1. Moran's "I" statistic
2. Geary's "C" statistic
3. Getis-Ord "G" statistic
4. Moran Correlogram
5. Geary Correlogram
6. Getis-Ord Correlogram

These indices would typically be applied to zonal data where an attribute value can be assigned to each zone. Six spatial autocorrelation indices are calculated. All require an intensity variable in the Primary File.

The discussion in the chapter will concentrate on defining the indices and demonstrating how they can be used. Specific instructions for running the routines are given at the end of the chapter while detailed information is provided in Chapter 2.

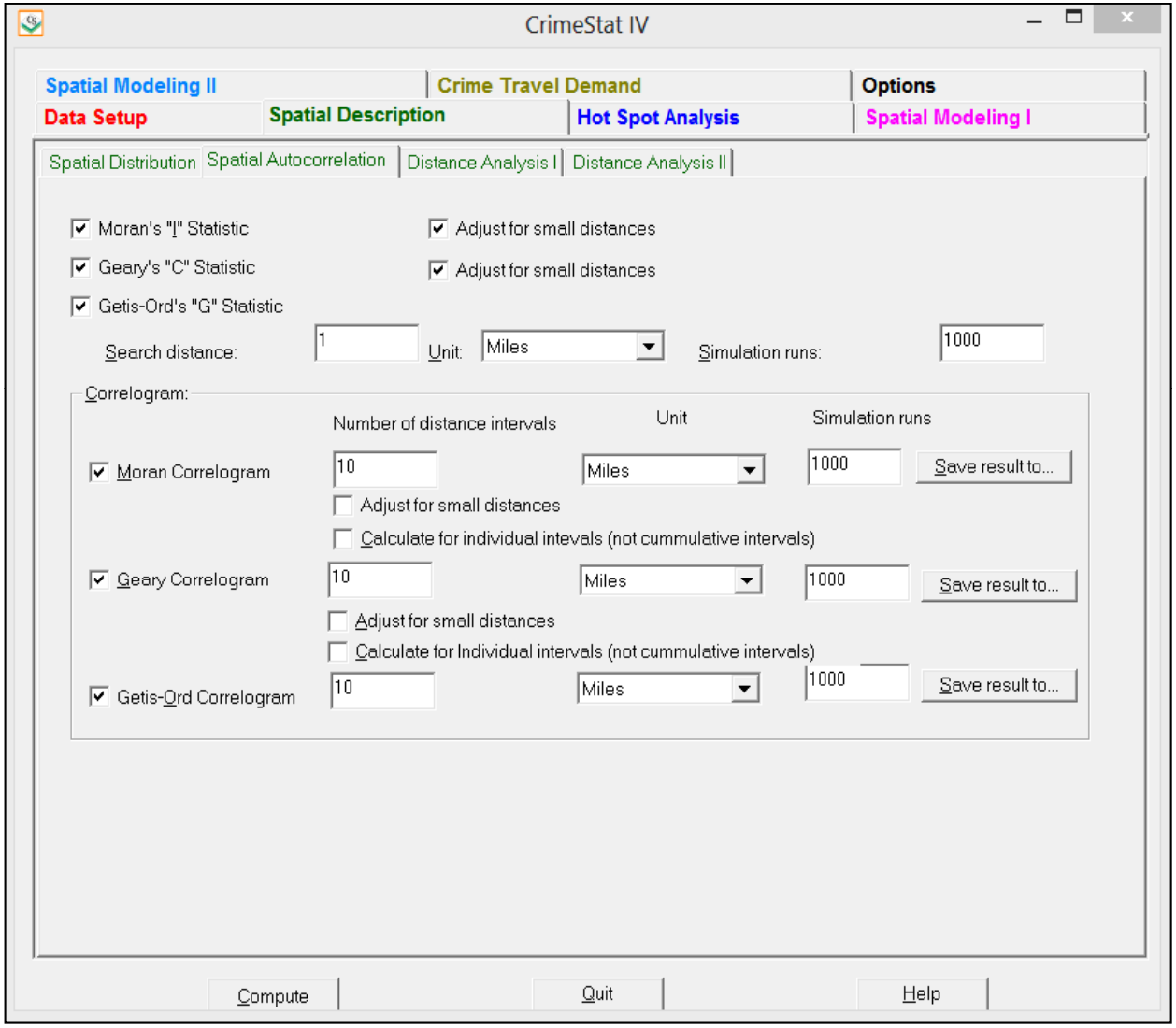
Spatial Autocorrelation

The concept of *spatial autocorrelation* is one of the most important in spatial statistics in that it implies a lack of spatial *independence*. Classical statistics assumes that observations are independently chosen and are spatially unrelated to each other. The intuitive concept is that the location of an incident (e.g., a street robbery, a burglary) is unrelated to the location of any other incident. The opposite condition - spatial autocorrelation, is a spatial arrangement of incidents such that the locations where incidents occur are related to each other; that is, they are not statistically independent of one another. In other words, spatial autocorrelation is a spatial arrangement where spatial independence has been violated.

When events or people or facilities are clustered together, we refer to this arrangement as *positive* spatial autocorrelation. Conversely, an arrangement where people, events or facilities

Figure 5.1:

Spatial Autocorrelation Statistics



are extremely dispersed is referred to as *negative* spatial autocorrelation; it is a rarer arrangement, but does exist (Levine, 1999).

However, many, if not most, social phenomena are spatially autocorrelated. In any large metropolitan area, most social characteristics and indicators, such as the number of persons, income levels, ethnicity, education, employment, and the location of facilities are not spatially independent, but tend to be concentrated.

There are practical consequences. Police and crime analysts know from experience that incidents frequently cluster together in what are called 'hot spots'. This non-random arrangement can allow police to target certain areas or zones where there are concentrations of crimes as well as prioritize areas by the intensity of incidents. Many of the incidents are committed by the same individuals. For example, if a particular neighborhood had a concentration of street robberies over a time period (e.g., a year), many of these robberies will have been committed by the same perpetrators. Statistical dependence between events often has common causes.

Statistically, however, non-spatial independence indicates that many statistical tools and inferences are inappropriate. For example, the use of a correlation coefficient or Ordinary Least Squares regression (OLS) model to predict a consequence (e.g., correlates or predictors of burglaries) assumes observations are randomly selected. If, however, the observations are spatially clustered, the estimates obtained from the correlation coefficient or OLS estimator will be biased and overly precise. The coefficients will be biased because areas with a higher concentration of events will have a greater impact on the model estimate and precision will be overestimated because concentrated events tend to have fewer independent observations than are being assumed. The spatial autocorrelation concept underlies almost all of *CrimeStat*'s spatial statistics tools.

Indices of Spatial Autocorrelation

Assigning Point Data to Zones

If a user has information on the location of individual events (e.g., robberies), then it is better to utilize that information with the point statistics discussed in Chapter 4 and the hot spot tools that will be discussed in Chapters 7 and 8. The individual-level information will contain all the uniqueness of the events.

However, sometimes it is not possible to analyze data at the individual level. The user may need to aggregate the individual data points to spatial areas (zones) in order to compare the events to data that are only obtained for zones, such as census data, or to model environmental correlates of the data points or may find that individual data are not available (e.g., when a police

department releases information by police beats but not individual streets). In this case, the individual data points are allocated to zones by, first, spatially assigning them to the zones in which they fall and, second, counting the number of points assigned to each zone. A user can do this with a GIS program or with the "Assign Primary points to Secondary Points" routine that will be discussed in Chapter 6.

In this case, the zone becomes the unit of analysis instead of the individual data points. All the incidents are assigned to a single geographical coordinate, typically the *centroid* of the zone, and the number of incidents in the zone (the count) becomes an *attribute* of the zone (e.g., number of robberies per zone; number of motor vehicle crashes per zone).

It should be obvious that when individual data points are assigned to zones, information is lost. Instead of capturing the unique locations of the individual events, all events that occur within a zone are assigned a single location. Thus, the distance between zones is a singular value for all the points in those zones whereas there is much greater variability with the distances between individual events.

Further, zones have attributes which are properties of the zone, not of the individual events. The attribute can be a *count* or a continuous variable for a distributional property of the zone (e.g., median household income; percentage of households below poverty level).[2]

Analysis then proceeds on the basis of the zonal information. The results will be different than for an analysis of the individual event information since the spatial characteristics are measured by single points for each zone (e.g., the centroid) and the attribute information is measured by a property of the zone, not the individual events (e.g., the count of events in the zone; a characteristic of the zone such as income level).

In other words, the user must realize that an analysis of zonal data is quite different from an analysis of individual data and that the conclusions might be different. Aggregating data to zones creates properties that may be different than those of individual events and that the relationships between variables at the zonal level also might be different than at the individual level. This is called an *ecological* relationship and there is a large literature on ecological inference and fallacies (see Freedman, 1999; Langbein & Lichtman, 1979).

Individual level data can also have attributes. For example, Levine and Lee (2013) analyzed journey-to-crime distances for offenders in Manchester, England. In this case, the attribute variable was the distance traveled and the statistics discussed in this chapter are

[2] There is no fundamental difference between a count variable and a continuous interval or ratio variable since a real number can be converted into a count by multiplying by a power of 10 (e.g., $1.23 = 123 \times 10^{-2}$). The statistics discussed in this chapter are applicable to either count or continuous data.

appropriate for analyzing that attribute data. Other examples of individual level data with attributes would be the age of the offender, the number of prior convictions, or the number of years of formal education. The key criterion is that the records must have an attribute which is either a count or an interval variable.

Spatial Autocorrelation Statistics for Attribute Data

There are a number of formal statistics that attempt to measure spatial autocorrelation at the zonal level or for individual level data with count or interval attributes. These statistics include simple indices, such as the Moran's I", Geary's C or the Getis-Ord "G" statistic, the application of these statistics to individual zones or records (discussed in Chapter 9), and multivariate indices such as the Markov Chain Monte Carlo spatial regression models (discussed in Chapter 19). The simple indices attempt to identify whether spatial autocorrelation exists for a single variable while the more complicated indices attempt to estimate variability in spatial autocorrelation in a study area of the effect of spatial autocorrelation on a particular attribute variable.

CrimeStat includes three global indices - Moran's I statistic, Geary's C statistic, and the Getis-Ord "G" statistic. It also includes *Correlograms* that apply each of these indices to different distance intervals. Moran, Geary, and Getis-Ord are *global* in that they represent a summary value for all the data points. In Chapter 9, we will present some local indicators of spatial autocorrelation that apply the Moran, Geary and Getis-Ord statistics to individual zones. But, for now, we are focused on describing the entire study area.

Moran's "I" Statistic

Moran's "I" statistic (Moran, 1950) is one of the oldest indicators of spatial autocorrelation. It is applied to zones or points that have attribute variables associated with them (intensities). For any continuous variable, X_i, a mean, $\bar{X}$, can be calculated and the deviation of any one observation from that mean, s_X, can also be calculated. The statistic then compares the value of the variable at any one location with the value at all other locations (Ebdon, 1988; Griffith, 1987; Anselin, 1992). Formally, it is defined as:

$$I = \frac{N \sum_i \sum_j w_{ij} \left(X_i - \bar{X}\right)(X_j - \bar{X})}{(\sum_i \sum_j w_{ij}) \sum_i (X_i - \bar{X})^2} \qquad (5.1)$$

where N is the number of cases, X_i is the value of a variable at a particular location, i, X_j is the value of the same variable at another location (where $i \neq j$), $\bar{X}$ is the mean of the variable and W_{ij} is a weight applied to the comparison between location i and location j.

In Moran's initial formulation, the weight variable, W_{ij}, was a contiguity matrix. If zone j is adjacent to zone i, the interaction receives a weight of 1. Otherwise, the interaction receives a weight of 0. Cliff and Ord (1973) generalized these definitions to include any type of weight. In more current use, W_{ij}, is a distance-based weight which is the inverse distance between locations i and j ($1/d_{ij}$). *CrimeStat* uses this interpretation. Essentially, it is a *weighted* Moran's I where the weight is an inverse distance.

Note that in adopting a distance-based weight, there are advantages and disadvantages. Contiguity (or adjacency) is a property of a zone, not a point. Thus, adjacency defines whether one zone is next to another zone whereas distance is the distance between single points that represent the zones (e.g., centroids). If two zones are, say, 0.25 miles apart, it is not known whether they are adjacent or not. In other words, in adopting a distance-based weight, information about adjacencies is lost. On the other hand, a distance-based weight is standardized. If two zones are adjacent, it is not known how far apart they are separated. Adjacencies can be misleading since they don't indicate the size of the adjacent zones whereas a specified distance is always constant.

The weighted Moran's I is similar to a correlation coefficient in that it compares the sum of the cross-products of values at different locations, two at a time, weighted by the inverse of the distance between the locations and with the variance of the variable. Like a correlation coefficient, it typically varies between -1.0 and + 1.0. However, this is not absolute as an example later in the chapter will show. When nearby points have similar values, their cross-product is high. Conversely, when nearby points have dissimilar values, their cross-product is low. Consequently, an "I" value that is high indicates more spatial autocorrelation than an "I" that is low.

However, unlike a correlation coefficient, the theoretical value of the index does not equal 0 for lack of spatial dependence, but instead is negative but very close to 0:

$$E(I) = -\frac{1}{N-1} \tag{5.2}$$

Values of "I" above the theoretical mean, E(I), indicate positive spatial autocorrelation while values of "I" below the theoretical mean indicate negative spatial autocorrelation.

Adjust for Small Distances

CrimeStat calculates the weighted Moran's I formula using equation 5.1. However, there is one problem with this formula that can lead to unreliable results. The distance weight between

two locations, W_{ij}, is defined as the reciprocal of the distance between the two points, consistent with Moran's original formulation:

$$W_{ij} = \frac{1}{d_{ij}} \tag{5.3}$$

Unfortunately, as d_{ij} becomes small, then W_{ij} becomes very large, approaching infinity as the distance between the points approaches 0. If the two zones were next to each other, which would be true for two adjacent blocks for example, then the pair of observations would have a very high weight, sufficient to distort the "I" value for the entire sample. Further, there is a scale problem that alters the value of the weight. If the zones are police precincts, for example, then the minimum distance between precincts will be a lot larger than the minimum distance between a smaller geographical unit, such as a block. We need to take into account these scales.

CrimeStat includes an adjustment for small distances so that the maximum weight can never be greater than 1.0. The adjustment scales distances to one mile, which is a typical distance unit in the measurement of crime incidents. When the small distance adjustment is turned on, the minimal distance is automatically scaled to be one mile. The formula used is:

$$W_{ij} = \frac{one\ mile}{one\ mile + d_{ij}} \tag{5.4}$$

in the units are specified. For example, if the distance units, d_{ij}, are calculated as feet, then:

$$W_{ij} = \frac{5{,}280}{5{,}280 + d_{ij}}$$

where 5,280 is the number of feet in a mile. This has the effect of insuring that the weight of a particular pair of point locations will not have an undue influence on the overall statistic. The traditional measure of "I" is the default condition in *CrimeStat*, but the user can turn on the small distance adjustment by clicking on the appropriate box.

Testing the Significance of Moran's "I"

The empirical distribution can be compared with the theoretical distribution by dividing by an estimate of the theoretical standard deviation:

$$Z(I) = \frac{I - E(I)}{S_{E(I)}} \tag{5.5}$$

where "I" is the empirical value calculated from a sample, E(I) is the theoretical mean of a random distribution and $S_{E(I)}$ is the theoretical standard deviation of E(I).

There are several interpretations of the theoretical standard deviation that affect the particular statistic used for the denominator as well as the interpretation of the significance of the statistic (Anselin, 1992). The most common assumption is that the standardized variable, Z(I), has a sampling distribution which follows a standard normal distribution, that is with a mean of 0 and a variance of 1. This is called the *normality* assumption.[3] A second interpretation assumes that each observed value could have occurred at any location, that is the location of the values and their spatial arrangement is assumed to be unrelated. This is called the *randomization* assumption and has a slightly different formula for the theoretical standard deviation of 5.13.[4] *CrimeStat* outputs the Z-values and p-values for both the normality and randomization assumptions.

Example: Testing Houston Burglaries with Moran's "I"

To illustrate the use of Moran's I with point locations, the data must have intensity values associated with each point. Since most crime incidents are represented as a single point, they do not naturally have associated intensities. It is necessary, therefore, to adapt crime data to fit the form required by Moran's I. One way to do this is assign crime incidents to geographical zones and count the number of incidents per zone.

Figure 5.2 shows 2006 burglaries in the City of Houston by individual Traffic Analysis Zones (TAZ). TAZ's are groupings of census blocks but designed to equalize the number of trips to and from the zone in the base year. They are typically very small in downtown Houston (typically a block in size) and much larger in the suburban parts of the City. With a GIS program, 26,480 burglary locations were overlaid on top of a map of 1,179 TAZ's and the number of burglaries within each TAZ were counted and then assigned to the TAZ as a variable (see the 'Assign primary points to secondary points' routine in Chapter 6).[5] The numbers varied from 0 burglaries (for 250 TAZ's) up to 284 burglaries incidents (for 1 TAZ). The map shows the plot of the number of burglaries per TAZ.

---

[3] The theoretical standard deviation of "I" under the assumption of normality is (Ebdon, 1985):

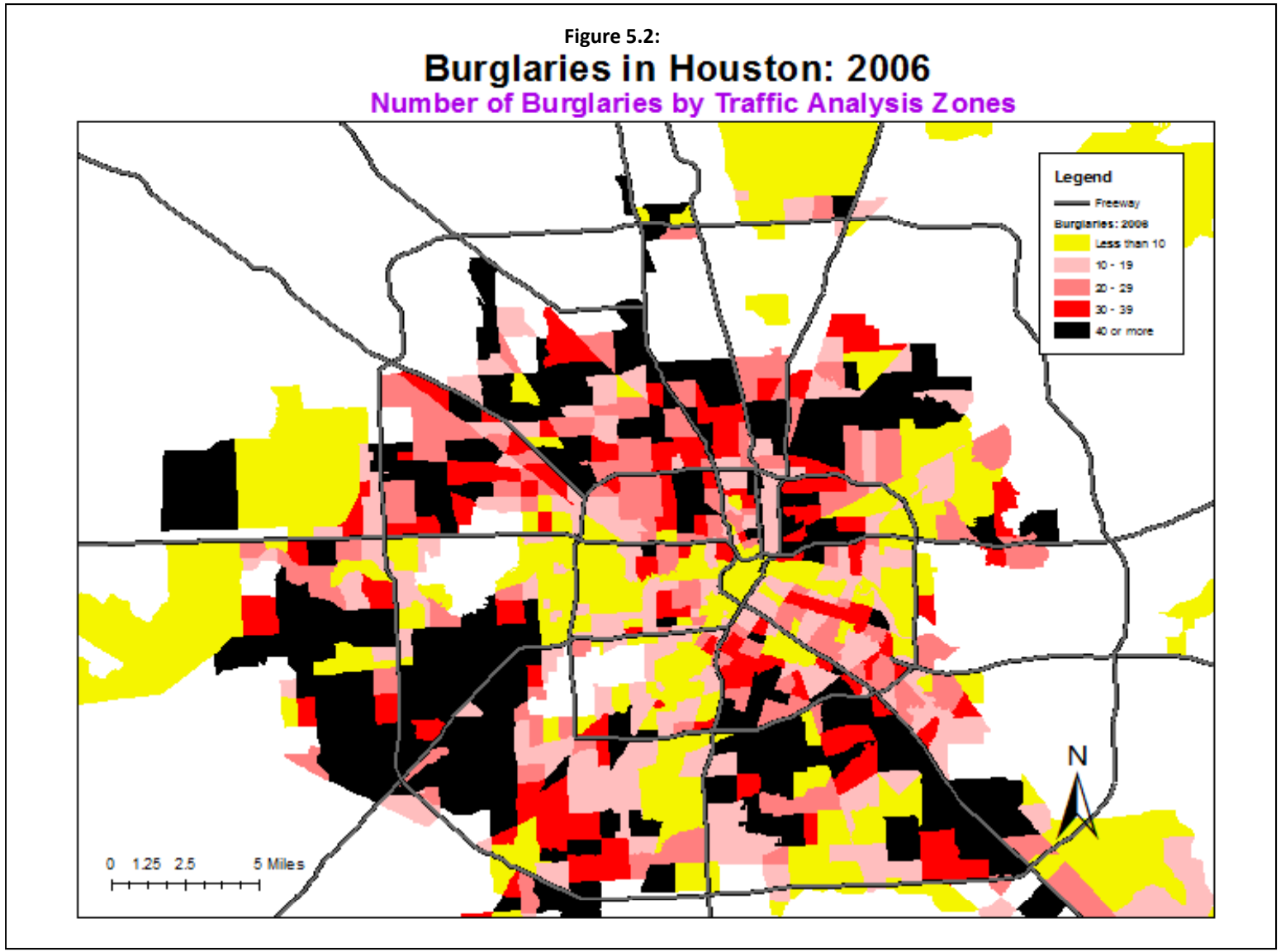
$$S_{E(I)} = \sqrt{\frac{N^2(\sum_i\sum_j W_{ij}^2)+3(\sum_i\sum_j W_{ij})^2-N\sum_i(\sum_j W_{ij})^2}{(N^2-1)(\sum_i\sum_j W_{ij})^2}}$$

[4] The formula for the theoretical standard deviation of "I" under the randomization assumption is (Ebdon, 1985):

$$S_{E(I)} = \sqrt{\frac{N[(N^2+3-3N)(\sum_I\sum_J W_{IJ}^2)+3(\sum_i\sum_j W_{ij})^2-N\sum_i(\sum_j W_{ij})^2]-k[(N^2-N)\sum_i\sum_j W_{ij}^2+6(\sum_i\sum_j W_{ij})^2-2N(\sum_i(\sum_j W_{ij})^2]}{(N-1)(N-2)(N-3)(\sum_i\sum_j W_{ij})^2}}$$

[5] The TAZ data were obtained from the Houston-Galveston Area Council, the Metropolitan Planning Organization for the Houston metro area.

Figure 5.2:


Burglaries in Houston: 2006
Number of Burglaries by Traffic Analysis Zones
Legend
Freeway
Burglaries: 2008
Less than 10
10 - 19
20 - 29
30 - 39
40 or more
N
0 1.25 2.5 5 Miles

Clearly, aggregating incident locations to zones, such as TAZ's, eliminates some information since all incidents within a block are assigned to a single location (the centroid of the block). The use of Moran's I, however, requires the data to be in this format. Using data in this form, Moran's I was calculated using the small distance adjustment because many TAZ's are very close together, especially in downtown Houston.

Figure 5.3 shows the output of the "I" in *CrimeStat.* "I" was 0.251790, the theoretical value of "I" as -0.000849, and the standard error of "I" as 0.002796. The test of significance using the normality assumption gave a Z-value of 213.20, a highly significant value. Below are the calculations for burglaries by TAZ:

$$Z\ (I_{veh}) = \frac{I_{veh} - E(I)}{S_{E(I)}} = \frac{0.251795 - (-0.000849)}{0.002796} = 213.20\ (p \leq .0001)$$

Comparing Moran's "I" for Two Distributions

Figure 5.4 shows the distribution of households in the city by TAZ. The calculations for the "*I*" of households are similar (not shown). It turns out that the "I" of households is 0.298117 while the theoretical "I" and the standard error of "I" are the same as for burglaries (because of the same zonal geography). One can compare an "I" value for one distribution with the "I" value for another distribution. For example, a Z-test can then be made of whether the "*I*" value of burglaries is statistically different than that of households. The calculations are shown below:

$$Z\left(I_{difference}\right) = \frac{I_{burg} - I_{hh}}{S_{E(I)}} = \frac{0.251795 - (0.298117)}{0.002796} = -16.57\ (p \leq .001)$$

where I_{burg} is the "I" value for burglaries, I_{hh} is the "*I*" value for households, and $S_{E(I)}$ is the standard deviation of "I" for households under the assumption of normality. The Z-test of the difference is -16.57, a highly significant difference. The high Z-value suggests that burglaries are even more clustered than the clustering of households. To put it another way, they are more clustered than would be expected based on the household distribution. As mentioned, this is an approximate test since the joint distribution of "I" for two empirical distributions of "I" is not known.

Geary's C Statistic

Geary's C statistic is similar to Moran's I (Geary, 1954). In this case, however, the interaction is not the cross-product of the deviations from the mean, but the deviation in intensities of each observation's location with one another. It is defined as:

Figure 5.3:

Moran's I Statistic Output

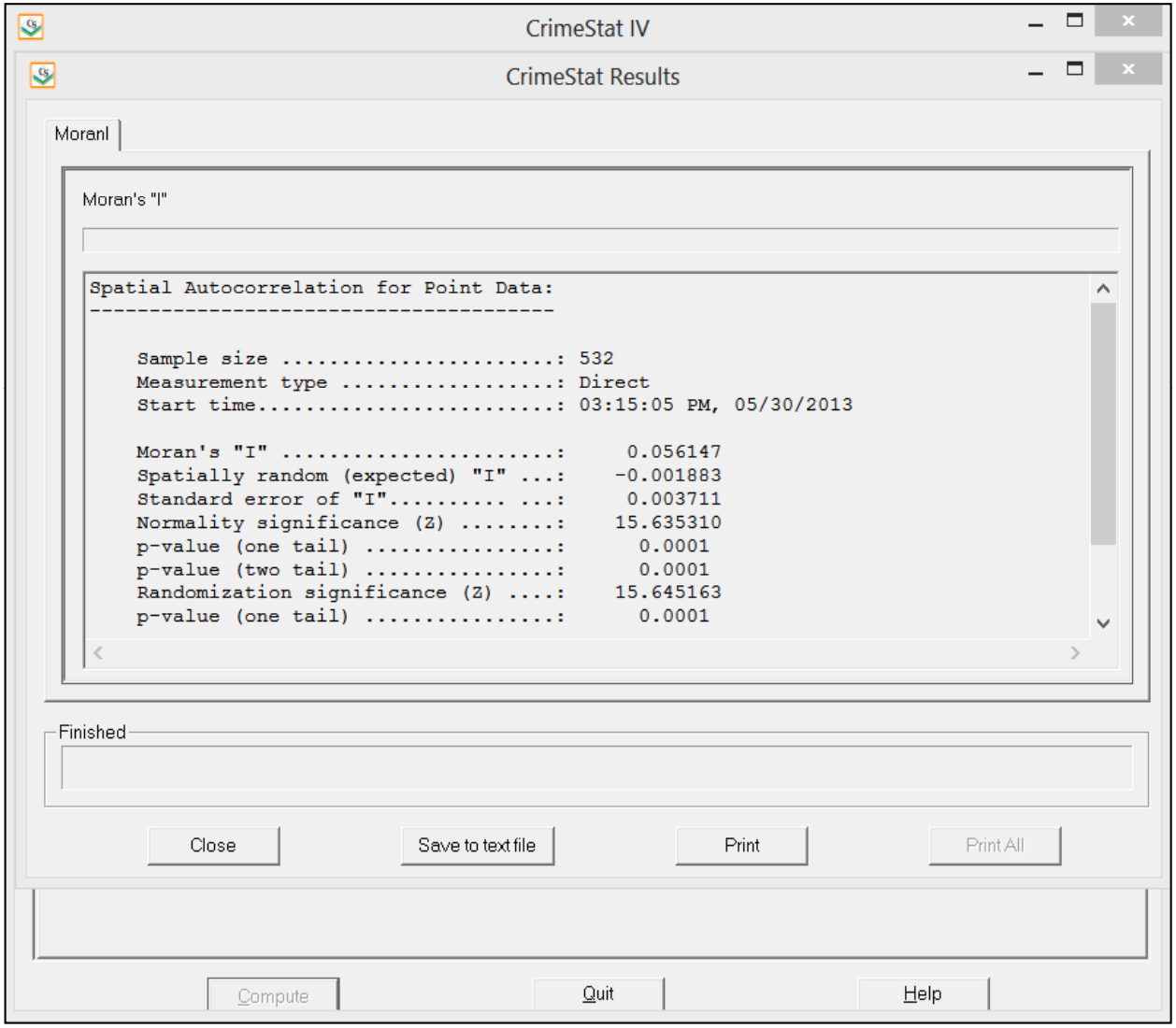
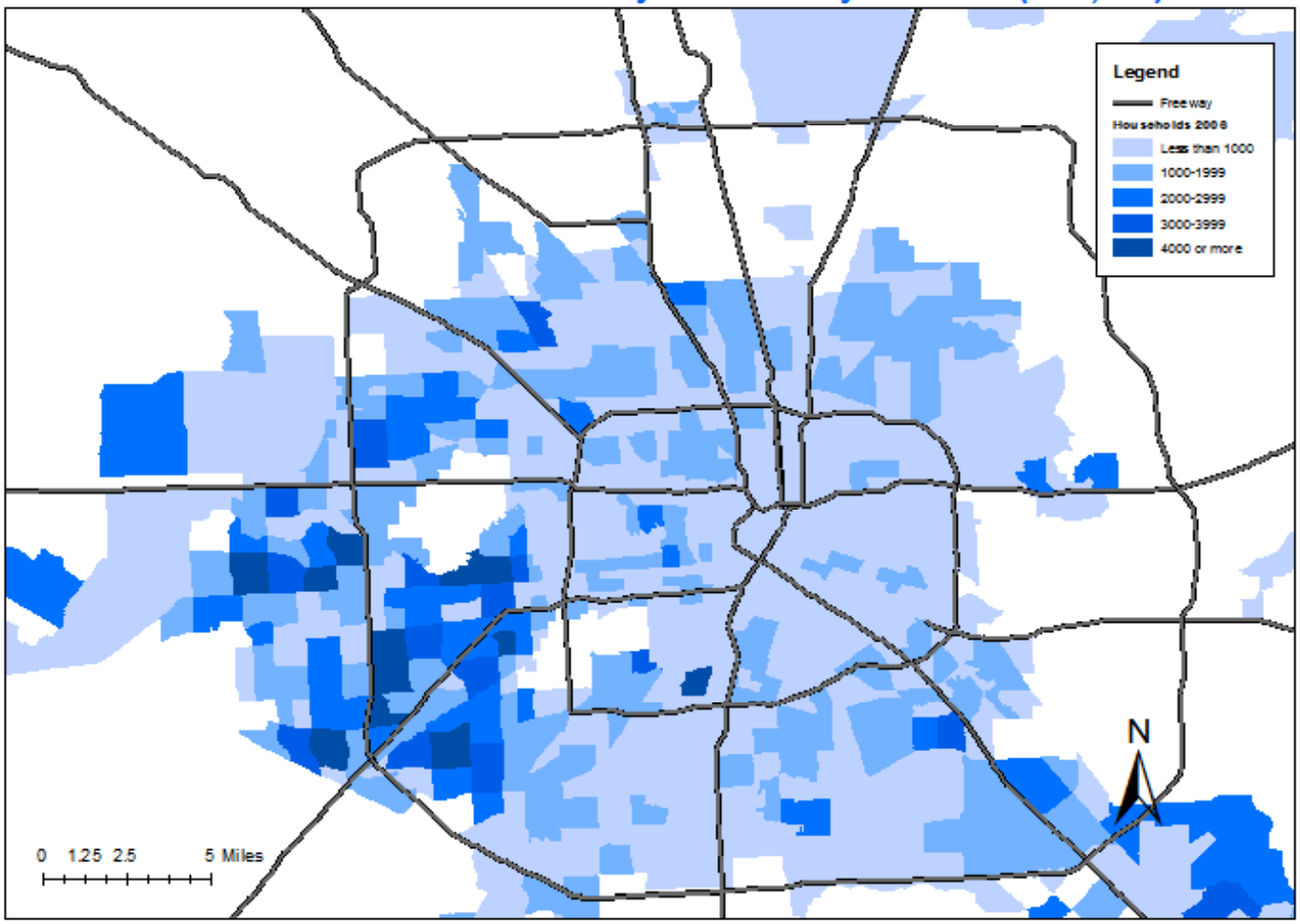


Figure 5.4:

Households in Houston: 2006

Number of Households by Traffic Analysis Zones (N=1,179)



$$C = \frac{(N-1)\sum_i W_{ij}(X_i - X_j)^2}{2(\sum_i \sum_j W_{ij}) \sum_i (X_i - \bar{X})^2} \quad (5.6)$$

The values of "C" typically vary between 0 and 2, although 2 is not a strict upper limit (Griffith, 1987). The theoretical value of "C" is 1; that is, if values of any one zone are spatially unrelated to any other zone, then the expected value of "C" would be 1. Values less than 1 (i.e., between 0 and 1) typically indicate positive spatial autocorrelation while values greater than 1 indicate negative spatial autocorrelation. Thus, this index is inversely related to Moran's "I". It will not provide identical inference because it emphasizes the differences in values between pairs of observations comparisons rather than the co- variation between the pairs (i.e., product of the deviations from the mean). The Moran coefficient gives a more global indicator whereas the Geary coefficient is more sensitive to differences in small neighborhoods.

Adjusted "C"

A more intuitive interpretation of *"C"* can be obtained by calculating an adjusted*"C"*:

$$Adjusted\ C = 1 - C \quad (5.7)$$

In this case, the adjusted *"C"* will be on the same scale as Moran's "I". An adjusted *"C"* value that is positive indicates positive spatial autocorrelation while an adjusted *"C"* value that is negative indicates negative spatial autocorrelation. An adjusted *"C"* of 0 indicates no spatial autocorrelation and is also the expected adjusted "C". *CrimeStat* calculates both the regular and adjusted "C" values.

Adjust for Small Distances

Like Moran's "I", the weights are defined as the inverse of the distance between the paired points:

$$W_{ij} = \frac{1}{d_{ij}} \quad (5.3) \text{ repeat}$$

However, the weights will tend to increase substantially as the distance between points decreases. Consequently, a small distance adjustment is allowed that ensures no weight is greater than 1.0:

$$W_{ij} = \frac{one\ mile}{one\ mile + d_{ij}} \quad (5.4) \text{ repeat}$$

The adjustment scales the distances to one mile in the distance units specified on the Primary file page (miles, feet, kilometers, meters, or nautical miles). This is the default condition although the user can calculate all weights as the reciprocal distance by turning off the small distance adjustment.

Testing the Significance of Geary's "C"

The empirical "C" distribution can be compared with the theoretical distribution by dividing by an estimate of the theoretical standard deviation

$$Z(C) = \frac{C - E(C)}{S_{E(C)}} \tag{5.8}$$

where *C* is the empirical "C", *E(C)* is the theoretical mean of a random distribution and $S_{E(C)}$ is the theoretical standard deviation of E(C). The usual test is to assume that the sample Z follows a standard normal distribution with mean of 0 and variance of 1 (normality assumption), though it is possible to calculate the standard error under a randomization assumption (Ripley, 1981).[6]

Note that for testing, the regular "C" value should be used since an adjusted standard error of "C" is not easily calculated. The adjusted "C" is useful for a quick intuitive appraisal as well as for the Geary Correlogram (see below).

Example: Testing Houston Burglaries with Geary's "C"

Using the same data on burglaries in the City of Houston, figure 5.5 illustrates the output. The regular "C" value for burglaries was 0.625702 with a Z-value of -20.00 ($p \leq .0001$). The "C" value of burglaries is *smaller* than the theoretical "C" of 1. Converting this measure an adjusted "C" gives 0.374298 and indicates positive spatial autocorrelation. That is, the index suggests that TAZ's with a high number of burglaries are adjacent to TAZ's also with a high number of burglaries. Thus, Geary's C confirms the evidence for positive spatial autocorrelation identified by Moran's "I".

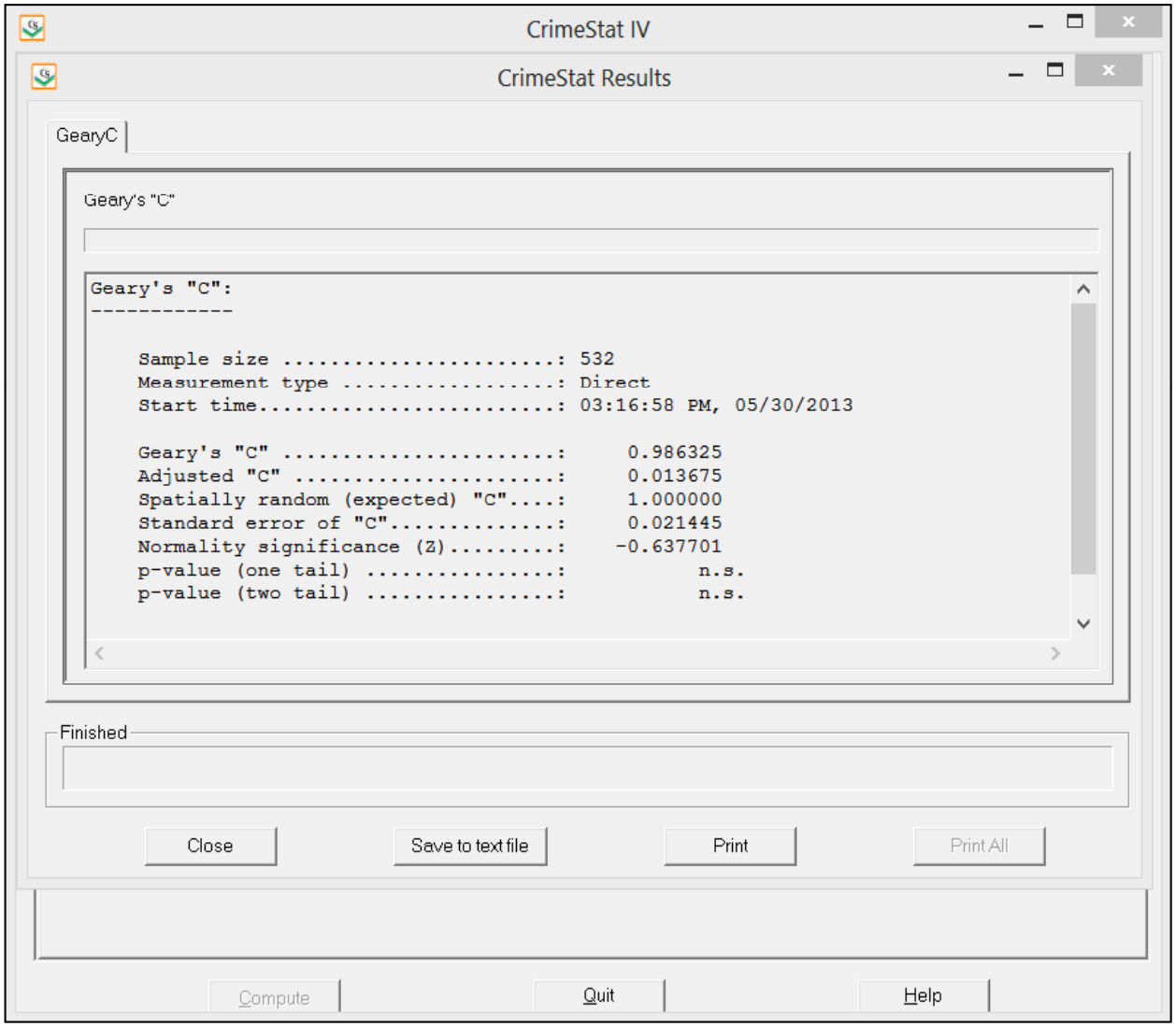
---

[6] The theoretical standard deviation for C under the normality assumption is (Ripley, 1981):

$$S_{E(I)} = \sqrt{\frac{2\sum_i\sum_j(W_{ij}^2) + \sum_i(\sum_j W_{ij})^2(N-1) - 4(\sum_i\sum_j W_{ij})^2}{2(N+1)(\sum_i\sum_j W_{ij})^2}}$$

Figure 5.5:

Geary's C Statistic Output



Comparing this to the distribution of households in Houston, the C value of households is also below the theoretical "C" of 1 and points to positive spatial autocorrelation ("C" = 0.643120 with a Z-value of -19.07; p≤.0001). Since both the regular "C" for burglaries and for households are below 1 (hence, indicating positive spatial autocorrelation), let us test the difference between the two as indicated by a Z-test of the difference.

$$\left(C_{difference}\right) = \frac{C_{burg} - C_{hh}}{S_{E(C)}} = \frac{0.625702 - (0.643120)}{0.018717} = -0.930598\ (n.s.)$$

In this case, there is no statistical difference between the distribution of burglaries and the distribution of households. Though both distributions show evidence of positive spatial autocorrelation, the Geary test cannot show a difference between the two whereas the Moran's "I" did show a difference.

Typically, Geary's "C" will be consistent with Moran's "I" though there are slight differences between the indices, as we see in this example. Because of the nature of the weighting, the Geary index is more sensitive to local clustering (second-order effects) than the Moran index, which is better seen as measuring first-order spatial autocorrelation. This illustrates how these indices have to be used with care and cannot be generalized by themselves. Each of them emphasizes slightly different information regarding spatial autocorrelation, yet neither is sufficient by itself. They should be used as part of a larger analysis of spatial patterning.[7]

Getis-Ord "G" Statistic

The Getis-Ord "G" statistic is also an index of global spatial autocorrelation but for values that fall within a specified distance of each other (Ord & Getis, 1995; Getis & Ord, 1992). When compared to an expected value of "G" under the assumption of no spatial association, it has the advantage over other two global spatial autocorrelation measures in that it can distinguish between 'hot spots' and 'cold spots', which neither Moran's "I" nor Geary's "C" can do.

The "G" statistic calculates the spatial interaction of the value of a particular variable in a zone with the values of that same variable in nearby zones, similar to Moran's "I" and Geary's

[7] Anselin (1992) points out that the results of the two indices are determined to a large extent by the type of weighting used. In the original formulation, where adjacent weights of 1 and 0 were used, the two indices were linearly related, though moving in opposite directions (Griffith, 1987). Thus, only adjacent zones had any impact on the index. With inverse distance weights, however, zones farther removed can influence the overall index so it is possible to have a situation whereby adjacent zones have similar values (hence, are positively autocorrelated) whereas zones farther away could have dissimilar values (hence, are negatively autocorrelated).

"C". Thus, it is also a measure of spatial association or interaction. Unlike the other two measures, it ***only*** identifies ***positive*** spatial autocorrelation, that is, where zones have similar values to their neighbors. It cannot detect negative spatial autocorrelation where zones have different values to their neighbors. But, unlike the other two global measures, it can distinguish between positive spatial autocorrelation where zones with high values are near to other zones with high values (***high positive spatial autocorrelation***) from positive spatial autocorrelation which where zones with low values are near to other zones also with low values (***low positive spatial autocorrelation***). Further, the "G" value is calculated with respect to a specified search distance (defined by the user) rather than to an inverse distance, as with the Moran's "I" or Geary's "C".

The formulation of the general "G" statistic presented here is taken from Lee and Wong (2005). It is defined as:

$$G(d) = \frac{\sum_i \sum_j W_j(d) X_i X_j}{\sum_i \sum_j X_i X_j} \tag{5.9}$$

for a variable, X. This formula indicates that the cross-product of the value of X at location "i" and at another zone "j" is weighted by a distance weight, $w_j(d)$ which is defined by either a '1' if the two zones are equal to or closer than a threshold distance, d, or "0" otherwise. The cross-product is summed for all other zones, j, over all zones, i. Thus, the numerator is a sub-set of the denominator and can vary between 0 and 1. If the distance selected is too small so that no other zones are closer than this distance, then the weight will be 0 for all cross-products of variable X. Hence, the value of G(d) will be 0. Similarly, if the distance selected is too large so that all other zones are closer than this distance, then the weight will be 1 for all cross-products of variable X. Hence, the value of G(d) will be 1.

There are actually two "G" statistics. The first one, G*, includes the interaction of a zone with itself; that is, zone "i" and zone "j" can be the same zone. The second one, G, does not include the interaction of a zone with itself. In *CrimeStat*, we only include the "G" statistic (i.e., there is no interaction of a zone with itself) because, first, the two measures produce almost identical results and, second, the interpretation of "G" is more straightforward than with G*. Essentially, with G, the statistic measures the interaction of a zone with nearby zones (a 'neighborhood'). See articles by Getis and Ord (1996) and by Khan, Qin and Noyce (2006) for a discussion of the use of G*.

Testing the Significance of "G"

By itself, the "G" statistic is not very meaningful. Since it can vary between 0 and 1, as the threshold distance increases, the statistic will always approach 1.0. Consequently, "G" is compared to an expected value of "G" under no significant spatial association. The expected "G" for a threshold distance, d, is defined as:

$$E[G(d)] = \frac{W}{N(N-1)} \tag{5.10}$$

where W is the sum of weights for all pairs and N is the number of cases. The sum of the weights is based on *symmetrical* counts of those zones within the threshold distance. That is, if zone 2 is within the threshold distance of zone 1, then zone 2 contributes a weight of 1 to zone 1. However, zone 1 contributes a weight of 1 to zone 2 as well. In other words, if two zones are within the threshold (search) distance, then they both contribute 2 to the total weight.

Note that, since the expected value of "G" is a function of the sample size and the sum of weights which, in turn, is a function of the search distance, it will be the same for all variables of a single data set in which the same search distance is specified. However, as the search distance changes, so will the expected "G" change.

Theoretically, the "G" statistic is assumed to have a normally distributed standard error. If this is the case (and we often do not know if it is), then the standard error of "G" can be calculated and a simple significance test based on the normal distributed be constructed. The variance of G(d) is defined as:

$$Var[G(d)] = E(G^2) - E(G)^2 \tag{5.11}$$

where

$$E(G)^2 = \frac{1}{(m_1^2 - m_2)^2 n^4}[B_0 m_2^2 + B_1 m_4 + B_2 m_1^2 m_2 + B_3 m_1 m_3 + B_4 m_1^4 \tag{5.12}$$

and where:

$$m_1 = \sum_i X_i \tag{5.13}$$
$$m_2 = \sum_i X_i^2 \tag{5.14}$$
$$m_3 = \sum_i X_i^3 \tag{5.15}$$
$$m_4 = \sum_i X_i^4 \tag{5.16}$$
$$N^4 = N(N-1)(N-2)(N-3) \tag{5.17}$$

$$S_1 = 0.5 \sum_i \sum_j (W_{ij} + W_{ji})^2 \quad (5.18)$$

$$S_2 = \sum_i (\sum_J W_{IJ} + \sum_J W_{JI})^2 \quad (5.19)$$

$$B_0 = (N^2 - 3N + 3)S_1 - NS_2 + 3W^2 \quad (5.20)$$

$$B_1 = -[(N^2 - N)S_1 - 2NS_2 + 3W^2] \quad (5.21)$$

$$B_2 = -[2NS_1 - (N + 3)S_2 + 6W^2] \quad (5.22)$$

$$B_3 = 4(N - 1)S_1 - 2(N + 1)S_2 + 8W^2 \quad (5.23)$$

$$B_4 = S_1 - S_2 + W^2 \quad (5.24)$$

where ***i*** is the zone being calculated, ***j*** is all other zones, and *N* is the sample size (Lee and Wong, 2005). Note that this formula is different than that written in other sources (e.g., see Lees, 2006) but is consistent with the formulation by Getis and Ord (1992).

The standard error of G(d) is the square root of the variance of G. Consequently, a Z-test can be constructed by:

$$S.E.[G(d)] = \sqrt{Var[G(d)]} \quad (5.25)$$

$$Z[G(d)] = \frac{G(d) - E[G(d)]}{S.E.[G(d)]} \quad (5.26)$$

Relative to the expected value of G, a positive Z-value indicates spatial clustering of high values (high positive spatial autocorrelation or 'hot spots') while a negative Z-value indicates spatial clustering of low values (low positive spatial autocorrelation or 'cold spots'). A "G" value around 0 typically indicates either no positive spatial autocorrelation, negative spatial autocorrelation (which the Getis-Ord cannot detect), or that the number of 'hot spots' more or less balances the number of 'cold spots'.

Note that the value of this test will vary with the search distance selected. One search distance may yield a significant spatial association for "G" whereas another may not. In other words, the statistic is useful for identifying distances at which spatial autocorrelation exists.

> In practice, one should use a small search distance to identify local spatial autocorrelation.

Also, and this is an important point, the expected value of "G" as calculated in equation 5.10 is only meaningful if the variable is positive. For variables with negative values, such as residual errors from a regression model, one cannot use equation 5.10 but, instead, must use a simulation to estimate confidence intervals.

Simulating Confidence Intervals for "G"

One of the problems with this test is that "G" may not actually follow a normal standard error. That is, if "G" was calculated for a specific distance, d, with random data, the distribution of the statistic may not be normally distributed. This would be especially true if the variable of interest is a skewed variable with some zones having very high values while the majority of zones having low values.

Consequently, the user has an alternative for estimating the confidence intervals using a Monte Carlo simulation. In this case, a *permutation* type simulation is run whereby the original values of the intensity variable, Z, are maintained but are randomly re-assigned for each simulation run (Anselin, 2008). This will maintain the distribution of the variable Z but will estimate the value of "G" under random assignment of this variable. The user can take the usual 95% or 99% confidence intervals based on the simulation.

> Keep in mind that a simulation may take time to run especially if the data set is large or if a large number of simulation runs are requested.

Example: Testing Simulated Data with the Getis-Ord "G"

To understand how the Getis-Ord "G" works and how it compares to the other two global spatial autocorrelation measures - Moran's "I" and the adjusted Geary's "C", three simulated data sets were created. In the first, a random pattern was created (Figure 5.6). In the second, a data set of extreme positive spatial autocorrelation was created (Figure 5.7) and, in the third, a data set of extreme negative spatial autocorrelation was created (Figure 5.8); the latter is essentially a checkerboard pattern.

Table 5.1 compares the three global spatial autocorrelation statistics on the three distributions. For the Getis-Ord "G", both the actual "G" and the expected "G" are shown. A one mile search distance was used for the Getis-Ord "G".

Figure 5.6:

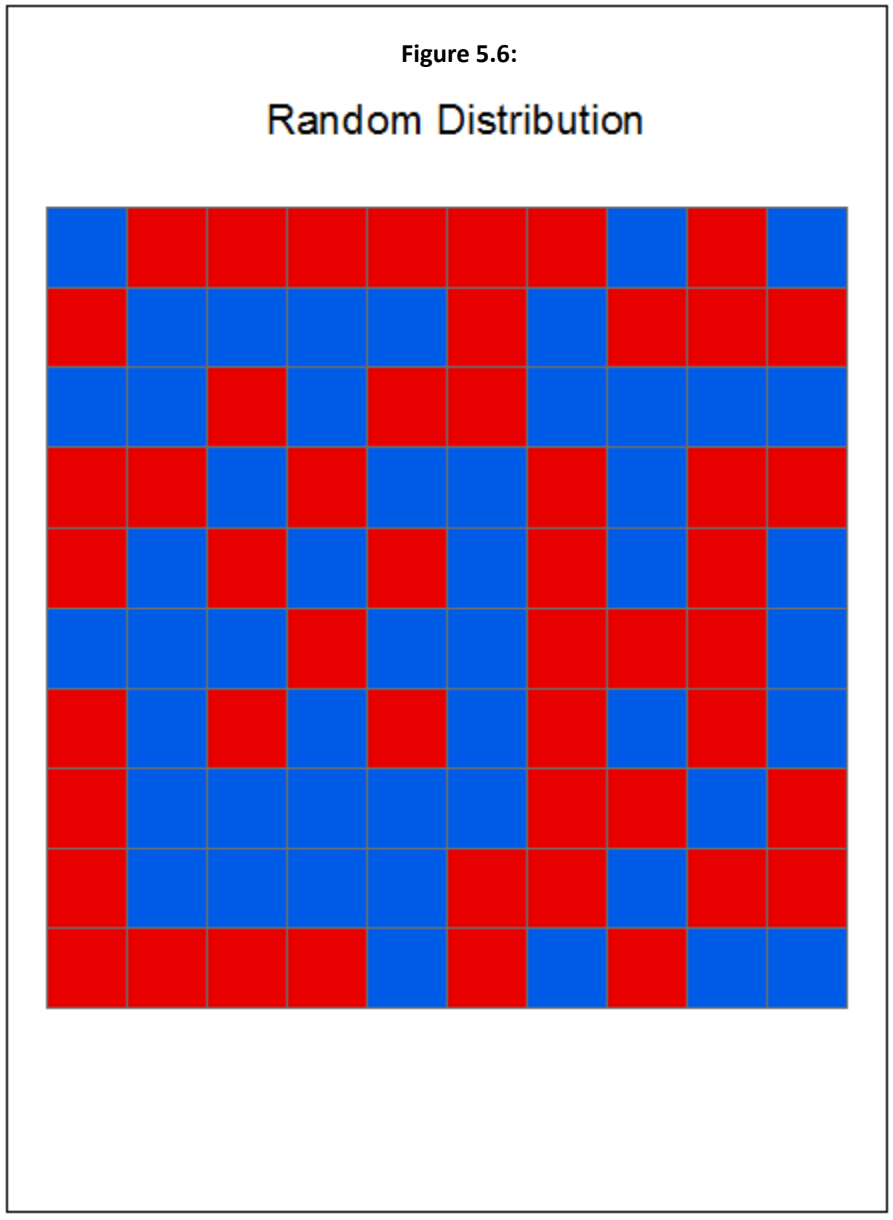


Figure 5.7:

Extreme Positive Spatial Autocorrelation

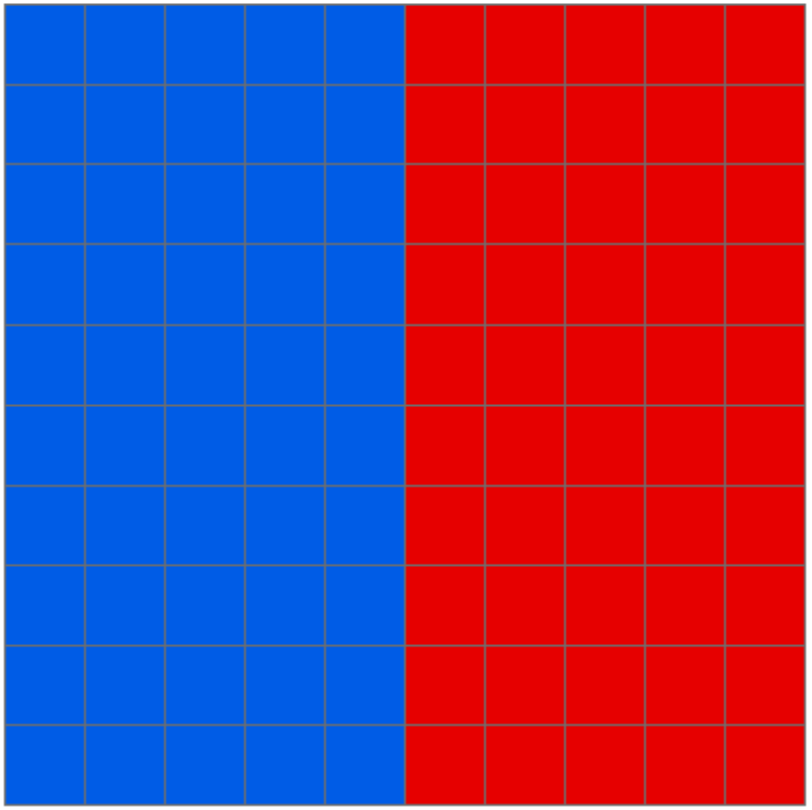


Figure 5.8:

Extreme Negative Spatial Autocorrelation

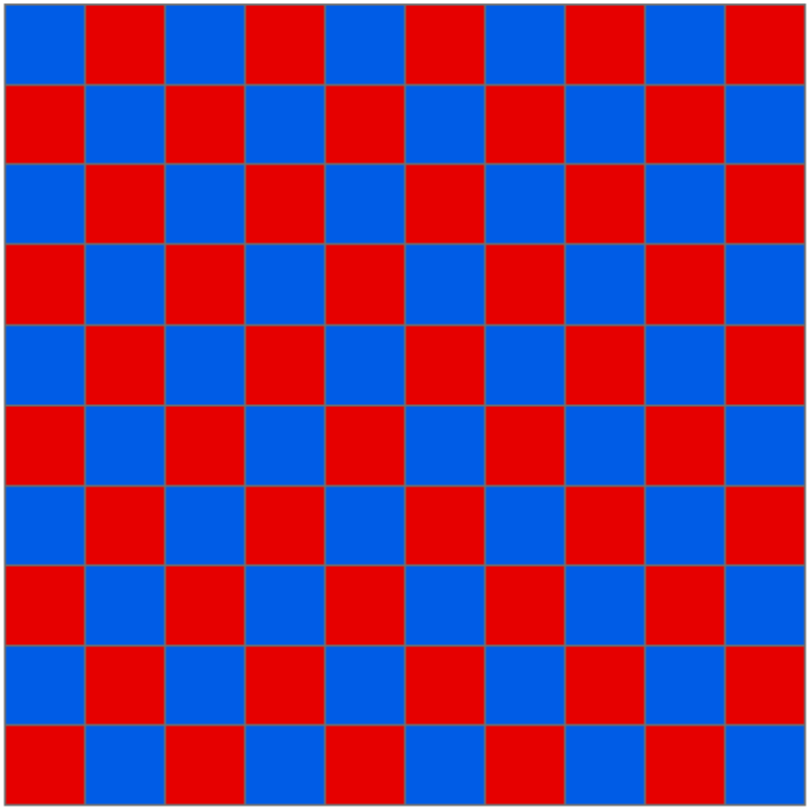


Table 5.1:
Global Spatial Autocorrelation Statistics for Simulated Data Sets
N = 100 Grid Cells

			----------*Getis-Ord "G"*----------	
Pattern	**Moran's "I"**	**Adjusted Geary's "C"**	**Observed "G"** (1 mi search)	**Expected "G"** (1 mi search)
Random	-0.007162 $^{n.s.}$	0.965278$^{n.s}$	0.151059$^{n.s}$	0.159596
Positive spatial autocorrelation	0.292008****	0.700912****	0.241015****	0.159596
Negative spatial autocorrelation	-0.060071***	-0.049471*	0.140803$^{n.s.}$	0.159596

n.s	not significant
*	p≤.05
**	p≤.01
***	p≤.001
****	p≤.0001

The random pattern is not significant for all three measures. That is, neither the Moran "I", the adjusted Geary"C", nor the Getis-Ord "G" is significantly different than the expected value under a random distribution. This is what would be expected since the data were assigned randomly.

For the extreme positive spatial autocorrelation pattern, on the other hand, all three measures show highly significant differences with a random distribution. Moran's "I" is highly positive. The adjusted Geary's "C is above 1.0, indicating positive spatial autocorrelation and the Getis-Ord "G" has a "G" value that is significantly higher than the expected "G" based on the theoretical standard error. The Getis-Ord "G", therefore, indicates that the type of spatial autocorrelation is high positive.

Finally, the extreme negative spatial autocorrelation pattern (Figure 5.8 above) shows different results for the three measures. Moran's "I" shows negative spatial autocorrelation and is highly significant (p≤.001). Geary's "C also shows negative spatial autocorrelation but it is significant only at the p≤.05 level. Finally, the Getis-Ord "G" is slightly smaller than the expected "G", which indicates low positive spatial autocorrelation, but it is not significant.

In other words, all three statistics can identify positive spatial correlation. Of these, Moran's "I" is a more powerful test than either Geary's "C" or the Getis-Ord "G". By 'power' is meant the ability to correctly reject a false null hypothesis (or, in statistical language, to avoid a Type II error). A data set for which Moran's "I" is barely statistically significant might very well fail with Geary's "C" or the Getis-Ord "G" since the Geary and Getis-Ord indices are not a powerful as the Moran index.

However, only Moran's "I" and Geary's "C" are able to detect negative spatial autocorrelation. On the other hand, only the Getis-Ord "G" can distinguish between high positive and low positive spatial autocorrelation. The Moran and Geary tests would show these conditions to be identical, as the example below shows.

Example: Testing Houston Burglaries with the Getis-Ord "G"

Now, let us take the 26,480 burglaries in the City of Houston for 2006 aggregated to 1,179 traffic analysis zones (figure 5.2 above). To compare the Getis-Ord "G" statistic with the Moran's "I" and the regular Geary's "C", the three spatial autocorrelation tests were run on this data set. The Getis-Ord "G" was tested with a search distance of 1 mile and 1000 simulation runs were made on the "G". Table 5.2 shows the three global spatial autocorrelation statistics for these data.

The Moran and Geary tests show that the Houston burglaries have significant positive spatial autocorrelation (zones have values that are similar to their neighbors). Moran's "I" is significantly higher than the expected "I" and the adjusted Geary's "C" is also significantly higher than the adjusted expected "C". However, the Getis-Ord "G" is lower than the expected "G value and is significant whether using the theoretical Z-test or the simulated confidence intervals (notice how the "G" is lower than the 2.5 percentile). This indicates that, in general, zones with low values are nearby other zones with low values. In other words, there is low positive spatial autocorrelation, suggesting a number of 'cold spots'.

Uses and Limitations of the Getis-Ord "G"

The advantage of the "G" statistic over the other two spatial autocorrelation measures is that it can distinguish between 'hot spots' and 'cold spots'. With Moran's "I" or Geary's "C", an indicator of positive spatial autocorrelation means that zones have values similar to their neighbors. However, the positive spatial autocorrelation could be caused by many zones with low values being concentrated, too. In other words, one cannot tell from those two indices whether the concentration is a hot spot or a cold spot. The Getis-Ord "G" can do this.

Table 5.2:

Global Spatial Autocorrelation Statistics for City of Houston Burglaries: 2001

N = 1,179 Traffic Analysis Zones

	Moran's "I"	Adjusted Geary's "C"	Getis-Ord "G" (1 mile search)
Observed	0.251790	0.374298	0.007063
Expected	-0.000849	0.000000	0.061753
Observed –Expected	0.252639	0.374298	-0.054690
Standard Error	0.002796	0.018717	0.007581
Z-test	90.36	20.00	-7.21
p-value	****	****	****
Based on simulation:			
2.5 percentile:	n.a.	n.a.	0.048664
97.5 percentile:	n.a.	n.a.	0.076445

n.s	not significant
*	p≤.05
**	p≤.01
***	p≤.001
****	p≤.0001

The main limitation of the Getis-Ord "G" is that it cannot detect negative spatial autocorrelation, a condition that, while rare, does occur. With the checkerboard pattern above (Figure 5.8), this test could not detect that there was negative spatial autocorrelation. For this condition, Moran's "I" or Geary's "C" would be more appropriate tests.

Moran Correlogram

Moran's "I", Geary's "C", and the Getis-Ord "G" indices are summary tests of global autocorrelation. That is, they summarize all the data with respect to spatial autocorrelation but do not distinguish different subsets. For examining particular sub-sets of data that are spatially autocorrelated, such as 'hot spots', 'cold spots' or space-time clusters, a different approach is required. Chapter 9 discusses the local Moran and local Getis-Ord statistics.

An alternative approach is to calculate the spatial autocorrelation statistics by different distance intervals. The *Moran* Correlogram calculates the "I" value by different distance intervals (or bins). When graphed, the plot indicates how concentrated or distributed is the spatial autocorrelation (Cliff and Haggett, 1988; Bailey and Gatrell, 1995). Essentially, a series of concentric circles is overlaid on the points and the Moran's I statistic is calculated for only

those points falling within each circle. The radius of the circle changes from a small circle to a very large one. As the circle increases, the "I" value approaches the global value.

In *CrimeStat*, the user can specify how many distance intervals (i.e., circles) are to be calculated. The default is 10, but the user can choose any other integer value. The routine takes the maximum distance between points and divides it into the number of specified distance intervals, and then calculates the "I" for those points falling within that radius.

Adjust for Small Distances

If the 'Adjust for small distances' box is checked, small distances are adjusted so that the maximum weighting is 1 (equation 5.4 above). This ensures that the "I" values for individual distances will not become excessively large or excessively small for points that are close together. The default value is no adjustment.

Simulation of Confidence Intervals

A permutation Monte Carlo simulation can be run to estimate approximate confidence intervals around the "I" value. Each simulation inputs random data and calculates the "I" value. The distribution of the random "I" values produce an approximate confidence interval for the actual (empirical) "I". To run the simulation, specify the number of simulations to be run (e.g., 100, 1000, 10000). The default is no simulations. The output percentiles are the 0.5^{th}, 2.5^{th}, 97.5^{th} and 99^{th}. Pairing the 2.5^{th} with the 97.5^{th} or the 0.5^{th} with the 99.will create approximate 95% or 99% confidence intervals.

Example: Moran Correlogram of Baltimore County Vehicle Theft and Population

For the three correlograms, we will use a different example than Houston burglaries. These are 1996 data on vehicle thefts from Baltimore County, MD. Figure 5.9 shows the distribution of 1996 vehicle thefts by Traffic Analysis Zones (TAZ) while figure 5.10 shows the Moran Correlogram for these thefts. Also shown in the graph are the maximum and minimum values from a Monte Carlo simulation of 1000 runs and the 2.5th and 97.5th percentiles to simulate approximate 95% confidence intervals (called 'credible intervals').

As seen, the "I" value at zero distance is about 0.60. As the distance between zones increase (i.e., the search circle radius gets larger), the "I" value drops off slowly until about 19 miles whereupon it approaches the global "I" value. Further, the curve for the "I" values is always higher than the 97^{th} percentile curve from the random simulation and indicating that vehicle thefts are more clustered than what would be expected on the basis of chance for all

Figure 5.9:

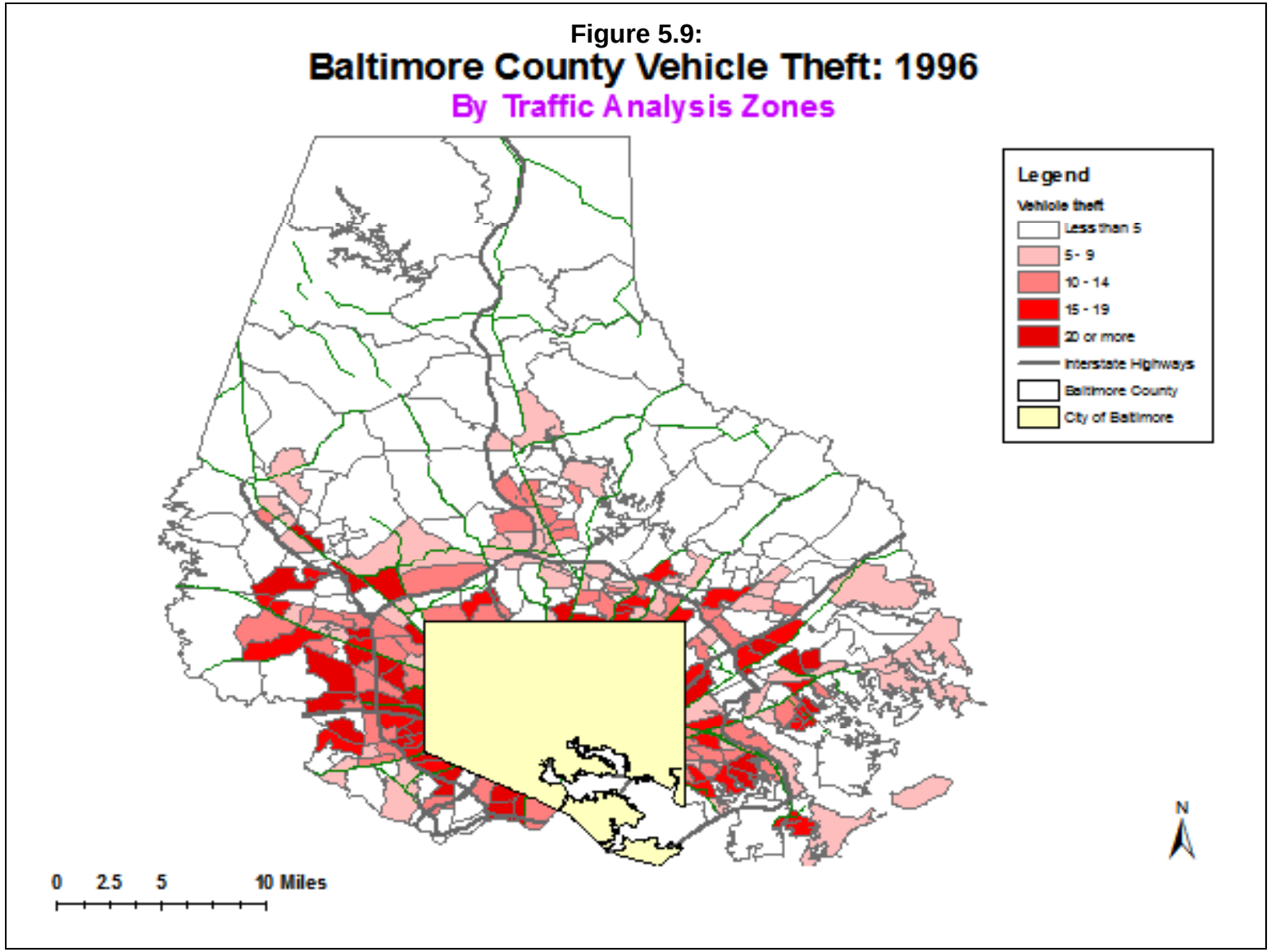
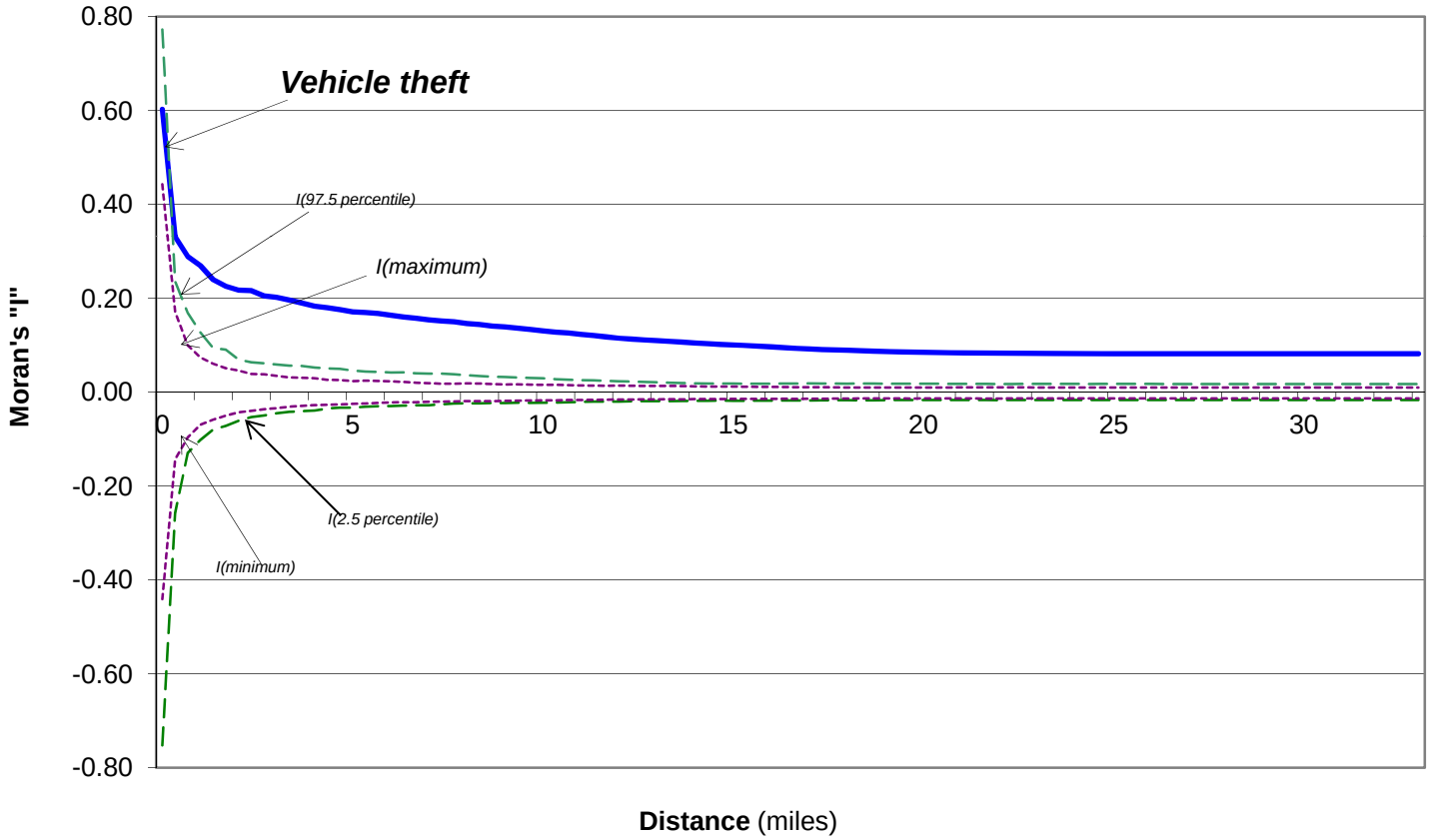

Baltimore County Vehicle Theft: 1996
By Traffic Analysis Zones
Legend
Vehicle theft
Less than 5
5 - 9
10 - 14
15 - 19
20 or more
Interstate Highways
Baltimore County
City of Baltimore
N
0 2.5 5 10 Miles

Figure 5.10:

Moran Correlogram: Baltimore County Vehicle Theft: 1996

"I" with 95% Confidence Intervals (N=1000 Monte Carlo Simulations)



distance separations. In other words, vehicle thefts appear to be highly clustered, much more so than would be expected by chance.

Now, compare this distribution with that the 1996 population (Figure 5.11). The 1996 data were estimated by the Baltimore Metropolitan Council, the regional planning agency. Comparing this map with Figure 5.9, intuitively it can be seen that population is more dispersed than vehicle thefts. Consequently, the Moran Correlogram shows much less spatial autocorrelation. The "I" value for zero distance is 0.39, lower than the 0.60 for vehicle thefts. The graph then drops off very quickly and approaches the global "I" value at about 3 miles. Further, from about 2 miles on, the "I" value is not different than what might be expected by chance since the curve falls between the 2.5^{th} percentile and the 97.5^{th} percentile. In other words, nearby TAZ's tend to have similar population levels, but there is no relationship between the population of TAZ's and those farther away.

Figure 5.13 compares the Moran Correlogram of vehicle theft with that of population by looking at only the positive "I" values. As seen, vehicle theft has a much higher "I" value for short distances than for population. The reason is most likely that a disproportionate number of vehicle thefts occur in commercial areas which, in turn, are more concentrated than the distribution of population.

Uses and Limitations of the Moran Correlogram

In other words, the Moran Correlogram provides information about the scale of spatial autocorrelation, whether it is more concentrated (as with the vehicle theft example) or more diffuse (as with the population example). This can be useful for gauging the extent to which 'hot spots' are truly isolated concentrations of incidents or whether they are by-products of spatial clustering over a larger area. In Chapter 7, we will examine a clustering algorithm that examines a hierarchy of clusters (e.g., first-order clusters that are within larger second-order clusters which, in turn, are within even larger third-order clusters). The Moran Correlogram provides a quick snapshot of the extent of spatial autocorrelation as a function of scale.

A second use for the Moran Correlogram is to estimate the type of kernel function that will be used for interpolation. In Chapter 8, this methodology is explained in detail. But, the key decision is to select a mathematical function that will interpolate data from point locations to grid cells. The shape of the Moran Correlogram and the spread is a good indicator of the type of mathematical function to use.

Figure 5.11:

Baltimore County Population: 1996 (estimated)

By Traffic Analysis Zones

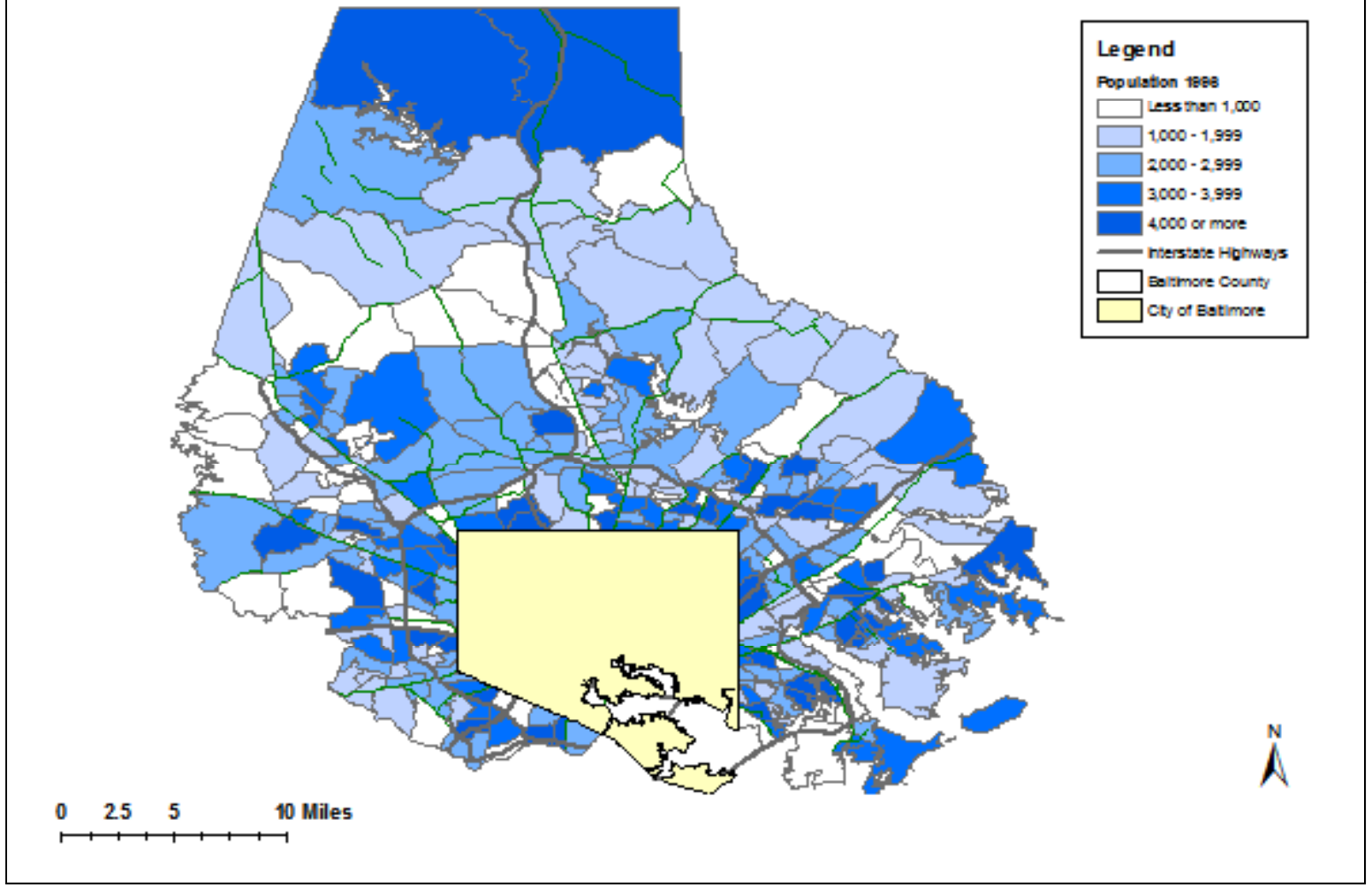


Figure 5.12:

Moran Correlogram: Baltimore County Population: 1996

"I" with 95% Confidence Intervals (N=1000 Monte Carlo Simulations)

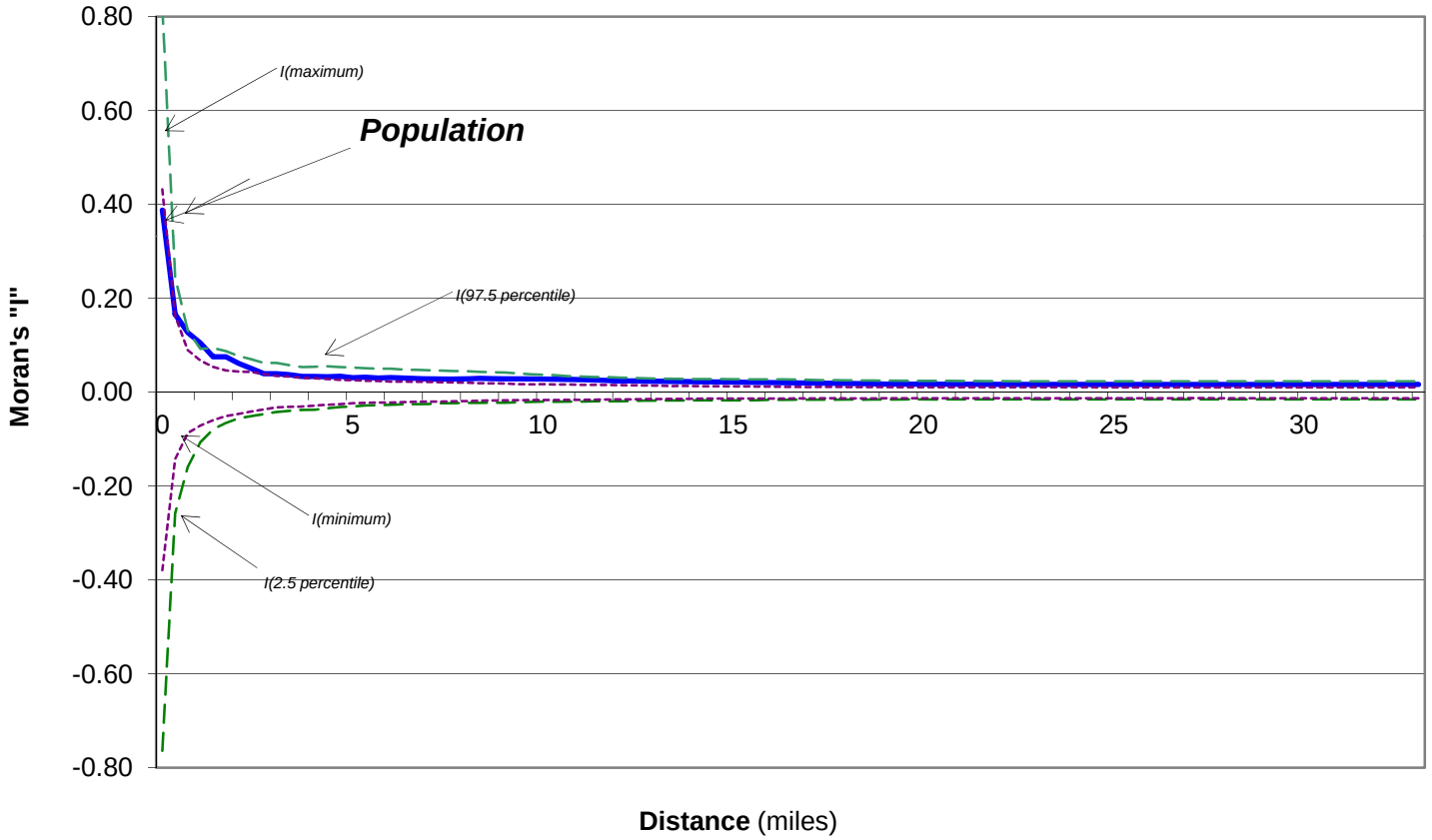
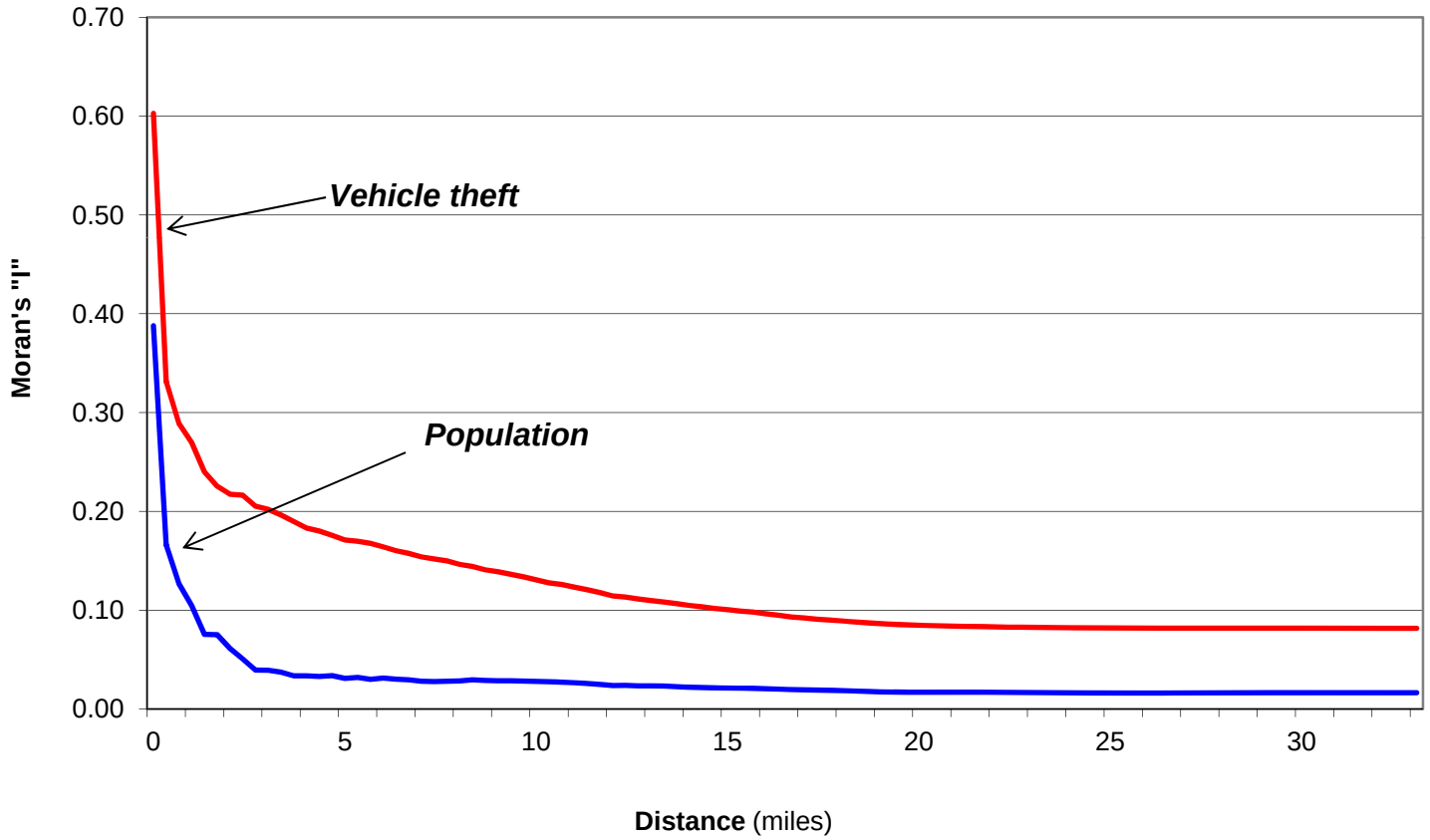


Figure 5.13:

Two Moran Correlograms of Baltimore County Vehicle Theft & Population: 1996



A third use for the Moran Correlogram is in identifying the degree of decline in spatial autocorrelation with distance (sometimes called *distance decay*) in choosing an appropriate parameter for spatial regression models. Chapter 19 will discuss this methodology.

On the other hand, like all global spatial autocorrelation statistics, the Correlogram will not indicate where there is clustering or dispersion, only that it exists. For that, we will have to examine tools that focus on concentrated events (or the opposite, the lack of concentration).

Geary Correlogram

The Geary Correlogram is similar to the Moran Correlogram in that it calculates the Geary "C" index for different distance intervals/bins. The user can select any number of distance intervals. The default is 10 distance intervals. The size of each interval is determined by the maximum distance between zones and the number of intervals selected. The output includes both the regular "C" and the adjusted "C". The graph presented on the results tab show the adjusted "C" since this is more intuitive and can be compared to the Moran Correlogram.

Adjust for Small Distances

If the 'Adjust for small distances' box is checked, small distances are adjusted so that the maximum weighting is 1 (see equation 5.4 above.) This ensures that the "C" values for individual distances won't become excessively large or excessively small for points that are close together. The default value is no adjustment.

Geary Correlogram Simulation of Confidence Intervals

Since the Geary's "C" statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a permutation Monte Carlo simulation is run whereby the original values of the variable, Z, are maintained but are randomly re-assigned for each simulation run. This will maintain the distribution of the variable Z but will estimate the value of "C" under random assignment of this variable. Specify the number of simulations to be run (e.g., 1000, 5000, 10000). Note, a simulation may take time to run especially if the data set is large or if a large number of simulation runs are requested.

Example: Geary Correlogram of Baltimore County Vehicle Thefts

Using the same data set on the Baltimore County vehicle thefts as shown in Figure 5.9 above, the Geary Correlogram was run with 100 intervals (bins). The routine was also run with 1000 simulations to estimate confidence intervals around the "C" value. Because it is more intuitive visually, the adjusted "C" was used instead of the regular "C".

Figure 5.14 illustrates the distance decay of the adjusted "C" as a function of distance along with the simulated 95% confidence interval. The theoretical adjusted "C" under random conditions is also shown. As seen, the "C" values are above 0 for all distances tested. However, when compared with the 2.5th and 97.5th percentiles from the simulated rescaled "C" for all intervals, the adjusted "C" values are not outside these percentiles for the very short distances but are from about 1.5 miles separation or greater. In other words, the graph suggests that the distribution of "C" for nearby zones is not different than what would be expected by chance. Only with increasing distance is the distribution clearly more clustered than chance.

This illustrates a subtle difference between the Geary and Moran indices. The Geary is more sensitive to local variations while the Moran reacts more to global variations. The Geary shows that there is positive spatial autocorrelation in vehicle theft for the immediate neighborhood around zones, but it is not much different than might be expected on chance. However, with increasing distance, positive spatial autocorrelation is shown. This suggests a type of sub-regional clustering of vehicle thefts; local clustering is limited but the events tend to be concentrated in only part of Baltimore County. As seen in Figure 5.9 above, the TAZ's nearer the border with the City of Baltimore had much higher vehicle theft numbers than the rural parts of the County.

The Geary Correlogram can also be used for comparison to other distributions, such as the comparison of vehicle theft with population as shown in Figure 5.13. This example will not be repeated here for the Geary Correlogram, but it does show that vehicle theft has higher "C" values than population over most distances, similar to the Moran Correlogram.

Uses and Limitations of the Geary Correlogram

Similar to the Moran and the Getis-Ord correlograms (see below), the Geary Correlogram is useful in order to determine the degree of spatial autocorrelation and how far away from each zone it typically extends. Since it is an average over all zones, it is a general indicator of the spread of the spatial autocorrelation. This can be useful for defining limits to search distances in other routines, such as the single kernel density interpolation routine where a fixed bandwidth would be defined to capture the majority of spatial autocorrelation. Its biggest limitation is that it is not as powerful a test as the Moran Correlogram.

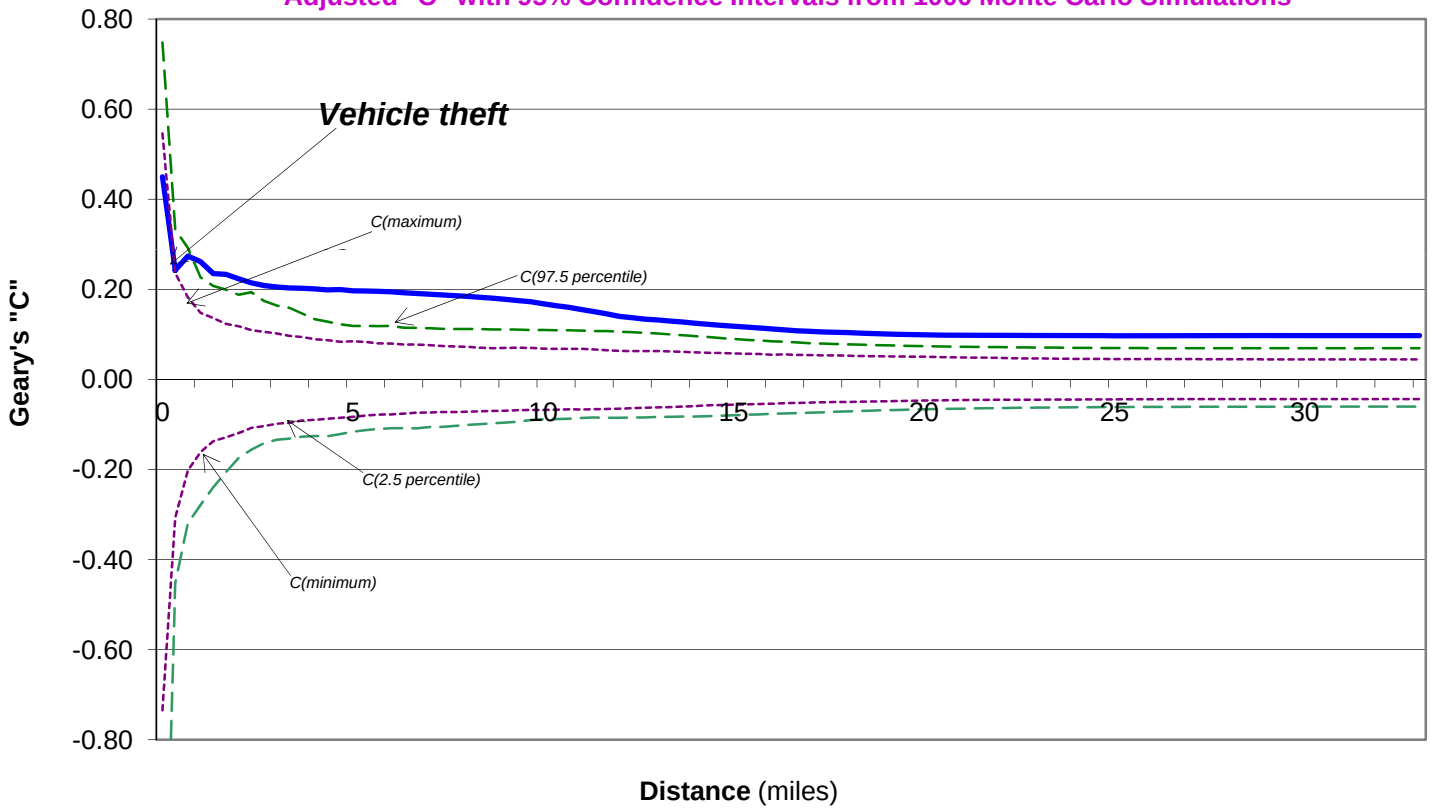
Getis-Ord Correlogram

The Getis-Ord Correlogram calculates the Getis-Ord "G" index for different distance intervals/bins. The statistic requires an intensity variable in the primary file and calculates the Getis-Ord "G" index for different distance intervals/bins. The user can select any number of

Figure 5.14:

Geary Correlogram:
Baltimore County Vehicle Theft: 1996

Adjusted "C" with 95% Confidence Intervals from 1000 Monte Carlo Simulations



distance intervals. The default is 10 distance intervals. The size of each interval is determined by the maximum distance between zones and the number of intervals selected.

Getis-Ord Correlogram Simulation of Confidence Intervals

Since the Getis-Ord "G" statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a permutation Monte Carlo simulation is run whereby the original values of the intensity variable, Z, are maintained but are randomly re-assigned for each simulation run. This will maintain the distribution of the variable Z but will estimate the value of "G" under random assignment of this variable. The user should specify the number of simulations to be run (e.g., 100, 1000, 10000). Note, a simulation may take time to run especially if the data set is large or if a large number of simulation runs are requested.

If a simulation is run, percentiles for the 0.5^{th}, 2.5^{th}, 97.5^{th} and 99^{th} percentiles are provided. Pairing the 2.5^{th} with the 97.5^{th} or the 0.5^{th} with the 99.will create approximate 95% or 99% confidence intervals. For the three correlograms, these statistics are provided for each of the distance bins.

Example: Getis-Ord Correlogram of Baltimore County Vehicle Thefts

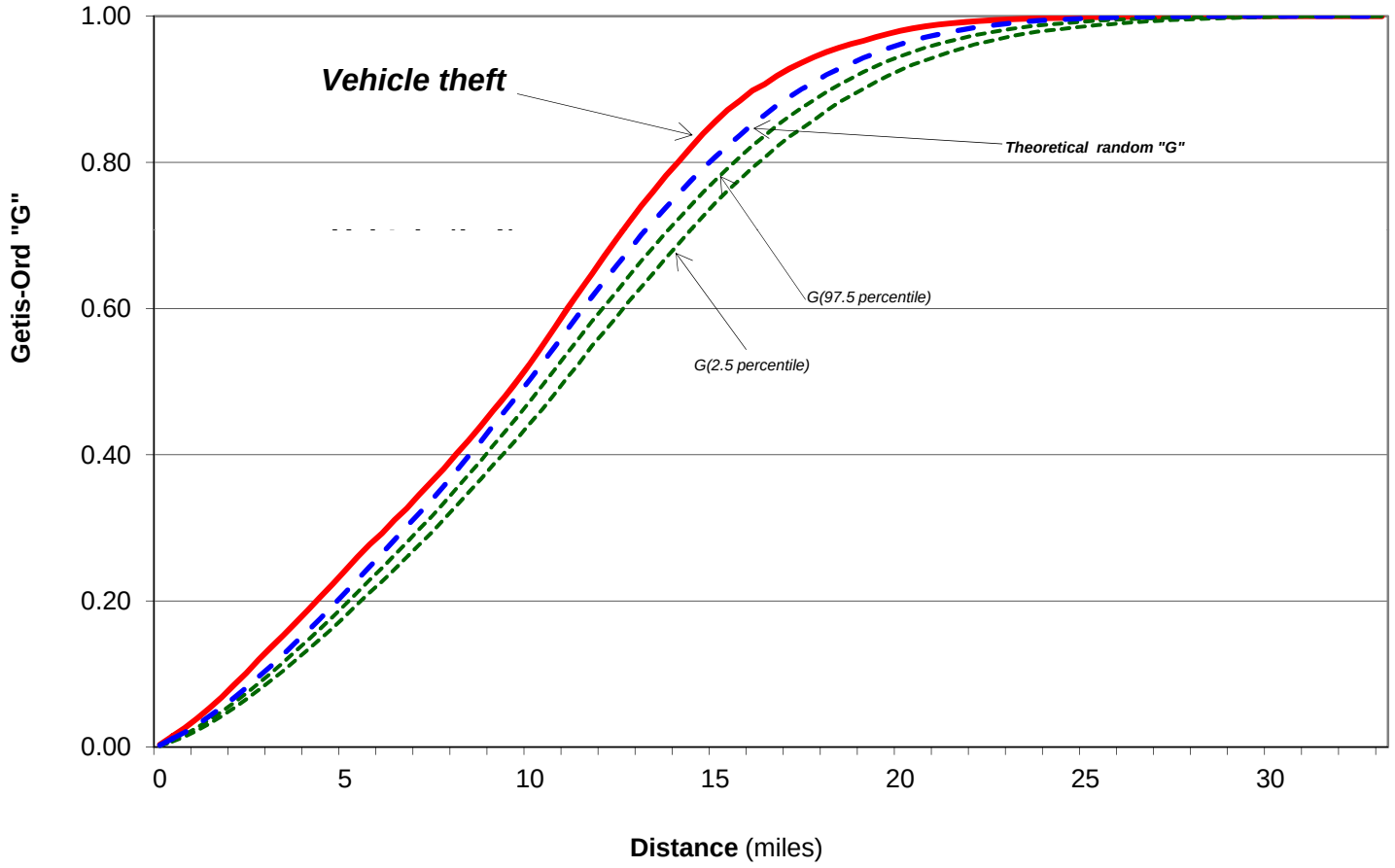
Using the same data set on the Baltimore County vehicle thefts as in figure 5.9, the Getis-Ord Correlogram was run. The routine was run with 100 intervals and 1000 Monte Carlo simulations in order to simulate 95% confidence intervals around the "G" value. The output was then brought into Excel to produce a graph. Figure 5.15 illustrates the distance decay of the "G", the expected "G", and the 2.5 and 97.5 percentile "G" values from the simulation.

Note that the "G" value *increases* with distance from close to 0 to close to 1 at the largest distance, around 33 miles. The actual "G" is higher than the expected "G" for all distances until the maximum, indicating that there is consistent high positive spatial autocorrelation in the data set. Since the Getis-Ord can distinguish a hot spot from a cold spot, the excess of "G" over the expected "G" indicates that there are somes zones with substantial numbers of vehicle thefts. Notice how the expected "G" also falls above the 97.5 percentile suggesting that there are more 'hot spots' than 'cold spots'. That is, if the zones were spatially re-arranged, then would not expect as much concentration as actually occurred.

Figure 5.15:

Getis-Ord Correlogram: Baltimore County Vehicle Theft: 1996

"G" with 95% Confidence Intervals from 1000 Monte Carlo Simulations



Uses and Limitations of the Getis-Ord Correlogram

Similar to the Moran Correlogram and the Geary Correlogram, the Getis-Ord Correlogram is useful in order to determine the degree of spatial autocorrelation and how far away from each zone it typically extends. Since it is an average over all zones, it is a general indicator of the spread of the spatial autocorrelation. This can be useful for defining limits to search distances in other routines, such as the single kernel density interpolation routine or the MCMC spatial regression module (see Chapters 10 and 19).

Unlike the other two correlograms, however, it can distinguish hot spots from cold spots. In the example above, there are more hot spots than cold spots since the "G" is greater than the expected "G" for all distances. The biggest limitation for the Getis-Ord Correlogram is that it cannot detect negative spatial autocorrelation whereby zones have different values from their neighbors. For that condition, which is rare, the other two correlograms should be used.

Running the Spatial Autocorrelation Routines

The six routines are defined on the Spatial Autocorrelation tab under spatial description. With the Moran and Geary routines, the user simply checks the box for each routine. If distance is to be adjusted for small distances, the user must check the appropriate box. For the Getis-Ord "G" routine, the user must specify a search distance and a unit of distance measurement (the default is 1 mile). For the three correlograms, the user must specify the number of intervals and the number of simulations that are to be run, if any.

The output for the six routines is somewhat similar. For the three global indices, statistics are provided on the index ("I", "C" or"G") and the expected value. For the three correlograms, these statistics are provided for each of the distance bins. If a simulation is run, percentiles for the 0.5^{th}, 2.5^{th}, 97.5^{th} and 99^{th} percentiles are provided. Pairing the 2.5^{th} with the 97.5^{th} or the 0.5^{th} with the 99.will create approximate 95% or 99% confidence intervals.

Guidelines for Examining Spatial Autocorrelation

To summarize, a number of indices for examining spatial autocorrelation have been presented. These indices are used with data in which there is an attribute variable, a count or interval variable associated with specific locations. Typically, the indices are used with data on zones since zonal information is published by many different agencies. However, the indices could also be used with individual data if there are attributes associated with the individual records.

While there is no single way to utilize these indices, the following are suggestions for using them. First, identify whether there is positive spatial autocorrelation using Moran's "I" and Geary's "C". Positive spatial autocorrelation indicates that zones are located near to other zones with similar values, either zones with high values on the variable being located near to zones also with high values or the opposite condition (low values nearby other low values).

If both the Moran "I" and Geary "C" (either regular or adjusted values) are both significant, this is strong evidence that there is sizeable spatial autocorrelation in the data. Whether the spatial autocorrelation is due to global (regional) factors or local clustering cannot be easily determined from the indices. On the other hand, if the Moran is significant, but the Geary is not, this could indicate that the clustering is a function of global concentration rather than local concentration since the Moran index is more sensitive to region-wide variation in the variable.

If there is negative spatial autocorrelation, which does occasionally happen, this indicates that zones with high values are located near to zones with low values, or the opposite. The user is advised to use one of the hot spot techniques described in Chapters 7, 8 and 9 to see if the hot spots can be isolated.

Second, if there is positive spatial autocorrelation, identify the type using the Getis-Ord "G" statistic. The Getis-Ord "G" is only applicable for positive spatial autocorrelation but can distinguish a predominance of high positive or low positive. High positive means that there are more zones with high values located near to other zones also with high values whereas low positive means the opposite (low near to low). The index is a type of average that weights the predominance of these types. In practice, there will be both types but the index indicates which is stronger. Since the Getis-Ord "G" requires a search distance, the user may have to run the Getis-Ord Correlogram first in order to identify a distance for which the positive spatial autocorrelation is most distinguishable from the theoretical random "G".

Third, examine the decline of the spatial autocorrelation with distance by using the three correlograms. While the Moran and Geary correlograms can be used for both positive and negative spatial autocorrelation, the Getis-Ord correlogram can only be used with positive spatial autocorrelation. The three correlograms will indicate how spatial autocorrelation varies by distance from each zone, on average. They can provide useful information about whether the concentration is very large, such as concentrated in the center of a metropolitan area, in which case the spatial autocorrelation is primarily a function of global factors. Alternatively, if the indices fall off very quickly, this suggests neighborhood (or local) effects rather than a dominant global pattern. In practice, there will be both types of factors, but the correlograms can indicate which is most important.

As with the global indices, the correlograms can provide useful information about the rate of decline in spatial autocorrelation (distance decay) for the kernel density routines (Chapter 10), the journey-to-crime routine (Chapter 13), the spatial regression routines (Chapter 19), or the trip distribution module of the Crime Travel Demand Model (Chapter 28).

In other words, identifying whether there is spatial autocorrelation and, if so, the type is important with zonal data (or with individual records having attributes) in that it is a first step in understanding where and why that spatial autocorrelation occurs. It is a necessary step in conducting hot spot analysis and in modeling the predictive factors that cause the spatial autocorrelation to occur. Chapter 9 examines hot spot identification routines appropriate for zonal data or individual data with attributes while Chapter 19 examines various regression tools for modeling the predictors of the spatial autocorrelation.

References

Anselin, L. (2008). Personal note on the testing of significance of the local Moran values.

Anselin, L. (1995). Local indicators of spatial association - LISA. *Geographical Analysis*. 27, No. 2 (April), 93-115.

Anselin, L.. (1992). *SpaceStat: A Program for the Statistical Analysis of Spatial Data*. Santa Barbara, CA: National Center for Geographic Information and Analysis, University of California.

Bailey, T. C. & Gatrell, A. C. (1995). *Interactive Spatial Data Analysis*. Longman Scientific & Technical: Burnt Mill, Essex, England.

Cliff, A. D. & Haggett, P. (1988). *Atlas of Disease Distributions*. Blackwell Reference: Oxford.

Cliff, A. & Ord, J. (1973). *Spatial Autocorrelation*. Pion: London.

Ebdon, D. (1988). *Statistics in Geography* (second edition with corrections). Blackwell: Oxford.

Freedman, D. A. (1999). Ecological inference and ecological fallacy. *International Encyclopedia of the Social and Behavioral Sciences*, Technical Report No. 549, October. http://www.stanford.edu/class/ed260/freedman549.pdf. Accessed March 26, 2012.

Geary, R. (1954). The contiguity ratio and statistical mapping. *The Incorporated Statistician*, 5, 115-145.

Getis, A. & Ord, J. K. (1996). Local spatial statistics: an overview. In Longley, P. & Batty, M. (eds), *Spatial Analysis: Modelling in a GIS Environment*. GeoInformation International: Cambridge, England, 261-277.

Getis, A. & Ord, J. K. (1992). The analysis of spatial association by use of distance statistics, *Geographical Analysis*, 24, 189-206.

Griffith, D. A. (1987). *Spatial Autocorrelation: A Primer*. Resource Publications in Geography, The Association of American Geographers: Washington, DC.

Khan, G., Qin, X. & Noyce, D. A. (2006). Spatial analysis of weather crash patterns in Wisconsin. 85th Annual meeting of the Transportation Research Board: Washington, DC.

References (continued)

Langbein, L. I. & Lichtman, A. J. (1978). *Ecological Inference*. Sage University Paper series on Quantitative Applications in the Social Sciences, series no. 07-010. Beverly Hills and London: Sage Publications.

Lee, J. & Wong, D. W. S. (2005). *Statistical Analysis with ArcView GIS and ArcGIS*. J. Wiley & Sons, Inc.: New York.

Lees, B. (2006). The spatial analysis of spectral data: Extracting the neglected data, *Applied GIS*, 2 (2), 14.1-14.13.

Levine, N. (1999). The effects of local growth management on regional housing production and population redistribution in California, *Urban Studies*. 1999. 36 12, 2047-2068.

Levine, N. & Lee, P. (2013). Crime travel of offenders by gender and age in Manchester, England. Leitner, M. (ed), *Crime Modeling and Mapping Using Geospatial Technologies,* Springer. 145-178.

Moran, P. A. P. (1950). Notes on continuous stochastic phenomena. *Biometrika*, 37, 17-23.

Ord, J. K. & Getis, A. (1995). Local spatial autocorrelation statistics: Distributional Issues and an Application. *Geographical Analysis*, Vol. 27, 1995, 286-306.

Ripley, B. D (1981). *Spatial Statistics*. John Wiley & Sons: New York.

Ripley, B. D. (1976). The second-order analysis of stationary point processes. *Journal of Applied Probability* 13: 255-66.

Attachments

Global Moran's I and Small Distance Adjustment: Spatial Pattern of Crime in Tokyo

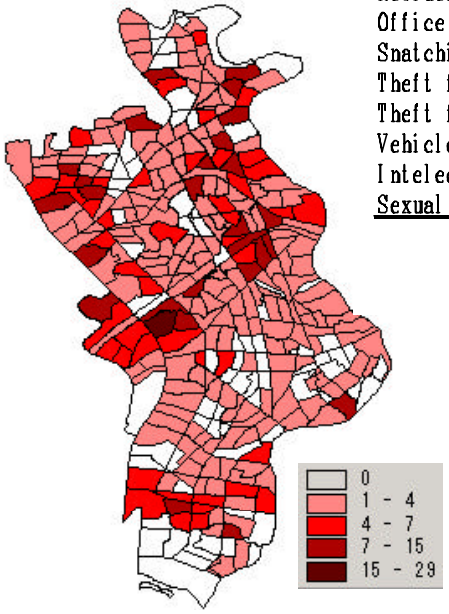
Takahito Shimada
National Research Institute of Police Science
National Police Agency, Chiba, Japan

Crimestat calculates spatial autocorrelation indicators such as Moran's I and Geary's C. These indicators can be used to compare the spatial patterns among crime types. Moran's I is calculated based on the spatial weight matrix where the weight is the inverse of the distance between two points. There is a problem that could occur for incident locations in that the weight could become very large as the distance between points become closer. In *Crimestat*, the small distance adjustment is available to solve this problem. The adjustment produces a maximum weight of 1 when the distance between points is 0.

The number of reported crimes in Tokyo increased from 1996 to 2000 although the city is generally very safe. For this analysis, 68,400 cases reported in the eastern parts of Tokyo were aggregated by census tracts (N=350). Then *Crimestat* calculated Moran's I for each crime type with and without the small distance adjustment.

The "I" value for most crime types, including burglary, theft, purse snatching, showed significantly positive autocorrelation. The results with and without the small distance adjustment were generally very close. The Pearson's correlation between the original and adjusted Moran's I is .98. Among 10 crime types, relatively strong spatial patterns were detected for car theft, sexual assaults, and residential burglary.

Spatial Patterns of Residential Burglary:
Moran's I = 0.023. z=7.58



Calculated Moran's I by Crime Types

Crime Type	Original Moran's I	z		Adjustment Moran's I	z	
Felonious Offense	0.018	4.09	**	0.003	0.96	
Violent Offense	0.030	6.27	**	0.007	3.03	**
Residential Burglary	0.055	11.21	**	0.023	7.58	**
Office Burglary	0.028	5.93	**	0.012	4.34	**
Snatching	0.031	6.48	**	0.006	2.45	*
Theft from Vender	0.030	6.38	**	0.012	4.28	**
Theft from Cars	0.081	16.08	**	0.044	13.75	**
Vehicle Theft	0.047	9.65	**	0.018	6.14	*
Intelectual Offense	0.023	4.99	**	0.003	1.79	
Sexual Assault	0.080	16.00	**	0.045	14.04	**

**: p<.01 *: p<.05

Preliminary Statistical Tests for Hotspots: Examples from London, England

Spencer Chainey
Jill Dando Institute of Crime Science
University College
London, England

Preliminary statistical tests for clustering and dispersion can provide insight into what types of patterns will be expected when the crime data is mapped. Global tests can confirm whether there is statistical evidence of clusters (i.e. hotspots) in crime data which can be mapped, rather than mapping data as a first step and struggling to accurately identify hotspots when none actually exist.

Using *CrimeStat*, four statistical tests were compared for robbery, residential burglary and vehicle crime data for the London Borough of Croydon, England. For the incident data, the standard distance deviation and nearest neighbor index were used. For crime incidents aggregated to Census block areas, Moran's I and Geary's C spatial autocorrelation indices were compared. The crime data is for the period June 1999 – May 2000.

Crime type	Number of crime records	Standard distance	NN Index	z-score (test statistic)	***Evidence of Clustering?***
Robbery	1132	3119.5 m	0.47	-34.2	***Yes***
Residential burglary	3104	3664.6 m	0.46	-57.5	***Yes***
Vehicle crime	9314	3706.2 m	0.26	-137.0	***Yes***

Crime type	**Moran's I**	**Geary's C**
All crime	0.0067	1.14
Robbery	0.0078	1.15
Residential burglary	0.014	0.99
Vehicle crime	0.0082	1.08

With the point statistics, all three crime types show evidence of clustering. Vehicle crime shows the more dispersed pattern suggesting that whilst hotspots do exist, they may be more spread out over the Croydon area than that of the other two crime types. For the two spatial autocorrelation measures, there are differences in the sensitivities of the two tests. For example, for robbery, there is evidence of global positive spatial autocorrelation (overall, Census blocks that are close together have similar values than those that are further apart). On the other hand, the Geary coefficient suggests that, at a smaller neighbourhood level, areas with a high number of robberies are surrounded by areas with a low number of robberies.

Chapter 6:

Distance Analysis I and II

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 6:

Distance Analysis I and II

In this chapter, the characteristics of the distances between points will be described. The previous chapter provided tools for describing the general spatial distribution of crime incidents or *first-order* properties of the incident distribution (Bailey and Gattrell, 1995). First-order properties are global because they represent the dominant pattern of distribution - where the points are centered, how far they spread out, and whether there is any orientation to the dispersion. *Second-order* (or *local*) properties, on the other hand, refer to sub-regional or 'neighborhood' patterns within the overall distribution. If there are distinct 'hot spots' where many crime incidents cluster together, their distribution is spatially related to something unique in the sub-region or neighborhood, and less to the global distribution Second-order characteristics indicate how particular environments concentrate crime incidents.

There are two distance analysis pages. In Distance analysis I, various second-order statistics are provided, including:

1. NN
2. Linear NN
3. Ripley
4. Assign primary points to secondary points

In Distance analysis II, there are four routines for calculating and outputting distance matrices. This chapter will discuss both sets of routines.

Distance Analysis I

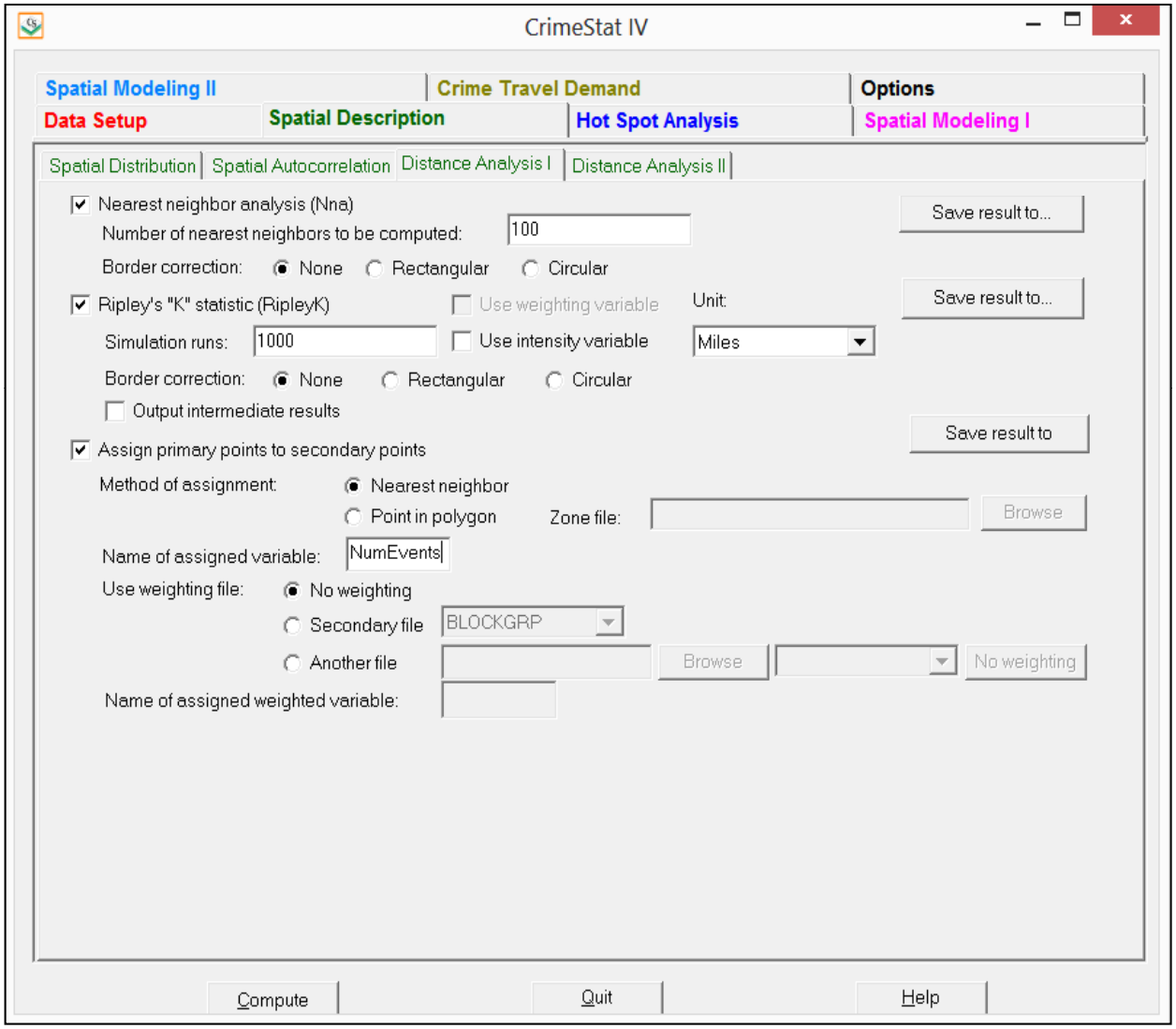
Figure 6.1 shows the Distance analysis I screen and the distance statistics on that page that are calculated by *CrimeStat*.

Nearest Neighbor Index

One of the oldest distance statistics is the *nearest neighbor index*. It is particularly useful because it is a simple tool to understand and to calculate. It was developed by two botanists in

Figure 6.1:

Distance Analysis I Screen



the 1950s (Clark and Evans, 1954), primarily for field work, but it has been used in many different fields for a wide variety of problems (Cressie, 1991). It has also become the basis of many other types of distance statistics, some of which are implemented in *CrimeStat*.

The nearest neighbor index compares the distances between nearest points and distances that would be expected on the basis of chance. It is an index that is the ratio of two summary measures. First, there is the *nearest neighbor distance*. For each point (or incident location) in turn,i, the distance to every other point, j, is calculated and minimum selected (the nearest neighbor). The nearest neighbors are then averaged over all points:

$$d_{NN} = \sum_{i=1}^{N} \sum_{i \neq j=1}^{N-1} \frac{Min(d_{ij})}{N} \tag{6.1}$$

where Min(d_{ij}) is the distance between each point and its nearest neighbor and N is the number of points in the distribution. Thus, in *CrimeStat*, the distance from a single point to every other point is calculated and the smallest distance (the minimum) is selected. Then, the next point is taken and the distance to all other points (including the first point measured) is calculated with the nearest being selected and added to the first minimum distance. This process is repeated until all points have had their nearest neighbor selected. The total sum of the minimum distances is then divided by N, the sample size, to produce an average minimum distance.

The second summary measure is the expected nearest neighbor distance if the distribution of points is completely spatially random. This is the *mean random distance* (or the mean random nearest neighbor distance). It is defined as:

$$d_{NN(ran)} = 0.5\sqrt{\frac{A}{N}} \tag{6.2}$$

where A is the area of the region and N is the number of incidents. Since A is defined by the square of the unit of measurement (e.g., square mile, square meters, etc.), it yields a random distance measure in the same units (i.e., miles, meters, etc.).[1] If defined on the measurement

[1] There is also a mean random distance for a dispersed pattern, called the *mean dispersed distance* (Ebdon, 1988). It is defined as:

$$d_{dispersed} = \frac{\sqrt{2}}{3^{1/4}\sqrt{\frac{N}{A}}}$$

where N is the number of points and A is the area. A nearest neighbor index can be set up comparing the observed mean neighbor distance with that expected for a dispersed pattern. *CrimeStat* only provides the traditional nearest neighbor index, but it does output the mean dispersed distance.

parameters page by the user, *CrimeStat* will use the specified area in calculating the mean random distance. If no area measurement is provided, *CrimeStat* will take the rectangle defined by the minimum and maximum X and Y points.

The nearest neighbor index is the ratio of the observed nearest neighbor distance to the mean random distance

$$NNI = \frac{d_{NN}}{d_{NN(ran)}} \tag{6.3}$$

Thus, the index compares the average distance from the closest neighbor to each point with a distance that would be expected on the basis of chance. If the observed average distance is about the same as the mean random distance, then the ratio will be about 1.0. On the other hand, if the observed average distance is smaller than the mean random distance, that is, points are actually closer together than would be expected on the basis of chance, then the nearest neighbor index will be less than 1.0. This is evidence for clustering. Conversely, if the observed average distance is greater than the mean random distance, then the index will be greater than 1.0. This would be evidence for dispersion, that points are more widely dispersed than would be expected on the basis of chance.

Testing the Significance of the Nearest Neighbor Index

Some differences from 1.0 in the nearest neighbor index would be expected by chance. Clark and Evans (1954) proposed a Z-test to indicate whether the observed average nearest neighbor distance was significantly different from the mean random distance (Hammond and McCullagh, 1978; Ripley, 1981). The test is between the observed nearest neighbor distance and that expected from a random distribution and is given by:

$$Z = \frac{d_{NN} - d_{NN(ran)}}{SE_{d(ran)}} \tag{6.4}$$

where the standard error of the mean random distance is approximately given by:

$$SE_{d(ran)} \cong \sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \tag{6.5}$$

with A being the area of region and N the number of points. There have been other suggested tests for the nearest neighbor distance as well as corrections for edge effects (see below).

However, equations 6.4 and 6.5 are used most frequently to test the average nearest neighbor distance. See Cressie (1991) for details of other tests.

Calculating the Statistics

Once nearest neighbor analysis has been selected, the user clicks on *Compute* to run the routine. The program outputs 11 statistics:

1. The sample size
2. The mean nearest neighbor distance
3. The standard deviation of the nearest neighbor distance
4. The minimum distance
5. The maximum distance
6. The mean random distance for both the bounding rectangle and the user input area, if provided
7. The mean dispersed distance for both the bounding rectangle and the user input area, if provided
8. The nearest neighbor index for both the bounding rectangle and the user input area, if provided
9. The standard error of the nearest neighbor index for both the maximum bounding rectangle and the user input area, if provided
10. A significance test of the nearest neighbor index (Z-test)
11. The p-values associated with a one tail and two tail significance test.

In addition, the output can be saved to a '.dbf' file, which can then be imported into spreadsheet or graphics programs.

Example 1: The Nearest Neighbor Index for Baltimore County Street Robberies

In 1996, there were 1,181 street robberies in Baltimore County. The area of the County is about 607 square miles and is specified on the measurement parameters page. *CrimeStat* returns the statistics shown in Table 6.1 with the NNA routine. The mean nearest neighbor distance was 0.116 miles while the mean nearest neighbor distance under randomness was 0.358. The nearest neighbor index (the ratio of the actual to the random nearest neighbor distance) is 0.3236. The Z-value of -44.4672 is highly significant. In other words, the distribution of the nearest neighbors of street robberies in Baltimore County is significantly smaller than what would be expected randomness.

It should be noted that the significance test for the nearest neighbor index is not a test for complete spatial randomness, for which it is sometimes mistaken. It is only a test whether the average nearest neighbor distance is significantly different than what would be expected on the basis of chance. In other words, it is a test of *first-order* nearest neighbor randomness.[2] There are also second-order, third-order, and so forth distributions that may or may not be significantly different from their corresponding orders under complete spatial randomness. A complete test would have to test for all those effects, what are called *K-order* effects.

Table 6.1:
Nearest Neighbor Statistics for
1996 Street Robberies in Baltimore County
(N=1181)

Mean nearest neighbor distance:	0.11598 mi
Mean random distance based on user input area:	0.35837 mi
Nearest neighbor index:	0.3236
Standard error:	0.00545 mi
Test Statistic (Z):	-44.4672
p-value (one tail)	≤.0001
p-value (two tail)	≤.0001

Example 2: The Nearest Neighbor Index for Baltimore County Residential Burglaries

The nearest neighbor index and test can be very useful for understanding the degree of clustering of crime incidents in spite of its limitations. For example, in Baltimore County, the distribution of 6051 residential burglaries in 1996 yields the following nearest neighbor statistics (Table 6.2).

The distribution of residential burglaries is also highly significant. Now, suppose we want to compare the distribution of street robberies (table 6.1) with that of residential burglaries

[2] Unfortunately, the term *order* when used in the context of nearest neighbor analysis has a slightly different meaning than when used as *first-order* compared to *second-order* statistics. In the nearest neighbor context, *order* really means *neighbor* whereas in the type of statistics context, *order* means the scale of the statistics, global or local. The use of the terms is historical

(table 6.2). The significance test is not very useful for the comparison because the sample sizes are so large (1181 v. 6051); the much higher Z-value for residential burglaries indicates primarily that there was a larger sample size to test it.

Table 6.2:
Nearest Neighbor Statistics for
1996 Residential Burglaries in Baltimore County
(N=6051)

Mean nearest neighbor distance:	0.07134 mi
Mean random distance based on user input area:	0.16761 mi
Nearest neighbor index:	0.4256
Standard error:	0.00113 mi
Test Statistic (Z):	-85.4750
p-value (one tail)	≤.0001
p-value (two tail)	≤.0001

However, comparing the relative nearest neighbor indices can be meaningful,

$$Relative\ NN\ Comparison = \frac{NNI_A}{NNI_B} \tag{6.6}$$

where NNI(A) is the nearest neighbor index for one group (A) and NNI(B) is the nearest neighbor index for another group (B). Thus, comparing street robberies with residential burglaries, we have:

$$\frac{NNI_A}{NNI_B} = \frac{NNI_{robberies}}{NNI_{burglaries}} = \frac{0.3236}{0.4256} = 0.7603 \tag{6.7}$$

In other words, the distribution of street robberies relative to an expected random distribution appears to be more concentrated than that of burglaries. There is not a simple significance test of this comparison since the standard error of the joint distributions is not known.[3] But the relatively greater concentration of robberies suggests that they are more likely to have 'hot spots'.

[3] It could be tested with a Monte Carlo simulation. Two separate random samples of 1181 'robberies' and 6051 'burglaries' would be drawn. The nearest neighbor distance for each sample would be calculated and the ratio of the two would be taken. The simulation would be repeated many times (e.g., 1000) to yield an approximate 95% credible interval. However, we have not implemented this simulation at this point.

This index, of course, does not prove that there are 'hot spots', but only points us towards the higher concentration of robberies relative to burglaries. In the previous chapter, it was shown that robberies had a smaller dispersion than burglaries. Here, however, the analysis is taken a step further to suggest that robberies are more concentrated than burglaries.

Use of Network Distance

In calculating the nearest neighbor index, network distance can be used to calculate the distance between points (see chapter 3). However, unless the data set is very small or you have a lot of patience, I highly recommend that you **do not** do this. Network calculations are very slow and will take a long time to complete for a large file.

K-Order Nearest Neighbor

As mentioned above, the nearest neighbor index is only an indicator of first-order spatial randomness. It compares the average distance for the nearest neighbor to an expected random distance. But what about calculating the second nearest neighbor, or the third nearest neighbor, or the 10th nearest neighbor? *CrimeStat* can construct K-order nearest neighbor indices. On the distance analysis page, the user specifies the number of nearest neighbor indices to be calculated.

The K-order nearest neighbor routine returns four columns:

5. The order, starting from 1
6. The mean nearest neighbor distance for each order (in meters)
7. The expected nearest neighbor distance for each order (in meters)
8. The nearest neighbor index for each order

For each order, *CrimeStat* calculates the Kth nearest neighbor distance for each observation and then takes the average. The expected nearest neighbor distance for each order is calculated by:

$$d_{K(ran)} = \frac{K(2K)!}{(2^K K!)^2 \sqrt{\frac{N}{A}}} \qquad (6.8)$$

where K is the order and ! is the factorial operation (e.g., 4! = 4 x 3 x 2 x 1; Thompson, 1956). The Kth nearest neighbor index is the ratio of the observed Kth nearest neighbor distance to the Kth mean random distance. There is not a good significance test for the Kth nearest neighbor

index due to the non-independence of the different orders, though there have been attempts (see examples in Getis and Boots, 1978; Aplin, 1983). Consequently, *CrimeStat* does not provide a test of significance.

There are no restrictions on the number of nearest neighbors that can be calculated. However, since the average distance increases with higher-order nearest neighbors, the potential for bias from edge effects will also increase. It is suggested that not more than 100 nearest neighbors be calculated.[4]

Nevertheless, the K-order nearest neighbor distance and index can be useful for understanding the overall spatial distributions. Figure 6.2 compares the K-order nearest neighbor index for street robberies with that of residential burglaries. The output was saved as a '.dbf' and was then imported into a graphics program. The graph shows the nearest neighbor indices for both robberies and burglaries up to the 50th order (i.e., the 50th nearest neighbor). The nearest neighbor index is scaled from 0 (extreme clustering) up to 1 (extreme dispersion). Since a nearest neighbor index of 1 is expected under randomness, the thin straight line at 1.0 indicates the expected K-order index. As can be seen, both street robberies and residential burglaries are much more concentrated than K-order spatial randomness. Further, robberies are more concentrated than even burglaries for each of the 50 nearest neighbors. Thus, the graph reinforces the analysis above that robberies are more concentrated than burglaries, and both are more concentrated than a random distribution.

In other words, even though there is not a good significance test for the K-order nearest neighbor index, a graph of the K-order indices (or the K-order distances) can give a picture of how clustered the distribution is as well as allow comparisons in clustering between the different types of crimes (or the same crime at two different time periods).

Graphing the K-order Nearest Neighbor

On the output page, there is a quick graph function that displays a curve similar to figure 6.2. This is useful for quickly examining the trends. However, a better graph is made by importing the 'dbf' file output into a spreadsheet or graphics program.

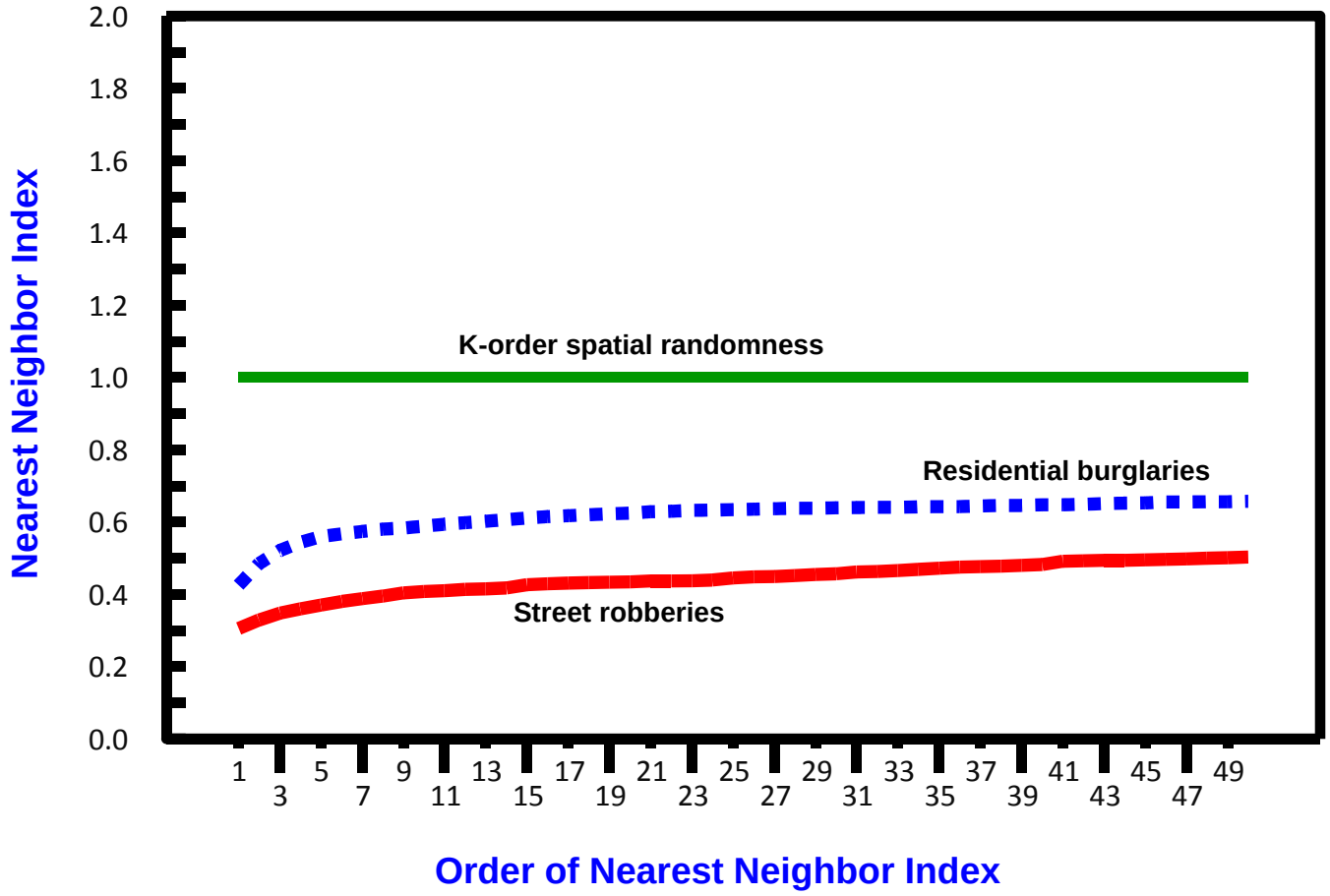
---

[4] There is not a hard-and-fast rule about how many K-order nearest neighbor distances should be calculated. Cressie (1991, p. 613) showed that error increases with increasing order and the degree of divergence from an edge-corrected measure increases over time. In a test case of 584 point locations, he showed that even after only 25 nearest neighbors, the uncorrected measure yields opposite conclusions about clustering from the corrected measures. So, as a rough rule, orders no greater than 2.5% of the cases should be calculated.

Figure 6.2:

K-Order Nearest Neighbor Indices

1996 Street Robberies and Residential Burglaries



Edge Effects

It should be noted that there are potential edge effects that can bias the nearest neighbor index. An incident occurring near the border of the study area may actually have its nearest neighbor on the other side of the border. However, since there are usually no data on the distribution of incidents outside the study area, the program selects another point within the study area as the nearest neighbor of the border point. Thus, there is the potential for exaggerating the nearest neighbor distance, that is, the observed nearest neighbor distance is probably greater than what it should be and, therefore, there is an *overestimation* of the nearest neighbor distance. In other words, the incidents are probably more clustered than what has been measured (see Cressie, 1991 for details). In *CrimeStat*, the K^{th}-order nearest neighbor can be adjusted for boundary (edge) effects.

Nearest Neighbor Edge Corrections

The default condition is no edge correction. However, one way that the measured distance to the nearest neighbor can be corrected for possible edge effects is to assume for each observed point that there is another point just outside the border at the closest distance. If the distance from a point to the border is shorter than to its measured nearest neighbor, then the nearer theoretical point is taken as a proxy for the nearest neighbor. This correction has the effect of reducing the average neighbor distance. Since it assumes that there is always another point at the border, it probably *underestimates* the true nearest neighbor distance. The true value is probably somewhere in between the measured and the assumed nearest neighbor distance.

CrimeStat has two different edge corrections. Because *CrimeStat* is not a GIS package, it cannot locate the actual border of a study area. One would need a topological GIS package in which the distance from each point to the nearest boundary is calculated. Instead, there are two different geometric models that can be applied. The first assumes that the study area is a rectangle while the second assumes that the study area is a circle. Depending on the shape of the actual study area, one or either of these models may be appropriate.

Rectangular study area

In the rectangular adjustment, the area of the study area, A, is first calculated, either from the user input on the measurement parameters tab or from the maximum bounding rectangle defined by the minimum and maximum X/Y values (see chapter 3). If the user provides an estimate of the area, the rectangle is proportionately re-scaled so that the area of the rectangle equals A.

Second, for each point, the distance to the nearest other point is calculated. This is the observed nearest neighbor distance for point i.

Third, the minimum distance to the nearest edge of the rectangle is calculated and is compared to the observed nearest neighbor distance for point i. If the observed nearest neighbor distance for point i is equal to or less than the distance to the nearest border, it is retained. On the other hand, if the observed nearest neighbor distance for point i is greater than the distance to the nearest border, the distance to the border is used as a proxy for the nearest neighbor distance of point i.

Circular study area

In the circular adjustment, first, the area of the study area is calculated, either from the user input on the measurement parameters tab (see chapter 3) or from the maximum bounding rectangle defined by the minimum and maximum X/Y values. If the user has specified a study area on the measurement parameters page, then that value is taken for A and the radius of the circle is calculated by

$$R = \text{SQRT}\,[A / \pi\,] \tag{6.9}$$

If the user has not specified a study area on the measurement parameters page, then A is calculated from the minimum and maximum X and Y coordinates (the bounding rectangle) and the radius of the circle is calculated with equation 6.9.

Second, for each point, the distance to the nearest other point is calculated. This is the observed nearest neighbor distance for point i. Third, for each point, i, the distance from that point to the mean center is calculated, R_i. Fourth, the minimum distance to the nearest edge of the circle is calculated using

$$R_{iC} = R - R_i \tag{6.10}$$

Fifth, for each point, i, the observed minimum distance is compared to the nearest edge of the circle, R_{iC}. If the observed nearest neighbor distance for point i is equal to or less than the distance to the nearest edge, it is retained. On the other hand, if the observed nearest neighbor distance for point i is greater than the distance to the nearest edge, the distance to the border is used as a proxy for the true nearest neighbor distance of point i.

For either correction

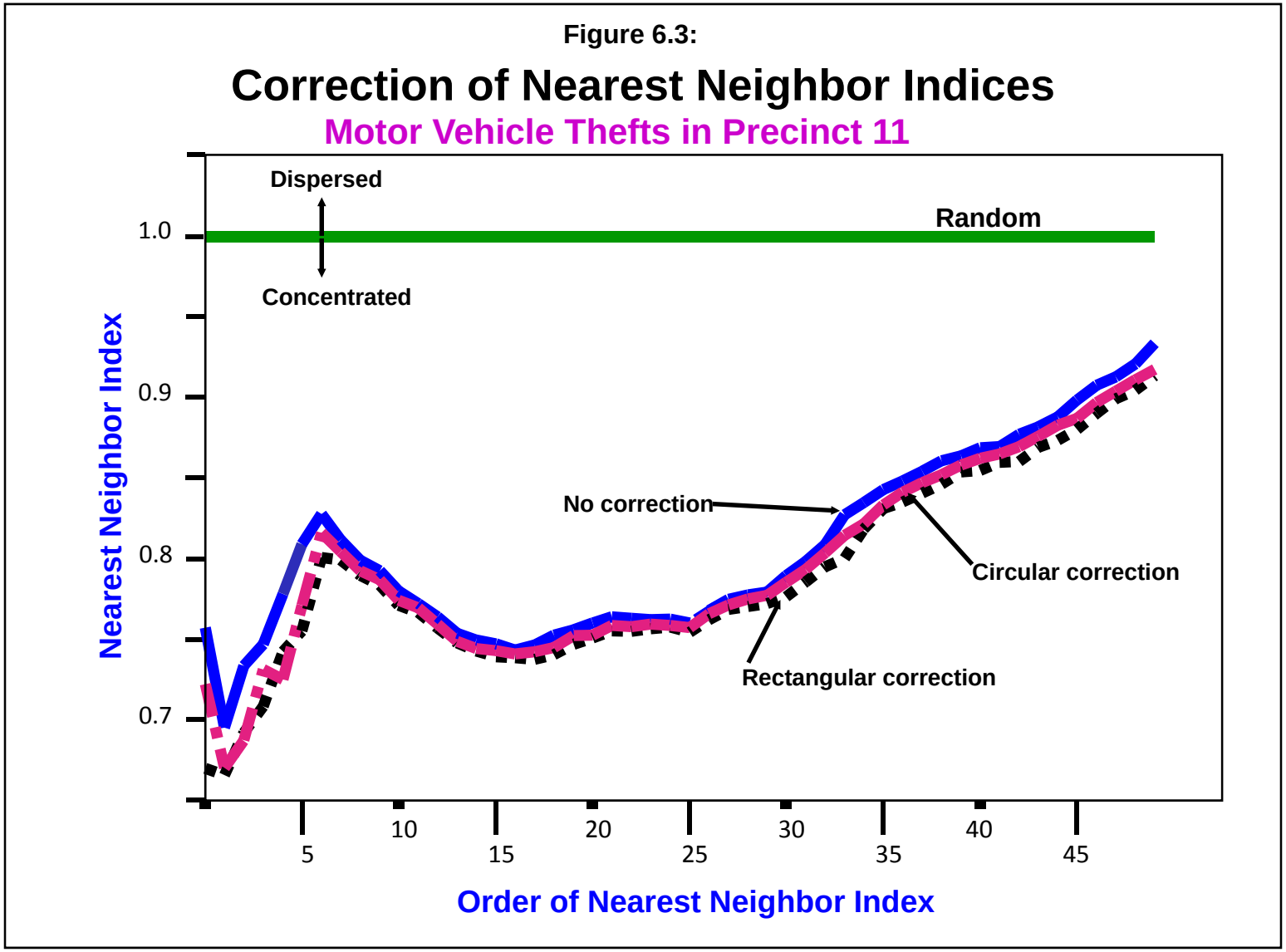
The average nearest neighbor distance is calculated and compared to the theoretical average nearest neighbor distance under random conditions. The indices and tests are as before (see chapter 4). Figure 6.3 below shows a graph of the K-order nearest neighbor index for the 50 nearest neighbors for 1996 motor vehicle thefts in police Precinct 11 of Baltimore County. The uncorrected nearest neighbor indices are compared with those corrected by a rectangle and a circle. As can be seen, both corrections are very similar to the uncorrected. However, they both show greater concentrations than the uncorrected index. The rectangular correction shows greater concentration than the circular because it is less compact (i.e., the average distance from the center of the geometric object to the border is slightly larger). In general, the rectangle will lead to more correction than the circle since it substitutes a greater nearest neighbor distance, on average, for a point nearer the border than to its measured nearest neighbor.

The user has to decide whether either of these corrections is meaningful or not. Depending on the shape of the study area, either correction may or may not be appropriate. If the study area is relatively rectangular, then the rectangular model may provide a good approximation. Similarly, if the study area is compact (circular), then the circular model may provide a good approximation. On the other hand, if the study area is of irregular shape, then either or both of these corrections may produce more distortion than the raw nearest neighbor index. One has to use these corrections with judgment. Also, in some cases, it may not make any sense to correct the measured nearest neighbor distances. In Honolulu, for example, one would not correct the measured nearest neighbor distances because there are no incidents outside the island's boundary.

Linear Nearest Neighbor Index

The *linear nearest neighbor index* is a variation on the nearest neighbor routine, but one applied to a street network. All distances along this network are assumed to travel along a grid, hence indirect distances are used. Whereas the nearest neighbor routine calculates the distance between each point and its nearest neighbor using direct distances, the linear nearest neighbor routine uses indirect ('Manhattan') distances (see chapter 3). Similarly, whereas the nearest neighbor routine calculates the expected distance between neighbors in a random distribution of N points using the geographical area of the study region, the linear nearest neighbor routine uses the total length of the street network.

Figure 6.3:


Correction of Nearest Neighbor Indices
Motor Vehicle Thefts in Precinct 11
Dispersed
Random
Concentrated
1.0
0.9
0.8
0.7
Nearest Neighbor Index
No correction
Circular correction
Rectangular correction
5
10
15
20
25
30
35
40
45
Order of Nearest Neighbor Index

The theory of linear nearest neighbors comes from Hammond and McCullagh (1978). The observed linear nearest neighbor distance, $L_{d(NN)}$, is calculated by *CrimeStat* as the average of indirect distances between each point and its nearest neighbor. The expected linear nearest neighbor distance is given by:

$$L_{d(ran)} = 0.5\frac{L}{N-1} \tag{6.11}$$

where L is the total length of street network and N is the sample size (Hammond and McCullagh, 1978, 279). Consequently, the linear nearest neighbor index is defined as:

$$LNNI = \frac{L_{d(NN)}}{L_{d(ran)}} \tag{6.12}$$

Testing the Significance of the Linear Nearest Neighbor Index

Since the theoretical standard error for the random linear nearest neighbor distance is not known, the author has constructed an approximate standard deviation for the observed linear nearest neighbor distance:

$$s_{Ld(NN)} \cong \sqrt{\frac{\sum_{i=1}^{N}\sum_{j=1}^{N-1}[Min(d_{ij})-Ld(NN)]^2}{N-1}} \tag{6.13}$$

where $Min(d_{ij})$ is the nearest neighbor distance for point i and $L_{d(NN)}$ is the average linear nearest neighbor distance. This is the standard deviation of the linear nearest neighbor distances. The standard error is calculated by:

$$SE_{L_{d(NN)}} = \frac{s_{Ld(NN)}}{\sqrt{N}} \tag{6.14}$$

An approximate significance test can be obtained by:

$$t = \frac{L_{d(NN)}-L_{d(ran)}}{SE_{L_{d(NN)}}} \tag{6.15}$$

where $L_{d(NN)}$ is the average linear nearest neighbor distance, $L_{d(ran)}$ is the expected linear nearest neighbor distance (equation 6.11), and $SE_{L_{d(NN)}}$ is the approximate standard error of the linear nearest neighbor distance (equation 6.14). Since the empirical standard deviation of the linear nearest neighbor is being used instead of a theoretical value, the test is a "*t*" rather than a Z-test.

Calculating the Statistics

On the measurements parameters page, there are two parameters that are input, the geographical area of the study region and the length of street network. At the bottom of the page, the user must select which type of distance measurement to use, direct, indirect or network. If the measurement type is direct or network, then the nearest neighbor routine returns the standard nearest neighbor analysis (sometimes called *areal* nearest neighbor). On the other hand, if the measurement type is indirect, then the routine returns the linear nearest neighbor analysis. To calculate the linear nearest neighbor index, therefore, distance measurement must be specified as ***indirect*** and the length of the street network must be defined.

Once nearest neighbor analysis has been selected, the user clicks on *Compute* to run the routine. The *Lnna* routine outputs 10 statistics:

1. The sample size
2. The mean linear nearest neighbor distance
3. The minimum linear distance between nearest neighbors
4. The maximum linear distance between nearest neighbors
5. The mean linear random distance
6. The linear nearest neighbor index
7. The standard deviation of the linear nearest neighbor distance
8. The standard error of the linear nearest neighbor distance
9. A significance test of the nearest neighbor index (t-test)
10. The p-values associated with a one tail and two tail significance test.

Example 3: Auto Thefts Along Two Baltimore County Highways

The linear nearest neighbor index is useful for analyzing the distribution of crime incidents along particular streets. For example, in Baltimore County, state highway 26 in the western part and state highway 150 in the eastern part have high concentrations of motor vehicle thefts (figure 6.4). In 1996, there were 87 vehicle thefts on highway 26 and 47 on highway 150. A GIS can be used with the linear nearest neighbor index to indicate whether these incidents are greater than what would be expected on the basis of chance.

Table 6.3 presents the data. Using the GIS, we estimate that there are 3,333.54 miles of roadway segments; this number was estimated by adding up the total length of the street network in the GIS. Of all the road segments in Baltimore County, there are 241.04 miles of major arterial roads of which state highway 26 has a total length of 10.42 miles and state highway 150 has a total road length of 7.79 miles.

Figure 6.4:

Vehicle Thefts in Baltimore County: 1996

Incident Distribution on State Highways 26 and 150

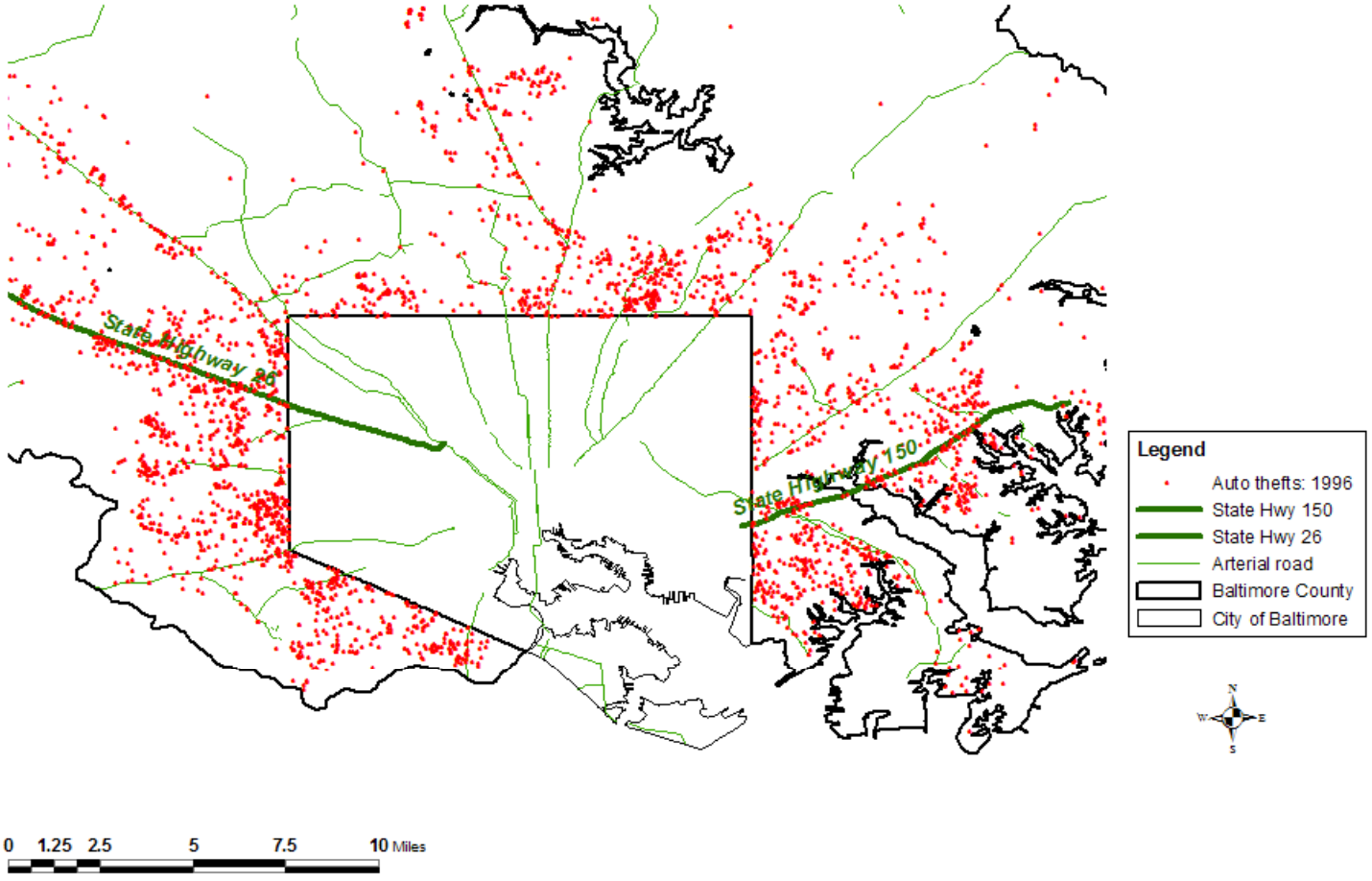


Table 6.3:

Comparison of 1996 Baltimore County Vehicle Thefts for Different Types of Roads

(N = 3774 Incidents)

Length of Road Segments:

Highway 26	10.42 mi
Highway 150	7.79 mi
All Major Arterials	241.04 mi
All Roads	3333.54 mi

Random Expected Distance Between Incidents: 0.44 miles

	Proportional To Network			***Proportional to Same Road***		
Where Incidents Occurred	**Number of Incidents**	**Expected Number *if* Random**	***"Relative to random"*** **Ratio of Frequency**	**Average Linear Nearest Neighbor Distance**	**Average Random Linear Nearest Neighbor Distance**	***"Relative to itself"*** **Linear Nearest Neighbor Index**
Highway 26	87	11.8	7.4	0.05 mi	0.06	0.96
Highway 150	47	8.8	5.3	0.08 mi	0.08	0.94
All major highways	607	272.8	2.2	0.13 mi	0.20	0.64 ($p \leq .001$)
All roads	3,774	3,774.0	1.0	0.09 mi	0.44	0.21 ($p \leq .001$)

The analysis is done proportional to the road network (i.e., all roads) and proportional to the same road. In 1996, there were 3,774 motor vehicle thefts in the county. If these thefts were distributed randomly, then the random expected distance between incidents would be 0.44 miles (equation 6.11). Using this estimate, Table 6.3 shows the number of incidents that would be expected on each of the two state highways if the distribution were random and the ratio of the actual number of motor vehicle thefts to the expected number. As can be seen, the distribution of motor vehicle thefts is not random. On all major highways, there are 2.2 times as many thefts as would be expected by a random spatial distribution.

In fact, in 1996, of 28,551 road segments in Baltimore County, only 7791 (27%) had one or more motor vehicle thefts occur on them; most of these are major roads. Further, on Highway 26 there were 7.4 times as much and on Highway 150 there were 5.3 times as much as would be expected if the distribution was random. Thus, relative to the entire network, these two highways had more than their share of auto thefts in 1996.

But what about the distribution of the incidents *along* each of these highways? If there was a spatial pattern to the incidents, such as clustering on the western edge or in the center, then police could use that information to more efficiently deploy vehicles to respond quickly to events. On the other hand, if the distribution along these highways were no different than a random distribution, then police vehicles must be positioned in the middle, since that would minimize the distance to all occurring incidents.

Unfortunately, the results appear to be close to a random distribution. *CrimeStat* calculated that for Highway 26, the average linear nearest neighbor distance was 0.05 miles which was close to the average random linear nearest neighbor distance (0.06 miles). The ratio - the linear nearest neighbor index, is 0.96 with a t-value of -0.16, which is not significantly different from chance.

Similarly, for Highway 150, the average linear nearest neighbor distance was 0.079 miles which, again, was almost identical to the average random linear nearest neighbor distance (0.084 miles); the nearest neighbor index was 0.94 and the t-value was -0.41 (not significant). In short, even though there was a higher concentration of vehicle thefts on these two state highways than would be expected on the basis of chance, the distribution *along* each highway is not very different than what would be expected on the basis of chance.[5]

---

[5] Because *CrimeStat* uses indirect distance for the linear nearest neighbor index (i.e. measurement only in an horizontal or vertical direction), there is a slight distortion that can occur if the incidents are distributed in a diagonal manner, such as with State Highways 26 and 150 in Figure 6.4. The distortion is very small, however. For example, with the incidents along State Highway 26, after rotating the incident points so that they fell approximately in a horizontal orientation, the observed average linear nearest neighbor distance

On the other hand, the distribution of vehicle thefts along all major highways was not random in 1996 nor was the distribution of vehicle thefts along all roads. For those two high volume highways, however, unfortunately, the distribution of auto thefts was random and the clustering that is evident on all highways and all roads is apparently occurring at other locations. Not every test shows clustering and an analyst should be able to recognize a distribution that is no different than random.

Linear K-Order Nearest Neighbor

In *CrimeStat*, There is also a K-order linear nearest neighbor analysis, as with the areal nearest neighbors. The user can specify how many additional nearest neighbors are to be calculated. The linear K-order nearest neighbor routine returns four columns:

1. The order, starting from 1
2. The mean linear nearest neighbor distance for each order (in meters)
3. The expected linear nearest neighbor distance for each order (in meters)
4. The linear nearest neighbor index for each order

Since the expected linear nearest neighbor distance has not been worked out for orders higher than one, the calculation produced here is a rough approximation. It applies equation 6.11 only adjusting for the decreasing sample size, N_k, which occurs as degrees of freedom are lost for each successive order. In this sense, the index is really the k-order linear nearest neighbor distance relative to the expected linear neighbor distance for the first order. It is not a strict nearest neighbor index for orders above one.

Nevertheless, like the areal k-order nearest neighbor index, the k-order linear nearest neighbor index can provide insights into the distribution of the points, even if the first-order is random. Figure 6.5 shows a graph of 50 linear nearest neighbors for 1996 residential burglaries and street robberies for Baltimore County. As with the areal k-order nearest neighbors (see figure 6.3) both burglaries and robberies show evidence of clustering. For both, the first nearest neighbors are closer together than a random distribution. Similarly, over the 50 orders, street

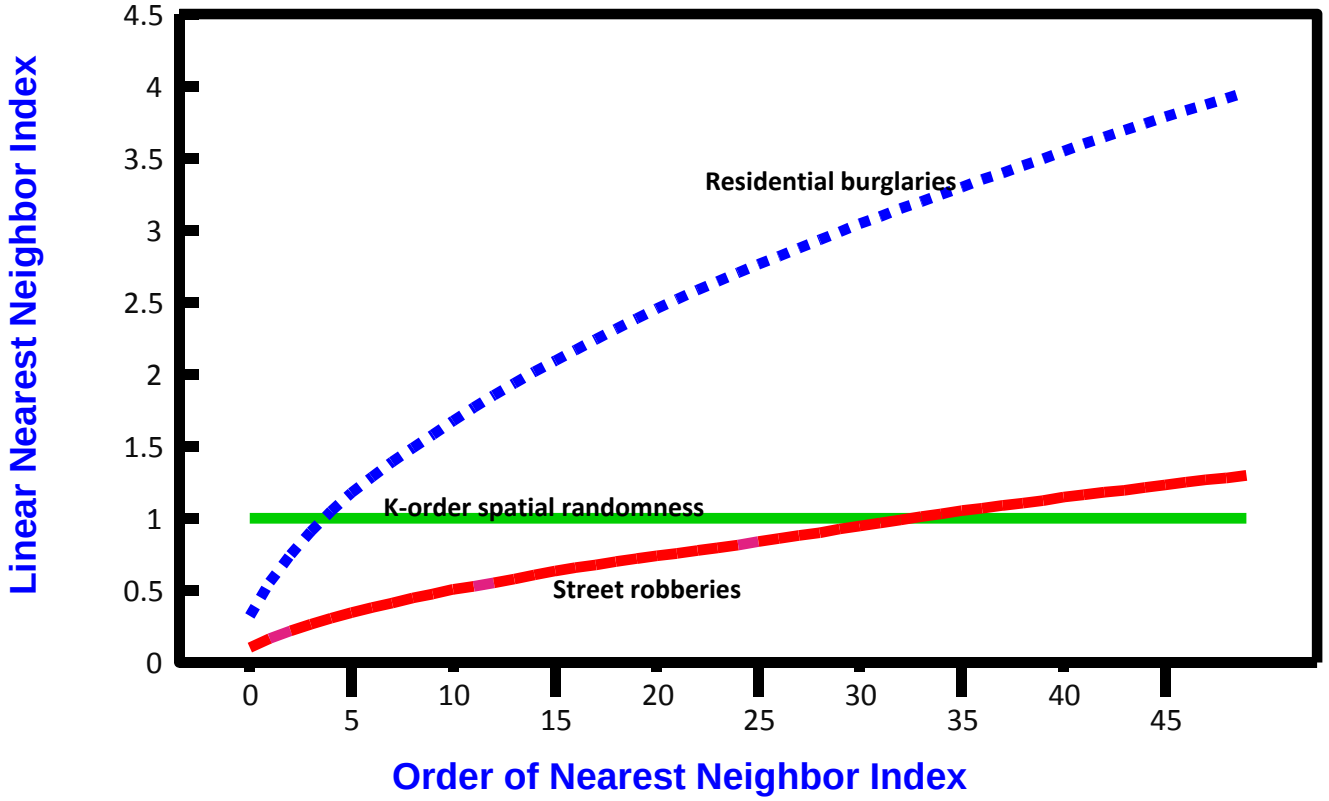
---

decreased slightly from 0.05843 miles to 0.05061 miles and the linear nearest neighbor index became 0.8354 (t=-.91; n.s.). In other words, the effects of the diagonal distribution lengthened the estimate for the average linear nearest neighbor distance by about 41 feet compared to the actual distances between incidents. For a very small sample, this could be a major source of error, but will be negligible for a lare sample. However, if a more precise measure is required, then the user should rotate the distribution so that the incidents have a horizontal or vertical orientation as closely as possible. An alternative is to calculate the regular nearest neighbor distance but use a network for distance calculations (see chapter 3).

Figure 6.5:

K-Order Linear Nearest Neighbor Indices

1996 Street Robberies and Residential Burglaries



robberies are more clustered than burglaries. However, measuring distance on a grid shows that for burglaries, there is only a small amount of clustering. After the fourth order neighbor, the distribution for burglaries is more dispersed than a random distribution. An interpretation of this is that there are small number of burglaries which are clustered, but the clusters are relatively dispersed. Street robberies, on the other hand, are highly clustered, up to over 30 nearest neighbors.

The linear k-order nearest neighbor distribution gives a slightly different perspective on the distribution than the area. For one thing, the index is slightly biased as the denominator - the K-order expected linear neighbor distance, is only approximated. For another thing, the index measures distance *as if* the street follow a true grid, oriented in an east-west and north-south direction. In this sense, it may be unrealistic for many places, especially if streets traverse in diagonal patterns; in these cases, the use of indirect distance measurement will produce greater distances than what actually occur on the network. Still, the linear nearest neighbor index is an attempt to approximate travel along the street network. To the extent that a particular jurisdiction's street pattern falls in this manner, it can provide useful information.

Graphing the Linear K-order Nearest Neighbor

On the output page, there is a quick graph function that displays a curve similar to figure 6.5 below. This is useful for quickly examining the trends.

Ripley's K Statistic

Ripley's K statistic is an index of non-randomness for different scale values (Ripley, 1976; Ripley, 1981; Bailey and Gattrell, 1995; Venables and Ripley, 1997). In this sense, it is a 'super-order' nearest neighbor statistic, providing a test of randomness for every distance from the smallest up to some specified limit. It is sometimes called the *reduced second moment measure*, implying that it is designed to measure second-order trends (i.e., local clustering as opposed to a general pattern over the region). However, it is also subject to first-order effects so that it is not strictly a second-order measure.

Consider a *spatially random* distribution of N points. If circles of radius, t_s, are drawn around each point, where s is the order of radii from the smallest to the largest, and the number of other points that are found within the circles are counted and then summed over all points (allowing for duplication), then the expected number of points under *complete spatial randomness* (csr) within that radius are:

$$E_{Id_i} = \frac{N}{A} K(t_s) = \frac{\pi t_s^2}{A} N \tag{6.16}$$

where N is the sample size, A is the total study area, and $K(t_s)$ is the area of a circle defined by radius, t_s. For example, if the cumulative area defined by a particular radius is one-fourth the total study area and *if* there is a spatially random distribution, on average approximately one-fourth of the cases will fall within one or more circles. Notice that individual points can be counted in multiple circles but the total number of points counted (excluding duplicates) is proportional to the cumulative area of the circle relative to the total area.

On the other hand, if the total number of points found within the circles for a particular radius placed over each point, in turn, is greater than that found in equation 6.16, this points to clustering, that is points are, on average, closer than would be expected on the basis of chance for that radius. Conversely, if the total number of points found within the circles for a particular radius placed over each point is, in turn, less than that found in equation 6.16, then this points to dispersion; that is points are, on average, farther apart than would be expected on the basis of chance for that radius. By counting the total number within a particular radius and comparing it to the number expected on the basis of complete spatial randomness, the statistic is an indicator of non-randomness.

In this sense, the K statistic is similar to the nearest neighbor distance in that it provides information about the average distance between points. However, it is more comprehensive than the nearest neighbor statistic for two reasons. First, it applies to all orders cumulatively, not just a single order. Second, it applies to all distances up to the limit of the study area because the count is conducted over successively increasing radii.

Under unconstrained conditions, K is defined as:

$$K(t_s) = \frac{A}{N^2} \sum_{i=1}^{N} \sum_{i \neq j}^{N-1} I(t_{ij}) \tag{6.17}$$

where $I(t_{ij})$ is the number of other points, j, found within distance, t_s, summed over all points, i. That is, a circle of radius, t_s, is placed over each point, i. Then, the number of other points, j, within the circle is counted. The circle is moved to the next i and the process is repeated. Thus, the double summation points to the count of all j's for each i, over all i's. Note, the count does *not* include itself, only other points.

After this process is completed, the radius of the circle is increased, and the entire process is repeated. Typically, the radii of circles are increased in small increments so that there are 100 intervals by which the statistic can be counted.

One can graph $K(t_s)$ against the distance, t_s, to reveal whether there is any clustering at certain distances or any dispersion at others (if there is clustering at some scales, then there must be dispersion at others). Such a plot is non-linear, however, typically increasing exponentially (Kaluzny, Vega, Cardoso, & Shelly, 1998). Consequently, $K(t_s)$ is transformed into a square root function, $L(t_s)$, to make it more linear. $L(t_s)$ is defined as:

$$L(t_s) = \sqrt{\frac{K(t_s)}{\pi}} - t_s \qquad (6.19)$$

That is, $K(t_s)$ is divided by π and then the square root is taken. Then the distance interval (the particular radius), t_s, is subtracted from this.[6] In practice, only the L statistic is used even though the name of the statistic, *K*, is based on the K derivation.

Because the $L(t_s)$ is a measure of second-order clustering, it is usually analyzed for only a short distance. In *CrimeStat*, the distance is set at one-third the side of a square defined by the area, $\frac{\sqrt{A}}{3}$, and 100 intervals (radii) are used. Figure 6.6 shows a graph of L(t) against distance for 1996 robberies in Baltimore County. As can be seen, L(t) increases up to a distance of about 3 miles whereupon it decreases again. A "pure" random distribution, known as *complete spatial randomness* (CSR), is shown as a horizontal line at L=0.

Comparison to a Spatially Random Distribution

To understand whether an observed K distribution is different from chance, one typically uses a random distribution. Because the sampling distribution of $L(t_s)$ is not known, a simulation can be conducted by randomly assigning points to the study area. Because any one simulation might produce a clustered or dispersed pattern strictly by chance, the simulation is repeated many times, typically 100 or more. Then, for each random simulation, the L statistic is calculated for each distance interval. Finally, after all simulations have been conducted, the highest and lowest L-values are taken for each distance interval. This is called an *envelope*. Thus, by comparing the distribution of L to the random envelope, one can assess whether the particular observed pattern is likely to be different from chance.[7]

---

[6] This form of the $L(t_s)$ is taken from Cressie (1991). In Ripley's original formulation, distance is not subtracted from the square root function (Ripley, 1976). The advantage of the Cressie formulation is that a complete random distribution will be a straight line that is parallel to the X-axis.

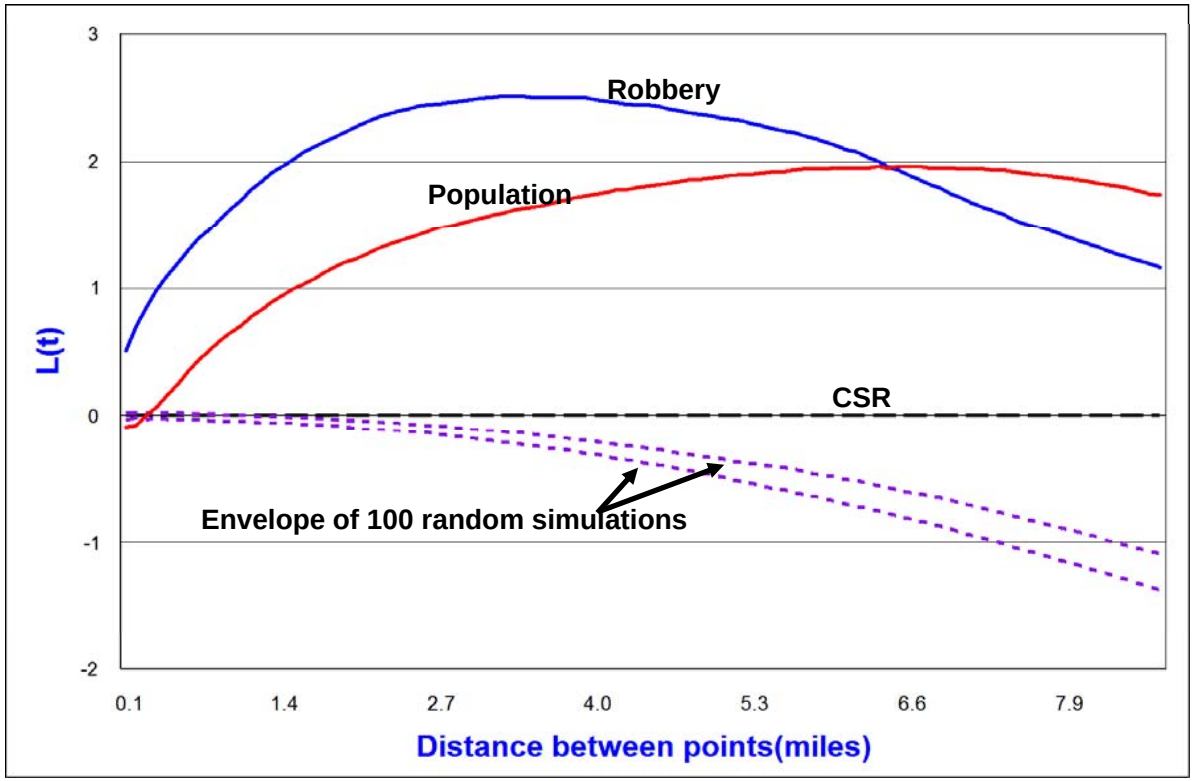
[7] Note that, since there is not a formal test of significance, the comparison with an envelope produced from a number of simulations provides only approximate confidence about whether the distribution differs from chance or not.

Figure 6.6:

"K" Statistic For 1996 Robberies

Compared to Random and 2000 Population Distributions

L(t) = Sqrt[K(t)/pi] - t



In figure 6.6, the L envelope of random data is much less concentrated than that for robberies, indicating that it is highly unlikely the concentration of robberies was due to chance.

Specifying simulations

Because simulations can take a long time, particularly if the data sets are large, the default number of simulations is 0. However, a user can conduct simulations by writing a positive number in the box (e.g., 10, 100, 300). If simulations are selected, *CrimeStat* will conduct the number of simulations specified by the user and will calculate the upper and lower limits for each distance interval, as well as the 0.5th, 2.5th, 5th, 95th, 97.5th and 99th percentile intervals; these latter statistics only make sense if many simulation runs are conducted (e.g. 1000). Approximate 95% credible intervals can be estimated by taking the 2.5th and 97.5th percentiles while approximate 99% credible intervals can be estimated by taking the 0.5th and 99.5th percentiles.[8]

The way *CrimeStat* conducts the simulation is as follows. It takes the maximum bounding rectangle of the distribution, that is the rectangle formed by the maximum and minimum X and Y coordinates respectively and re-scales this (up or down) until the rectangle has an area equal to the study area (defined on the measurement parameters page). It then assigns N points, where N is the same number of points as in the incident distribution, using a uniform random number generator to this rectangle and calculates the L statistic. It then repeats the experiment for the number of specified simulations, and calculates the above statistics. For example, with 1181 robberies for 1996, the Ripley's K function calculates the empirical L statistics for 100 distance intervals and compares this to *M* simulations of 1181 points randomly distributed over a rectangle, where *M* is a user-defined number.

In practice, the simulation test also has biases associated with edges. Unlike the theoretical L under uniform conditions of complete spatial randomness (i.e., stretching in all directions well beyond the study area) where L is a straight horizontal line, the simulated L also declines with increasing distance separation between points. This is a function of the same type of edge bias.

Comparison to Baseline Populations

For most social distributions, such as crime incidents, randomness is not a very

[8] With simulations, statisticians usually refer to their percentiles as *credible* intervals rather than *confidence intervals*, preferring to leave the latter term to formal statistical tests where the mathematical distribution of the standard error is known.

meaningful baseline. Most social characteristics are non-random. Consequently, to find that the amount of clustering that is occurring is greater than what would be expected on the basis of chance is not very useful for crime analysts. However, it is possible to compare the distribution of L for crime incidents with the distribution of L for various baseline characteristics, for example, for the population distribution or the distribution of employment. In almost all metropolitan areas, population is more concentrated towards the center than at the periphery; the drop-off in population density is very sharp as was shown in the last chapter. All other things being equal, one would expect more incidents towards the metropolitan center than at the periphery. Consequently, the average distance between incidents will be shorter in the center than farther out. This is nothing more than a consequence of the distribution of people. However, to say something about concentrations of incidents above-and-beyond that expected by population requires us to examine the pattern of population as well as of crime incidents.

Use of Intensity or Weight Variable

CrimeStat allows the use of intensity and weighting variables in the calculation of the K statistic. The user must define an intensity or a weight variable (or both in special circumstances) on the primary file page. The K routine will then use the intensity (or weight) in the calculation of L. In the current version, if there is an intensity, however, no simulation can be run. The reason is that the sampling distribution of the intensity variable is unknown and it would be difficult to find a candidate distribution from which to draw samples. In a future version, we may allow permutation-type simulations whereby the original intensity values are maintained but they are randomly re-assigned to the existing X/Y coordinates. For now, though, there is no simulation when there is an intensity variable.

In Figure 6.6 above, there is an envelope produced from 100 random simulations as well as the L distribution from the 2000 population; the latter variable was obtained by taking the centroid of traffic analysis zones from the 2000 census and using population as the intensity variable. As can be seen, the amount of clustering for robberies is greater than both the random envelope as well as the distribution of population. The robbery function is higher than the population function up to about 6 miles. This indicates that robberies are more concentrated than what would be expected from the population distribution for a fairly large area.

In other words, robberies are more clustered than even what would be expected on the basis of the population distribution and this holds for distances up to about 6 miles, whereupon the distribution of robberies is indistinguishable from a random distribution. For larger distance separations, the L function has little utility since it is usually used to understand localized spatial autocorrelation (Bailey and Gattrell, 1995).

For comparison, figure 6.7 below shows the distribution of 1996 burglaries, again compared to a random envelope and the distribution of population. Burglaries are more clustered than population, but less so than for robberies; the L value is higher for robberies than for burglaries for near distances but becomes more dispersed at about 3 miles; it is still more concentrated than a random distribution, however, as seen by the random envelope.. Thus, the distribution of L confirms the result that burglaries tend to be spread over a much larger geographical area in smaller clusters than street robberies, which tend to be more concentrated in large clusters. In terms of looking for 'hot spots', one would expect to find more with robberies than with burglaries.

Edge Corrections for Ripley's K

The L statistic is prone to edge effects just like the nearest neighbor statistic. That is, for points located near the boundary of the study area, the number enumerated by any circle for those points will, all other things being equal, necessarily be less than points in the center of the study area because points outside the boundary are not counted. Further, the greater the distance between points that are being tested (i.e., the greater the radius of the circle placed over each point), the greater the bias. Thus, a plot of L against distance will show a declining curve as distance increases as figures 6.6 and 6.7 show.

There are various adjustments to the function to help correct the bias. One is a 'guard rail' within the study area so that points outside the guard rail, but inside the study area can only be counted for points inside the guard rail, but cannot be used for enumerating other points within a circle placed over them (that is, they can only be j's and not i's, to use the language of equation 6.16). Such an operation, however, requires manually constructing these guard rails and enumerating whether each point can be both an enumerator and a recipient or a recipient only. For complex boundaries, such as are found in most police departments, this type of operation is extremely tedious and difficult.[9]

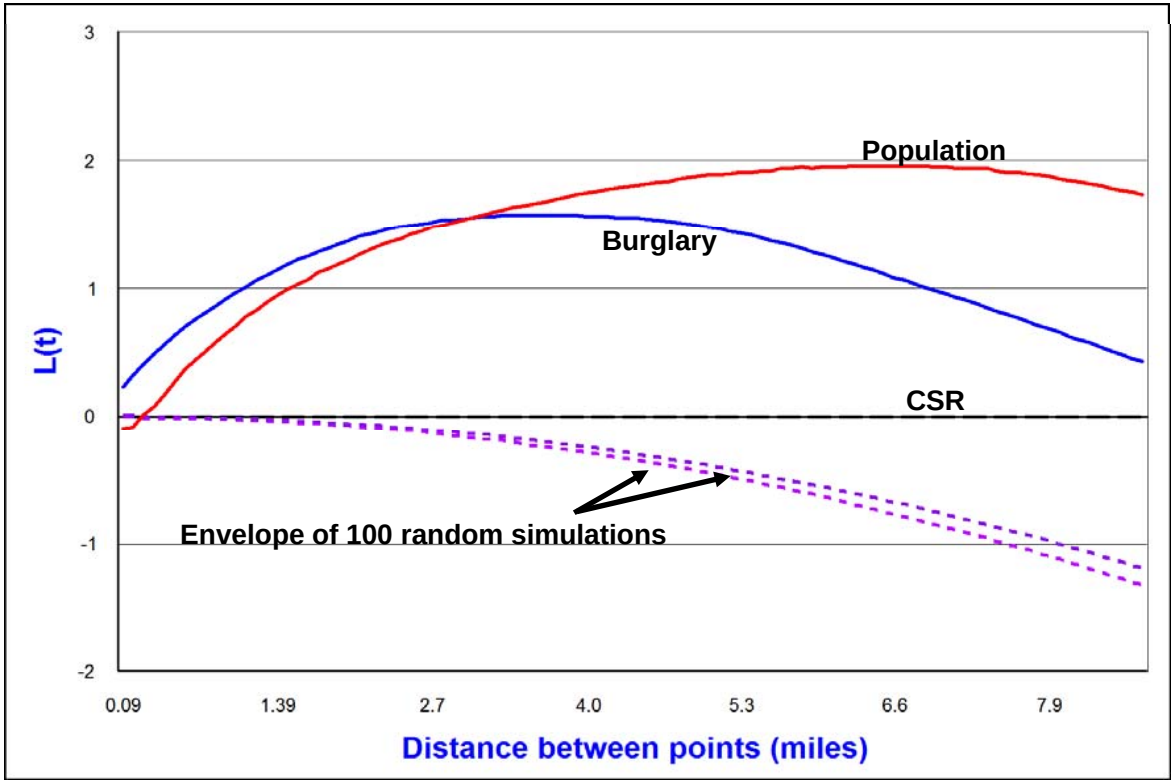
[9] The 'guard rail' concept, while frequently used, is a poor methodology because it involves ignoring data near the boundary of a study area. That is, points within the guard rail are only allowed to be selected by other points and not, in turn, be allowed to select others. This has the effect of throwing out data that could be very important. It is analogous to the old, but fortunately now discarded, practice of throwing out 'outliers' in regression analysis because the outliers were somehow seen as 'not typical'. The guard rail concept is also poor policing practice since incidents occurring near a border may be very important to a police department and may require coordination with an adjacent jurisdiction. In short, use mathematical adjustments for edge corrections or, failing that, leave the data as it is.

Figure 6.7:

"K" Statistic For 1996 Burglaries

Compared to Random and 2000 Population Distributions

L(t) = Sqrt[K(t)/pi] - t



Similarly, Ripley has proposed a simple weighting to account for the proportion of the circle placed over each point that is within the study area (Venables and Ripley, 1997). Thus, equation 6.17 is re-written as:

$$K(t_s) = \frac{A}{N^2} \sum_{i=1}^{N} \sum_{i \neq j}^{N-1} W_{ij}^{-1} I(t_{ij}) \quad (6.20)$$

where W_{ij}^{-1} is the inverse of the proportion of the circumference of a circle of radius, t_s, placed over each point that is within the total study area. Thus, if a point is near the study area border, it will receive a greater weight because a smaller proportion of the circle placed over it will be within the study area. An alternative weighting scheme can be found in Marcon and Puech (2003).

In *CrimeStat*, two possible corrections are conducted. One assumes that the study area is a rectangle while the other assumes that it is a circle.

Rectangular correction

In the rectangular correction for Ripley's K, the search circle radius, R_j, is compared to the edge of an assumed rectangle with area, A, centered at the mean center. First, the area to be analyzed is defined. If the user has specified a study area on the measurement parameters page, then that value for A is taken. The maximum bounding rectangle is taken (i.e., rectangle defined by the minimum and maximum X/Y values) and proportionately re-scaled so that the area of the rectangle is equal to A. If the user does not specify an area on the measurement parameters page, then the bounding rectangle defined by the minimum and maximum X/Y values is taken for A.

Second, for each point, the minimum distance to the nearest edge of this rectangle is calculated in both the horizontal and vertical directions, d_{minRX} and d_{minRY}. Third, each of the minimum distances is compared to the search circle radius, R_j:

1. If neither the minimum distance in the X-direction, d_{minRX}, nor the minimum distance in the Y-direction, d_{minRY}, are less than the search circle radius, R_j, then the circle falls entirely within the rectangle and E = 1;

2. If either the minimum distance in the X-direction, d_{minRX}, or the minimum distance in the Y-direction, d_{minRY}, but NOT BOTH, are less than the search circle radius, R_j, then part of the search circle falls outside the rectangle and an adjustment is necessary. An approximate adjustment is made that is inversely proportional to the area of the search circle within the rectangle. The values of E

will vary between 1 and 2 since up to one-half of the search circle could fall outside the rectangle;

3. If both the minimum distance in the X-direction, d_{minRX}, and the minimum distance in the Y-direction, d_{minRY}, are less than the search circle radius, R_j, then a greater adjustment is required since E could vary between 1 and 4 since up to three-fourth of the search circle could fall outside the rectangle.

The formulas used to calculate the rectangular weights are:

Radius does not extend beyond the rectangle

$$W_{ij}^{-1} = k = 1 \tag{6.21}$$

Radius extends beyond one edge of the rectangle (but not two)

$$W_{ij}^{-1} = k = \frac{2\pi}{2\pi - 2\cos\left\{-1\left[\frac{d(minR)}{R_i}\right]\right\}} \tag{6.22}$$

Radius extends beyond two edges of the rectangle

$$W_{ij}^{-1} = k = \frac{2\pi}{1.5\pi - \cos\left\{-1\left[\frac{d(minR_x}{R_i}\right]\right\} - \cos\left\{-1\frac{[d(minR_y]}{R_i}\right\}} \tag{6.23}$$

While intuitive, this weight, W_{ij}^{-1}, is prone to cause upward 'drift' in the K function, so a log transformation is used:

$$W'_{ij}{}^{-1} = \ln(W_{ij}^{-1}) + 1 \tag{6.24}$$

This has the effect of tempering the drift somewhat.

Circular correction

In the circular correction for Ripley's K, the search circle radius, R_j, is compared to the edge of an assumed circle with area, A, centered at the mean center. First, the area to be analyzed is defined. If the user has specified a study area on the measurement parameters page, then that value for a is taken. The radius of the circle, R_j, is calculated by equation 6.9 above. If

the user has not specified a study area on the measurement parameters page, then A is calculated from the maximum bounding rectangle and the radius of the circle is calculated by equation 6.9 above.

Second, for each point, the distance from that point to the mean center, R_j, is calculated. The nearest distance from the point to the circle's edge is given by

$$R_{jC} = R - R_j \tag{6.25}$$

Third, the search circle radius, R_j, is compared to the nearest edge of the circle, R_{iC}, and the weight will vary from 1 (point and radius totally within the study area) to 2.3834 (point is located exactly on boundary of area circle). The formulas for the circular correction are:

$$\theta = arccos\frac{r^2+t_c-R^2}{2rt_c} \tag{6.26}$$

$$W_{ij}^{-1} = k = \pi / \theta \tag{6.27}$$

where *r* is the radius of the search circle, R is the radius of the circular study area, and t_c is the distance from the point to the center of the circular study area.

For either correction

During the calculation of Ripley's K, each point is multiplied by the weight and the K and L statistics are calculated as before. The simulation of random point distributions is treated in an analogous way. While intuitive, this weight, W_{ij}^{-1}, is prone to cause upward 'drift' in the K function, so a log transformation is used:

$$W'_{ij}^{-1} = \ln(W_{ij}^{-1}) + 1 \tag{6.28}$$

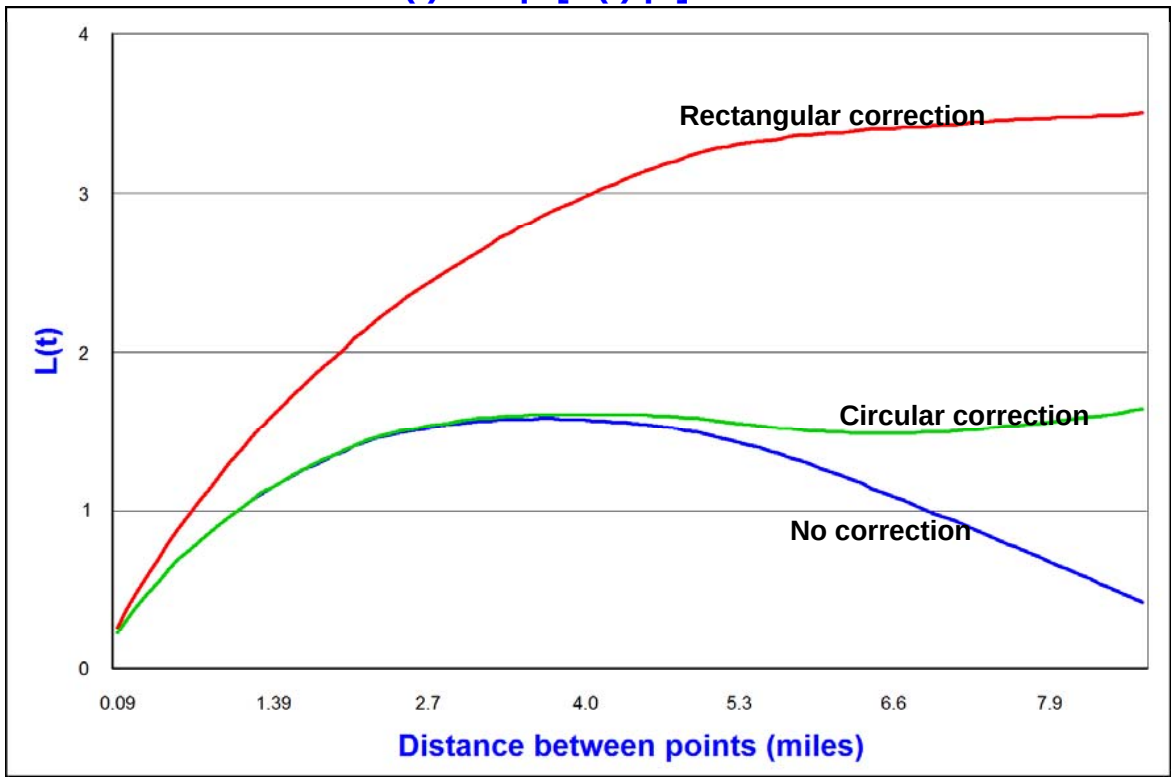
This has the effect of tempering the drift somewhat. Figure 6.8 below shows a Ripley's K distribution for 1996 Baltimore County burglaries, with and without edge corrections. As can be seen, the uncorrected L distribution decreases and falls below the theoretical random count (complete spatial randomness, L=0) after about 7 miles whereas neither the L distribution with the rectangular correction nor the L distribution with the circular distribution do so. As expected, the rectangular distribution produces the most concentration.

Figure 6.8:

"K" Statistic For 1996 Burglaries

With Different Types of Corrections

L(t) = Sqrt[K(t)/pi] - t



Output Intermediate Results

There is a box labeled "Output intermediate results". If checked, a separate dbf file will be output that lists the intermediate calculations. The file will be called "RipleyTempOutput.dbf". There are five output fields:

1. The point number (POINT), starting at 0 (for the first point) and proceeding to N–1 (for the Nth point)
2. The search radius in meters (SEARCHRADI)
3. The count of the number of *other* points that are within the search radius (COUNT)
4. The weight assigned, calculated from equations 6.25 or 6.29 above (WEIGHT)
5. The count times the weight (CTIMESW)

This output can be useful for examining the counts for specific points or for trying out alternative weighting schemes.

Some Cautions in Using Ripley's K

While Ripley's K is a powerful tool for analyzing spatial autocorrelation (usually clustering, rather than dispersion), like any statistic it is prone to biases. Edge biases have been discussed above, but there are others. First, there is a sample size issue. The routine calculates 100 separate L(t) values, one for each distance bin. However, the precision of any one L(t) value is dependent on the sample size. With a small sample, there is insufficient data to estimate 100 independent values of L(t). While the Monte Carlo simulation partly can account for that bias, it has to be realized that the precision of the interpretation is suspect. For example, in comparing two similar distributions, say robberies and burglaries, unless the sample size is large differences for any one bin could easily be due to chance. One would need a very different type of procedure to estimate the 'standard error' of two functions with a small sample. But, I would suspect that there would be many bins for which they would be indistinguishable (shown as the two functions crisscrossing each other).

Users should be very cautious in drawing conclusions about differences in the L function with small samples. Even with sample sizes greater than 100, the imprecision of any one L(t) value is considerable. Until the sample sizes get into the hundreds, precision is an issue for specific L(t) values.

A second caution has to do with the scale of the interpretation. Data sets with strong *first-order* properties (i.e., a high degree of central concentration of incidents) will exert bias on

Ripley's K statistic. Thus, any data set that is correlated with human populations will most likely have a very strong 'central tendency'. Thus, there will be a high degree of concentration in the L values for even near distances. This was seen in the robbery and burglary data shown above. The K statistic was created to estimate *second-order* spatial autocorrelation, namely localized clustering. However, if the first-order effect is so dominant, then it is hard to disentangle it from a second-order effect. In other words, it is often not clear whether the clustering that is observed in Ripley K is due to primary, first-order clustering or actual localized, second-order clustering. That is why it is generally wise to use the K statistic for short distance ranges and not for larger distance separations. For larger distance separations, it is almost impossible to tell whether the effect is due to the large central concentration of the population or whether there are interactions between neighborhoods at a large scale.

There are different ways to handle to problem, none of which are perfect. For example, one can estimate a first-order concentration effect and then apply Ripley's K to the residuals. Alternatively, one can use a baseline population to calculate a rate and test for concentration only in the rates, not the volumes of incidents. In chapters 7 and 9, there will be a discussion of using a baseline population to control for first-order effects. But, whether this is done or not, the user should be aware of the interaction between first-order and second-order (or localized) effects.

The third caution has to do with the shape of the boundaries in interpreting the K statistic. This is particularly true when an edge correction is applied. Unless the study area was an actual rectangle, the correction may alter the interpretation compared to the uncorrected L. There are some subtle differences between the two, however, so some care should be used. The empirical L is obtained from the points within the study area, the geography of which is usually irregular. The random L, however, is calculated from a rectangle or a circle. Thus, the differences in the shape comparisons may account for some variations.

The realism of the corrected function depends on the validity of the underlying assumptions. If it is likely that there are points outside the study area, then a weighting may produce a more realistic interpretation of the L function. On the other hand, if the density of the points outside the study area is lower (e.g., if the study area is a metropolitan area, then the area outside is more likely to be suburban or rural and of low population density), then the weighting will exaggerate the function relative to what it should be. In the extreme case, if the study area is an island (e.g., Honolulu), then there are no points outside the study area and no weighting is justified. Even when weighting would be justified, the actual boundary is probably not a rectangle or a square so that the geometric correction above may distort the L function, too. In short, some understanding of the basis for weighting is necessary to produce a reasonable L function.

Assign Primary Points to Secondary Points

This routine will assign each primary point to a secondary point and then will sum by the number of primary points assigned to each secondary point. The routine is useful for summarizing data. For example, if the primary file represents the number of robberies and the secondary file represents the centroids of census tracts, then the routine will assign all robberies to a census tract and will then sum the number of robberies in each census tract. The result is a count of the number of primary points for each secondary point (zone). Other examples might be to assign students to the nearest school or to assign patients to the nearest hospital. There are many uses for summarizing data by another data reference. In the Trip Generation module (under Crime Travel Demand - see Chapter 27), a model is developed for the number of crimes originating in each zone and a separate model for the number of crimes ending in each zone. The "Assign primary points to secondary points" routine is a good way to summarize the number of crimes by zones.

There are two methods for assigning the primary points to the secondary.

Nearest Neighbor Assignment

This routine assigns each primary point to the secondary point to which it is closest. It goes through all the primary points and sums the number assigned to each secondary point. Thus, the logical operation is 'nearest to'. If there are two or more secondary points that are exactly equal, the assignment goes to the first one on the list.

Point-in-polygon Assignment

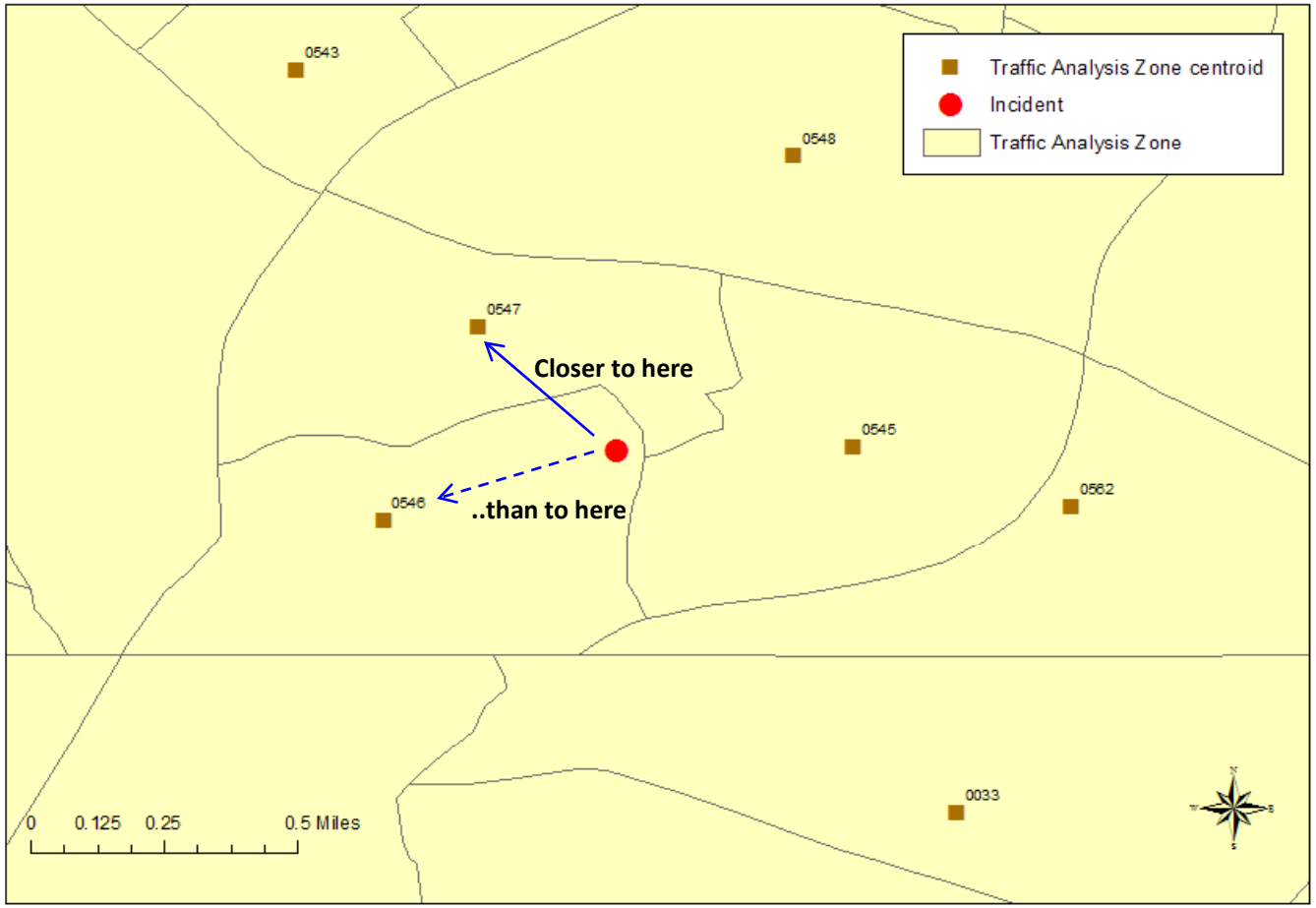
This routine assigns each primary point to the secondary point for which it falls within its polygon (zone). The point-in-polygon assignment reads a zonal boundary file (in ArcGIS 'shp' file format) and determines which zone each primary point falls within. In this case, the logical operation is 'belongs to'. A zone (polygon) shape file must be provided and the routine checks which secondary zone each primary point falls within.

Most GIS packages can do a point-in-polygon operation but few allow a nearest neighbor assignment. In general, the two are similar though there will be differences due to the irregular shape of zone boundaries. For example, figure 6.9 below shows an incident that is within Traffic Analysis Zone (TAZ) 0546, but is actually closer to the centroid of TAZ 0547. The characteristics associated with this incident are more likely to be associated with the characteristics of the second zone than the zone to which it belongs. The decision on which criteria to use in assigning the incident to a zone depends on how integral is the zone to which it

Figure 6.9:

Incident Assignment

Point in Relation to Traffic Analysis Zone Boundaries and Centroids



belongs. If the zones are bounded by major arterials, then travel behavior within the zone will be defined by those arterials; in this case, it would probably be prudent to use the point-in-polygon assignment. On the other hand, if the zone boundaries are not a fundamental separation, then the nearest neighbor assignment would probably produce a better fit to the incident since the characteristics of the closer zone are liable to hold for the incident. In short, the user must decide on which theoretical basis to assign points.

Zone file

If the point-in- polygon method is used, an *ArcGIS* zonal shape file must be defined under the routine. This is a polygon file that defines the zones to which the primary points are assigned. The zonal shape file correspond to secondary file (see Secondary file), but will be the full shape file as opposed to the 'dbf' portion of the file. For each point in the primary file, the routine identifies which polygon (zone) it belongs to and then sums the number of points per polygon.

On the other hand, if the nearest neighbor method is used, then only the secondary file need be defined.

Name of assigned variable

Specify the name of the summed variable. The default name is FREQ.

Use Weighting File

The primary file records can be weighted by another file. This would be useful for correcting the totals from the primary file. For example, if the primary file were robbery incidents from an arrest record, the sum of this variable (i.e. the total number of robberies) may produce a biased distribution over the secondary file zones because the primary file was not a random sample of all incidents (e.g., if it came from an arrest record where the distribution of robbery arrests is not the same as the distribution of all robbery incidents).

The secondary file or another file can be used to adjust the summed total. The weighting variable should have a field that identifies the ratio of the true to the measured count for each zone. A value of 1 indicates that the summed value for a zone is equal to the true value; hence no adjustment is needed. A value greater than 1 indicates that the summed value needs to be adjusted upward to equal the true value. A value less than 1 indicates that the summed value needs to be adjusted downward to equal the true value.

If another file is to be used for weighting, indicate whether it is the secondary file or, if another file, the name of the other file.

Name of assigned weighted variable

For a weighted sum, specify the name of the variable. The default will be ADJFREQ.

Save Result

For both routines, the output is a 'dbf' file. Define the file name. Note: be careful about using the same name as the secondary file as the saved file will have the new variable. It is best to give it a new name.

A new variable will be added to this file that gives the number of primary points in each secondary file zone and, if weighting is used, a secondary variable will be added which has the adjusted frequency.

Example: Assigning Robberies to Zones

To illustrate the routine, table 6.4 shows the results of summarizing 1,181 robberies that occurred in 1997 in 325 Baltimore County Traffic Analysis Zones. The two methods are compared. Only the first 30 assignments are shown. In general, they give similar results. However, there are differences due to the method. One is that the nearest neighbor method will assign points on the basis of proximity while the point-in-polygon method will not. In the case of the Baltimore County robberies, some of these were assigned to a City of Baltimore TAZ because those TAZ's were closer, rather than to a Baltimore County TAZ. Another is that if a zone is very irregular, points may be assigned to it under the point-in-polygon method which may be quite far away.

Thus, the user has to decide which method makes the most sense. If the purpose is to assign incidents to the zone which it is most likely to be related, for example, when developing a data set for zonal modeling (see Chapter 26), then the nearest neighbor method may produce a better representation. The incidents are then assigned to a zone which has characteristics that probably will be related to the factors causing the incidents in the first place. On the other hand, if the object is to assign incidents on the basis of membership (e.g., assigning crimes to police precincts), then the point-in-polygon method will be the most accurate.

Table 6.4:

Assigning Incidents to Zones

1997 Robberies (N=1181) and Traffic Analysis Zones (M=325)

TAZ	Point-in-Polygon	Nearest Neighbor
0401	0	0
0402	0	0
0403	1	1
0404	0	0
0405	0	0
0406	0	0
0407	0	0
0408	0	0
0409	0	0
0410	0	0
0411	0	0
0412	0	0
0413	0	0
0414	1	1
0415	0	0
0416	0	0
0417	0	0
0418	0	0
0419	0	0
0420	0	0
0421	0	0
0422	0	1
0423	0	0
0424	1	0
0425	3	0
0426	2	2
0427	3	2
0428	0	0
0429	5	5
0430	0	0

Distance Analysis II

The remaining distance analysis routines are on the Distance Analysis II page. Figure 6.10 shows the page.

Distance Matrices

CrimeStat has the capability for outputting distance matrices. There are four types of matrices that can be output.

1. First, the distance between every point in the primary file and every other point can be calculated in miles, nautical miles, feet, kilometers or meters. This is called the *Within File Point-to-Point* matrix (Matrix).

2. Second, if there is also a secondary file, *CrimeStat* can calculate the distance from every point in the primary file to every point in the secondary file, again in miles, nautical miles, feet, kilometers or meters. This is called the *From Primary File Points to Secondary File Points* matrix (Imatrix).

3. Third, if there is a reference file defined, the distance from each primary point to each grid cell can be computed. This is called the *From Primary File Points to Grid* matrix (PGMatrix).

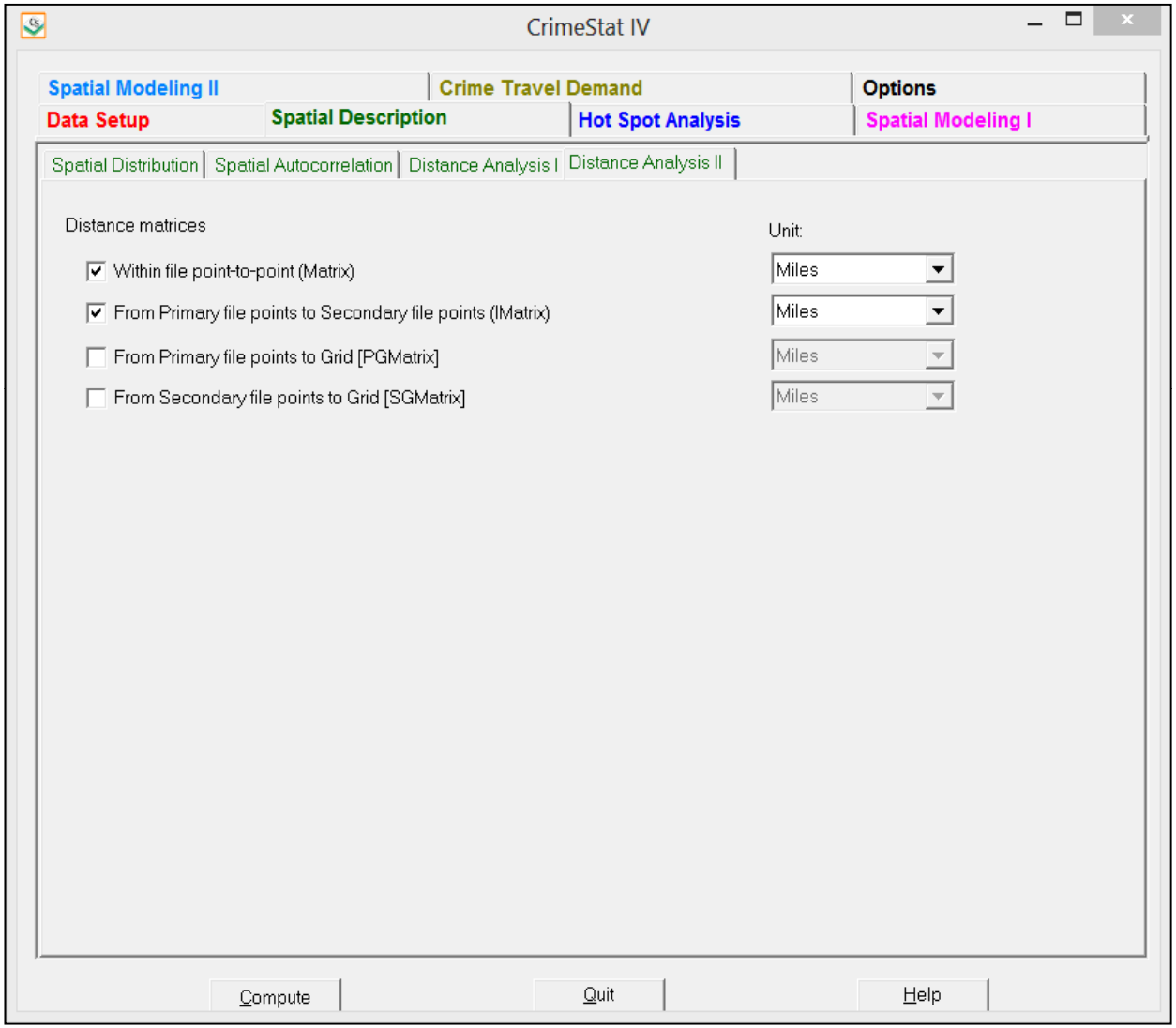
4. Fourth, if there is also a secondary file and a reference file, the distance from each secondary point to each grid cell can be computed. This is called the *From Secondary File Points to Grid* matrix (SGMatrix).

Each of these types of matrices can be displayed or saved to an Ascii text file for import into another program. Each matrix defines incidents by the order in which they occur in the files (i.e., Record number 1 is listed as '1'; record number 2 is listed '2'; and so forth). Only a subset of each matrix is displayed on the results tab. However, there are horizontal and vertical slider bars that allow the user to scroll through the matrix. The user should move the vertical slide bar first to an approximate proportion of the matrix and click the *Go* button. The matrix will scroll through the rows of the matrix to a place which represents that proportion indicated in the slide bar. The user can then scroll across the rows with the upper slide bar.

The matrices can be used for various purposes. The *within file point-to-point matrix* can be used to examine distances between particular incidents. The *saved Ascii '.txt' matrix* can also

Figure 6.10:

Distance Analysis II Screen



be imported into a network program for estimating transportation routes. The *primary-to-secondary file matrix* can be used in optimization routines, for example in trying to assess optimal allocation of police cars in order to minimize response time in a police district. The distances to the grid cells can be used to compare the distances for different distributions to a central location (e.g., a police station). There are many applications where distances are the primary unit of analysis. However, the user will need other software to read the files.

Be careful in outputting distances, though, because the files will generally be very large. For example, a primary file of 1000 incidents when interpolated to 9000 grid cells (100 columns x 90 rows) will produce 9 million paired comparisons. Such a file will take a lot of disk space. For that reason, we only allow output to an Ascii text file.

References

Aplin, G. (1983). *Order-Neighbour Analysis*. Concepts and Techniques in Modern Geography No. 36. Institute of British Geographers, Norwich, England: Geo Books.

Bailey, T. C. & Gatrell, A. C. (1995). *Interactive Spatial Data Analysis*. Longman Scientific & Technical: Burnt Mill, Essex, England.

Clark, P. J. & Evans, F. C. (1954). Distance to nearest neighbor as a measure of spatial relationships in populations. *Ecology*, 35, 445-453.

Cressie, N. (1991). *Statistics for Spatial Data.* New York: J. Wiley & Sons, Inc.

Ebdon, D. (1988). *Statistics in Geography* (second edition with corrections). Blackwell: Oxford.

Getis, A. & Boots, B. (1978). *Models of Spatial Processes: An Approach to the Study of Point, Line and Area Patterns*. London: Cambridge University Press.

Hammond, R. & McCullagh, P. (1978). *Quantitative Techniques in Geography: An Introduction*. Second Edition. Clarendon Press: Oxford, England.

Kaluzny, S. P., Vega, S. C., Cardoso, T. P., & Shelly, A. A. (1998). *S+ Spatial Stats: User Manual for Windows and Unix*. Springer: New York.

Ripley, B. D (1981). *Spatial Statistics*. John Wiley & Sons: New York.

Ripley, B. D. (1976). The second-order analysis of stationary point processes. *Journal of Applied Probability* 13: 255-66.

Thompson, H. R. (1956). Distribution of distance to nth neighbour in a population of randomly distributed individuals. *Ecology*, 37, 391-394.

Venables, W.N. & Ripley, B. D. (1997). *Modern Applied Statistics with S-Plus (second edition)*. Springer-Verlag: New York.

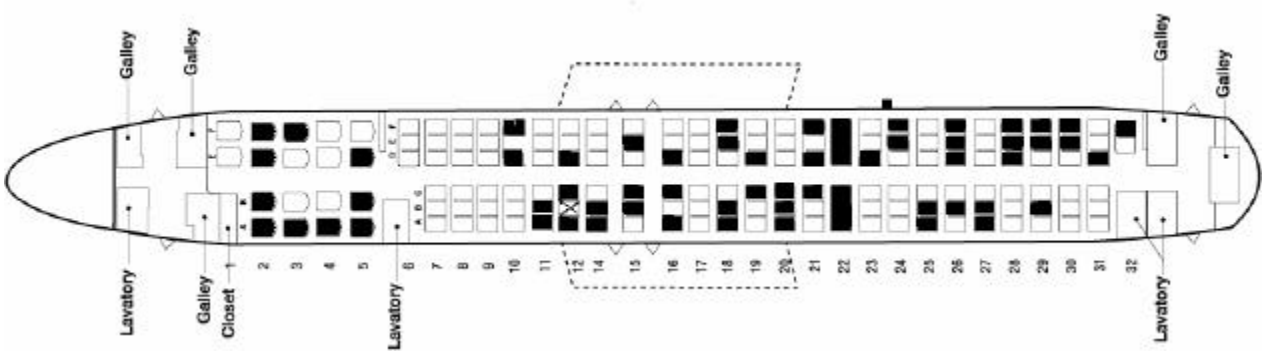
Attachments

SARS and the Distribution of Passengers on an Airplane

Marta A. Guerra
Senior Staff Epidemiologist,
Centers for Disease Control and Prevention
Atlanta, GA

Illness in passengers on board airplanes occurs rather frequently, and investigations are performed to assess whether transmission to other passengers has occurred. During 2002, several passengers with Severe Acute Respiratory Syndrome (SARS) traveled to the United States by airplane while they were infectious. Since transmission of SARS can be airborne, there is concern that it could spread during an airline flight. A survey was undertaken on a flight where a confirmed SARS case was on board. Serum samples of passengers were taken to evaluate if transmission of SARS had occurred during the flight, and whether transmission is related to sitting near the SARS case.

The nearest neighbor index was used to compare the distances between the seats of passengers on this flight to distances expected on the basis of chance. A grid (7 m x 32 m) was superimposed on the airline seat configuration, and each seat was assigned an X, Y coordinate based on the width (x) and the length (y) of the airplane. In the diagram below, the seat location of the SARS index case is indicated by an X, and the passengers' seat locations are shaded in black.



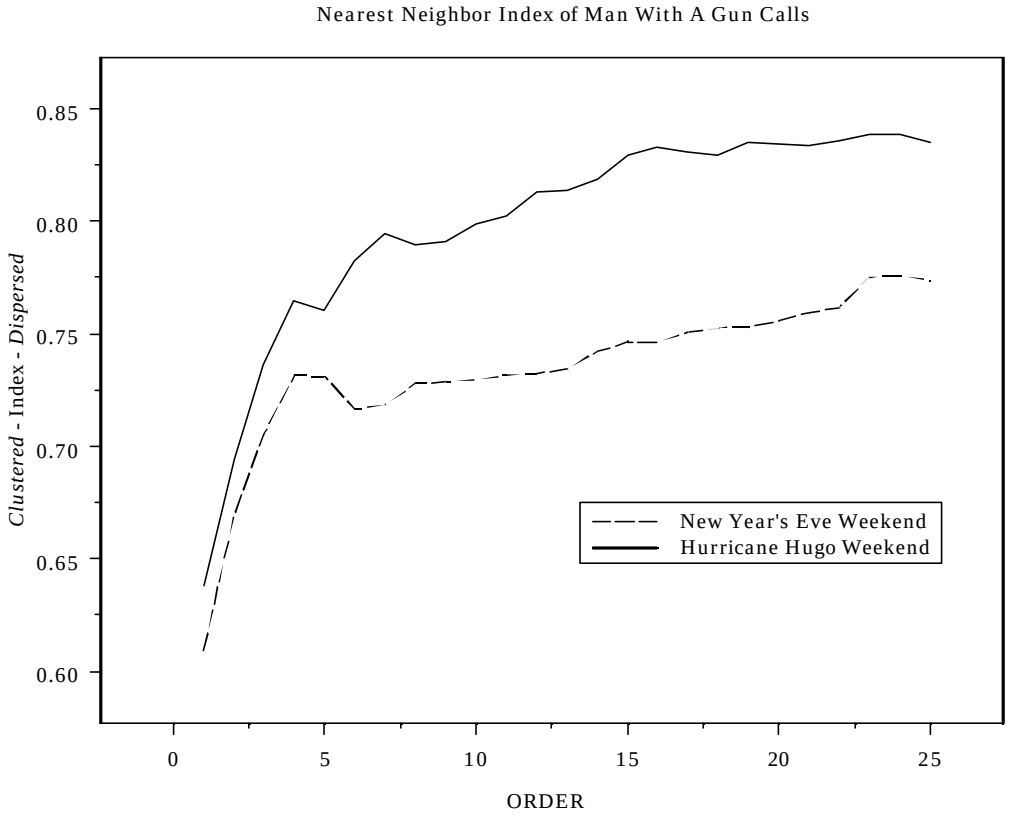
Nearest Neighbor Statistics for Airline Flight with SARS Case

The nearest neighbor index of passengers' seats was 0.931 indicating that the distribution was random, not clustered. This preliminary analysis was important in order to establish that the seating arrangement of the passengers was random and independent, and that the passengers' seats were not clustered around the SARS case. Therefore, if any passengers have positive serum samples for SARS, we would be able to evaluate their locations in relation to the SARS case and assess patterns of transmission. In this survey, however, there was no evidence of transmission since none of the passengers had positive serum samples for SARS.

Nearest Neighbor Analysis
Man With A Gun Calls
Charlotte, N.C.: 1989

James L. LeBeau
Administration of Justice
Southern Illinois University-Carbondale

A comparison was made of *Man with a Gun* calls for the weekend in which Hurricane Hugo hit the North Carolina coast (September 22 – 24) with the following New Year's Eve weekend (December 29-31, 1989). There were 146 *Man with a Gun* calls during the Hurricane Hugo weekend compared to 137 calls for New Year's Eve.



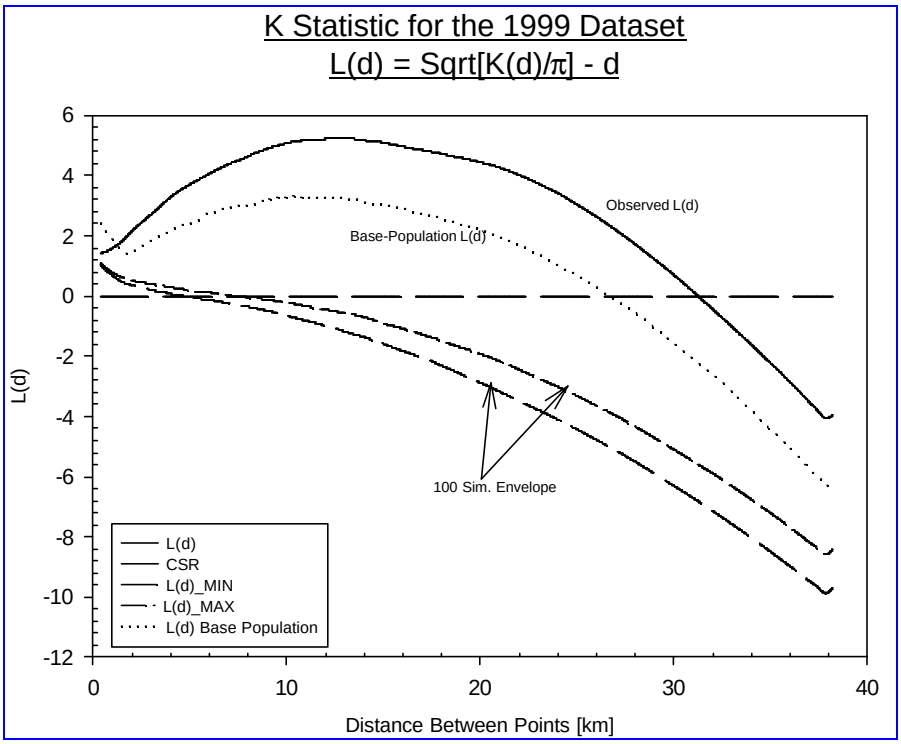
The Nearest Neighbor Index in *CrimeStat* was used to compare the distributions. From the onset, the Hurricane Hugo *Man With a Gun* locations are more dispersed than New Year's Eve. After the 5th nearest neighbor (Order 5) the differences become more pronounced

K-Function Analysis to Determine Clustering in the *Police Confrontations* Dataset in Buenos Aires Province, Argentina: 1999

Gastón Pezzuchi, Crime Analyst
Buenos Aires Province Police Force
Buenos Aires, Argentina

Sometimes crime analysts tend to produce beautiful hot spot maps without any formal evidence that clustering is indeed present in the data. One excellent and powerful tool that *CrimeStat* provides is the computation of the K function, which summarizes spatial dependence over a wide range of scales, and uses the information of all events.

We computed the K function using 1999 police confrontations data (mostly shootings) within our study area[1] and ran 100 Monte Carlo simulations in order to test for spatial randomness[2] (see figure below); the K function showed clustering up to about 30 Km. Yet, spatial randomness is not a particularly meaningful hypothesis to test considering that the "population at risk" are highly clustered. Hence we used police deployment data as a base population and calculated the K function for that data set. As can seen, the amount of clustering for the confrontation dataset is much greater than both the random envelope as well as the distribution of police officers.



[1] A years worth dataset of events occurring within a 9,500 km2 area around the Federal Capital (29 counties).

[2] Remember that Pr(L(d) > Lmax) = Pr(L(d) < Lmin) = 1 / (m + 1) where m is the number of independent simulations,

CrimeStat IV

Part III: Hot Spot Analysis

Chapter 7:

Hot Spot Analysis of Points: I

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Chapter 7:
Hot Spot Analysis of Points: I

In this and the next two chapters, we describe ten tools for identifying clusters of crime incidents. Six of the tools apply to points while four apply to zones. The discussion has been divided into three chapters primarily because of the length of the discussion. This chapter discusses the concept of a *hot spot* and four hot spot techniques: the mode, fuzzy mode, nearest neighbor hierarchical clustering, and risk-adjusted nearest neighbor hierarchical clustering. Chapter 8 discusses STAC and the K-means algorithm. Chapter 9 discusses Anselin's Local Moran, the Getis-Ord Local "G", the zonal nearest neighbor hierarchical clustering algorithm, and the risk-adjusted zonal nearest neighbor hierarchical methods. However, the ten techniques should be seen as a continuum of approaches towards identifying hot spots.

Hot Spots

Typically called *hot spots* or *hot spot areas*, these are concentrations of incidents within a limited geographical area that appear over time (Braga & Weisburd, 2010). Police have learned from experience that there are particular environments that attract crimes in larger-than-expected concentrations, so-called *crime generators*. Sometimes these hot spot areas are defined by particular activities (e.g., drug trading; Weisburd & Green, 1995; Weisburd, Maher, & Sherman, 1992; Sherman, Gartin & Buerger, 1989; Maltz, Gordon, & Friedman, 1989), other times by specific concentrations of land uses (e.g., skid row areas, bars, adult bookshops, itinerant hotels), and sometimes by interactions between activities and land uses, such as thefts at transit stations or bus stops (Block & Block, 1995; Levine, Wachs & Shirazi, 1986). Whatever the reasons for the concentration, they are real and are known by most police departments.

While there are some theoretical concerns about what links disparate crime incidents together into a cluster, nonetheless, the concept is very useful (Chainey, Thompson, & Uhlig, 2008; Levine, 2008). Police officers patrolling a precinct can focus their attention on particular environments because they know that crime incidents will continually reappear in these places. Crime prevention units can target their efforts knowing that they will achieve a positive effect in reducing crime with limited resources (Sherman & Weisburd, 1995). In short, the concept is very useful.

Nevertheless, the concept is a perceptual construct. Hot spots do not exist in reality, but are areas where there is sufficient clustering of certain activities (in this case, crime) such that they get labeled such. There is not a border around these incidents, but a gradient where people draw an imaginary line to indicate the location at which the hot spot *starts*. In reality, any

variable that is measured, such as the density of crime incidents, will be continuous over an area, being higher in some parts and lower in others. Where a line is drawn in order to define a hot spot is somewhat arbitrary.

Statistical Approaches to the Measurement of Hot Spots

Unfortunately, measuring a hot spot is also a complicated problem. There are literally dozens of different statistical techniques designed to identify hot spots (Everitt, Landau and Leese, 2001). Many, but not all, of the techniques are typically known under the general statistical label of *cluster analysis*. These are statistical techniques aimed at grouping cases together into relatively coherent clusters. All of the techniques depend on optimizing various statistical criteria, but the techniques differ among themselves in their methodology as well as in the criteria used for identification. Because hot spots are perceptual constructs, any technique that is used must approximate how someone would perceive an area. The techniques do this through various mathematical criteria.

Types of Cluster Analysis (Hot Spot) Methods

Several typologies of cluster analysis have been developed as cluster routines typically fall into several general categories (Everitt, 2011; Can and Megbolugbe, 1996):

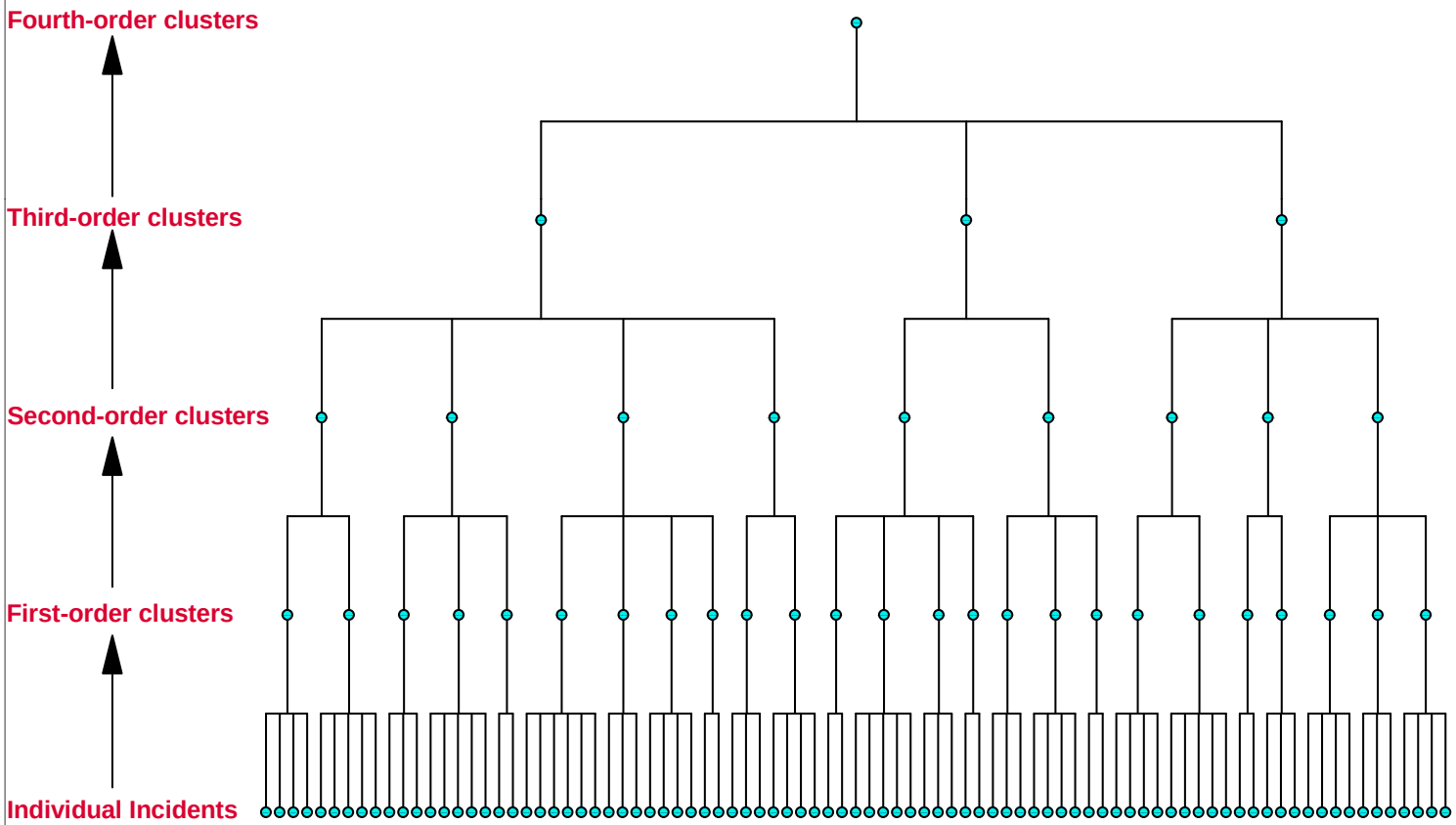
1. *Point locations*. This is the most intuitive type of cluster involving the number of incidents occurring at different locations. Locations with the most number of incidents are defined as hot spots. *CrimeStat* includes two point location techniques: the Mode and Fuzzy Mode;

2. *Hierarchica*l techniques (Sneath, 1957; McQuitty, 1960; Sokal & Sneath, 1963; King, 1967; Sokal & Michener, 1958; Ward, 1963; Hartigan, 1975) are like an inverted tree diagram in which two or more incidents are first grouped on the basis of some criteria (e.g., nearest neighbor). Then, the pairs are grouped into second-order clusters. The second-order clusters are then grouped into third-order clusters, and this process is repeated until either all incidents fall into a single cluster or else the grouping criteria fail. Thus, there is a hierarchy of clusters that can be displayed with a dendogram (an inverted tree diagram).

   Figure 7.1 shows an example of a hierarchical clustering where there are four orders (levels) of clustering; the visualization is non-spatial in order to show the linkages. In this example, all individual incidents are grouped into first-order

Figure 7.1:

Hierarchical Clustering Technique



clusters that, in turn, are grouped into second-order clusters that, in turn, are grouped into third-order clusters and which all converge into a single fourth-order cluster. Many hierarchical techniques, however, do not group all incidents or all clusters into the next highest level. *CrimeStat* includes four hierarchical techniques: Nearest Neighbor Hierarchical Clustering (Nnh) routine and Risk-adjusted Nearest Neighbor Hierarchical Clustering (Rnnh) routines in this chapter and Zonal Nearest Neighbor Hiearchical Clustering (Znnh) and Risk-adjusted Nearest Neighbor Zonal Hierarchical Clustering (RZnnh) routines in Chapter 9.

3. *Partitioning* techniques, frequently called the K-means technique, partition the incidents into a specified number of groupings, usually defined by the user (Thorndike, 1953; MacQueen, 1967; Ball and Hall, 1970; Beale, 1969). Thus, all points are assigned to one, and only one, group. Figure 7.2 shows a partitioning technique where all points are assigned to clusters and are displayed as ellipses. *CrimeStat* includes one partitioning technique, a K-means partitioning technique;

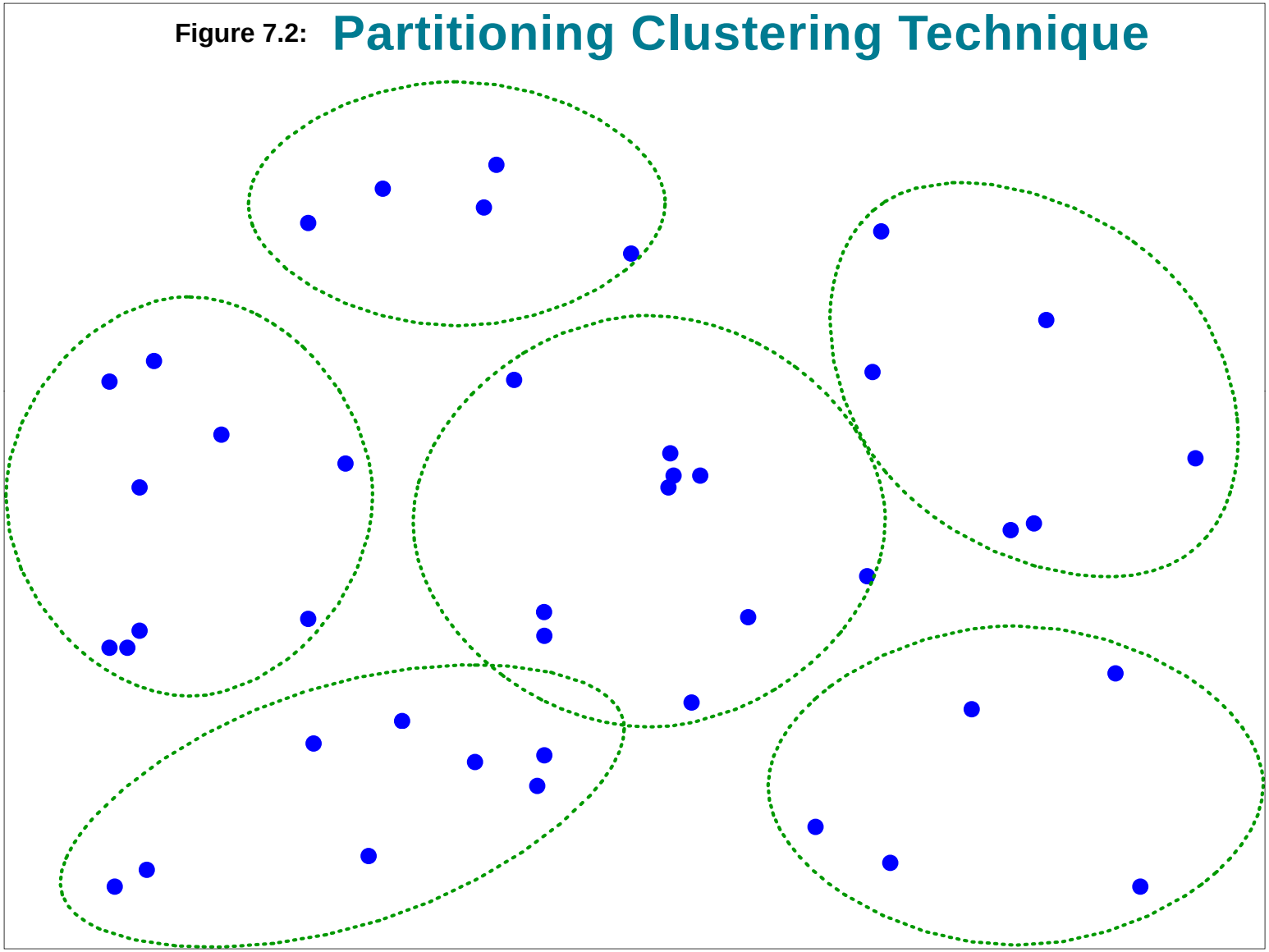
4. *Scan statistics* that apply a search circle uniformly throughout the study area, either to each point or to each node of a reference grid (Block & Block, 1999; Kulldorff, 1997; Block & Block, 1995; Block, 1994; Openshaw, Craft, Charlton, & Birch, 1988; Openshaw, Charlton, Wymer, & Craft, 1987.

5. *Density* techniques identify clusters by searching for dense concentrations of incidents (Bailey & Gattrell, 1995; Silverman, 1986; Gitman & Levine, 1970; Weishart, 1969; Carmichael, George, & Julius, 1968; Cattell & Coulter, 1966). CrimeStat has two density search routines: a Single-kernel Density (K) method and a Dual-kernel Density Interpolation (Dk); this is discussed in chapter 10;

6. *Clumping* techniques involve the partitioning of incidents into groups or clusters, but allow overlapping membership (Jones & Jackson, 1967; Needham, 1967; Jardine & Sibson, 1968; Cole & Wishart, 1970);

7. *Risk-based* techniques identify clusters in relation to an underlying base 'at risk' variable, such as population, employment, or active targets (Jefferis, 1998; Kulldorff and Nagarwalla, 1995). *CrimeStat* includes three risk-based techniques - a Risk-adjusted Nearest Neighbor Hierarchical Clustering routine, discussed in this chapter; a Zonal Risk-adjusted Nearest Neighbor Hierarchical Clustering routines discussed in Chapter 9; and a Dual Kernel Density method, discussed in Chapter 10).

Figure 7.2: Partitioning Clustering Technique



8. *Zonal* clustering techniques identify contiguous zones that have either high levels or similar levels of an attribute variable or (Getis & Ord, 1996; Anselin, 1995). *CrimeStat* includes four zonal clustering methods: Anselin's Local Moran; the Getis-Ord Local "G"; Zonal Nearest Neighbor Hierarchical Clustering; and Zonal Risk-adjusted Nearest Neighbor Hierarchical Clustering.

9. *Miscellaneous* techniques are other methods that are less commonly used (Everitt, 2011).

Many of these methods are hybrids of these classes. For example, the *Risk-adjusted Nearest Neighbor Hierarchical Clustering* routine is primarily a risk-based technique but involves elements of clumping while *STAC* is primarily a partitioning method but with elements of hierarchical grouping.

Optimization Criteria

In addition to the different types of cluster analysis, there are different criteria that distinguish techniques applied to space (Everitt, 2011). Among these are:

1. The *definition* of a cluster - whether it is a discrete grouping or a continuous variable; whether points must belong to a cluster or whether they can be isolated; whether points can belong to multiple clusters.

2. The *choice of variables* in addition to the X and Y coordinates - whether weighting or intensity values are used to define similarities.

3. The measurement of *similarity and distance* - the type of geometry being used; whether clusters are defined by closeness or not; the types of similarity measures used.

4. The *number* of clusters - whether there are a fixed or variable number of clusters; whether users can define the number or not.

5. The geographical *scale* of the clusters - whether clusters are defined by small or larger areas; for hierarchical techniques, what level of abstraction is considered optimal.

6. The *initial selection* of cluster locations ('seeds') - whether they are mathematically or user defined; the specific rules used to define the initial seeds.

7. The *optimization routines* used to adjust the initial seeds into final locations - whether distance is being minimized or maximized; the specific algorithms used to readjust seed locations.

8. The *visual display* of the clusters, once extracted - whether drawn by hand or by a geometrical object (e.g., an ellipse, a convex hull); the proportion of cases represented in the visualization.

This is not the place to provide a comprehensive review of cluster techniques (see Everitt, 2011 for such a review). Nevertheless, it should be clear that with the several types of cluster analysis and with the many criteria that can be used for any particular technique provides a large number of different techniques that could be applied to an incident data base. It should be realized that there is not a single solution to the identification of hot spots, but that different techniques will reveal different groupings and patterns among the groups. A user must be aware of this variability and must choose techniques that can complement other types of analysis. It would be very naive to expect that a single technique can reveal the existence of hot spots in a jurisdiction that are unequivocally clear. In most cases, analysts are not sure why there are hot spots in the first place. Until that is solved, it would be unreasonable to expect a mathematical or statistical routine to solve that problem.

Cluster Routines in *CrimeStat*

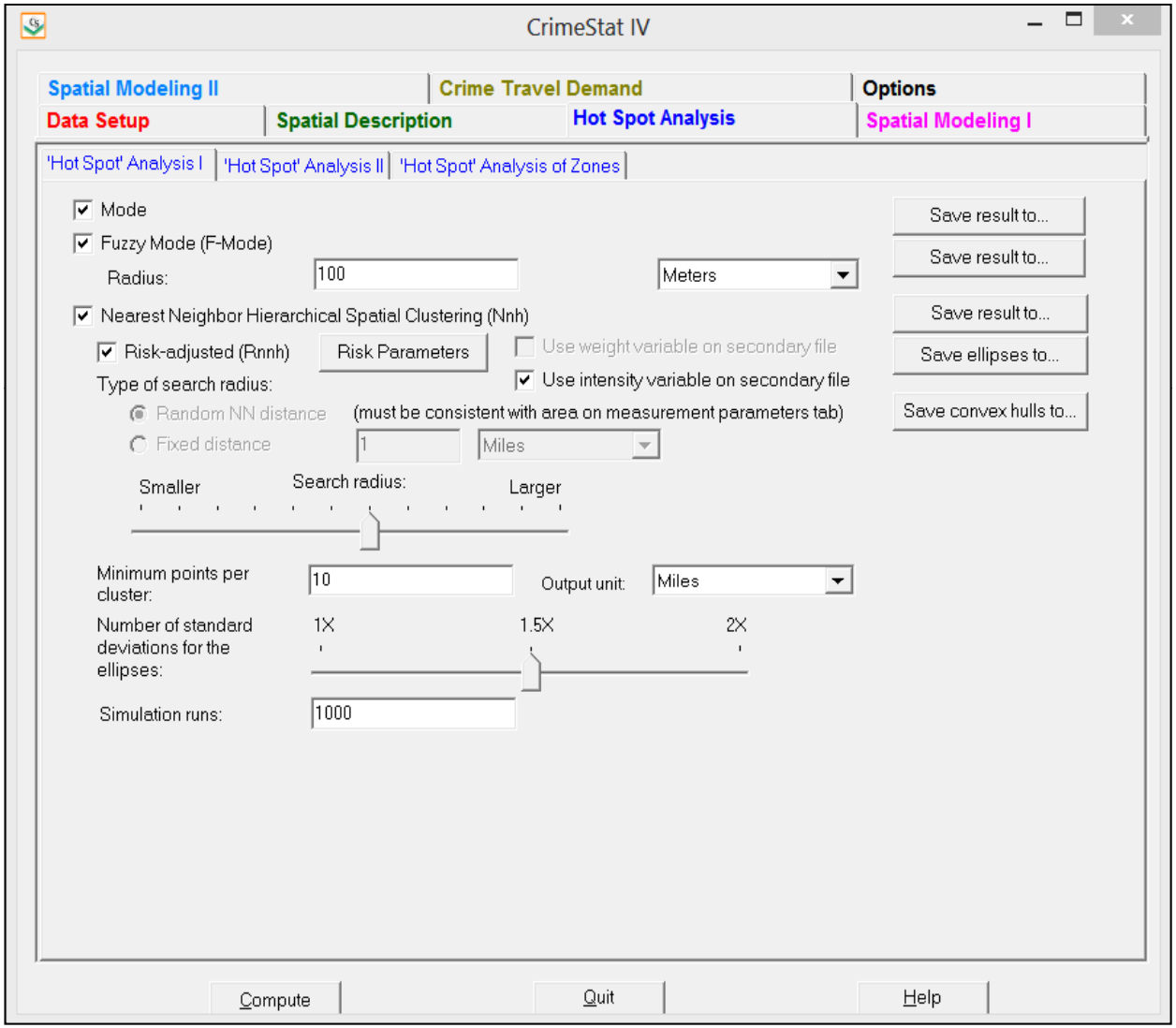
Figure 7.3 shows the Hot Spot Analysis I page. Because of the variety of cluster techniques, *CrimeStat* includes ten techniques that cover the range of techniques that have been used:

1. The Mode
2. The Fuzzy Mode
3. Nearest neighbor hierarchical clustering
4. Risk-adjusted nearest neighbor hierarchical clustering
5. The Spatial and Temporal Analysis of Crime (*STAC*) module
6. K-means clustering
7. Anselin's Local Moran
8. Getis-Ord Local "G"
9. Zonal nearest neighbor hierarchical clustering
10. Zonal risk-adjusted nearest neighbor hierarchical clustering

These are not the only techniques, of course, and analysts should use them as complements to other types of analysis. Because of the number of routines, these routines have

Figure 7.3:

Hot Spot Analysis I Screen



been allocated to two different setup tabs in *CrimeStat* called Hot Spot' Analysis I and Hot Spot Analysis II. However, they should be seen as one collection of similar techniques. This chapter will discuss the first four of these and the next two chapters the remaining ones.

Mode

The *mode* is the most intuitive type of hot spot. It is the location with the largest number of incidents. The *CrimeStat* Mode routine calculates the frequency of incidents occurring at each unique location (a point with a unique X and Y coordinate), sorts the list, and outputs the results in rank order from the most frequent to the least frequent.

Only locations that are represented in the primary file are identified. The routine outputs a 'dbf' file that includes four variables:

1. The rank order of the location with 1 being the location with the most incidents, 2 being the location with the next most incidents, 3 being the location with the third most incidents, and so forth until those locations that have only one incident each;

2. The frequency of incidents at the location. This is the number of incidents occurring at that location;

3. The X coordinate of the location; and

4. The Y coordinate of the location.

To illustrate, Table 7.1 presents the formatted output for the ten most frequent locations for 14,853 motor vehicle thefts that occurred within the City of Baltimore or Baltimore County in 1996.[1] Figure 7.4 maps the ten locations with the most vehicle thefts (two were tied for rank three and two were tied for rank nine). The map displays the locations with a round symbol, the size of which is proportional the number of incidents. Also, the number of incidents at the location is displayed. These vary from a high of 43 vehicle thefts at location number 1 to a low of 15 vehicle thefts at location numbers 9 and 10. In order to know what these locations represent, the user will have to overlay other GIS layers over the points. In the example, of the ten locations, eight are at shopping centers, one is the parking lot of a train station, and one is the parking lot of a large organization.

1 The output in Table 7.1 has been formatted. *CrimeStat* only outputs an Ascii file.

Figure 7.4:

Metropolitan Baltimore Vehicle Thefts: 1996

10 Most Frequent Vehicle Theft Locations

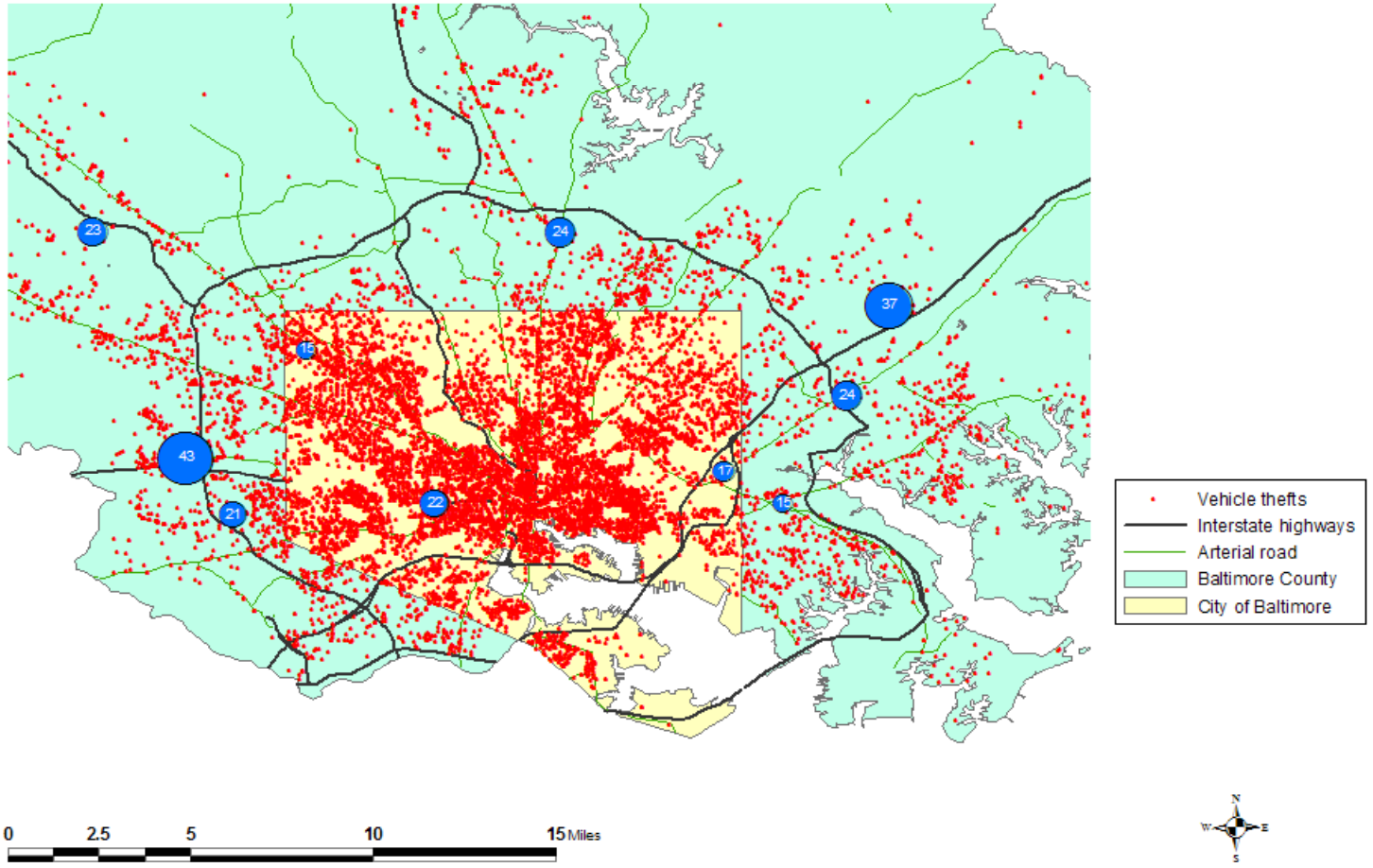


Table 7.1:

Mode Output for
Most Frequent Locations for Motor Vehicle Thefts

City of Baltimore and Baltimore County: 1990

(ONLY 10 SHOWN)

Mode:

N = 14,853

Rank	Freq	X	Y
1	43	-76.7507	39.3115
2	37	-76.4710	39.3741
3	24	-76.4880	39.3372
4	24	-76.6015	39.4042
5	23	-76.7877	39.4046
6	22	-76.6517	39.2927
7	21	-76.7319	39.2880
8	17	-76.5363	39.3060
9	15	-76.7026	39.3560
10	15	-76.5128	39.2927
Etc.			

The mode is a very simple measure, but one that can be very useful. In the example, it is clear that most vehicle thefts occur at institutional settings, where there are a collection of parked vehicles. In the case of the shopping centers, the Baltimore County Police Department are aware of the number of vehicles stolen at these locations and work with the shopping center management offices to try to reduce the thefts. It also turns out that shopping centers are the most frequent locations for stolen vehicle retrievals, so it works both ways.

Fuzzy Mode

The usefulness of the mode, however, is dependent on the degree of resolution for the geo-referencing of incidents. In the case of the Baltimore vehicle thefts, thefts locations were assigned a single point at the address. Thus, all thefts occurring at any one shopping center are assigned the same X and Y coordinates. However, there are situations when the assignment of a coordinate will not be a good indicator of the hot spot location. For example, assigning the vehicle theft location to a particular stall in a parking lot will lead to few, if any, locations

coming up more than once. In this case, the mode would not be a useful statistic at all. Another example is assigning the vehicle theft location for the parking lot of a multi-building apartment complex to the address of the owner. In this case, what is a highly concentrated set of vehicle thefts become dispersed because the owners live at different addresses within the complex.

Consequently, *CrimeStat* includes a second point location hot spot routine called the *Fuzzy Mode*. This allows the user to define a small search radius around each location to include events that occur *around* or near that location. For example, a user can put a 50 yard or 100 meter search radius and the routine will calculate the number of incidents that occur at each location *and* within a 50 yard or 100 meter radius.

The aim of the statistic is to allow the identification of locations where a number of incidents may occur, but where there may not be precision in measurement.[2] For example, if several apartment complexes share a parking lot, any vehicle theft in the lot may be assigned to the address of the owner, rather than to the parking lot. In this case, the measurement is imprecise. Plotting the location of the vehicle thefts will make it appear that there are multiple locations, when, in fact, there is only approximately one.

Another example would be the measurement of motor vehicle crashes that all occur at a single intersection. If the measurement of the location is very precise, the crashes could be assigned to slightly different locations when, in fact, they occurred at more or less the same location. In other words, the fuzzy mode allows a flexible classification of a location where the analyst can vary slightly the area around a location.

The fuzzy mode output file is also a 'dbf' file and, like the mode, also includes four output variables:

1. The rank order of the location with 1 being the location with the most incidents, 2 being the location with the next most incidents, 3 being the location with the third most incidents, and so forth until only those locations which have only one incident each;

2 In the statistical literature, this type of statistic is known as a spatial scan with a fixed circular window (Kulldorff, 1997; Kulldorff and Nagarwalla, 1995). However, our emphasis here is on defining approximate point locations where there is either measurement error or very small locational differences. In this sense, the term 'fuzzy' is more similar to the classification literature where imprecise boundaries exist and an incident can belong to two or more groups (Bezdek, 1981; McBratney and deGruijter, 1992; Xie and Beni, 1991).

2. The frequency of incidents at the location. This is the number of incidents occurring at that location;

3. The X coordinate of the location; and

4. The Y coordinate of the location.

Note that allowing a search radius around a location means that incidents are counted multiple times, one for each radius they fall within. If used carefully, the fuzzy mode can allow the identification of high incident locations more precisely than the mode routine. But, because of the multiple counting of incidents that occurs, the frequency of incidents at locations will change, compared to the mode, as well as possibly the hierarchy.

To illustrate this, Figure 7.5 maps the top 13 locations for vehicle thefts identified by the fuzzy mode routine using a search radius of 300 feet (four were tied for number 2 and eight were tied for number 5). The 13 locations are displayed by a single magenta triangle and are compared to the 10 locations identified by the mode (blue circle). Notice that two of the 13 locations are clustered at the same places as those identified by the mode, but the other two triangles are different locations. Two of these locations have multiple fuzzy modes. The most southeastern triangle in Baltimore County actually includes three fuzzy modes while the one triangle within the City of Baltimore actually includes eight fuzzy modes.

Figure 7.6 zooms in to display the eight clustered locations within the City of Baltimore, each of which has a fuzzy mode count of 29 vehicle thefts. The eight fuzzy mode locations are actually eight parking lots within the Mondawin Shopping Mall. Since the parking lots are within 300 feet of each other, each has a cumulative count of 29 vehicle thefts. In other words, the fuzzy mode has identified a general location where there are multiple sub-locations in which vehicle thefts occur.

Uses of the Fuzzy Mode

The fuzzy mode routine can be useful because it allows the identification of small hot spot areas, rather than exact locations. Any one location may not have a sufficient number of incidents that occur at that location, but because it is close to other locations that have incidents occurring, the cumulative count may actually be quite high. Additional examples when it might be useful are in identifying multiple parking lots in parks or in identifying common parking areas for multi-unit buildings (e.g., large apartment complexes).

Figure 7.5:

Metropolitan Baltimore Vehicle Thefts: 1996

13 Most Frequent Vehicle Theft Locations within 300 Feet Search Radius

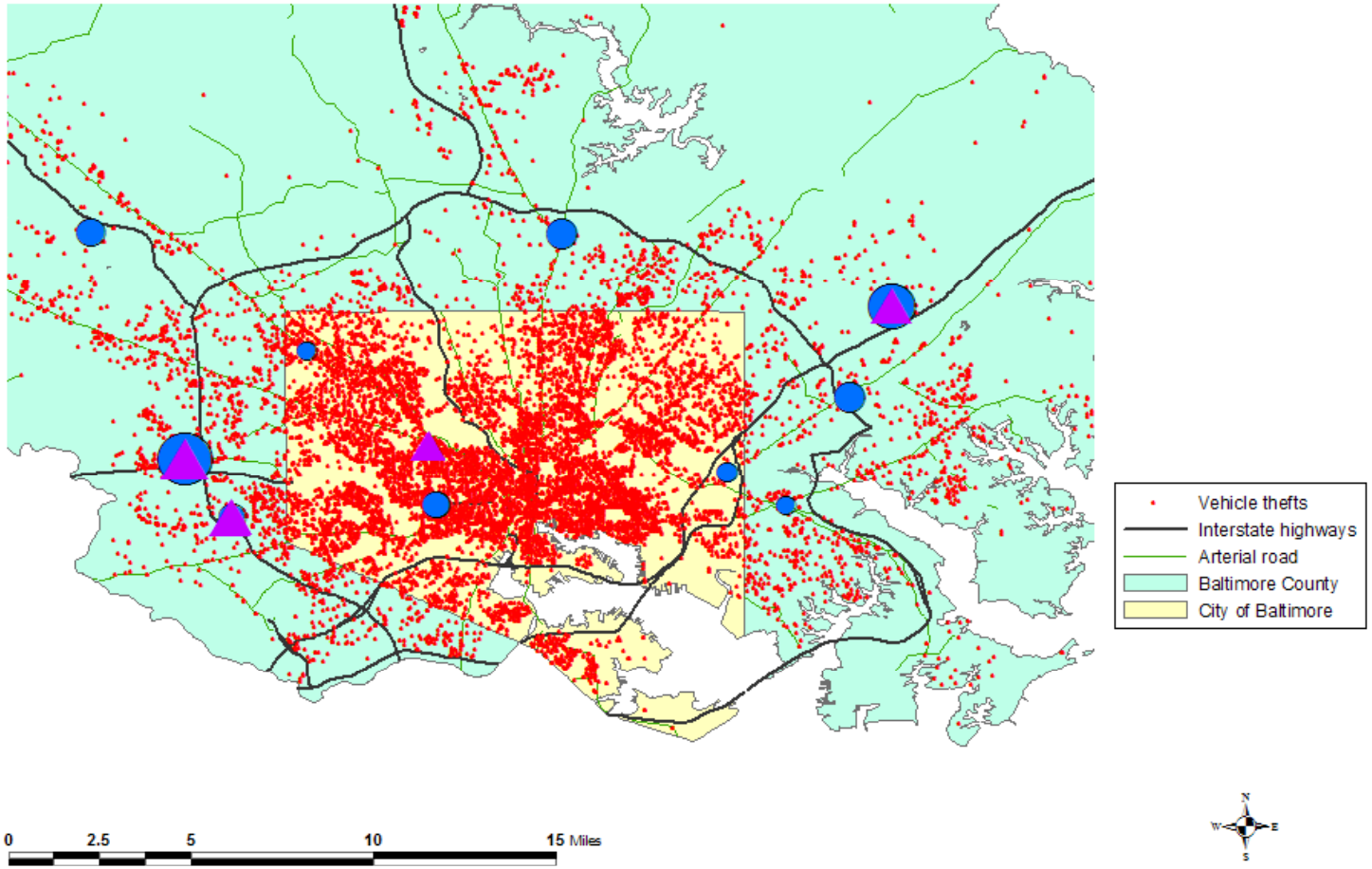
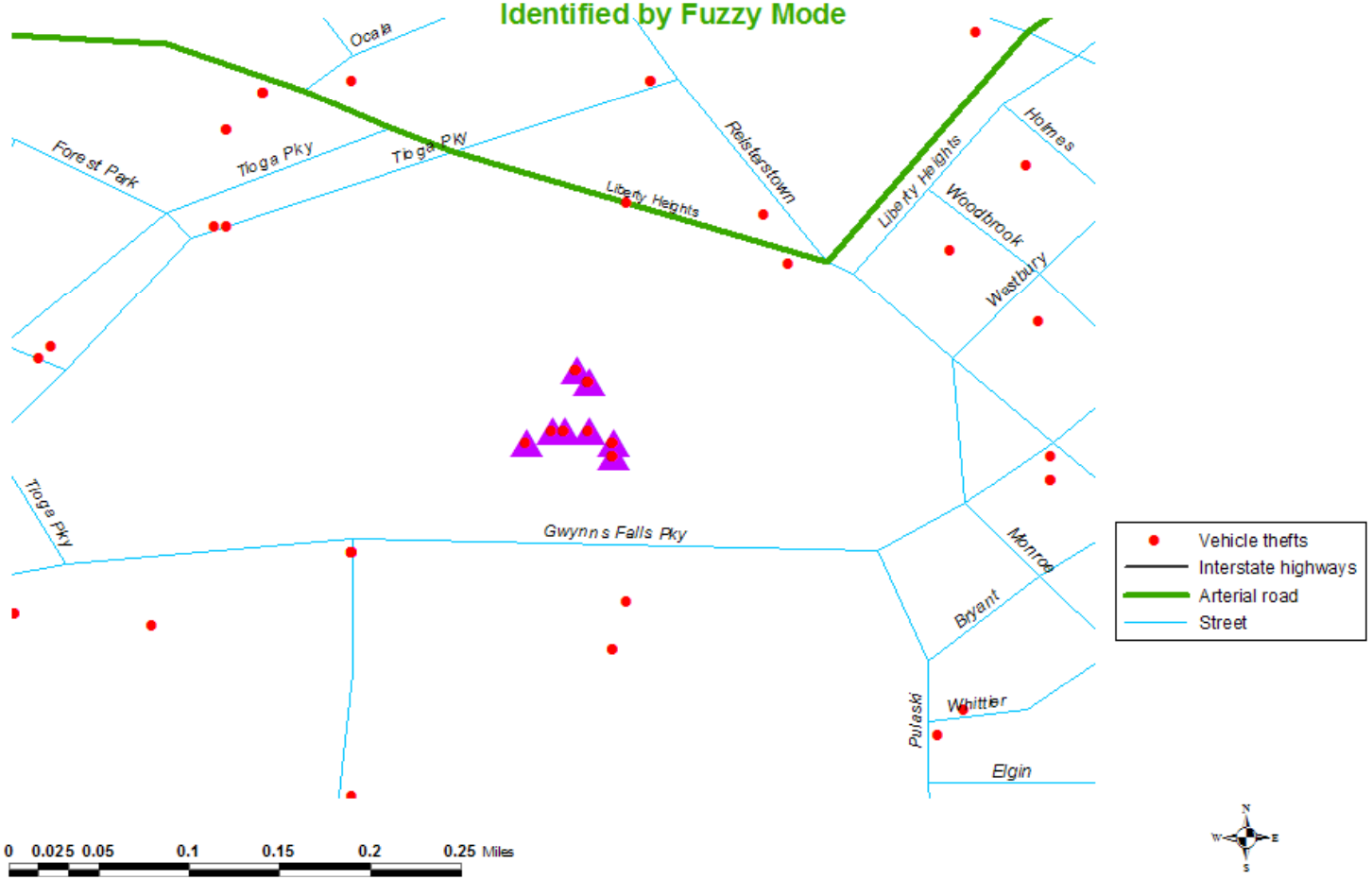


Figure 7.6:

Metropolitan Baltimore Vehicle Thefts: 1996

8 Concentrated Vehicle Theft Clusters within 300 Feet Search Radius

Identified by Fuzzy Mode



The method would also be useful for identifying hot spots when exact coordinates are specified for each incident. For example, in the parking lot example above, if each vehicle theft were identified by a stall number, as opposed to a single coordinate for the entire parking lot, few vehicle thefts would occur in exactly the same location. Allowing a search radius around the coordinates (the fuzzy part of the frequency count) allow a number of events to be grouped together whereas exact locations might not identify that grouping.

Limitations of the Fuzzy Mode

On the other hand, the fuzzy mode does involve duplicate counts points that are close to each other will be counted multiple times. This can allow distortion. By changing the search radius, the number of incidents counted for any one location changes as well as it's order in the hierarchy. For example, when a quarter mile search radius was used, all top locations occurred within a short distance of each other (not shown). In short, the user must be careful in using the fuzzy mode for analysis.

Nearest Neighbor Hierarchical Clustering

We now turn to methods that identify hot spot areas, as opposed to individual points that are clustered or are the center of a cluster. The *nearest neighbor hierarchical clustering* (or Nnh for short) routine in *CrimeStat* identifies groups of incidents that are spatially close. It is a hierarchical clustering routine that clusters points together on the basis of several criteria. The clustering is repeated until either all points are grouped into a single cluster or else the clustering criteria fail. Hierarchical clustering methods are among the oldest cluster routines (Everitt, Landau and Leese, 2001; King, 1967; Systat, 2008). Among the clustering criteria that have been used are the nearest neighbor method (Johnson, 1967; D'andrade. 1978), farthest neighbor, the centroid method (King, 1967), median clusters (Gowers, 1967), group averages (Sokal and Michener, 1958), and minimum error (Ward, 1967).

The *CrimeStat* Nnh routine is a variation on this approach but has its own unique algorithm. It uses a method that defines a *threshold distance* and compares the threshold to the distances for all pairs of points. Only points that are closer to one or more other points than the threshold distance are selected for clustering. In addition, the user can specify a minimum number of points to be included in a cluster. Only points that fit both criteria - closer than the threshold and belonging to a group having the minimum number of points, are clustered at the first level (first-order clusters).

The routine then conducts subsequent clustering to produce a hierarchy of clusters. The first-order clusters are themselves clustered into second-order clusters. Again, only clusters that are spatially closer than a threshold distance (calculated anew for the second level) are included.

The second-order clusters, in turn, are clustered into third-order clusters, and this re-clustering process is continued until either all clusters converge into a single cluster or, more likely, the clustering criteria fails.

Criterion 1: Threshold Distance

The first criterion in identifying clusters is whether points are closer than a specified threshold distance. There are two alternatives in selecting the threshold distance: 1) a random nearest neighbor distance (the default); or 2) a fixed distance.

Random nearest neighbor distance

The default alternative is to use the expected random nearest neighbor distance for first-order nearest neighbors. The user specifies a *one-tailed* confidence interval around the random expected nearest neighbor distance. The t-value corresponding to this probability level, t, is selected from the Student's t-distribution under the assumption that the degrees of freedom are at least 120.[3] This selection is controlled by a slide bar under the routine (see Figure 7.3). From chapter 6, the mean random distance was defined as:

$$d_{NN(ran)} = 0.5\sqrt{\frac{A}{N}} \qquad \text{repeat 6.2}$$

where A is the area of the region and N is the number of incidents and the standard error of the mean random distance is:

$$SE_{d(ran)} \cong \sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \qquad \text{repeat 6.5}$$

where A is the area of the region and N is the sample size (number of incidents). The confidence interval around that distance is defined as:

$$Confidence\ interval = \ d_{NN(ran)} \pm t * SE_{d(ran)} \qquad (7.1)$$

where t is the t-value associated with a probability level in the Student's t-distribution.

The approximate lower limit of this confidence interval is:

3 This is the next highest degree of freedom in the Student's t-table below infinity.

$$Lower\ limit\ of\ confidence\ interval = d_{NN(ran)} - t * SE_{d(ran)}$$

$$\cong 0.5\sqrt{\frac{A}{N}} - t\sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \tag{7.2}$$

and the upper limit of this confidence interval is:

$$Upper\ limit\ of\ confidence\ interval = d_{NN(ran)} + t * SE_{d(ran)}$$

$$\cong 0.5\sqrt{\frac{A}{N}} + t\sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \tag{7.3}$$

The confidence interval defines a probability for the distance between any *pair* of points. For example, for a specific *one-tailed* probability, *p*, fewer than *p*% of the incidents would have nearest neighbor distances smaller than this selected limit *if* the distribution was spatially random. *If* the data were spatially random and if the mean random distance is selected as the threshold criteria (the default position on the slide bar), approximately 50% of the pairs will be closer than this distance. For randomly distributed data, if a $p \leq .05$ level is taken for t (two steps to the left of the default or the fifth in from the left), then only about 5% of the pairs would be closer than the threshold distance. Similarly, if a $p \leq .75$ level is taken for t (one step to the right of the default or the fifth in from the right), then about 75% of the pairs would be closer than the threshold distance.

In other words, the threshold distance is a probability level for selecting any *two* points (a pair) on the basis of a chance distribution. The slide bar has 12 levels and is associated with a probability level for a t-distribution from a sample of 120 or larger. From the left, the p-values are approximately (Table 7.2):

Taking a broader conception of this, if there is a spatially random distribution, then for all distances between unique pairs of points, of which there are

$$Combinations = \frac{N(N-1)}{2} \tag{7.4}$$

fewer than *p*% will be shorter than this threshold distance.

Table 7.2:
Approximate Probability Values Associated with Threshold Scale Bar

Position	Scale Bar Probability	Description
1	0.00001	Far left point of slide bar
2	0.0001	Second from left
3	0.001	Third from left
4	0.01	Fourth from left
5	0.05	Fifth from left
6	0.1	Sixth from left
7	0.5	Sixth from right (default value)
8	0.75	Fifth from right
9	0.9	Fourth from righ
10	0.95	Third from righ
11	0.99	Second from righ
12	0.999	Far right point of slide bar

This does not mean, however, that the probability of finding a cluster is equal to this probability. It only indicates the probability of selecting two points (a pair) on the basis of a chance distribution. If additional points are to be included in the cluster, then the probability of obtaining the cluster will be less. Thus, the probability of selecting three points or four points or more points on the basis of chance will be much smaller.

Area must be defined correctly

Note that it is *very* important that area be defined correctly for this routine to work. If the user defines the area on the measurement parameters page (see chapter 3), the Nnh routine uses that value to calculate the threshold distance. If the user does not define the area on the measurement parameters page, the routine calculates the area from the minimum and maximum X/Y values (the bounding rectangle), which will usually be a larger area. In either case, the routine will be able to calculate a threshold distance and run the routine.

However, if the area units are defined incorrectly on the measurement parameters page, then the routine will certainly calculate the threshold distance wrongly. For example, if data are in feet but the area on the measurement parameters page are defined in square miles, most likely the routine will not find any points that are farther apart the threshold distance since that distance is defined in miles. In other words, it is essential that the area units be consistent with the data for the routine to properly work.

Fixed distance

The second alternative for selecting a threshold distance is to choose a fixed distance (in miles, nautical miles, feet, kilometers, or meters). The user checks the "Fixed distance" box and selects a threshold distance. The main advantage in this approach is that the search radius can be specified exactly. This is useful for comparing the number of clusters for different distributions (e.g., the number of robbery hot spots compared to burglary hot spots using a search radius of 0.5 miles). The main disadvantage of this method is that the choice of a threshold is subjective. The larger the distance that is selected, the greater the likelihood that clusters will be found by chance. Of course, this can be tested using a Monte Carlo simulation (see below).

Criterion 2: Minimum Number of Points

Whichever method is used for selecting a threshold distance, a second clustering criterion is the minimum number of points that are required for each cluster. This criterion is used to reduce the number of very small clusters. With large data sets, hundreds, if not thousands, of clusters can be found if only pairs of points are selected as being closer than a threshold distance. To minimize numerous very small clusters as well as reduce the likelihood that clusters could be found by chance, the user can set a minimum number restriction. The default is 10. This decision does not affect the selection of the clusters, only the number that are output. By decreasing this number, more clusters are output; conversely, by increasing this number, fewer clusters are output. The routine will only include points in the final clustering that are part of clusters in which the minimum number is found.

First-order Clustering

Using these criteria, *CrimeStat* constructs a first-order clustering of the points (see endnote *i*). For each first-order cluster, the center of minimum distance is output as the cluster center, which can be saved as a '.dbf' file.

Second and Higher-order Clusters

The first-order clusters are then tested for second-order clustering. The procedure is similar to first-order clustering except that the cluster centers (the center of minimum distance for each) are now treated as 'points' which themselves are clustered (see endnote *ii*). The process is repeated until no further clustering can be conducted. Either all sub-clusters converge into a single cluster, the threshold distance criterion fails, or there are fewer than four seeds in the higher-order cluster.

Visualizing the Cluster Output

To identify the approximate cluster location, *CrimeStat* allows the cluster to be output as either as an ellipse, a convex hull, or both.

Ellipse output

A standard deviational ellipse is calculated for each cluster (see chapter 4 for the definition). The user can choose between 1 standard deviation (the default), 1.5 standard deviations, or 2 standard deviations (indicated on the interface by 1X, 1.5X, and 2X). Typically, one standard deviation will cover more than 50% of the cases, one and a half standard deviations will cover more than 90% of the cases, and two standard deviations will cover more than 99% of the cases, although the exact percentage will depend on the distribution. The user specifies the number of standard deviations to save as ellipses in *ArcGIS* '.shp', *MapInfo* '.mif', *Google Earth* 'kml' (if the data are in spherical coordinates), or various Ascii formats.

Be careful as standard deviations can create an exaggerated view of the underlying cluster. The ellipse, after all, is an abstraction from the points in the cluster that may be arranged in an irregular manner. For example, for a regional view, a 1 standard deviational ellipse may not be very visible while for a small area, a 2 standard deviational ellipse may be too big. The user has to balance the need to accurately display the cluster compared to making it easier for a viewer to understand its location.

Convex hull output

A convex hull is calculated for each cluster (see chapter 4 for definition). The convex hull draws a polygon around the points in the cluster. It is a literal definition of the cluster, as opposed to the ellipse which is an abstraction. The convex hull can be saved in *ArcGIS* '.shp', *MapInfo* '.mif', *Google Earth* 'kml', or various Ascii formats.

Ellipse or convex hulls?

With the choice of an ellipse or a convex hull, the user can visualize clusters in two different ways. There are advantages and disadvantages of each approach. The convex hull has the advantage of being a polygon that corresponds exactly to the cluster. For neighborhood level analysis, it is probably preferable to the ellipse, which is an abstraction. On the other hand, any convex hull is based on a sample (e.g., this year's robberies compared to last year's robberies) and like any sample will vary from one instance to another. It may not capture all the space associated with the hot spot. The shape of a convex hull is often un-intuitive, following the outline of the incidents. An ellipse, on the other hand, is more general and will usually be more

stable from year to year. It usually looks better on a map or at least users seem to understand it better; it is a more familiar graphical object than an irregular polygon. The biggest disadvantage to an ellipse is that it forces a certain shape on the data, whether there are incidents in every part of it or not. So, in extreme cases, one finds ellipses that go outside of study area boundaries or extend into reservoirs or lakes or other features that are logically impossible to have incidents. At the same time, the ellipses may not include locations that are actually part of the hot spot..

In short, the user needs to balance the generality and visual familiarity of an ellipse with the limits of the actual hot spot. Probably for a small scale, regional perspective, the ellipses are preferable since a viewer can quickly see where the hot spots are located. For detailed neighborhood-level work, however, the convex hull is probably better since it shows where the incidents actually occurred.

Abstraction of incidents with second- and higher-order clusters

One thing to note is that second- and higher-order clusters can be visually misleading. The second-order clusters may visually encompass points that were not clustered in the first-order but they only are calculated using the centroids of the first-order clusters. Thus, in a GIS, one could select all incidents that fall within the boundaries of the second-order cluster (whether defined by an ellipse or a convex hull) and the number will generally be more than the points that were accumulated from the first-order clusters. A user needs to be aware of this as second- and higher-order clusters are abstractions from first- and earlier-order clusters.

Guidelines for Selecting Parameters

In the Nnh routine, the user has to define three parameters - the threshold distance, the minimum number of points, and the visual output of the hot spots. For a fixed threshold distance, the user has to choose a distance that is meaningful. For crime incidents, probably the threshold distance should not be more than 0.5 miles and, preferably, smaller.

If the random nearest neighbor distance is used as a threshold, the p-value is selected with a likelihood slider bar (see Figure 7.3). This bar indicates a range of p-values from 0.00001 (i.e., the likelihood of obtaining a pair by chance is 0.001%) to 0.999 (i.e., the likelihood of obtaining a pair by chance is 99.9%). The slider bar actually controls the value of *t* in equation 7.3, which varies from -3.719 to +3.090. The smaller the t-value, the smaller the threshold distance. With smaller threshold distances, fewer clusters are extracted and are typically smaller (although not always). Thus, they are more likely to be *not* due to chance.

If only pairs of points were being grouped, then the threshold distance would be critical. For example, with the default $p \leq .5$ value, then about half the pairs would be selected by chance

if the data were truly random. However, since there are a minimum number of points that are specified, the likelihood of finding a cluster with the minimum number of points is much smaller. The larger the minimum number selected, the smaller the likelihood of obtaining a cluster by chance.

Therefore, one can think of the slide bar as a filter for grouping points. One can make the filter smaller (moving the slide bar to the left) or larger (moving the slide bar to the right). There will be some effect on the final number of clusters, but the likelihood of obtaining a cluster by chance will be generally low. Statistically, there is more certainty with small threshold distances than with larger ones using this technique. Thus, a user must trade off the number of clusters and the size of an area that defines a cluster with the likelihood that the result could be due to chance.

This choice will depend on the needs of the user. For interventions around particular locations, the use of a small threshold distance may actually be appropriate; some of the ellipses seen in Figure 7.7 below cover only a couple of street segments. These define micro-neighborhoods. On the other hand, for a patrol route, for example, a cluster the size of several neighborhoods might be more appropriate. A patrol car would need to cover a sizeable area and having a larger area to target might be more appropriate than a 'micro' environment. However, there will be less precision with a larger cluster size in this type of area.

A second criterion is the minimum number of points that are required to define a cluster. If a cluster does not have this minimum number, *CrimeStat* will ignore the seed location. Without this criterion, the Nnh routine could identify clusters of two or three incidents each. A hot spot of this size is usually not very useful. Consequently, the user should increase the number to ensure that the identified cluster represents a meaningful number of cases. The default value is 10, but the user can type in any other value.

The user may have to experiment with several runs to get a solution that appears right. As a rule of thumb, start with the default settings. If there appears to be too many clusters, tighten up the criteria by selecting a lower probability for grouping a pair by chance (i.e., shifting the threshold distance to the left) or by increasing the minimum number of points required to be defined as a cluster (e.g., from 10 to 20). On the other hand, if there appears to be too few clusters, loosen the criteria by selecting a higher probability for grouping pairs by chance (i.e., shifting the threshold distance to the right) or decreasing the minimum number of points in a cluster (e.g., from 10 to 5). Then, once an appropriate solution has been found, the user can fine tune the results by slight changes.

In general, the minimum number of points criterion is more critical for the number of clusters than the threshold distance, though the latter can also influence the results. For example, with the 1996 Baltimore County robbery data set (N=1181 incidents), a minimum of 26 and a

maximum of 28 clusters were found by changing the threshold distance from the minimum p-value ($p \leq 0.00001$) to the maximum p-value ($p \leq 0.999$). On the other hand, changing the minimum number of points per clusters from 10 to 20 reduced the number of clusters found (with the default threshold distance) from 26 to 11.

The third criterion is the visual display of the clusters. The convex hull is literal; it will draw a polygon around the points in the cluster. The ellipse, on the other hand, requires a decision by the user on the number of standard deviations to be displayed. The choices are one (the default), one and a half, and two standard deviations. Typically, one standard deviation will cover more than 50% of the cases; one and a half standard deviations will cover more than 90% of the cases, while two standard deviations will cover more than 99% of the cases although the exact percentage will depend on the distribution.

In general, I recommend using a 1.5 as the default as 1 standard deviation will be often be too small while 2 standard deviations can create an exaggerated view of the underlying cluster. The user has to balance the need to accurately display the cluster compared to making it easier for a viewer to understand its location.

Nnh Output Files

The Nnh routine has six outputs: First, for each cluster that is identified, the hierarchical order and the cluster number; Second, for each cluster that is calculated, the mean center of the cluster. Only 45 of the seed locations are displayed on the screen. The user can scroll down or across by adjusting the horizontal and vertical slider bars and clicking on the *Go* button. This can be saved as a '.dbf' file; Third, the standard deviational ellipses of the clusters is shown, whether the graphical output is an ellipse or a convex hull. The size of the ellipses is determined by the number of standard deviations to be calculated (see above); Fourth, the number of points in the cluster; Fifth, the area of the ellipse; and, Sixth, the density of the cluster (number of points divided by area).

The ellipses and convex hulls can be saved in *ArcGIS* '.shp', *MapInfo* '.mif', *Google Earth* 'kml', or various Ascii formats. Because there are also orders of clusters (i.e., first-order, second-order, etc.), there is a naming convention that distinguishes the order.

Naming conventions for ellipses

For the ellipses, the convention is

Nnh<O><*username*>

where *O* is the order number and *username* is a name provide by the user. Thus,

Nnh1robbery

are the first-order clusters for a file called 'robbery' and

Nnh2NightBurglaries

are the second-order clusters for a file called 'NightBurglaries'. Within files, clusters are named

Nnh<O>Ell<N><*username*>

where*O* is the order number, *N* is the ellipse number and *username* is the user-defined name of the file. Thus,

Nnh1Ell10robbery

is the tenth ellipse within the first-order clusters for the file 'robbery' while

Nnh2Ell1NightBurglaries

is the first ellipse within the second-order clusters for the file 'NightBurglaries'.

For the convex hulls, the name will be output with a 'CNNH1' prefix for the first-order clusters, a 'CNNH2' prefix for the second-order clusters, and a 'CNNH3' prefix for the third-order clusters. Higher-order clusters will index only the number.

In other words, names of files and features can get complicated. The easiest way to understand this, therefore, is to import the file into one of the GIS packages and display it.

Example 1: Nearest Neighbor Hierarchical Clustering of San Antonio Robberies

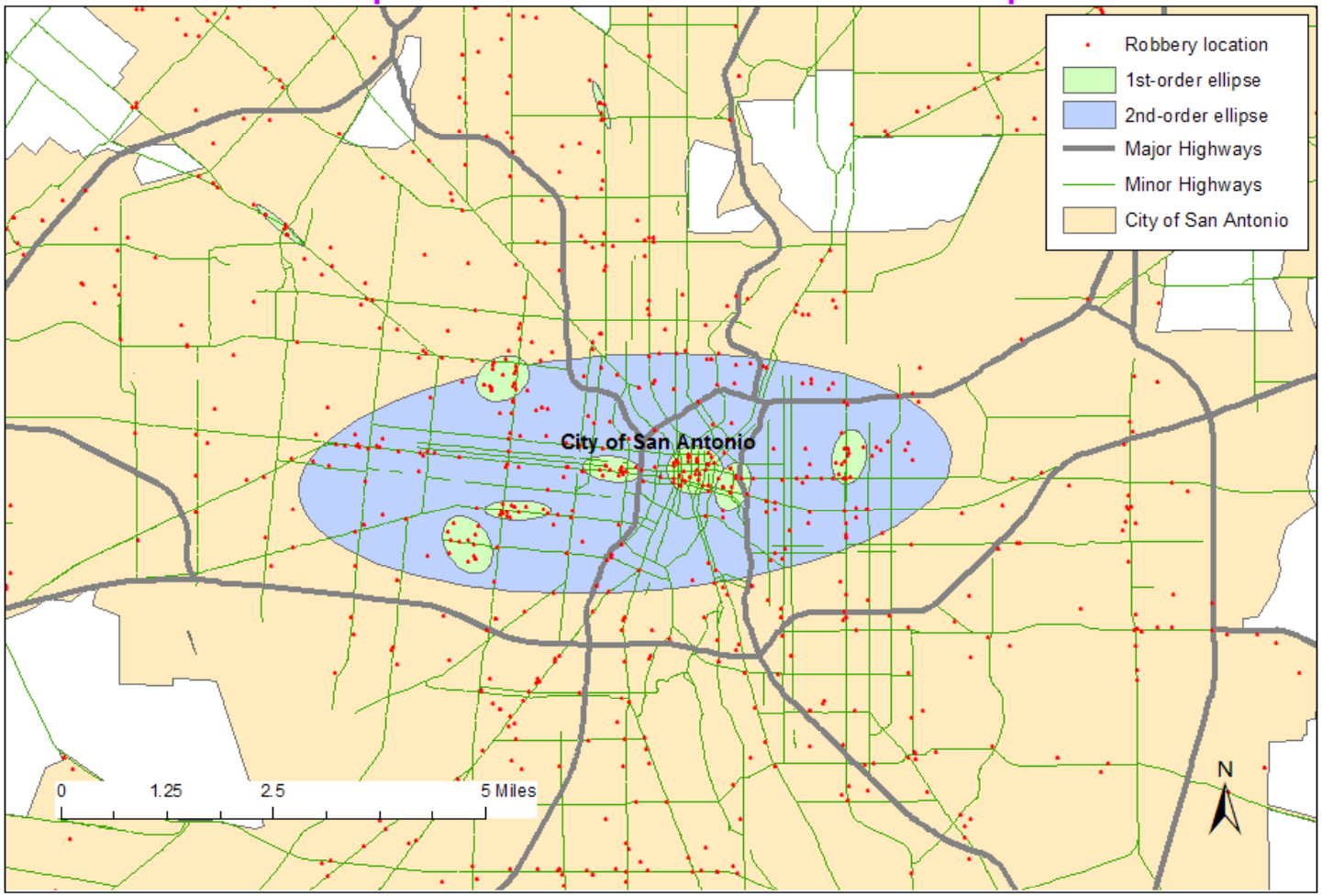
The Nnh routine was applied to 1,116 robberies that occurred in 2003 in San Antonio, TX. A default one-tailed probability level of .05 (or 5%) was selected for the threshold distance and each cluster was required to contain a minimum of 10 points (the default). Using these criteria, *CrimeStat* returned 9 first-order clusters and one second-order cluster. The 9 first-order clusters varied from 37 incidents for one cluster to 7 incidents for two clusters. Figure 7.7 shows the first-order clusters and the second-order cluster displayed as 1.5 standard deviational ellipses.

Since the criteria for clustering is the lower limit of the mean random distance, the distances involved are very small, as can be seen. Note, the standard deviational ellipse is

Figure 7.7:

San Antonio Robberies: 2003

Ellipses of 1st-order and 2nd-order Hot Spots



defined by the points in the cluster but is an abstraction, rather a literal definition. Thus, there is not a one-to-one match between the ellipse boundaries and the points included. For example, the top cluster had 37 points yet the 1.5 standard deviational ellipse included only 36 of those points.

Figure 7.8 shows the shows the same clusters as in Figure 7.7 but the clusters are displayed as convex hulls rather than ellipses. As seen, the convex hulls are irregular in shape and more limited in geographical spread; they show only the incidents that are clusters. Notice how the one second-order cluster defined by the convex hull is much more constrained than the ellipse definition of it.

Note also that the second-order cluster includes incidents that were not clustered in the first-order clusters. Thus, the area included in the second-order cluster is much greater than the sum of the first-order clusters from which it was derived. This may lead to a wider definition of a larger hot spot which may be real or not. One has to keep in mind that the second- and higher-order clusters are abstractions of the first-order clusters, and are not clusters by themselves.

Figure 7.9 zooms and compares the seven central clusters in terms of the ellipses and the corresponding convex hulls. Notice how the convex hulls are much more compact. Also, how the convex hulls 'stick out' beyond the ellipses for four of the clusters. Again, this is because the ellipse is a mathematical abstraction whose central axes are defined by the points, whereas the convex hull is defined by a polygon that defines an outer boundary.

From a policing viewpoint, a convex hull is probably more useful in that it shows where the hot spot incidents are actually located. As mentioned above, the polygons created by the convex hulls are irregular and are, therefore, less familiar to most people. Consequently, for presentations of crime patterns at a regional level or even neighborhood-level for non-specialists, the ellipses may convey better where the hot spots are located.

Simulating Statistical Significance

Testing the significance of clusters from the Nnh routine is complex. Conceptually, using the random nearest neighbor distance for the threshold distance defines the probability that two points could be grouped together on the basis of chance. The test is for the confidence interval around the first-order nearest neighbor distance for a random distribution. If the probability level is p%, then approximately p% of all pairs of points would be found under a random distribution. Under this situation, we would know whether the number of clusters (pairs) that were found were significantly greater than would be expected on the basis of chance.

The problem is, however, that the routine is not just clustering pairs of points, but clustering as many points as possible that fall within the threshold distance since there is an

Figure 7.8:

San Antonio Robberies: 2003

Convex Hulls of 1st-order and 2nd-order Hot Spots

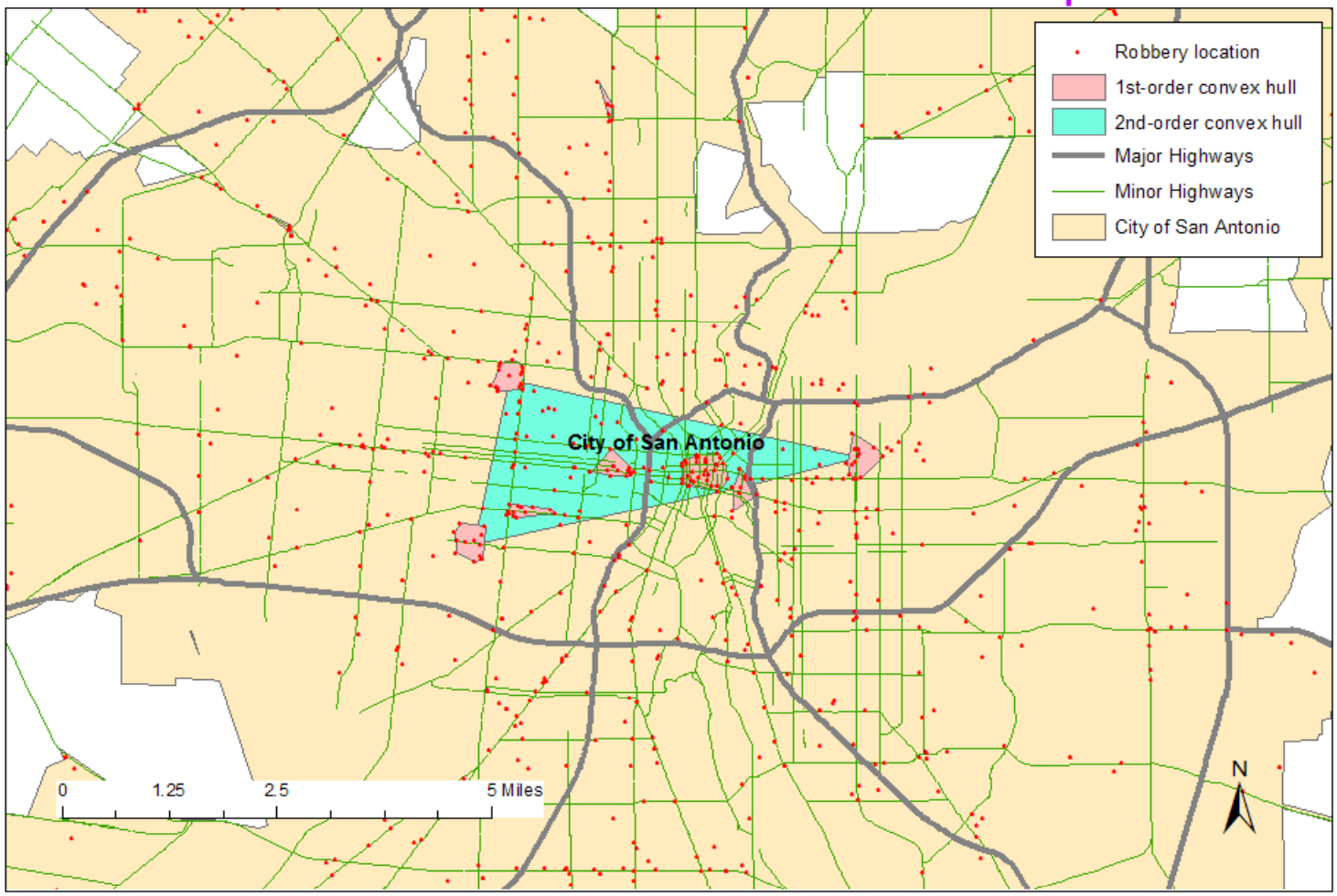
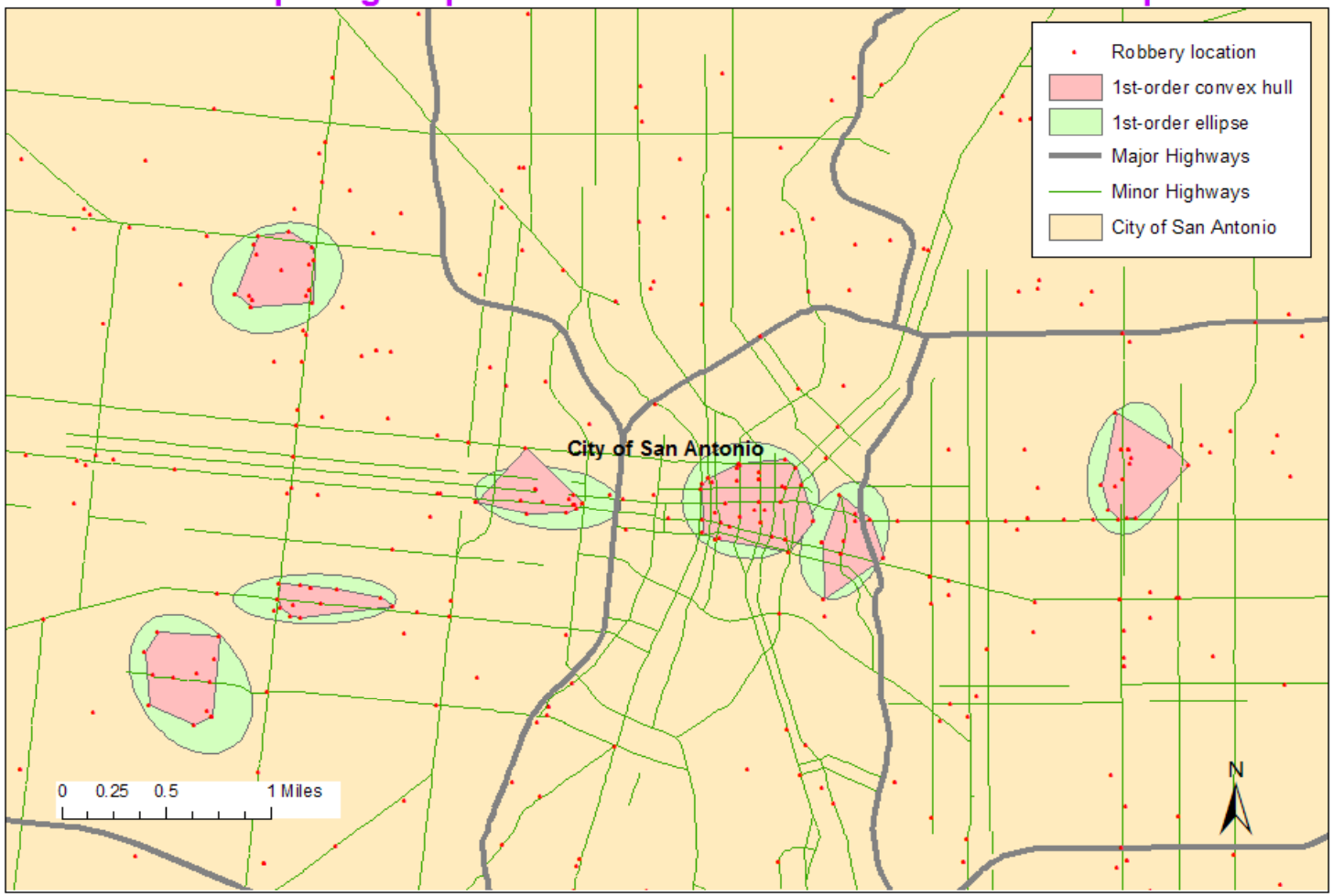


Figure 7.9:

San Antonio Robberies: 2003

Comparing Ellipses and Convex Hulls of 1st-order Hot Spots



additional requirement that there be a specified minimum number of points, with the minimum defined by the user. The probability distribution for this situation is not known. Consequently, there is a necessity to resort to a Monte Carlo simulation of randomness under the conditions of the Nnh test (Dwass, 1957; Barnard, 1963).

CrimeStat includes a Monte Carlo simulation routine that produces approximate confidence intervals (called *credible intervals*) for the first-order Nnh clusters that have been identified. Second- and higher-order clusters are not simulated since their structure depends on the first-order clusters. Essentially, the routine assigns *N* cases randomly to a rectangle with the same area as the defined study area, *A*, and evaluates the number of clusters according to the defined parameters (i.e., threshold distance and minimum number of points). It repeats this simulation *K* times where *K* is defined by the user (e.g., 100, 1,000, 10,000). By running the simulation many times, the user can assess approximate credible intervals for the particular first-order Nnh.

The output includes five columns and twelve rows:

Columns:

1. The percentile,
2. The number of first-order clusters found for that percentile,
3. The area of the cluster for that percentile,
4. The number of points in the cluster for that percentile, and
5. The density of points (per unit area) for that percentile.

Rows:

1. The minimum (smallest) value obtained,
2. 0.5^{th} percentile,
3. 1^{st} percentile,
4. 2.5^{th} percentile,
5. 5^{th} percentile,
6. 10^{th} percentile,
7. 90^{th} percentile,
8. 95^{th} percentile,
9. 97.5^{th} percentile,
10. 99^{th} percentile,
11. 99.5^{th} percentile, and
12. The maximum (largest) value obtained.

The percentiles are calculated as follows. First, over all simulation runs (e.g., 1000), the routine calculates the number of first-order clusters obtained for each run, sorts them in ascending order, and defines the percentiles for the list. Thus, the minimum is the fewest number of clusters obtained over all runs, the 0.5 percentile is the lowest half of a percent for the number of clusters obtained over all runs, and so forth until the maximum number of clusters obtained over all runs. The routine does *not* calculate second- or higher-order clusters since those are dependent on the first order clustering. Second, within each run, the routine calculates the number of points per cluster, the area of each ellipse, and the density of each ellipse. Then, it groups all clusters together, over all runs, and sorts them into a list. The percentiles for individual clusters are then calculated. Note that the points refer to the cluster whereas the area and density refer to the ellipses, which is a geometrical abstraction from the cluster.

When a Monte Carlo simulation of 1000 iterations was run on the San Antonio robbery data, no clusters were found. That is, given the criteria that were used for clustering (the default random nearest neighbor distance and a minimum of 10 incidents per cluster), it would be very unlikely to find any clusters on the basis of chance!

To illustrate how a simulation which found random clusters looks, Table 7.3 presents an Nnh run that was conducted on a Baltimore County robbery data base (N=1181 incidents) using the default threshold distance ($p \leq .5$ for grouping a pair by chance) and a minimum number of points of at least five for each cluster. Then, 1000 Monte Carlo runs were conducted with simulated data. With the actual data, the Nnh routine identified 69 first-order clusters and 7 second-order clusters. Table 7.3 presents the parameters for the first ten first-order clusters.

In examining a simulation, one has to select percentiles as choice points. In this example, we use the 95^{th} percentile. That is, we are willing to accept a one-tailed Type I error of only 5% since we are only interested in finding a greater number of clusters than by chance. For the simulation, look at each column of the simulation results in turn. Column 2 presents the number of clusters found in each simulation. Over the 1000 runs, there was a minimum of one cluster found (for at least one simulation) and a maximum of 7 clusters found (for at least one simulation). That is, running 1000 simulations of randomly assigned data only yielded between 1 and 7 clusters using the parameters defined in the particular Nnh run. The 95^{th} percentile was 3. It is highly unlikely that the 69 first-order clusters that were identified would have been due to chance. That is, we would have expected at most three of them to have been due to chance. It appears that the robbery data is significantly clustered, though we have only tested significance through a random simulation.

Of course, the routine is not going to identify which three clusters could have been selected on the basis of chance. However, realistically the three clusters would be those with the lowest density, number of points per unit area (e.g., points per square mile; points per square

Table 7.3:
Simulated Confidence Intervals for Nnh Routine
Baltimore County Robberies: N=1181

Nearest Neighbor Hierarchical Clustering:

Sample size.....:	1181
Likelihood of grouping pair of points by chance....:	0.50000 (50.000%)
Z-value for confidence interval.....:	0.000
Measurement type.....:	Direct
Output units.....:	Miles, Squared Miles, Points per Squared Miles
Clusters found.....:	76
Simulation runs.....:	1000

Displaying ellipses starting from 1 (*ONLY 10 SHOWN*)

Order	Cluster	Mean X	Mean Y	Rotation	X-Axis	Y-Axis	Area	Points	Density
1	1	-76.44927	39.31455	77.09164	0.28303	0.09636	0.08568	40	66.828013
1	2	-76.60219	39.40050	11.98132	0.11540	0.27452	0.09952	33	331.580616
1	3	-76.44601	39.30490	16.66988	0.21907	0.16239	0.11176	25	23.684859
1	4	-76.78123	39.36088	25.36983	0.27643	0.14530	0.12618	29	229.826284
1	5	-76.73103	39.34319	67.71617	0.19445	0.16058	0.09810	29	295.628310
1	6	-76.72945	39.28910	79.88383	0.16428	0.25957	0.13396	29	216.476166
1	7	-76.51486	39.25986	87.32563	0.19148	0.29428	0.17703	27	152.520725
1	8	-76.45374	39.32106	54.57635	0.15150	0.18261	0.08692	7	80.538112
1	9	-76.75368	39.31132	89.56994	0.19748	0.22914	0.14216	22	154.753006
1	10	-76.71641	39.29139	10.43857	0.15048	0.16879	0.07980	14	175.444372

Etc.

Distribution of the number of clusters found in simulation (percentile):

Percentile	Clusters	Area	Points	Density
min	1	0.03845	5	15.615111
0.5	1	0.04922	6	16.608967
1.0	1	0.05603	6	17.162252
2.5	1	0.06901	6	18.570113
5.0	1	0.08243	6	19.468353
10.0	1	0.10045	6	21.256559
90.0	2	0.28706	7	61.173748
95.0	3	0.31074	7	73.463654
97.5	3	0.32442	7	87.550868
99.0	4	0.35279	8	115.460337
99.5	5	0.36489	8	122.625375
max	7	0.38424	9	156.056837

kilometer). Thus, the user could assume that the three clusters with the lowest density are less certain to be real than due to chance.

Column 3 shows the areas of clusters that were found over the 1000 runs using the ellipse as a definition for the clusters. For the individual clusters, the simulation showed a range from about 0.04 to 0.38. The 95th percentile was 0.31. In the actual Nnh, the area of clusters varied between 0.05 and 0.27, indicating that *all* first-order clusters were smaller than the smallest value found in the simulation. In other words, the real clusters are more compact than random clusters even though the random clusters were subject to the same threshold distance as the real data. This is not always true, but, in this case, it is.

Column 4 presents the number of points found per cluster in the simulation; these varied between 5 and 9 points per cluster. The 95th percentile was 7. With the actual data, the number of points varied between 5 and 40. Thus, some of the clusters could have been due to chance, at least in terms of the number of points per cluster. Analyzing the distribution (not shown), 27 of the 69 clusters had 7 or fewer points. In other words, about 39% had only as many points as might be expected on the basis of a chance distribution. Putting it another way, about 40% of the clusters had more points than would be expected on the basis of chance 95% of the time.

Finally, column 5 presents the density of points found per cluster. Since the output unit is squared miles, density is the number of points per square mile. The simulation presents a range from 15.6 points per square mile to 156.1 points per square mile. The 95th percentile was 73.4 points per square mile. The actual Nnh, on the other hand, finds a range of densities from 27.1 points per square mile to a very high number (11071821 points per square mile). Again, there is overlap between the actual clusters and what might be expected on the basis of chance; 26 out of 69 clusters have densities that are lower than the 95th percentile found in the simulation. Again, about 38% have densities are not different than would be expected on the basis of chance.

It should be clear that testing the significance of a cluster analysis is complex. In the example, some of the criteria chosen were definitely different than a chance distribution while other criteria were not very different. However, which of these criteria should be used to evaluate the actual distribution? We argue that it should be the number of incidents/points identified in the cluster, rather than the area or density by themselves since the area has to be defined by a polygon (ellipse or convex hull). The number of points is the relevant criterion since it is one of the criteria used for the clustering in the Nnh algorithm (the other being points that are closer than the threshold distance.

Uses of Hierarchical Clustering

There are four uses for the nearest neighbor hierarchical clustering technique. First, it can identify small geographical environments where there are concentrated incidents. This can be useful for specific targeting, either by police deployment or community intervention. There are clearly micro-environments that generate crime incidents and the Nhh technique tends to identify these small environments because the lower limit of the mean random distance is used to group the clusters. The user can, of course, control the size of the grouping area by loosening or tightening either the threshold distance or the minimum number of required points. Thus, the sizes of the clusters can be adjusted to fit particular groupings of points.

Second, the technique can be applied to any entire data set, such as for Baltimore County and Baltimore City, and need not only be applied to smaller geographical areas, such as precincts. This increases the ease of use for analysts and can facilitate comparisons between different areas without having to limit arbitrarily the data set.

Third, the linkages between several small clusters can be seen through the second- and higher-order clusters. Frequently, hot spots are located near other hot spots which, in turn, are located still near other hot spots.

In other words, the clustering of incidents, such as robberies, is hierarchical. With the San Antonio robbery data, we found two levels of grouping (first-order and second-order). With larger datasets, however, frequently third-order or, even, fourth-order hot spots can be found. Within these large areas, there are smaller hot spots and within some of those hot spots, there are even smaller ones. In other words, there are different scales to the clustering of points - different geographical levels, if you will, and the hierarchical clustering technique can identify these levels.

Typically, in cities as well as in small towns, there is a greater concentration towards the center of the settlement or city than at the periphery. This concentration necessarily means there will be more incidents (of any sort) towards the center than toward the periphery. The Nnh routine captures this logic very nicely because it seeks clusters systematically from the incident level upwards. More first-order clusters are going to be found in the center than in the periphery and this is also going to be true for second- and higher-order clusters since they build systematically on the first-order clusters. One can think of the first-order clusters as ‘building blocks’ for spatial autocorrelation. Thus, theoretically, hierarchical clusters capture the organization of a human settlement, particularly a city, in a way that no other clustering technique does.

Fourth, each of the levels implies different policing strategies. For the smallest level, officers can intervene effectively in small neighborhoods, as discussed above. Second-order clusters, on the other hand, are more appropriate as patrol areas; these areas are larger than first-order clusters, but include several first-order clusters within them. If third- or higher-order clusters are identified, these are generally areas with very high concentrations of crime incidents over a fairly large section of the jurisdiction. The areas start to approximate precinct sizes and need to be thought of in terms of an integrated management strategy - police deployment, crime prevention, community involvement, and long-range planning. Thus, the hierarchical technique allows different security strategies to be adopted and provides a coherent way of approaching these communities and gives flexibility to the analyst in order to choose an appropriate level of analysis. This depends, of course, on the need. For patrol cars covering an area, such as is common in the United States, larger hot spot areas are more appropriate. Police cars will drive around the area and will cover blocks and neighborhoods that don't necessarily have high crime in order to demonstrate their presence as well as make their behavior less predictable. For this use, second- or higher-order hot spots would be appropriate. Also, some of the techniques discussed in Chapters 8 and 10 are also appropriate for larger area analysis.

However, if the policing strategy involves working with businesses or even residents to develop, for example, a business- or neighborhood watch program, then the boundaries of the hot spot need to be defined fairly specifically, perhaps a block or two. Choosing a larger area may diffuse efforts and reduce the effectiveness of the intervention. Even more precise boundary definition are needed for public infrastructure improvements, such as improved lighting or closed circuit television systems (CCTV). The public works departments that install these improvements need to know exactly where to put the lights or CCTV cameras.

In other words, the analytical need is going to depend on the particular type of intervention or program that will be introduced and the hierarchical clusters provide a range of scales from which an appropriate one could be chosen.

Limitation to Hierarchical Clustering

There are also limitations to the technique, some technical and others theoretical. First, the method only clusters incidents (points); a weighting or intensity variable will have no effect. In Chapter 9, we introduce a variant of the Nnh that allows weighting incidents and can be applied to zonal data. The results are reasonable approximations to clusters of zones, but they lack the specificity of the incident data.

Second, the size of the grouping area is dependent on the sample size when the confidence interval around the mean random distance is used as the threshold distance criteria (see equation. 6.2). For crime distributions that have many incidents (e.g., burglary), the

threshold distance will be a lot smaller than distributions that have fewer incidents (e.g., robbery). In theory, a hot spot is dependent on an environment, not the number of incidents. Thus, that approach does not produce a consistent definition of a hot spot area. Using a fixed distance for the threshold distance can partly overcome this. However, the fixed distance needs to be tested for randomness using the Monte Carlo simulation.

Third, there is some arbitrariness in the technique due to the minimum points rule. This implicitly requires the user to define a meaningful cluster size, whether the number of minimum points required is 5, 10, 15 or whatever. To some extent, this is how patterns are defined by human beings; with one or two incidents in a small area, people do not perceive any pattern. As soon as the number of incidents increases, say to 10 or more, people perceive the pattern. This is not a statistical way for defining regularity, but it is a human way. However, it can lead to arbitrariness since two different users may interpret the size of a hot spot differently. Similarly, the selectivity of the p-value, vis-a-via the Student's t-distribution, can allow variability between users.

In short, the technique produces a consistent result, but one subject to manipulation by users. Hierarchical techniques are, of course, not the only clustering procedures to allow users to adjust the parameters; in fact, almost all the cluster techniques have this property. But it is a statistical weakness in that it involves subjectivity and is not necessarily consistently applied across users.

Finally, there is no substantive theory or rationale behind the clusters. They are empirical derivatives of a procedure. Again, many clustering techniques are empirical groupings and also do not have any explanatory theory.[4] If one is looking for a substantive hot spot defined by a unique constellation of land uses, activities, and targets, the technique does not provide any insight into why the clusters are occurring or why they could be related. I will return to this point at the end of the next chapter, but it should be remembered that these are empirical groupings, not necessarily substantive ones.

Risk-Adjusted Nearest Neighbor Hierarchical Clustering

CrimeStat also includes a risk-adjusted nearest neighbor hierarchical clustering routine (Rnnh), which is a variation on the Nnh routine discussed above. It combines the hierarchical clustering capabilities of the Nnh routine with kernel density interpolation technique that is discussed in Chapter 10.

---

4 A number of clustering techniques have a statistical theory behind them (e.g., Kulldorff, 1997), but not a substantive theory. While one can define consistent statistical criteria for identifying hot spots, this does not constitute an explanation for why the hot spots occurred. For this, other information is necessary.

The Nnh routine identifies clusters of points that are close together. That is, it will identify groups of points that are closer together than a threshold distance and in which the minimum number of points is greater than a user-defined value. Many of these clusters, however, are due to a high concentration of persons in the vicinity. That is, because the population is not arranged randomly over a plane, but is, instead, highly concentrated in population centers, there is a higher likelihood of incidents happening (whatever they are) simply due to the higher population concentration. In the above examples, many of the clusters for Baltimore burglaries or vehicle thefts were due primarily to a high concentration of households and vehicles in the center of the metropolitan area. In fact, one would normally expect a higher concentration of incidents in the center since there are more persons residing in the center and, certainly, more persons being concentrated there during the daytime through employment, shopping, cultural attendance, and other urban activities.

For many police purposes, the concentration of incidents is of sufficient interest in itself. Police have to intervene at high incidence locations irrespective of whether there is also a larger population at those locations. The demands for policing and responding to community emergency needs is population sensitive since there are more demands where there are more persons. From a service viewpoint, the concentration of incidents is what is important.

But for other purposes, the concentration of incidents relative to the baseline population is of interest. Crime prevention activities, for example, are aimed at reducing the number of crimes that occur for every area in which they are applied. For these purposes, the *rate* of decrease in the number of crimes is the prime focus. Similarly, after-school programs are aimed at neighborhoods where there is a high risk of crime, whether or not there is also a large population. In other words, for many purposes, the *risk* of crime or other types of incidents is of paramount importance, rather than the *volume* (i.e., absolute amount) of crime by itself. If the aim is to assess where there are high risk clusters, then the Nnh routine is not appropriate.

CrimeStat includes a Risk-adjusted Nearest Neighbor Hierarchical Clustering routine (or Rnnh) that defines clusters of points that are closer than what would be expected on the basis of a baseline population. It does this by dynamically adjusting the threshold distance in the Nnh routine according to the distribution of a second, baseline variable. Unlike the Nnh routine where the threshold distance is constant throughout the study area (i.e., it is used to pair points irrespective of where they are within the area), the Rnnh routine adjusts the threshold distance according to what would be expected on the basis of the baseline variable. It is a *risk* measure, rather than a volume measure.

Dynamic Adjustment of the Threshold Distance

To understand how this works, think of a simple example. In a typical metropolitan area, there are more people living towards the center than in the periphery. There are topographical and social factors that might modify this (e.g., an ocean, a mountain range, a lake), but in general population densities are much higher in the center than in the suburbs. If a different baseline variable were selected than population, for example, employment, one would generally find even higher concentrations since central city employment tends to be very high relative to suburban employment. Thus, if population or employment (or another variable that is correlated with population density) is taken as the baseline, then one would expect more people and, hence, more incidents occurring in the center rather than the periphery. In other words, all other things being equal, there should be more robberies, more burglaries, more homicides, more vehicle thefts, and more of any other type of event in the center than in the periphery of an urban area. This is just a by-product of urban societies.

Using this idea to cluster incidents together, then, intuitively, the threshold distance must be adjusted for the varying population densities. In the center, the threshold must be short since one would expect there to be more persons. Conversely, in the periphery - the far suburbs, the threshold distance must be a lot longer since there are far fewer persons per unit of area. In other words, *dynamic adjustment* of the threshold grouping distance means changing the distance inversely proportional to the population density of the location; in the center, a high density means a short threshold distance and in the periphery, a low density means a larger threshold distance.

Kernel Adjustment of the Threshold Distance

To implement this logic, *CrimeStat* overlays a standard grid and uses an interpolation algorithm, based on the kernel density method, to estimate the expected number of incidents per grid cell *if* the actual incident file was distributed according to the baseline variable. Chapter 10 discusses in detail the kernel density method and the reader should be familiar with the method before attempting to use the Rnnh routine. If not, the author highly recommends that Chapter 10 be read before reading the rest of this section.

Steps in the Rnnh Routine

The Rnnh routine works as follows:

1. Both primary and secondary files are required. The primary file is the basic file of incidents (e.g., robberies) while the secondary file is the baseline variable (e.g., population of zones; all crimes as a baseline; or another baseline variable). If the

baseline variable is identified by zones, the user must define both the X and Y coordinates as well as the variable assigned to the zone (e.g., population); the latter will typically be an intensity or weight variable (see Chapter 3).

2. A grid is defined in the reference file tab of the data setup section (see Chapter 3). The Rnnh routine takes the lower-left and upper-right limits of the grid, but uses a standard number of columns (50).

3. The area of the study is defined in the measurement parameters tab of the data setup section (see Chapter 3). If no area is defined, the routine uses the area of the entire grid.

4. The user checks the Risk-adjusted box under the Nnh routine. The risk variable is estimated with the parameters defined in the Risk Parameters box. These are the kernel parameters. Without going into detail, the user must define:

   A. The method of interpolation, which is the type of kernel used: normal, uniform, quartic, triangular, or negative exponential. The normal distribution is the default.

   B. The choice of bandwidth, whether a fixed or adaptive (variable) bandwidth is used. For a fixed bandwidth, the user must define the size of the interval (e.g., 0.5, miles; 2 kilometers). For an adaptive bandwidth, the user must define the minimum sample size to be included in the circle that defines the bandwidth. The default is an adaptive bandwidth with a minimum sample size of 100 incidents.

   C. The output units, which are points per unit of area: squared miles, squared nautical miles, squared feet, squared kilometers, or squared meters. The default is squared miles.

   D. Also, if an intensity or weight variable is used (e.g., the centroids of zones with population being an intensity variable), the intensity or weight box should be checked (be careful about checking both if there are both an intensity and a weight variable).

   Consult Chapter 10 for more detail about these parameters.

5. Once the baseline variable (the secondary file) is interpolated to the grid using the above parameters, it is converted into absolute densities (points per grid cell) and

re-scaled to the same sample size as the primary incident file. This has the effect of making the interpolation of the baseline variable the same sample size as the incident variable. For example, if there are 1000 incidents in the primary file, the interpolation of the secondary file will be re-scaled so that all grid cells add to 1000 points, irrespective of how many units the secondary variable actually represented. This creates a distribution for the primary file (the incidents) that is proportional to the secondary file (the baseline variable) if the primary file had the same distribution as the secondary file. It is then possible to compare the actual distribution of the incident variable with the expected distribution *if* it was similar to the baseline variable.

6. Once the risk parameters have been defined, the selection of parameters is similar to the Nnh routine with one exception.

   A. The threshold probabilities are selected with the scale bar. The probabilities are identical to those in Table 7.2.

   B. However, for each grid cell, a *unique threshold distance* is defined using formulas similar to equations 7.1 and 7.2. The difference is, however, that the formulas are applied to each grid cell with a unique distance for each grid cell (formulas 7.5-7.8):

$$Mean\ random\ distance\ of\ grid\ cell\ i = d(ran)_i = 0.5\sqrt{\frac{A_i}{N_i}} \tag{7.5}$$

where A_i is the area of the grid cell and N_i is the *estimated number of points* from the kernel density interpolation. Thus, each grid cell has its own unique expected number of points, N_i, its own unique area, A_i (though, in general, all grid cells will have approximately equal areas), and, consequently, its own unique threshold distance.

$$Confidence\ interval\ for\ mean\ random\ distance\ of\ grid\ cell\ i\ =$$

$$d(ran)_i \pm SE_{d(ran)_i} = 0.5\sqrt{\frac{A_i}{N_i}} \pm t\frac{0.26136}{\sqrt{\frac{N_i^2}{A_i}}} \tag{7.6}$$

where the Mean Random Distance of Grid Cell i, A_i and Ni are as defined above, t is the t-value associated with a probability level in the Student's t-distribution (defined by the scale bar).

The lower limit of this confidence interval is:

$$Lower\ limit\ of\ confidence\ interval\ for\ mean\ random\ distance\ of\ grid\ cell\ i =$$

$$0.5\sqrt{\frac{A_i}{N_i}} - t\frac{0.26136}{\sqrt{\frac{N_i^2}{A_i}}} \tag{7.7}$$

and the upper limit of this confidence interval is

$$Upper\ limit\ of\ confidence\ interval\ for\ mean\ random\ distance\ of\ grid\ cell\ i =$$

$$0.5\sqrt{\frac{A_i}{N_i}} + t\frac{0.26136}{\sqrt{\frac{N_i^2}{A_i}}} \tag{7.8}$$

C. In addition, the user defines a minimum sample size for each cluster, as with the Nnh routine.

6. The actual incident points are then identified by the grid cell that they fall within and the unique threshold distance (and confidence interval) for that grid cell. For each pair of points that are compared for distance, there is, however, asymmetry since the threshold distance for each point may be different if they are in different grid cells. That is, the unique threshold distance for point A will not necessarily be the same as that for point B. The Rnnh routine, therefore, requires the distance between each pair of points to be the *shorter* of the two distances between the points.

7. Once pairs of points are selected, the Rnnh routine proceeds in the same way as the Nnh routine.

In other words, points are clustered together according to two criteria. First, they must be closer than a threshold distance. However, the threshold distance varies over the study area and is inversely proportional to the baseline variable. Only points that are closer together than would be expected on the basis of the baseline variable are selected for grouping. Second, clusters are required to have a minimum number of points with the minimum being defined by the user. The result is clusters that are more concentrated than would be expected, not just from chance but, from the distribution of the baseline variable. These are *high risk* clusters.

Guidelines for Selecting Parameters

The guidelines for selecting parameters in the Rnnh routine are similar to the Nnh except the user must also model the baseline variable using a kernel density interpolation. There are several guidelines that should be followed in developing the model.

Area must be defined correctly

First, it is essential that area be defined correctly for this routine to work. If the user defines the area on the measurement parameters page (see chapter 3), the Rnnh routine uses that value to calculate the area of each grid cell and, in turn, the grid-specific threshold distance. If the user does not define the area on the measurement parameters page, the routine calculates the total area from the minimum and maximum X/Y values (the bounding rectangle) and uses that value to calculate the area of each grid cell and, in turn, the grid-specific threshold distance. In either case, the routine will be able to calculate a threshold distance for each grid cell and run the routine.

However, if the area units are defined incorrectly on the measurement parameters page, then the routine will certainly calculate the grid cell-specific threshold distances wrongly. For example, if data are in feet but the area on the measurement parameters page are defined in square miles, most likely the routine will not find any points that are farther apart than any of the grid cell threshold distances since each distance will be defined in miles. In other words, it is essential that the area units be consistent with the data for the routine to properly work.

Use kernel bandwidths that produce stable estimates

Second, the bandwidth for the baseline variable must be defined in such a way as to produce a stable density estimate of the variable. Be careful about choosing a very small bandwidth. This could have the effect of creating clusters at the edges of the study area or very large clusters in low population density areas. For example, in low population density areas, there will probably be fewer persons or events than in more built-up areas. This will have the effect on the Rnnh calculation of producing a very large matching distance. Points that are quite far apart could be artificially grouped together, producing a very large cluster. Using a larger bandwidth will usually produce a more stable average.

The process is a little like tuning a shortwave radio, adjusting the dial until the signal is detected. We suggest that the user first develop a good density model for the baseline variable (see Chapter 10). The user has to develop a trade-off between identify areas of high and low population concentration to produce an estimate that is statistical reliable (stable).

One can think of two types of 'fine tuning' that must occur. One is the 'background' variation that has to be tuned (the baseline 'at risk' variable). This is done through the kernel density interpolation. If too narrow a bandwidth is selected, the density surface will have numerous undulations with small 'peaks' and 'valleys'; this could produce unreal and unstable risk estimates. A grid cell with a very small density value could produce an extremely large threshold distance whereas a grid cell with a very low density could produce an extremely small threshold distance. Conversely, if too large a bandwidth is selected, the density surface will not differentiate very well and each grid cell will have, more or less, the same threshold distance. In this case, the Rnnh routine would yield a result not very different from the Nnh routine.

Another is the tuning of the clusters themselves through the threshold adjustment and minimum size criteria. If a large threshold probability is selected, too many incidents may be grouped; conversely, if a small threshold probability is selected, the result may be too restrictive. Similarly, if a small minimum sample size for clusters is used, there could be too many clusters whereas the opposite will happen if a large minimum sample size is chosen (i.e., zero clusters). The user must experiment with both these types of adjustment to produce a sensible cluster solution that captures the areas of high risk, but no more.

Example 2: Simulated Rnnh Clustering

To illustrate the logic of the Rnnh routine, a simulated example is presented. Two hundred points (incidents) were assigned to eight groups in the Baltimore metropolitan region (Figure 7.10). The figure shows the points in relation to year 2000 population density. Each group contained 25 individual points that were grouped exactly the same. However, three of the groups were placed in more dense areas of the region - one in central Baltimore, one in Towson to the north, and one is Reisterstown to the north east. The other five groups were placed in less populated areas. The Nnh and Rnnh routines were compared with these data. One would expect the Nnh routine to cluster the 200 points into eight groups whereas the Rnnh routine should identify only five groups in the low density areas. The reason for three of the groups not being clustered by the Rnnh is due to their higher population densities; all other things being equal, there should be more incidents in higher density areas than in lower density areas. Figures 7.11 and 7.12 show exactly this solution.

In other words, the Nnh routine clustered the points together irrespective of the distribution of the baseline population whereas the Rnnh routine clustered the points together relative to the baseline population (in this case, population). The specific parameter used were the default threshold distance (random nearest neighbor distance), a minimum of 15 points per cluster, and, for the Rnnh parameters, a normal kernel with a fixed interval of 0.5 miles.

Figure 7.10:

Incidents in Relation to Population Density

Baltimore: 1990

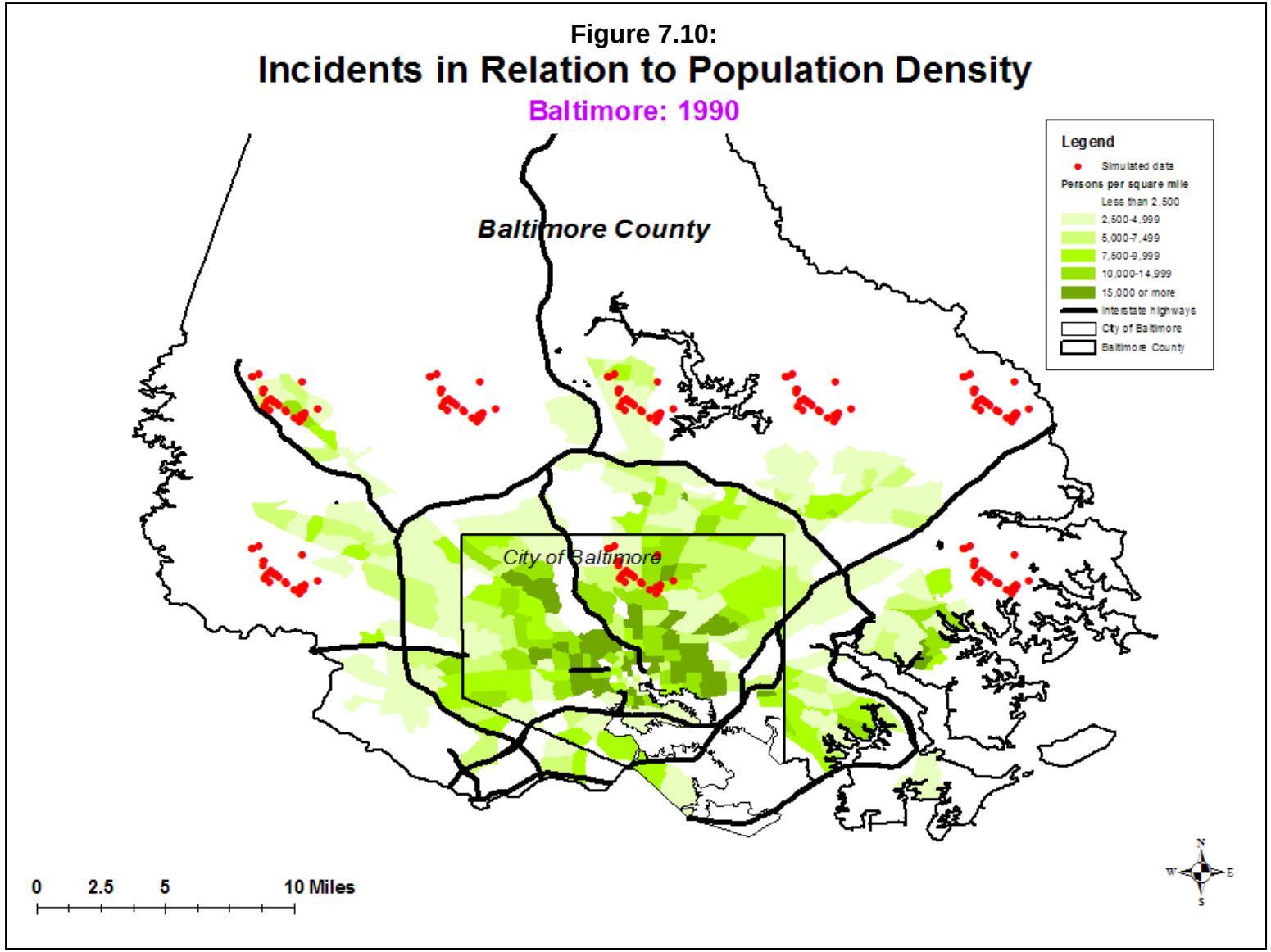


Figure 7.11:

Incidents in Relation to Population Density

Baltimore: 1990

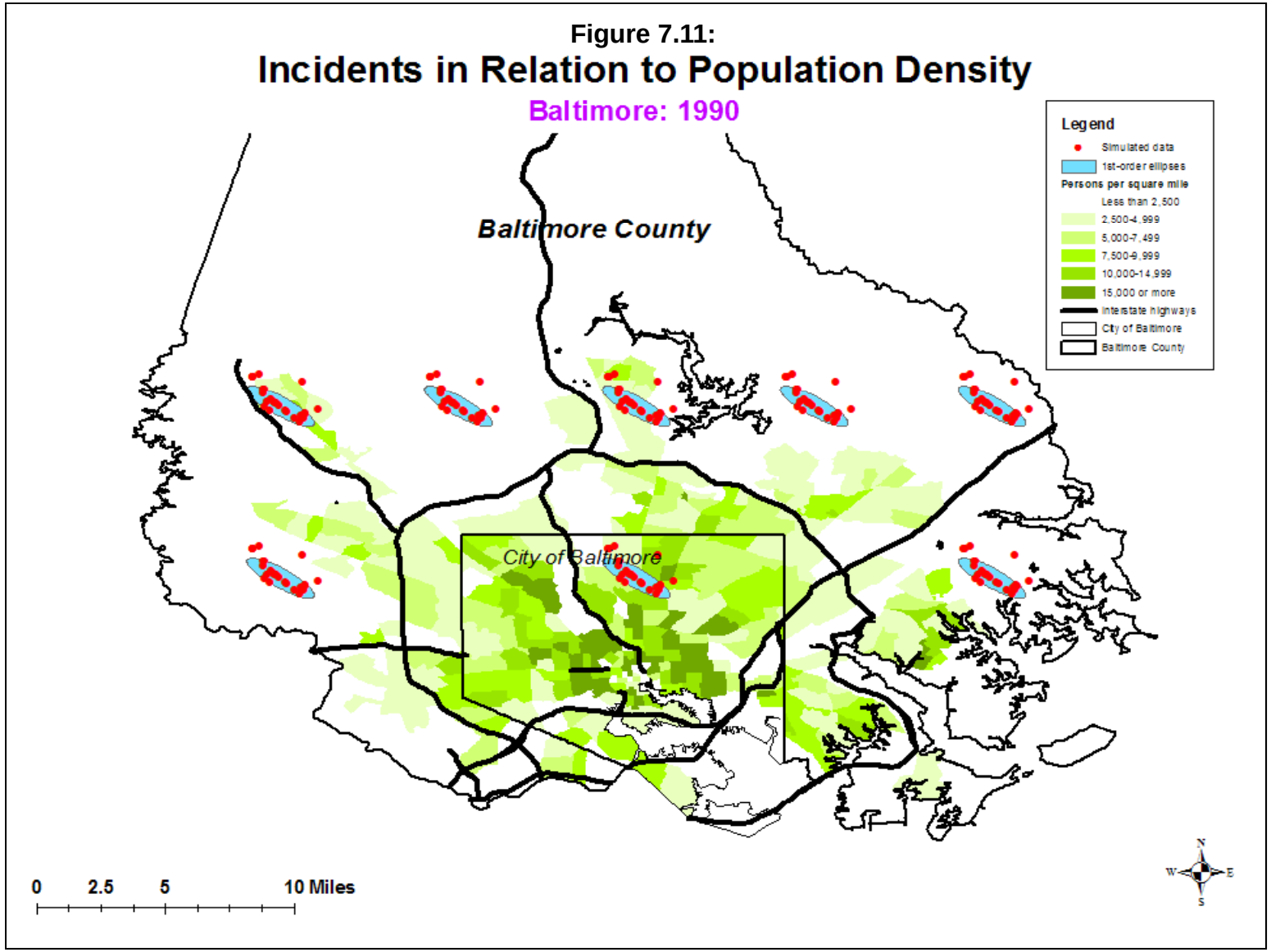
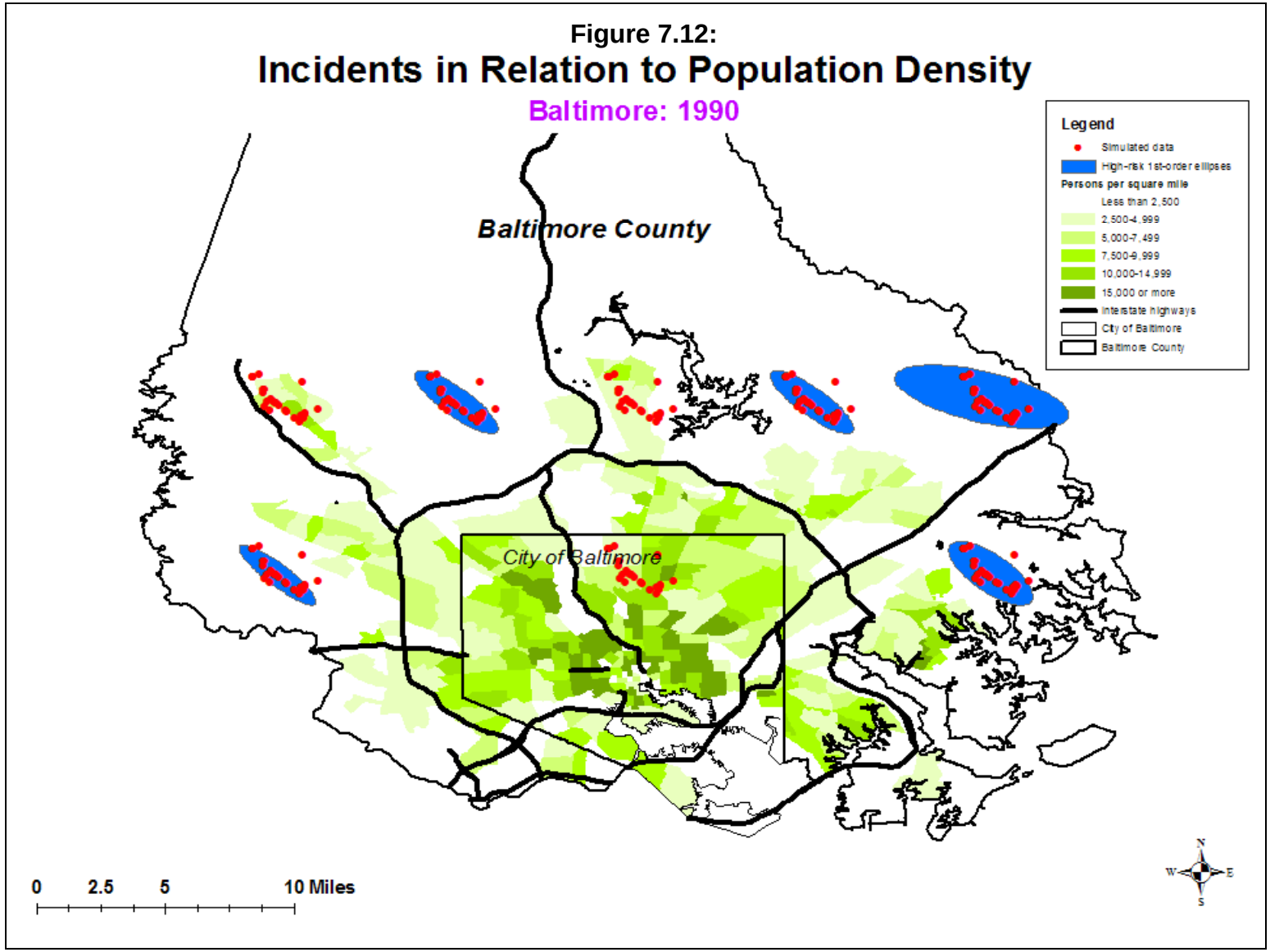


Figure 7.12:

Incidents in Relation to Population Density

Baltimore: 1990



Rnnh Output Files

The output files are similar to the Nnh routine. The Rnnh routine has three outputs. First, final seed locations of each cluster and the parameters of the selected standard deviational ellipse are calculated for each cluster. These can be output to a '.dbf' file or saved as a text ('.txt') file. Only 45 of the seed locations are displayed on the screen. The user can scroll down or across by adjusting the horizontal and vertical slider bars and clicking on the *Go* button.

Second, for each order that is calculated, *CrimeStat* calculates the mean center of the cluster. This can be saved as a '.dbf' file. Third, either standard deviational ellipses or convex hulls of the clusters can be saved in in *ArcGIS* '.shp', *MapInfo* '.mif', *Google Earth* 'kml' (if the coordinates are spherical), or various Ascii formats. Again, the convex hulls display polygons around the incidents whereas the ellipses are determined by the number of standard deviations to be calculated (see above). For small geographical area a 1X standard deviational ellipse may be appropriate since a 1.5X or 2X standard deviational ellipse can create an exaggerated view of the underlying cluster. On the other hand, for a regional view, a 1X standard deviational ellipse may not be very visible. The user has to balance the need to accurately display the cluster compared to making it easier for a viewer to understand its location.

As with the Nnh second- and higher-order clusters, these may cover incidents that were not clustered in the first-order. Thus, one has to be careful in interpreting second- and higher-order clusters. Essentially, these are abstractions made up of first-order clusters. In the routine, the first-order clusters are the primary clusters while the higher-order ones are ways to group the first-order clusters.

Naming conventions for ellipses

Because there are also orders of clusters (i.e., first-order, second-order, etc.), there is a naming convention that distinguishes the order.

For the ellipses, the convention is

Rnnh<O><*username*>

where *O* is the order number and *username* is a name provide by the user. Thus,

Rnnh1robbery

are the first-order clusters for a file called 'robbery' and

Rnnh2burglary

are the second-order clusters for a file called 'burglary'. Within files, clusters are named

Rnnh<O>Ell<N><*username*>

where *O* is the order number, *N* is the cluster number and *username* is the user-defined name of the file. Thus,

Rnnh1Ell10robbery

is the tenth cluster within the first-order clusters for the file 'robbery' while

Rnnh2Ell1burglary

is the first cluster within the second-order clusters for the file 'burglary'.

For the convex hulls, the cluster numbers are the same as the ellipses but the prefix name is output with a 'CRNNH1' prefix for the first-order clusters, a 'CRNNH2' prefix for the second-order clusters, and a 'CRNNH3' prefix for the third-order clusters. Higher-order clusters will index only the number.

Example 3: Rnnh Clustering of Vehicle Thefts

A second example is the clustering of 2003 San Antonio robberies relative to the 2000 population of census block groups. The test is for clusters of robberies that are more concentrated than would be expected on the basis of the population distribution.[5] Using the default threshold probabilities, a minimum sample size per cluster of 10, but a normal kernel function with a 0.5 mile fixed bandwidth, the Rnnh routine identified five first-order and one second-order cluster (Figure 7.13); the incidents are not shown.

Compare this distribution with the results of the Nnh on the same data, using the same parameters (Figure 7.14). The Nnh found 9 first-order clusters and one second-order cluster. To illustrate the differences in the baseline population, the ellipses of both the regular (Nnh) and risk-adjusted (Rnnh) clusters are overlaid on top of 2000 population density of census block groups. The cluster locations where there are both high volume (Nnh) and high-risk (Rnnh) involve two areas of low population density (just north of downtown) and one area of high

5 It is not an exact risk test since we are comparing 2003 robberies with 2000 population. It is an approximate risk test.

Figure 7.13:

San Antonio Robbery Risk: 2003

Ellipses of 1st-order and 2nd-order Risk-adjusted Hot Spots

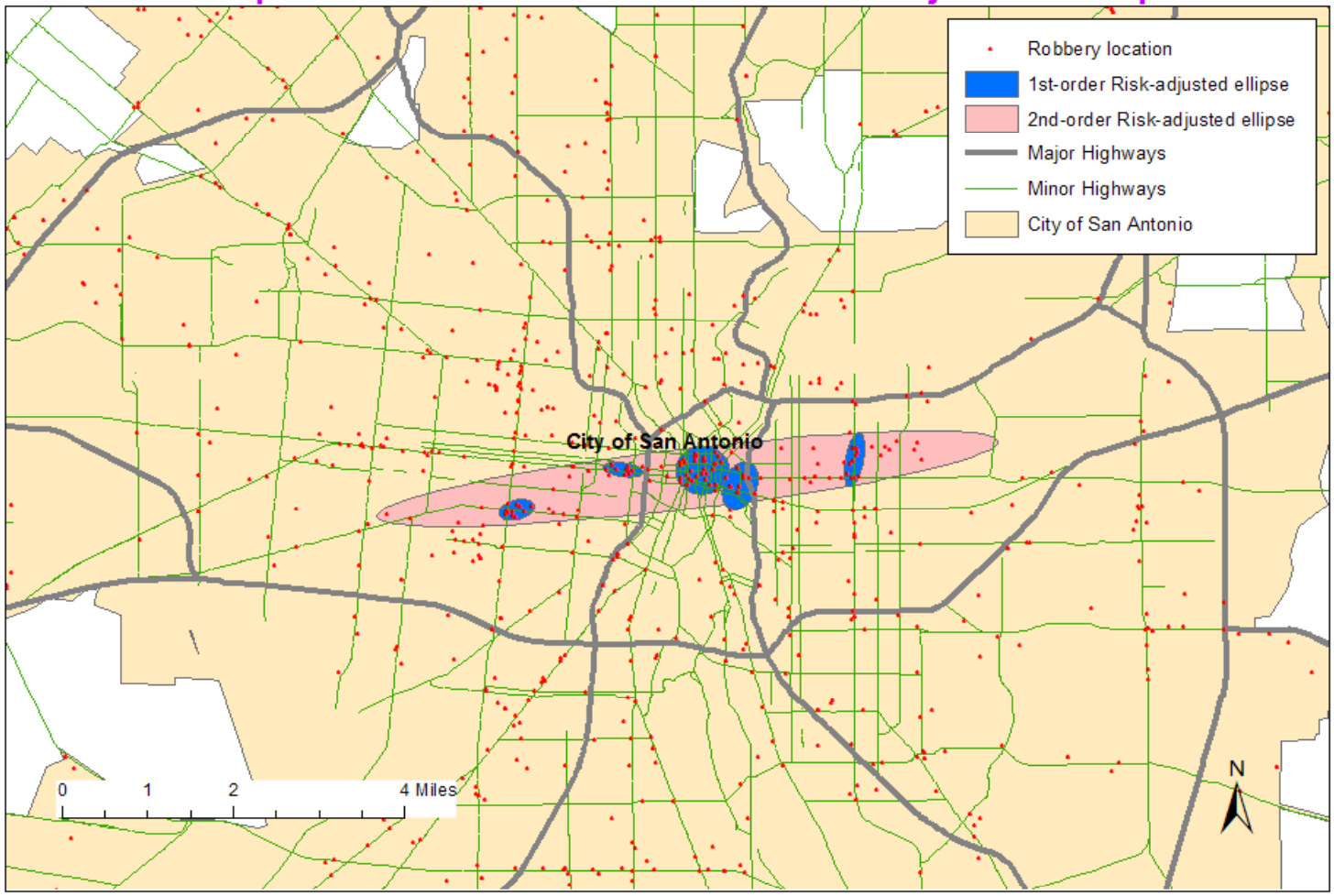
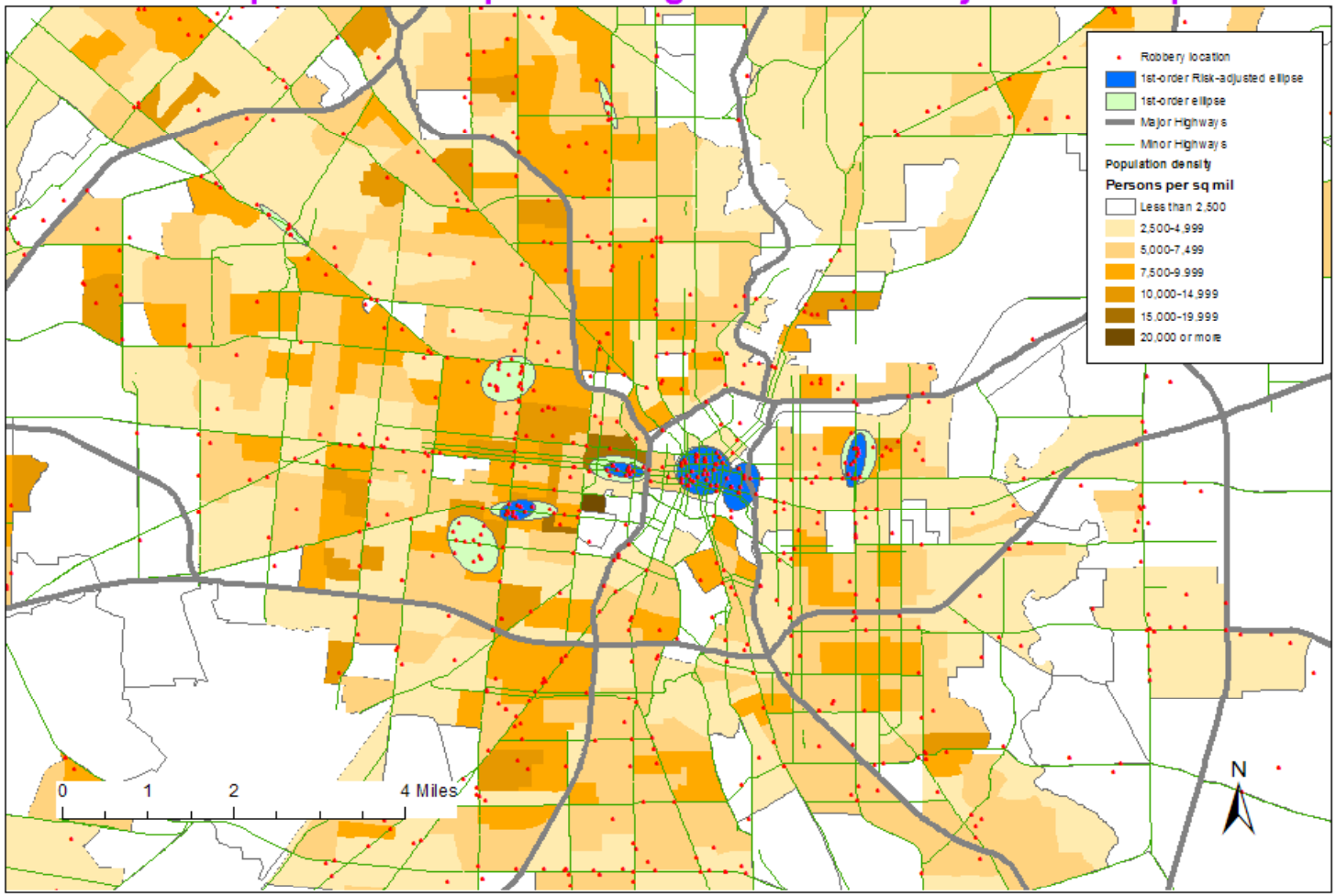


Figure 7.14:

San Antonio Robbery Risk: 2003

Comparison of Ellipses of Regular and Risk-adjusted Hot Spots



population density (just outside the downtown area); in this latter case, the number of robberies is so high that the area is both high volume and high risk. The fifth overlapping cluster is to the west of downtown and is an area of moderate population density. On the other hand, the four regular cluster locations that are only high volume (Nnh only) are in areas of low to moderate population density. In other words, the Rnnh routine identified areas of high *risk* for robberies whereas the Nnh routine identified areas of high *volume*.

Simulating Statistical Significance

Because the sampling distribution of the clustering method is not known, the Rnnh routine allows Monte Carlo simulations to approximate confidence intervals, similar to the Nnh routine (Dwass, 1957; Barnard, 1963). The output is identical to the Nnh routine. Essentially, it produces credible intervals for the number of first-order clusters, the area of clusters, the number of points in each cluster, and the density of each cluster. Second- and higher-order clusters are not simulated since their structure depends on the first-order clusters. The user can see whether the first-order cluster structure is different than that which is produced by a random distribution. See the notes above under Nnh for more details.

Uses of the Technique

The risk-adjusted nearest neighbor hierarchical clustering routine has several uses. First, like the high volume nearest neighbor hierarchical clustering (Nnh) routine, it allows a hierarchy of clusters to be identified, from first-order to second- or higher-order. As we see repeatedly with population dynamics, spatial clusters are frequently clustered together. One can think of them as small zones of concentrated events that are, in turn, close to other zones of concentrated events.

Second, unlike the Nnh, the Rnnh routine allows these clusters to be defined in terms of risk. Thus, it controls for the predominance of the *population at risk*. This is particularly important in epidemiological studies where the number of disease incidents is always related to the population at risk. The risk indicates a location where there are factors that are causing the disease to erupt. But, in crime analysis, too, analyzing incidents in relation to the number of potential victims can indicate problem neighborhoods where additional factors are triggering the outbreak (e.g., particular land uses that encourage disorder such as bars or pawn shops; poor social cohestion). Crime prevention efforts, in particular, often target neighborhoods of high risk and not just high volume of incidents. The Rnnh can be a valuable tool in the identification of such neighborhoods.

Third, the Rnnh routine goes beyond simply clustering events on the basis of proximity and frequency and applies a single variable that can account for the distribution. In other words,

the baseline variable is the first step in developing a model for explaining the distribution of the incidents, in this case the baseline variable itself. In addition to focusing policing efforts on high volume or high risk neighborhoods, there needs to be an effort to build a statistical model of the phenomenon itself, both for prediction as well as for theory development.

Limitations of the Technique

However, as with all methods, there are some limitations of the technique that are partly shared with the Nnh routine. First, the method only clusters incidents (points); a weighting or intensity variable will have no effect. In Chapter 9, we will introduce a zonal variant of the Rnnh that allows a risk measure to be applied to zonal data. But, the Rnnh by itself is only applicable to individual point locations.

Second, the size of the grouping area is dependent on the sample size if the confidence interval around the mean random distance is used as the threshold distance criteria. However, since the threshold distance is adjusted dynamically, this has less effect than in the Nnh since it is now a relative comparison rather than an absolute distance.

Third, there is arbitrariness in the technique due to the minimum points rule. Different users could define the minimum differently, which could lead to different conclusions about the location of high risk clusters. Finally, unique to the Rnnh, the method requires both an incident file (the primary file) and a baseline file (the secondary file.

Nevertheless, the Rnnh routine is a useful technique for identifying clusters that are more concentrated than would be expected on the basis of the population distribution.

References

Anselin, L. (1995). Local indicators of spatial association - LISA. *Geographical Analysis*. 27, No. 2 (April), 93-115.

Bailey, T. C. & Gatrell, A. C. (1995). *Interactive Spatial Data Analysis*. Longman Scientific & Technical: Burnt Mill, Essex, England.

Ball, G. H. & Hall, D. J. (1970). A clustering technique for summarizing multivariate data. *Behavioral Science*, 12, 153-155.

Barnard, G. A. (1963). Comment on 'The Spectral Analysis of Point Processes' by M. S. Bartlett, *Journal of the Royal Statistical Society*, Series B, 25, 294.

Beale, E. M. L. (1969). *Cluster Analysis.* Scientific Control Systems: London.

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press: New York.

Block, C. R. (1994). STAC hot spot areas: a statistical tool for law enforcement decisions. In *Proceedings of the Workshop on Crime Analysis Through Computer Mapping*. Criminal Justice Information Authority: Chicago, IL.

Block, R. & Block, C. R. (1999) Risky places: a comparison of the environs of rapid transit stations in Chicago and the Bronx in John Mollenkopf (ed), *Analyzing Crime Patterns: Frontiers of Practice*, Sage Publishing: Beverly Hills, CA.

Block, R. & Block, C. R. (1995). Space, place and crime: hot spot areas and hot places of liquor-related Crime in John E. Eck & David Weisburd (eds.), *Crime and Place*. Crime Prevention Studies, Volume 4. Criminal Justice Press: Monsey, NY. 147-185.

Braga, A. & Weisburd, D. (2010). Policing Problem Places: Crime Hot Spots and Effective Prevention. Oxford: Oxford University Press.

Can, A. & Megbolugbe, I. (1996). The geography of underserved mortgage markets. Paper presented at the American Real Estate and Urban Economics Association meeting. May.

Carmichael, J. W., George, L.A. & Julius, R.S. (1968). Finding natural clusters. *Systematic Zoology*, 17, 144-150.

References (continued)

Cattell, R. B. & Coulter, M.A. (1966). Principles of behavioural taxonomy and the mathematical basis of the taxonome computer program. *British Journal of Mathematical and Statistical Psychology*, 19, 237-269.

Chainey, S., Thompson, L. & Uhlig, S. (2008). The utility of hotspot mapping for predicting spatial patterns of crime. *Security Journal*, **21**, 4-28.

Cole, A. J. & Wishart, D. (1970). An improved algorithm for the Jardine-Sibson method of generating overlapping clusters. *Comparative Journal*, 13, 156-163.

D'andrade, R. (1978). U-Statistic Hierarchical Clustering *Psychometrika*, 4,58-67.

Dwass, M (1957). Modified randomization tests for nonparametric hypotheses. *Annals of Mathematical Statistics*, 28, 181-187.

Everitt, B. S. (2011). *Cluster Analysis* (5th edition). J. Wiley: London.

Everitt, B. S., Landau, S. & Leese, M. (2001). *Cluster Analysis*. 4th Edition. Oxford University Press: New York.

Getis, A. & Ord, J. K. (1996). Local spatial statistics: an overview. In Longley, P. & Batty, M. (eds), *Spatial Analysis: Modelling in a GIS Environment*. GeoInformation International: Cambridge, England, 261-277.

Gitman, I. & Levine, M. D. (1970). An algorithm for detecting uniomodal fuzzy sets and its application as a clustering technique. *IEE Transactions on Computers*, 19, 583-593.

Hartigan, J. A. (1975). *Clustering Algorithms*. John Wiley & Sons, Inc.: New York.

Jardine, N. & Sibson, R. (1968). The construction of hierarchic and non-hierarchic classifications. *Comparative Journal*, 11, 117-184.

Jefferis, E. (1998). A multi-method exploration of crime hot spots. Crime Mapping Research Center, National Institute of Justice: Washington, DC.

Johnson, S. C. (1967), Hierarchical Clustering Schemes *Psychometrika*, 2,241-254.

References (continued)

Jones, K. S. & Jackson, D. M. (1967). Current approaches to classification and clump finding at the Cambridge Language Research Unit. *Comparative Journal*, 10, 29-37.

King, B. F. (1967). Step wise clustering procedures. *Journal of the American Statistical Association*. 62, 86-101.

Kulldorff, M. (1997). A spatial scan statistic, *Communications in Statistics - Theory and Methods*, 26, 1481-1496.

Kulldorff, M. & Nagarwalla, N. (1995). Spatial disease clusters: Detection and inference, *Statistics in Medicine*, 14, 799-810.

Levine, N. (2008). "The 'hottest' part of a crime hotspot: Comments on "The utility of hotspot mapping for predicting spatial patterns of crime" by Chainey, S. Thompson, L. & Uhlig, S.". *Security Journal*, 21, 295-302.

Levine, N., Wachs, M. & Shirazi, E. (1986). "Crime at Bus Stops: A Study of Environmental Factors". *Journal of Architectural and Planning Research.* 3 (4), 339-361.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *5th Berkeley Symposium on Mathematics, Statistics and Probability*. Vol 1, 281-298.

McBratney, A. B. & deBruijter, J. J. (1992). A continuum approach to soil classification by modified fuzzy k-means with extragrades, *Journal of Soil Science*, 43, 159-175.

McQuitty, L. L. (1960). Hierarchical syndrome analysis. *Educational and Psychological Measurement*, 20, 293-304.

Maltz, M. D., Gordon, A. C., & Friedman, W. (1990). *Mapping Crime in Its Community Setting: Event Geography Analysis*. Springer-Verlag: New York.

Needham, R. M. (1967). Automatic classification in linguistics. *The Statistician*, 17, 45-54.

Openshaw, S. A., Craft, A. W., Charlton, M., & Birch, J. M. (1988). Invetigation of leukemia clusters by use of a geographical analysis machine, *Lancet*, 1, 272-273.

References (continued)

Openshaw, S. A., Charlton, M, Wymer, C, & Craft, A. (1987). A Mark 1 geographical analysis machine for the automated analysis of point data sets. *International Journal of Geographical Information Systems*, 1, 335-358.

Sherman, L. W. & Weisburd, D. (1995). General deterrent effects of police patrol in crime hot spots: a randomized controlled trial. *Justice Quarterly*. 12, 625-648.

Sherman, L. W.., Gartin, P. R. & Buerger, M. E. (1989). Hot spots of predatory crime: routine activities and the criminology of place. *Criminology*, 27(1), 27-56.

Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman & Hall: London.

Sneath, P. H. A. (1957). The application of computers to taxonomy. *Journal of General Microbiology*, 17, 201-226.

Sokal, R. R. & Sneath, P. H. A. (1963). *Principles of Numerical Taxonomy*. W. H. Freeman & Co.: San Francisco.

Sokal, R. R. & Michener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38, 1409-1438.

Systat, Inc. (2008). *Systat 13: Statistics I*. SPSS, Inc.: Chicago.

Thorndike, R. L. (1953). Who belongs in a family?. *Psychometrika*, 18, 267-276.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*. 58, 236-244.

Weisburd, D. & Green, L. (1995). Policing drug hot spots: the Jersey City drug market analysis experiment. *Justice Quarterly*. 12 (4), 711-735.

Weisburd, D., Maher, L.& Sherman, L. (1992). Contrasting crime general and crime specific theory: the case of hot-spots of crime. *Advances in Criminological Theory*, 4, 45-70.

Weishart, D. (1969). Mode analysis. In Cole, A. J. (ed), *Numerical Taxonomy*, Academic Press: New York.

References (continued)

Xie, X. L. & Beni, G. (1991). A validity measure for fuzzy clustering. IEEE Trans. Pattern Analysis Machine Intell., 13, 841-847.

Endnotes

i. The particular steps are as follows:

1. All distances between pairs of points are calculated, using either direct or indirect distance as defined on the measurements parameters page. The matrix is assumed to be symmetrical, that is the distance between A and B is assumed to be identical to the distance between B and A.

2. The mean expected random distance is calculated using formula 6.2 and the threshold distance (the confidence interval for the corresponding t) is calculated using formulas 7.2 and 7.3 depending on whether it is a lower or upper confidence interval. The particular interval is selected by the user on the slide bar.

3. All pairs that are separated by a distance smaller than the threshold distance are selected for clustering and placed in a *reduced matrix*. Any incident point that does not have another point within the threshold distance is not clustered.

4. In the reduced matrix, for each point the number of other points that are within the threshold distance are counted and are sorted in descending order.

5. The incident point with the largest number of below threshold distances is selected for the initial seed of the first cluster.

6. All other points that are within the threshold distance of the initial seed point are selected for the initial cluster 1 and temporarily removed from the reduced matrix.

7. The process is repeated for the remaining points in the reduced matrix (i.e., an initial seed is selected, all points within the threshold distance of that seed are clustered, and all the points are temporarily removed).

8. For each of the initial clusters that were identified, the center of minimum distance (CMD) is calculated to identify the cluster center.

9. The clustering process is repeated but using the CMD for each cluster to define each cluster. This process continues until no points change their cluster membership.

10. Once all the points in the reduced matrix have been initially clustered, the total number of points within each initial cluster is counted. If the number is equal to or greater than the minimum specified, then the cluster is kept. If the number is less than the minimum specified, then the cluster is dropped.

11. The final clusters are sorted in descending order of the number of points and the mean center of each is calculated to identify the cluster center.

12. The second- and higher-order clusters use the CMD of the first-order clusters as 'points' and follow the same algorithm.

ii. The particular steps are as follows:

1. Using the same p-values selected in the first-order, the mean random expected distance is calculated. However, the sample size is the number of first-order clusters identified, not the original number of points. Thus, the threshold distance is calculated by

$$Second - order\ threshold\ distance = d_{NN2(ran)} + t * SE_{d1(ran)} \qquad (7.8)$$

where $d_{NN2(ran)}$ is random nearest neighbor distance among the first-order clusters (i.e., with *M* first-order clusters rather than *N* points) and $SE_{d1(ran)}$ is the standard error of the random nearest neighbor distance among the first-order clusters. Thus, there is a different threshold distance for the second-order clustering. The t-value specified in the first-order clustering is maintained for second- and higher-order clustering.

2. All distances between first-order cluster centers are calculated and only those that are smaller than the second-order threshold distance are selected for second-order clustering.

3. If there are no distances between first-order cluster centers that are smaller than the second-order threshold distance, then the clustering process ends.

Endnotes (continued)

4. If there are distances between first-order cluster centers that are smaller than the second-order threshold distance, then the steps specified in endnote *i* above are repeated to produce second-order clusters. A minimum of four first-order clusters is required to allow a second-order cluster and four previous-order clusters to allow a higher-order cluster.

5. If there are second-order clusters, then this process is repeated to either extract third-order clusters or to end the clustering process if no distances between second-order cluster centers are smaller than the (new) third-order threshold distance or if there are fewer than four new seeds in the cluster.

6. The process is repeated until no further clustering can be conducted, either all sub-clusters converge into a single cluster or the threshold distance criteria fails or there are fewer than four seeds in the higher-order cluster.

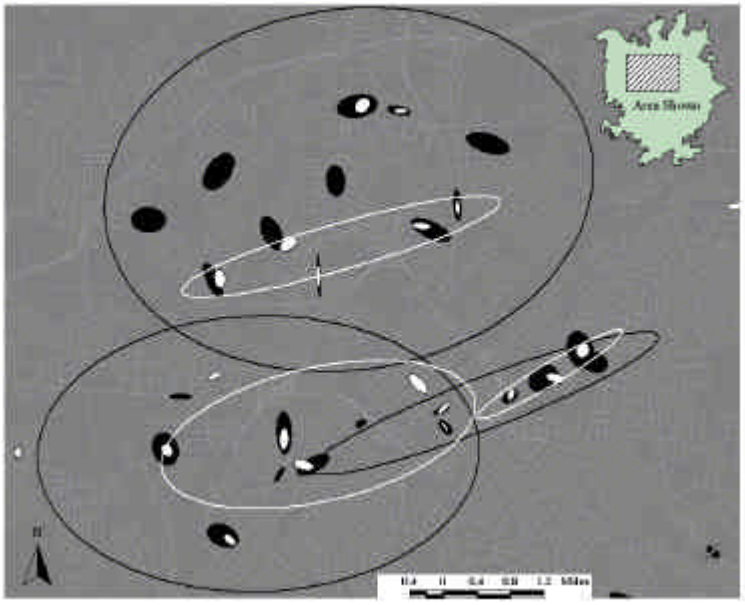
Attachments

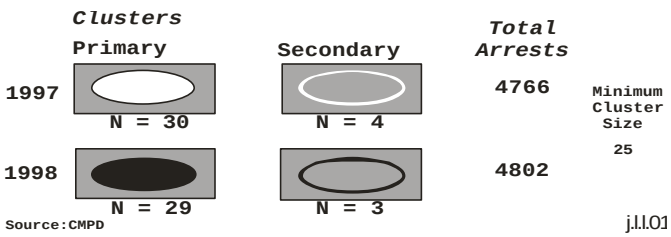
Visualizing Change in Drug Arrest Hot Spots Using Nearest Neighbor Hierarchical Clustering: Charlotte, N.C. 1997 – 98

James L. LeBeau
Administration of Justice
Southern Illinois University at Carbondale

Stephen Schnebly
Criminology & Criminal Justice
University of Missouri – St Louis

The *CrimeStat* Nearest Neighbor Hierarchical clustering routine and GIS were used for defining, comparing, analyzing, and visualizing changes in drug arrest clusters between 1997 and 1998. Using a minimum cluster size of 25 arrests some of the emerging patterns or relationships include: 1) the overlapping of secondary clusters, but those emerging during 1998 were much larger, especially in the north because of new primary clusters; 2) many primary clusters during 1997 remaining static or increasing in area during 1998; and 3) the disappearing of some 1997 primary clusters during 1998, with new clusters emerging close by implying displacement.



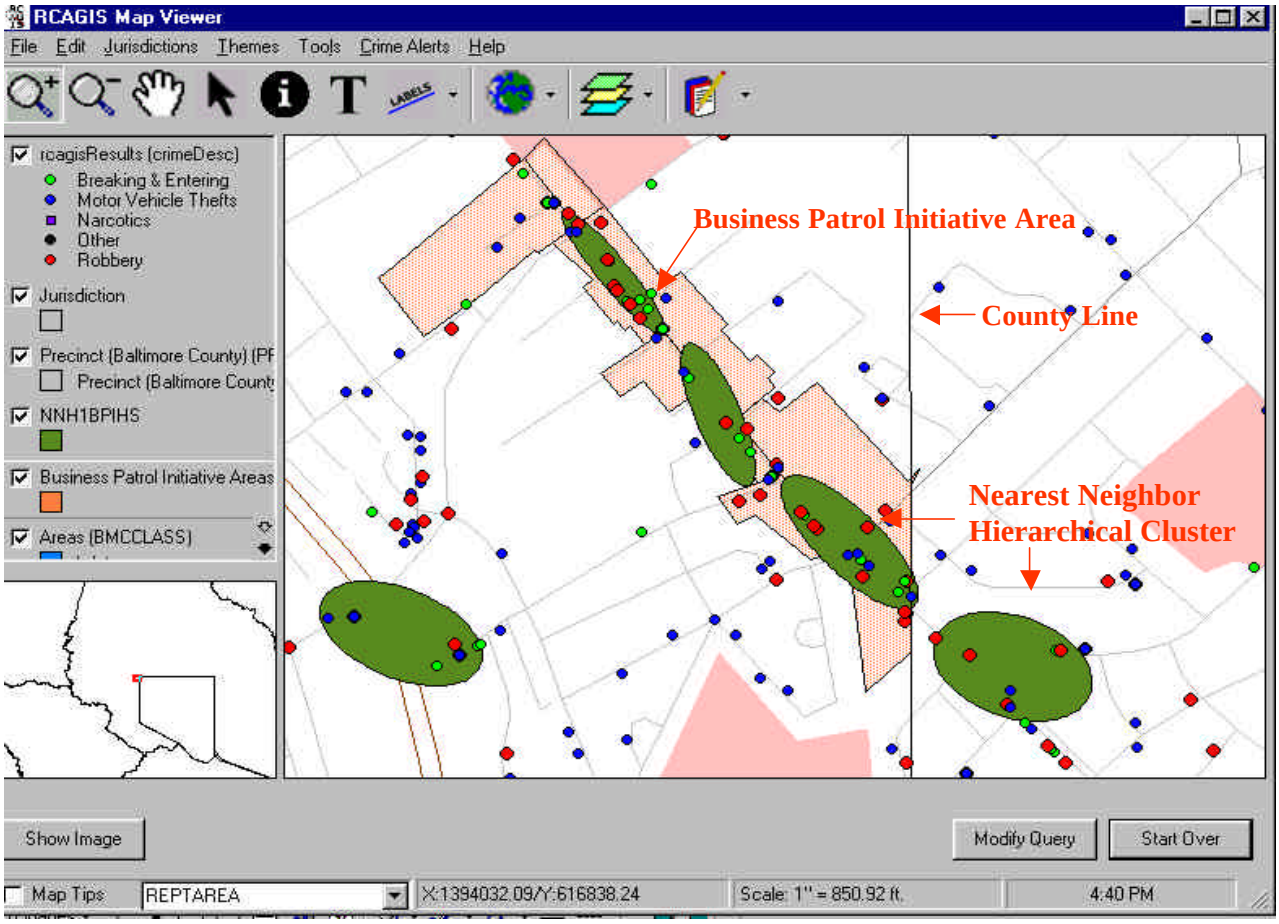


Using Nearest Neighbor Hierarchical Clustering to Identify High Crime Areas Along Commercial Corridors

Philip R. Canter
Baltimore County Police Department
Towson, Maryland

Robberies in Baltimore County had increased by 45% between 1990 and 199, and by 1997, were the highest on record. In 1997, 73% of all reported robberies in Baltimore County were occurring in commercial areas. The department wanted to target commercial districts with intensive patrol and outreach programs. These high crime commercial districts were identified as Business Patrol Initiative (BPI) areas. A total of 40 police officers working two 8-hour shifts were assigned to BPI areas. Robberies in the BPI areas declined by 26.7% during the first year of the program and another 13.8% one year following the BPI program.

Police analysts used *CrimeStat*'s Nearest Neighbor Hierarchical clustering (Nnh) method to identify high crime areas along commercial corridors. The Nnh routine was very effective in identifying commercial areas having the highest concentration of crime. The clustering also demonstrated that commercial crime was not restricted to county borders; rather, crime crossed municipal boundaries into neighboring jurisdictions. A neighboring jurisdiction was shown the crime cluster map, leading to their decision to implement a similar BPI program.

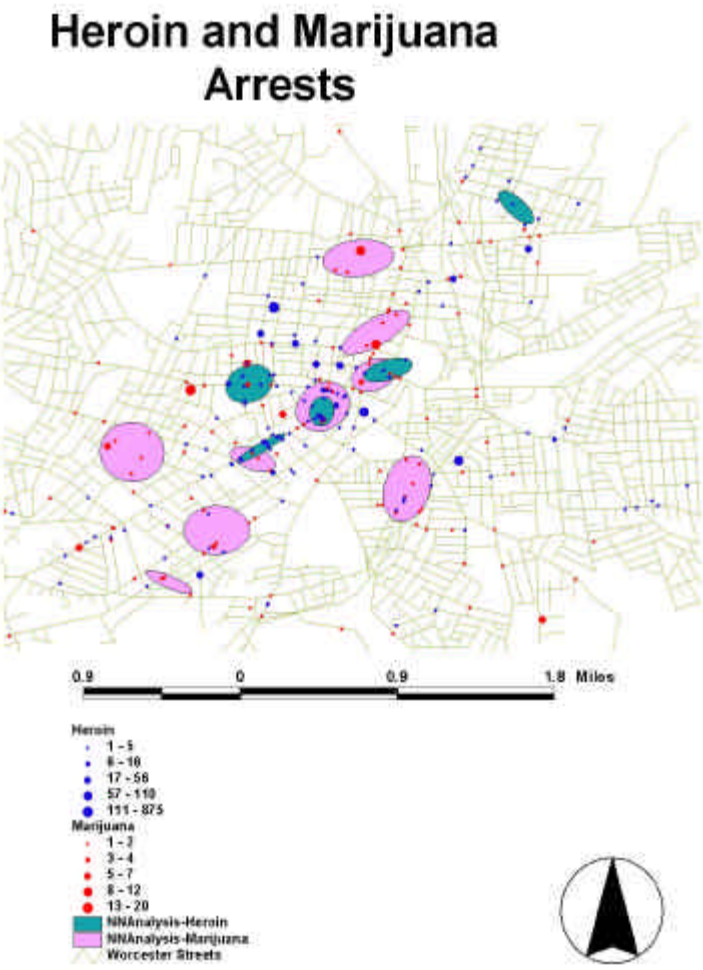


Arrest Locations as a Means for Directing Resources

Daniel Bibel
Massachusetts State Police
Crime Reporting Unit
Framingham, Massachusetts

The Massachusetts State Police is collecting incident addresses as part of its state-level implementation of the FBI's National Incident Based Reporting System (NIBRS). They intend to develop a regional and statewide crime mapping and analysis program. As an example of the type of analysis that can be done with the enhanced NIBRS database, the State Police's Crime Reporting Unit analyzed year 2000 drug arrests for one city in the Commonwealth, focusing on arrests for possession of heroin and marijuana. The arrest locations were plotted, with the size of points proportionate to the amount of drugs seized. A nearest neighbor clustering analysis was done of the data. It indicates that, while there is some small amount of overlap, the arrest locations for the two drug types are generally different.

This type of analysis can be very useful for smaller police agencies that do not have the resources to conduct their own analysis of crime data. It may also prove useful for crime problems with cross-jurisdictional boundaries.

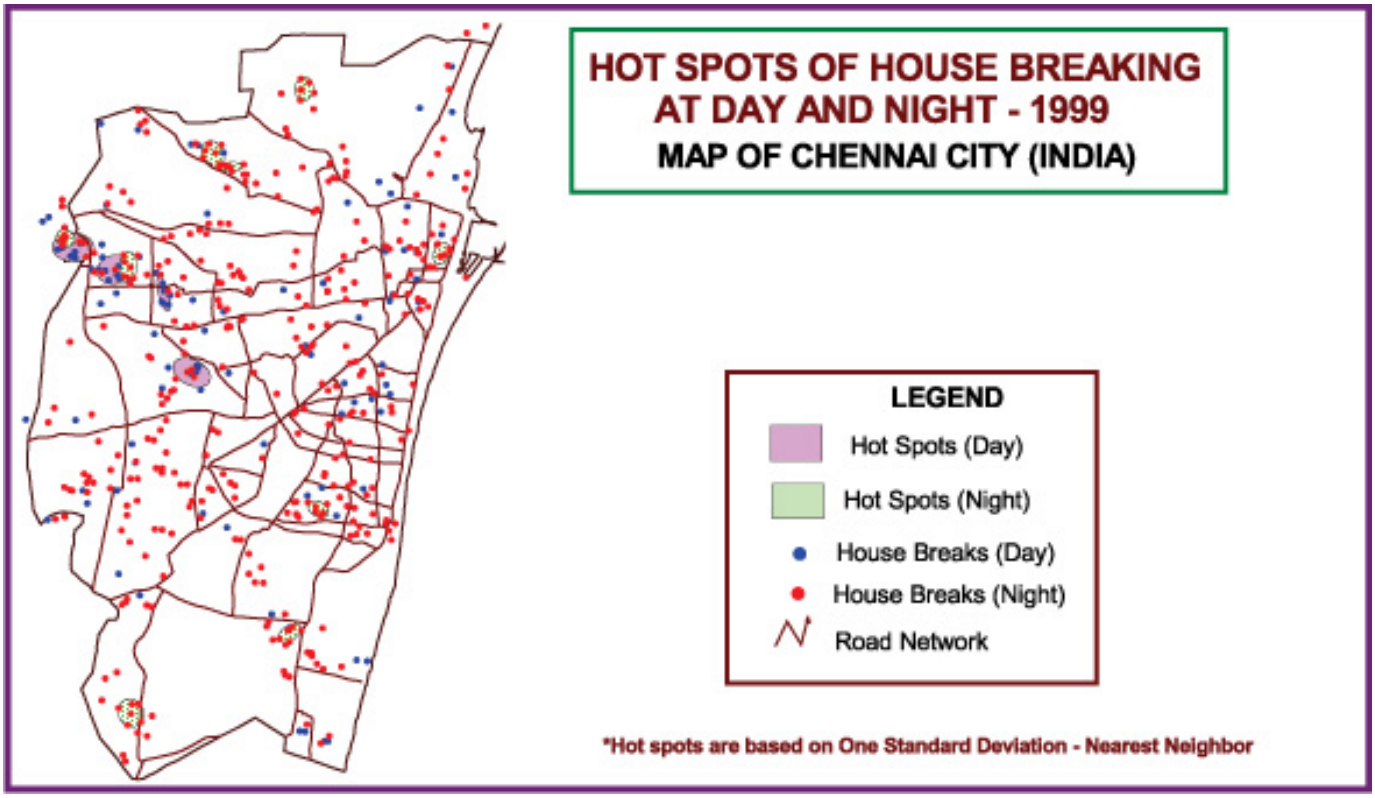


Use of *CrimeStat* in Crime Mapping in India:
An Application for Chennai City Policing

Jaishankar Karuppannan
Department of Criminology & Criminal Justice
Manonmaniam Sundaranar University
Tamil Nadu, India

The present study was done as an implementation of GIS technology in Chennai (Madras), India. In the present study hotspot analysis was done with the help of *CrimeStat*. We converted the output to *Arcview* shape files.

When hotspot analysis examined changes over a period of time, the change seemed to be significant. There exists not only a change in the location of the hotspots, but also in their areal extent. The numbers of hotspots also differ over time. The map shows hotspots for residential burglary for both day and night. The hot spots for daytime house break-ins are confined to a smaller area in the west of the city, whereas the hot spots for nighttime residential break-ins are seen in all parts of the city. In particular, the Posh area of Anna Nagar is more prone to daytime burglaries. In this area, a higher proportion of couples work, which appears to make the homes in this neighborhood more open for burglaries.



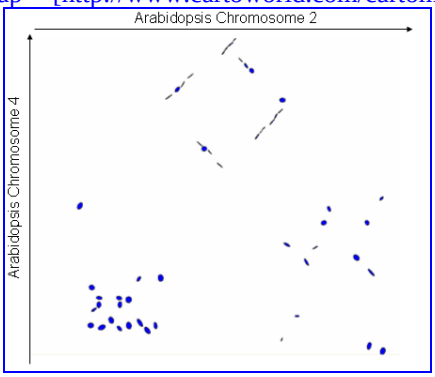
Identifying Duplications in Genomic Data
Using the *CrimeStat* Nearest Neighbor Hierarchical Spatial Clustering Routine

Nathalie Pavy and Jean Bousquet
Université Laval,G1K 7P4 Québec, QC, Canada, nathaliepavy@yahoo.fr

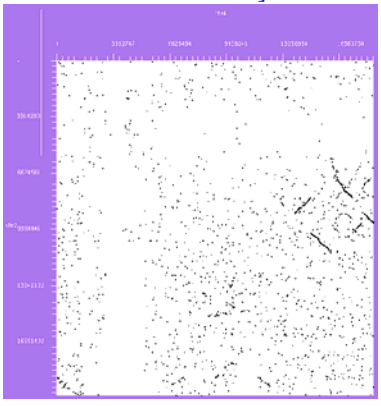
Sequencing projects provide the foundation for studying the organization of whole genomes. Comparisons of genomic sequences from related species provide a new insight into genome evolution for instance by showing locally conserved chromosomic segments. Detecting such conservation is far from trivial. Indeed, chromosome rearrangements, duplications and gene losses may hide traces of ancestry. The Nearest Neighbor Hierarchical Clustering routine (NNH) was applied to analyze regions duplicated between *Arabidopsis* chromosomes 2 and 4. These are well known for sharing similar series of genes derived from segmental duplication. Based on sequence similarities, each gene located on chromosome 2 was associated to one or several similar genes located on chromosome 4. Coordinates used as input for the NNH routine were the gene ranks along the chromosomes. A total of 53 clusters made of at least 6 similar genes were recovered. The significance of this finding was assessed with 1000 Monte Carlo simulations; only three clusters would be expected by chance alone (P>0.01). The gene clusters identified with the NNH approach were consistent with known duplicated chromosomic regions. The clusters found by using the NNH approach were vizualised with the GIS software CartoMap™. This graphical representation highlights in a visually comprehensive way the patterns of duplicated regions. The shape of the clusters and the relative positions of these reflect various evolutionary events that led to the structure of the present genome, as shown below (top-left): linear patterns indicate large segmental duplications with conserved gene order with or without inversion, and large dots indicate more condensed gene clusters.

Clusters of at least six genes found on *Arabidopsis* chromosome 2 and duplicated on chromosome 4.

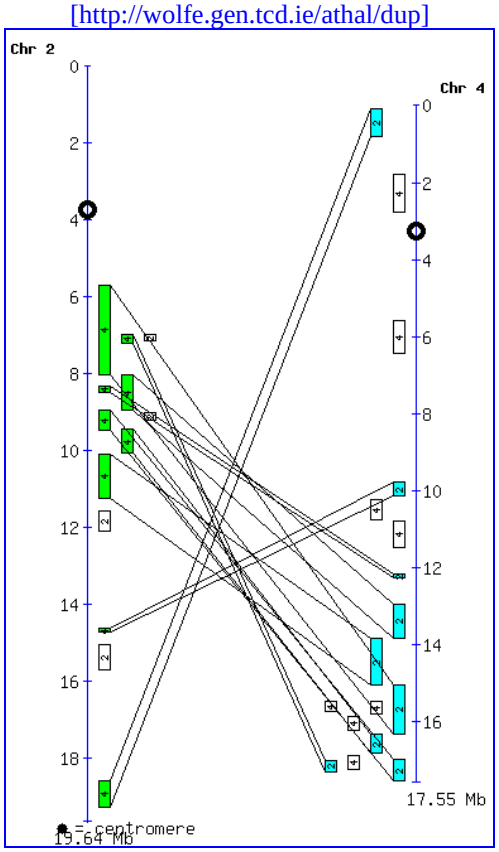
Clusters found with the NNH routine and visualized with CartoMap™ [http://www.cartoworld.com/cartomap.htm]



Dot-Plot obtained by using DAGchainer [http://dagchainer.sourceforge.net/] [Haas et al., 2004, Bioinformatics 20:3643-3646]



Clusters extracted from the Paralogon database [http://wolfe.gen.tcd.ie/athal/dup]

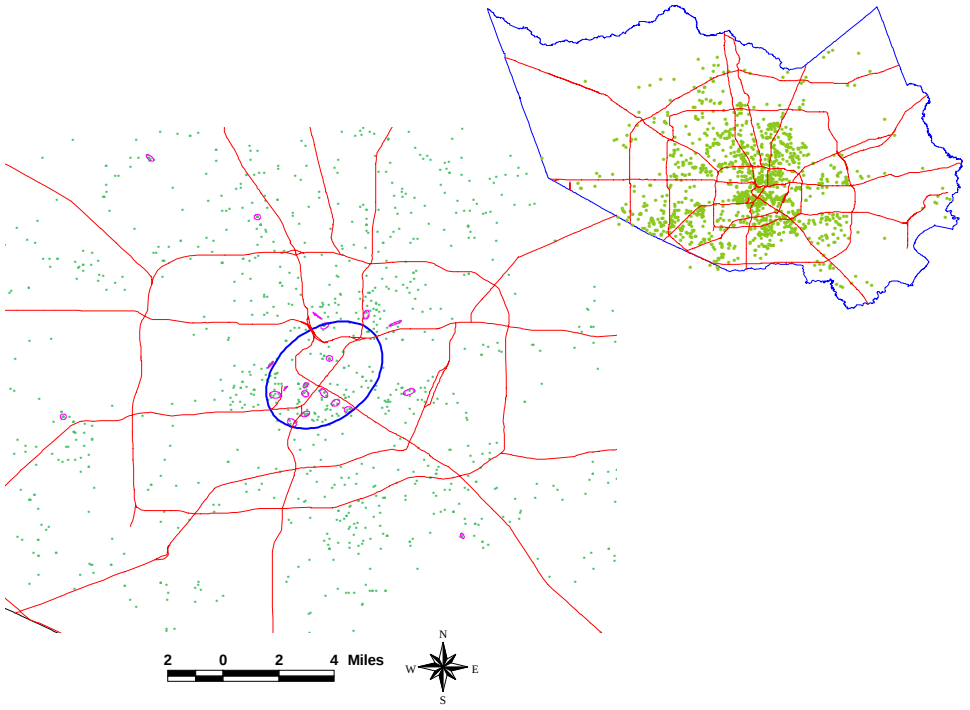


Risk Adjusted Nearest Neighbor Hierarchical Clustering of Tuberculosis Cases in Harris County, Texas: 1995 to 1998

Matthew L. Stone, MPH
Epidemiology and Program Evaluation Unit
University of California at San Francisco/California Department of Health Services
Sacramento, CA

Data was collected from an ongoing, population-based, active surveillance and molecular epidemiology study of tuberculosis cases reported to the City of Houston Tuberculosis Control Office from October 1995 to September 1998. During this time, 1774 cases of tuberculosis were reported and 1480 of those who participated in this study were successfully geocoded.

CrimeStat was used to make an initial survey of potential hot spot areas of tuberculosis cases where more focused TB control efforts could be implemented. Given a .05 level of significance for grouping a pair of points by chance and a minimum of five cases per cluster, 24 first-order clusters and one second-order cluster were detected after adjusting for the underlying population. Most first-order clusters were detected in the center of Harris County, including the metropolitan downtown area. By adjusting for the underlying population, the clusters identify areas with higher than average TB incidence. Some of these clusters are homeless shelters as many homeless persons are particularly prone to TB.



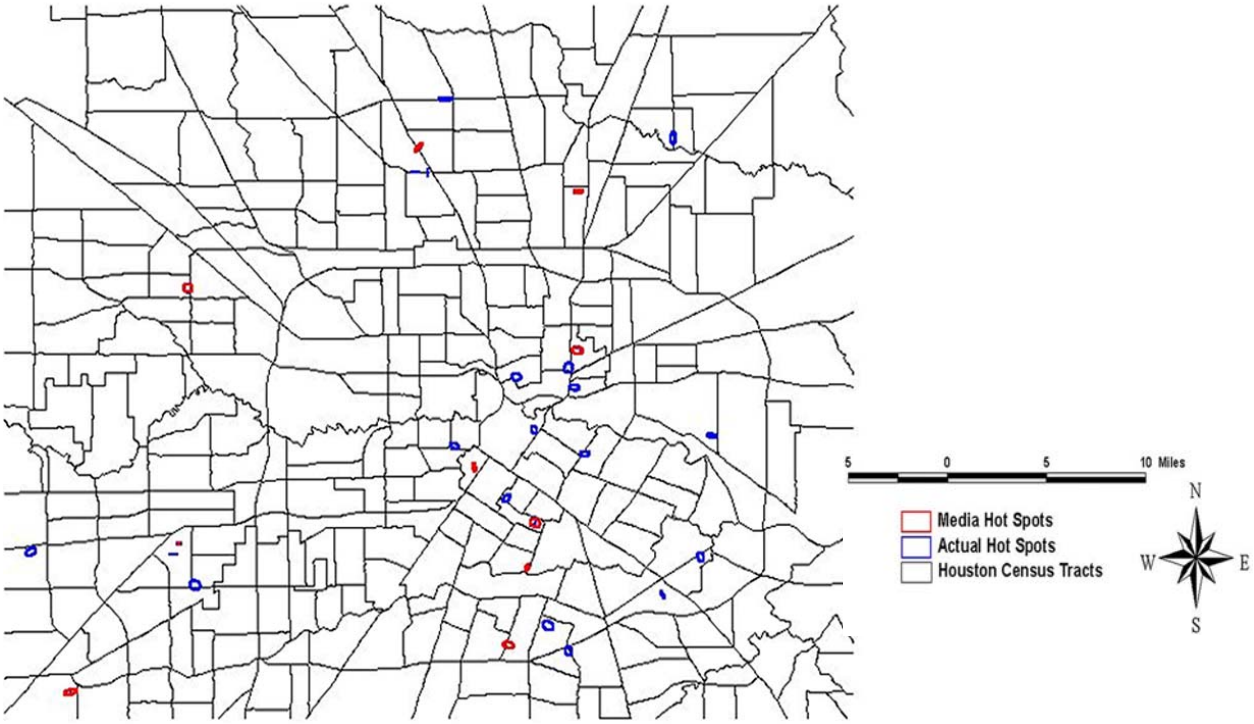
Using Risk Adjusted Nearest Neighbor Hierarchical Clustering to Compare Actual and Media Hotspots of Homicide

Derek J. Paulsen
Department of Criminal Justice and Police Studies
Eastern Kentucky University

Crimestat offers an excellent method for determining risk adjusted hot spots of crime incidents within a jurisdiction. Risk-adjusted nearest neighbor hierarchical spatial clustering (Rnnh) is a spatial clustering routine that groups points together based on both proximity to other points and the distribution of a baseline variable. In this example two different Rnnh analyses were conducted and compared for homicides in Houston, Texas. The first involves homicide incident locations adjusted for the population of each census tract, while the second involves incidents that were covered in the newspaper adjusted for the homicide rate of each census tract. The purpose of this analysis is to determine if there are differences in the spatial clustering of actual homicide incidents and those that are covered in the newspaper.

The preferences for the analysis were the same for both Rnnh analyses. For the primary file (homicide incidents & incidents covered in the newspaper) the pair probability search radius was set at .01, with a minimum of 10 points per cluster. For the secondary file (population & homicide rate), a quartic kernel density interpolation was used with an adaptive bandwidth and a minimum sample size of 100. Importantly, the analysis showed that media hot spots and actual hot spots do not coincide. Media coverage showed homicides to be concentrated in different areas than they are actually concentrated.

Actual Homicide Hot Spots vs. Media Coverage Hot Spots in Houston Texas



Seizures of Tiger Parts and Derivatives in India during 2000 – 2012

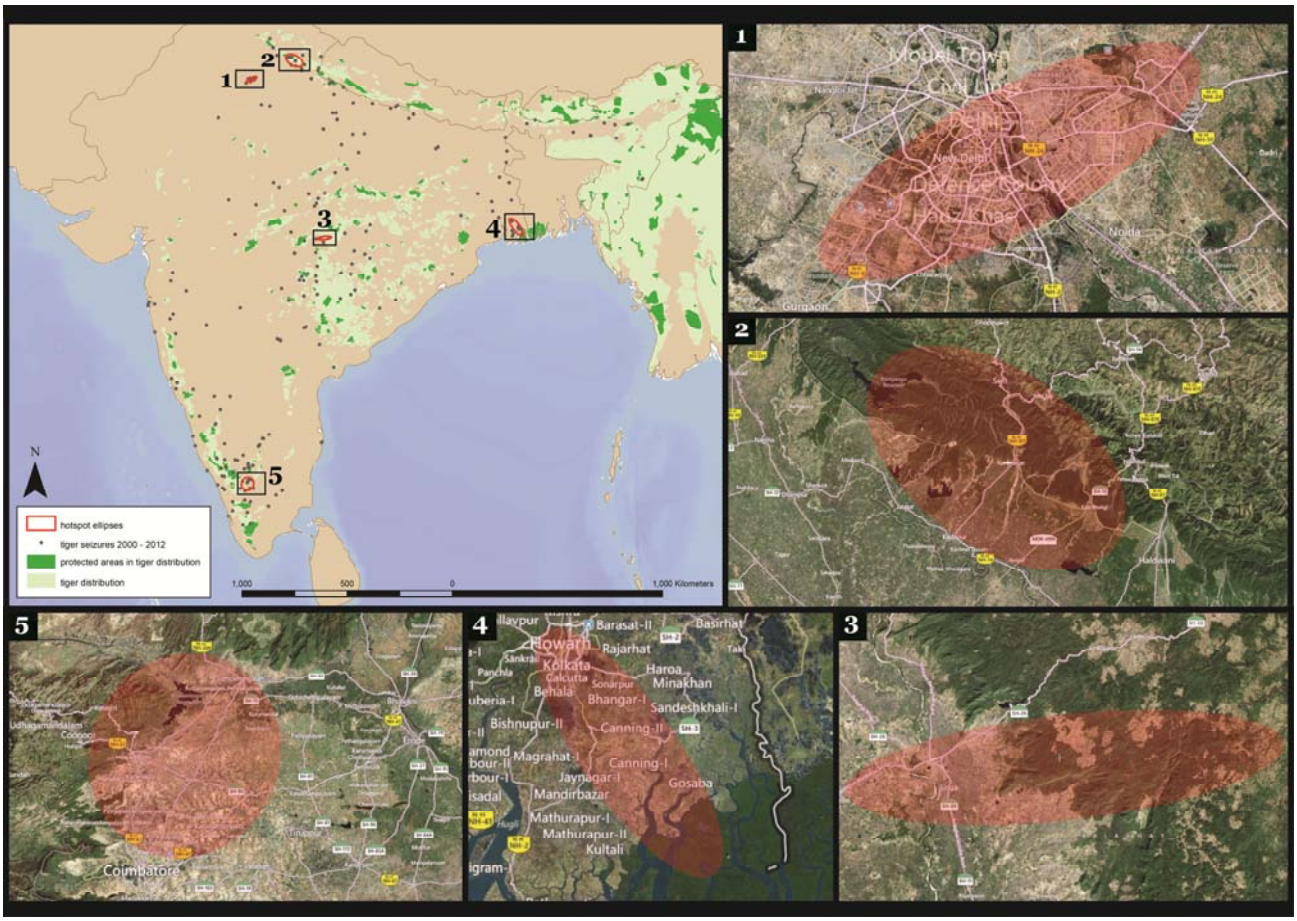
Sarah Stoner
TRAFFIC International
Kuala Lumpur, Malaysia

India is home to over half of the world's wild Tiger population and as a consequence records the greatest number of seizures globally. Since 2000, 336 seizures have been reported equating to an estimated 529 dead Tigers. Hotspot analysis of Tiger seizures has never been conducted in India and determining where clusters of activity exist is problematic.

Using the Crimestat nearest neighbour hierarchical clustering routine (*Nnh*), five significant clusters of seizures were identified. ArcGIS was used to map both the seizures and one standard deviational ellipses and were overlaid on tiger distribution and Protected Area* layers. Four of the ellipses were related to towns or cities which are also within close proximity of a Tiger reserve. Furthermore, transboundary trading of Tigers is prevalent but often securing agreement to combat trade at this level is challenging. Two clusters were also close to the borders of Nepal and Bangladesh. These findings will create leverage for law enforcement agencies to focus on the areas where seizures are most likely to occur to affect the greatest impact and will help create collaborative partnerships with neighbouring countries to tackle the issue at a regional level.

***A clearly defined geographical space, recognised, dedicated and managed, through legal or other effective means, to achieve the long-term conservation of nature with associated ecosystem services and cultural values. **SOURCE**: World Database Protected Area*

Figure 1: Tiger seizures in India (2000-2012, n=336



Chapter 8:

Hot Spot Analysis of Points: II

Richard Block
Dept. of Sociology
Loyola University
Chicago, IL

Carolyn Rebecca Block
Illinois Criminal Justice
Information Authority
Chicago, IL

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 8:
Hot Spot Analysis of Points: II

This chapter continues the discussion of hot spots. Two additional routines are discussed: the STAC routine and the K-Means routine. Figure 8.1 displays the Hot Spot Analysis II page. The first of these routines, the Spatial and Temporal Analysis of Crime (STAC), was developed by the Illinois Criminal Justice Information Authority and integrated into *CrimeStat* in version 2. The second routine - K-Means, is a partitioning technique. We will start first with STAC.

Spatial and Temporal Analysis of Crime (*STAC*)

The amount of information available in an automated pin map can be enormous. When geographic information systems were first introduced into policing, there were few ways to summarize the huge reservoir of mapped information that was suddenly available. In 1989, police departments in Illinois asked the Illinois Criminal Justice Information Authority to develop a technique to identify Hot Spot Areas, the densest clusters of points on a map (Block, 1994; Block & Block, 1999; Block & Block, 1995). The result was STAC, the first crime hot spot program.[1] Through the years, bells and whistles have been added to STAC, but the algorithm has remained essentially the same. STAC is a quick, visual, easy-to-use program for identifying Hot Spot Areas.

The STAC Hot Spot Area routine in *CrimeStat* searches for and identifies the densest clusters of incidents based on the scatter of points on the map. The STAC Hot Spot Area routine creates areal units from point data and identifies the major concentrations of points for a given distribution. It then represents each dense area by either a standard deviational ellipse or a convex hull.

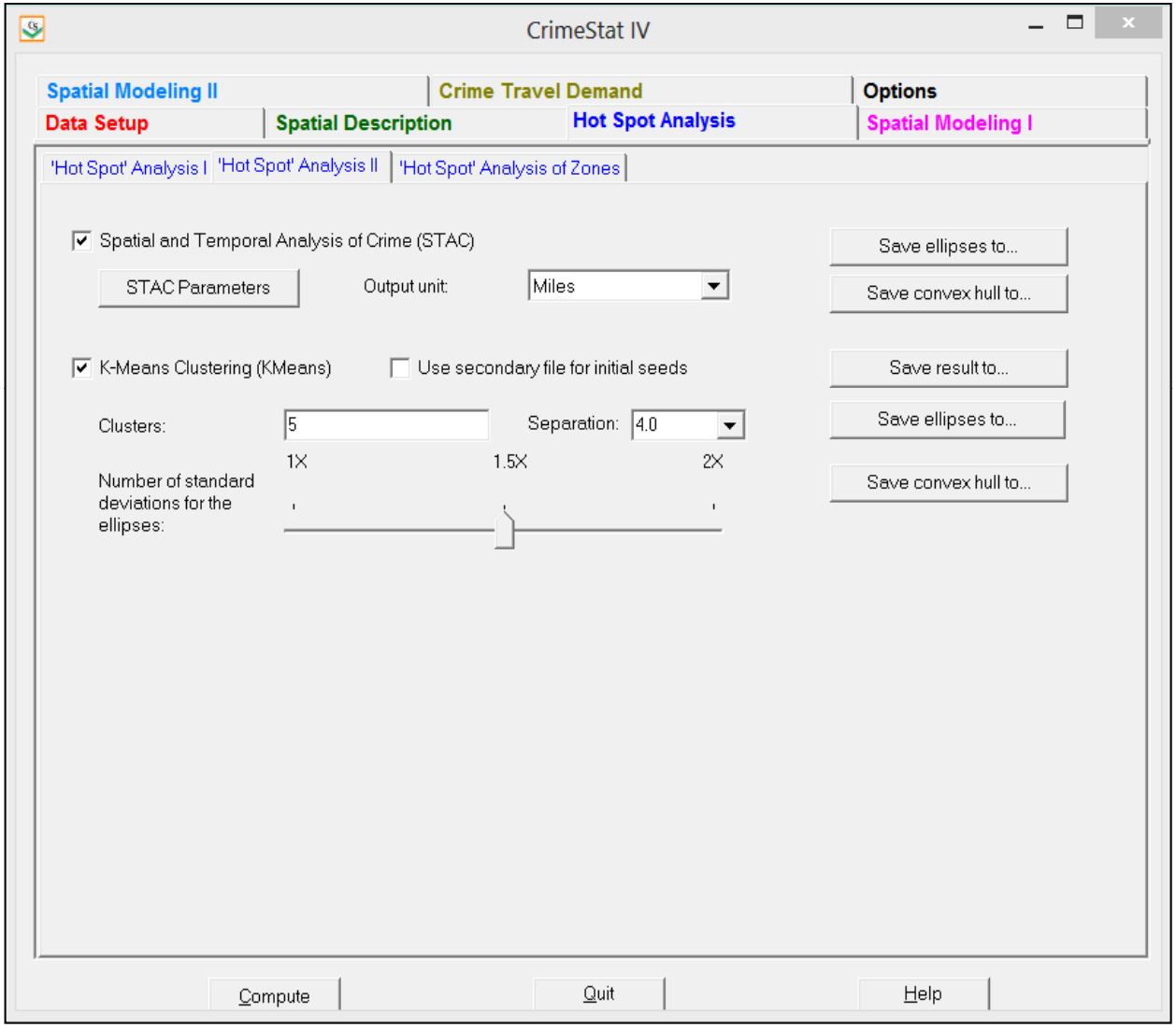
STAC is a scan-type clustering algorithm in which a circle is repeatedly laid over a grid and the number of events within the circle are counted (Openshaw, Charlton, Wymer and Craft, 1987; Openshaw, Craft, Charlton, and Birch, 1988; Turnbull, Iwano, Burnett, Howe, and Clark, 1990; Kulldorff, 1997). It, thus, shares with those other scan routines the property of multiple tests, but it differs in that the overlapping clusters are combined into larger cluster until there are no longer any overlapping circles. Thus, STAC clusters can be of differing sizes.

---

1 STAC is an abbreviation for Spatial and Temporal Analysis of Crime. The temporal section of the program was superceded by several other programs and was not updated for the millennium. Because many law enforcement users refer to STAC ellipses, we have retained that name.

Figure 8.1:

Hot Spot Analysis II Screen



The STAC Hot Spot Area routine in *CrimeStat* searches for and identifies the densest clusters of incidents based on the scatter of points on the map and then creates areal units from point data. It does this by identifying major concentrations of points for a given distribution and represents each dense area by either a standard deviational ellipse or a convex hull, or both (see Chapter 4). The boundaries of the ellipses or convex hulls can easily be displayed as mapped layers by standard GIS software.

STAC is not constrained by artificial or political boundaries, such as police beats or census tracts. This is important, because clusters of events and places (such as drug markets, gang territories, high violence taverns, or graffiti) do not necessarily stop at the border of a police beat.[2] Also, shading over an entire area may make it seem that the whole neighborhood is high-crime (or low-crime), even though the area may contain only one or two dense pockets of crime. Therefore, area-shaded maps could be misleading. In contrast, STAC Hot Spot Areas are based on the actual clusters of events or places on the map.

STAC is designed to help the crime analyst summarize a vast amount of geographic information so that practical policy-related issues can be addressed, such as resource allocation, crime analysis, beat definition, tactical and investigation decisions, or development of intervention strategies. An immediate concern of a law enforcement user of crime points on a GIS is the identification of areas that contain especially dense clusters of events. These pockets of crime demand police attention and can indicate different things for various crimes. For instance, a grouping of Criminal Damage to Property offenses could indicate gang activity. If motor vehicle thefts consistently cluster in one section of town, it could point to the need to change patrol patterns and procedures.

To take an example, Figure 8.2 shows the location of the seven densest Hot Spot Areas of street robbery in 1999 in Chicago. Four of the seven span the boundaries of police districts and two cover only a small part of a larger district. In a shaded area map, these dense clusters of robbery might be not easily identifiable. An area that is really dense might appear to be low-crime because it is divided by an arbitrary boundary. Using a shaded areal map aggregating the data within each district would give a general idea of the distribution of crime over the entire map, but it would not tell exactly where the clusters of crime are located.

For example, Figure 8.3 zooms in on Hot Spot Area 4 (the northernmost Hot Spot Area in Figure 8.2). Hot Spot Area 4 covers parts of two districts (shown by a pink boundary line in Figure 8.2). There are also four beats (shown by blue boundary lines). The shaded map indicates

2 However, there may be inadequate or, even, a lack of data on the other side of a border so that a hot spot is not fully defined.

Figure 8.2: STAC Hot Spots for 1999 Street Robberies

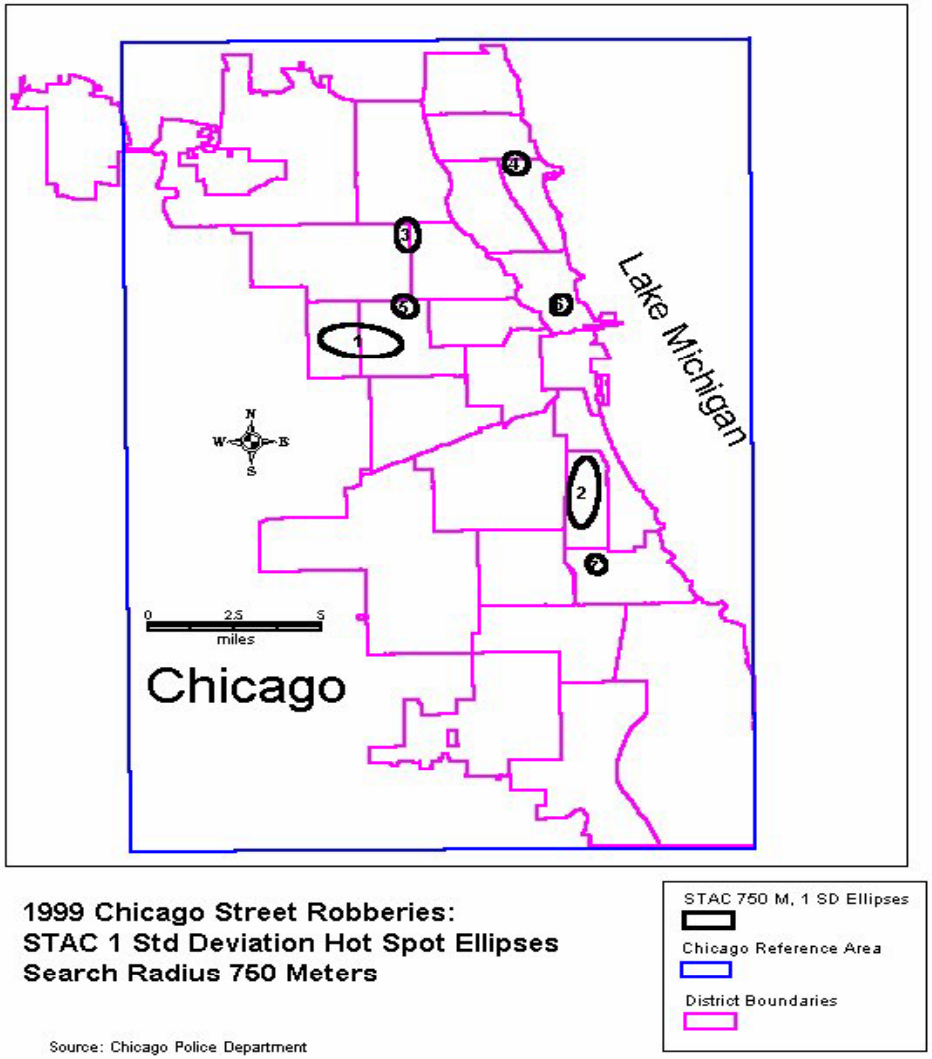
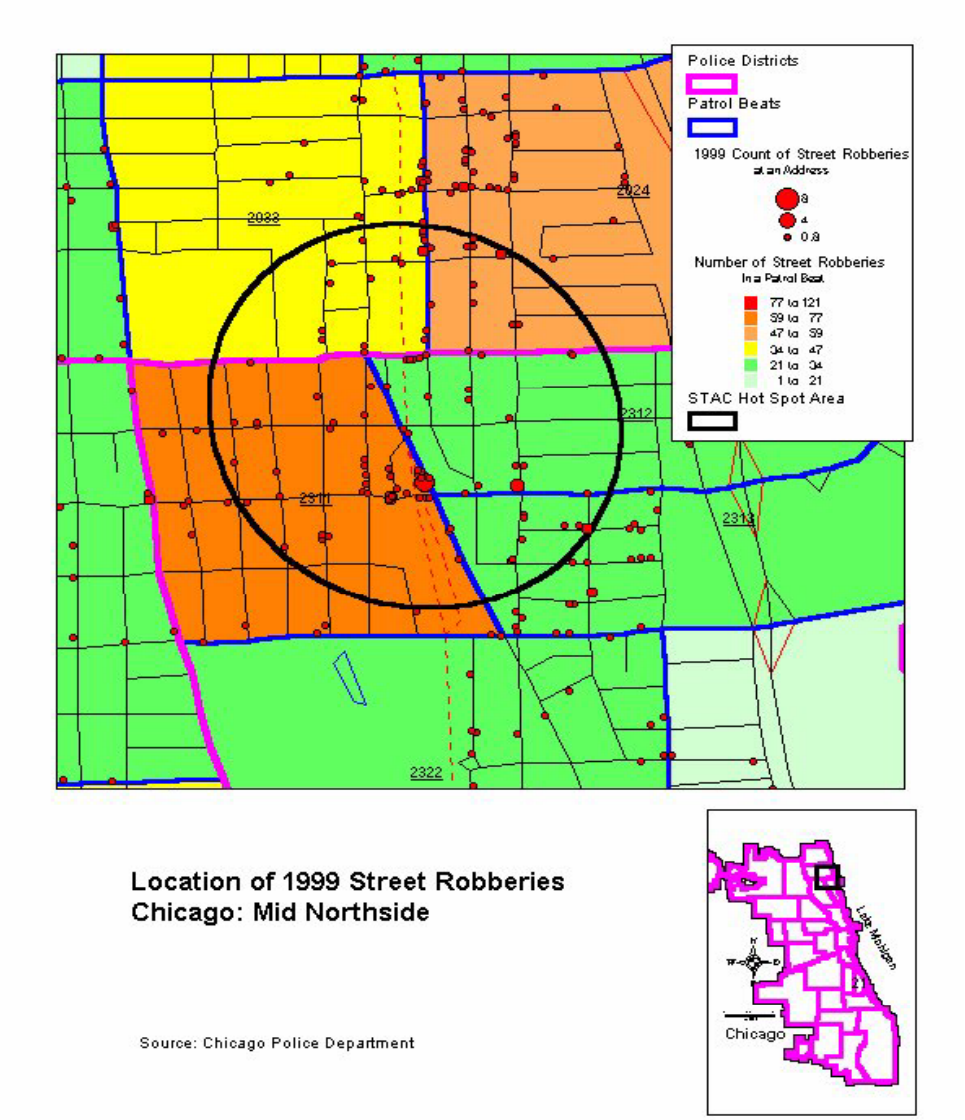


Figure 8.3: STAC 1999 Street Robbery Hot Spot Area 4



many incidents in beat 2311, but few in beats 2312, and 2313.[3] The incident distribution indicates that, while few incidents occurred overall in 2312 and 2313, most of the incidents that did occur were near to beat 2311. Incidents in beat 2311 mainly occurred on its eastern boundary. Portions of the beat were relatively free from street robbery. The Hot Spot Area identifies this clustering that spans beats and districts. Hot Spot Areas that overlap beat and district boundaries might suggest that patrol officers in these neighboring areas should coordinate their efforts in combating crime.

How STAC Identifies Hot Spot Areas

The following procedures identify hot spots in STAC. The program implements a search algorithm, looking for Hot Spot Areas:

1. STAC lays out a 20 x 20 grid structure (triangular or rectangular, defined by the user) on the plane defined by the area boundary (defined by the user on the Measurement Parameters page).

2. At each intersection of grid lines, there is a node. STAC places a circle on every node of the grid with a radius equal to 1.414 (the square root of 2) times the specified search radius. Thus, the circles overlap.

3. STAC counts the number of points falling within each circle, and ranks the circles in descending order. Multiple events can be counted at the same location.

4. STAC records all circles with at least two data points along with the number of points within each circle up to a maximum of 25 circles,. The X and Y coordinates of any node with at least two incidents within the search radius are recorded along with the number of data points found for each node.

5. These circles are then ranked in descending order according to the number of points and the top 25 search areas are selected.

6. If a point belongs to two different circles, the points within the circles are combined. This process is repeated until there are no overlapping circles. This routine avoids the problem of data points belonging to more than one cluster, and the additional problem of different cluster arrangements being possible with the same points. The result is called Hot Clusters.

3 The first two digits of a beat number designate the District.

Using the data points in each Hot Cluster, the routine calculates the standard deviational ellipse or convex hull (see Chapter 4). These are called *Hot Spot Areas*. Because the standard deviational ellipse is a statistical summary of the Hot Cluster points, it may not contain every Hot Cluster point. It also may contain points that are not in the Hot Cluster. On the other hand, the convex hulls will create a polygon around all points in the cluster.

The user can specify different search radii and re-run the routine. Given the same area boundary, different search radii will often produce different numbers of Hot Clusters. A search radius that is either too large or too small may fail to produce any. Experience and experimentation are needed to determine the most useful search radii.

Steps in Using *STAC*

STAC is available on the Hot Spot Analysis II tab under Spatial Description (see Figure 8.1). A brief summary of the steps is as follows:

1. *STAC* requires a primary file and a reference file (see Chapter 3). Optionally, STAC will use coverage area (on the Measurement Parameters tab) for simulation runs. Note: while *STAC* runs quite quickly, it runs more quickly with a Euclidean coordinate system such as UTM or State Plane.

2. Define the reference file (see Chapter 3). While *CrimeStat* does not include a data base manager or query system, a user can carry out analysis of different areas of a jurisdiction by using the boundaries of several reference areas. For example, define all of Chicago as a reference area and define each of the twenty-five police districts as additional reference areas. Hot Spot Areas can be identified for the city as a whole and for each district. In other words, the same incident file may be used for analysis of different map areas by using multiple reference files.
3. Define the search radius. Generally, a two-stage analysis is best. Start with a larger search radius and then analyze Hot Spot Areas with a smaller search radius. A search radius of more than one mile may not yield useful results in an area the size of Chicago (230 square miles).

4. Set the output units to miles or kilometers.

5. Specify the file output name for the ellipses or convex hulls.

6. Click on the *STAC* parameters button.

The object of *STAC* is to identify hot spots and display them with ellipses or convex hulls. Its key function is visual. Save the ellipses or hulls in the form most appropriate for the

system (e.g., *ArcGIS, MapInfo*). Because the ellipses or convex hulls are generated as polygons, they can be used for selections, queries, or thematic maps in a GIS. In addition to the ellipses and convex hulls, a table is output with all the information on density and location for each ellipse. It can be saved to a 'dbf' file, which can then be read by any spreadsheet program. The ellipses and convex hulls are numbered in the same order as the printed output.

STAC Parameters

The two most important parameters for running STAC are the boundary of the study area (reference area) and the search radius. A detailed discussion of the parameters follows. Figure 8.4 shows the *STAC* parameters screen.

Search radius

The search radius is the key setting in *STAC*. In general, the larger the search radius, the more incidents that will be included in each Hot Cluster and the larger the ellipse that will be displayed. Smaller search radii generally result in more ellipses of a smaller size.

A good strategy is to initially use a larger radius and then re-analyze areas that are 'hot' with a smaller radius. In Chicago, we have found that a 0.5 mile radius is appropriate for the city as a whole and a 0.25 mile search radius for one of the 25 districts. It will be necessary to experiment to determine an appropriate search radius.

Units

Specify the units for the search radius. The default is miles and the default search radius is 0.5 miles.

Minimum points per cluster

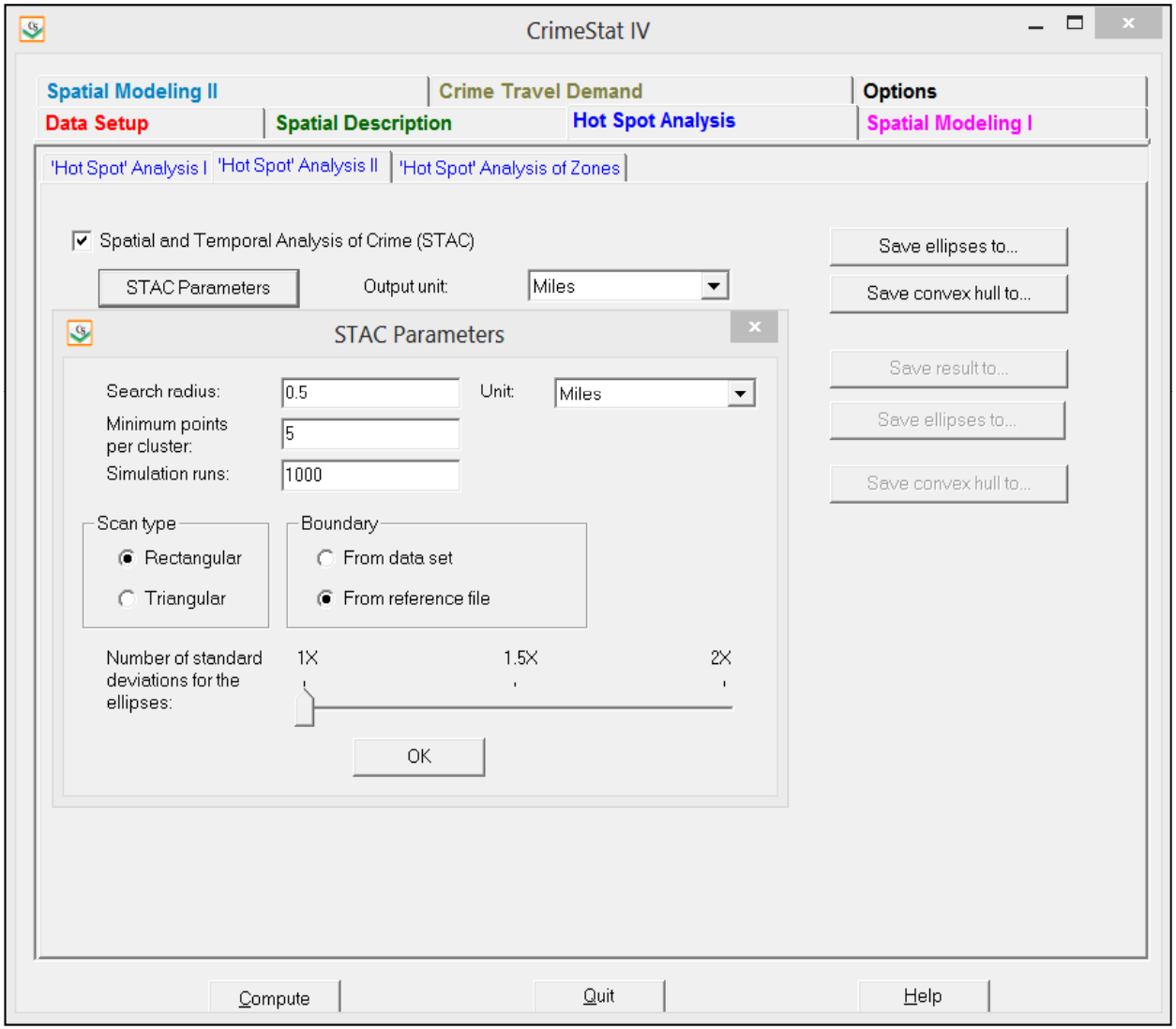
Specify the minimum number of points to be included in a Hot Cluster. The limit for the minimum points in a Hot Cluster is two. The usual choice is to use a minimum of 10.

Boundary

Select the reference file to be used for the analysis. The user can choose the boundary from the data set (i.e., the minimum and maximum X/Y values) or from the reference boundary.

Figure 8.4:

STAC Parameters Setup



In our opinion, the choice of the reference boundary is best. If the data set is used to define the reference boundary, the rectangle defined by the minimum and maximum X and Y coordinates will be used.

Scan type

Select the scan type for the grid. Choose Rectangular if the analysis area has a mostly grided street pattern. Chose Triangular if the analysis area generally has an irregular street pattern.

Graphical output files

Select whether the graphical output will be displayed as standard deviational ellipse or as convex hulls, or both (see Chapter 4). For ellipses, select the number of standard deviations for the ellipses. One (1X), 1.5X, and 2X standard deviational ellipses can be selected.

One standard deviational ellipse should be sufficient for most analysis. While 1X standard deviational ellipses rarely overlap, 1.5X and 2X standard deviational ellipses often do. A larger ellipse will include more of the Hot Cluster points; a small ellipse will produce a more focused Hot Cluster identification. The user will have to work out a balance between defining a cluster precisely compared to making it so large as to be unclear.

Simulation runs

Specify whether any simulation runs are to be made. To test the significance of *STAC* clusters, it is necessary to run a Monte Carlo simulation (Dwass, 1957; Barnard, 1963). *CrimeStat* includes a Monte Carlo simulation routine that produces approximate confidence intervals (called *credible intervals*) for the particular STAC model that has been run. Essentially, the Monte Carlo simulation assigns N cases randomly to a rectangle with the same area as specified on the Measurement Parameters tab and evaluates the number of clusters according to the defined parameters (i.e., search radius). The simulation routine repeats the random clustering *K* times, where *K* is defined by the user (e.g., 100, 1,000, 10,000).

By running the simulation many times, the user can assess credible intervals for the particular number of clusters and density of clusters. The default is zero simulation runs.. If a simulation run is selected, the user should identify the area of the study region on the Measurement Parameters tab. It is better to use the jurisdictional area rather than the reference area if the jurisdiction is irregularly shaped. For those jurisdictions, using the area defined by the

reference file coordinates (minimum X/Y and maximum X/Y) may result in identifying areas as hot spots that are not.

To compare the STAC output with the Monte Carlo simulation, there are two criteria that can be used – the number of clusters and cluster density (incidents per unit area). However, these tend to have contrary trends which depend on the search circle. Since STAC works by, first, counting incidents that fall within a search circle and, second, by aggregating overlapping search circles, a larger search circle will tend to show fewer, but higher density, clusters than would be expected on the basis of chance. The difference between the density of incidents in *STAC* ellipses in a spatially random data set and the *STAC* ellipses in the actual data set is a test of the strength of the clustering detected by *STAC*. Alternatively, a smaller search circle will tend to identify more clusters than would be expected on the basis of chance. In general, for citywide planning purposes, use a larger search circle (e.g., 0.5 miles) while for neighborhood planning purposes, use a smaller search circle (e.g., 0.1 miles or 0.25 miles).

Output

Ellipses or convex hulls

The ellipses are output with a prefix of 'St' before the output file name while the convex hulls are output with a prefix of 'Cst'before the output file name. *ArcGIS* 'shp' files can be opened as themes and can also be added as a *MapInfo* layer using the Universal Translator Tool. *MapInfo* Mif/Mid files must be imported using the command 'Table Import'. Both *MapInfo* and *ArcGIS* files are polygons and can be used for queries and thematic mapping. Google Earth 'kml' file can be displayed in that program.

Printed Output

Table 8.1 shows the printed output. Be sure to record the file name and the reference file (if any that is used). The output includes:

1. The first section of the output documents parameter settings and file size. Sample size indicates the number of points in the file specified in the setup.

2. Measurement Type indicates the type of distance measurement, direct or Indirect (Manhattan).

3. Scan Type indicates a rectangular or triangular grid specified in the setup.

Table 8.1:
Printed Output for STAC
1999 Street Robberies on Chicago's Northeast Side

```
Spatial and Temporal Analysis of Crime:
---------------------------------------

    Sample size ...........:  1181
    Measurement type ......:  Direct
    Scan type....... ......:  Rectangular
    Input units .... ......:  Degrees
    Output units ... ......:  Miles, Squared Miles, Points per Squared Miles
    Standard Deviations ...:  1
    Search radius..........:  804.672000
    Boundary...............:  -76.83302,39.23274 to -76.38390,39.59103
    Points inside boundary.:  1179
    Simulation runs .......:  1000
```

Cluster	Mean X	Mean Y	Rotation	X-Axis	Y-Axis	Area	Points	Ellipse Density
1	-76.44915	39.31484	89.41867	1.04768	0.25053	0.82460	106	128.546688
2	-76.73681	39.28658	69.91502	0.22142	0.88202	0.61354	63	102.682109
3	-76.57098	39.38499	37.10812	0.34793	0.82213	0.89863	61	67.880882
4	-76.77129	39.35987	11.26360	0.94336	0.26216	0.77695	61	78.511958
5	-76.51830	39.26019	8.37773	0.43717	0.25497	0.35017	43	22.796997
6	-76.60231	39.40086	14.84392	0.17969	0.29466	0.16634	36	16.423811
7	-76.73087	39.34246	41.07812	0.31007	0.25885	0.25215	35	38.806566
8	-76.75451	39.31110	74.78196	0.19154	0.31572	0.18998	24	26.326405

Distribution of the number of clusters found in simulation (percentile):

Percentile	Clusters	Area	Points	Density
min	12	0.01113	5	4.673554
0.5	13	0.02389	5	4.924993
1.0	13	0.03587	5	4.977644
2.5	14	0.05081	5	5.236646
5.0	14	0.06177	5	5.505124
95.0	19	1.24974	14	82.281060
97.5	19	1.39923	16	101.053102
99.0	20	1.58861	17	140.078387
99.5	20	1.67065	19	209.279368
max	20	2.08665	23	449.401912

4. Input Unit indicates the units of the coordinates specified in the setup, degrees (if latitude/longitude) or meters or feet (if projected).

5. Output Units indicate the unit of density and length specified in the setup for the output and ellipses. Output Units are generally, miles or kilometers.

6. Search Radius is the units specified in the setup. In Figure 8.2 above, this is meters.

7. Boundary identifies the coordinates of the lower left and upper right corner of the study area.

8. Points inside the boundary count the number of points within the reference file. This may be fewer than the number of points in the total file when a smaller area is being used for analysis (see Table 8.1).

9. Simulation Runs indicate the number of runs, if any specified in the setup.

10. Finally, STAC printed output provides summary statistics for each Hot Spot Area:

    A. Cluster identification number for each ellipse. This corresponds to their order in a table view in *ArcGIS* or the browser in *MapInfo*.

    B. Mean X and Mean Y - Coordinates of the mean center of the ellipse.

    C. Rotation- the degrees the ellipse is rotated (0 is horizontal; 90 is vertical).

    D. X-axis and Y-axis - the length (in the selected output units) of the x and y axis. In the example, the length of the x axis of ellipse 1 is 1.04768 miles.

    E. Area - the area of the ellipse in square units. Ellipses are ordered according to their size. In the example, Ellipse 1 is 0.8246 square miles.

    F. Points - the number of points in the Hot Cluster. In the example, there are 61 points in cluster 3.

    G. Cluster Density - the number of points per square unit. The largest cluster is not necessarily the densest. In this output, cluster eight is the smallest, but its density is higher than two other clusters.

H. The distribution of the simulations (if specified).

Note that the number of actual clusters in the example (8) is smaller than the number that would be expected if the data were randomly distributed at the 95 percentile (19). The reason for this is that STAC aggregates smaller clusters that are close to each other, that is where their search circles overlap. Hence, with a large search circle, as in this output – 0.5 miles, will generally lead to fewer clusters than a Monte Carlo simulation. On the other hand, the cluster density indicates that two of the clusters (1 and 2) have a higher density than the 95 percentile density. These clusters are most likely real clusters, rather than random collections, and should be the focus of further analysis.

The best way to print or save *CrimeStat* printed output is to place the cursor inside the output window and *Select all,* then copy and paste the selection into a word processing document in landscape mode. Make sure to adequately annotate the file, especially the type of incidents, the reference boundary, and the name of the output file. This can be very important for future reference.

Example: A STAC Analysis of 1999 Chicago Street Robberies

STAC Hot Spot Areas were calculated for all street (or sidewalk or alley) robberies occurring in Chicago in 1999 (n=13,009).[4] There were 13,007 within the search boundary. The search radius was set for 750 meters (approximately 0.5 mile), and the ellipses were set to one standard deviation. Ten was the minimum number of incidents per cluster.

In Figure 8.2 (shown earlier), STAC detected seven ellipses. The areas of the seven ellipses ranged from 5 square kilometers to 0.7 square kilometers, and the number of incidents in an ellipse ranged from 760 to 153. The smallest ellipse (number 7 in Figure 8.2) was the densest, 222 robberies per square kilometer. Of the 13,007 street robberies, 2,375 were in a cluster. Therefore, 18 percent of all of Chicago's street robberies in 1999 occurred in 6% of its 233 square mile area.

To map the results, the ellipse boundaries were imported into *MapInfo* as a mif/mid file and overlaid on a map of police districts. The large blue rectangle in Figure 8.2 designates the search boundary (reference file). O'Hare Airport was excluded because exact geo-coding is not possible for the few street robberies that occurred there. At a city-wide scale, the map is interesting, but is mainly useful for confirming what is already known. Ellipse 1, on the west

4 The Chicago Police Department made available the incidents in this analysis to Richard Block for the evaluation of the Chicago Alternative Police Strategy (CAPS).

side, has had a high level of violence for many years. Ellipses 2 and 6 are centered on areas where high rise public housing projects are gradually being abandoned. Overall, these ellipses are not very useful for tactical purposes. However, they point out that four Hot Spot Areas cross District boundaries, and that the large number of street robberies in these areas might be lost in separate district reports.

A Neighborhood STAC Analysis

The presence of Ellipse 4 (the northernmost ellipse in Figure 8.2) might be unexpected to many Chicagoans. The mid-Northside, near the Lake Michigan, is generally considered to be a relatively affluent and safe neighborhood. However, the neighborhood around Ellipse 4 has had a high level of crime for many years. It was an entertainment center in the Roaring Twenties, and several institutions of that era remain. Today it is an area with multiple, often conflicting, uses. A more detailed analysis of the neighborhood with the help of STAC may point to specific areas that need increased patrol or prevention activities.

The second step of STAC analysis was to define a focused search boundary area around Ellipse 4. This was done easily by creating a new map layer in MapInfo and drawing a rectangle around the desired study area. Clicking on the study area gave the required *CrimeStat* reference boundary maximum and minimum coordinates. Using this more focused boundary, STAC was run a second time with a 200 meter search radius and the same file of 13,009 cases. The search boundary (reference file) now contained 442 incidents. STAC detected three ellipses that contained 231 incidents. The STAC ellipses were then imported into *MapInfo* and mapped (Figure 8.5).

As the area covered by a map grows smaller, detailed information about crime patterns and the community can be added. In this map, the STAC ellipses were overlaid on the locations of incidents (sized according to the number occurring at each location) and streets.[5] Much of the area is relatively crime-free. The most frequent locations for street robbery do not coincide with main streets. Street robbery incidents tend to cluster near rapid transit stations and the blocks immediately surrounding them. For example, Argyle Street, between Broadway and Sheridan, is the site of 'New China Town'. It is an area with a number of street robberies and is a destination area for 'Northsiders' who want an inexpensive Chinese or Vietnamese meal.

---

5 In general a designated main surface street occurs every mile on Chicago↘s grid, and there are eight blocks to the mile. In this map, Lawrence and Ashland are main Grid streets. In this area, there are also several diagonal main streets that either follow the lake shore or old Indian trails.

There is a particularly risky area in the neighborhood of Broadway and Wilson adjacent to Truman Community College. In a previous analysis of the Bronx, Fordham University was shown to be a similar attractor for robbery incidents. Colleges supply good targets for street robbery. Also, authority for security is split between the college and the city police. The area around Broadway and Wilson has been risky for many years. Ninety years ago, it was the northern terminus of rapid transit, and the site of several very inexpensive hotels, two of which still existed. Today the area has several pawn shops and currency exchanges. There is an ATM located in the EL station. In 1999, the area looked dangerous and dirty. Finally, the area has many blind corners and alleys that could serve as sites for robbery; this is unusual for Chicago.

The census block that includes the northwest corner of Broadway and Wilson ranked fifth among Chicago's 21,000 census blocks in number of street robberies in 1999.[6]

Changes need to be made to reduce the risk of street robbery in this area. Mapping identifies a problem with street robberies, but to investigate possible changes it is necessary to go beyond mapping. Aside from changes in patrol practices, what physical changes might aid in crime reduction? The campus has very little parking. The administration assumes that students take public transportation, but many do not. A secure parking garage that could serve both the elevated station and the school could be constructed (vacant land is available). In addition, increased police patrol in the area between the school and the el station could be implemented.

Advantages of STAC

STAC has a number of advantages as a clustering algorithm:

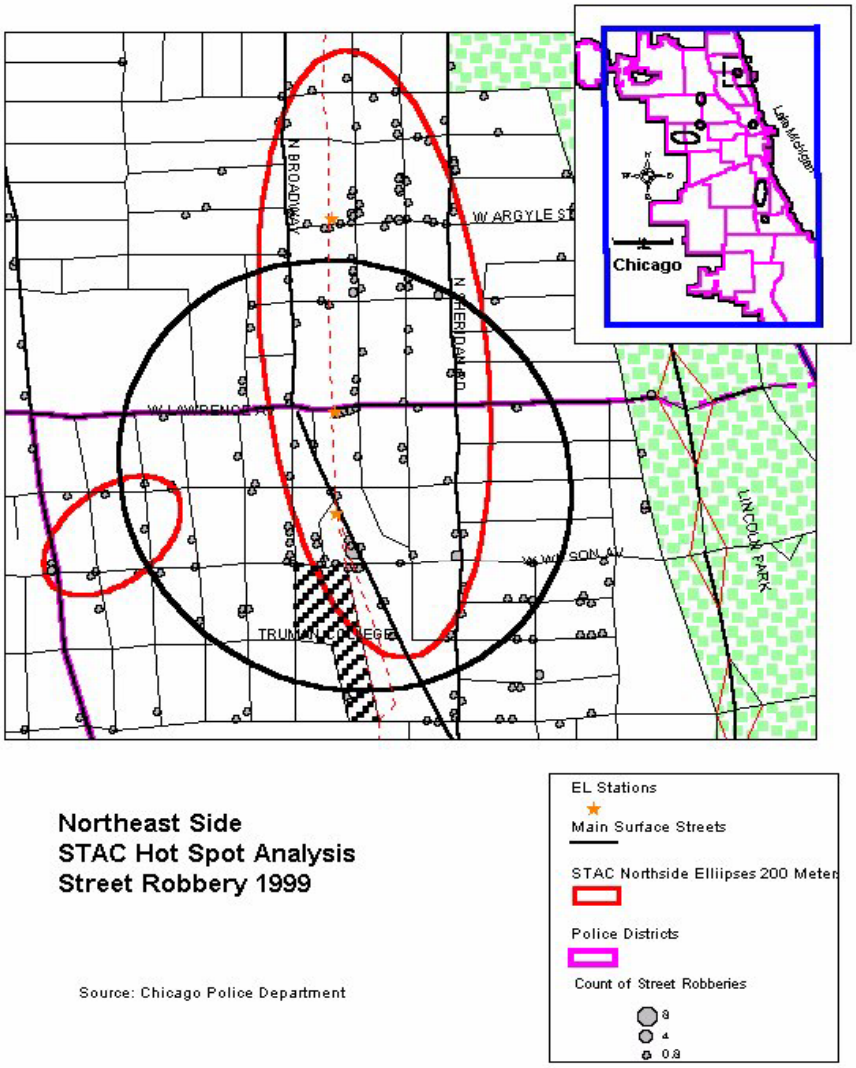
1. The routine can analyze a very large number of cases quickly. It is very fast using a Euclidean projection such as UTM or State Plane, but not quite as fast using spherical coordinates (latitude/longitude).

2. The user can control the approximate size of the ellipses through the search radius, the minimum number of points per hot spot, and the study area. These features allow for a brod search for Hot Spot Areas over an entire city and a second search concentrating on a smaller area and more focused Hot Spot Areas for local tactical use.

---

6 This example was originally conducted with CrimeStat II. In subsequent years, many of these suggestions were implemented and the area is no longer a hot spot.

Figure 8.5: STAC Hot Spots for Northeast Side Street Robberies



3. STAC and Nnh hierarchical clustering (discussed in Chapter 7) are complimentary. The Nnh first derives small ellipses and then aggregates to larger ones. The recommended STAC procedure is to first derive large area ellipses and then break these down into smaller areas for tactical analysis. There should be some convergence between the two approaches.

4. The visual display of STAC ellipses or convex hulls is quite intuitive, especially for area-wide interventions (e.g., patrol beats)

5. Hot spots need not be limited to a single kind of crime, place or even. For example, ellipses of drug crime can be overlaid on those for burglary. Some causal factors are also analyzable with STAC ellipses. For example, ellipses of street robbery can be compared to those for liquor licenses.

6. STAC is a density search clustering method that adapts itself to the size of the clusters. Essentially, it looks for areas of common, high density.

7. Unlike the Nnh routine, which has a constant threshold (search radius), STAC can create clusters of unequal size because overlapping clusters are combined until there is no overlap.

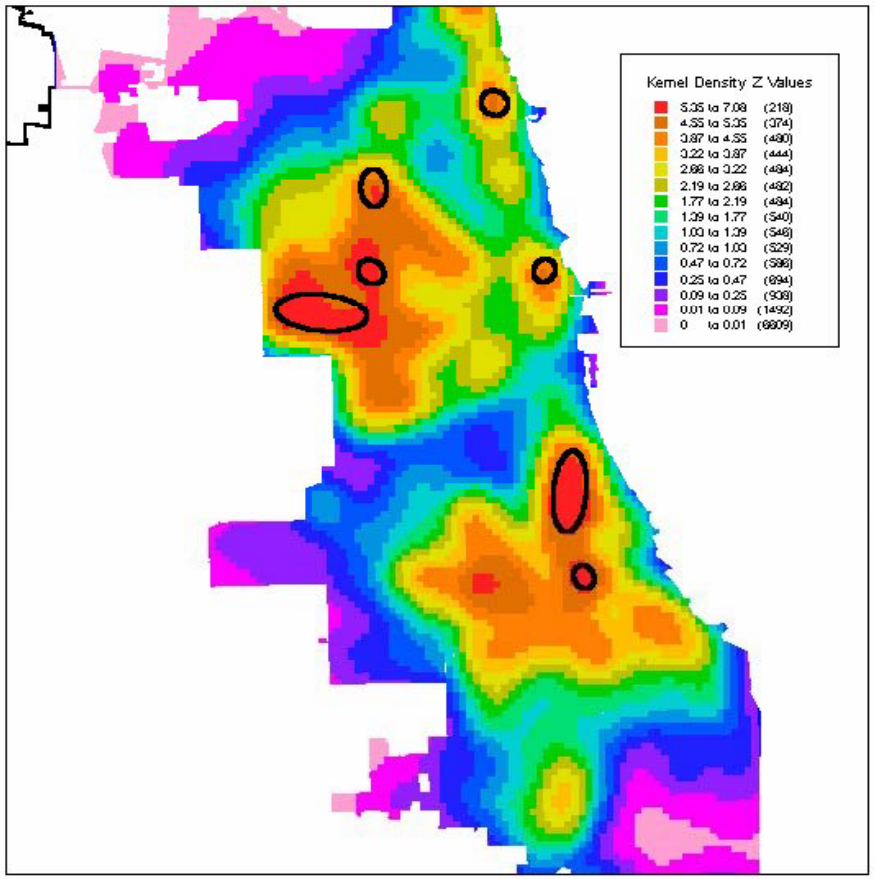
Limitations of STAC

There are also some limitations to using STAC:

1. The distribution of incidents within clusters is not necessarily uniform. The user should be careful not to assume that it is. A mapped theme of the Mode routine (see Chapter 7) according to number of incidents or the single kernel density interpolation (see Chapter 10) overlaid with STAC ellipses are good ways to overcome this problem (see Figure 8.5 above and Figure 8.6 below).

2. STAC tends to create larger clusters than the Nnh. The reason is that it combines points from overlapping search circles. It is unable to identify smaller clusters that are part of a larger grouping (a hierarchy) and, instead, tends to choose the larger grouping. The result is the density of events in STAC clusters are not as intense as in Nnh first-order clusters, but are more similar to Nnh second-order clusters (Chainey, Thompson & Uhlig, 2008; Levine, 2008).

Figure 8.6: STAC Robbery Hot Spots and Kernel Density Estimation



Chicago Street Robbery 1999 :
Comparison of STAC and
Single Kernel Density Estimation

For example, with a 1996 Baltimore County burglary file of 6,051 incidents, the default settings for STAC (0.5 mile search radius and a minimum of 5 points per cluster) produced 8 clusters compared to 158 for the Nnh with its default settings (random nearest neighbor distance and a minimum of 10 points per cluster). Depending on the purpose of the clustering, this can be an advantage or a disadvantage. STAC clusters can identify areas for patrol beats but are less able to identify very small areas where there is an intense concentration of events and which require geographically-specific interventions (e.g., improving street lighting, setting up block-wide security strategies).

3. Small changes in the STAC study area boundary can result in quite different depictions of the ellipses, even with the same study area measurement. Retaining the same reference file for repeated analyses alleviates this problem. The analysis should also be documented for the analysis parameters.

4. Because STAC aggregates overlapping search circles, it tends to miss identifying smaller clusters that are close to each other. This is particularly true when a larger search circle is used. Thus, the method tends to increase Type II statistical errors (failing to reject a false null hypothesis). The use of smaller search circles can minimize this problem. While there are definite uses in a larger search circle, for example in identifying patrol areas or multi-neighborhood crime hot spots, the user needs to be aware of how the search circle can affect the number of clusters identified and the potential for missing clusters that are actually separate yet close to each other.

5. STAC is based on the distribution of events. Neither land use nor risk factors is accounted for. It is up to the analyst to identify the characteristics that make a Hot Spot 'hot'.

Nevertheless, if used carefully, STAC is a useful tool for detecting clusters and can allow an analyst to experiment with varying search radii and reference boundaries.

K-Means Partitioning Clustering

The *K-Means* clustering routine (Kmeans) is a partitioning procedure where the data are grouped into *K* groups defined by the user. A specified number of seed locations, *K*, are defined by the user (Fisher, 1958; MacQueen, 1967; Aldenderfer and Blashfield, 1984; Systat, 2008). The routine tries to find the best positioning of the *K* centers and then assigns each point to the center that is nearest. Like the nearest neighbor hierarchical (Nnh) routine, the Kmeans assigns

points to one, and only one, cluster. However, unlike the Nnh procedure, all points are assigned to clusters. Thus, there is no hierarchy to the assignment; that is there are no second- or higher-order clusters. It is part of a family of cluster methods called *supervised clustering* (Finley & Joachins, 2005; Eick, Zeidat & Zhao, 2004).

The technique is useful when a user want to control the grouping. For example, if there are 10 police precincts in a jurisdiction, an analyst might want to identify the 10 most compact clusters, one for each precinct. Alternatively, if a previous analysis has shown there were 24 clusters, then an analyst could check whether the clusters have shifted over time by also asking for 24 clusters. By definition, the technique is somewhat arbitrary since the user defines how many clusters are to be expected. Whether a cluster should be considered a hot spot or not should depend on the extent to which a user wants to replicate hot spots.

The theory of the K-Means procedure is relatively straightforward. The implementation is more complicated. K-Means represents an attempt to define an optimal number of *K* locations where the sum of the distance from every point to each of the *K* centers is minimized. It is a variation of the old location theory paradigm of how to locate *K* facilities (e.g., police stations, hospitals, shopping centers) given the distribution of population (Haggett, Cliff, and Frey, 1977). That is, how does one identify *supply* facilities in relation to the location of *demand*? In theory, solving this question is an empirical solution, what is frequently called *global optimization*. One tries every combination of *K* objects where *K* is a subset of the total population of incidents (or people), *N*, and measures the distance from every incident point to every one of the *K* locations. The particular combination which gives the minimal sum of all distances (or all squared distances) is considered the best solution. In practice, however, solving this is computationally almost impossible, particularly if *N* is large. For example, with 6000 incidents grouped into 20 partitions (clusters), one cannot solve this with any normal computer since there are:

$$\frac{6000!}{20!5980!} = 1.456x10^{57} \tag{8.1}$$

combinations. No computer can solve that number and few spreadsheets can calculate the factorial of *N* greater than about 127.[7] In other words, it is almost impossible to solve computationally.

Practically, therefore, the different implementations of the K-Means routine make initial guesses about the *K* locations and then optimize the seating of this location in relation to the

7 The total number of ways for selecting K distinct combinations of *N* incidents, irrespective of order, is $\frac{N!}{K!(N-K)!}$ (Burt and Barber, 1996, 155).

nearby points. This is called *local optimization*. Unfortunately, each K-Means routine has a different way to define the initial locations so that two K-Means procedures will usually not produce the same results even if *K* is identical (Everitt, 2011; Systat, 2008; Everitt, Landau & Leese, 2001).

CrimeStat K-Means Routine

The K-Means routine in *CrimeStat* also makes an initial guess about the *K* locations and then optimizes the distribution locally. The procedure that is adopted makes initial estimates about location of the K clusters (seeds), assigns all points to its nearest seed location, re-calculates a center for each cluster which becomes a new seed, and then repeats the procedure all over again. The procedure stops when there are very few changes to the cluster composition (see endnote *i*).

The default K-Means clustering routine follows an algorithm for grouping all point locations into one, and only one, of these *K* groups. There are two general steps: 1) the identification of an initial guess (seed) for the location of the *K* clusters, and 2) local optimization which assigns each point to the nearest of the *K* clusters. First, a grid is overlaid on the data set and the number of points falling within each grid cell is counted. The grid cell with the most points is the initial first cluster and the centroid of the cell becomes the initial seed location.

The second initial cluster is the grid cell with the next most points that are separated by at least:

$$Separation = t * 0.5\sqrt{\frac{A}{N}} \tag{8.2}$$

where t is the Student's t-value for the .01 significance level (2.358), A is the area of the region, and N is the sample size. Again, the centroid of the grid cell becomes the initial second seed location. A third initial cluster is then selected which is the grid cell with the third most points and which is separated from the first two grid cells by at least the separation factor defined above. This process is repeated until *K* initial seed locations are chosen.

The algorithm then conducts *local optimization*. It assigns each point to the nearest of the initial K seed locations to form an initial cluster. For each of the initial clusters, the routine then calculates the center of minimum distance and re-assigns all points to the center of minimum distance to which it is closest. This becomes the second iteration of clusters with the center of minimum distance being the second seed location.

The routine repeats this process (assigning each point to the nearest seed location, re-calculating the center of minimum distance for each cluster to form a new seed location, and then re-assigning all points to the nearest new seed location) until no points change clusters. Finally, for each cluster, the routine outputs to the screen the statistics for a 1X standard deviational ellipse and can also output the results graphically as either standard deviational ellipses (1X, 1.5X, or 2X) or as convex hulls.

Control over Initial Selection of Clusters

Changing the separation between clusters

One problem with this approach is that in highly concentrated distributions, such as with most crime incidents in a metropolitan area, the separation between clusters may not be sufficiently large to detect clusters farther away from the concentration; the algorithm will tend to sub-divide concentrated groupings of incidents into multiple clusters rather than seek clusters that are less concentrated and, usually, farther away. To increase the flexibility of the routine, *CrimeStat* allows the user to modify the initial selection of clusters since this has a large effect on the final grouping (Everitt, 2011).

There are two ways the initial selection of cluster centers can be modified. First, the user can increase or decrease the ***separation factor***. Formula 8.2 is still used to separate each of the initial clusters, but the user can either select a t-value from 1 to 10 from the drop down menu or write in any number for the separation, including fractions, to increase or decrease the separation between the initial clusters. The default separation is set at 4. The effect of this is to modify the grid cell sizes for the initial cluster so as to force larger or small distances between the clusters.

Figure 8.7 shows a simulation of eight clusters in Baltimore County, four of which have higher concentrations than the other two. Figure 8.8 shows the results of running the K-Means clustering routine twice, both of which requested K=8 groupings. However, in one of the partitions there was a separation of 4 (the default separation shown as dashed green ellipses) while the other partition had a separation of 18 (solid blue ellipses). As seen, the partition with the larger separation captures the eight clusters better. With the smaller separation (4), the routine subdivided the dense cluster in the west into three separate clusters while combining one of these with the grouping of points directly to the north. Similarly, it combined two groupings in the northern part of the study area into a single cluster. The effect of increasing the separation was to produce a better visual fit with the groupings of the points.

Figure 8.7:

Separated Data and K-Means Clustering

Data Grouped into Eight Clusters

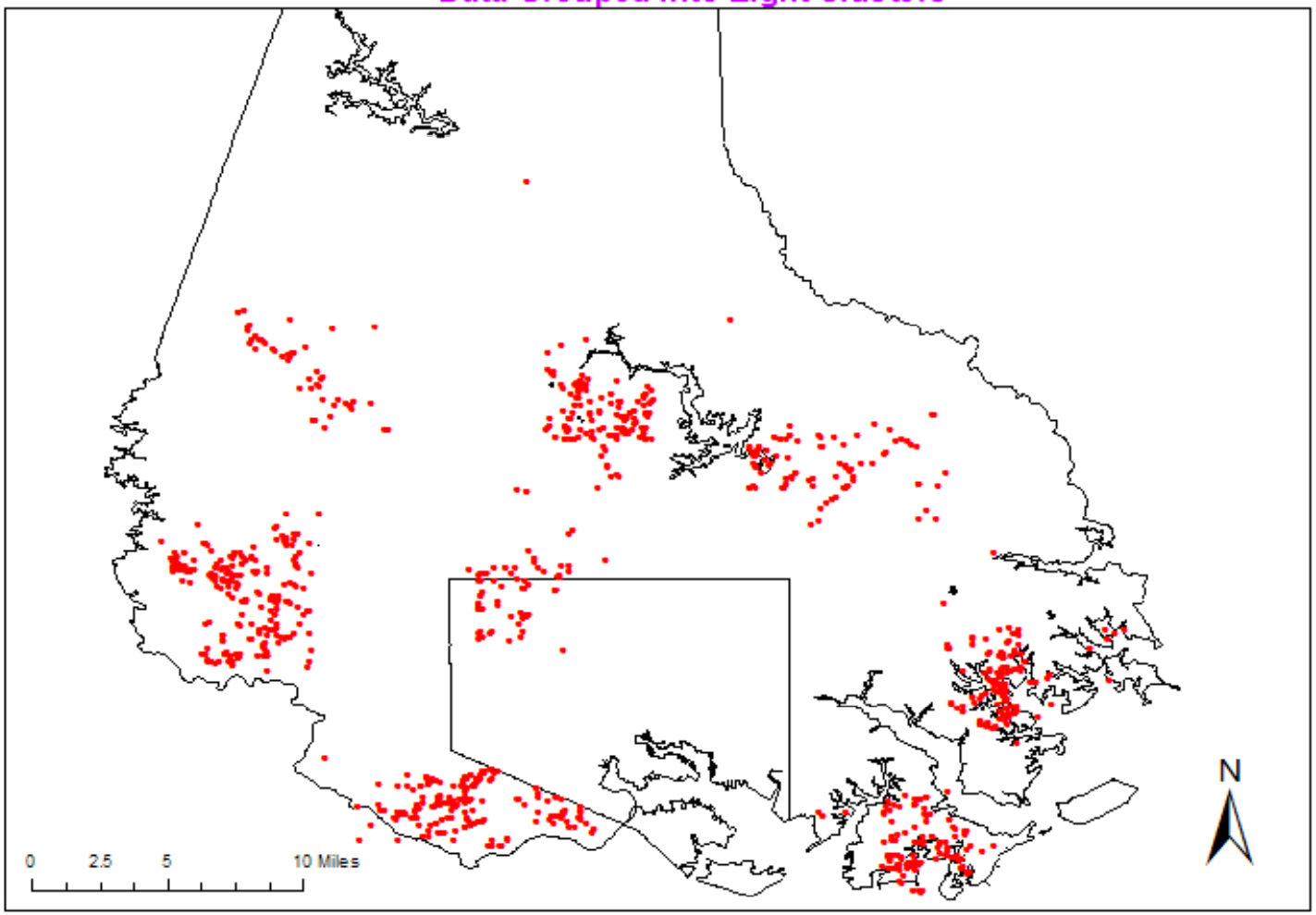
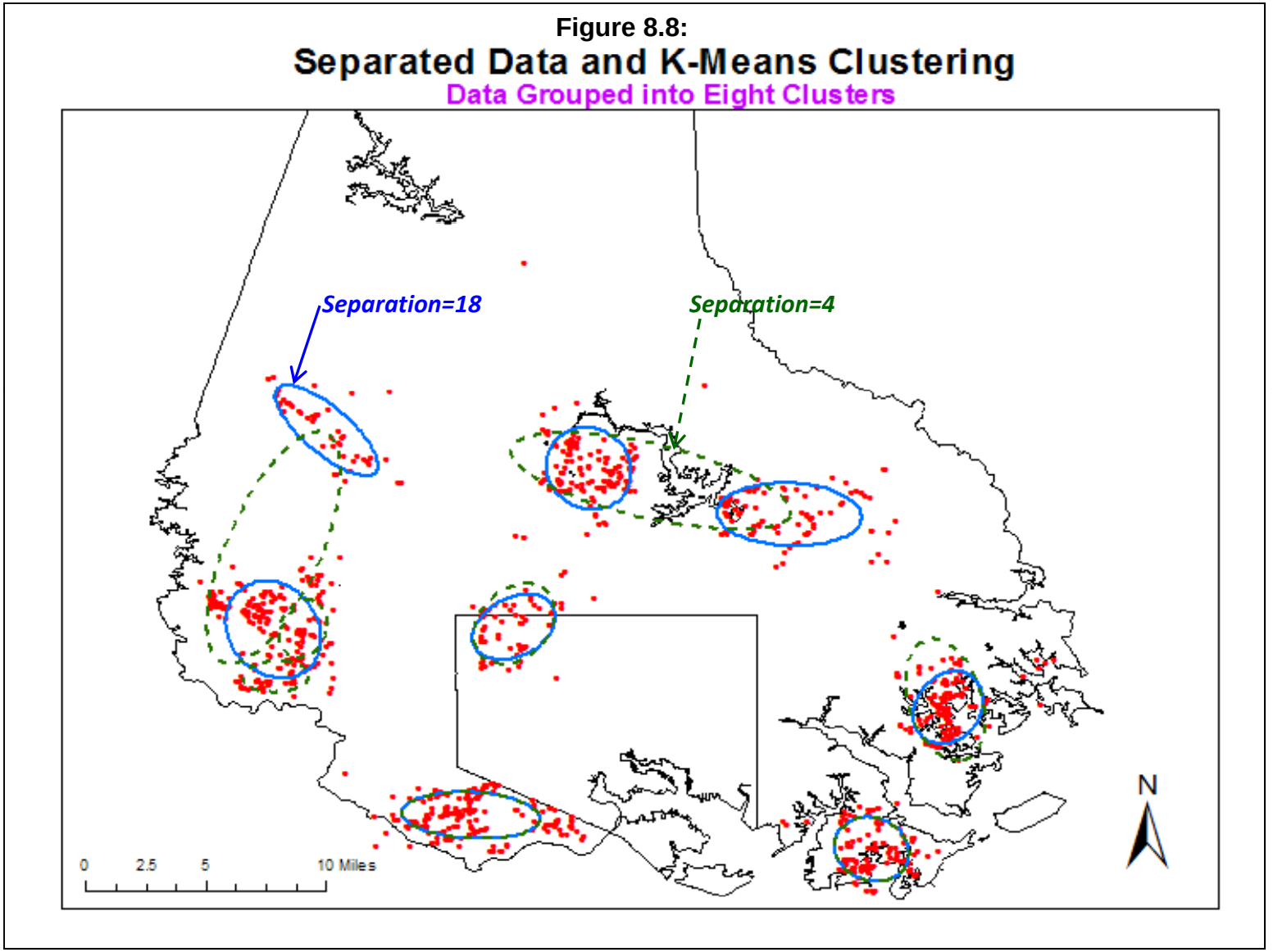


Figure 8.8:


Separated Data and K-Means Clustering
Data Grouped into Eight Clusters
Separation=18
Separation=4
0
2.5
5
10 Miles
N

One has to be careful tweaking the cluster structure, however. For example, as we increased the separation beyond 18, the number of clusters actually decreased. A separation of 20 produced only 7 clusters while a separation of 30 produced only 3. The algorithm could not solve for 8 clusters with such a large separation between them being required.

Selecting the initial seed locations

A second way to control the initial selection of clusters is that the user can define the actual locations for the initial cluster centers. This approach was used by Friedman and Rubin (1967) and Ball and Hall (1970). In *CrimeStat*, the user-defined locations are entered with the secondary file which lists the location of the initial clusters. The routine uses the number of points in the secondary file as *K* and the X/Y coordinates of each point as the initial seed locations. It then proceeds in the same way with local optimization.

When eight points that were approximately in the middle of the eight clusters in Figure 8.7 were input as the secondary file, the K-Means routine immediately identified the eight clusters (results not shown). Again, depending on the purpose the user can test a particular clustering by requiring the routine to consider that model, at least for the initial seed location. The routine will conduct local optimization for the rest of the clustering, as in the above method.

K-Means Screen Output

The K-Means output has both screen and graphical output. The screen output includes the parameters for the 1X standard deviational ellipse of each cluster in the table. In addition, the routine can output graphically the clusters as standard deviational ellipses (1X, 1.5X, or 2X) or convex hulls. The convex hull draws a polygon around all the points in a cluster (see Chapter 4). Hence it is a literal description of the extent of the cluster. The ellipse, on the other hand, is an abstraction for a cluster and may be arranged in an irregular manner. For a small area, a 1X standard deviational ellipse or a convex hull would be a good way to display the ellipses but may not be very visible with a regional view. The user has to balance the need to accurately display the cluster compared to making it easier for a viewer to understand its location.

Mean squared error

In addition, the output for each cluster lists two additional statistics:

$$Sum\ of\ squares\ of\ cluster\ C = SSE_C = \sum_{i=1}^{N_C}[(X_{iC} - \bar{X}_C)^2 + (Y_{iC} - \bar{Y}_C)^2] \qquad (8.3)$$

$$Mean\ squared\ error\ of\ Cluster\ C = MSE_C = \frac{SSE_C}{(N_C - 1)} \quad (8.4)$$

where X_{iC} is the X value of a point that belongs to cluster C, Y_{iC} is the Y value of a point that belongs to cluster C, $MeanX_C$ is the mean X value of cluster C (i.e., of only those points belonging to C), $MeanY_C$ is the mean Y value of cluster C, and N_C is the number of points in cluster C.

There is also a total sum of squares and a total mean square error which is summed over all clusters:

$$Total\ sum\ of\ squares = SS = \sum_{C=1}^{K} SSE_C \quad (8.5)$$

$$Total\ mean\ squared\ error = MSE = \frac{SS}{(N - K - 1)} \quad (8.6)$$

where SSE_C is the sum of squares for cluster C, N is the total sample size, and K is the number of clusters. The sum of squares is the squared deviations of each cluster point from the center of minimum distance while the mean squared error is the average of the squared deviations for each cluster corrected for degrees of freedom.

The sum of squares (or sum of squared errors) is frequently used as a criterion for identifying 'goodness of fit' (Everitt, 2011; Everitt, Landau & Leese, 2001; Aldenderfer & Blashfield, 1984; Gersho & Gray, 1992). In general, for a given number of clusters, *K*, partitions with a smaller sum of squares and, correspondingly, a smaller mean square error are better defined than partitions with a larger sum of squares and larger mean squared error. Similarly, a K-Means solution that produces a smaller overall sum of squares is a tighter grouping than a grouping that produces a larger overall sum of squares.

But, there can be exceptions. If there are points which are outliers, that is which do not obviously fall into one cluster or another, re-assigning them to one or another cluster can distort the sum of squares statistics. Also, in highly concentrated distributions, such as with crime incidents, a smaller sum of squares criteria can be obtained by splitting the concentrations rather than clustering less central and less dense groups of incidents (such as in Figure 8.7). The result, while minimizing the sum of squared errors from the cluster centers, will be less desirable because the peripheral clusters are ignored. Thus, these statistics are presented for the user's information only. In assigning points to clusters, *CrimeStat* still uses the distance to the nearest seed location, rather than a solution that minimizes the sum of squared distances.

K-Means Graphical Output

Finally, the K-Means clustering routine (Kmeans) can output clusters graphically as either ellipses or convex hulls, similar to the other clustering routines. For the ellipses, the user can choose between 1X, 1.5X, and 2X standard deviations. The ellipses are output with the prefix 'KM' before the file name. It should be noted, however, that the ellipses are an abstraction of the cluster. The clusters are *not* necessarily arranged in ellipses. They are for visualization purposes only. For the convex hull, the routine draws a polygon around the points in each cluster. The graphical convex hulls are output with the prefix 'CKM' before the file name.

Naming convention for K-Means clusters

The naming convention for the K-Means outputs is:

Km<*username*>	[for the ellipse]
Ckm<*username*>	[for the convex hull]

where *username* is the name of the file provided by the user. Within the file, each cluster is named

KmEll<N><*username*>	[for the ellipse]
CkmHull<N><*username*>	[for the convex hull]

where *N* is the cluster number and *username* is the name of the file provided by the user. For example,

KmEll3robbery

is the third ellipse for the file called 'robbery' and

CkmHull12burglary

is the 12th convex hull for the file called 'burglary'.

For the ellipses, a slide-bar allows ellipses to be defined for 1X, 1.5X, and 2X standard deviations and can be output in *ArcGIS* '.shp', *MapInfo* '.mif' or various Ascii formats. The convex hulls, on the other hand, draw a polygon around the clustered points.

Example: K-Means Clustering of Baltimore County Street Robberies

In *CrimeStat*, the user specifies the number of groups to sub-divide the data. Using the 1996 robbery incidents for Baltimore County, the data were partitioned into 10 groups with the K-Means routine (Figure 8.9). As can be seen, the clusters tend to fall along the border with Baltimore City. But there are three more dispersed clusters, one concentrated in the central eastern part of the county and two north of the border with the City. Because these clusters are very large, a finer mesh clustering was conducting by partitioning the data into 34 clusters (Figure 8.10). Thirty-five clusters were requested but the routine only found 34 seed location. Consequently, it outputted 34 clusters, which are displayed as ellipses. Though the ellipses are still larger than those produced by the nearest neighbor hierarchical procedure (see figure 7.7 in Chapter 7), there is some congruency; clusters identified by the nearest neighbor procedure have corresponding ellipses using the K-Means procedure.

Figure 8.11 shows a section of southwest Baltimore County with four full clusters and three partial clusters visible. They are displayed as convex hulls. Looking at the distribution, several clusters make intuitive sense while a couple of others do not. For example, the two clusters at the top of the map highlight a concentration along a major arterial (U.S. Highway 40). Similarly, the cluster in the middle right appears to capture incidents along two arterial roads. However, the other three full clusters do not appear to capture meaningful patterns and appear somewhat arbitrary.

Other uses of the K-Means algorithm are possible. For example, one problem that affects most police departments is the need to allocate personnel throughout a city in a balanced and fair way. Too often, some police precincts or districts are overburdened with Calls for Service whereas others have more moderate demand. The issue of re-drawing or re-assigning police boundaries in order to re-establish balance is a continual one for police departments. The K-Means algorithm can help in defining this balance, though there are many other factors that will affect particular boundaries. The number of groupings, K, can be chosen based on the number of police districts that exist or that are desired. The locations of division or precinct stations can be entered in a secondary file in order to define the initial 'seed' locations. The K-Means routine space. Once an agreed upon solution is found, it is easy to then re-assign police beats to fit the new arrangement.

Advantages and Disadvantages of the K-Means Procedure

In short, the K-Means procedure will divide (partition) the data into the number of groups specified by the user, *K*. Whether these groups make any sense or not will depend on how carefully the user has selected clusters. Choosing too many will lead to defining patterns that do

Figure 8.9:

Baltimore County Robbery Hot Spots: 1996

K-Means Clustering with K=10

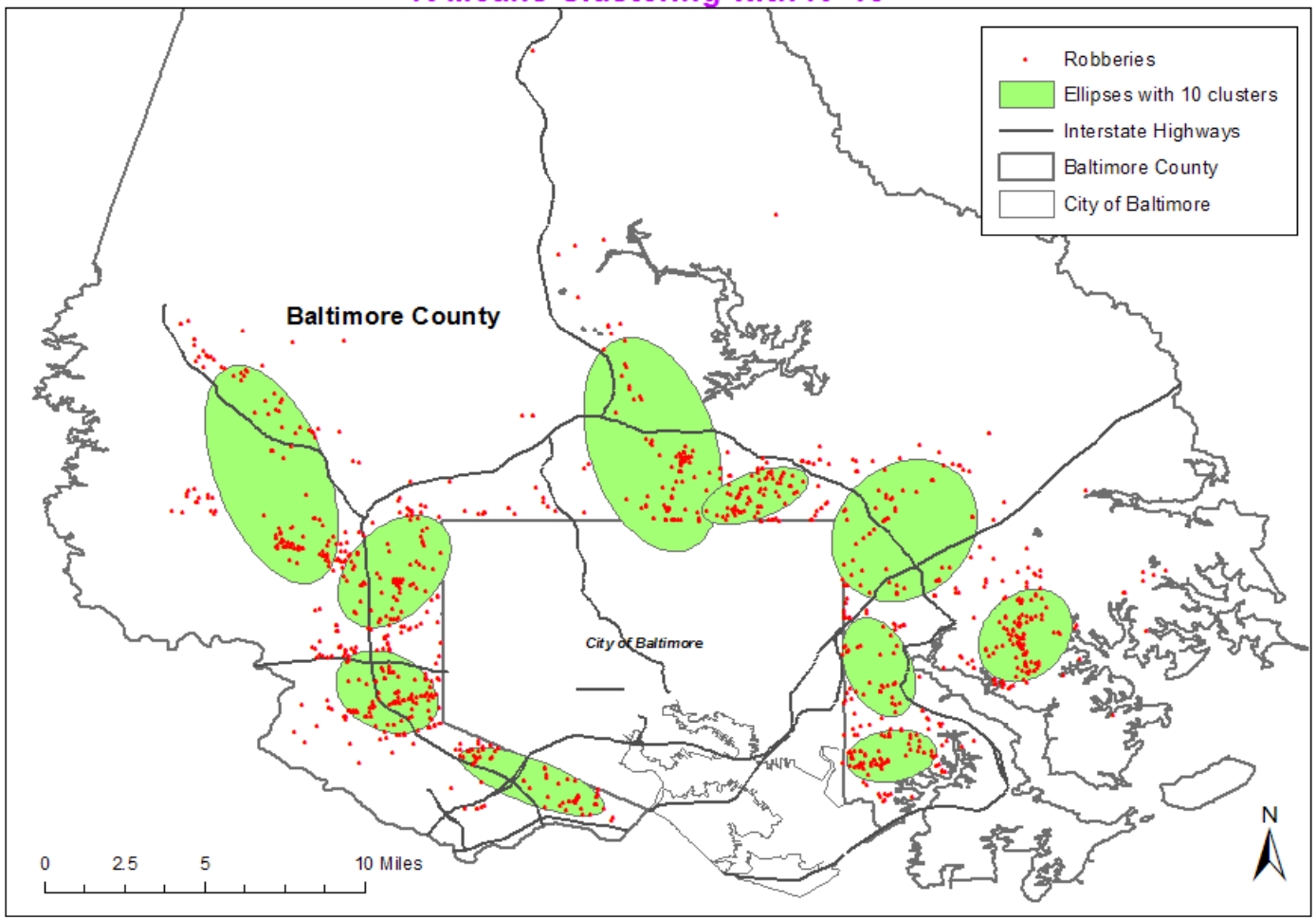


Figure 8.10:

Baltimore County Robbery Hot Spots: 1996

K-Means Clustering with K=34

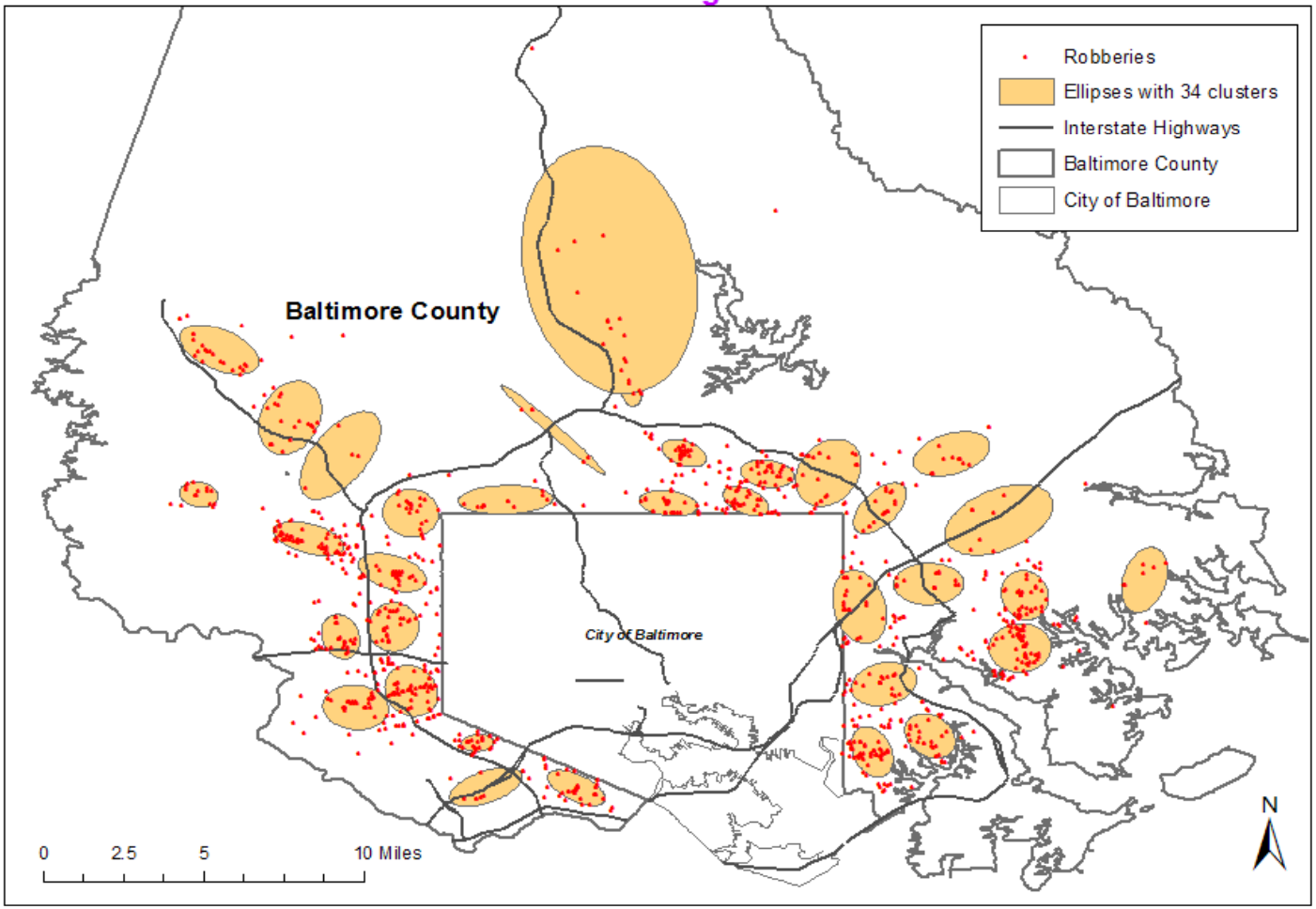
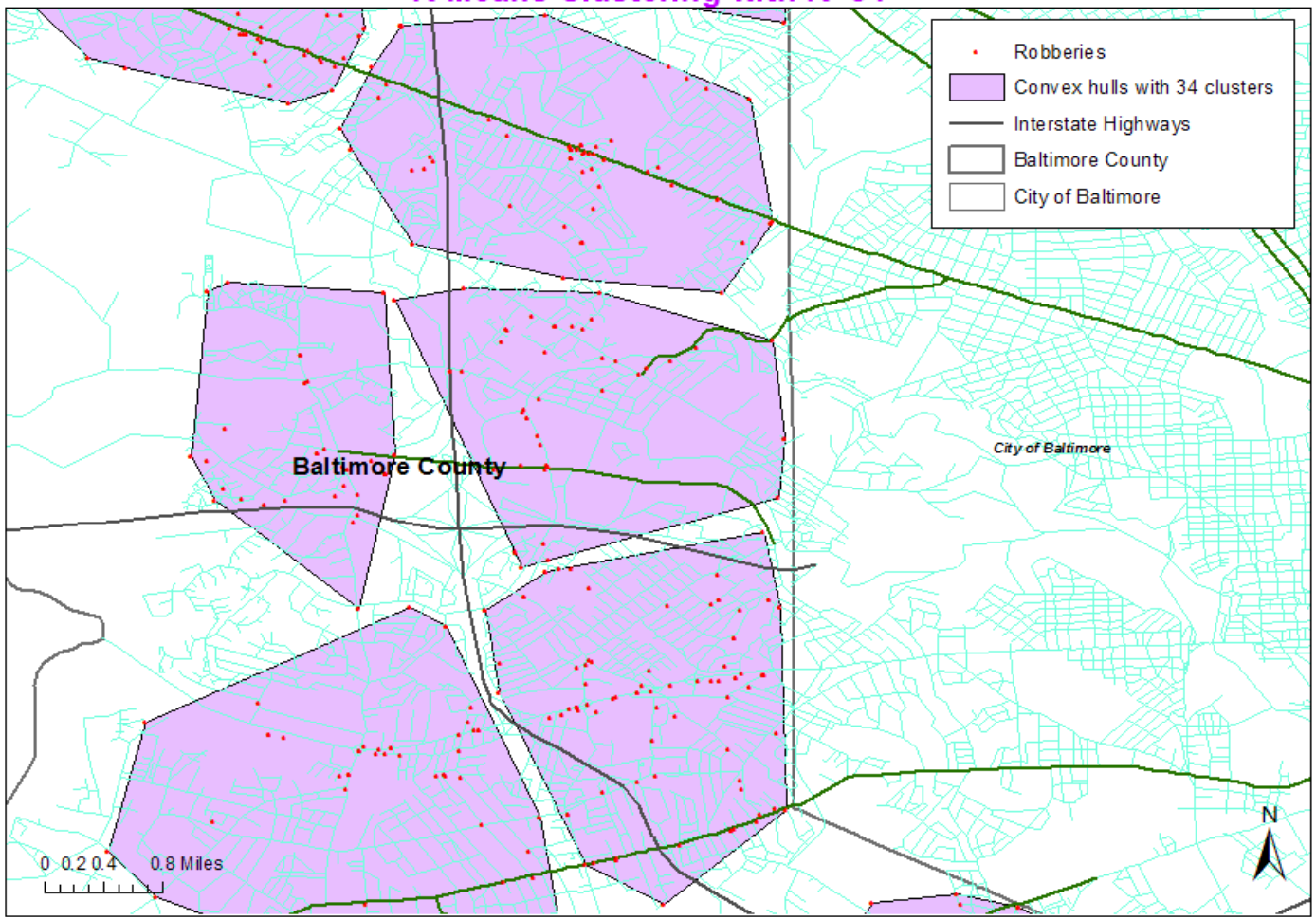


Figure 8.11:

Southwest Baltimore County Robbery Hot Spots: 1996

K-Means Clustering with K=34



not really exist whereas choosing too few will lead to poor differentiation among neighborhoods that are distinctly different.

This choice is both a strength and weakness of the technique. The K-Means procedure provides a great deal of control for the user and can be used as an exploratory tool to identify possible hot spots. Whereas the nearest neighbor hierarchical method produces a solution based on geographical proximity with most clusters being very small and STAC identifies autonomous areas of high density, the K-Means can allow the user to control the size of the clusters. In terms of policing, the K-Means is better suited for defining larger geographical areas than the nearest neighbor method, perhaps more appropriate for a patrol area than for a particular hot spot. Again, if carefully used, the K-Means gives the user the ability to fine tune a particular model of hot spots, adjusting the size of the clusters (vis-a-via the number of clusters selected) as well as their separation in space in order to fit a particular pattern which is known.

Yet it is this same flexible characteristic that makes the technique potentially difficult to use and prone to misuse. Since the technique will divide the data set into *K* groups, there is no assumption that these *K* groups represent real hot spots or not. A user cannot just arbitrarily put in a number and expect it to produce meaningful results. A more extensive discussion of this issue can be found in Murray and Grubesic (2002). Grubesic and Murray (2001) present some newer approaches in the K-Means methodology.

The technique is, therefore, better seen as both an exploratory tool as well as a tool for refining a hot spot search. If the user has a good idea of where there should be hot spots, based on community experience and the reports of beat officers, then the technique can be used to see if the incidents actually correspond to the perception. It also can help identify hot spots which have not been perceived or identified by officers. Alternatively, it can identify hot spots that do not really exist and which are merely by-products of the statistical procedure. Experience and sensitivity are needed to know whether an identified hot spot is real or not.

Some Thoughts on the Concept of Hot Spots

Advantages of the Concept

The six techniques discussed in this and the last chapter have both advantages and disadvantages. Among the advantages are that they attempt to isolate areas of high concentration of incidents and can, therefore, help law enforcement agencies focus their resources on these areas. One of the powerful uses of a hot spot concept is that it is focused. It can provide new information about locations that police officers or community workers may not recognize

(Rengert, 1995). Given that most police departments are understaffed, a strategy that prioritizes intervention is very appealing. The hot spot concept is imminently practical.

Another advantage to the identification of hot spots is that the techniques systematically implement an algorithm. In this sense, they minimize bias on the part of officers and analysts since the technique operates somewhat independently of preconceptions. As has been mentioned, however, these techniques are not totally without human judgment since the user must make decisions on the number of hot spots and the size of the search radius, choices that can allow different users to come to different conclusions. There is probably no way to get around subjectivity since law enforcement personnel may not use a result unless it partly confirms what they already know. But, by implementing an algorithm, it forces users to at least go through the steps systematically.

A third advantage is that these techniques are visual, particularly when used with a GIS. The mode and fuzzy mode routines output the results as a dbf file, which can be displayed in a GIS as a proportional circle. The Nnh, Rnnh, Stac, and Kmeans routines can output the results directly as graphical objects, either as standard deviational ellipses or as convex hull, which can be displayed directly in a GIS. Visual information can help crime analysts and officers to understand the distribution of crime in an areas, a necessary step in planning a successful intervention. We should never underestimate the importance of visualization in any analysis.

Limitations of the Concept

However, there are also some distinct limitations to the concept of a hot spot, some technical and some theoretical. The choice involved in a user making a decision on how strict or how loose to create clusters allows the potential for subjectivity, as has been mentioned. In this sense, isolating clusters (or hot spots) can be as much an art as it is a science. There are limits to this, however. As the sample size goes up, there is less difference in the result that can be produced by adjusting the parameters. For example, with 6,000 or more cases, there is very little difference between using the 0.1 significance level in the nearest neighbor clustering routine and the 0.001 significance level.[8] Thus, the subjectivity of the user is more important for smaller samples than larger ones.

8 On one test of 6,051 burglaries with a minimum cluster size requirement of 10 incidents, for example, we obtained 100 first-order clusters, 9 second-order clusters, and no third-order clusters by using a 0.1 significance level for the nearest neighbor hierarchical clustering routine. When the significance level was reduced to 0.001, the number of clusters extracted was 97 first-order clusters, 8 second-order clusters, and no third-order clusters.

A second problem with the hot spot concept is that it is usually applied to the volume of incidents and not to the underlying risk. Clusters (or hot spots) are defined by a high concentration of incidents within a small geographical area, that is, on the volume of incidents within an area. This is an implicit *density* measure - the number of incidents per unit of area (e.g., incidents per square mile). But higher density can also be a function of a higher population at risk.

For some policing policies, this is fine. For example, beat officers will necessarily concentrate on high incident density neighborhoods because so much of their activity revolves around those neighborhoods. From a viewpoint of providing concentrated policing, the density or volume of incidents is a good index for assigning police officers (Sherman and Weisburd, 1995). From the viewpoint of ancillary security services, such as access to emergency medical services, neighborhood watch organizations, or residential burglar alarm retail outlets, areas with higher concentrations of incidents may be a good focal point for organizing these services.

But for other law enforcement policies, a density index is not a good one. From the viewpoint of crime prevention, for example, high incident volume areas are not necessarily unsafe and that effective preventive intervention will not necessarily lead to reduction in crime. It may be far more effective to target high risk areas rather than high volume areas. In high risk areas, there are special circumstances which expose the population to higher-than-expected levels of crime, perhaps particular concentrations of activities (e.g., drug trading) or particular land uses that encourage crime (e.g., skid row areas) or particular concentrations of criminal activities (e.g., gangs). A prevention strategy will want to focus on those special factors and try to reduce them.

Risk, which is defined as the number of incidents relative to the number of potential victims/targets, is only loosely correlated with the volume of incidents. Yet, hot spots are usually defined by volume, rather than risk. The risk-adjusted hierarchical nearest neighbor clustering routine, discussed in Chapter 7, is the only tool among these that identifies risk, rather than volume. It is clear that more tools will be needed to examine hot spot locations that are more at risk.

The final problem with the hot spot concept is more theoretical. Namely, given a concentration of incidents, how do we explain it? To identify a concentration is one thing. To know how to intervene is another. It is imperative that the analyst discover some of the underlying causes that link the events together in a systematic way. Otherwise, all that is left is an empirical description without any concept of the underlying causes. For one thing, the concentration could be random or haphazard; it could have happened one time, but never again. For another, it could be due to the concentration of the population *at risk*, as discussed above.

But, it could also be due to the concentration of activities that attract offenders along with victims. In Chapter 14 and, again, in Chapter 28, we examine locations where offenders are attracted. Many of these are shopping malls, which is where a lot of crime occurs. Thus, the hot spot could be a destination as much an origin variable. Finally, the concentration could be circumstantial and not be related to anything inherent about the location.

The point here is that an empirical description of a location where crime incidents are concentrated is only a first step in defining a real 'hot spot'. It is an *apparent* 'hot spot'. Unless the underlying vector (cause) is discovered, it will be difficult to provide adequate intervention. The causes could be environmental (e.g., concentrations of land uses that attract attackers and victims) or behavioral (e.g., concentrations of gangs). The most one can do is try to increase the concentration of police officers. This is expensive, of course, and can only be done for limited periods. Eventually, if the underlying vector is not dealt with, incidents will continue and will overwhelm the additional police enforcement. In other words, ultimately, reducing crime around a 'hot spot' will need to involve many other policies than simply police enforcement, such as community involvement, gang intervention, land use modification, job creation, the expansion of services, and other community-based interventions. In this sense, the identification of an empirical 'hot spot' is frequently only a window into a much deeper problem that will involve more than targeted enforcement.

References

Aldenderfer, M. & Blashfield, R. (1984). *Cluster Analysis*. Sage: Beverly Hills, CA.

Ball, G. H. & Hall, D. J. (1970). A clustering technique for summarizing multivariate data. *Behavioral Science*, 12, 153-155.

Barnard, G. A. (1963). Comment on 'The Spectral Analysis of Point Processes' by M. S. Bartlett, *Journal of the Royal Statistical Society*, Series B, 25, 294.

Block, C. R. (1994). STAC hot spot areas: a statistical tool for law enforcement decisions. In *Proceedings of the Workshop on Crime Analysis Through Computer Mapping*. Criminal Justice Information Authority: Chicago, IL.

Block, R. & Block, C. R. (1999) Risky places: a comparison of the environs of rapid transit stations in Chicago and the Bronx in John Mollenkopf (ed), *Analyzing Crime Patterns: Frontiers of Practice*, Sage Publishing: Beverly Hills, CA.

Block, R. & Block, C. R. (1995). Space, place and crime: hot spot areas and hot places of liquor-related Crime,. In Eck, J. E. & Weisburd, D. (eds.), *Crime and Place*. Crime Prevention Studies, Volume 4. Criminal Justice Press: Monsey, NY, 147-185.

Chainey, S., Thompson, L. & Uhlig, S. (2008). The utility of hotspot mapping for predicting spatial patterns of crime. *Security Journal*, **21**, 4-28.

Dwass, M (1957). Modified randomization tests for nonparametric hypotheses. *Annals of Mathematical Statistics*, 28, 181-187.

Eick, C. F., Zeidat, N. & Zhao, Z. (2004). Supervised clustering: Algorithms and applications. proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI04) , Boca Raton, FL>

Everitt, B. S. (2011). *Cluster Analysis* (5th edition). J. Wiley: London.

Everitt, B. S., Landau, S. & Leese, M. (2001). *Cluster Analysis*. 4th Edition. Oxford University Press: New York.

References (continued)

Finley, T. & Joachims, T. (2005). Supervised clustering with support vector machines. *Proceedings of the 22nd International Conference on Machine Learning*. Bonn, Germany.

Fisher, W. (1958). On grouping for maximum homogeneity. *Journal of the American Statistical Association*. **53**, 789-798.

Friedman, H. P. & Rubin, J. (1967). On some invariant criteria for grouping data, *Journal of the American Statistical Association*, **62**, 1159-1178.

Gersho, A. & Gray, R. (1992). *Vector Quantization and Signal Compression*. Kluwer Academic Publishers: Dordrecht, Netherlands.

Grubesic, T. H. & Murray, A. T. (2001). Detecting hot spots using cluster analysis and GIS. Paper presented at Annual Conference of the Crime Mapping Research Center, Dallas, TX. http://www.ojp.usdoj.gov/cmrc.

Haggett, P., Cliff, A. D. & Frey, Allan (1977). *Locational Analysis in Human Geography* (2nd edition). Edward Arnold: London.

Kulldorff, M. (1997). A spatial scan statistic, *Communications in Statistics - Theory and Methods*, 26, 1481-1496.

Levine, N. (2008). "The 'hottest' part of a crime hotspot: Comments on "The utility of hotspot mapping for predicting spatial patterns of crime" by Spencer Chainey, Lisa Tompson, and Sebastian Uhlig". *Security Journal*, 21, 295-302.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *5th Berkeley Symposium on Mathematics, Statistics and Probability*. Vol 1, 281-298.

Murray, A.T. & Grubesic, T. H. 2002. Identifying Non-hierarchical Clusters. *International Journal of Industrial Engineering*, 9, 86-95.

Openshaw, S. A., Craft, A. W., Charlton, M., & Birch, J. M. (1988). Invetigation of leukemia clusters by use of a geographical analysis machine, *Lancet*, 1, 272-273.

References (continued)

Openshaw, S. A., Charlton, M., Wymer, C. & Craft, A. W. (1987). A mark 1 analysis machine for the automated analysis of point data sets, *International Journal of Geographical Information Systems*, 1, 335-358.

Rengert, G. F. (1995). Comparing cognitive hot spots to crime hot spots. In Carolyn Rebecca Block, C. R., Dabdoub M. & Fregly, S. *Crime Analysis Through Computer Mapping*. Police Executive Research Forum: Washington, DC, 33-47.

Sherman, L. W. & Weisburd, D. (1995). General deterrent effects of police patrol in crime hot spots: a randomized controlled trial. *Justice Quarterly*. 12, 625-648.

Systat, Inc. (2008). *Systat 13: Statistics I*. SPSS, Inc.: Chicago.

Turnbull, B. W., Iwano, E.J., Burnett, W. S., Howe, H. L. & Clark, L. C. (1990). Monitoring for clusters of disease: application to leukemia incidence in upstate New York, *American Journal of Epidemiology*, 132, S136-S143.

Endnotes

i. The steps are as follows:

Global Selection of Initial Seed Locations

A. A 100 x 100 grid is overlaid on the point distribution; the dimensions of the grid are defined by the minimum and maximum X and Y coordinates.

B. A separation distance is defined, which is:

$$Separation = t * 0.5\sqrt{\frac{A}{N}}$$

where *t* is the Student's t-value for the .01 significance level (2.358), *A* is the area of the region, and *N* is the sample size. The separation distance was calculated to prevent adjacent cells from being selected as seeds.

C. For each grid cell, the number of incidents found are counted and then sorted in descending order.

D. The cell with the highest number of incidents found is the initial seed for cluster 1.

E. The cell with the next highest number of incidents is temporarily selected. If the distance between that cell and the seed 1 location is *equal to or greater than* the separation distance, this cell becomes initial seed 2.

F. If the distance is less than the separation distance, the cell is dropped and the routine proceeds to the cell with the next highest number of incidents.

G. This procedure is repeated until *K initial seeds* have been located thereby selecting the remaining cell with the highest number of incidents and calculating its distance to all prior seeds. If the distance is equal to or greater than the separation distance, then the cell is selected as a seed. If the distance is less than the separation distance, then the cell is dropped as a seed candidate. Thus, it is possible that *K* initial seeds cannot be identified because of the inability to locate *K* locations greater than the threshold distance. In this case, *CrimeStat* keeps the number it has located and prints out a message to this effect.

Local Optimization of Seed Locations

H. After the *K* initial seeds have been selected, all points are assigned to the nearest initial seed location. These are the initial cluster groupings.

I. For each initial cluster grouping in turn, the center of minimum distance is calculated. These are the second seed locations.

J. All points are assigned to the nearest second seed location.

K. For each new cluster grouping in turn, the center of minimum distance is calculated. These are third seed locations.

L. Steps J and K are repeated until no more points change cluster groupings. These are the final seed locations and cluster groupings.

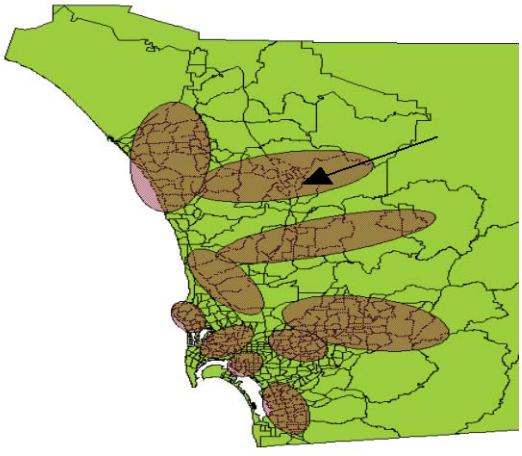
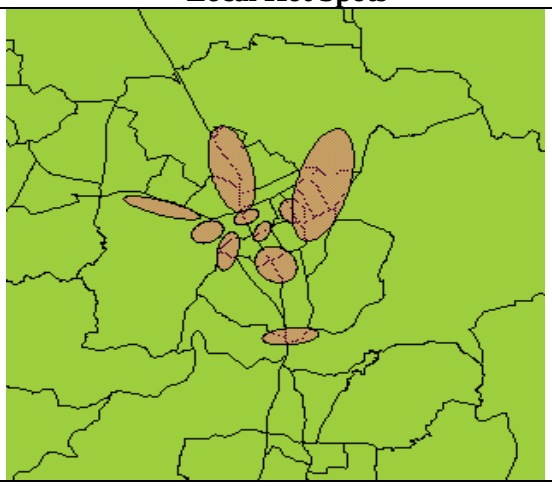
Attachments

K-Means Clustering as an Alternative Measure of Urban Accessibility

Richard J. Crepeau
Department of Geography and Planning
Appalachian State University
Boone, NC

The relationship between land use and the transportation system is an important issue. Many planners recognize that transportation policies, practices and outcomes affect changes in land use, and vice versa, but there is disagreement as to how best to describe this phenomenon. Traditional methods include measures of accessibility via a matrix of zones (tracts, traffic analysis zones, etc.). However, there are limits to the way interaction and accessibility is described with such discrete units.

Through the use of K-Means clustering, an alternate measure of accessibility can be calculated. Rather than relying on census geography, the left map shows ten retail clusters in San Diego County (1995) as calculated by *CrimeStat*'s K-Means clustering technique (using 1x standard deviational ellipse). The retail hot spots were calculated using a geocoded point file of retail establishments in the county. These clusters are not bound by census geography and allow a more realistic appraisal about the attractiveness of specific regions within the county. An analyst can then determine if residential location within a hot spot has an effect on travel patterns, or if there is a relationship between proximity to a hot spot and travel behavior. While this example illustrates a measure of regional retail attractiveness, the flexibility of *CrimeStat* allows an analyst to evaluate these relationships on a local level, thus allowing a scope of inquiry from regional to local accessibility (as shown in right map, which uses the same parameters as the left figure, but limiting its sample to retail in a sub-region of San Diego County noted by the arrow).

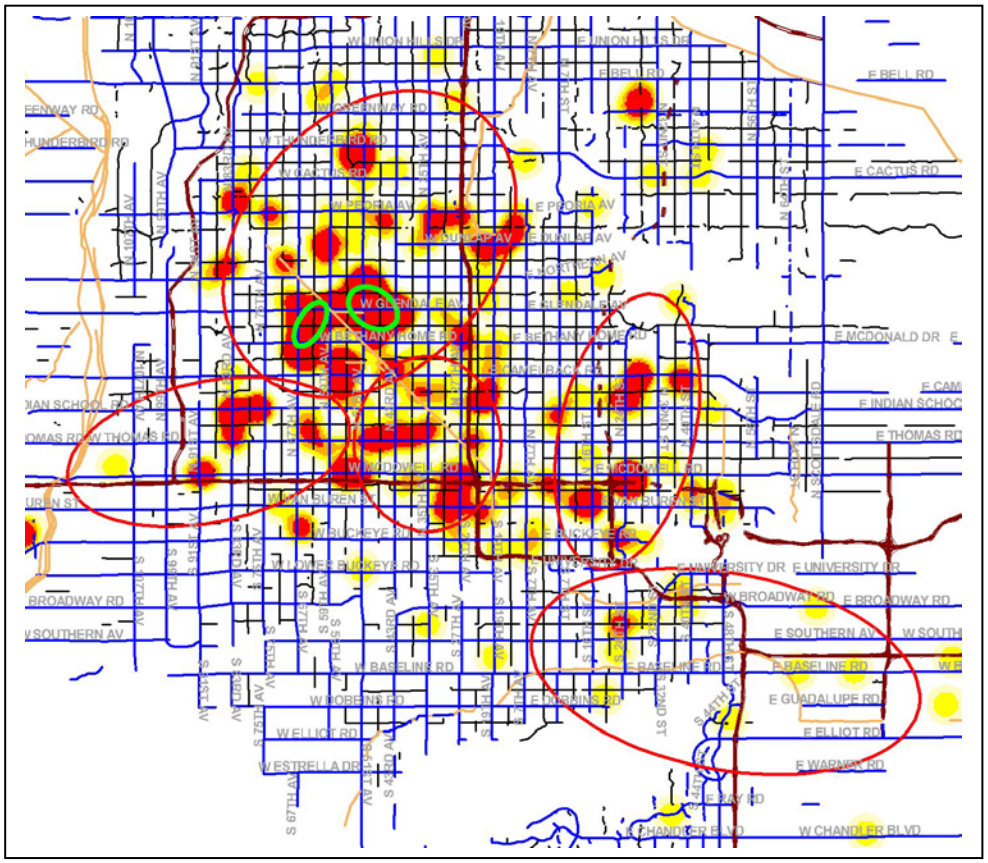
Regional Hot Spots	Local Hot Spots
	

Hot Spot Verification in Auto Theft Recoveries

Bryan Hill
Glendale Police Department
Glendale, AZ

We use *CrimeStat* as a verification tool to help isolate clusters of activity when one application or method does not appear to completely identify a problem. The following example utilizes several *CrimeStat* statistical functions to verify a recovery pattern for auto thefts in the City of Glendale (AZ). The recovery data included recovery locations for the past 6 months in the City of Glendale which were geocoded with a county-wide street centerline file using *ArcView*.

First, a spatial density "grid" was created using *Spatial Analyst* with a grid cell size of 300 feet and a search radius of 0.75 miles for the 307 recovery locations. We then created a graduated color legend, using standard deviation as the classification type and the value for the legend being the *CrimeStat* "Z" field that is calculated.



In the map, the K-means (red ellipses), Nnh (green ellipses) and *Spatial Analyst* grid (red-yellow grid cells) all showed that the area was a high density or clustering of stolen vehicle recoveries. Although this information was not new, it did help verify our conclusion and aided in organizing a response.

Chapter 9:

Hot Spot Analysis of Zones

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 9:
Hot Spot Analysis of Zones

In this chapter, we will discuss methods for identifying hot spots with zonal data. The user should be thoroughly familiar with the information presented in Chapter 5 on spatial autocorrelation indices because two of the same indices are used for the analysis of local variations in zones.

We are going to look at four techniques for analyzing hot spots with zonal data or with individual level data that have attributes (count or interval variables that measure a characteristic associated with the X and Y coordinates). These are Anselin's Local Moran, the Getis-Ord Local "G", the Zonal Nearest Neighbor Hierarchical Clustering algorithm, and the Risk-adjusted Zonal Nearest Neighbor Hierarchical Clustering algorithm. Figure 9.1 shows the Hot Spot Analysis of Zones page.

Assigning Point Data to Zones

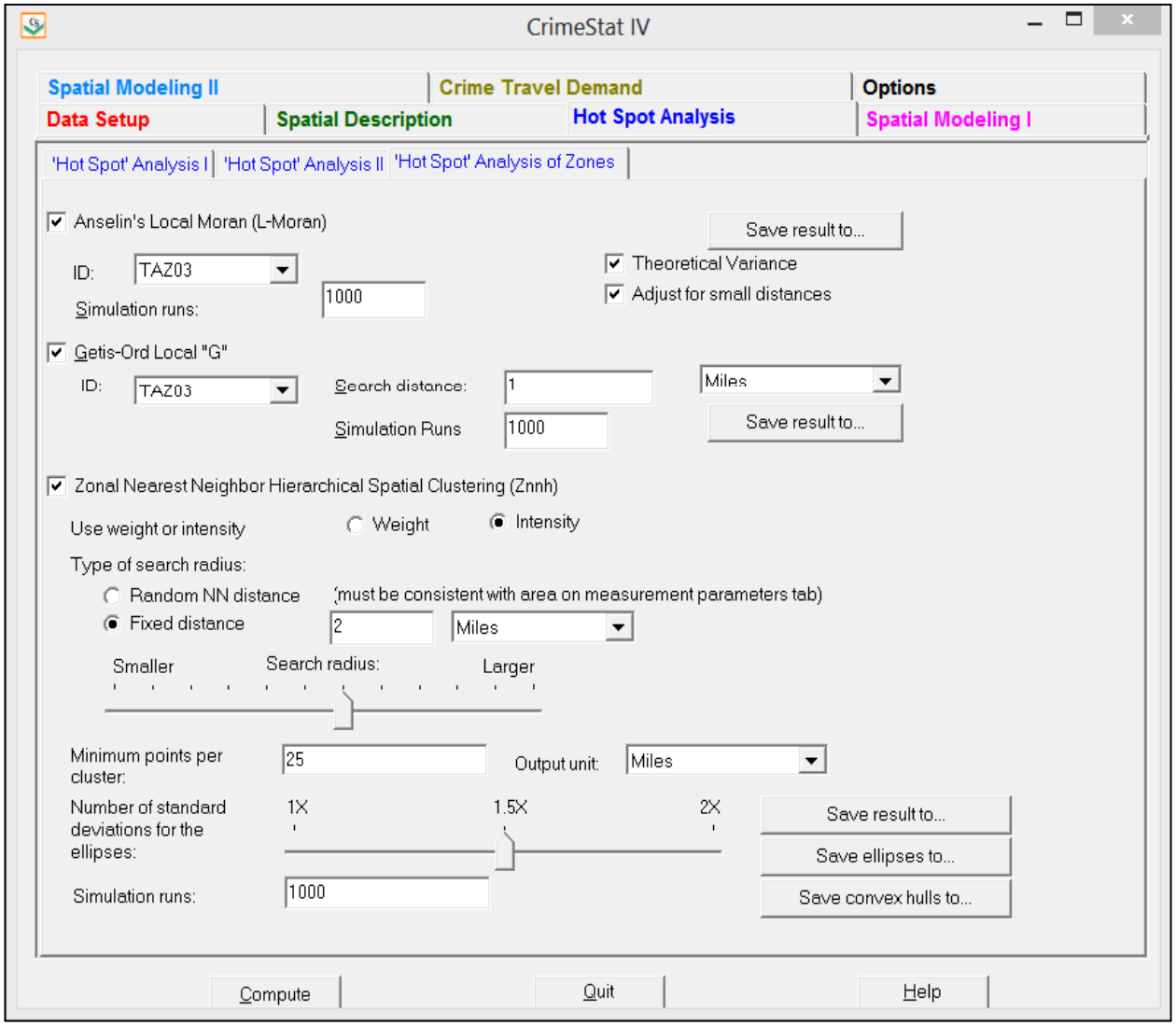
If a user has information on the location of individual events (e.g., robberies), then it is better to utilize that information with the hot spot techniques discussed in Chapters 7 and 8. The individual-level information will contain all the uniqueness of the events.

However, sometimes it is not possible to analyze data at the individual level. The user may need to aggregate individual data points to spatial areas (zones) in order to compare the events to data that are only obtained for zones, such as census data, or to model environmental correlates of the data points or may find that individual data are not available (e.g., when a police department releases information by police beats but not individual streets). Zonal data can include crime counts by zone, socio-economic information (e.g., collected by the census or estimated by a Metropolitan Planning Organization), or some other data that are aggregated to the small areas. In other words, the zone becomes the unit of analysis instead of the individual data points.

Since the zones are not events, they have to be spatially analyzed by assuming that all the data resides at a single point within the zone. This is usually the centroid (the geographical center of the zone) but sometimes the center of minimum distance (the point at which the sum of the distances to all other points is minimized) has been used, too, especially if the zone is very irregularly shaped. However, when individual data points are assigned to zones, information is lost. For example, the distance between zones is a singular value for all the points in those zones whereas there is much greater variability with the distances between individual events. Also,

Figure 9.1:

Hot Spot Analysis of Zones Screen



topological information, such as the shape of the zone or the number of other zones that are adjacent, is lost.

For the spatial autocorrelation indices, the interaction between zones is defined by distance. There are advantages and disadvantages. Contiguity (or adjacency) is a property of a zone, not a point. Thus, adjacency defines whether one zone is next to another zone whereas distance is the distance between single points that represent the zones (e.g., centroids). For example, if two zones are 0.25 miles apart, it is not known whether they are adjacent or not. In other words, in adopting a distance-based weight, information about adjacencies is lost. On the other hand, a distance-based weight is standardized. If two zones are adjacent, it is not known how far apart they are separated. Adjacencies can be misleading since they do not indicate the size of the adjacent zones whereas a specified distance is always constant.

The zonal data also must include an *attribute* variable, a variable associated with the zone (e.g., number of robberies; median household income; percentage of households living below poverty level). The attribute can be a *count* or a continuous variable for a distributional property of the zone (e.g., median household income; percentage of households below poverty level) or even a binary variable (e.g.,1 v. 0).[1] The indices discussed in this chapter are applied to the interaction between the attribute variable of the central zone and other zones, weighted by the distance between them.

Individual level data can also have attributes. For example, Levine and Lee (2013) analyzed journey-to-crime distances for offenders in Manchester, England. In this case, the attribute variable was the distance traveled and the statistics discussed in this chapter are appropriate for analyzing that attribute data. Other examples of individual level data with attributes would be the age of the offender, the number of prior convictions, or the number of years of formal education. The key criterion is that the records must have an attribute which is either a count or an interval variable.

Local Indicator of Spatial Association

The basic concept behind a zone-specific measure of spatial autocorrelation is that of a *local indicator of spatial association* (*LISA*) and has been discussed by a number of researchers (Mantel, 1967; Getis, 1991; Anselin, 1995). For example, Anselin (1995) defines this as any statistic that satisfies two requirements:

---

[1] There is no fundamental difference between a count variable and a continuous interval or ratio variable since a real number can be converted into a count by multiplying by a power of 10 (e.g., $1.23 = 123 \times 10^{-2}$). The statistics discussed in this chapter are applicable to either count or continuous data.

1. The *LISA* for each observation indicates the extent to which there is significant spatial clustering of similar values around that observation; and

2. The sum of the *LISA*s for all observations is proportional to the global indicator of spatial association:

$$L_i = f\,g(Y_i)\sum_{ji=1}^{K} h(Y_{ji}) \qquad (9.1)$$

where L_i is the local indicator of zone i, $g(Y_i)$ is a function of the value of an intensity variable, Y_i, at location i, $h(Y_{ji})$ is a weight function of the values of the intensity variable observed in the neighborhood j_i of i, and f is a scaling constant to ensure that the sum of L_i equals the global spatial autocorrelation index.

The function of the intensity variable can be a raw score, Y_i, a Z-transformation of the intensity variable, such as:

$$Z_i = \frac{(Y_i - \bar{Y})}{S_Y} \qquad (9.2)$$

where $\bar{Y}$ is the mean of Y and S_Y is the standard deviation of variable Y, or some other function.

In other words, a *LISA* is an indicator of the extent the value of an observation is affected by its neighboring observations. This requires two conditions. The first is that each observation has a value of an attribute variable that can be assigned to it (i.e., an intensity or weight value) in addition to its X and Y coordinates. For crime incidents, this means data must be aggregated into zones (e.g., number of incidents by census tracts, zip codes, or police reporting districts).

Second, the *neighborhood* has to be defined. This could be either adjacent zones, all other zones negatively weighted by the distance from the observation zone, or all other zones negatively weighted by the distance from the observation zone up to some distance whereupon the weight is zero afterward (a bandwidth). Once these are defined, the *LISA* indicates the value of the observation zone in relation to its neighborhood.

Anselin's Local Moran

Anselin's Local Moran statistic was developed by Luc Anselin and is the oldest LISA statistic (Anselin, 1995). The procedure applies Moran's "I" statistic to individual zones (see Chapter 5), allowing them to be identified as similar or different to their nearby pattern.

The definition of "I_i" is from Getis and Ord (1996):

$$I_i = \frac{(Z_i - \bar{Z})}{S_Z^2} \sum_{j=1}^{N-1} [W_{ij}(Z_j - \bar{Z})] \quad (9.3)$$

where Z_i is the intensity of observation i, $\bar{Z}$ is the mean intensity over all observations, Z_j is intensity for all other observations, j (where j ≠ i), S_Z^2 is the variance over all observations, and W_{ij} is a distance weight for the interaction between observations i and j. The first term in equation 9.3 refers only to observation i while the second term is the sum of the weighted values for all other observations (but not including i itself).

The expected "I_i" is defined as:

$$E(I_i) = \frac{\sum_{i=1}^{N} W_{ij}}{N-1} \quad (9.4)$$

where Wij is the distance weight for the interaction between observations i and i. The variances of I_i are somewhat complicated (see endnote *i* for the formulas).

Similarity or Dissimilarity

Since the global Moran's "I" statistic measures similarity in observations over a study area (see Chapter 5), the local Moran "I_i" also indicates the similarity of a zone relative to its neighbors. Thus, in neighborhoods where both the zone and its neighbors have high attribute values, the Local Moran will be positive indicating that the particular zone is similar (i.e., also 'high'). Similarly, in neighborhoods where both the zone and its neighbors have 'low' attribute values, the Local Moran also will be positive indicating that the zone is similar to its neighbors (i.e., also 'low'). When the Local Moran statistic is positive, this is an indicator of *similarity*, not absolute value of the intensity variable.

Conversely, if a zone has a high value of the intensity variable while its neighbors have low values or, alternatively, it has a low value while the neighbors have high values, then the Local Moran statistic will be negative. *Dissimilarity* is an indicator of either a hot spot or a cold spot, in other words zones that are different from their neighborhood. Hot spots would be seen if the number of incidents in a zone is much higher than in the nearby zones. Cold spots would be seen if the number of incidents in a zone is much lower than in the nearby zones.

In other words, the Local Moran statistic indicates whether the zone is similar or dissimilar to its neighbors.

ID Field

The user should indicate a field for the ID of each zone. This ID will be saved with the output and can then be linked with the input file (Primary File) for mapping.

Distance Weights

The weights, W_{ij}, can be either an indicator of the adjacency of a zone to the observation zone (i.e., '1' if adjacent; 0 if not adjacent) or a distance-based weight which decreases with distance between zones i and j. Adjacency indices are useful for defining near neighborhoods; the adjacent zones have full weight while all other zones have no weight. Distance weights, on the other hand, are useful for defining spatial interaction; zones which are farther away can have an influence on an observation zone, although one that is much less. *CrimeStat* uses distance weights, in two forms.

First, there is a traditional distance decay function:

$$W_{ij} = \frac{1}{d_{ij}} \tag{9.5}$$

where d_{ij} is the distance between the observation zone, i, and another zone, j. For example, a zone which is two miles away has half the weight of a zone that is one mile away.

Small distance adjustment

Second, there is an adjustment for small distances. The weight index becomes problematic with small distance between zones since the weight will approach infinity for d_{ij} -> 0. To correct for this, the routine includes an adjustment for small distances so that the maximum weight can be never be greater than 1.0 (see Chapter 5). The adjustment scales distance to one mile, which is a typical distance for crime analysis. When the small distance adjustment is turned on, the minimal distance is scaled automatically to be one mile. The formula used is:

$$W_{ij} = \frac{one\ mile}{one\ mile + d_{ij}} \tag{9.6}$$

in whichever distance units are specified (miles, kilometers, etc).

Output for Each Zone

The output is for each zone includes:

1. The sample size
2. The ID identifier
3. The X coordinate
4. The Y coordinate
5. The "I_i" value
6. The expected "I_i".

If the variance box is checked, the program will also calculate the variance, standard error, and a Z-test of "I_i" for each zone. The default is for the variance not to be calculated.

Simulation of Confidence Intervals for Anselin's Local Moran

There are two ways to estimate confidence intervals for Anselin's Local Moran. First, the routine can calculate the variance and, for each zone, the standardized "I_i" score to produce a Z-test of the significance of the "I_i". Assuming the sample size is greater than 120, 95% percent confidence intervals can be estimated by:

$$95\%\ confidence\ intervals = I_i \pm 1.98SE_i \tag{9.7}$$

and 99% confidence intervals can be estimated by:

$$99\%\ confidence\ intervals = I_i \pm 2.58SE_i \tag{9.8}$$

One problem with this test is that "I_i" may not actually follow a normal standard distribution. That is, if "I_i" is calculated for all zones with random data, the distribution of the statistic may not be (and often will not be) normally distributed. This would be especially true if the variable of interest, Z, is skewed with some zones having very high values while the majority having low values, as is typically true with crime distributions.

Second, the user can estimate confidence intervals (called *credible intervals*) using a Monte Carlo simulation. A *permutation* type simulation is run whereby the locations of the zones are kept and the original values of the intensity variable, Z, are maintained but randomly re-assigned to zones for each simulation run. This will maintain the structure of the attribute "Z" variable but will estimate the value of "I_i" for each under random assignment of this variable.

> Note that a simulation may take time to run especially if the data set is large or if a large number of simulation runs are requested.

If a permutation Monte Carlo simulation is run to estimate credible intervals, specify the number of simulations to be run (e.g., 1,000, 5,000, 10000). In addition to the "I_i" for each zone, the expected "I_i" and the variance (if requested), the output includes the results that were obtained by the simulation for:

1. The minimum "I_i" value
2. The maximum "I_i" value
3. The 0.5 percentile of "I_i"
4. The 2.5 percentile of "I_i"
5. The 97.5 percentile of "I_i"
6. The 99.5 percentile of "I_i"

The two percentile pairs (2.5 and 97.5; 0.5 and 99.5) create approximate 95% and 99% credible intervals respectively. The minimum and maximum "I_i" values create an 'envelope' around each zone. It is important to run enough simulations to produce reliable estimates.

The tabular results can be printed, saved to a text file or saved as a '.dbf' file with a *LMoran<root name>* prefix with the root name being provided by the user. For the latter, specify a file name in the "Save result to" in the dialogue box. The 'dbf' file can then be linked to the input 'dbf' file by using the ID field as a matching variable. This would be done if the user wants to map the "I_i" variable, the Z-test, or those zones for which the "I_i" value is either higher than the 97.5 or 99.5 percentiles or lower than the 2.5 or 0.5 percentiles of the simulation results.

Example 1: Local Moran Statistics for Baltimore Auto Thefts

Using data on 14,853 motor vehicle thefts for 1996 in both Baltimore County and Baltimore City, the number of incidents occurring in each of 1,349 census block groups was calculated (Figure 9.2). As seen, the pattern shows a higher concentration towards the center of the metropolitan area, as would be expected, but that the pattern is not completely uniform.

There are many block groups within the City of Baltimore with very low counts of auto thefts and there are block groups within the County with very high counts. Using these data, *CrimeStat* calculated the Local Moran statistic with the variance box checked and the small distance adjustment used. The range of I_i values varied from -37.26 to +180.14 with a mean of 5.20. The standardized Local Moran 'Z' varied from -12.71 to 50.12 and with a mean of 1.61. Figure 9.3 maps the distribution. Because a negative I_i value indicates dissimilarity, these values have been drawn in red compared to blue for a positive I_i value. As seen, in both the City of Baltimore and the County of Baltimore, there are block groups with large negative I_i values, indicating that they differ from the surrounding block groups.

Figure 9.2:

1996 Motor Vehicle Thefts

Number of Auto Thefts Per Block Group

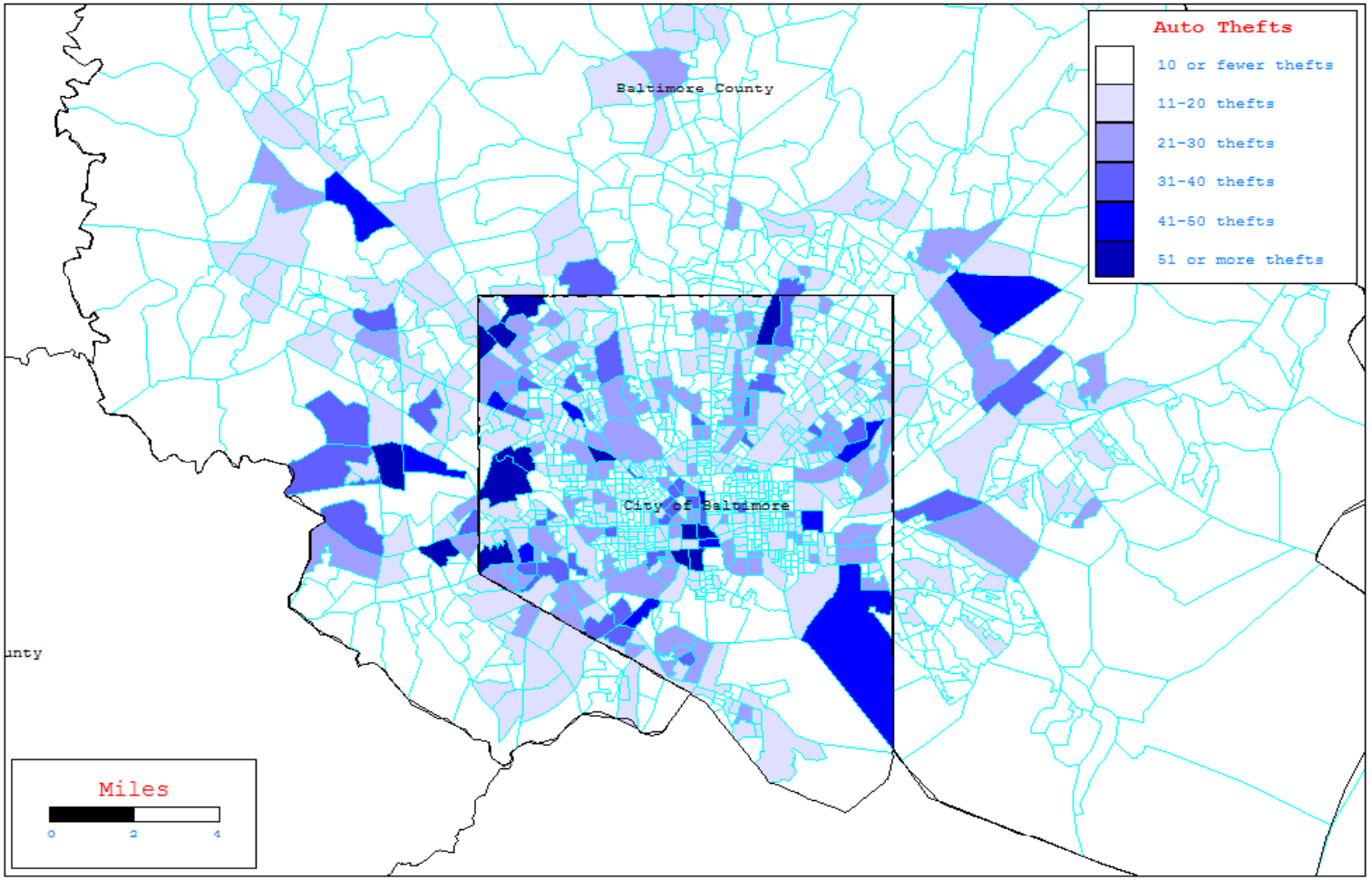
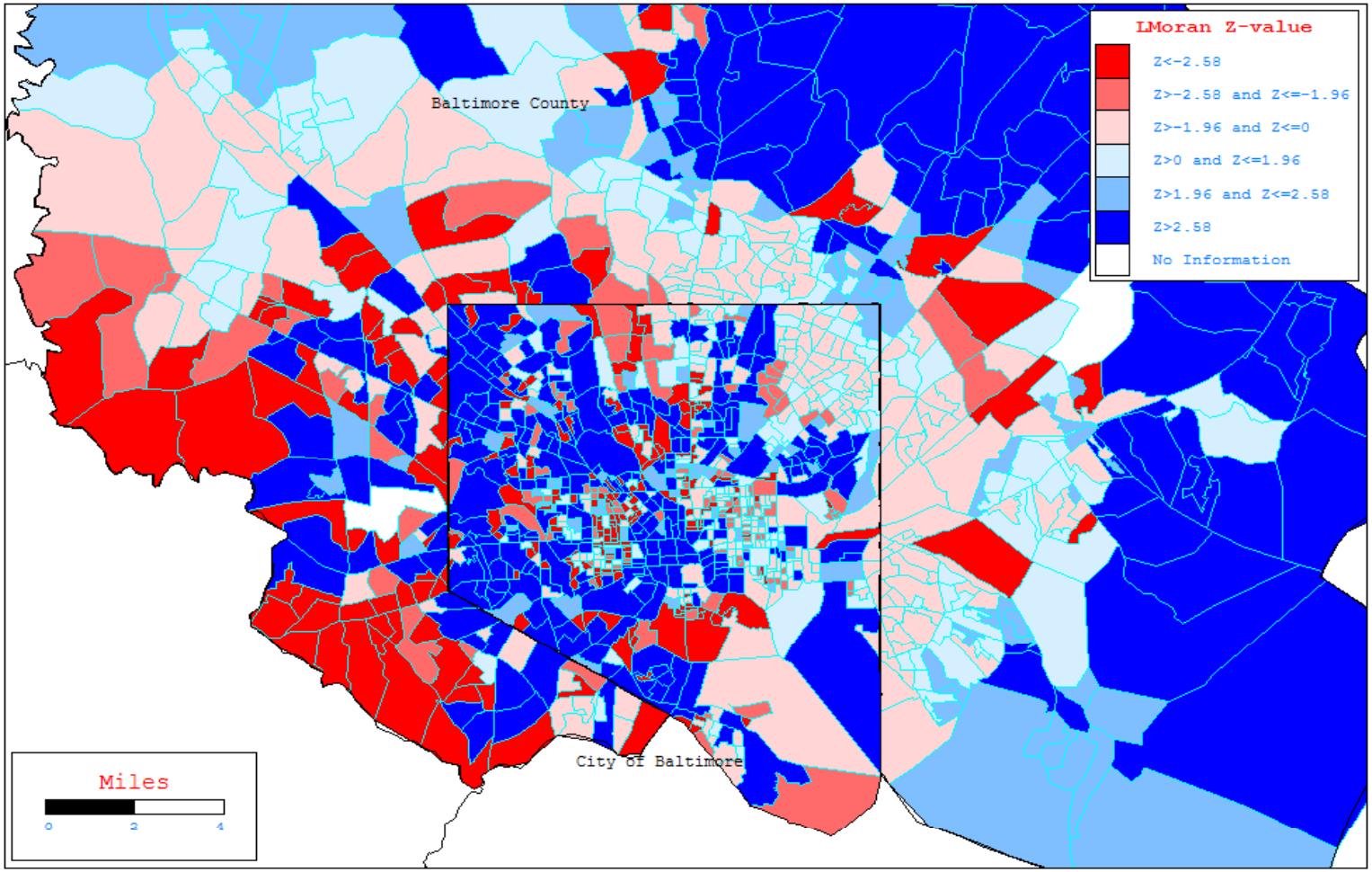


Figure 9.3:

Local Spatial Autocorrelation of 1996 Vehicle Thefts

Local Moran Z-Value of Block Groups



For example, in the central part of Baltimore City, there is a small area of about eight block groups with low numbers of auto thefts, compared to the surrounding block groups. These form a 'cold spot'. Consequently, they appear in dark tones in Figure 9.3 indicating that they have high I_i values (i.e., negative spatial autocorrelation). Similarly, there are several block groups on the western side of the County which have relatively high numbers of auto thefts compared to the surrounding block groups. They form a hot spot. Consequently, they also appear in dark tones in Figure 9.3 because this indicates positive spatial autocorrelation, having values that are similar to the surrounding blocks. In other words, similarity is shown in blue and dissimilarity in red.

Example 2: Simulated Local Moran Confidence Intervals for Houston Burglaries

To illustrate the simulated confidence intervals, we apply the Local Moran statistic to burglaries in the City of Houston shown in figure 9.4. The data were 26,480 burglaries that occurred in 2006. They were aggregated to 1,179 traffic analysis zones (TAZ). Anselin's Local Moran statistic was calculated on each of the TAZ's with 1,000 Monte Carlo simulations being calculated. Figure 9.5 shows a map of the calculated local "I_i" values. It can be seen that there are many more zones of positive spatial autocorrelation where the zones are similar to their neighbors. In most of these cases, the zone has few burglaries whereas it is surrounded by zones that also have few burglaries. A few zones have negative spatial autocorrelation. In most of the cases, the zones have many burglaries and are surrounded by zones with few burglaries.

Confidence intervals were calculated in two ways. First, the theoretical variance was calculated and a Z-test computed. This is done in *CrimeStat* by checking the 'theoretical variance' box. The test assumes that "I_i" is normally distributed, which may or may not be a valid assumption. Second, a Monte Carlo simulation was used to estimate the 99% confidence intervals (i.e., outside the 0.5 and 99.5 percentiles).

Table 9.1 shows the results for four records. The four records illustrate different combinations. In the first record (TAZ 522), the "I_i" value is 0.000373, indicating positive spatial autocorrelation (i.e., nearby zones have similar values). Comparing it to the 95% credible intervals, it is larger than the 97.5^{th} percentile. In addition, the Z-test, based on the theoretical variance, is positive. Thus, both the simulated confidence intervals and the theoretical confidence interval indicate that the "I_i" for this zone is significant.

Figure 9.4:

Burglaries in Houston: 2006

Number of Burglaries by Traffic Analysis Zones

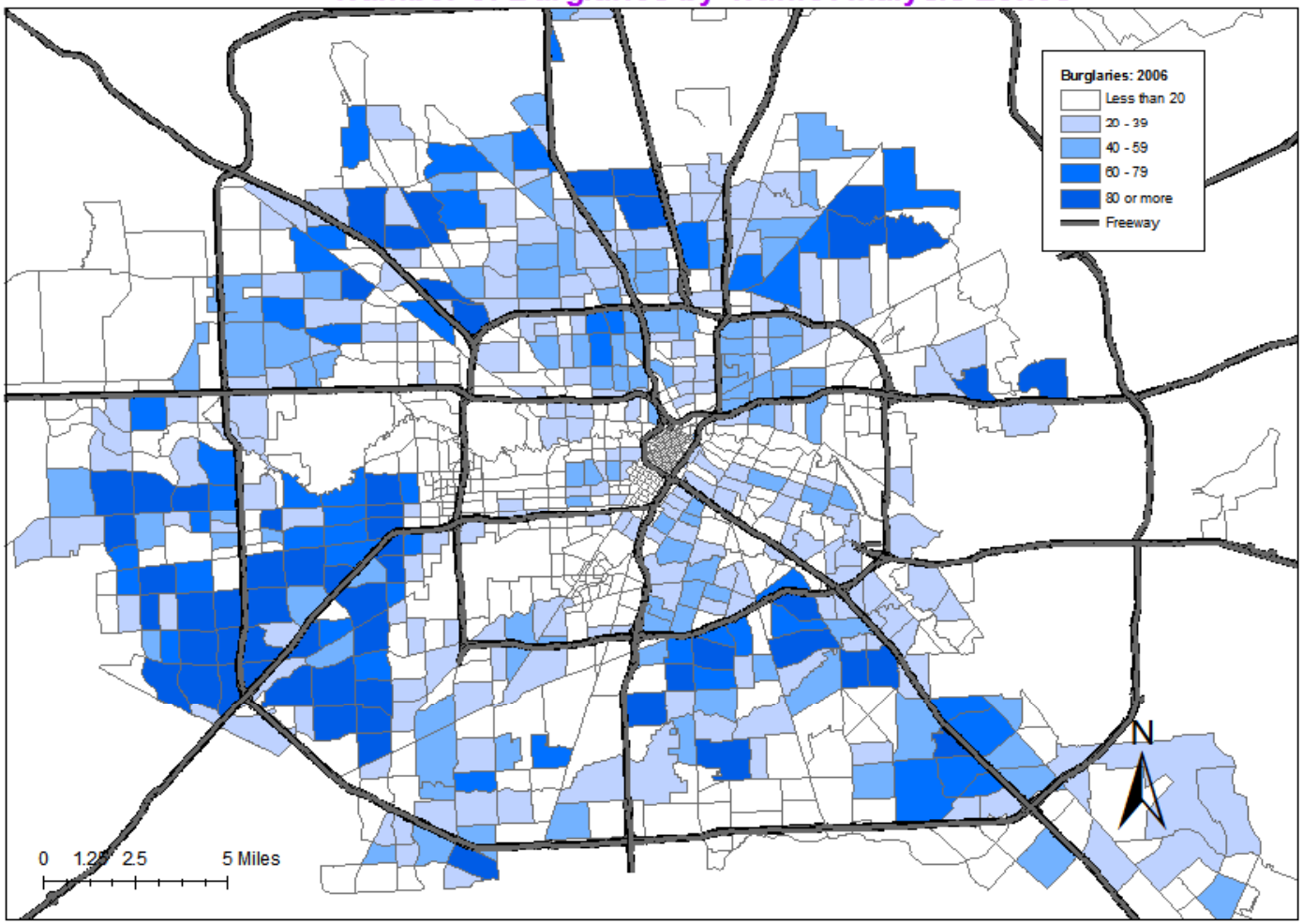


Figure 9.5:

Burglary Hot Spots in Houston: 2006

Local Moran "I" for Traffic Analysis Zones

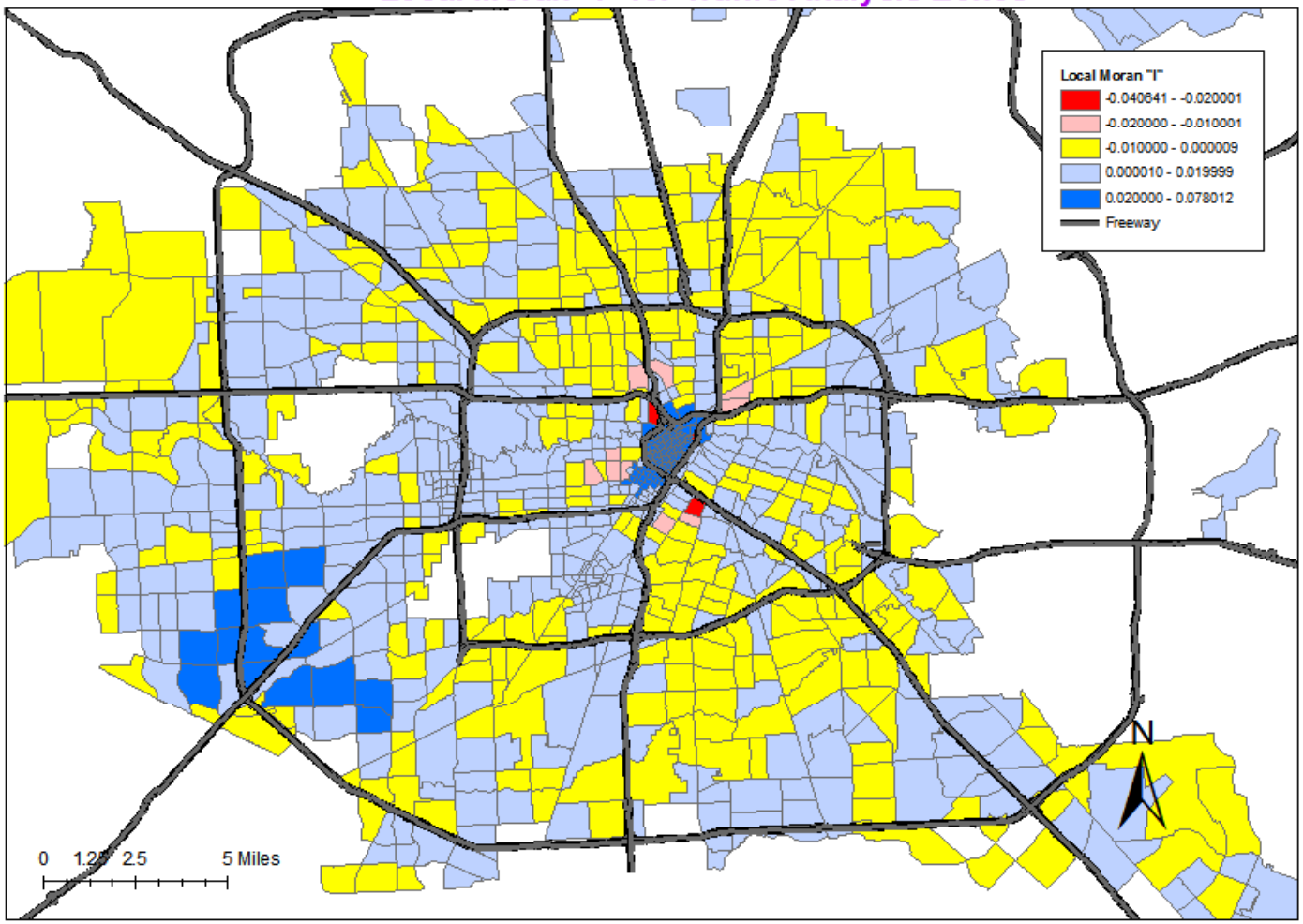


Table 9.1:

Anselin's Local Moran 95% Confidence & Credible Intervals

4 Cases Estimated from Theoretical Variance and from Monte Carlo Simulation

					Simulated		*Theoretical*	
TAZ	**X**	**Y**	**"I_i"**	**Expected**	**0.5 %**	**97.5 %**	**Z-test**	**p**
522	3152030	13941900	**0.000373**	-0.000010	-0.000856	0.000216	2.29	0.05
534	3200630	13955800	**0.000345**	-0.000007	-0.000516	0.000226	1.82	n.s.
182	3126150	13842900	**-0.040641**	-0.000087	-0.014287	0.007292	-9.69	0.0001
384	3156740	13879400	**-0.000886**	-0.000018	-0.001259	0.000593	-2.20	0.05

In the second record (TAZ 534), the "I_i" value is 0.000345, also indicating positive spatial autocorrelation. However, the "I_i" value is greater than the 97.5th percentile, indicating that the simulation suggests the "I_i" is greater than what would be expected by chance. On the other hand, the Z-test, based on the theoretical distribution, is not significant. Thus, there is an inconsistency between simulation test and the Z-test.

In the third record (TAZ 182), there is consistency between the simulated and theoretical significance tests. The "I_i" is negative (-0.040641), indicating negative spatial autocorrelation (i.e., the has different values than nearby zones). The simulation shows that the "I_i" is more negative than the simulated 5th percentile and the Z-test is also significantly negative.

The fourth record (TAZ 384) shows a negative "I_i", indicating negative spatial autocorrelation (i.e., nearby zones have different values). But there is inconsistency in the test. The simulation shows that this "I_i" falls between the 5th and 97.5th percentiles, indicating non-significance, whereas the Z-test suggests the "I_i" is significant.

In general, simulated confidence intervals will be similar to the theoretical ones. But, there can be discrepancies. The reason is that the sampling distribution of "I_i" may not be (and probably is not) normally distributed. Of the 1,179 traffic analysis zones, 661 showed significant "I_i" values according to the simulated 99% credible intervals (i.e., either equal to or smaller than the 0.5 percentile or equal to or greater than the 99.5 percentile) while 688 of the zones showed significant "I_i" values according to the theoretical Z-test at the 99% level (i.e., having a Z-value equal to or less than -2.58 or equal to or greater than 2.58). It would behoove the user to estimate the number of zones that are significant according to both the simulated and theoretical confidence intervals before making a decision as to which criterion to use.

Therefore, both the simulated confidence interval and the theoretical distribution should be used with caution. The best mapping solution may be to map only those zones that are highly

significant with both tests showing substantial significance. Or, alternatively, map only those zones with the highest positive or highest negative "I_i" values.

Uses of Anselin's Local Moran

Anselin's Local Moran has a number of uses. First, it can identify zones that are different (dissimilar) from its neighbors. This can be a good first step in finding locations that either have higher crime numbers (a hot spot) or lower crime numbers (a cold spot) than the neighboring areas. This can focus police efforts on identifying the problems that cause the zone to be higher in the case of a hot spot or to identify factors that mitigate crime in the case of a cold spot.

Second, another use of Anselin's Local Moran statistic is to identify 'outliers', zones that are very different from their neighbors. In this case, zones with a high negative I value (e.g., with an "I_i" smaller than two standard deviations below the mean) are indicative of outliers. They either have a high number of incidents whereas their neighbors have a low number or, the opposite, a low number of incidents amidst zones with a high number of incidents. Identifying the outliers can focus on zones that are unique (and which should be studied) or, in multivariate analysis, on zones that need to be statistically treated differently in order to minimize a large modeling error (e.g., creating a dummy variable for the extreme outliers in a regression model).

In short, the Local Moran statistic can be a useful tool for identifying zones that are dissimilar from their neighborhood. To use the Local Moran statistic, however, requires that the data be summarized into zones in order to produce the necessary intensity value. Given that most crime incident databases will list individual events without intensity or weight values assigned, this will entail additional work by a law enforcement agency.

Limitations of Anselin's Local Moran

There are several limitations to the method. First, because it is an index of similarity, a positive "I_i" value does not necessarily indicate a hot spot. The positive "I_i" value could be due to zones with low values of the intensity variable surrounded by other zones that also have low values. Thus, in terms of using the method to identify hot spots of zones can lead to ambiguous results. It is best seen as a first step in identifying hot spot zones.

Second, there are concerns about the statistical criterion used to identify a zone as being similar or dissimilar to its neighbors. One has to be suspect about a technique that finds significance in more than half the cases. It would probably be more conservative to use 99% confidence intervals for identifying zones that show positive or negative spatial autocorrelation rather than using 95% confidence intervals or, better yet, choosing only those zones that have

very negative or very positive "I_i" values. Unfortunately, this characteristic of Anselin's local Moran is also true of the local Getis-Ord statistic, which is discussed below. The significance tests, whether simulated or theoretical, are not strict enough and, thereby, increase the likelihood of a Type I (false positive) error. A user must be very careful in interpreting "I_i" values for individual zones and would be better served choosing only the very highest or lowest.

For a detailed discussion of problems in conducting tests on local spatial autocorrelation statistics, such as the local Moran or Getis-Ord Local "G" (to be discussedbelow), see Waller and Gottway (2004; p. 238).

Getis-Ord Local "G"

The Getis-Ord Local G statistic applies the Getis-Ord "G" statistic to individual zones to assess whether particular zoness are spatially related to the nearby zones (see Chapter 5). Unlike the global Getis-Ord "G" but like Anselin's Local Moran, the Getis-Ord Local "G" is applied to each individual zone. The formulation presented here is taken from Wong and Lee (2005). The "G" value is calculated with respect to a specified search distance (defined by the user), namely:

$$G_i(d) = \frac{\sum_i W_{ij}(d) X_j}{\sum_j X_j} \tag{9.9}$$

$$E[G_i] = \frac{W_i}{(N-1)} \tag{9.10}$$

$$Var(G_i) = E(G_i^2) - [E(G_i)]^2 \tag{9.11}$$

$$E[G_i^2] = \frac{1}{(\sum_j X_j)^2}\left[\frac{W_i(n-1-W_i)\sum_j X_j^2}{(N-1)(N-2)}\right] + \frac{W_i(W_i-1)}{(N-1)(N-2)} \tag{9.12}$$

where $\boldsymbol{w_j}$ is the weight of zone "j" from zone "i", $\boldsymbol{W_i}$ is the sum of weights for zone "i", and $\boldsymbol{n}$ is the number of cases.

The standard error of G(d) is the square root of the variance of G. Consequently, a Z-test can be constructed by:

$$S.E.[G(d)] = \sqrt{Var[G(d)]} \tag{9.13}$$

$$Z[G(d)] = \frac{G(d)-E[G(d)]}{S.E.[G(d)]} \tag{9.14}$$

A good example of using the Getis-Ord local “G” statistic in crime mapping is found in Chainey and Racliffe (2005, pp. 164-172).

ID Field

The user should indicate a field for the ID of each zone. This ID will be saved with the output and can then be linked with the input file (Primary File) for mapping.

Search Distance

The user must specify a search distance for the test and indicate the distance units (miles, nautical miles, feet, kilometers, meters,

Getis-Ord Local “G” Simulation of Confidence Intervals

Since the Getis-Ord “G” statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a *permutation* type Monte Carlo simulation can be run whereby the original values of the intensity variable, Z, for the zones are maintained but are randomly re-assigned to zones for each simulation run. This will maintain the distribution of the variable Z but will estimate the value of G for each zone under random assignment of this variable. Specify the number of simulations to be run (e.g., 100, 1000, 10000).

Output for Each Zone

The output is for each zone includes:

1. The sample size
2. The ID
3. The X coordinate
4. The Y coordinate
5. The “G”
6. The expected “G”
7. The difference between “G” and the expected “G”
8. The standard deviation of “G”
9. A Z-test of "G" under the assumption of normality for the zone

and if a simulation is run:

10. The 0.5 percentile of “G” for the zone

11. The 2.5 percentile of “G” for the zone
12. The 97.5 percentile of “G” for the zone
13. The 99.5 percentile of “G” for the zone

The two pairs of percentiles (5 and 95; 2.5 and 97.5; 0.5 and 99.5) create approximate 95% and 99% credible intervals respectively around each zone. The minimum and maximum “G” values create an ‘envelope’ around each zone. However, unless a large number of simulations are run, the actual “G” value may fall outside the envelope for any zone. The tabular results can be printed, saved to a text file or saved as a '.dbf' file. For the latter, specify a file name in the “Save result to” in the dialogue box. The file is saved with a *LGetis-Ord<root name>* prefix with the root name being provided by the user.

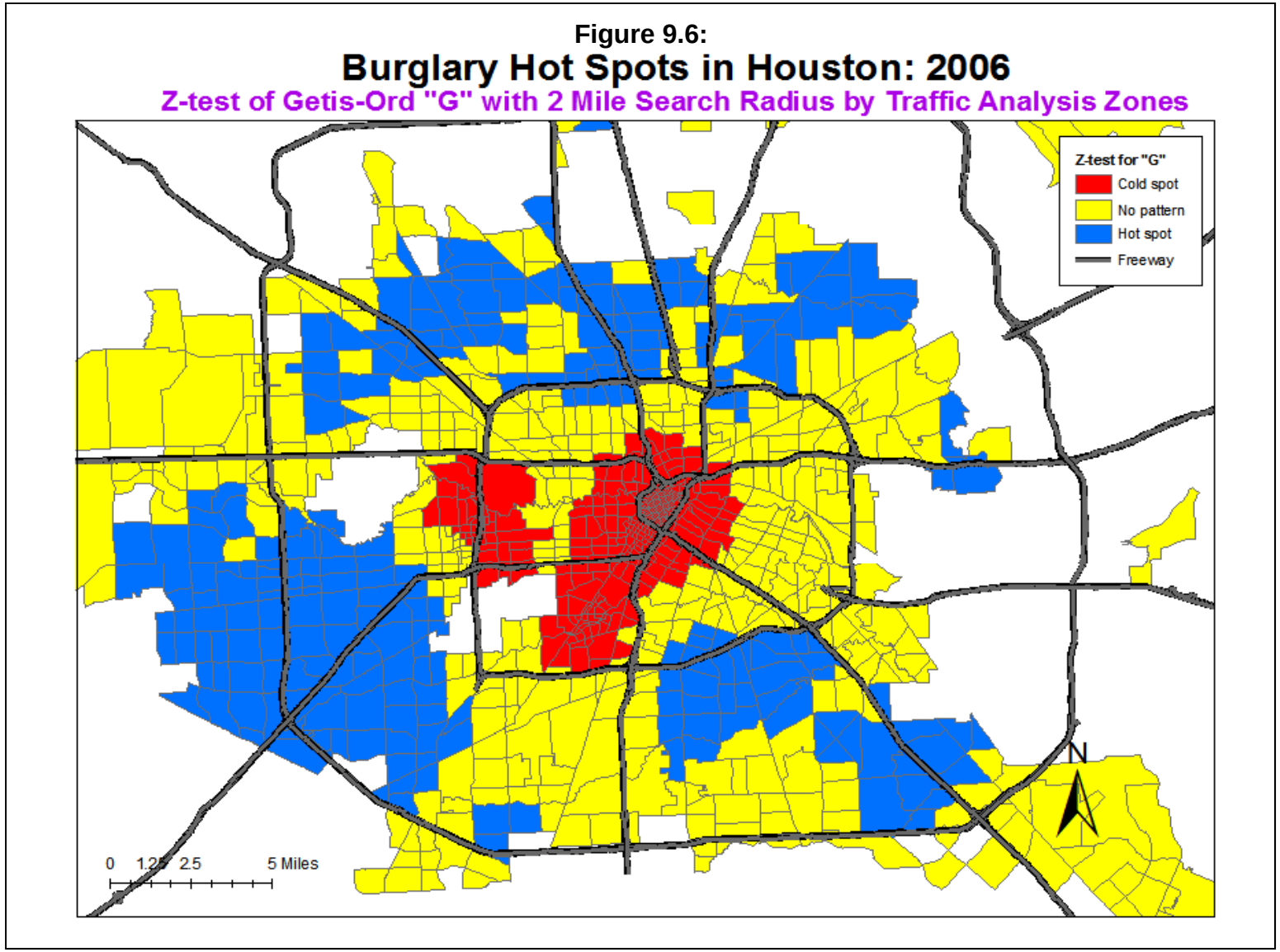
The ‘dbf’ output file can be linked to the Primary File by using the ID field as a matching variable. This would be done if the user wants to map the “G” variable, the expected “G”, the Z-test, or those zones for which the “G” value is either higher than the 97.5 or 99.5 percentiles or lower than the 2.5 or 0.5 percentiles of the simulation results respectively (95% or 99% confidence intervals).

Example: Testing Houston Burglaries with the Getis-Ord Local “G”

Using the same data set on the Houston burglaries as above, the Getis-Ord Local “G” was run with a search radius of 2 miles. The output file was then linked to the input file using the ID field to allow the mapping of the local “G” values. Figure 9.6 illustrates the Z-test of the Getis-Ord Local “G” for different zones. The map displays the significance of the Z-test (the difference between the “G” and the expected “G” relative to the standard error of “G”). Zones with a Z-test of +1.96 or higher are shown in blue (hot spots). Zones with Z-tests of -1.96 or smaller are shown in red (cold spots) while zones with a Z-test between -1.96 and +1.96 are shown in yellow (no pattern).

As seen, there are some very distinct patterns of zones with high positive spatial autocorrelation and low positive spatial autocorrelation. Examining the original map of burglaries by TAZ (Figure 9.4), it can be seen that where there are many burglaries, the zones tend to show high positive spatial autocorrelation (hot spots) in Figure 9.6. Conversely, where there are few burglaries, the zones show either low positive spatial autocorrelation (‘cold spots’) or, more commonly, no pattern in Figure 9.6. In particular, the greater downtown Houston area, and area southwest of downtown that includes the Texas Medical Center and a commercial area west of downtown around the IH 610 ‘loop’ show areas of significant ‘cold spots’. These are areas dominated by commercial or office buildings and generally have relatively few burglaries.

Figure 9.6:


Burglary Hot Spots in Houston: 2006
Z-test of Getis-Ord "G" with 2 Mile Search Radius by Traffic Analysis Zones
Z-test for "G"
Cold spot
No pattern
Hot spot
Freeway
N
0 1.25 2.5 5 Miles

Uses of the Getis-Ord Local "G"

The Getis-Ord Local "G" is very good at identifying hot spots and also good at identifying cold spots. As mentioned, Anselin's Local Moran can only identify positive or negative spatial autocorrelation, that is, whether the zones are similar or dissimilar. Those zones with positive spatial autocorrelation could occur because zones with high values are nearby other zones with high values or they could occur because zones with low values are nearby other zones with low values. The Getis-Ord Local "G" can distinguish those two types.

Limitations of the Getis-Ord Local "G"

The biggest limitation with the Getis-Ord Local "G", which also applies to the global Getis-Ord and Getis-Ord Correlogram routines (see Chapter 5), is that it cannot detect negative spatial autocorrelation where a zone is surrounded by neighbors that are different (either having a high value surrounded by zones with low values or having a low value and being surrounded by zones with high values). In actual use, both the Anselin's Local Moran and the Getis-Ord Local "G" should be used to produce a full interpretation of the rsults.

Another limitation is that the significance tests are too weak, allowing too many zones to show significance. In the data shown in Figure 9.6, 63% of the zones (740) were statistically significant by the Z-test! A simulation of credible intervals also showed a very high proportion having G values greater or less than the 95% credible intervals. Thus, there is a substantial Type I error with this statistic (false positives), a similarity it shares with Anselin's Local Moran.

Reducing the search radius will reduce the number of zones with significant Z-scores. For example, with a 1 mile search radius, only 44% of the zones were statistically significant by the Z-test. But, given the size of the zones, there is a limit to how small a search radius can be made. With the Houston block groups, for example, the average area of a block group is 0.48 square miles. If a typical block group size is viewed as a square having that area, then each side would be about 0.7 miles in length. Choosing a search radius smaller than 0.7 would end up with many zones not having neighbors selected, especially farther away from the city center where zones are generally much larger in size. This would lead to an unrealistic estimate of the amount of spatial autocorrelation. In other words, there is a trade-off between the precision of the search radius and the accuracy of the "G" estimate. In this case, a search radius of two miles is a realistic search radius for this geographical distribution.

Waller and Gottway (2004, p. 238) point out that there are four problems with the testing of LISA statistics since the measures are interrelated: First, the distributional properties remain largely unknown. Second, multiple tests lead to overly rejecting the null hypothesis, which we

have demonstrated above. Third, the LISA's of neighboring zones are often highly correlated due to using the same data and, fourth, many of the tests are based on small samples sizes since the number of events in any one zone may be limited. A random simulation can overcome the first problem by using the empirical distribution as a basis for calculating credible intervals, but it cannot overcome the next three.

In short, a user should be very careful in interpreting zones with significant "G" values and would probably be better served by choosing only those zones with the highest or lowest "G" values.

Zonal Nearest Neighbor Hierarchical Clustering

The zonal nearest neighbor hierarchical spatial clustering routine applies the nearest neighbor hierarchical clustering algorithm (Nnh; see Chapter 7 for the background and details) to zonal data. The point-based Nnh is a constant-distance clustering routine that groups points together on the basis of spatial proximity. A threshold distance is defined and the minimum number of points that are required for each cluster specified. The output can be displayed with ellipses or convex hulls.

On the other hand, in the zonal Nnh (Znnh), the algorithm is adjusted to allow *weighting* of each zone usually applied to a single point within the zone (e.g., a centroid). Thus, if the 'point' is a centroid of a zone, then the weighting is an attribute assigned to that centroid (e.g., population, employment, median household income). Clusters are groups of adjacent zones that have much higher weights than non-clustered zones.

The routine requires a primary file (e.g., robberies) that is weighted with the weight or intensity variable (see Primary File). On the Znnh routine, the user defines a weighting variable, a threshold distance, the minimum number of values of the weighting variable that are required for each cluster, and the type of output size, either standard deviational ellipses or convex hulls.

The routine identifies first-order clusters that represent groups of zones that are closer together than the threshold distance, that have the highest weights, and in which there is at least the minimum number of zones specified by the user (the minimum is 3 zones). Clustering is hierarchical in that the first-order clusters are treated as separate 'points' to be clustered into second-order clusters, and the second-order clusters are treated as separate 'points' to be clustered into third-order clusters, and so on. Higher-order clusters will be identified only if the distances between their centers are closer than the new threshold distance.

For example, if the attribute to be grouped is the number of crimes in a zone, then the routine identifies adjacent zones that have high concentrations of crimes. The user can modify the number of clusters identified and the relative size of them by changing the search radius or the minimum number of attributes that must be grouped together. The results can be output as either standard deviational ellipses or convex hulls.

Weighting Variable

Each zone must be weighted by an attribute variable. This is the weight or intensity variable defined on the Primary File page. The user specifies whether the weight or the intensity variable is to be used for the attribute. The default is Intensity.

Clustering Criteria

Two criteria are used to group zones together.

Criterion 1: Threshold Distance

The first criterion in identifying clusters is whether zones are closer than a specified threshold distance. There are two alternatives in selecting the threshold distance: 1) a fixed distance (the default is 2 miles); or 2) a random nearest neighbor distance.

Fixed distance

Unlike the Nnh routine for clustering points (Chapter 7), the default alternative for selecting a threshold distance in the Znnh is to choose a fixed distance (in miles, nautical miles, feet, kilometers, or meters). The user checks the "Fixed distance" box and selects a threshold distance. The default value is 2 miles but the user can change this.

The main advantage of this method is that, first, the search radius can be specified exactly and, second, unlike points, zones do not overlap and are spatially dispersed. The distance between adjacent zones may be substantial especially for large zones at the periphery of an urban area. Thus, to capture adjacent zones that have high values of the attribute variable requires choosing a search radius that is large.

The main disadvantage of this method is that the choice of a threshold is subjective. There is no reason why any particular search radius should be chosen. Further, the larger the distance that is selected, the greater the likelihood that clusters will be found by chance. This can be tested using a Monte Carlo simulation (see below).

Random nearest neighbor distance

The alternative is to use the expected random nearest neighbor distance for first-order nearest neighbors. The user specifies a *one-tailed* confidence interval around the random expected nearest neighbor distance. The t-value corresponding to this probability level, t, is selected from the Student's t-distribution under the assumption that the degrees of freedom are at least 120.[2] This selection is controlled by a slide bar under the routine (see Figure 9.1). From Chapter 6, the mean random distance is defined as:

$$d_{NN(ran)} = 0.5\sqrt{\frac{A}{N}} \tag{9.15}$$

where *A* is the area of the region and *N* is the number of zones and the standard error of the mean random distance is:

$$SE_{d(ran)} \cong \sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \tag{9.16}$$

where *A* is the area of the region and N is the number of zones. The confidence interval around that distance is defined as:

$$Confidence\ interval = \ d_{NN(ran)} \pm t * SE_{d(ran)} \tag{9.17}$$

where *t* is the t-value associated with a probability level in the Student's t-distribution.

The approximate lower limit of this confidence interval is:

$$Lower\ limit\ of\ confidence\ interval = \ d_{NN(ran)} - t * SE_{d(ran)}$$

$$\cong 0.5\sqrt{\frac{A}{N}} - t\sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \tag{9.18}$$

2 This is the next highest degree of freedom in the Student's t-table below infinity.

and the upper limit of this confidence interval is:

$$Upper\ limit\ of\ confidence\ interval = d_{NN(ran)} + t * SE_{d(ran)}$$

$$\cong 0.5\sqrt{\frac{A}{N}} + t\sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \quad (9.19)$$

The confidence interval defines a probability for the distance between any *pair* of zones. For example, for a specific *one-tailed* probability, *p*, fewer than *p*% of the zones would have nearest neighbor distances smaller than this selected limit *if* the distribution was spatially random. *If* the data were spatially random and if the mean random distance is selected as the threshold criteria (the default position on the slide bar), approximately 50% of the pairs will be closer than this distance. For randomly distributed data, if a $p \leq .05$ level is taken for t (two steps to the left of the default or the fifth in from the left), then only about 5% of the pairs would be closer than the threshold distance. Similarly, if a $p \leq .75$ level is taken for t (one step to the right of the default or the fifth in from the right), then about 75% of the pairs would be closer than the threshold distance.

Table 9.2:

Approximate Probability Values Associated with Threshold Scale Bar

Position	**Scale Bar Probability**	**Description**
1	0.00001	Far left point of slide bar
2	0.0001	Second from left
3	0.001	Third from left
4	0.01	Fourth from left
5	0.05	Fifth from left
6	0.1	Sixth from left
7	0.5	Sixth from right (default value)
8	0.75	Fifth from right
9	0.9	Fourth from righ
10	0.95	Third from righ
11	0.99	Second from righ
12	0.999	Far right point of slide bar

In other words, the threshold distance is a probability level for selecting any *two* zones (a pair) on the basis of a chance distribution. The slide bar has 12 levels and is associated with a probability level for a t-distribution from a sample of 120 or larger. From the left, the p-values are approximately (see Table 9.2 above):

Taking a broader conception of this, if there is a spatially random distribution, then for all distances between pairs of zones, of which there are

$$Combinations = \frac{N(N-1)}{2} \tag{9.18}$$

fewer than *p*% will be shorter than this threshold distance.

Area must be defined correctly

Note that it is *very* important that area be defined correctly for this routine to work. If the user defines the area on the measurement parameters page (see Chapter 3), the Znnh routine uses that value to calculate the threshold distance. If the user does not define the area on the measurement parameters page, the routine calculates the area from the minimum and maximum X/Y values (the bounding rectangle), which will usually be a larger area. In either case, the routine will be able to calculate a threshold distance and run the routine.

However, if the area units are defined incorrectly on the measurement parameters page, then the routine will certainly calculate the threshold distance wrongly. For example, if data are in feet but the area on the measurement parameters page are defined in square miles, most likely the routine will not find any zones that are farther apart the threshold distance since that distance is defined in miles. In other words, it is essential that the area units be consistent with the data for the routine to properly work.

Criterion 2: Zones with the Highest Number of Attributes

The second criterion involves the weighting of each zone. With zonal data, each zone has an attribute value, defined either by the intensity variable or weight variable on the Primary File page. Clusters are defined by those zones that are within the threshold distance but which have the highest combined value of the attribute variable. The algorithm looks for a 'center' of three of more zones for which the total value of the attribute variable is highest. Like the Nnh routine, the process is iterative, first finding an approximate center and then re-calculating it with respect to the total value of the attribute variable for those zones within the threshold distance of the center. Eventually, the process stabilizes and the routine quits.

Table 9.3 presents a simple example. Suppose there are two zones (A and B) within the second matrix and each has three other zones closer than the threshold distance (C, D, E for Zone A and F, G, H for Zone B). In this example, Zone A would be chosen as the initial center for the first cluster because the sum of the weights (for itself and for the three other zones that are within the threshold distance) add to 85 whereas the sum of the weights for the other points for Zone B only add to 65 even though Zone B had a higher weight for itself than Zone A.

Table 9.3:
Example of Weighting Pairs of Zones by Attributes

	Zone A		**Zone B**
Other Zones	**Weighting**	**Other Zones**	**Weighting**
A (itself)	10	B (itself)	20
C	20	F	10
D	30	G	15
E	25	H	20
	---		---
TOTAL:	**85**		**65**

The routine then removes the zones selected for the first cluster (A, C, D, and E). It then attempts to find a second cluster. In this example, there is only one other (B, F, G, and H), which is then removed from the matrix. If there were more zones, the routine would look for additional centers of clusters.

Having completed an initial identification of cluster centers, the routine then calculates the center of minimum distance (CMD) for the selected points and then calculates those zones that are within the threshold distance of the CMD. It repeats the process for a second cluster. After a second round of clustering, the routine repeats the process for a third cluster. The iterations continue until no zones change clusters and the calculated center of minimum distance changes very little.

First-order Clusters

Using these criteria, *CrimeStat* constructs a first-order clustering of the zones. For each first-order cluster, the center of minimum distance is output as the cluster center, which can be saved as a '.dbf' file.

Second and Higher-order Clusters

The first-order clusters are then tested for second-order clustering. The procedure is similar to first-order clustering except that the cluster centers (the center of minimum distance for each) are now treated as 'points' which themselves are clustered (see endnote ii). The process is repeated until no further clustering can be conducted. Either all sub-clusters converge into a single cluster, the threshold distance criterion fails, or there are fewer than four seeds in the higher-order cluster.

Note that this process is similar to that of the Nnh routine discussed in Chapter 7 except the selection of clusters is function of the total value of the attribute variable and not just the distance between zones.

Simulating Confidence Intervals

A Monte Carlo simulation can be run to estimate the approximate confidence intervals around first-order Znnh clusters. Second- and higher-order clusters are not simulated since their structure depends on first-order clusters. The user specifies the number of simulation runs and the Znnh clustering is calculated for randomly assigned data. The random output is sorted and percentiles are calculated. The output includes the number of first-order clusters, the area, the number of points, the number of zones, and the density.

Confidence intervals can be estimated from these percentiles. The two most commonly used ones are the 95% (defined by the 2.5 and 97.5 percentiles) and the 99% (defined by the 0.5 and 99.5 percentiles). The simulated data that is used can be viewed by checking the 'Dump simulation data' box on the Options tab.

Type of Graphical Output

The type of graphical output is specified, either standard deviational ellipses or convex hulls around the zones identified in each cluster. If the output is to be ellipses, then the output size for the clusters can be adjusted by the second slide bar. These are the number of standard deviations defined by the ellipse, from one standard deviation (the default value) to three standard deviations. Typically, one standard deviation will cover about 50-60% of the zones (and a higher percentage of the total of the weighting variable) whereas three standard deviations will cover more than 99% of the zones. On the other hand, if the output is to be convex hulls, the routine outputs a convex hull for each identified cluster.

Ellipse cluster output

The results can be output graphically as an ellipse to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or Google Earth 'kml' (if the coordinate system is spherical) files. A file name should be provided. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

First and higher-order ellipses will be output as separate objects. The prefix will be 'Znnh1' for the first-order ellipses, 'Znnh2' for the second-order ellipses, and 'Znnh3' for the third-order ellipses. Higher-order ellipses will only index the number.

Output size for ellipses

The cluster output size can be adjusted by the lower slide bar. This specifies the number of ellipse standard deviations to be calculated for each cluster: one standard deviation (1X - the default value), one and a half standard deviations (1.5X), or two standard deviations (2X). The default value is one standard deviation. Typically, one standard deviation will cover more than half the zones in a cluster whereas two standard deviations will cover more than 99% of the zones in a cluster, though the exact percentage will depend on the distribution. Slide the bar to select the number of standard deviations for the ellipses. The output file is saved as *Znnh<number><file name>* with the file name being provided by the user. The number is the order of the clustering (i.e., 1, 2…).

Restrictions on the number of clusters can be placed by defining a minimum number of zones that are required. The default is 10 and the minimum is 3. If there are too few zones allowed, then there will be many very small clusters. By increasing the number of required zones, the number of clusters will be reduced.

Convex hull cluster output

The clusters can also be output as convex hulls to *ArcGIS* 'shp', *MapInfo* 'mif', various ASCII formats, or Google Earth 'kml' (if the coordinate system is spherical) files. Specify a file name. For MapInfo 'mif' format, the user has to define up to nine parameters including the name of the projection and the projection number. If the MapInfo system file MAPINFOW.PRJ is placed in the same directory as CrimeStat, then a list of common projections with their appropriate parameters is available to be selected.

The name will be output with a ‘CZnnh1’ prefix for the first-order clusters, a ‘CZnnh2’ prefix for the second-order clusters, and a ‘Cznnh3’ prefix for the third-order clusters. Higher-order clusters will index only the number.

Note that unlike the Nnh clustering algorithm for points, discussed in Chapter 7, the zonal Nnh generally has much larger search areas. Consequently the convex hulls will be much larger than the ellipses, even the 2x ellipse (the opposite is true with the Nnh).

Tabular Output

The routine outputs six results for each cluster that is calculated:

1. The hierarchical order and the cluster number
2. The mean center of the cluster (Mean X and Mean Y)
3. The standard deviational ellipse of the cluster (the rotation and the lengths of the X and Y axes)
4. The number of zones in the cluster
5. The area of the cluster
6. The density of the cluster (the total weight of the zones divided by area)

and if a simulation is run:

7. The minimum for the spatially random Znnh simulations:
8. The maximum for the spatially random Znnh simulations
9. The 0.5 percentile for the spatially random Znnh simulations
10. The 1 percentile for the spatially random Znnh simulations
11. The 2.5 percentile for the spatially random Znnh simulations
12. The 5 percentile for the spatially random Znnh simulations
13. The 10 percentile for the spatially random Znnh simulations
14. The 90 percentile for the spatially random Znnh simulations
15. The 95 percentile for the spatially random Znnh simulations
16. The 97.5 percentile for the spatially random Znnh simulations
17. The 99 percentile for the spatially random Znnh simulations
18. The 99.5 percentile for the spatially random Znnh simulations

Example 1: Simulated Clustering of Zones

To illustrate the Znnh routine, a dispersed cluster structure for an arbitrary variable with five main groupings was created with 1,179 City of Houston Traffic Analysis Zones (TAZ). The

five clusters can be labeled as central, southwest, northwest, northeast and southeast. Figure 9.7 illustrates the pattern that was created.

Four separate search areas were selected with a minimum of 25 'events' being required of the attribute variable:

1. 2 miles
2. 5 miles
3. 8 miles
4. 12 miles

Figures 9.8-9.12 illustrate the results of the clustering using these search distances with the standard deviational ellipse. Figure 9.11 also shows the convex hull of the search radius. Notice that a search radius of 2 miles produces small clusters and did not cover the clusters in the northeast, the southeast and most of the southwest. The reason is that TAZs for those areas are quite large with many being larger than 2 miles.

A 5 mile search radius covered the five clusters though the clusters are still small. The 8 mile search radius appeared to fit the data better while the 12 mile search radius produced too large ellipses with one large one for the central area. Note that Figure 9.11 shows the convex hulls of the 8 mile search radius and which covers most of the TAZs of the City of Houston.

Example 2: Clustering of Houston Burglaries by Traffic Analysis Zones

The second example examines burglaries in the City of Houston in 2006. In that year, 24,935 burglaries were recorded. The data from which these came were assigned to blocks. Each of the burglaries was geocoded to the mid-block and then aggregated into 1,179 TAZs. Figure 9.4 above illustrates the pattern of burglaries in Houston.

The Znnh routine was run with four different search radii and with a minimum of 25 burglaries being required for each cluster:

1. 0.5 miles
2. 2 miles
3. 5 miles
4. 8 miles

Figure 9.7:

Test of Znnh Routine

Arbitrary Dispersed Variable

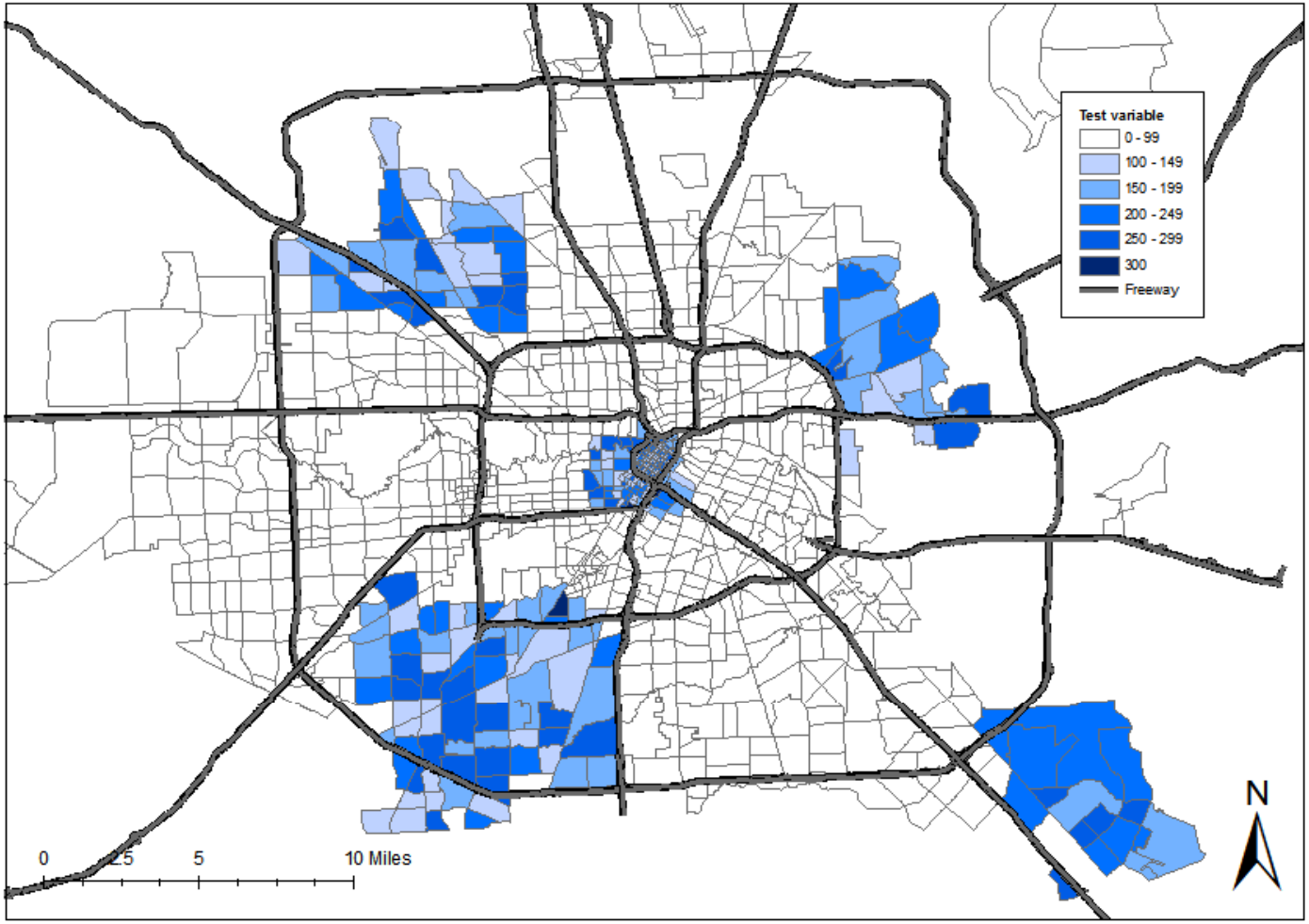


Figure 9.8:

Test of Znnh Routine

Identified Hot Spots with 2 Miles Search Radius and Minimum Number of Events=25

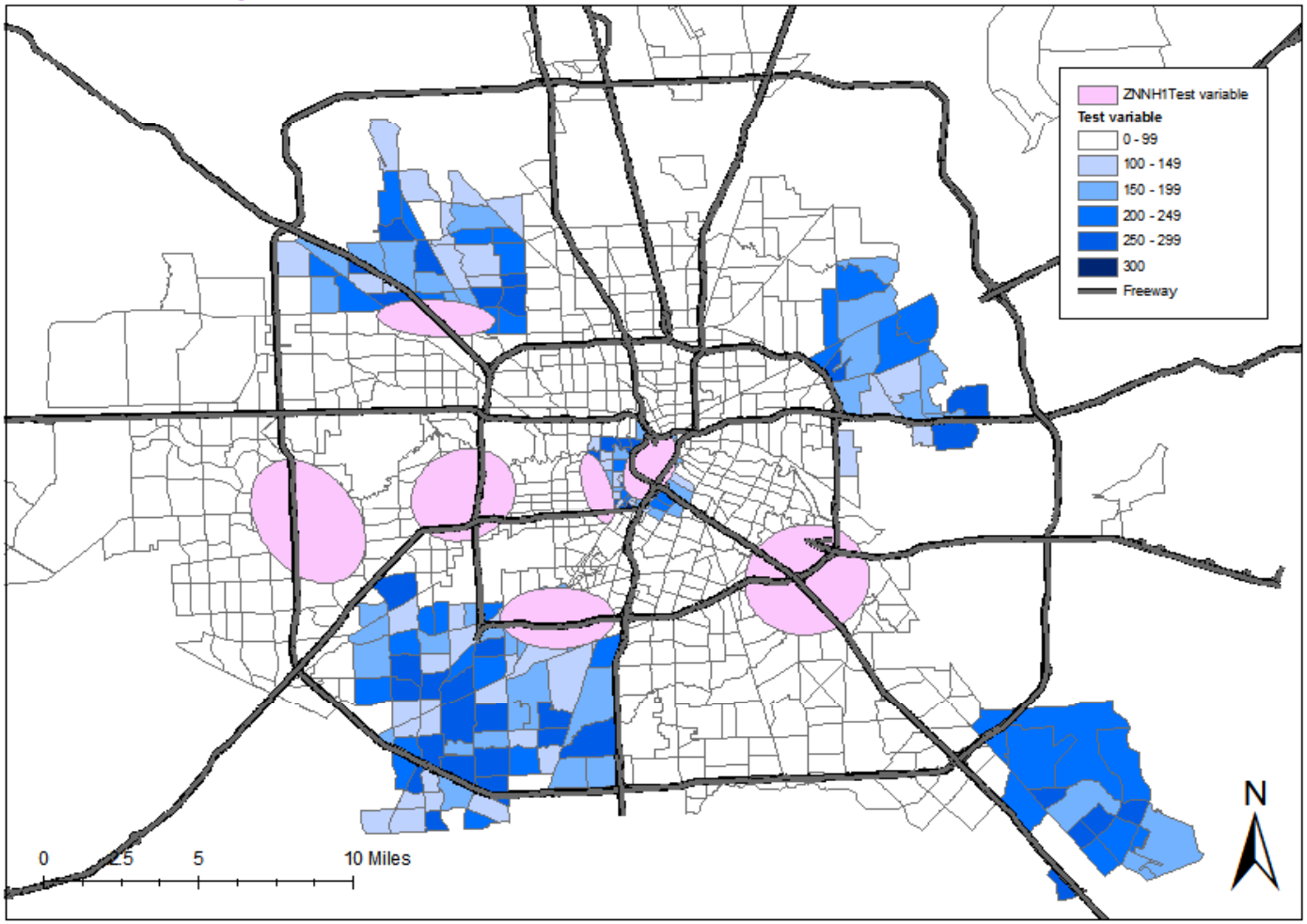


Figure 9.9:

Test of Znnh Routine

Identified Hot Spots with 5 Miles Search Radius and Minimum Number of Events=25

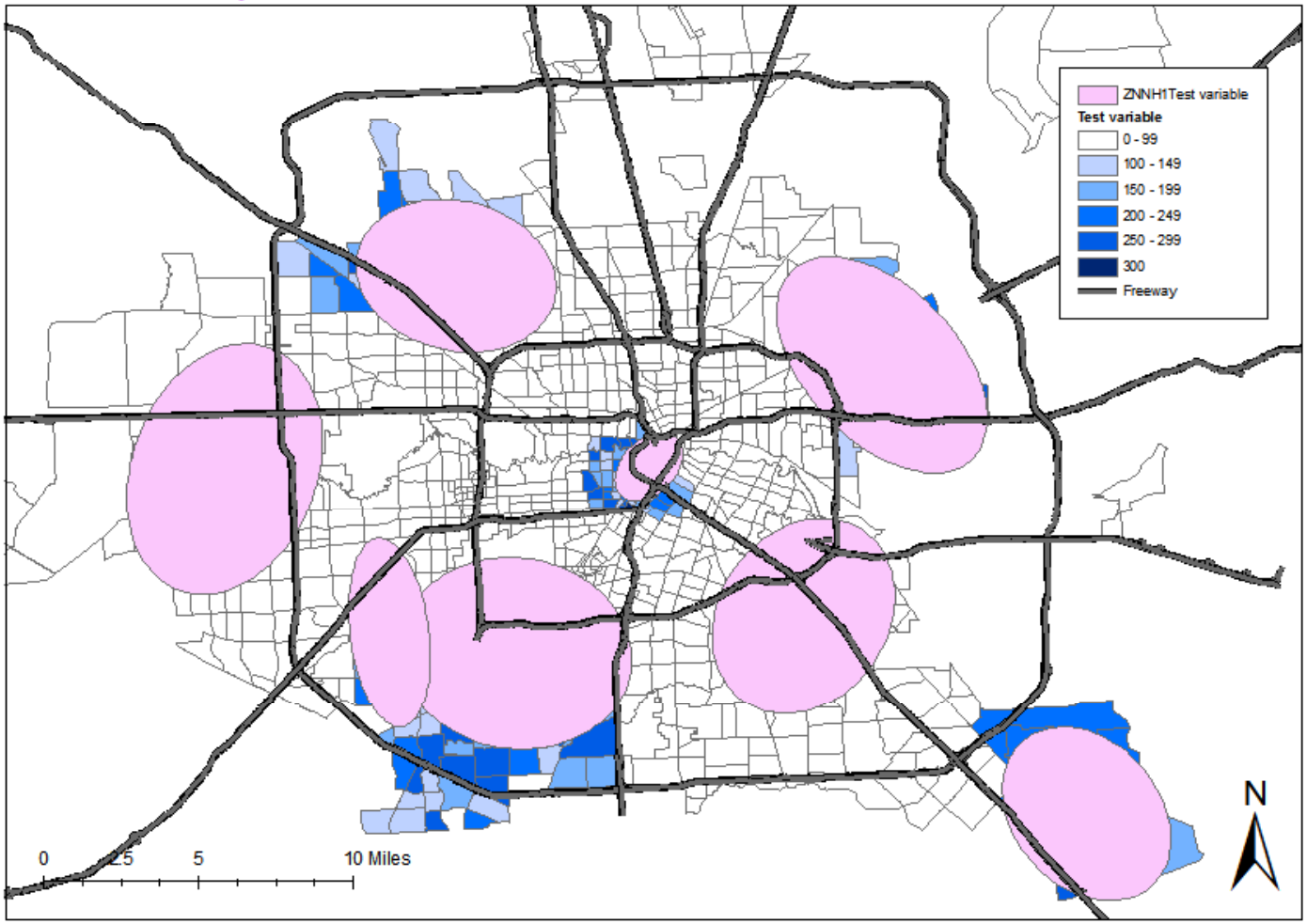


Figure 9.10:

Test of Znnh Routine

Identified Hot Spots with 8 Miles Search Radius and Minimum Number of Events=25

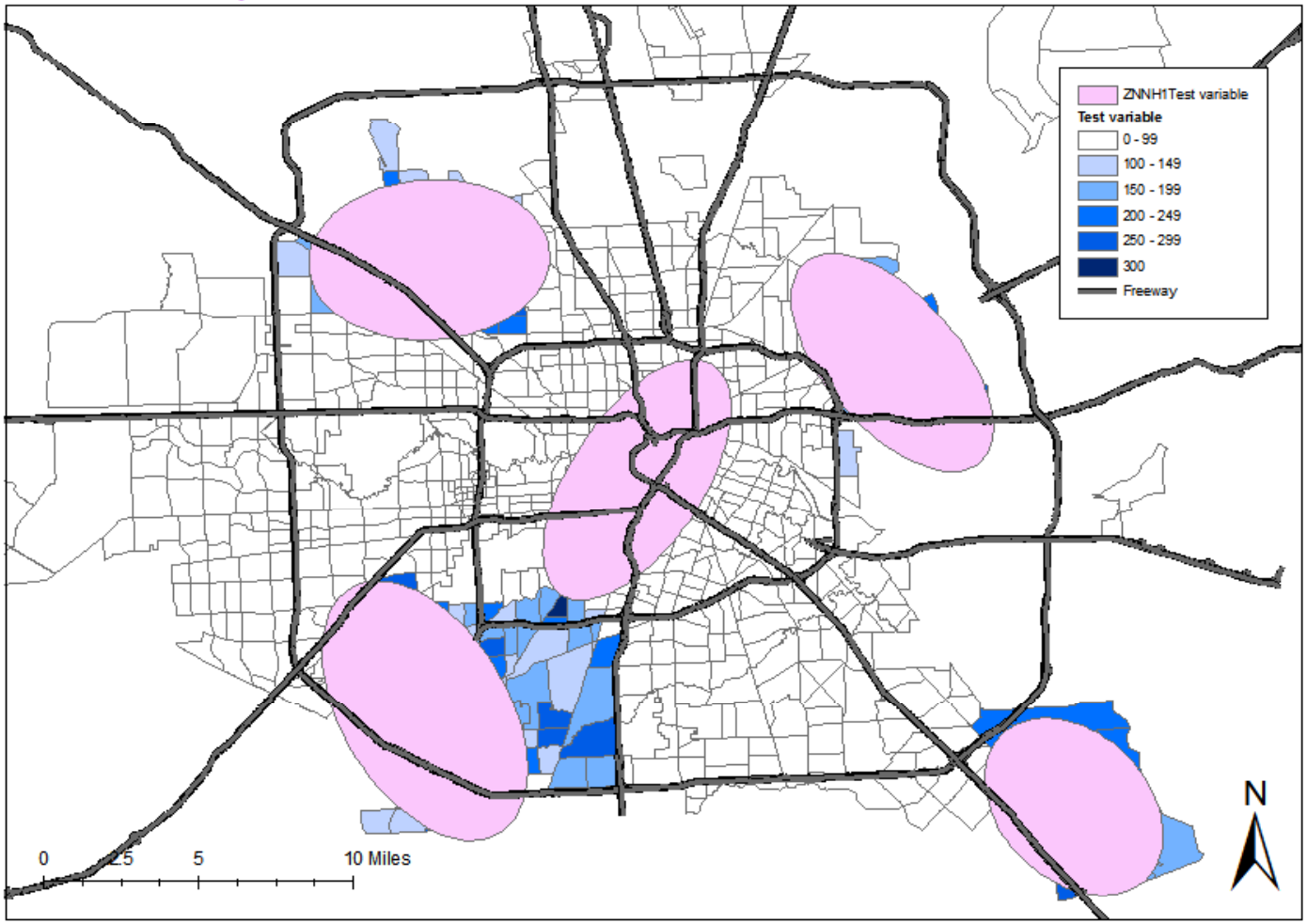


Figure 9.11:

Test of Znnh Routine

Identified Hot Spots with 8 Miles Search Radius and Minimum Number of Events=25

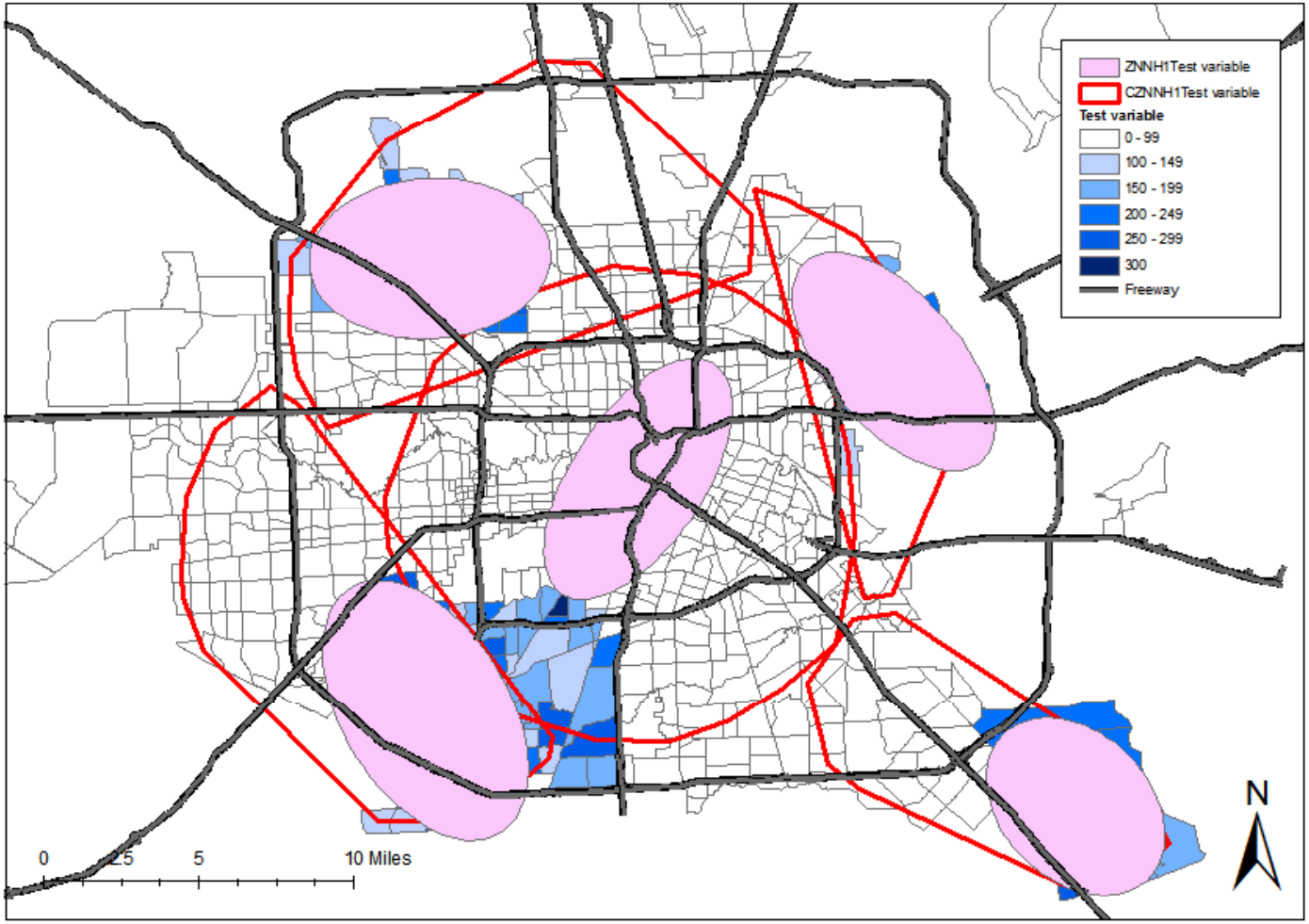
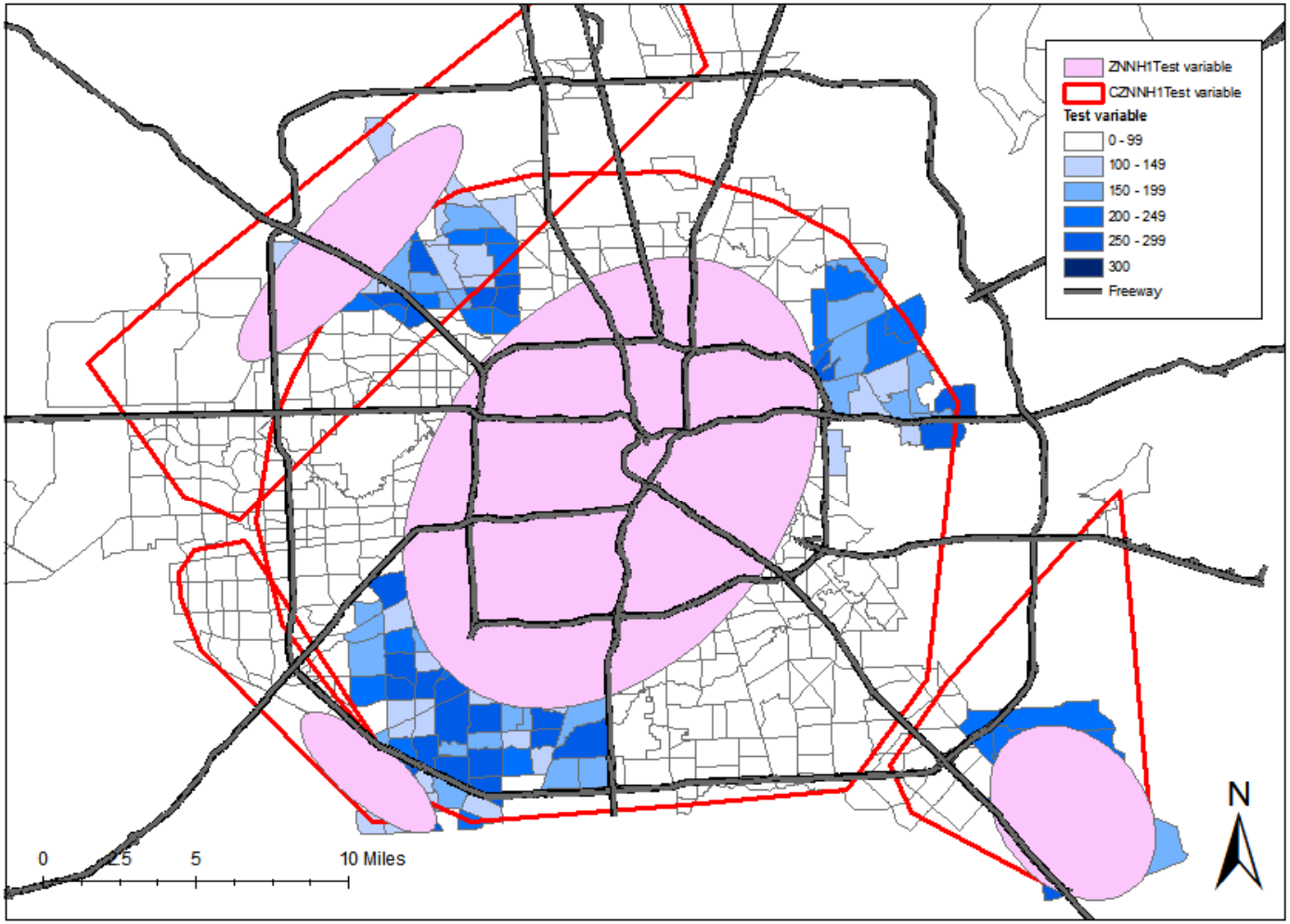


Figure 9.12:

Test of Znnh Routine

Identified Hot Spots with 12 Miles Search Radius and Minimum Number of Events=25



Figures 9-13 shows the results of the 0.5 mile search radius. Four clusters were identified, but they were very small and covered only the downtown Houston area. The reason is that with a half mile radius, only very small TAZ's can be captured within the radius and these are typically in the central downtown area. Further, they do not capture many burglaries, only 139 of the 24,935. However, they do a better job of capturing high density burglary TAZ's, defined as burglaries per square mile (Figure 9.14)

Figure 9.15 through 9.17 show the results of using 2, 5 and 8 mile search radii. The 2 mile search radius produced 10 small clusters; the 5 mile search radius produced 9 medium-sized clusters, and the 8 mile search radius identified 5 moderately large clusters. Clearly the cluster structure produced by the 2 mile search radius was also too small to fit the citywide pattern whereas either the 5 mile search radius or the 8 mile search radius seemed to best fit the overall data. Depending on whether the user wants smaller or larger clusters would determine which of these is selected.

Keep in mind that there is a danger is using large search radii since the likelihood of obtaining clusters by chance increases. To illustrate this, two Monte Carlo simulations of 1000 runs was made with both the 0.5 and the 8 mile search radius. Table 9.4 compares the actual clusters with the simulated clusters.

With the 0.5 mile search radius, no clusters were identified in the Monte Carlo simulation. This indicates that the clusters identified in Figure 9.13 are most likely real. On the other hand, with the 8 mile search radius and randomly distributed data, the expected number of clusters would be expected to vary between 5 and 8 clusters 95% of the time. This is calculated as the credible interval defined by the 2.5^{th} and 97.5^{th} percentiles. Thus, the five clusters obtained by the Znnh are not significantly greater than or smaller than what would be expected by chance. Similarly, the area of the ellipses, the number of attribute points captured and the number of zones are not significantly different than what would be expected by chance.

In short, the distribution that was obtained was not fundamentally different from a chance distribution. This is primarily the result of selecting a very large search radius. A user has to balance the choice between a small search radius which would capture clusters that are statistically much less likely to be due to chance but which cover only a small proportion of the study area with a larger search radius to capture the overall pattern but which increases the likelihood of identifying clusters by chance. In other words, there is a precision versus utility choice with a zonal clustering algorithm such as the Znnh.

Figure 9.13:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 0.5 Mile Search Radius and Minimum Number of Events=25

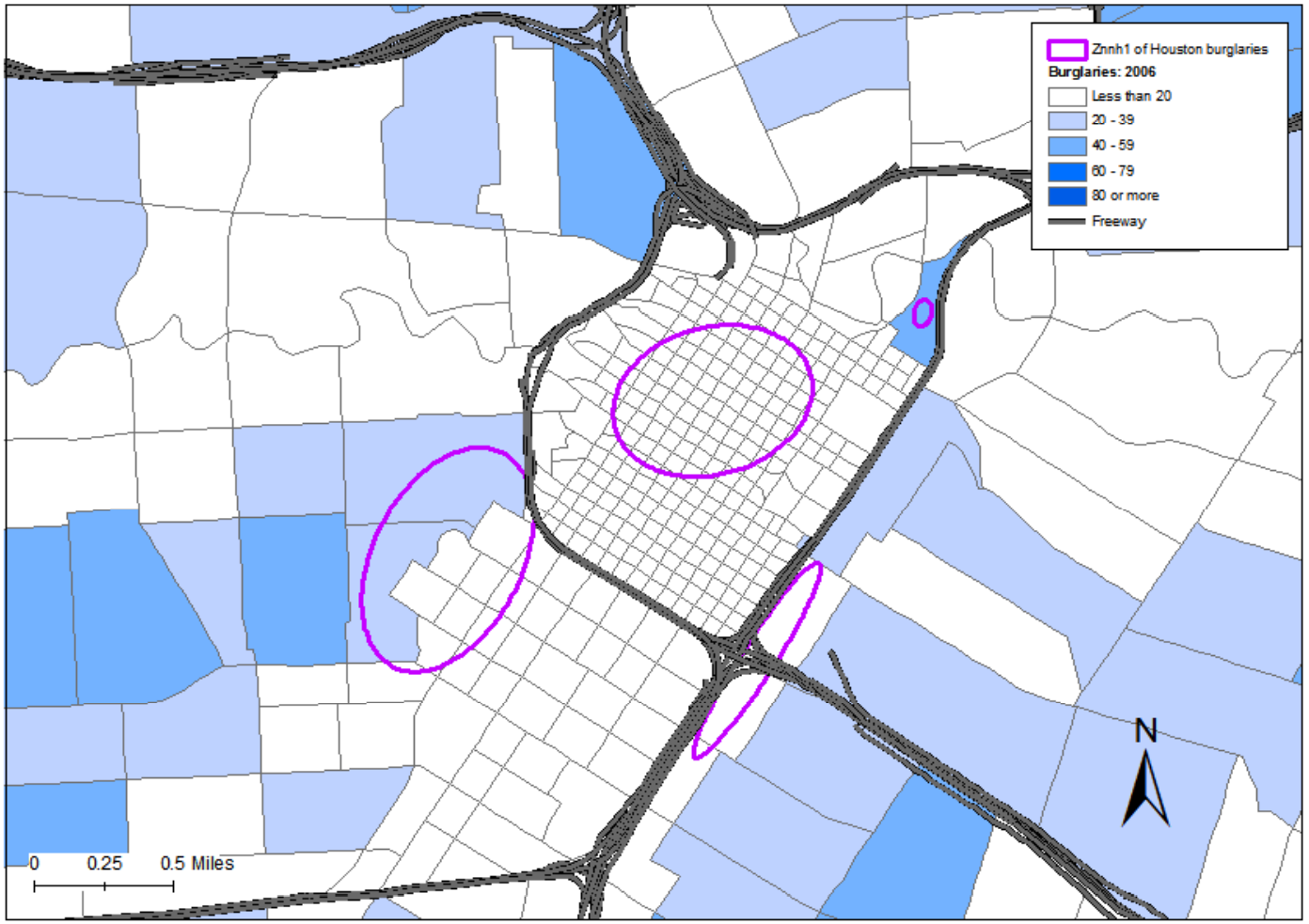


Figure 9.14:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 0.5 Mile Search Radius and Minimum Number of Events=25

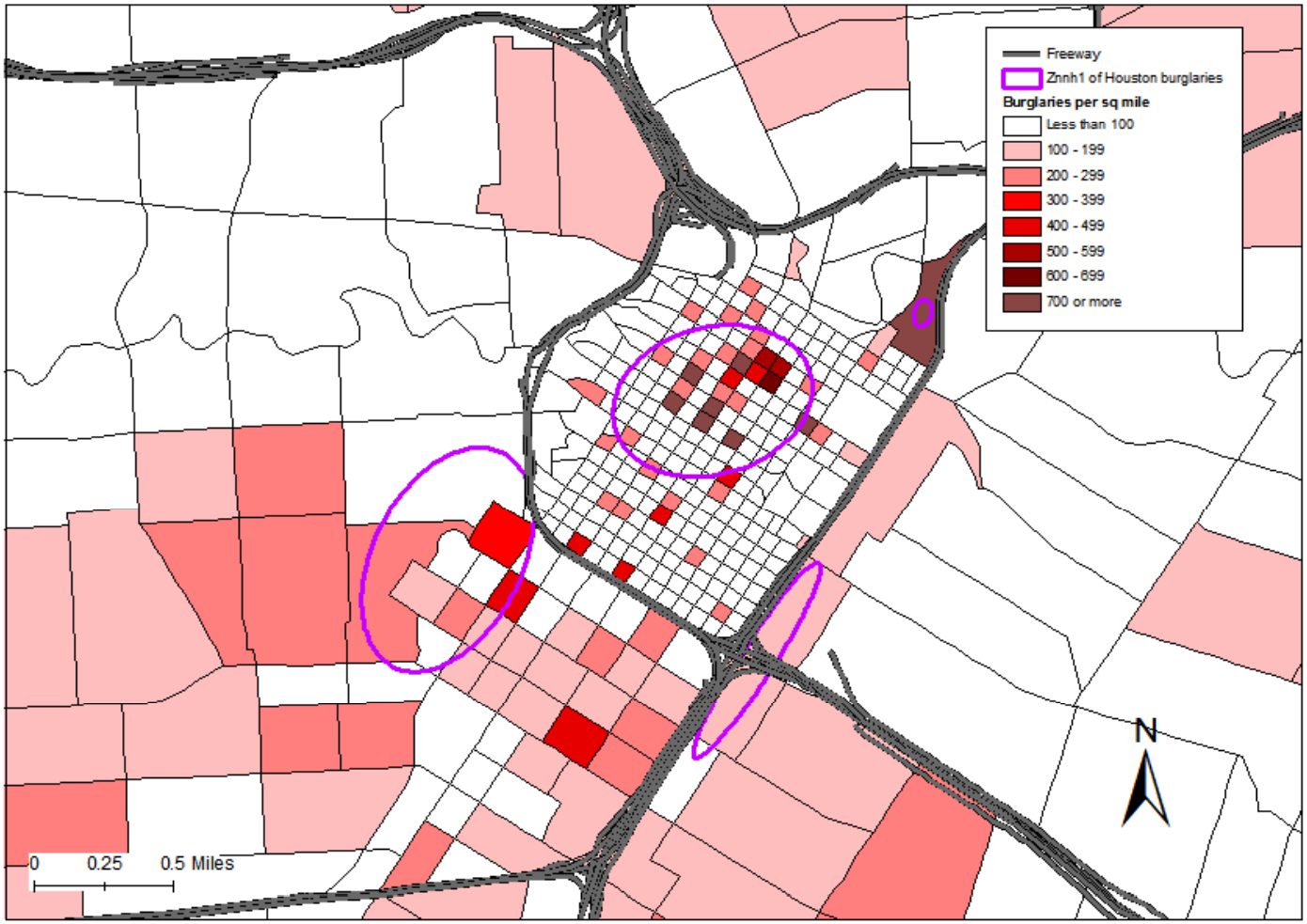


Figure 9.15:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 2 Mile Search Radius and Minimum Number of Events=25

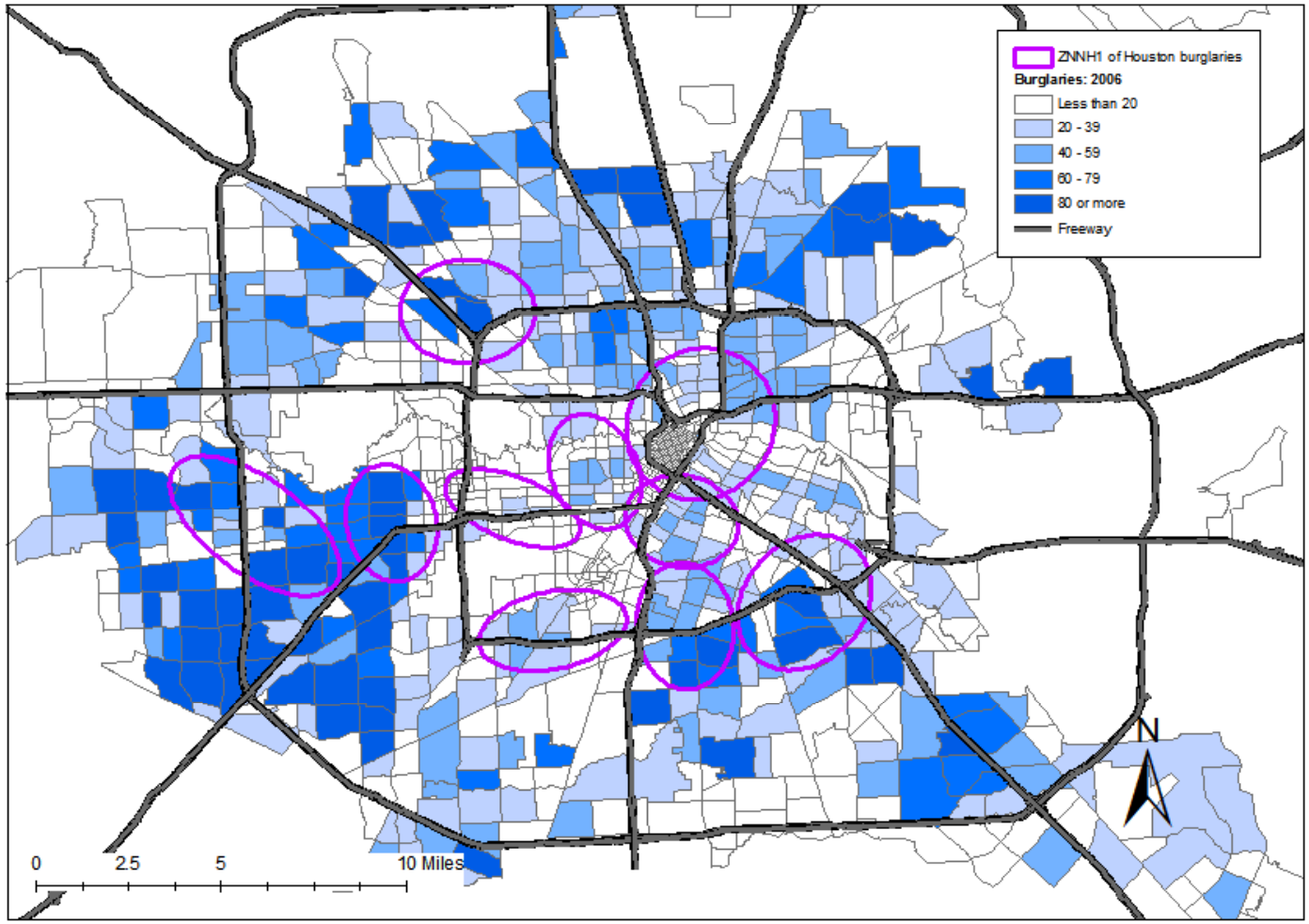


Figure 9.16:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 5 Mile Search Radius and Minimum Number of Events=25

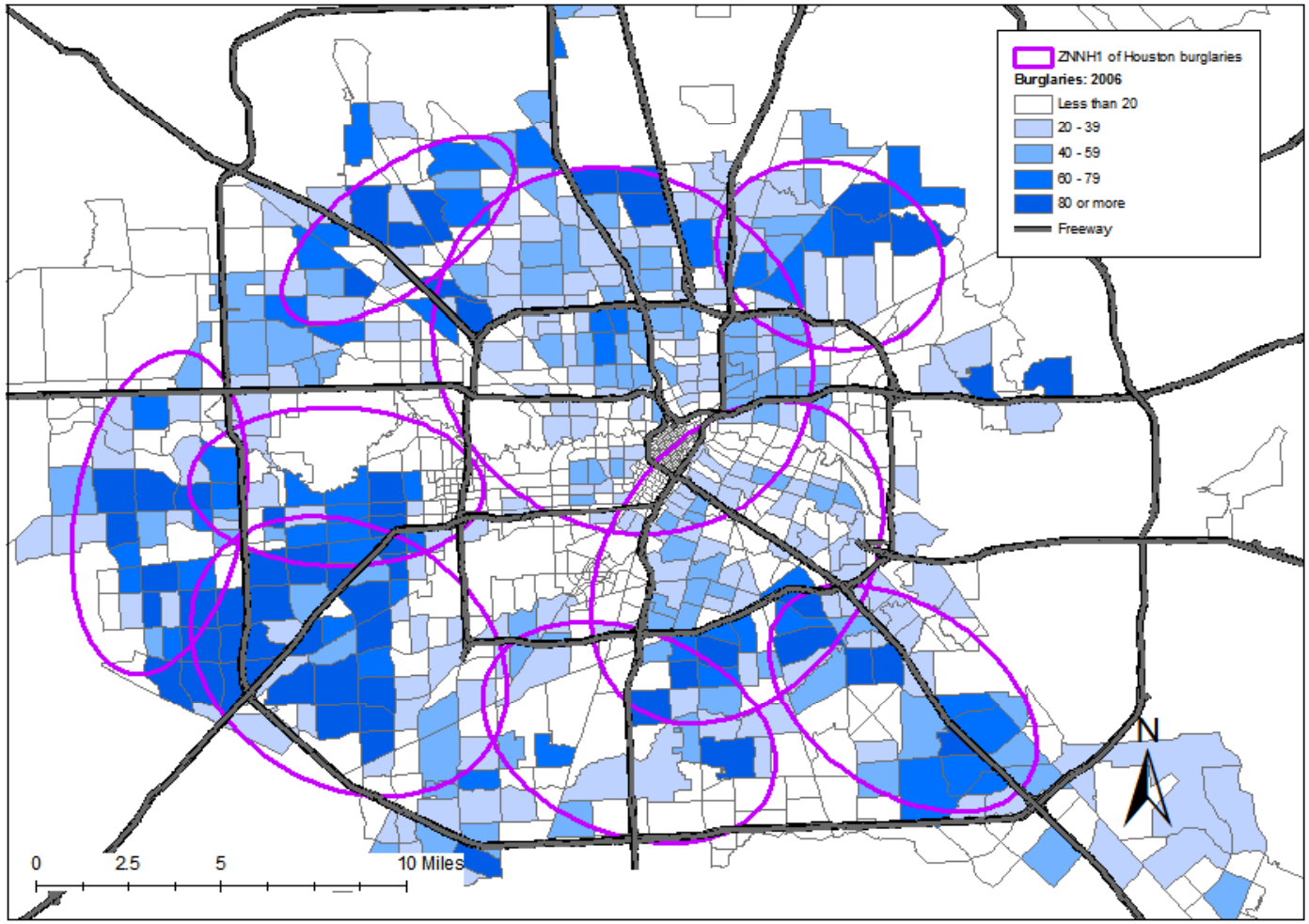


Figure 9.17:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 8 Mile Search Radius and Minimum Number of Events=25

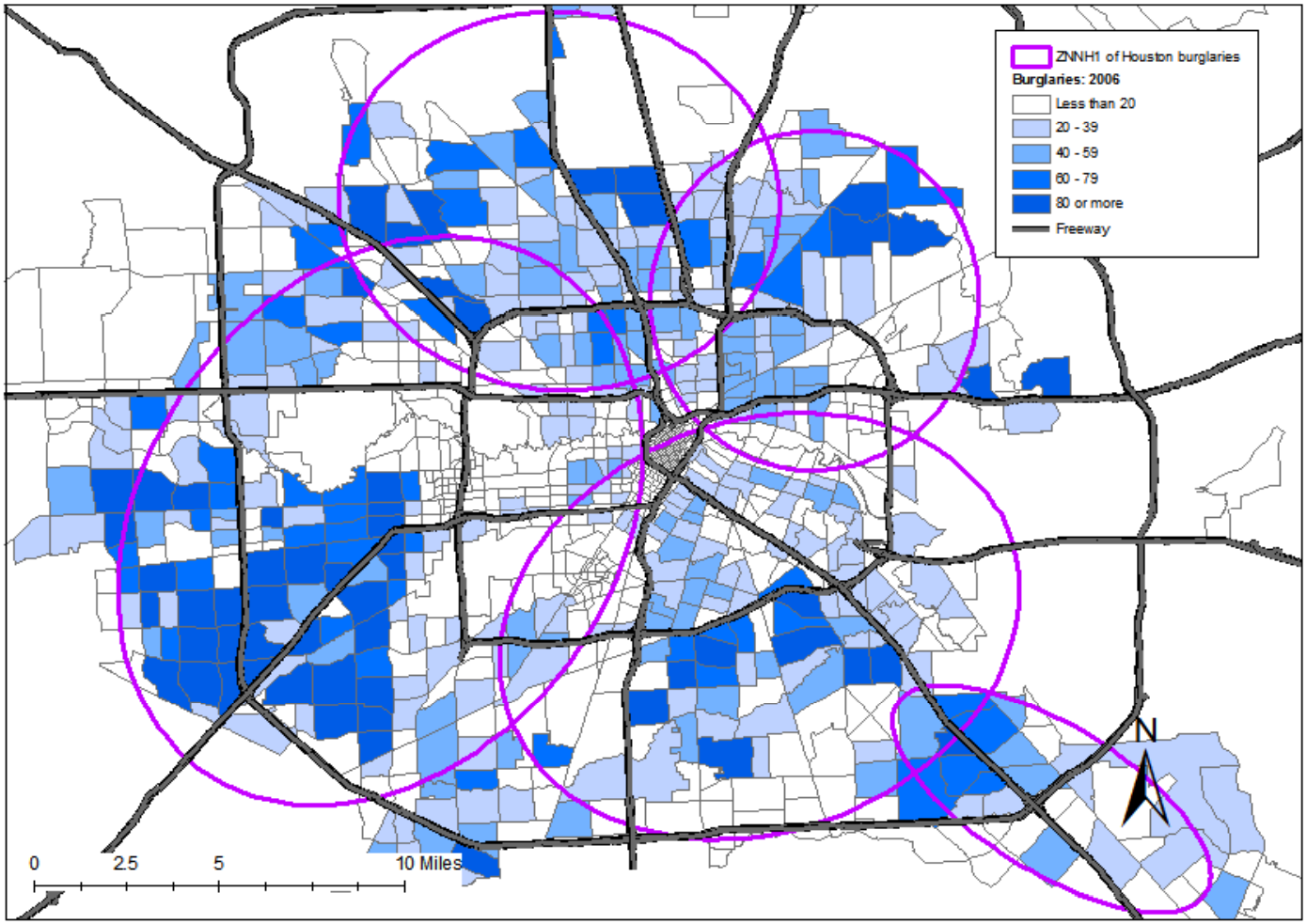


Table 9.4:

Zonal Nearest Neighbor Hierarchical Clustering of Houston Burglaries

(N= 24,935)

0.5 mile search radius, Minimum points per cluster=25

Cluster	Area of Ellipse *(sq mi)*	Number of Points	Number of Zones	Density
1	0.005	56	35	11,633.1
2	0.310	68	155	219.3
3	0.081	29	36	359.4
4	0.374	99	34	264.7

No clusters found in simulation

8 mile search radius, Minimum points per cluster=25

Cluster	Area of Ellipse *(sq mi)*	Number of Points	Number of Zones	Density
1	171.758	12,749	623	74.227
2	99.028	3,253	91	32.849
3	130.048	5,070	288	38.986
4	65.936	2,418	86	36.672
5	31.450	681	26	21.653

Percentile	Clusters	Area of Ellipse *(sq mi)*	Number of Zones	Density
0.5	4	10.34	25	0.707
1.0	5	11.04	25	0.730
2.5	5	12.29	25	0.777
5.0	5	13.74	25	0.862
10.0	5	15.74	26	0.938
90.0	7	239.21	467	2.092
95.0	8	241.59	471	2.183
97.5	8	243.05	477	2.353
99.0	8	244.67	481	2.631
99.5	8	245.06	483	2.798

Uses of Zonal Nearest Neighbor Hierarchical Clustering

This brings up one of the dilemmas in using a zonal clustering technique. On the one hand, since zones do not overlap, the dispersion is much more spread out than with individual events. As seen in Chapter 7, the regular nearest neighbor hierarchical clustering routine (Nnh) produced quite small clusters. With zonal data, however, all the events are assigned to a single point within the zone which either creates a cluster associated with a single or else a dispersion between adjacent zones that have a higher concentration. Since the identification of a single zone is not very useful, the Znnh routine requires a minimum of three adjacent zones to be included in a cluster.

Still, the Znnh can be useful for describing overall cluster patterns in a study area even with the increased uncertainty associated with large search radii. As Figures 9.16 and 9.17 illustrate, meaningful areas of higher concentration can be identified even though the identified clusters cannot be empirically distinguished from a chance distribution. This gives the user flexibility in defining groupings of zones which can then be used for various purposes (e.g., assigning patrols or defining contingency areas).

Limitations of Zonal Nearest Neighbor Hierarchical Clustering

On the other hand, the Znnh routine does have some limitations. The first was shown above, namely that to ensure that clusters are substantially different from that expected by chance, only small search radii can be chosen. However, given that most zones are associated with population density with the smallest zones being in the downtown center but increasing in size with distance from the center, the use of a small search radius becomes less useful.

Second, choosing a larger search radius can produce a cluster structure that appears to fit the data better but cannot be empirically distinguished from a chance distribution. Since there is not a single criterion that can be used to select among these, there is a certain amount of arbitrariness in the selection of a search radius or in the minimum number of events/attribute values specified. A user will have to experiment with different combinations to find the cluster structure that best fits the data. In this sense, the Znnh routine is more similar to the K-means clustering routine discussed in Chapter 8 than the Nnh routine in Chapter 7.

The best solution, of course, is to use the location of individual events and cluster them with either either Nnh, STAC or K-means routines discussed in Chapters 7 and 8. The Znnh routine should only be used if the data are organized by zones and cannot be disaggregated. In this case, the user must be aware of the limitations of the Znnh method and of the trade-off between precision (certainty) and utility.

A third limitation is that the cluster structure will almost certainly be different than had the individual events been clustered using the point-based Nearest Neighbor Hierarchical Clustering routine (Nnh). The requirement that zones do not overlap and that all events are assigned to the centroid of the zone ensures that the Znnh clusters will almost always be larger in size than the point-based Nnh clusters. In short, assigning events to zones and then clustering the zones will produce a larger and less focused cluster structure than the events themselves. The Znnh is only useful when it is not possible to disaggregate events to individual locations.

References

Anselin, L. (1995). Local indicators of spatial association - LISA. *Geographical Analysis*. 27, No. 2 (April), 93-115.

Chainey, S. & Ratcliffe, J. (2005). *GIS and Crime Mapping*, John Wiley & Sons, Inc.:Chichester, Sussex, England.

Getis, A. (1991). Spatial interaction and spatial auto-correlation: a cross-product approach. *Environment and Planning A*, 23, 1269-1277.

Getis, A. & Ord, J. K. (1996). Local spatial statistics: an overview. In Longley, Paul & Batty, Michael (eds), *Spatial Analysis: Modelling in a GIS Environment*. GeoInformation International: Cambridge, England, 261-277.

Mantel, N. (1967). The detection of disease clustering and a generalized regression approach. *Cancer Research*, 27, 209-220.

Waller, L. A. & Gotway, C. A. (2004). *Applied Spatial Statistics for Public Health Data*. John Wiley & Sons: Hoboken, NJ.

Wong, D. W. S. & Lee, J. (2005). *Statistical Analysis of Geographic Information with ArcView GIS and ArcGIS*. J. Wiley & Sons, Inc.: New York.

Endnotes

i. The variance of the Local Moran is defined in three steps:

A. First, define b_2.

$$b_2 = \frac{\sum_{i=1}^{N} \frac{(X_i - \bar{X})^4}{N}}{[\sum_{i=1}^{N} \frac{(X_i - \bar{X})^2}{N}]^2}$$

This is the fourth moment around the mean divided by the squared second moment around the mean.

B. Second, define $2w_{i(kh)}$:

$$2w_{i(kh)} = \sum_{k=1}^{N-1} \sum_{h=1}^{N-1} W_{ik} W_{ih}$$

where k ≠ i and h ≠ i. This term is twice the sum of the cross-products of all weights for *i* with themselves, using k and h to avoid the use of identical subscripts. Since each pair of observations, *i* and *j*, has its own specific weight, a cross-product of weights are two weights multiplied by each other (where i ≠ j) and the sum of these cross-products is twice the sum of all possible interactions irrespective of order (i.e., $W_{ij} = W_{ji}$). Because the weight of an observation with itself is zero (i.e., $W_{ii} = 0$), all terms can be included in the summation.

C. Third, define the variance, standard deviation, and an approximate (pseudo) standardized score of I_i:

$$Var(I_i) = \frac{(\sum_{i=1}^{N} \sum_{j=1}^{N} W_{ij}^2)(N - b_2)}{N-1} + \frac{2w_{i(kh)}(2b_2 - N)}{(N-1)(N-2)} + \frac{(\sum_{i=1}^{N} \sum_{j=1}^{N} W_{ij}^2)}{(N-1)^2}$$

$$S(I_i) = \sqrt{Var(I_i)}$$

$$Z(I_i) = \frac{I_i - E(I_i)}{S(I_i)}$$

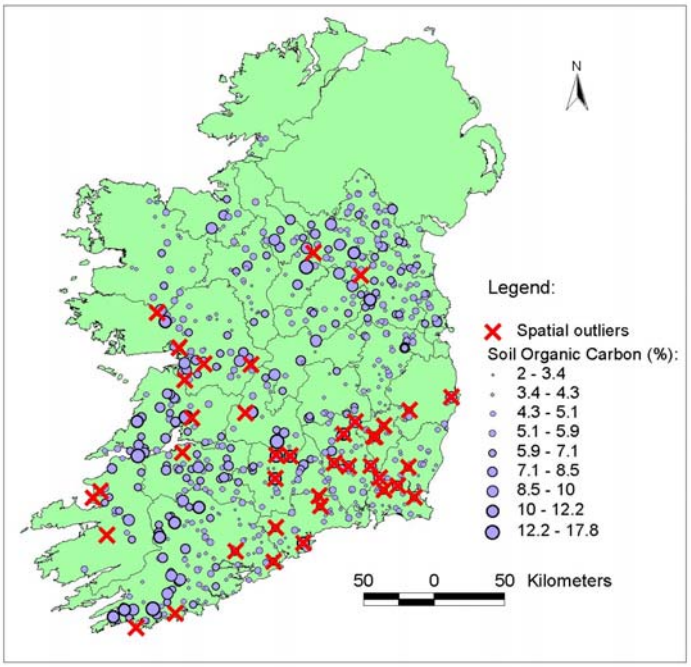
Attachments

Using Local Moran's "I" to Detect Spatial Outliers in Soil Organic Carbon Concentrations in Ireland

Chaosheng Zhang[1] David McGrath[2]
Lecturer in GIS Research Officer
[1] Department of Geography, National University of Ireland, Galway, Ireland
[2] Teagasc, Johnstown Castle Research Centre, Wexford, Ireland

One objective in the study of soil organic carbon concentrations is to produce a reliable spatial distribution map. A geostatistical variogram analysis was applied to study the spatial structure of soils in Ireland for the purpose of carrying out a spatial interpolation with the Kriging method. The variogram looks at similarities in organic carbon concentrations as a function of distance. In the analysis, a relatively poor variogram was observed, and one of the main reasons was the existence of spatial outliers. Spatial outliers make the variogram curve erratic and hard to interpret, and impair the quality of the spatial distribution map.

CrimeStat was used to identify the spatial outliers. The parameter of the standardized Anselin's Local Moran's "I (z)" was used. When $z < -1.96$, the sample was defined as a spatial outlier. Out of 678 soil samples, a total of 39 samples were detected as spatial outliers, and excluded in the spatial structure calculation. As a consequence, the variogram curve was significantly improved. This improvement made the final spatial distribution map more reliable and trustable.



Spatial outliers are clearly different from the majority of samples nearby. Compared with the samples nearby, high value spatial outliers are found in the southeastern part, and low value spatial outliers are located in the western and northern parts of the country.

CrimeStat IV

Part IV: Spatial Modeling I

Chapter 10:

Kernel Density Interpolation

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 10:
Kernel Density Interpolation

Introduction

In this chapter, we discuss tools aimed at interpolating incidents, using the kernel density approach. *Kernel Density Interpolation* (sometimes called *Kernel Density Estimation*) is a technique for generalizing incident locations to an entire area. Whereas the spatial distribution and hot spot statistics provide statistical summaries for the data incidents themselves, interpolation techniques generalize those data incidents to the entire region. In particular, they provide *density* estimates for all parts of a region (i.e., at any location). The density estimate is an intensity variable, a Z-value, that is estimated at a particular location. Consequently, it can be displayed by either surface maps or contour maps that show the intensity at all locations.

There are many interpolation techniques, such as Kriging, trend surfaces, local regression models (e.g., Loess, splines), and Dirichlet tessellations (Anselin, 1992; Cleveland, Grosse & Shyu, 1993; Venables & Ripley, 1997). Most of these require a variable that is being estimated as a function of location. However, *kernel density estimation* is an interpolation technique that is appropriate for individual point locations (Silverman, 1986; Härdle, 1991; Bailey & Gatrell, 1995; Burt & Barber, 1996; Bowman & Azalini, 1997).

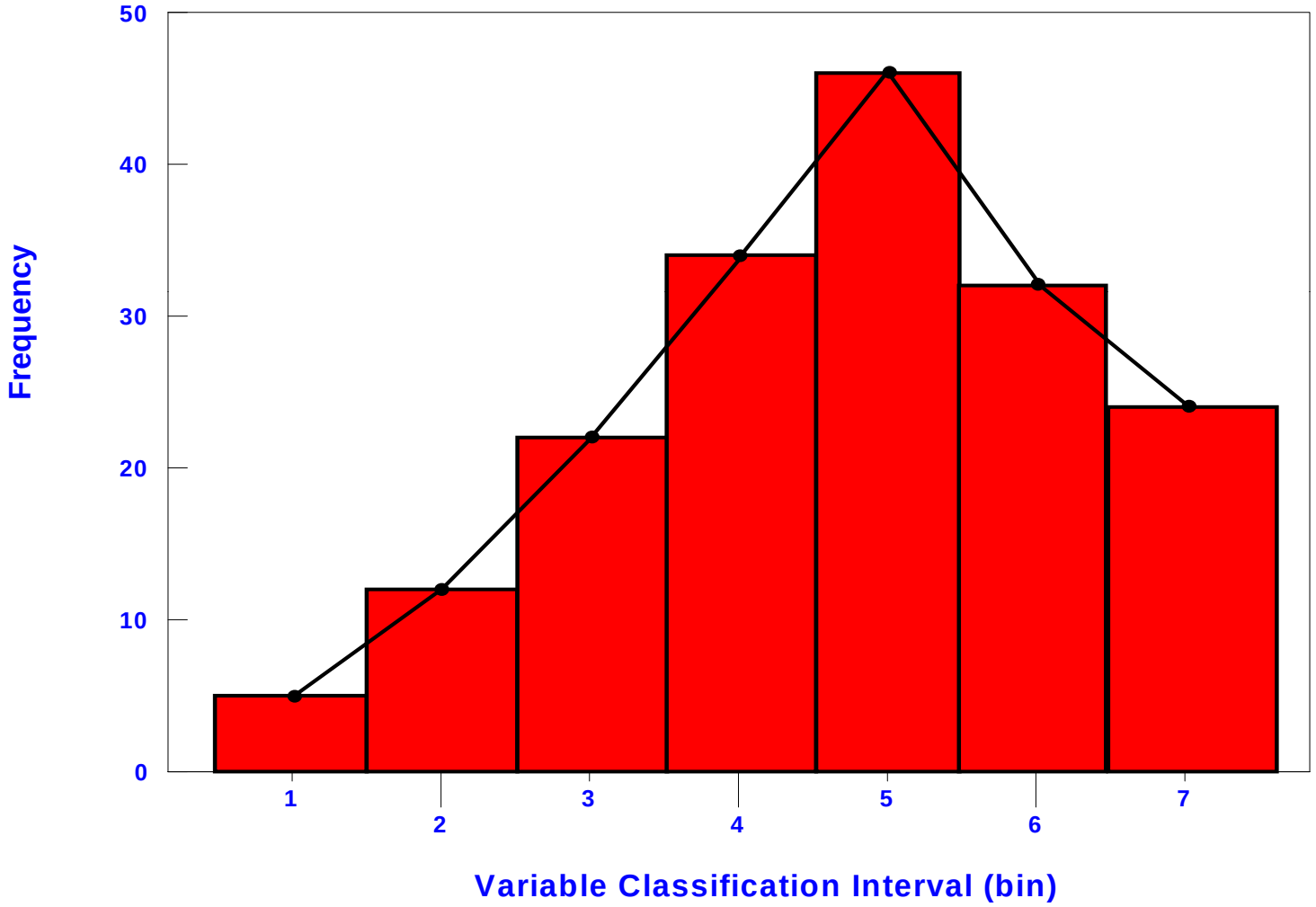
Kernel Density Estimation

Kernel density estimation involves placing a symmetrical surface over each point, evaluating the distance from the point to a reference location based on a mathematical function, and summing the value of all the surfaces for that reference location. This procedure is repeated for all reference locations. It is a technique that was developed in the late 1950s as an alternative method for estimating the density of a histogram (Rosenblatt, 1956; Whittle, 1958; Parzen, 1962). A histogram is a graphic representation of a frequency distribution. A continuous variable is divided into intervals of size, s (the interval or bin width), and the number of cases in each interval (bin) are counted and displayed as block diagrams. The histogram is assumed to represent a smooth, underlying distribution (a density function). However, in order to estimate a smooth density function from the histogram, traditionally researchers have linked adjacent variable intervals by connecting the midpoints of the intervals with a series of lines (Figure 10.1).

Figure 10.1:

CONSTRUCTING A DENSITY ESTIMATE FROM HISTOGRAM

Method of Connecting Midpoints



Kernel Estimates as an Alternative to Histograms

Unfortunately, doing this causes three statistical problems (Bowman & Azalini, 1997):

1. Information is discarded because all cases within an interval are assigned to the midpoint. The wider the interval, the greater the information loss.

2. The technique of connecting the midpoints leads to a discontinuous and not smooth density function even though the underlying density function is assumed to be smooth. To compensate for this, researchers will reduce the width of the interval. Thus, the density function becomes smoother with smaller interval widths, although still not very smooth. Further, there are limits to this technique as the sample size decreases when the bin width gets smaller, eventually becoming too small to produce reliable estimates.

3. The technique is dependent on an arbitrarily defined interval size (bin width). By making the interval wider, the estimator becomes cruder and, conversely, by making the interval narrower, the estimator becomes finer. However, the underlying density distribution is assumed to be smooth and continuous and not dependent on the interval size of a histogram.

To handle this problem, Rosenblatt (1956), Whittle (1958) and Parzen (1962) developed the kernel density method in order to avoid the first two of these difficulties; the bin width issue still remains. What they did was to place a smooth *kernel function* over each point and sum the functions for each location on the scale. Figure 10.2 illustrates the process with five point locations. As seen, over each location, a symmetrical kernel function is placed; by symmetrical is meant that is falls off with distance from each point at an equal rate in both directions around each point. In this case, it is a normal distribution, but other types of symmetrical distribution have been used. The underlying density distribution is estimated by summing the individual kernel functions at *all* locations to produce a smooth cumulative density function. Notice that the functions are summed at every point along the scale and not just at the point locations. The advantages of this are that, first, each point contributes equally to the density surface and, second, the resulting density function is continuous at all points along the scale.

The third problem mentioned above, interval size, still remains since the width of the kernel function can be varied. In the kernel density literature, this is called *bandwidth* and refers essentially to the width of the kernel. Figure 10.3 shows a kernel with a narrow bandwidth placed over the same five points while Figure 10.4 shows a kernel with a wider bandwidth placed over the points. Clearly, the smoothness of the resulting density function is a result of the bandwidth size.

Figure 10.2:

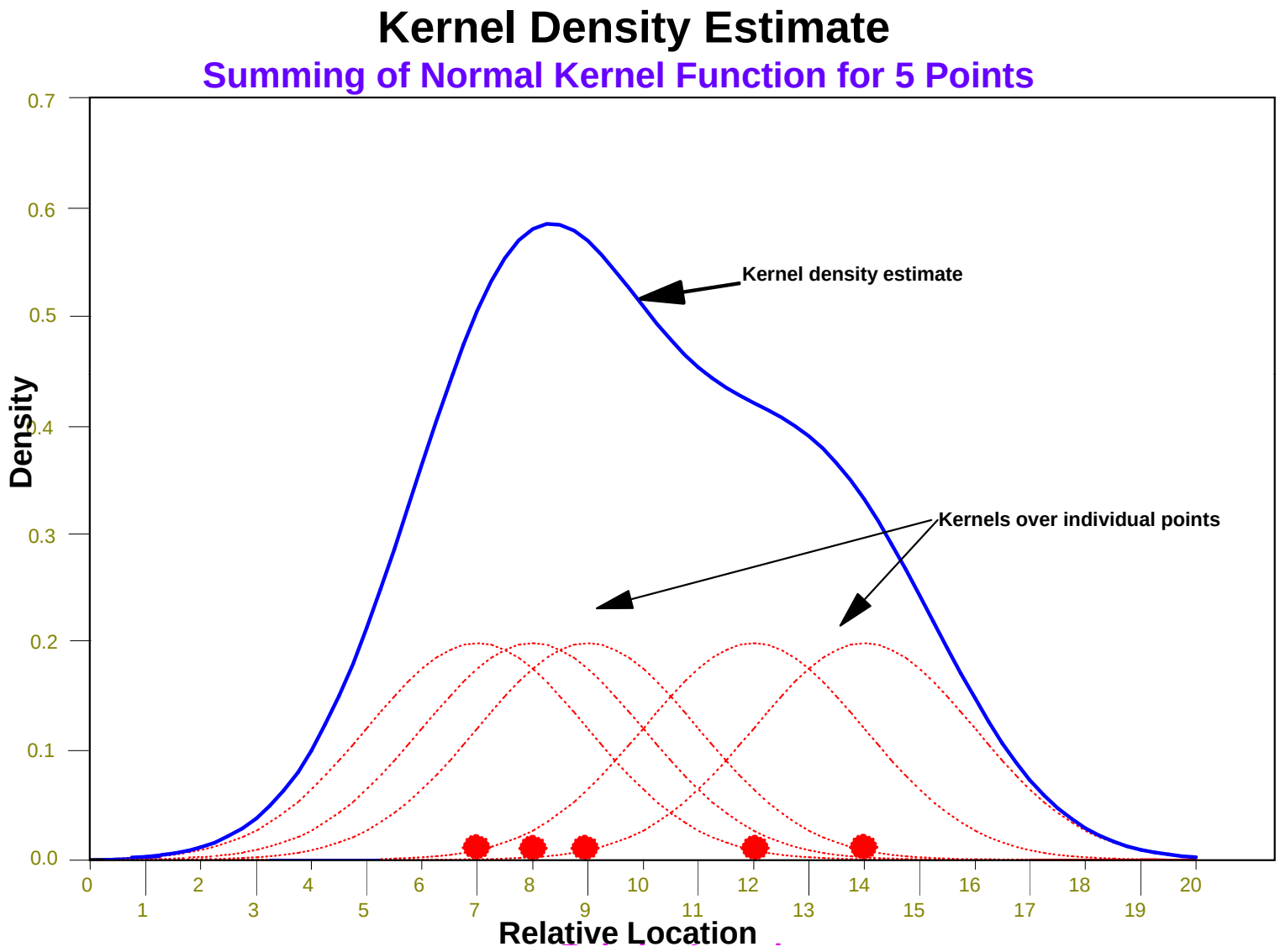

Kernel Density Estimate
Summing of Normal Kernel Function for 5 Points
Kernel density estimate
Kernels over individual points
Density
Relative Location
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

Figure 10.3:

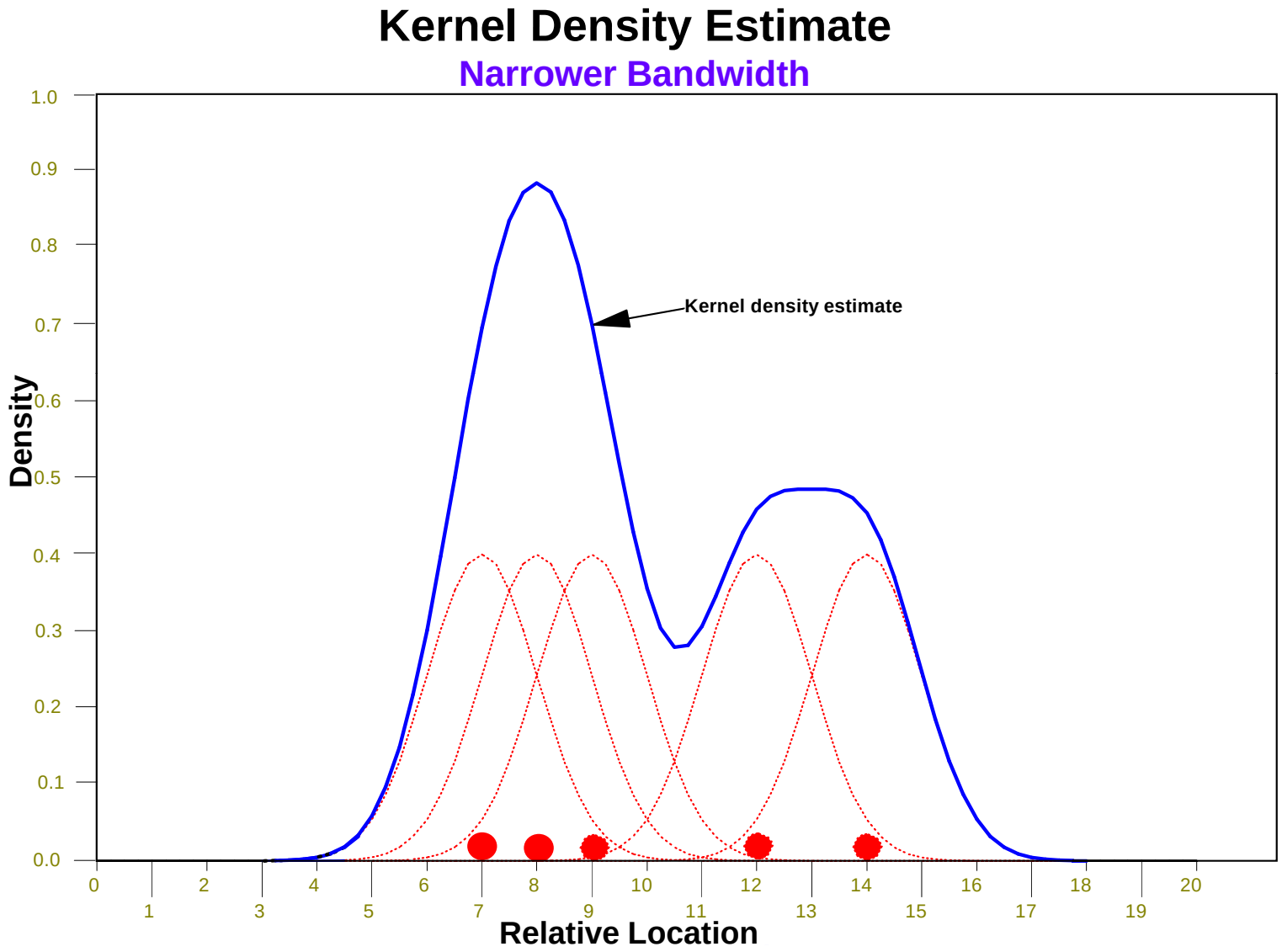
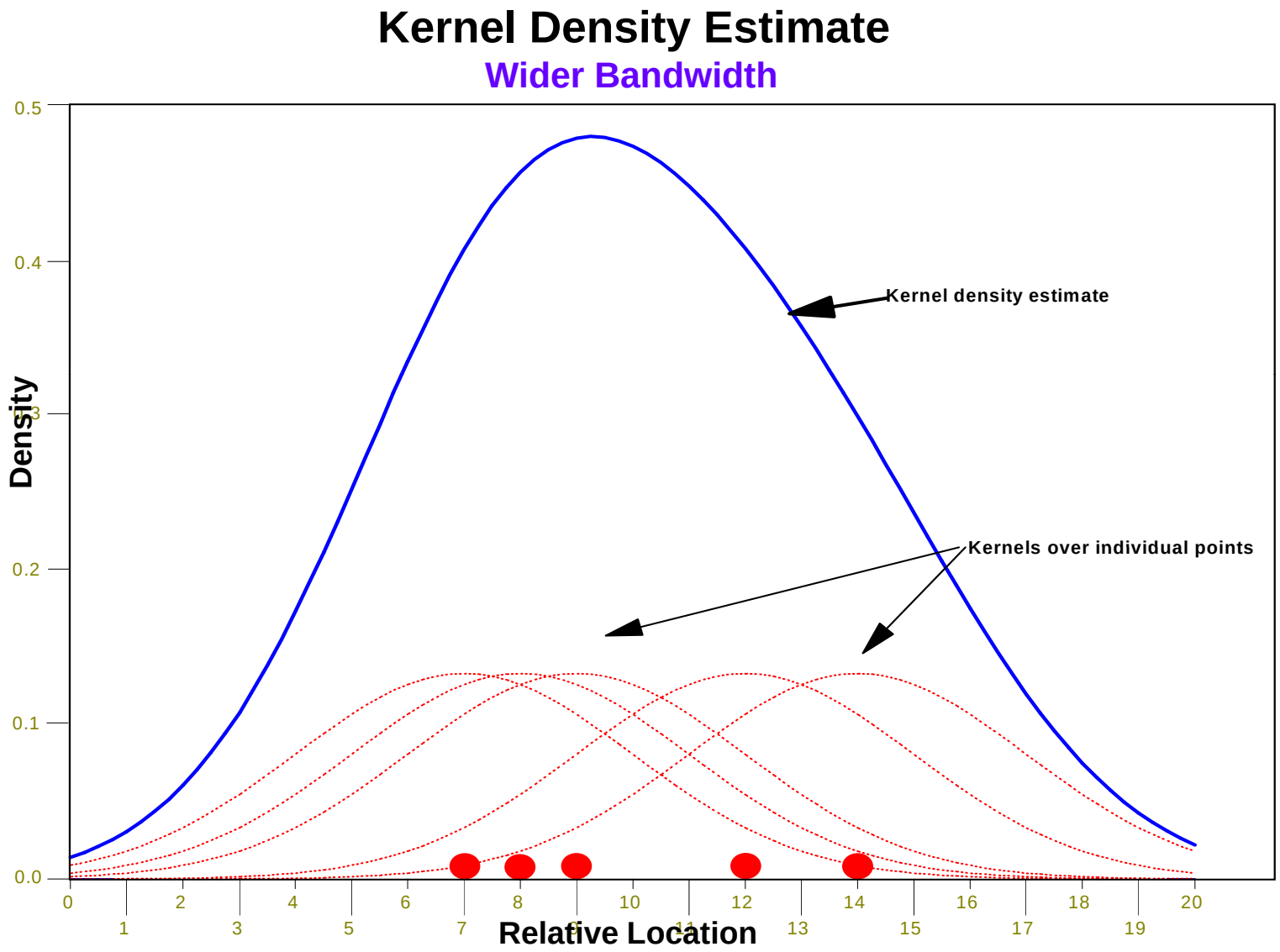

Kernel Density Estimate
Narrower Bandwidth
Kernel density estimate
Density
Relative Location
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

Figure 10.4:


Kernel Density Estimate
Wider Bandwidth
Density
0.5
0.4
0.3
0.2
0.1
0.0
Kernel density estimate
Kernels over individual points
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
Relative Location

Kernel Functions

There are a number of different kernel functions that have been used in applications. Figure 10.5 illustrates five different kernel functions that are available in *CrimeStat*.

The first is the ***normal distribution*** and is the most commonly used (Kelsall & Diggle, 1995a). It has the following functional form:

$$g(j) = \sum_{i=1}^{N} \left[KW_i I_i \frac{1}{h^2 2\pi} e^{-\frac{d_{ij}^2}{2h^2}} \right] \quad (10.1)$$

where $g(x_j)$ is the density of cell j, d_{ij} is the distance between cell j and an incident location, i, h is the standard deviation of the normal distribution (the bandwidth), K is a constant, W_i is a weight at the point location and I_i is an intensity at the point location. This function extends to infinity in all directions and, thus, will be applied to any location in the region. In *CrimeStat*, the constant K is initially set to 1 and then re-scaled to ensure that either the densities or probabilities sum to their appropriate values (i.e., N for densities and 1.00 for probabilities).

In other words, the density of cell j is the sum over all incidents of a distance function where the function is the normal distribution. Each cell, in turn, is evaluated with this function and the result is a density estimate for every cell in the reference grid.

In *CrimeStat*, there are four alternative kernel functions that can be used, all of which have a circumscribed bandwidth (search area) unlike the normal distribution. The ***quartic*** function is applied to a limited area around each incident point defined by the radius, h. It falls off gradually with distance until the bandwidth radius is reached. Its functional form is:

1. Outside the specified bandwidth, h:

$$g(j) = 0 \quad (10.2)$$

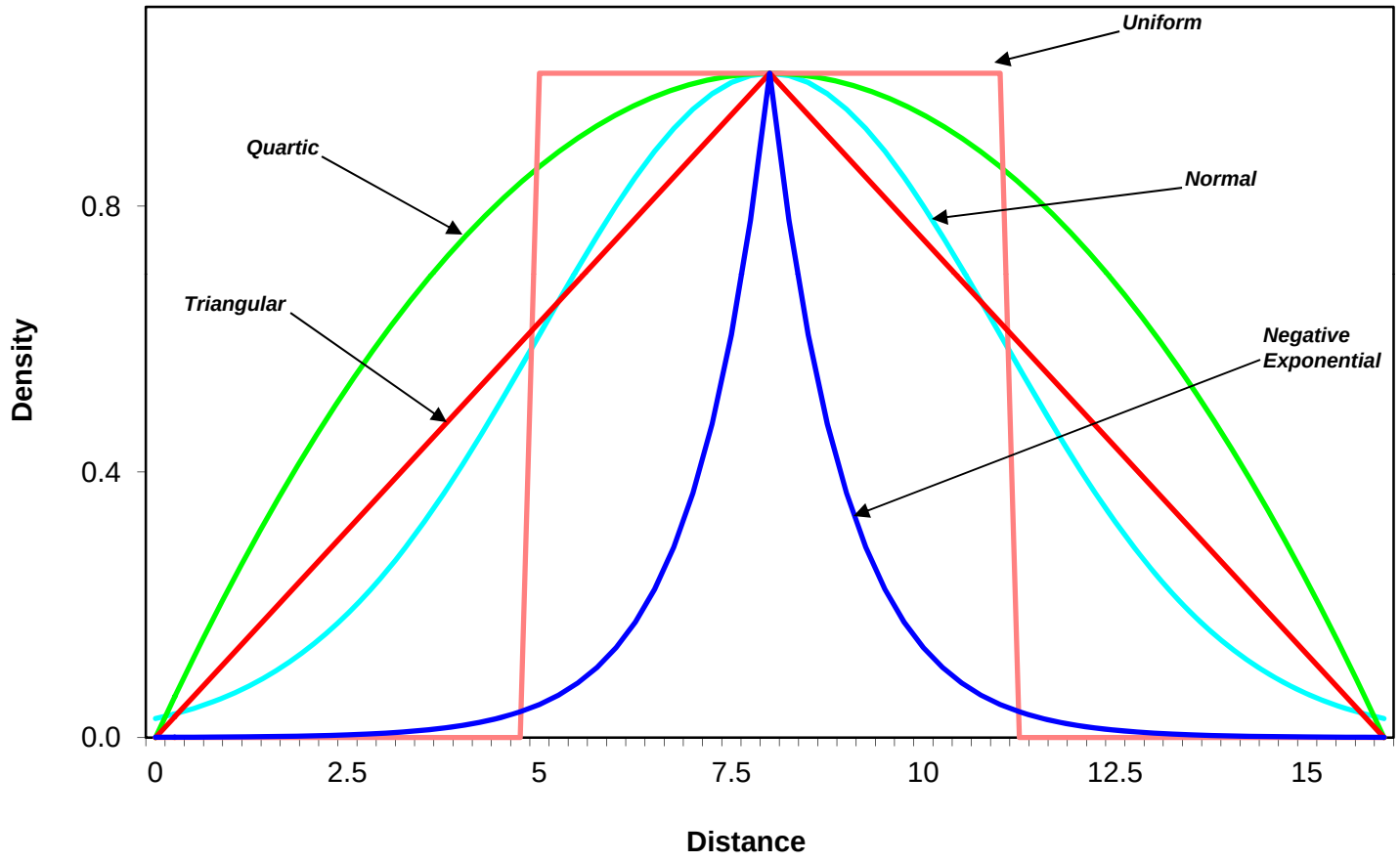
2. Within the specified bandwidth, h:

$$g(j) = \sum_{i=1}^{M_j} \left[KW_i I_i \frac{3}{h^2 2\pi} (1 - \frac{d_{ij}^2}{h^2})^2 \right] \quad (10.3)$$

where $g(j)$ is the density of cell j, d_{ij} is the distance between cell j and an incident location, i, h is the radius of the search area (the bandwidth) , K is a constant, W_i is a weight at the point location, and I_i is an intensity at the point location. The summation is for the incidents that are within the

Figure 10.5:

Five Types of Kernel Functions



bandwidth. Thus, each cell, j, has a different number of incidents that fall within the bandwidth search area, M_j. In *CrimeStat*, the constant K is initially set to 1 and then re-scaled to ensure that either the densities or probabilities sum to their appropriate values (i.e., N for densities and 1.00 for probabilities).

The ***triangular*** (or conical) distribution falls off evenly with distance, in a linear relationship. It also has a circumscribed radius and is, therefore, applied to a limited area around each incident point, h. Compared to the quartic function, it decays more rapidly. Its functional form is:

1. Outside the specified bandwidth, *h*:

$$g(j) = 0 \tag{10.4}$$

2. Within the specified bandwidth, *h*:

$$g(j) = \sum_{i=1}^{M_j}[W_i I_i (K - \frac{K}{h}) d_{ij}] \tag{10.5}$$

where $g(x_j)$ is the density of cell j, d_{ij} is the distance between cell j and an incident location, i, h is the radius of the search area (the bandwidth) , K is a constant, W_i is a weight at the point location, and I_i is an intensity at the point location. The summation is for the incidents that are within the bandwidth. Thus, each cell, j, has a different number of incidents that fall within the bandwidth search area, M_j. In *CrimeStat*, the constant K is initially set to 0.25 and then re-scaled to ensure that either the densities or probabilities sum to their appropriate values (i.e., N for densities and 1.00 for probabilities).

The ***negative exponential*** (or peaked) distribution falls off very rapidly with distance up to the circumscribed radius. Its functional form is:

1. Outside the specified bandwidth, *h*:

$$g(j) = 0 \tag{10.6}$$

2. Within the specified bandwidth, *h*:

$$g(j) = \sum_{i=1}^{M_j} W_i I_i K e^{A d_{ij}} \tag{10.7}$$

where $g(x_j)$ is the density of cell j, d_{ij} is the distance between cell j and an incident location, i, h is the radius of the search area (the bandwidth) , K is a constant, A is an exponent, W_i is a weight at the point location, and I_i is an intensity at the point location. The summation is for the

incidents that are within the bandwidth. Thus, each cell, j, has a different number of incidents that fall within the bandwidth search area, M_j. In *CrimeStat*, *A* is set to 3 while *K* is initially set to 1 and then re-scaled to ensure that either the densities or probabilities sum to their appropriate values (i.e., N for densities and 1.00 for probabilities).

Finally, the ***uniform*** distribution weights all points within the circle equally. Its functional form is:

1. Outside the specified bandwidth, *h*:

$$g(j) = 0 \tag{10.8}$$

2. Within the specified bandwidth, *h*:

$$g(j) = \sum_{i=1}^{M_j} W_i I_i K \tag{10.9}$$

where $g(x_j)$ is the density of cell j, K is a constant, W_i is a weight at the point location, and I_i is an intensity at the point location. The summation is for the incidents that are within the bandwidth. Thus, each cell, j, has a different number of incidents that fall within the bandwidth search area, M_j. Initially, K is set to 0.1 but then re-scaled to ensure that either the densities or probabilities sum to their appropriate values (i.e., N for densities and 1.00 for probabilities).

Kernel Parameters

The user can select these five different kernel functions to interpolate the data to the grid cells. They produce subtle differences in the shape of the interpolated surface or contour. The normal distribution weighs all points in the study area, though near points are weighted more highly than distant points. The other four techniques use a circumscribed circle around the grid cell. The uniform distribution weighs all points within the circle equally. The quartic function weighs near points more than far points, but the fall off is gradual. The triangular function weighs near points more than far points within the circle, but the fall off is more rapid. Finally, the negative exponential weighs near points much more highly than far points within the circle and the decay is very rapid.

The use of any of one of these depends on how much the user wants to weigh near points relative to far points. Using a kernel function which has a big difference in the weights of near versus far points (e.g., the negative exponential or the triangular) tends to produce finer variations within the surface than functions which weight more evenly (e.g., the normal distribution, the quartic, or the uniform); these latter ones tend to *smooth* the distribution more.

Shape and size of the bandwidth

However, Silverman (1986) has argued that it does not make that much difference as long as the kernel is symmetrical. There are also edge effects that can occur and there have been different proposed solutions to this problem (Venables & Ripley, 1997).

There have also been variations on the size of the bandwidth with various formulas and criteria proposed (Silverman, 1986; Härdle, 1991; Venables & Ripley, 1997). Generally, bandwidth choices fall into either fixed or adaptive (variable) kernels (Kelsall & Diggle, 1995a; Bailey & Gatrell, 1995). *CrimeStat* follows this distinction, which will be explained below.

Another suggestion is to use the Moran correlogram, which was discussed in Chapter 5, to estimate the shape of the weighting function (Cliff & Haggett, 1988; Bailey & Gattrell, 1995). This would be appropriate for variables that have *weights*, such as population or employment. The Moran correlogram displays the degree of spatial autocorrelation as a function of distance. Whether the autocorrelation falls off quickly or more slowly can be used to select an approximate kernel function (e.g., a negative exponential function decays quickly whereas a quartic function decays very slowly). The bandwidth could also be selected by the distance at which the Moran correlogram levels off (i.e., approaches the global I value). This would lead to an estimate that minimizes spatial autocorrelation in the data set. It would be good for capturing major trends in the data, but would not be good for identifying local clusters (hot spots) since the bandwidth distance would incorporate most of a metropolitan area.

Three-dimensional kernels

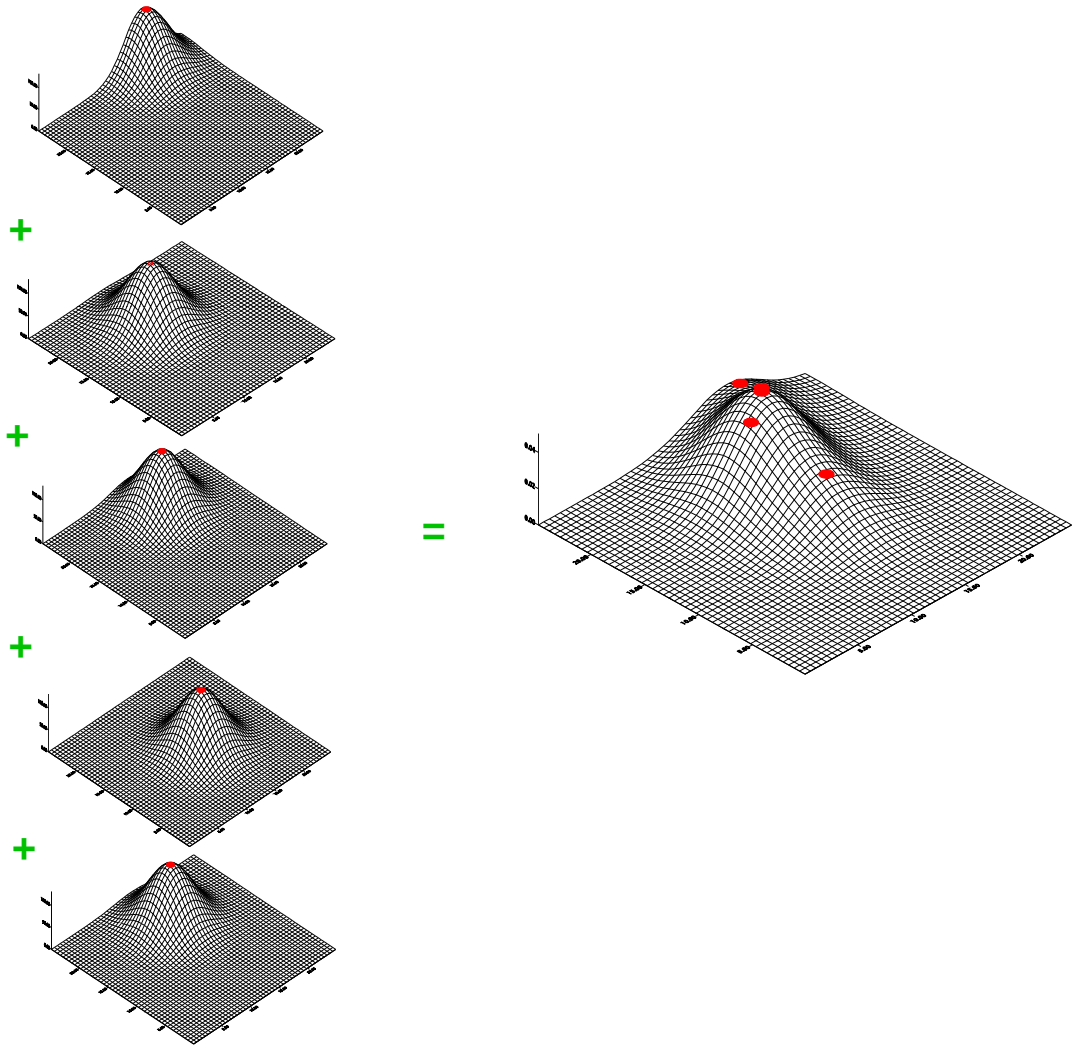
The kernel function can be expanded to more than two dimensions (Härdle, 1991; Bailey & Gatrell, 1995; Burt & Barber, 1996; Bowman & Azalini, 1997). Figure 10.6 shows a three-dimensional normal distribution placed over each of five points with the resulting density surface being a sum of all five individual surfaces. Thus, the method is particularly appropriate for geographical data, such as crime incident locations. The method has also been developed to relate two or more variables together by applying a kernel estimate to each variable in turn and then dividing one by the other to produce a three-dimensional estimate of *risk* (Kelsall & Diggle, 1995a; Bowman & Azalini, 1997).

Significance testing of density estimates is more complicated. Current techniques tend to focus on simulating surfaces under spatially random assumptions (Bowman & Azalini, 1997; Kelsall & Diggle, 1995b). Because of the still experimental nature of the testing, *CrimeStat* does not include any testing of density estimates in this version.

Figure 10.6:

Kernel Density Surface

Summing of Normal Kernel Surface for 5 Points



CrimeStat Kernel Density Methods

CrimeStat has two kernel density interpolation routines. The first applies to a single variable while the second to the relationship between two variables. Both routines have a number of options. In addition, kernel density interpolation is used in several other *CrimeStat* routines including journey-to-crime modeling, Bayesian journey-to-crime modeling, and Head-Bang interpolation. Those latter techniques will be discussed in Chapters 11, 13, and 14.

Figure 10.7 shows the Interpolation I screen in *CrimeStat* and the two routines that are available. Users indicate their choices by clicking on the tab and menu items. For either technique, it is necessary to have a reference file, which is usually a grid placed over the study region (see chapter 3). The reference file represents the region to which the kernel estimate will be generalized. Figure 10.8 illustrates a reference grid over the Baltimore region with 100 columns and 90 rows.

Single Kernel Density Interpolation

The **single kernel density** routine in *CrimeStat* is applied to a distribution of point locations, such as crime incidents. It can be used with either a primary file or a secondary file; the primary file is the default. For example, the primary file can be the location of motor vehicle thefts. The points can also have a weighting or an associated intensity variable (or both). For example, the points could represent the location of police stations while the weights (or intensities) represent the number of calls for service. Again, the user must be careful in having both weighting and intensity variables as the routine will use both variables in calculating densities, which could lead to double weighting.

It is necessary to define the appropriate file on the Primary or Secondary file pages. Also, it is necessary to define a reference file, either an existing file or one generated by *CrimeStat* (see Chapter 3). There are other parameters that must be defined.

File to be Interpolated

First, the user must indicate whether the Primary file or the Secondary file (if used) is to be interpolated.

Figure 10.7:

Interpolation I Screen

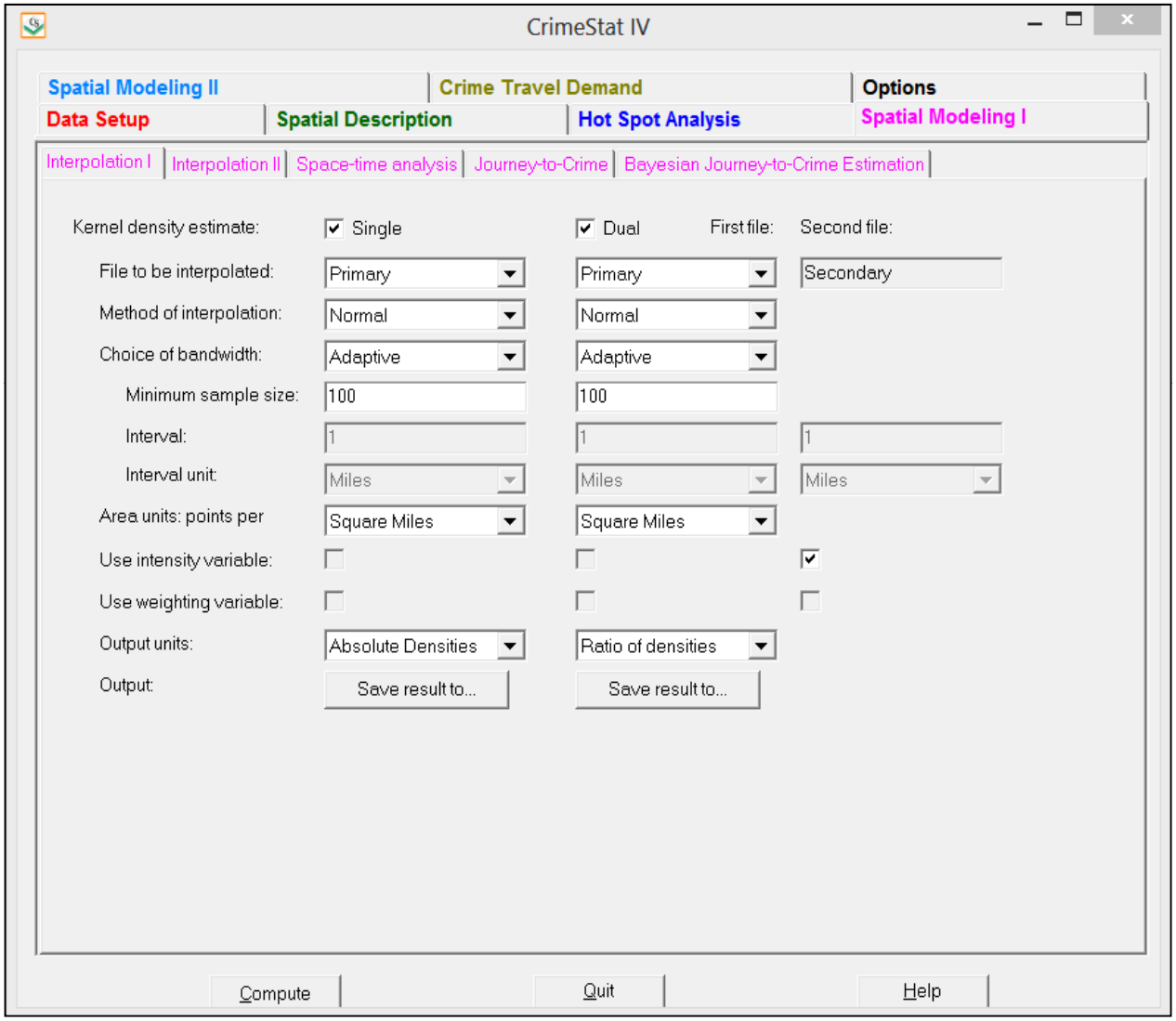
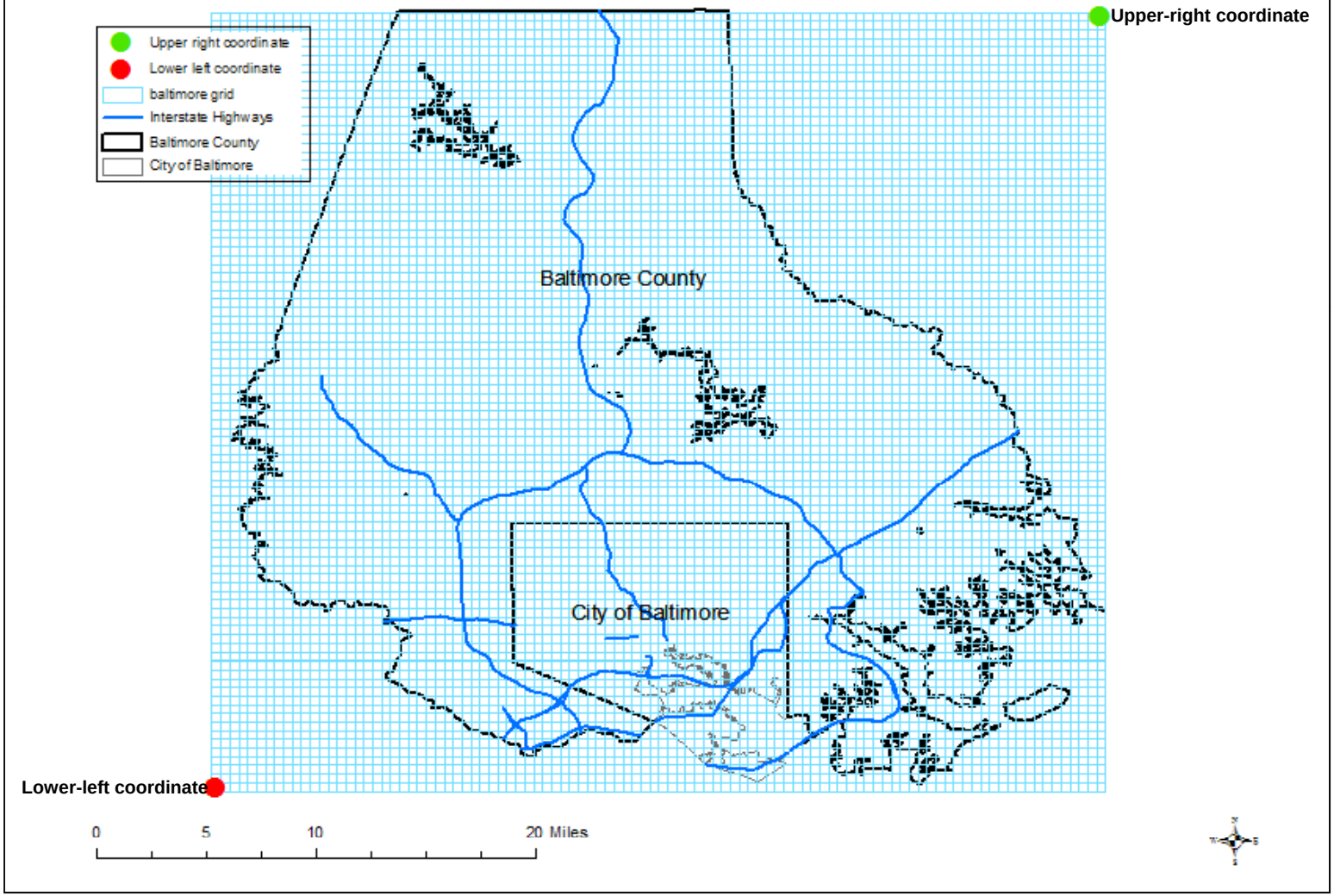


Figure 10.8:

Grid Cell Structure for Baltimore Region

100 Columns by 90 Rows



Method of Interpolation

Second, the user must indicate the method of interpolation. Five types of kernel density estimators are used:

1. Normal distribution (bell; default)
2. Uniform (flat) distribution
3. Quartic (spherical) distribution
4. Triangular (conical) distribution
5. Negative exponential (peaked) distribution

In our experience, there are advantages to each. The normal distribution produces an estimate over the entire region whereas the other four produce estimates only for the circumscribed bandwidth radius. If the distribution of points is sparse towards the outer parts of the region, then the four circumscribed functions will not produce estimates for those areas, whereas the normal will. Conversely, the normal distribution can cause some edge effects to occur (e.g., spikes at the edge of the reference grid), particularly if there are many points near one of the boundaries of the study area. The four circumscribed functions will produce less of a problem at the edges, although they still can produce some spikes. Within the four circumscribed functions, the uniform and quartic tend to smooth the data more whereas the triangular and negative exponential tend to emphasize 'peaks' and 'valleys'. The differences between these different kernel functions are small, however. The user should probably start with the default normal function and adjust accordingly to how the surface or contour looks.

Choice of Bandwidth

Third, the user must indicate how bandwidths are to be defined. There are two types of bandwidth for the single kernel density routine, fixed interval or adaptive interval.

Fixed interval

With a fixed bandwidth, the user must specify the interval to be used and the units of measurement (square miles, square nautical miles, square feet, square kilometers, or square meters). Depending on the type of kernel estimate used, this interval has a slightly different meaning. For the normal kernel function, the bandwidth is the standard deviation of the normal distribution. On the other hand, for the uniform, quartic, triangular, or negative exponential kernels, the bandwidth is the radius of the search area to be interpolated.

There are few guidelines for choosing a particular bandwidth other than by visual inspection (Venables & Ripley, 1997). Some have argued that the bandwidth be no larger than

the finest resolution that is desired and others have argued for a variation on random nearest neighbor distances (see Spencer Chainey's article in the attachments section of this chapter). Others have argued for particular sizes (Silverman, 1986; Härdle, 1991; Kadafar, 1996; Farewell, 1999; Talbot, Kulldorff, Forand, & Haley, 2000; see endnote ***i***). There does not seem to be consensus on this issue. Consequently, *CrimeStat* leaves the definition up to the user.

Typically, a narrower bandwidth interval will lead to a finer mesh density estimate with lots of 'peaks and valleys'. A larger bandwidth interval, on the other hand, will lead to a smoother distribution and, therefore, less variability between areas. While smaller bandwidths show greater differentiation among areas (e.g., between 'hot spot' and 'low spot' zones), one has to keep in mind the statistical precision of the estimate. If the sample size is not very large, then a smaller bandwidth will lead to more imprecision in the estimates, and the 'peaks and valleys' may show nothing more than random variation. On the other hand, if the sample size is large, then a finer density estimate can be produced. In general, it is a good idea to experiment with different fixed intervals to see which results make the most sense.

Adaptive interval

An adaptive bandwidth adjusts the bandwidth interval so that a ***minimum*** number of points are found. This has the advantage of providing constant precision of the estimate over the entire region. Thus, in areas that have a high concentration of points, the bandwidth is narrow whereas in areas where the concentration of points is sparser, the bandwidth will be larger. This is the default bandwidth choice in *CrimeStat* since we believe that consistency in statistical precision is paramount. The degree of precision is generally dependent on the sample size of the bandwidth interval. The default is a minimum of 100 points within the bandwidth radius. The user can make the estimate more fine grained by choosing a smaller number of points (e.g., 25) or more smooth by choosing a larger number of points (e.g., 200). Again, experimentation is necessary to see which results make the most sense.

Output Unit

Fourth, the user must indicate the measurement units for the density estimate in points per square miles, square nautical miles, square feet, square kilometers, or square meters. The default is points per square mile.

Intensity or Weighting Variable

If an intensity or weighting variable is to be used (and has been defined on the Primary or Secondary file page), the appropriate box must be checked. Be careful about using both intensity and weighting variables to avoid 'double weighting'.

Density Calculation

Finally, the user must indicate the type of output for the density estimates. There are three types of calculation that can be conducted with the kernel density routine. The calculations are applied to each reference cell:

1. The kernel estimates can be calculated as *absolute density* estimates using equations 10.1-10.9, depending on what type of kernel function is used. The estimates at each reference cell are re-scaled so that the sum of the densities over all reference grids equals the total number of incidents. That is, the estimate is the number of incidents/points that occurred in each grid cell. This is the default choice.

2. The kernel estimates can be calculated as *relative density* estimates. These divide the absolute densities by the area of the grid cell. It has the advantage of interpreting the density in terms that are familiar. Thus, instead of a density estimate represented by points per grid cell, the relative density will convert this to points per square mile or points per square kilometer.

3. The densities can be converted into *probabilities* by dividing the density at any one cell by the total number of incidents.

Since the three types of calculation are directly interrelated, the output surface will not differ in its variability. The choice would depend on whether the calculations are used to estimate absolute densities, relative densities, or probabilities. For comparisons between different types of crime or between the same type of crime and different time periods, usually absolute densities are the unit of choice (i.e., incidents per grid cell). However, to express the output as a probability, that is, the likelihood that an incident would occur at any one location, then outputting the results as probabilities would make more sense. For display purposes, however, it makes no difference as both look the same.

Output File

The results can be displayed in an output table or can be output into two formats: 1) Raster grid formats for display in a surface mapping program- *Surfer for Windows* '.dat' format (Golden Software, 2008) or *ArcGIS Spatial Analyst* 'asc' format (ESRI, 2012); or 2) Polygon grids in *ArcGIS* '.shp', *MapInfo* '.mif' or various Ascii formats. However, all but *Surfer for Windows* require that the reference grid be created by *CrimeStat*.[1]

1 *CrimeStat* will output the geographical boundaries of the reference grid and will assign a third-variable

Example 1: Kernel Density Estimate of Baltimore County Street Robberies

An example can illustrate the use of the single kernel density routine. Figure 10.9 shows a *Surfer for Windows* output of 1180 street robberies for 1996 in Baltimore County. The reference grid was generated by *CrimeStat* and had 100 columns and 90 rows. Thus, the routine calculated the distance between each of the 10,800 reference cells and each of the 1180 robbery incident locations, evaluated the kernel function for each measured distance, and summed the results for each reference cell. The normal distribution kernel function was selected for the kernel estimator and an adaptive bandwidth with a minimum sample size of 100 was chosen as the parameters.

There are three views in the figure: 1) a map view showing the location of the incidents; 2) a surface view showing a three-dimensional interpolation of robbery density; and 3) a contour view showing contours of high robbery density. The surface and contour views provide different perspectives. The surface shows the peaks very clearly and the relative density of the peaks. As can be seen, the peak for robberies on the eastern part of the County is much higher than the two peaks in the central and western parts of the County. The contour view can show where these peaks are located; it is difficult to identify location clearly from a three-dimensional surface map. Highways and streets could be overlaid on top of the contour view to identify more precisely where these peaks are located.

Figure 10.10 shows an *ArcGIS* map of robbery density with the robbery incident locations overlaid on top of the density contours. Here, we can see quite clearly that there are three strong concentrations of incidents, one on the west side, one on the northern border between Baltimore City and Baltimore County, and one on the east side which blends with a smaller peak in the southeast corner of the County.

From one perspective, the kernel estimate is a better 'hot spot' identifier than the cluster analysis routines discussed in Chapters 7 and 8. Cluster routines group incidents into clusters and distinguish between incidents which belong to the cluster and those which do not belong. Depending on which mathematical algorithms are used, different clustering routines will return differing allocations of incidents to clusters. The kernel estimate, on the other hand, is a continuous surface; the densities are calculated at *all* locations; thus, the user can visually inspect

---

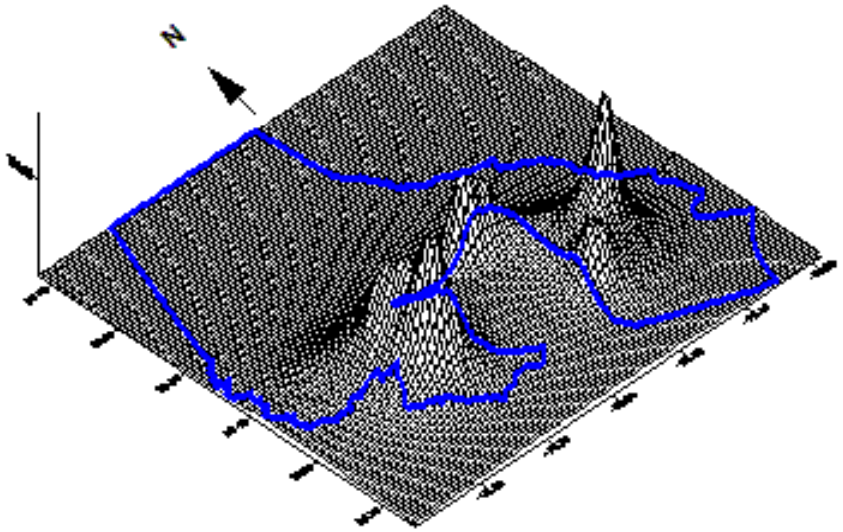
(called *Z*) as the density estimate. *ArcGIS* '.shp' files can be read directly into the program. For *MapInfo*, on the other hand, the output is in MapInfo Interchange Format (a '.mif' and a '.mid' file). The files must be imported to a *MapInfo* '.tab' file. For both output formats, the values of Z can be shown as a thematic map but the ranges must be adjusted to illustrate the high density locations (i.e., the default values in most GIS programs will not display the densities very well). On the other hand, the default interval values for *Surfer for Windows* and *ArcGIS Spatial Analyst* provide a reasonably good visualization.

Figure 10.9:

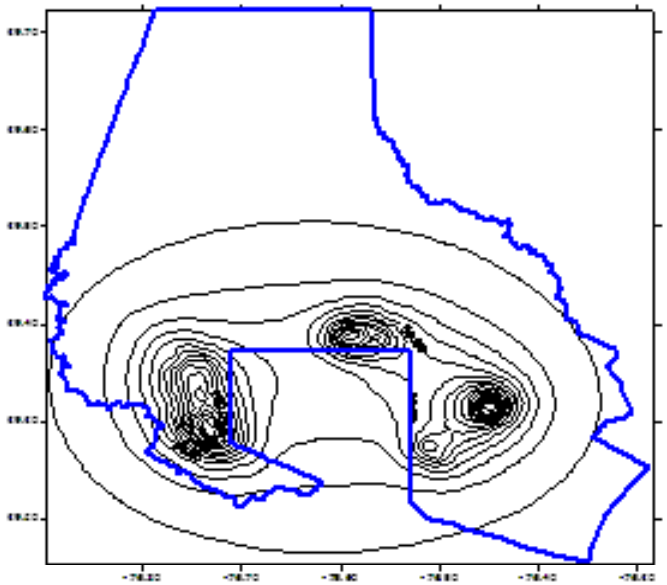
Baltimore County Robberies: 1996-97

Kernel Density Interpolation

Surface View

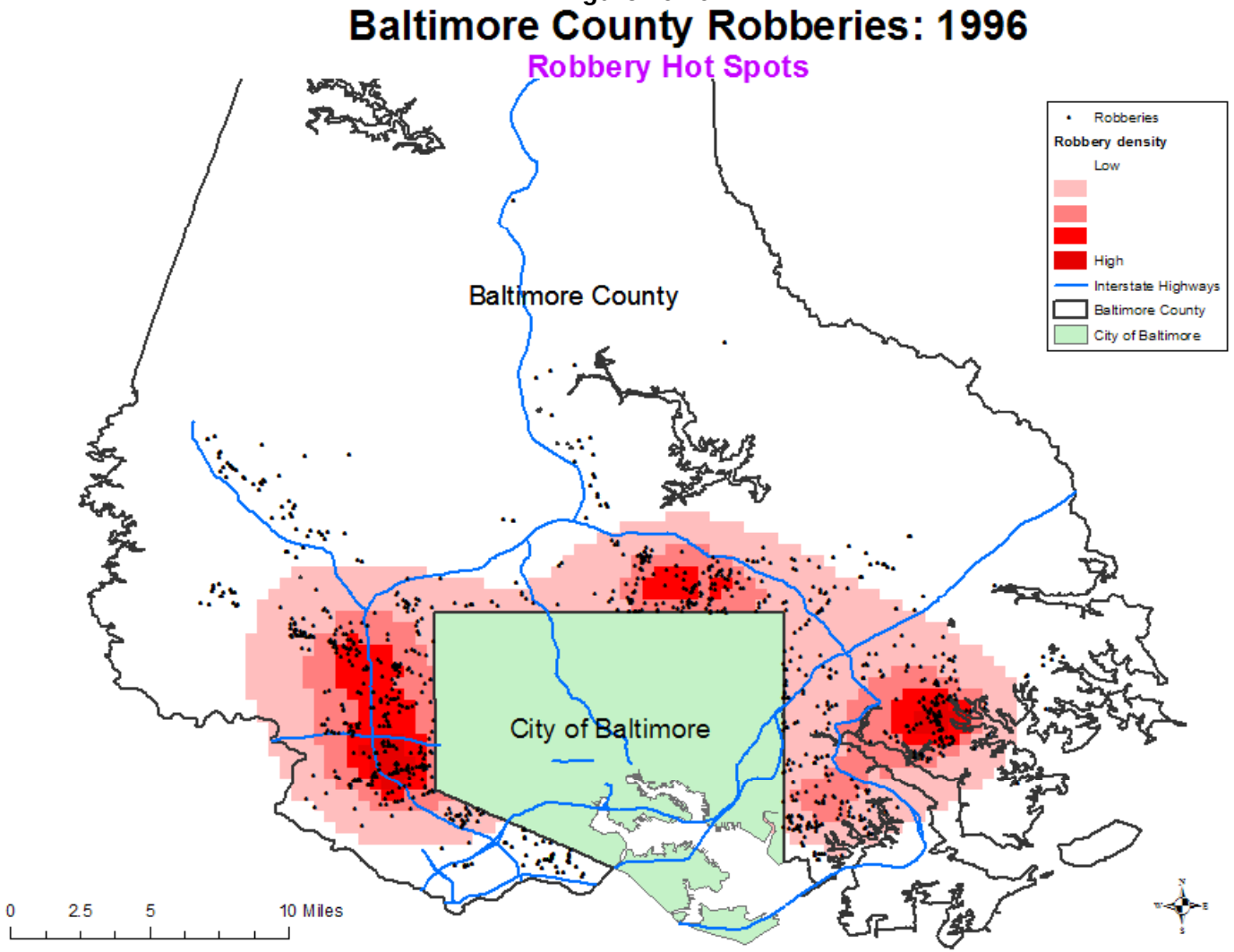


Contour View



Ground Level View

Figure 10.10:


Baltimore County Robberies: 1996
Robbery Hot Spots
Robberies
Robbery density
Low
High
Interstate Highways
Baltimore County
City of Baltimore
Baltimore County
City of Baltimore
0
2.5
5
10 Miles

the variability in density and decide what to call a 'hot spot' without having to define arbitrarily where to cut-off the 'hot spot' zone.

Going back to the *Surfer for Windows* output, Figure 10.11 shows the effects of varying the bandwidth parameters. There are three fixed bandwidth intervals (0.5, 1, and 2 miles respectively) and there are two adaptive bandwidth intervals (a minimum of 25 and 100 points respectively). As can be seen, the fineness of the interpolation is affected by the bandwidth choice. For the three fixed intervals, an interval of 0.5 miles produced a finer mesh interpolation than an interval of 2 miles, which tended to 'oversmooth' the distribution. Perhaps, the intermediate interval of 1 mile gives the best balance between fineness and generality? For the two adaptive intervals, the minimum sample size of 25 gave very specific peak locations whereas the adaptive interval with a minimum sample size of 100 gave a smoother distribution.

Which of these should be used as the *best* choice would depend on how much confidence the analyst has in the results. A key question is whether the 'peaks' are real or merely by-products of small sample sizes. The best choice would be to produce an interpolation that fits the experience of the department and officers who travel an area. Again, experimentation and discussions with beat officers will be necessary to establish which bandwidth choice should be used in future interpolations.

Note in all five of the interpolations, there is some bias at the edges with the City of Baltimore (the three-sided area in the central southern part of the map). Since the primary file only included incidents for the County, the interpolation nevertheless has estimated some likelihood at the edges; these are *edge biases* and need to be ignored or removed with an ASCII editor.[2] Further, the wider the interval chosen, the more bias was produced at the edge.

Dual Kernel Density Interpolation

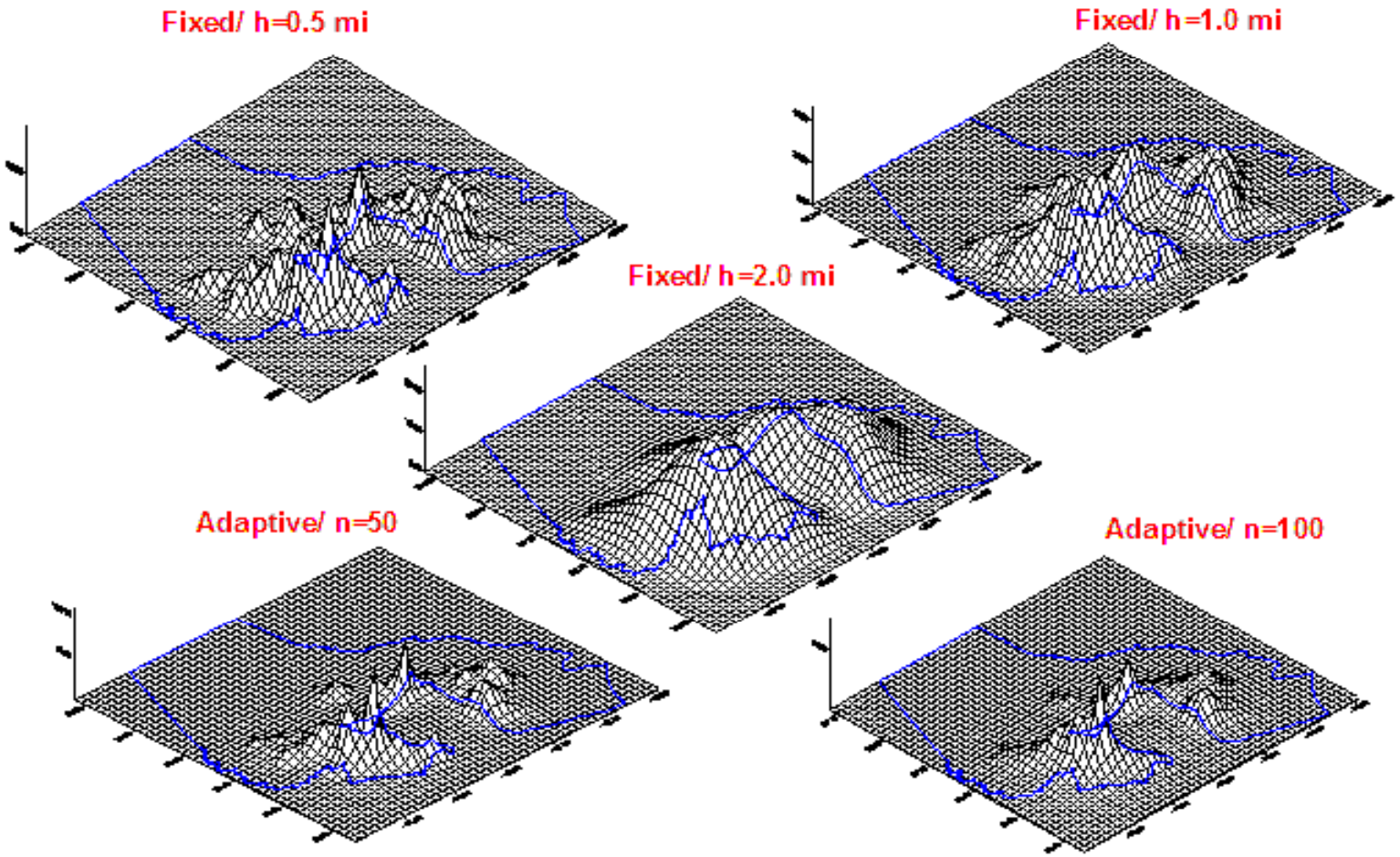
The **dual kernel density** routine in *CrimeStat* is applied to ***two*** distributions. For example, the primary file could be the location of auto thefts while the secondary file could be the centroids of census tracts with the population of the census tract being an intensity variable. The dual routine must be used with *both* a primary file *and* a secondary file. Also, it is

[2] All the *CrimeStat* outputs except for *ArcGIS* 'shp' files are in ASCII. There are usually 'edge effects' and values interpreted outside the actual geographical area. These can be removed with an ASCII editor by substituting '0' for the values at the edges or outside the study region. For 'shp' files, the values at the edges can be edited within the *ArcGIS* program. Another alternative is to 'cut out' the cells that are beyond the study area. Care must be taken, however, to not edit an output file too much otherwise it will bear little relationship to the calculated kernel estimate.

Figure 10.11:

Interpolation of Baltimore County Vehicle Thefts: 1996

Different Smoothing Parameters



necessary to define a reference file, either an existing file or one generated by *CrimeStat* (see Chapter 3). Several parameters need to be defined.

File to be Interpolated

The user must indicate the order of the interpolation. The routine uses the language *first* file and *second* file in making the comparison (e.g., dividing the first file by the second; adding the first file to the second). The user must indicate which is the first file - the Primary or the Secondary. The default is that the Primary file is the first file.

Method of Interpolation

The user must indicate the type of kernel estimator. As with the single kernel density routine, five types of kernel density estimators are used

1. Normal distribution (bell; default)
2. Uniform (flat) distribution
3. Quartic (spherical) distribution
4. Triangular (conical) distribution
5. Negative exponential (peaked) distribution

In our experience, there are advantages to each. The normal distribution produces an estimate over the entire region whereas the other four produce estimates only for the circumscribed bandwidth radius. If the distribution of points is sparse towards the outer parts of the region, then the four circumscribed functions will not produce estimates for those areas, whereas the normal will. Conversely, the normal distribution can cause some edge effects to occur (e.g., spikes at the edge of the reference grid), particularly if there are many points near one of the boundaries of the study area. The four circumscribed functions will produce less of a problem at the edges, although they still can produce some spikes. Within the four circumscribed functions, the uniform and quartic tend to smooth the data more whereas the triangular and negative exponential tend to emphasize 'peaks' and 'valleys'. The differences between these different kernel functions are small, however. The user should probably start with the default normal function and adjust accordingly to how the surface or contour looks.

Choice of Bandwidth

The user must define the bandwidth parameter. There are three types of bandwidths for the single kernel density routine - fixed interval, adaptive interval, or variable interval.

Fixed interval

With a fixed bandwidth, the user must specify the interval to be used and the units of measurement (square miles, square nautical miles, square feet, square kilometers, or square meters). Depending on the type of kernel estimate used, this interval has a slightly different meaning. For the normal kernel function, the bandwidth is the standard deviation of the normal distribution. For the uniform, quartic, triangular, or negative exponential kernels, the bandwidth is the radius of the search area to be interpolated. Since there are two files being compared, the fixed interval is applied both to the first file and the second file.

Adaptive interval

An adaptive bandwidth adjusts the bandwidth interval so that a minimum number of points (sample size) is found. This sample size is applied to both the first file and the second file. It has the advantage of providing constant precision for the kernel estimate over the entire region. Thus, in areas that have a high concentration of points, the bandwidth is narrow while in areas where the concentration of points is sparser, the bandwidth will be wider. This is the default bandwidth choice in *CrimeStat* since consistency in statistical precision is important. The degree of precision is generally dependent on the sample size of the bandwidth interval. The default is a minimum of 100 points. The user can make the estimate finer by choosing a smaller number of points (e.g., 25) or smoother by choosing a larger number (e.g., 200).

Variable interval

With a variable interval, each file (the first and the second) have different intervals. For both, the units of measurements must be specified (square miles, square nautical miles, square feet, square kilometers, or square meters).

There is a good reason why a user might want to use variable intervals. In comparing two kernel estimates, the most common comparison is to divide one by the other. However, if the density estimate for a particular cell for the denominator variable approaches zero, then the ratio will blow up and become a very large number. Visually, this will be seen as spikes in the distribution, the result, usually, of too few cases. In this case, the user might decide to smooth the denominator more than numerator to reduce these spikes. For example, the interval for the first file (the numerator) could be 0.5 miles whereas the interval for the second file (the denominator) could be 1 mile. Experimentation will be necessary to see whether this is warranted. But, in our experience, excessively large densities happen when either there are too few cases or there is an irregular boundary to the region with a number of incidents grouped at one of the edges.

Use kernel bandwidths that produce stable estimates

Note that with a dual kernel calculation, particularly the ratio of one variable to another, it is important not to choose too small a bandwidth. This could have the effect of creating spikes at the edges of the study area or in low population density areas. For example, in low population density areas, there will probably be fewer events than in more built-up area. For the denominator of a ratio estimate, an extremely low value could cause the ratio to be exaggerated (a 'spike') relative to neighboring grid cells. Using a larger bandwidth will produce a more stable average.

Output Unit

The user must indicate the measurement units for the density estimate in points per square miles, square nautical miles, square feet, square kilometers, or square meters.

Intensity or Weighting Variable

If an intensity or weighting variable is to be used (and has been defined on the Primary or Secondary file page), the appropriate box must be checked. Be careful about using both intensity and weighting variables to avoid 'double weighting'.

Density Calculation

The user must indicate the type of density output. There are six types of density calculations that can be conducted with the dual kernel density routine. The calculations are applied to each reference cell:

1. There is the *ratio of densities.* This is the first file divided by the second file. This is the default choice. For example, if the first file is the location of auto thefts incidents and the second file is the location of census tract centroids with the population assigned as an intensity variable, then ratio of densities would divide the kernel estimate for auto thefts by the kernel estimate for population and would be an estimate of auto thefts risk.

2. There is also the *log ratio of densities*. This is the natural logarithm of the density ratio, that is

$$\text{Log ratio of densities} = \text{Ln}\left[g(x_j) / g(y_j) \right] \qquad (10.10)$$

where $g(x_j)$ is the density estimate for the first file and $g(y_j)$ is the density estimate for the second file. For a variable that has a spatially skewed distribution such that most reference cells have very low density estimates, but a few have very high density estimates, converting the ratio into a log function will tend to mute the spikes that occur. This measure has been used in studies of risk (Kelsall & Diggle, 1995b).

3. There is the *absolute difference in densities*. This is the first file minus the second file. This can be a useful output for examining differential effects. For example, by using the centroids of census block groups (see example 2 below) with the population of the census block group assigned as an intensity variable, the difference in population between two different census years can be estimated. Since the spatial arrangements of the block groups changes slightly from one census to the next (the U. S. Census Bureau suggests that census units be drawn so that there are approximately equal populations in each unit), estimating the difference in kernel densities between two census can show where the changes have occurred irrespective of the particular census units.

4. There is the *relative difference in densities*. Like the relative density in the single-kernel routine (discussed above), the relative difference in densities first standardizes the densities of each file by dividing by the grid cell area in familiar units (square miles or square kilometers) and then subtracts the secondary file relative density from the primary file relative density. This can be useful in calculating changes between two time periods, for example in calculating a change in relative density between two censuses or a change in the crime density between two time periods.

5. There is the *sum of the densities*. This is the density estimate for the first file plus the density estimate for the second file. A possible use of the sum operation is to combine two different density surfaces, for example the density of robberies plus the density of assaults;

6. Finally, there is the *relative sum of densities*. The relative sum of densities first standardizes the densities of each file by dividing by the grid cell area in familiar units (square miles or square kilometers) and then adds the secondary file relative density to the primary file relative density. This can be useful for identifying the total effects of two distributions. For example, the total impact of robberies and burglaries on an area can be estimated by taking the relative density of robberies and adding it to the relative density of burglaries. The result is the combined

relative density of robberies and burglaries per unit area (e.g., robberies and burglaries per square mile).

Output File

Finally, the user must specify the file formats for the output. The results can be output in three forms. First, the results are displayed in an output table. Second, the results can be output into two raster grid formats for display in a surface mapping program: *Surfer for Windows* format as a '.dat' file (Golden Software, 2008) and *ArcGIS Spatial Analyst* format as an 'asc' file (ESRI, 2012). Third, the results can be output as polygon grids into *ArcGIS* '.shp', *MapInfo* '.mif' and various Ascii formats (see footnote 1). All but *Surfer for Windows* require that the reference grid be created by *CrimeStat*.

Example 2: Kernel Density Estimates of Vehicle Thefts Relative to Population

As an example of the use of the dual kernel density routine, the dual routine is applied in both the City of Baltimore and the County of Baltimore to 14,853 motor vehicle theft locations for 1996 relative to the 1990 population of census block groups. Again, a reference grid of 100 columns by 108 rows was generated by *CrimeStat*.

Figure 10.12 shows the resulting density estimate as a *Surfer for Windows* output; again, there is a map view, a surface view, and a contour view. The normal kernel function was used and an adaptive bandwidth of 100 points was selected. As seen, there is a very high concentration of auto theft incidents within the central part of the metropolitan area. The contour view suggests five or six peak areas that are close to each other.

Much of this concentration, however, is produced by high population density in the metropolitan center. Figure 10.13, for example, shows the kernel estimate for 1349 census block groups for both the City of Baltimore and the County of Baltimore with the 1990 population assigned as the intensity variable. Again, the normal kernel function was used with an adaptive bandwidth of 100 points being selected. The map shows three views: 1) a surface view; 2) a contour view; and 3) a ground level view looking directly north. The distribution of population is, of course, also highly concentrated in the metropolitan center with two peaks, quite close to each other with several smaller peaks.

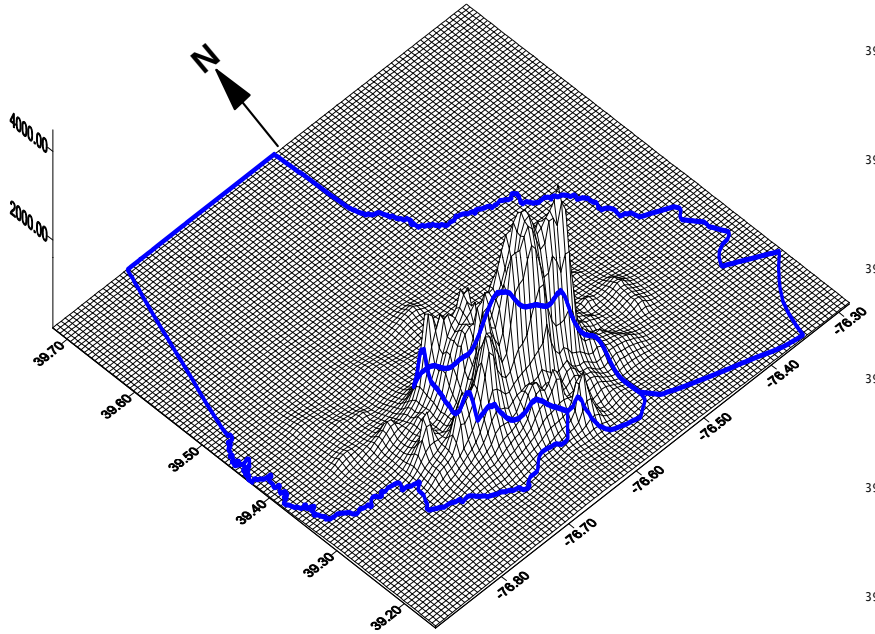
When these two kernel estimates are compared using the dual kernel density routine, a more complicated picture emerges (Figure 10.14). This routine has conducted three operations: 1) it calculated the distance between each of the 10,800 reference cells and the 14,853 auto theft locations, evaluated the kernel function for each measured distance, and summed the results for each reference cell; 2) it calculated the distance between each of the 10,800 reference cells and

Figure 10.12:

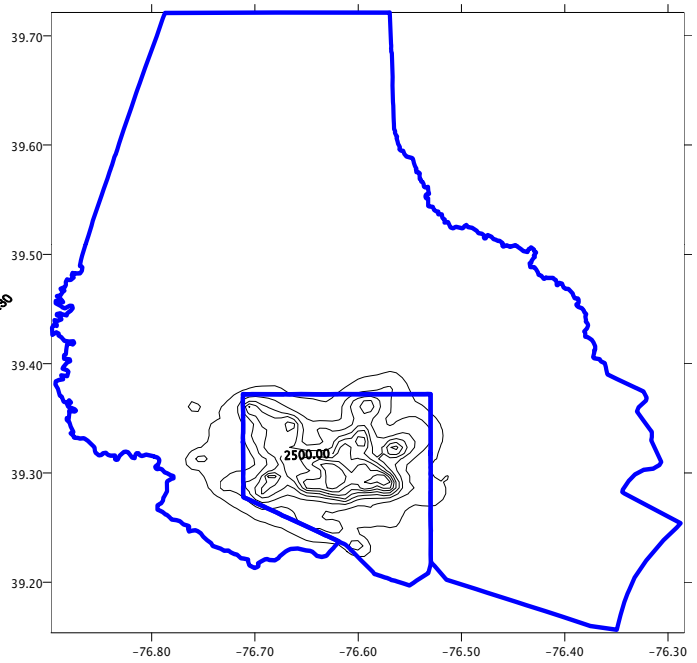
Baltimore Metropolitan Vehicle Thefts: 1996

Three Views of Kernel Density Interpolation

Surface View



Contour View



Ground Level View

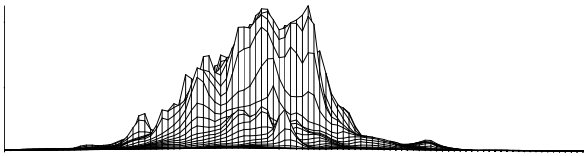
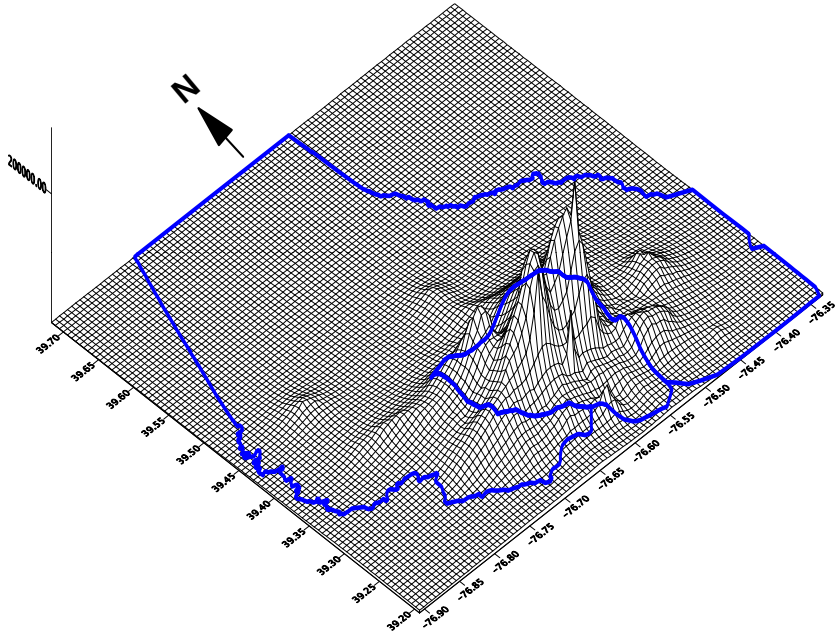


Figure 10.13:

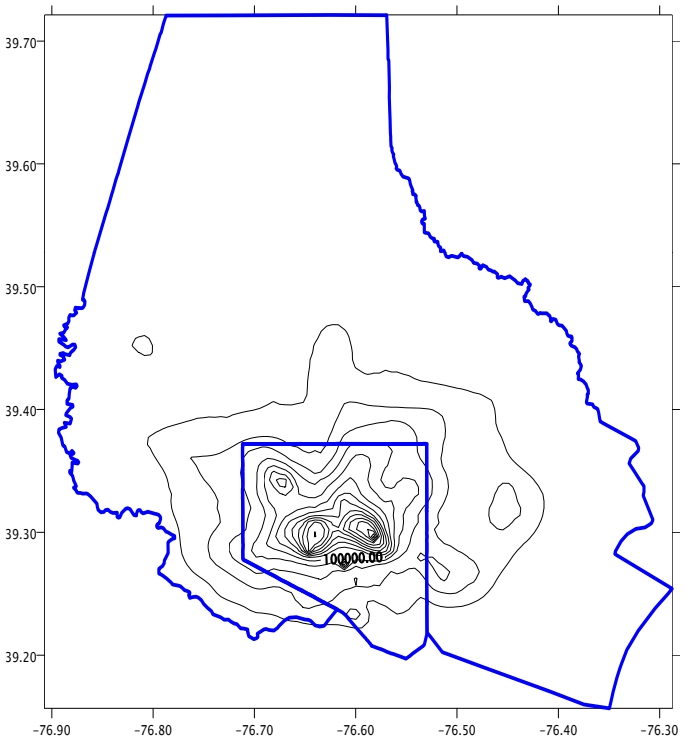
Baltimore Metropolitan Population: 1990

Three Views of Kernel Density Interpolation

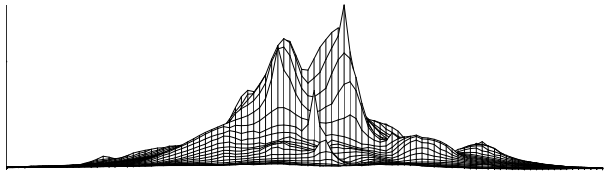
Surface View



Contour View



Ground Level View



the 1349 census block groups with population as an intensity variable, evaluated the kernel function for each intensity-weighted distance, and summed the results for each reference cell; and 3) divided the kernel density estimate for auto thefts by the kernel density estimate for population for each reference cell location.

While the concentration of motor vehicle thefts relative to population ('motor vehicle theft risk") is still high in the metropolitan center, there are bands of high risk that spread outward, particularly along major arterials. There are now many 'hot spot' areas that have a high distribution of motor vehicle thefts relative to the residential population. We could, of course, refine this analysis further by taking, for example, employment as a baseline variable rather than population; employment is a better indicator for the daytime population distribution whereas the residential population is a better indicator for nighttime population distribution (Levine, Kim, & Nitz, 1995a; 1995b).

Example 3: Kernel Density Estimates and Risk-adjusted Clustering of Robberies Relative to Population

The final example shows how the dual kernel interpolation compares with the risk-adjusted nearest neighbor clustering, discussed in chapter 6. Figure 10.15 shows 15 first-order and two second-order risk-adjusted clusters overlaid on the dual kernel estimate of 1996 robberies relative to 1990 population.[3] As seen, there is a correspondence between the identified risk-adjusted clusters and the dual kernel interpolation of the ratio of robberies to population. For a broad regional perspective, the interpolation produces an adequate model of where there is a high robbery risk. At the neighborhood level, however, the risk-adjusted clusters are more specific and would be preferable for use by police in identifying high-risk locations.

The advantage of a dual kernel density interpolation routine is that two variables can be related together. By interpolating one variable to a reference grid and then interpolating a second variable to the same reference grid, the two variables have been interpolated to the same geographical units. The two interpolations can then be related, by dividing, subtracting, or summing. As has been mentioned throughout this manual, one of the problems with techniques that depend on the concentration of incidents is that they ignore the underlying population at-risk. With the dual routine, however, we can start to examine the risk and not just the concentration.

---

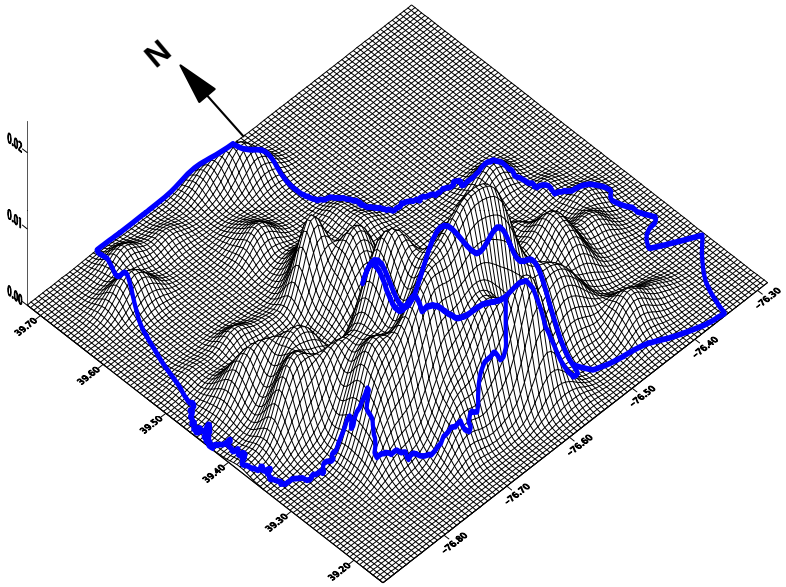
[3] The risk-adjusted hierarchical clustering (Rnnh) method defined the largest search radius but a minimum of 25 points being required to be clustered. The kernel estimate for both the Rnnh and the dual-kernel routines used the normal distribution function with an adaptive bandwidth of 25 points.

Figure 10.14:

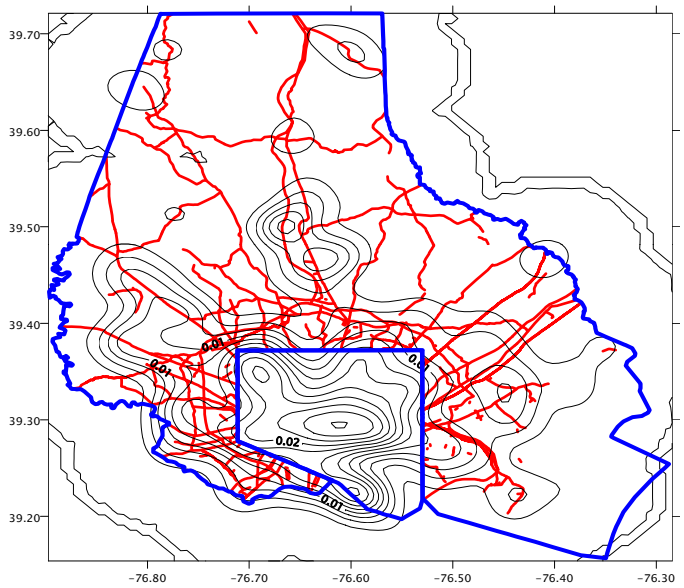
Baltimore Metropolitan Vehicle Theft Risk: 1990-1996

Ratio of Interpolation of 1996 Auto Thefts to 1990 Population

Surface View



Contour View



Ground Level View

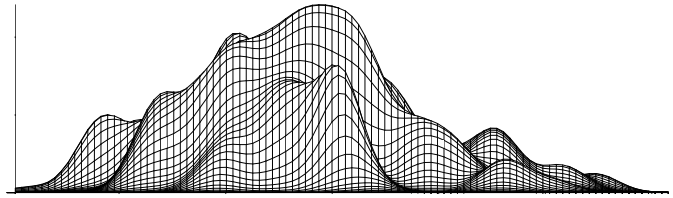
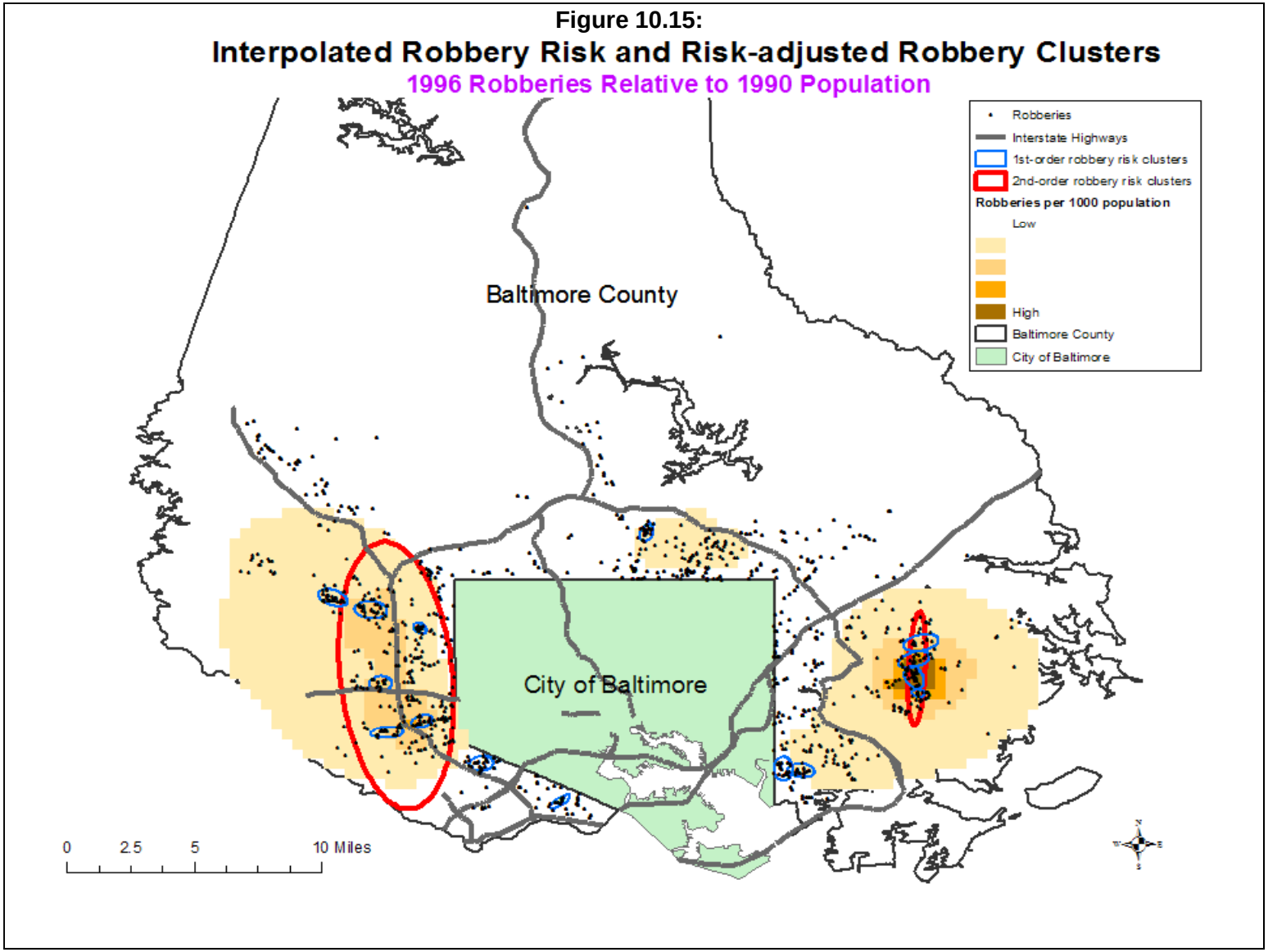


Figure 10.15:

Interpolated Robbery Risk and Risk-adjusted Robbery Clusters

1996 Robberies Relative to 1990 Population



Visually Presenting Kernel Estimates

Whether the single- or dual-kernel estimate is used, the result is a grid interpretation of the data. By scaling these values in a GIS program, a visualization of the data is obtained. Areas with higher densities can be shown in darker tones and those with lower densities can be shown in lighter tones; some people do the opposite with the high density areas being lighter.

To make the visualization even more realistic, one could use a GIS program to cut out those grid cells that are outside the study area or are on water bodies. Before doing this, however, be sure to re-scale the estimated "Z" values so that they will sum to the total of the original grid. For example, if the original sample size was 1000, then the grid cells will sum to 1000 if the absolute density option is chosen. If, say, 20% of these cells are then removed to improve the visualization, then the grid cell Z values have to be re-scaled so that their sum will continue to be 1000. A simple way to do this is to, first, add up the Z values for the remaining cells and, second, multiply each grid cell Z by the ratio of the original sum to the reduced sum.

Advantages and Limitations of Kernel Density Interpolation

There are advantages and limitations to the kernel density interpolation method for hot spot analysis.

Advantages of Kernel Density Interpolation

The main advantage of kernel density interpolation is its ability to visualize a broad, regional view of events. Whereas each of the hot spot analysis techniques discussed in Chapters 7 and 8 (and 9 for the Zonal Nnh) drew boundaries around the hot spots, kernel density interpolation provides density estimates through the study area. One can see all the high density and low density areas simultaneously. For example, this can provide a police department with an overview of the high crime areas and can form the basis of patrol deployment. Essentially, for a city-wide or region-wide view, there is no better technique (Chainey, Thompson & Uhlig, 2008).

Limitations of Kernel Density Interpolation

At the same time, there are limitations to the approach for hot spot analysis. There are three statistical problems. First, the method does depend on overgeneralizing data. By interpolating *N* data points to *M* grid cells where *M* is almost always much greater than *N*, means that the data are being shared across many grid cells. This can lead to overgeneralization of results. For example, 10,000 cases seems like a large data set (which it normally is), but when it is generalized to 10,000 grids (100 columns x 100 rows), this leaves an average of 1 data point

per grid cell. It is well known that sampling error is very high with small samples and almost infinitely high with a sample size of 1. Yet the method pools the data so that every grid cell is represented by all (for the normal distribution kernel) or most (for the other kernels) data points. This leads to additional spatial autocorrelation among the estimates since each grid cell shares the same data points with adjacent grid cells. The practical effect of this is that a hot spot can appear to be larger than it truly is. Too many users are taking kernel density interpolations as evidence of hot spots even with very small samples. These hot spots will turn out to be nothing more than random variation. Again, if used carefully, the method can provide an overview of crime density in a study area. But, one has to be very, very careful in using the method to define specific hot spots.

Second, like other hot spot analysis methods, kernel density interpolation is effected by the choice of kernel used and the selected bandwidth. The normal distribution kernel, for example, will smooth the data and eliminate small nodules ('peaks and valleys') whereas the quartic and exponential kernels will emphasis the small nodules. Whether the more granular variation in density estimates is valid or not depends on the sample size. With a small sample size, small hot spots may be nothing more than random variation and may not be real. Unless the sample size is very large (meaning 10,000 or more cases), we recommend using the normal distribution kernel to avoid finding *false hot spots*. In addition, the selected bandwidth determines the smoothness of the visualization. Again, if the sample size is large, a smaller bandwidth is appropriate whereas a larger bandwidth is more desirable for smaller samples. One has to consider the *precision* of the estimates, which is a function of the sample size (i.e., larger is better).

Third, because the technique smoothes data, it is often inappropriate for small area analysis. It will lead to generalization of data points into adjacent areas from where the events occur and can lead to false conclusions (Levine, 2008). For example, motor vehicle crashes typically occur on freeways, highways, major arterial roads, and minor arterial roads. Few occur on residential (neighborhood) streets, typically less than 15%. Levine (2009) found that only 11% of motor vehicle crashes in the Houston metropolitan area occurred on local roads even though these roads accounted for 61% of the total road mileage in the region. The likelihood of a crash occurring on any particular local road is extremely small. However, since nearly one half of the crashes occurred at intersections, the method would generalize crashes at two intersecting arterial roads into the adjacent neighborhood streets when, in reality, very few crashes will occur on those streets.

Similarly, Levine (2008) showed how vehicle thefts that were concentrated in parking lots in a commercial area of Houston were generalized by kernel density interpolation into the local residential neighborhoods. In other words, the method produces spatial distortion

especially for small area (large-scale), neighborhood-level analysis. Very often, hot spots are very limited spatially, sometimes into an area less than half a block wide.

Taking another example, in a 1986 study of dangerous bus stops in Los Angeles, the most serious one was identified on one corner of a Hollywood (CA) intersection (Levine, Wachs & Shirazi, 1986). The hot spot involved a drug trade that occurred at a food stand on the corner and was supplied by drug dealers who used a bar near the intersection for cover. The crimes occurred only on that corner of the intersection; the other three corners had no crime events.

In other words, the amount of smoothing involved in kernel density interpolation will distort spatial relationships for very small hot spots and will make it appear as if there is a risk in nearby blocks when there might not be such a risk. The use of one of the cluster routines discussed in chapters 7 and 8 would be more appropriate for small area analysis.

Conclusion

Kernel density estimation is one of the most utilized spatial statistical techniques. There is currently research on the use of this technique in both the statistical theory and in developing applications. For crime analysis, the technique represents a powerful way of conducting both regional hot spot analysis as well as being able to link the hot spots to an underlying population-at-risk. It can be used both for police deployment by targeting areas of high concentration of incidents as well as for prevention by targeting areas with high risk. It can also be used as a research tool for analyzing two or more distributions. Caution has to be used in adapting the method for small area (large scale) neighborhood type of analysis. Other techniques are more appropriate for that level.

References

Anselin, L. (1992). *SpaceStat: A Program for the Statistical Analysis of Spatial Data*. Santa Barbara, CA: National Center for Geographic Information and Analysis, University of California.

Bailey, T. C. & Gatrell, A. C. (1995). *Interactive Spatial Data Analysis*. Longman Scientific & Technical: Burnt Mill, Essex, England.

Bowman, A. W. & Azzalini, A. (1997). *Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations*. Oxford Science Publications, Oxford University Press: Oxford, England.

Burt, J. E. & Barber, G. M. (1996). *Elementary Statistics for Geographers* (second edition). The Guilford Press: New York.

Chainey, S., Thompson, L., & Uhlig, S. (2008). The utility of hotspot mapping for predicting spatial patterns of crime. *Security Journal*, **21**, 4-28.

Cleveland, W. S., Grosse, E., & Shyu, W. M. (1993). Local regression models. In John M. Chambers & Trevor J. Hastie, *Statistical Models in S*. Chapman & Hall: London.

Cliff, A. D. & Haggett, P. (1988). *Atlas of Disease Distributions*. Blackwell Reference: Oxford.

ESRI (2012). *ArcGIS 10.0*. Environmental Systems Research Institute: Redlands, CA. http://www.esri.com/software/arcgis/index.html.

Farewell, D.l (1999). Specifying the bandwidth function for the kernel density estimator. http://www.iph.cam.ac.uk/bugs/documentation/coda03/node44.html.

Golden Software. 2008. *Surfer® for Windows (Ver. 10)*. Golden Software, Inc.: Golden, CO.

Härdle, W. (1991). *Smoothing Techniques with Implementation in S*. Springer-Verlag: New York.

Kafadar, K. (1996). Smoothing geographical data, particularly rates of disease. *Statistics in Medicine* 15(23), 2539-2560.

References (continued)

Kelsall, J. E. & Diggle, P.J. (1995a). Kernel estimation of relative risk, *Bernoulli*, 1, 3-16.

Kelsall, J. E. & Diggle, P.J. (1995b). Non-parametric estimation of spatial variation in relative risk. *Statistical Medicine*, 14, 2335-2342.

Levine, N. (2009). "A motor vehicle safety planning support system: The Houston experience". In S. Geertman and J. Stillwell, *Planning Support Systems: Best Practice and New Methods*. Springer. 93-111.

Levine, N. (2008). "The 'hottest' part of a crime hotspot: Comments on "The utility of hotspot mapping for predicting spatial patterns of crime" by Spencer Chainey, Lisa Tompson, and Sebastian Uhlig". *Security Journal*, 21, 295-302.

Levine, N., Kim, K. E. & Nitz, L. H. (1995a). Spatial analysis of Honolulu motor vehicle crashes: I. Spatial patterns. *Accident Analysis & Prevention*, 27(5), 663-674.

Levine, N., Kim, K. E.& Nitz, L. H. (1995b). Spatial analysis of Honolulu motor vehicle crashes: II. Generators of crashes. *Accident Analysis & Prevention*, 27(5), 675-685.

Levine, N., Wachs, M. & Shirazi, E. (1986). Crime at Bus Stops: A Study of Environmental Factors. *Journal of Architectural and Planning Research*. 3 (4), 339-361.

Parzen, E. (1962). On the estimation of a probability density and mode. *Annuals of Mathematical Statistics*, 33, 1065-1076.

Rosenblatt, M. (1956). Remarks on some non-parametric estimates of a density function. *Annuals of Mathematical Statistics*, 27, 832-837.

Scott, D. W. (1992). *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley & Sons: New York.

Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman & Hall: London.

Talbot T. O., Kulldorff, M., Forand S. P., & Haley V. B. (2000). Evaluation of spatial filters to create smoothed maps of health data. *Statistics in Medicine*, 19, 2399-2408.

References (continued)

Venables, W.N. & Ripley, B. D. (1997). *Modern Applied Statistics with S-Plus (second edition)*. Springer-Verlag: New York.

Whittle, P. (1958). On the smoothing of probability density functions. *Journal of the Royal Statistical Society, Series B*, 55, 549-557.

Endnotes

i. There are differences in opinion about how wide a particular fixed bandwidth should be determined. The smoothing is done for a distribution of values, Z. If there are only unique points (and, hence, there is no Z value at a point), the distances between points can be substituted for Z. Thus, MeanD is the mean distance, sd(D) is the standard deviation of distance, and iqr(D) is the inter-quartile range of distances between points. These would be substituted for MeanZ, sd(Z), and iqr(Z) respectively

Silverman (1986; 45-47; Härdle, 1991; Farewell, 1999) proposed a bandwidth, h, of:

$$h = 1.06 * \min\{sd(Z)\frac{iqr(Z)}{1.34}\}N^{-\frac{1}{5}}$$

where *min* is the minimum of the next two terms, *sd(Z)* is the standard deviation of the variable, Z, being interpolated, *iqr(Z)* is the inter-quartile range of Z, and N is the sample size.

Bowman and Azzalini (1997; 31) defined a slightly different optimal bandwidth for a normal kernel:

$$h = \{\frac{4}{3N}\}^{\frac{1}{5}} * SD(Z)$$

To avoid being influenced by outlier, they suggested using the median absolute deviation estimator for sd(Z):

$$MAD(Z) = median\{\frac{Z(i)-MedianZ}{0.6745}\}$$

Scott (1992) suggested an upper bound on the normal kernel of

$$h = 1.144 * sd(Z) * N^{-1/5}$$

Bailey and Gatrell (1995, 85-87) offered a rough choice for the bandwidth of

$$h = 0.68N^{-\frac{1}{5}}$$

but suggested that the user could experiment with different bandwidths to explore the surface.

On the other hand, the concept of an adaptive bandwidth is based more on sampling theory (Bailey & Gatrell, 1995). By increasing the bandwidth until a fixed number of points are counted ensures that the level of precision is constant throughout the region. As with all sampling, the standard error of the estimate is a function of the sample size; a larger sample leads to smaller error. In general, if there was independent sampling, the 95% confidence interval of a bandwidth for a normal kernel could be approximated by:

$$95\%\ confidence\ interval = \bar{Z} \pm 1.96\frac{0.5}{N(h)^{\frac{1}{2}}}sd(Z)$$

where N(h) is the adaptive sample size (the number of points counted within the bandwidth for the adaptive kernel). This assumes that a point has an equal likelihood of falling within the bandwidth of one cell compared to an adjacent cell (i.e., it sits on the boundary of the bandwidth circle). The adaptive bandwidth criterion requires that the bandwidth be increased until it captures the specified number of points.

Endnotes (continued)

On average, if there are N points in a region of area, A, and if the adaptive sample size is N(p), then the average area required to capture N(p) points is:

$$A(p) = \frac{AN(p)}{N}$$

and the average bandwidth, Mean(h), is:

$$Mean(h) = \sqrt{\frac{A(p)}{\pi}} = \sqrt{\frac{AN(p)}{N\pi}}$$

Each of these provide different criteria for the bandwidth size with the adaptive being the most conservative. For example, for a standardized distribution with 1000 data points, a standardized mean of Z of 0 and a standardized standard deviation of 1, the Silverman criteria would produce a bandwidth of 0.2663; the Bowman and Azzalini criteria would produce a bandwidth of 0.2661; the Scott criteria would produce a bandwidth of 0.2874 and the Bailey and Gatrell criteria would produce a bandwidth of 0.1708. For the adaptive interval, if the required adaptive sample size is 25, then the average bandwidth would be approximately 0.3162 (this assumes that the area is a circle with a radius of 2 standardized standard deviations).

Attachments

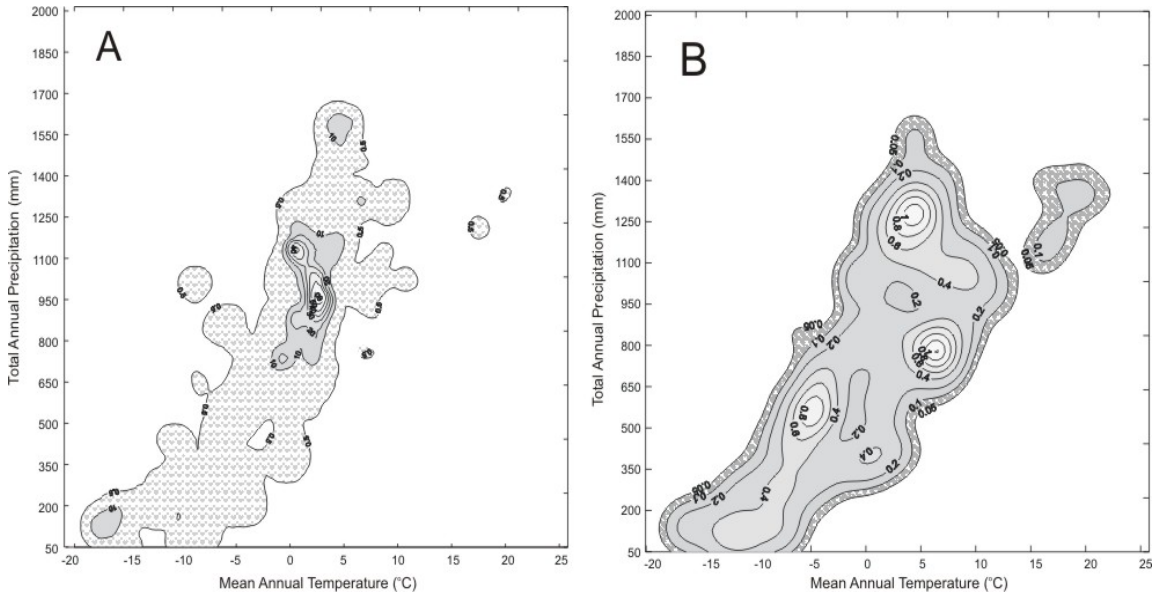
Kernel Density Interpolation to Estimate Sampling Bias in the Climatic Response of *Sphagnum* Spores in North America

Mike Sawada
Laboratory for Applied Geomatics and GIS Science
University of Ottawa, Department of Geography, Canada

Sphagnum moss, the dominant species of bogs, thrives under certain ranges of temperature and precipitation. *Sphagnum* releases spores for reproduction and these are transported, often long distances, by wind and water. Thus, the presence of a spore in the fossil record may not indicate nearby *Sphagnum* plants. However, spores should be most numerous near *Sphagnum* plants. Over time, these spores and pollen from other plants accumulate in lake and bog sediments and leave a fossil record of vegetation history.

We wanted to use the amount of fossil *Sphagnum* spores in different parts of North America to infer past climates. To do so, we had to first show that *Sphagnum* spores are most abundant in climates where *Sphagnum* plants thrive and secondly, that this center of abundance is not biased sampling because of under sampling in parts of climate space. First, we developed a *Sphagnum* spore response surface showing the relative abundance of spores along the axes of temperature and precipitation (Fig. A).

CrimeStat was used in the second stage to develop a kernel density surface using a quartic kernel for 3007 sample sites within climate space (Fig. B). These were smoothed and visualized in *Surfer*. The surface showed that the intensity of points is higher in regions surrounding the response maximum. This gave us confidence that the *Sphagnum* response was real since other parts of climate space are well sampled but unlikely to produce high spore proportions. This fact allowed climate inferences to be made within the fossil record for past time periods using the amount of *Sphagnum* spores present.



Figures modified from Gajewski, Viau, Sawada et al. 2001. Global Biogeochemical Cycles,

Describing Crime Spatial Patterns By Time of Day in Belo Horizonte

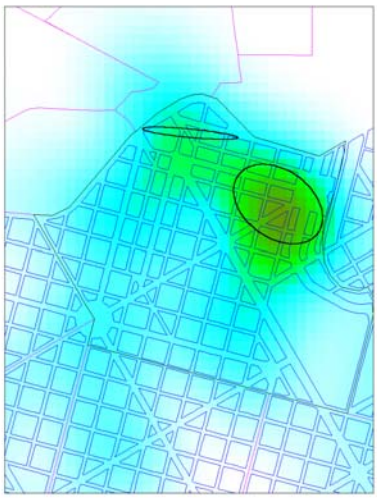
Renato Assunção, Cláudio Beato, Bráulio Silva
CRISP, Universidade Federal de Minas Gerais, Brazil

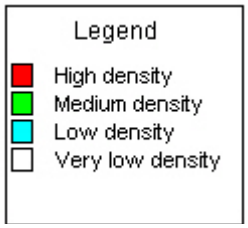
We used the kernel density estimate to visualize time trends for crime occurrences on a typical weekday. We found markedly different spatial distributions depending on the time, with the amount of crime varying and the hot spots, identified by the ellipses, appearing in different places.

The analysis used 1114 weekday robberies from 1995 to 2000 in downtown Belo Horizonte. Breaking the data into hours, we used the normal kernel, a fixed bandwidth of 450 meters and outputted densities option (points per square unit of area). Note that the latter option could be useful if one is interested only in the hot spot locations, and not in the distribution during the day. To make the ellipses, we used the nearest neighbor hierarchical spatial clustering technique with a minimum of 35 incidents. We output the results to *MapInfo*, keeping the same scale for all maps. Four of them are shown below.

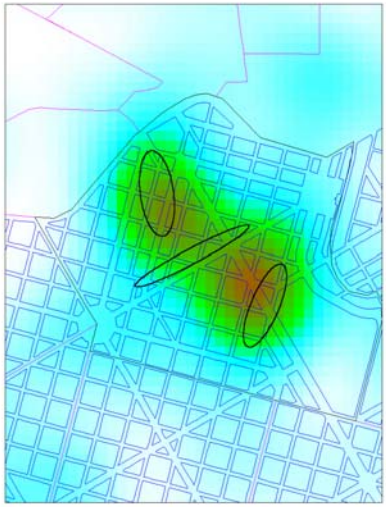
9:00 AM

1:00 PM

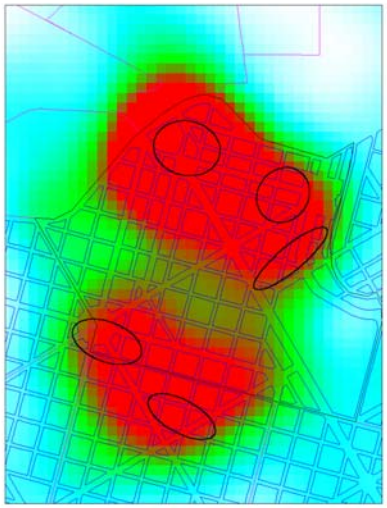




7:00 PM



11:00 PM



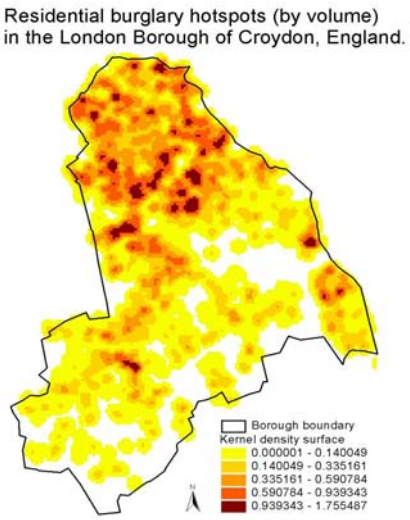
Using Kernel Density Smoothing and Linking to *ArcGIS*: Examples from London, England

Spencer Chainey
Jill Dando Institute of Crime Science
University College
London, England

CrimeStat offers an effective method for creating kernel density surfaces. The example below uses residential burglary incidents in the London Borough of Croydon, England for the period June 1999 – May 2000 (N=3104). The single kernel routine was used to produce a kernel density surface representing the distribution of residential burglary.

The kernel function used was the quartic, which is favoured by most crime mappers as it applies added weight to crimes closer to the centre of the bandwidth. Rather than choosing an arbitary interval it is useful to use the mean nearest neighbour distance for different orders of K, which can be calculated by *CrimeStat* as part of a nearest neighbour analysis. For the Croydon data, an interval of 269 metres was chosen, which relates to a mean nearest neighbour distance at a K-order of 13. The output units were densities in square kilometres and was output to *ArcGIS.*

Kernel density estimation is a particularly useful method as it helps to precisely identify the location, spatial extent and intensity of crime hotspots. It is also visually attractive, so helping to invoke further enquiry and the reasoning behind why crime and disorder is concentrated. The density surface that is created can reflect the distribution of incidents against the natural geography of the area of interest, including representing the natural boundaries, such as reservoirs and lakes, or an alignment that follows a particular street in which there is a high concentration of offending. The method is also less subjective if clear guidelines are followed for the setting of parameters.

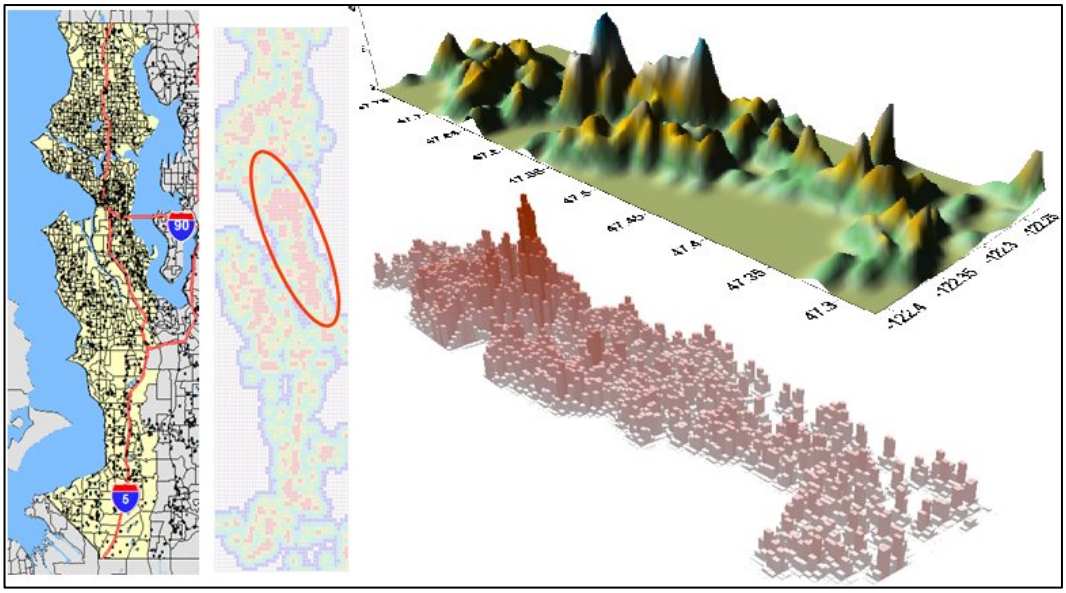


Infant Death Rate and Low Birth Weight in the I-5 Corridor of Seattle and King County

Richard Hoskins
Washington State Department of Health
Olympia, Washington

Although the infant death rate (< 1 year old) has been steadily declining in Washington, the incidence of low birth weight (< 2500 gms) is increasing. This is a significant public health problem, resulting in suffering and high medical cost. If we know where the rates are high at a neighborhood level we can develop more efficient and effective programs. The goal is to determine regions where rates are clustered and to characterize those regions with respect to SES variables from the US Census.

Birth and infant death data were geocoded to the street level. In order to detect clusters of high infant death ***and*** low birth weight, several *CrimeStat* tools were used. We find that using several tools at once helps detect regions where something untoward is going on and also helps develops guesses about where other problems might be expected develop.



I-5 corridor in King County | **Kernel density interpolation** | **Top: 3-D map: empirical Bayes rate Bottom: Prism map: SMR**

The result of a kernel density interpolation using a normal estimator is shown above along with an empirical Bayes rate and standardized mortality ratio (SMR) calculated in SAS and mapped in Maptitude (www.caliper.com). Starting with over 2,500 infant deaths, about 25,000 low weight births (out of over 500,000 live births) occurred in the Seattle I-5 corridor region in King County from 1989-2002. The kernel density method was used to detect high rate regions. A clearly articulated region and ridge appears on the grid of the kernel density map and the 3D and prism maps.

The Risk of Violent Incidents Relative to Population Density in Cologne Using the Dual Kernel Density Routine

Dietrich Oberwittler and Marc Wiesenhütter
Max Planck Institute for Foreign and International Criminal Law
Freiburg, Germany

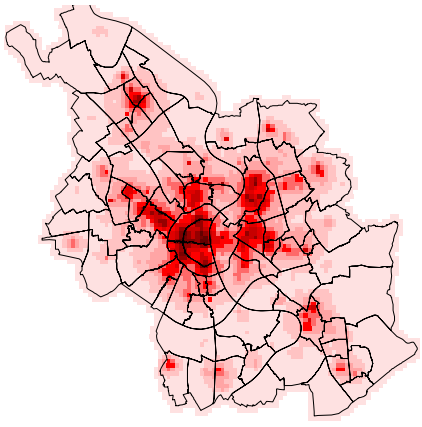
When estimating the density of street crimes within a metropolitan area by interpolating crime incidents, the result is usually a very high concentration in the city center. However, there is also a very high concentration of people either living or pursuing their daily routine activities in these areas. The question emerges how likely is a criminal event when taking into account the number of people spending their time in these areas. The *CrimeStat* dual kernel density routine is able to estimate a ratio density surface of crime relative to the 'population at risk'.

In this example, data on ‘calls to the police’ for assault and battery from April 1999 to March 2000 (N=6363 calls) and population from Cologne were used. Exact information on the number of people spending their time in the city does not exist. Therefore, 1997 counts of passengers entering and leaving the public transport system at each of 550 stations and bus stops in the city was used as a proxy variable. The number of persons at each station or bus stop was assigned to adjacent census tracts and added to the resident population resulting in a crude measure of the 'population at risk'.

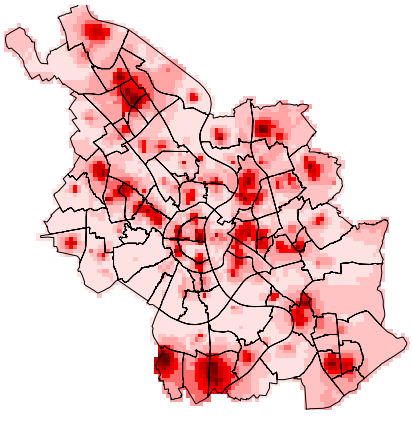
In the dual kernel routine, the density estimate of crime incidents is compared to the density estimate of the population at risk, defined by the centroids of census tracts with the number of persons as an intensity variable. We chose the normal method of interpolation and adaptive intervals with a minimum of five points. The adaptive bandwidth adjusts for the fact that there are fewer incidents and census tracts at the edges of the city, resulting in a relatively smoother density surface for the ratio. The results were output to *ArcView*.

The effect of adjusting the crime distribution for the underlying 'population at risk' becomes quite visible. Whereas the *concentration of crime* is highest in the city center (left map), the *crime risk* (right map) is in fact much higher in several more distant areas that are known for high concentrations of socially disadvantaged persons. Given the imperfect nature of the population data these results should be interpreted as a broad view on the distribution of crime risk that, nevertheless, has important policy implications.

Single kernel density of crime incidences
(assault & battery, Cologne 1999/2000)



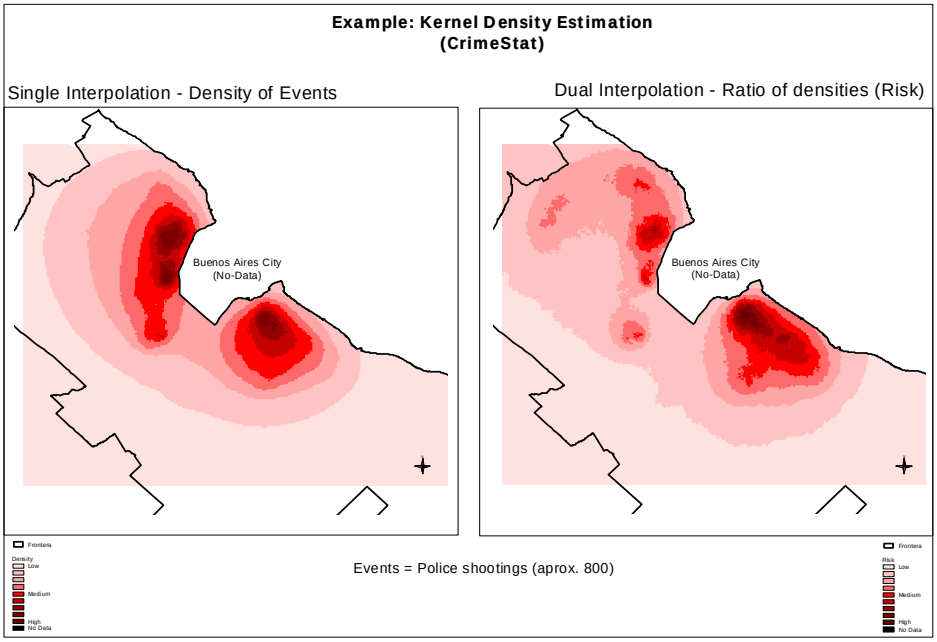
Dual kernel density of crime incidences
relative to population at risk



Kernel Density Interpolation of Police Confrontations in Buenos Aires Province, Argentina: 1999

Gastón Pezzuchi
Crime Analyst
Buenos Aires Province Police Force
Buenos Aires, Argentina

One of our first tryouts with the *CrimeStat* software involved the calculation of both single and dual kernel density interpolations using data on 1999 confrontations with the police within Buenos Aires Province, an area that covers 29 counties around the Federal Capital. The confrontations include mostly gun fights with the police but also other attacks (e.g., knives, rocks, sticks). In the last three years, there has been an increase in confrontations with the police. The single interpolation shows a density surface that gives a good picture of the ongoing level of violence while the dual interpolations shows a risk surface using the personnel deployment data; the latter are confrontations relative to the number of police deployed. Typically, police are allocated to areas according to crime rates.\



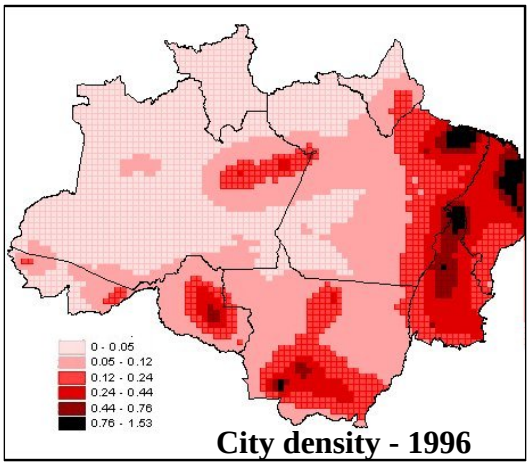
Both images are quite different, suggesting varying policing strategies. For example, though there are two well-defined hot spot areas in the Province (one in the north, the other in the south), the high levels of risk detected in the southern areas came as a complete surprise. The northern area has a higher crime rate than the southern area, hence a high police deployment. However, the level of confrontation is approximately equal between the two areas.

Evolution of the Urbanization Process in the Brazilian Amazonia

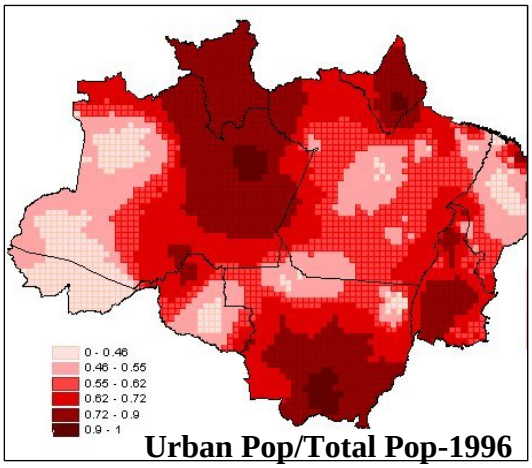
Silvana Amaral, Antônio Miguel V. Monteiro, Gilberto Câmara, José A. Quintanilha
INPE, Instituto Nacional de Pesquisas Espaciais, Brazil

The Brazilian Amazon rain forest is the world's largest contiguous area of tropical rain forest in the world. During the last three decades, the region has experienced the largest urban growth rates in Brazil, a process that has reorganized the network of human settlements in the region. We used the *CrimeStat* single and dual kernel density routines to visualize trends in urbanization from 1996 to 2000 in Amazonia. Two variables were used to measure urbanization: 1) the concentration of urban nuclei (city density); and 2) the ratio of urban to total population.

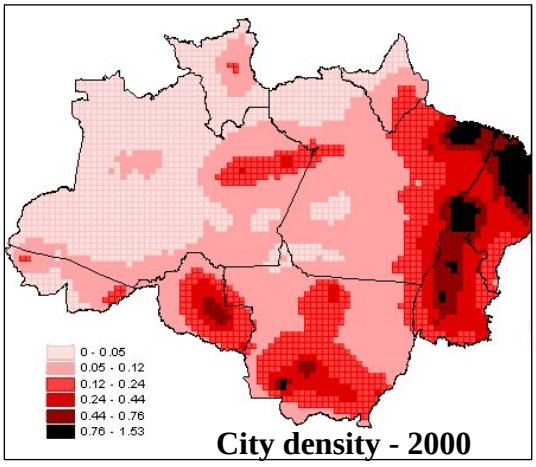
The concentration of cities was spatially associated with federal roads in the eastern and southern portions, and along the Amazonas River in the middle of the region. Additionally, the surfaces of urban population show that city density is not always associated with large urban populations. From 1996 to 2000 city density increased in the western Amazonia (Pará state) at a greater rate than the growth of the urban population. In the southeastern part of the region (Rondônia state), there were many urban centers. But the ratio of urban to total population was small, indicating that they are predominately agricultural regions.



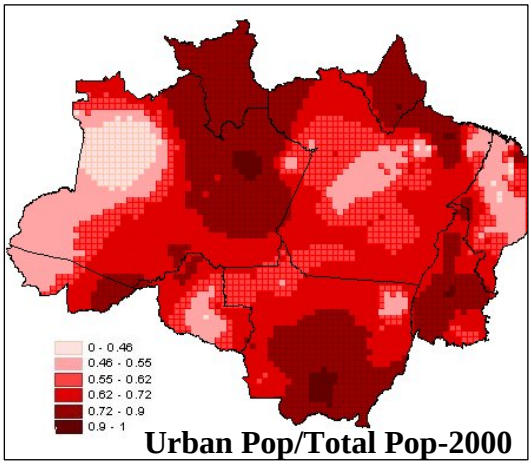
City density - 1996



Urban Pop/Total Pop-1996



City density - 2000



Urban Pop/Total Pop-2000

Using Small Area Estimation to Target Health Services in Harris County, TX

Thomas F. Reynolds, MS
University of Texas-Houston School of Public Health

In Texas, the City of Houston and Harris County organized a Public Health Task force to make recommendations concerning the provision of health services for those without health insurance. Task force members wanted to know approximately how many area citizens did not have health insurance.

Data from the two most recent Current Population Survey Annual Social and Economic Supplements (CPS-ASEC, 2003-04) were used to derive a synthetic estimate using a stratified model. Estimates were calculated at census tract and block group levels. Selected political divisions were clipped from base maps for political officials and legislators.

Percentages are indicative of risk. On the other hand, numbers are essential for targeting physical resources. There is seldom a perfect correspondence between high percentages and large numbers. For example, an area with a concentration of multi-family housing may have a relatively small percentage, but a large number, of uninsured. Percentage maps of the uninsured (figure 1) are generally clustered and informative; however, due to large variations in population numbers at both levels of census geography, maps of the population densities of uninsured proved most valuable to officials (figure 2).

CrimeStat was used to develop the density maps. The single kernel density routine was used to estimate the density of block group values using the centroid to represent the values and the number of uninsured as an intensity value. The Moran Correlogram was used to select the type of kernel for the single-kernel interpolation (a uniform distribution) and an optimal bandwidth.

Fig. 1: Percent Uninsured

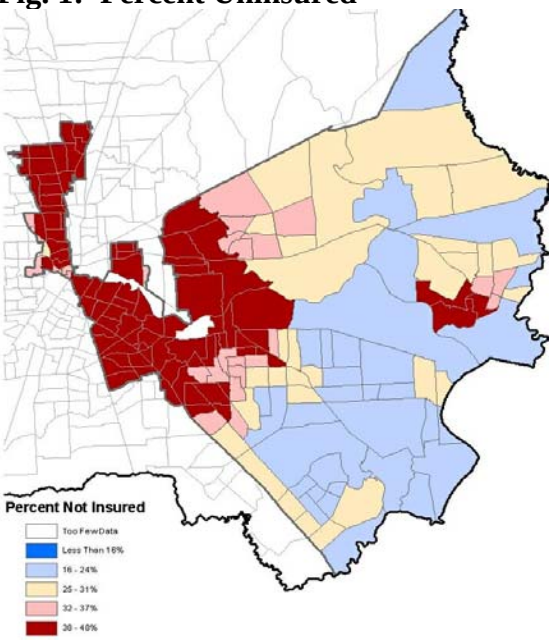
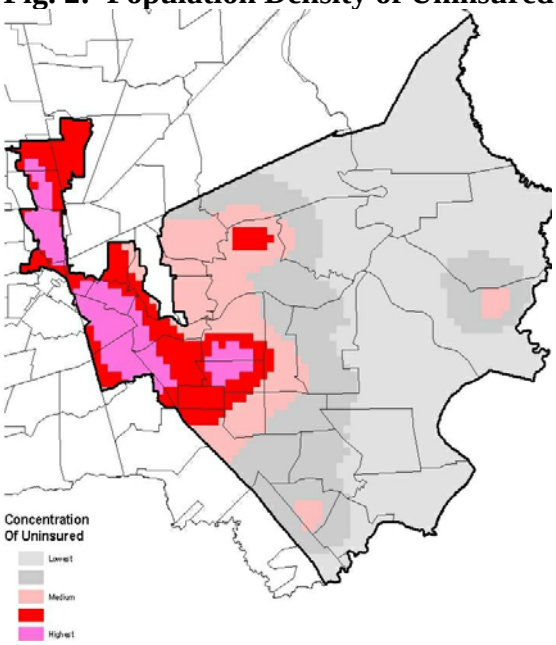


Fig. 2: Population Density of Uninsured

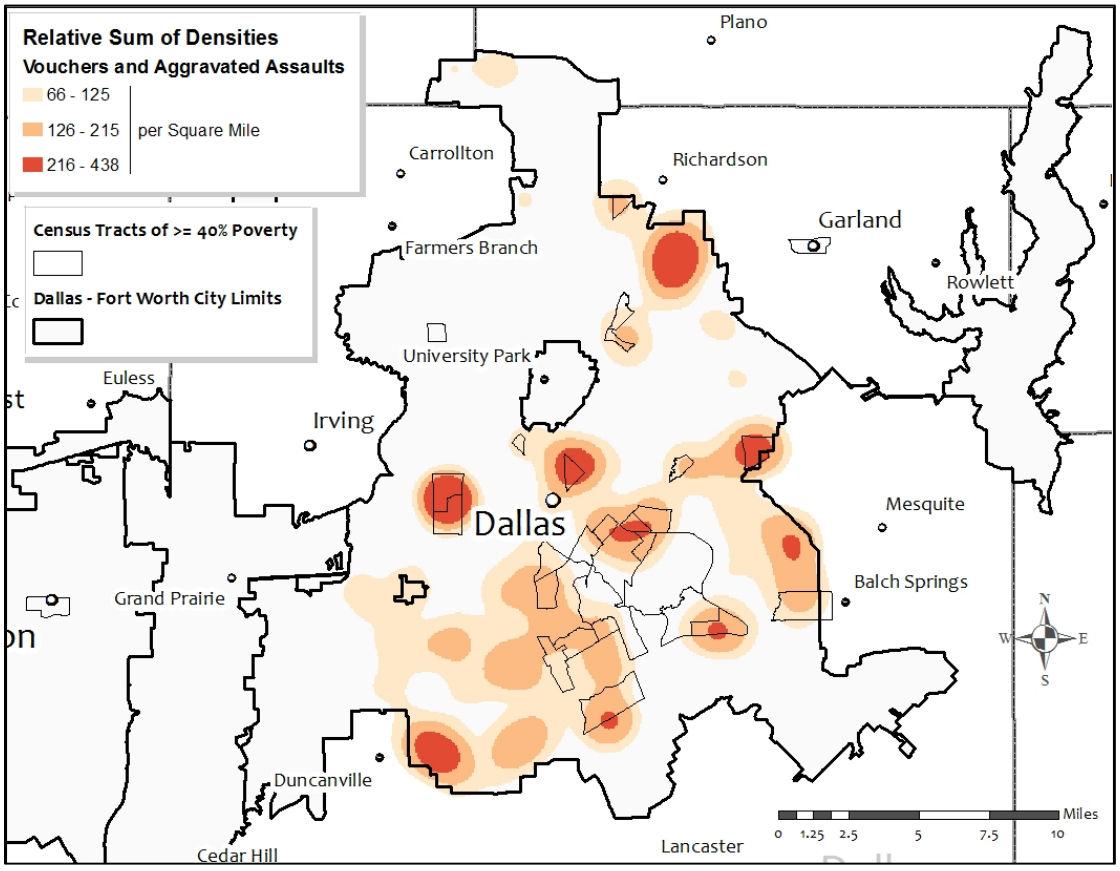


Identifying Voucher Holder and Crime Concentrations using Dual Kernel Density Estimation

Ron Wilson, U.S. Department of Housing and Urban Development

Public housing authorities and law enforcement are cooperating to reduce neighborhood crime where Housing Choice Voucher Program (HCVP) participants concentrate. When police departments and housing authorities can identify geographically combined voucher holder and crime concentrations, more specific strategies can be employed to reduce those crimes and prevent victimization. Aggravated assaults are common in and around neighborhoods where HCVP households concentrate. I used the relative sum of densities option of the Dual Kernel Density Estimation routine in *CrimeStat IV* to identify where HCVP households and aggravated assaults were concentrated in 2010.

Several areas of voucher holder and aggravated assault concentrations are revealed with gradations in density, some in census tracts with high poverty. The Dallas Police Department might deploy varying community policing approaches in these areas based on concentration grade to reduce assault opportunities while building relationships with neighborhood residents. The Dallas Public Housing Authority may help voucher holders find safer neighborhoods to relocate outside the concentrated areas, in particular to areas with low poverty. These findings may also help Dallas city officials craft separate place-based polices that work to eliminate the root causes of aggravated assaults in these areas.



Chapter 11:

Head-Bang Interpolation

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Chapter 11:
Head-Bang Interpolation

Interpolation II Tab

The interpolation II tab includes the Head-Bang routine and the Interpolated Head-Bang routine. Figure 11.1 show the graphical interface for the Interpolation II page, which includes the Head-Bang and the Interpolated Head-Bang routines.

Head-Bang Technique

The Head-Bang statistic is a weighted two-dimensional smoothing algorithm that is applied to zonal (polygon) data, such as census tracts, traffic analysis zones, or zip codes. It was developed at the National Cancer Institute to smooth out 'peaks' and 'valleys' in health data that occur because of small numbers of events (Pickle, Mungiole, Jones, Gretchen, & White, 1996; Mungiole, Pickle, & Simonson, 2002; Pickle & Su, 2002).

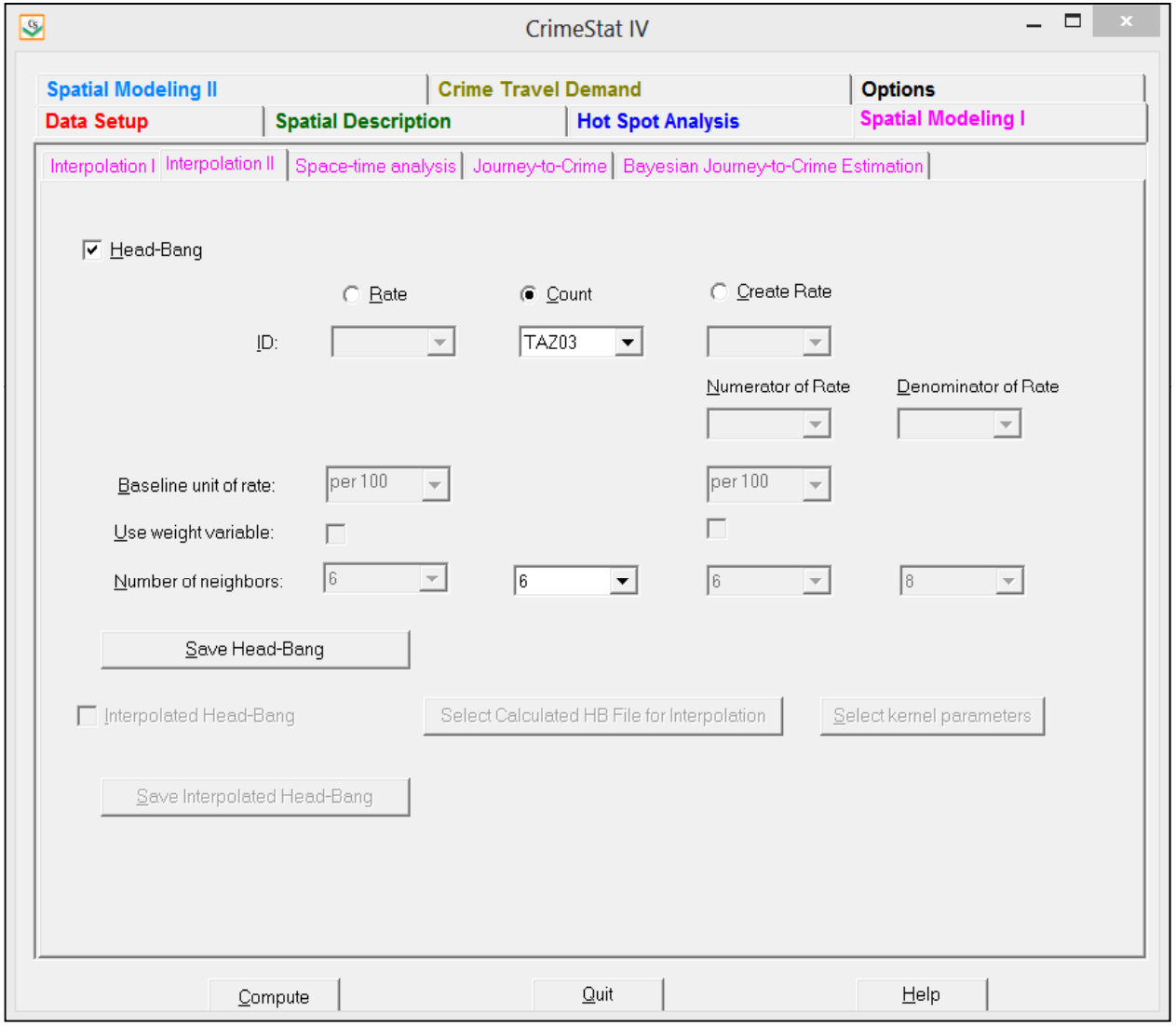
For example, with lung cancer rates (lung cancer cases relative to population), counties with small populations could show extremely high lung cancer rates with only an increase of a couple of cases in a year or, conversely, very low rates if there was a decrease in a couple of cases. On the other hand, counties with large populations will show stable estimates because their numbers are larger; their confidence intervals will be small because changes from one year to the next will have little effect on their rates.

However, unlike other smoothing techniques, such as kernel density interpolation (discussed in Chapter 10), the Head-Bang is designed to remove small-scale local variations within a data set while preserving regional trends. It is particularly useful where there are large differences in the population sizes of the different zones, which can lead to huge variability in the rates over the study area.

The aim of the Head-Bang statistic, therefore, is to smooth out the values for smaller geographical zones while generally keeping the values for larger geographical zones. It is a variance reduction technique. The methodology is based on the idea of a median-based Head-Banging smoother proposed by Tukey and Tukey (1981) and later implemented by Hansen (1991) in two dimensions. Mean smoothing functions tend to over-smooth in the presence of edges while median smoothing functions tend to preserve the edges.

Figure 11.1:

Interpolation II Screen



Median Smoother

The Head-Bang routine applies the concept of a median smoothing function to a three-dimensional plane (an area where there is information about an attribute that occurs at each location). It is a moving average that is placed over each location (zone) but which includes information on the values of the neighboring locations. But, unlike kernel density estimation which interpolates the data to a grid, the Head-Bang preserves the zonal organization.

The Head-Bang algorithm used in *CrimeStat* is a simplification of the methodology proposed by Mungiole, Pickle and Simonson (2002) but similar to that used by Pickle and Su (2002).[1] Consider a set of zones with a variable being displayed. In a raw map, the value of a variable assigned to any one zone is independent of the values for nearby zones. However, in a Head-Bang smoothing, the value of any one zone becomes a function of the values of nearby zones. It is a moving average that provides an estimate for the value of the zone, similar to kernel density interpolation (discussed in Chapter 10) but it preserves the geographical arrangements of the zones. It is useful for eliminating extreme values in a distribution and adjusting the values of zones to be similar to their neighbors.

Data Elements

There are three elements to the data that are relevant to the Head-Bang. First, there is a zone structure. All the data are represented by zones. Second, for each zone, there is a variable of interest that will be smoothed, Z_i. This variable could be a count (e.g., the number of crimes), a rate (e.g., the number of crimes relative to the population), or even a continuous variable (e.g., median household income). Third, each zone has a weight variable, W_i (e.g., population). The values of the variable of interest, Z_i, are then estimated using the weights, W_i.

Zone Structure

The user has to choose an appropriate zone organization. A trade-off has to be made between the geographical size of the zone and the value of the weighting variable (typically population). Ideally, the zones should be as small as possible in terms of area (e.g., census tracts or even blocks) so that local variations in rates can be seen. On the other hand, a small geographical zone can have a small population, which creates volatility in the rate from one year to the next. While the technique can smooth the rates of zones with small populations, huge variability in the rates over time might be seen. Thus, choosing larger population zones would

[1] The Head-Bang statistic is sometimes written as Head Bang or even Headbang. We prefer to use the term with the hyphen (Head-Bang).

produce more stable rates. The technique has been used on zones as large as counties for national comparisons (Pickle & Su, 2002). With large zones, local variations cannot be seen, though for national comparisons that is less critical.

Local Neighborhood

The procedure works through a local neighborhood around each zone. A neighborhood can be defined in different ways. The *CrimeStat* routine uses the nearest neighbor routine (discussed in Chapter 6) to identify the *K* nearest neighbors where *K* is defined by the user. Thus, distance is the criterion for identifying a zone as being a nearest neighbor, consistent with the approach of Mungiole, Pickle and Simonson (2002). An alternative definition is that of contiguity (or adjacency) so that the neighbors share a common border. In this version of *CrimeStat*, this definition is not implemented.

For each zone in turn (the central zone), a set of neighbors is defined. Mungiole and Pickle (1999) found that 6 neighbors generally produced small errors between the actual values and the smoothed values, and that increasing the number did not reduce the error substantially. On the other hand, increasing the number of neighbors smoothed the data too much. They also found that choosing fewer than 6 neighbors could sometimes produce unstable results.

Triplets

In the original formulation of the Head-Bang technique (Hansen, 1991; Mungiole, Pickle & Simonson, 2002), the Head-Bang was applied to *triplets* around the central point (zone), the zone which is to be smoothed. Since the aim of the technique is to smooth zones that have similar underlying rates while highlighting regional differences in rates, the idea was to create a separation or cleavage in the neighborhood around the central zone. Hence, Hansen (1991) proposed the concept of a *triplet*.

A triplet is two other points (endpoints) that forms a straight line with the central point.[2] The line should separate neighboring zones with higher rates from those with lower rates. However, since the three points will not usually be in a perfect straight line, the angle formed between the central point and the two endpoints must be greater than a certain threshold. They typically used a minimum angle of 135^0 separation.

2 They call this *collinear*. However, the term collinear has different meanings including variables that are high correlated with each other. To avoid confusion, this chapter does not use that term.

Screens

The two endpoints are then assigned to two groups (called *screens*). Of the two endpoints, the one with the higher value is assigned to a *high screen* while the one with the lower value is assigned to a *low screen*. After all neighborhood points have been assigned to each screen, the median of each screen is taken and the value of the central point compared to these.

Pickle and Su (2002), however, simplified the procedure by simply dividing the values of the neighbors into two screens irrespective of whether they formed a triplet with the central point or not. The results are virtually identical to the initial Head-Bang results. Consequently, that procedure is adopted in the *CrimeStat* version. The values of the neighbors are sorted from high to low irrespective of whether they form triplets or not and divided into the high screen and 'low screen groupings at the middle record. If the number of neighbors is even, then the two groups are of equal size and mutually exclusive; on the other hand, if the number of neighbors is odd, then the middle record is counted twice, once with the high screen and once with the low screen.

For each screen, the median value is calculated. The median of the high screen is called the *high median* and the median of the low screen is called the *low median*.

Decision Rules for Head-Bang Statistic

The value of the central zone is then compared to these two medians. The decision rules are as follows:

1. First, if the value of the central zone falls between the two medians (low and high, then the central zone retains its value.

2. Second, if the value of the central zone is *either* higher than the high median *or* lower than the low median, then its weight determines whether it is adjusted. It is compared to the screen to which it is closest (high or low). If it has a weight that is greater than all the weights of its closest screen, then it maintains it value.

   For example, if the central zone has a value greater than the high median but also greater has a weight greater than any of the high screen zones, then it still maintains its value.

3. However, if its weight is smaller than the weights of ***any*** zone in its closest screen, then the central zone takes the value of the median for the screen to which it is closest.

For example, if the central zone is closer to the high median than to the low median but has a weight that is smaller than one or more zones in the high screen, then it takes the high median as its value. Similarly, if the central zone is closer to the low median than to the high median but has a weight that is smaller than one or more zones in the low screen, then it takes the low median as its value.

4. After all points (zones) have been assigned estimates for the variable of interest, Z_i, the process is repeated 9 more times to ensure that the final smoothing is stable.

The logic ensures that if a central zone is large in size relative to its neighbors (i.e., has a greater weight), then its observed rate is most likely an accurate indicator of risk. However, if the zone is smaller in size than its neighbors, then its value is adjusted to be like its neighbors. In this case, extreme rates, either high or low, are reduced to moderate levels (smoothed). 'Peaks' or 'valleys' are minimized while the values of real edges in the data are maintained.

Example to Illustrate Decision Rules

A simple example will illustrate this process. Suppose the intensity variable is a rate (as opposed to a count; see below). For each point, the eight nearest neighbors are examined. Taking one zone ("A"), suppose the eight nearest neighbors of zone A have the following values (Table 11.1).3 Note that the value at the central point (zone A) is not included in this list. These eight are the nearest neighbors only.

Table 11.1:
Example: Nearest Neighbors of Zone "A"

Neighbor	Rate	Weight
B	10	1000
C	15	3000
D	12	4000
E	7	1500
F	14	2300
G	16	1200
H	10	2000
I	12	2500

3 Mungiole and Pickle (1999) found that 6 neighbors generally produced small errors between the observed and smoothed values. However, sometimes adding more neighbors can improve the stability. The example above uses 8 neighbors.

Next, the 8 neighbors are sorted from the lowest rate to the highest (Table 11.2). The record number (neighbor) and weight value are sorted along with the rate.

Table 11.2:
Sorted Nearest Neighbors of Zone "A"

Neighbor	Rate	Weight
E	7	1500
B	10	1000
H	10	2000
D	12	4000
I	12	2500
F	14	2300
C	15	3000
G	16	1200

Third, a cumulative sum of the weights is calculated starting with the lowest rate (Table 11.3). Fourth, the neighbors are then divided into two groups at the median. Since the number of records is even, then the low screen records are E, B, H, and D while the high screen records are I, F, C and G. The weighted medians of the low screen and high screen are calculated. Since these are rates, the low screen median is calculated from the first four records while the high screen median is calculated from the second four records.

Table 11.3:
Cumulative Weights for Nearest Neighbors of Zone "A"

Neighbor	Rate	Weight	Cumulative Weight
E	7	1500	1500
B	10	1000	2500
H	10	2000	4500
D	12	4000	8500
I	12	2500	11000
F	14	2300	13300
C	15	3000	16300
G	16	1200	17500

Using record E as an example, the calculations are as follows (assume the baseline is 'per 10,000'). The rate is multiplied by the weight and divided by the baseline (for example, 7 * 1500/10000 = 1.05). This is called the ***score***; it is an estimate of the count of events in that zone. Table 11.4 shows the scores for each record and the cumulative score. The cumulative score of each screen is divided in half to obtain the median.

For the low screen, the median score is 8.85/2 = 4.425. This falls between records H and D. To estimate the rate associated with this median score, the interval in scores between records H and D is interpolated, and then converted to rates. The interval between records H and D is 4.80 (8.85-4.05). The low screen median score, 4.425, is (4.425-4.05)/4.80 = 0.0781 of the distance for that interval. For the rates, the interval between records H and D is 2 (12-10). Thus, 0.0781 of that interval is 0.1563. This is added to the rate of record H to yield a low median rate of 10.1563.

For the high screen, the median score is 12.64/2 = 6.32. This falls between records F and C. To estimate the rate associated with this median score, the interval in scores between records F and C is interpolated, and then converted to rates. The interval between records F and C is 4.50 (10.72-6.22). The low screen median score, 6.32, is (6.32-6.22)/4.50 = 0.0222 of the distance in that interval. The interval between the rates of records F and C is 1 (15-14). Thus, 0.0222 of that interval is 0.0222. This is added to the rate of record F to yield a high median rate of 14.0222.

Table 11.4:

Cumulative Scores by Screens for Nearest Neighbors of Zone "A"

Low screen

Neighbor	Rate	Weight	"Score" Rate*Weight/Baseline	Cumulative Score
E	7	1500	1.05	1.05
B	10	1000	1.00	2.05
H	10	2000	2.00	4.05
D	12	4000	4.80	8.85

High screen

Neighbor	Rate	Weight	"Score" Rate * Weight/Baseline	Cumulative Score
I	12	2500	3.00	3.00
F	14	2300	3.22	6.22
C	15	3000	4.50	10.72
G	16	1200	1.92	12.64

Finally, the rate associated with the central zone (zone A in our example) is compared to these two medians. If its rate falls between these medians, then it keeps its value. For example, if the rate of zone A is 13, then that falls between the two medians (10.1563 and 14.0222).

On the other hand, if its rate falls outside this range (either lower than the low median or higher than the high median), its value is determined by its weight relative to the screen to which it is closest. For example, suppose zone A has a rate of 15 with a weight of 1700. In this case, its rate is higher than the high median (14.0222) but its weight is smaller than three of the weights in the high screen. Therefore, it takes the high median as its new smoothed value. Relative to its neighbors, it is smaller than three of them so that its value is probably too high.

But, suppose it has a rate of 15 and a weight of 3000? Even though its rate is higher than the high median, its weight is also higher than the four neighbors making up the high screen. Consequently, it keeps its value. Relative to its neighbors, it is a large zone and its value is probably accurate.

For counts (discussed below), the comparison is simpler because all weights are equal. Consequently, the count of the central zone is compared directly to the two medians. If it falls between the medians, it keeps its value. If it falls outside the two medians, then it takes the one to which it is closest (the high median if it has a higher value or the low median if it is lower).

Rates and Counts

The original Head-Bang statistic was applied to rates (e.g., number of lung cancer cases relative to population). In the *CrimeStat* implementation, the routine can be applied to counts (volumes) or rates or can even be used to estimate a rate from counts. Counts have no weights (i.e., they are self-weighted). In the case of rates, though, they should be weighted (e.g., by population). The most plausible weighting variable for a rate is the same baseline variable used in the denominator of the rate (e.g., population, number of households) because the rate variance is proportional to 1/baseline (Pickle and Su, 2002).

Need to Define Intensity variable

Whether a count or a rate variable is to be analyzed, an *Intensity* variable must be defined on the Primary file page (see Chapter 3). The Intensity variable is the variable that will be smoothed. If it is not defined, then the Head-Bang routine will not be available. Note that a separate weight variable on the Primary file page should also be used to weight the data if a rate is being analyzed. But, this can only work in conjunction with a defined intensity variable. In other words, be sure that an intensity variable is defined to use the Head-Bang routine.

Smoothed median for count variable

With a count variable, there is only a count (the number of events). There is no weighting of the count since it is self-weighting (i.e., the number equals its weight). In *CrimeStat*, the count variable is defined as the Intensity variable on the Primary file page. For a count variable:

1. If the value of the central zone falls between the two medians (low median and high median), then the central zone retains its value.

2. If the value of the central zone is higher than the high median, then it takes the high median as its smoothed value;

3. If the value of the central zone is lower than the low median, then it takes the low median as its smoothed value.

Smoothed median for rate variable

With a rate, there is both a value (the rate) and a weight. The rate variable is defined as the Intensity variable on the Primary file page. However, there is a separate weight that must be applied to this rate to distinguish a large zone from a small zone. In *CrimeStat*, the weight variable is always defined on the Primary file page as the Weight variable.

Depending on whether the rate is input as part of the original data set or created out of two variables from the original data set, it will be defined slightly differently. If the rate is part of the original data set, then it is defined as the intensity variable on the Primary file page. However, if the rate is created out of two variables from the Primary file data set, it is defined on the Head-Bang interface under 'Create rate'.

Setup

For either a rate or a count, the statistic requires an *intensity* variable be defined in the Primary file. The user must specify whether the variable to be smoothed is a rate variable, a count variable, or two variables that are to be combined into a rate. If a weight is to be used (for either a rate or the creating of a rate from two count variables), then it must be defined as an Intensity on the Primary file page. Note that if the intensity variable is a rate, it should also be weighted. A typical weighting variable is the population size of the zone.

The user has to complete the following steps to run the routine:

1. **Define input file** and coordinates on the Primary file page

2. **Define an intensity variable**, Z(intensity), on the Primary file page.

3. OPTIONAL: **Define a weighting variable** in the weight field on the Primary file page (for rates and for the creating of rates from two count variables)

4. **Define an ID variable** to identify each zone.

5. **Select data type**:

   A. **Rate**: the variable to be smoothed is a rate variable which calculates the number of events (the numerator) relative to a baseline variable (the denominator).

      a. The baseline units should be defined, which is an assumed multiplier in powers of 10. The default is ‘per 100' (percentages) but other choices are 0 (no multiplier used), ‘per 10' (rate is multiplied by 10), ‘per 1000', ‘per 10,000', ‘per 100,000', and ‘per 1,000,000'. This is not used in the calculation but for reference.

      b. If a weight is to be used, the ‘Use weight variable’ box should be checked.

   B. **Count**: the variable to be smoothed is a raw count of the number of events. There is no baseline used.

   C. **Create Rate**: A rate is to be calculated by dividing a count variable by a baseline variable.

      a. The user must define the count (numerator) and baseline (denominator) variables.

      b. The baseline scale ***must*** be defined, which is an assumed multiplier in powers of 10. The default is ‘per 100' (percentages) but other choices are 1 (no multiplier used),

'per 10' (rate is multiplied by 10), 'per 1000', 'per 10,000', 'per 100,000', and 'per 1,000,000'. This is used in the calculation of the rate.

c. If a weight is to be used, the 'Use weight variable' box should be checked.

6. **Select number of neighbors**. In CrimeStat, the number of neighbors can run from 4 through 40. The default is 6. If the number of neighbors selected is even, the routine divides the data set into two equal-sized groups. If the number of neighbors selected is odd, then the middle zone is used in calculating both the low median and the high median. It is recommended that an even number of neighbors be used (e.g., 4, 6, 8, 10).

7. **Select output file.** The output can be saved as a dbase 'dbf' file. If the output file is a rate, then the prefix RateHB is used. If the output is a count, then the prefix VolHB is used. If the output is a created rate, then the prefix CrateHB is used.

8. **Run the routine** by clicking 'Compute'.

Output

The Head-Bang routine creates a 'dbf' file with the following variables:

1. The ID field
2. The X coordinate
3. The Y coordinate
4. The smoothed intensity variable (called 'Z_MEDIAN'). Note that this is not a Z score but a smoothed intensity (Z) value
5. The weight applied to the smoothed intensity variable. This will be automatically 1 if no weighting is applied.

The 'dbf' file can then be linked to the input 'dbf' file by using the ID field as a matching variable. This would be done if the user wants to map the smoothed intensity variable.

Example 1: Using the Head-Bang Routine for Mapping Houston Burglaries

Figure 11.2 shows a map of Houston burglaries by traffic analysis zones. The mapped variable is the number of burglaries committed in 2006. On the Head-Bang interface, the 'Count' box was checked, indicating that the number of burglaries will be estimated. The number of neighbors was left at the default 6. The output 'dbf' file was then linked to the input 'dbf' file using the ID field to allow the smoothed intensity values to be mapped. The variable is mapped with 5 equal-size intervals. Quintiles could also have been used.

Figure 11.3 show a smoothed map of the number of burglaries conducted by the Head-Bang routine. To be consistent with Figure 11.2, 5 equal-size intervals were also used. Comparing the two maps, it can be seen that there are fewer zones in the Head-Bang map that are in the lowest interval/bin (in yellow). The actual count was 498 zones with scores of less than 10 in Figure 11.3 compared to 528 zones in Figure 11.2. Also, there are fewer zones in the highest interval/bin (in black) as well. The actual count was 181 zones with scores of 40 or more in Figure 11.3 compared to 215 zones in Figure 11.2. In other words, the Head-Bang routine has assigned many of the highest or lowest values to the median values of their neighbors.

Example 2: Using the Head-Bang Routine for Mapping Burglary Rates

The second example shows how the Head-Bang routine can smooth rates. In the Houston burglary data base, a rate variable was created that divided the number of burglaries in 2006 by the number of households in 2006. This variable was then multiplied by 1000 to minimize the effects of decimal place (the baseline unit). Figure 11.4 show the raw burglary rate (burglaries per 1,000 households) for the City of Houston in 2006.

The Head-Bang routine was set up to estimate a rate for this variable (Burglaries per 1000 Households). On the Primary file page, the intensity variable was defined as the calculated rate (burglaries per 1,000 households) because the Head-Bang will smooth the rate. Also, a weight variable was selected on the Primary file page. In this example, the weight variable was the number of households. With any rate, there is always the potential of a small zone producing a very high rate. Consequently, the estimates were weighted to ensure that the values of each zone are proportional to their size. Zones with larger numbers of households will keep their values whereas zones with small numbers of households will most likely change their values to be closer to their neighbors.

On the Head-Bang interface, the 'Rate' box was checked (Figure 11.5). The ID variable was selected (which is also TAZ03). The baseline number of units was set to 'per 1000'; this is for information purposes, only, and will not affect the calculation.

Figure 11.2:

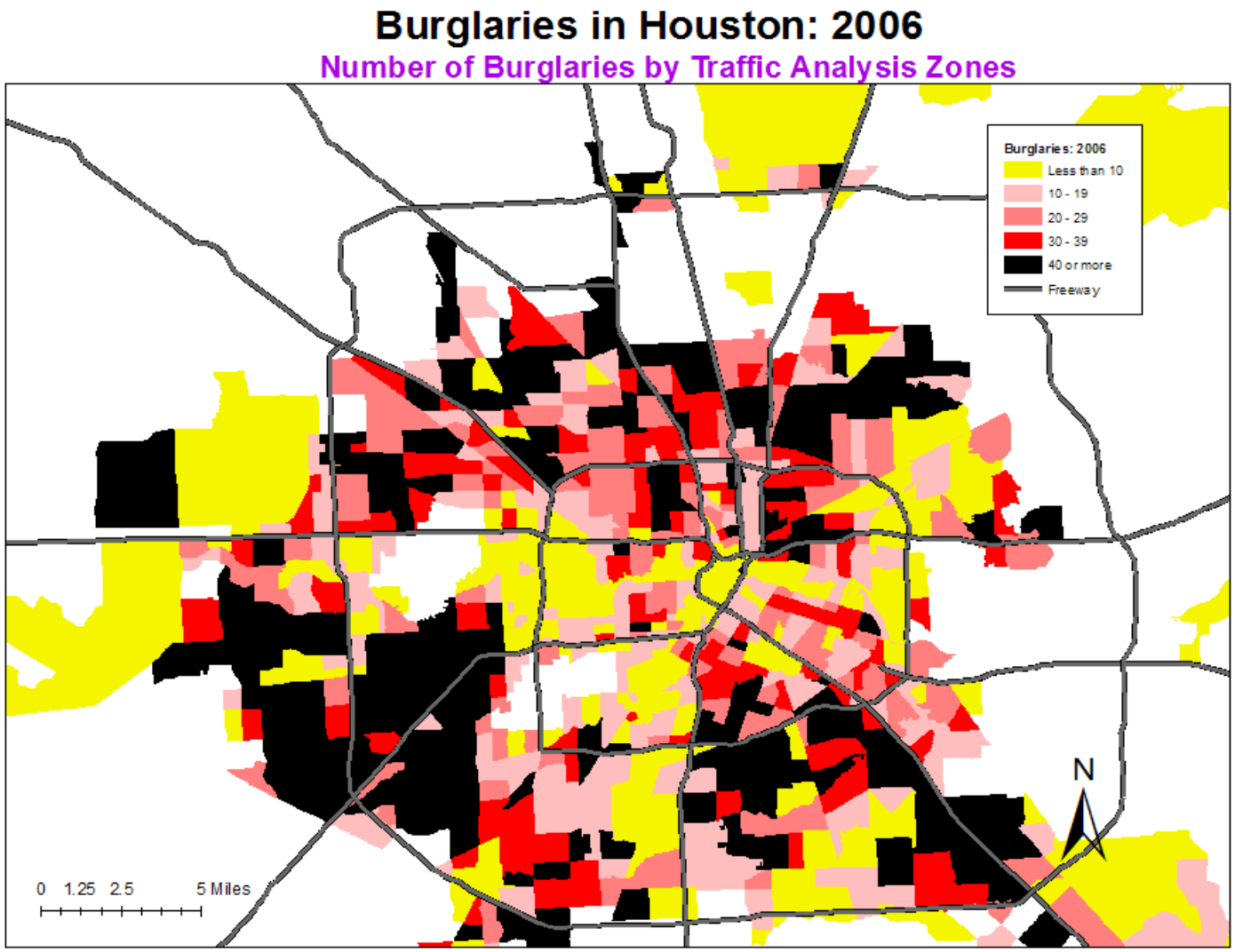

Burglaries in Houston: 2006
Number of Burglaries by Traffic Analysis Zones
Burglaries: 2006
Less than 10
10 - 19
20 - 29
30 - 39
40 or more
Freeway
N
0 1.25 2.5 5 Miles

Figure 11.3:

Head-Bang Estimate of Burglaries in Houston: 2006

Smoothed Number of Burglaries by Traffic Analysis Zones

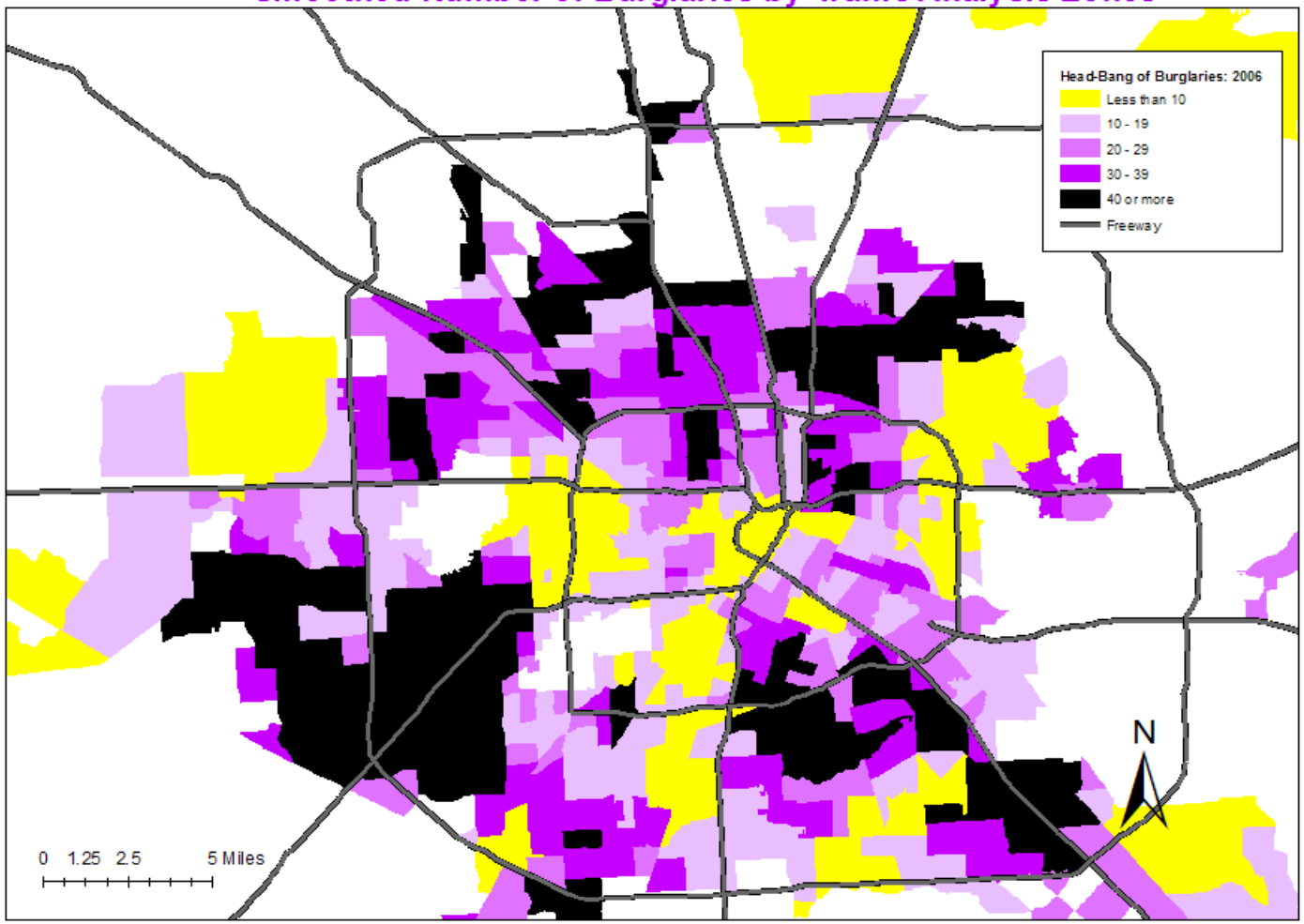


Figure 11.4:

Burglary Rate in Houston: 2006

Number of Burglaries per 1000 Households

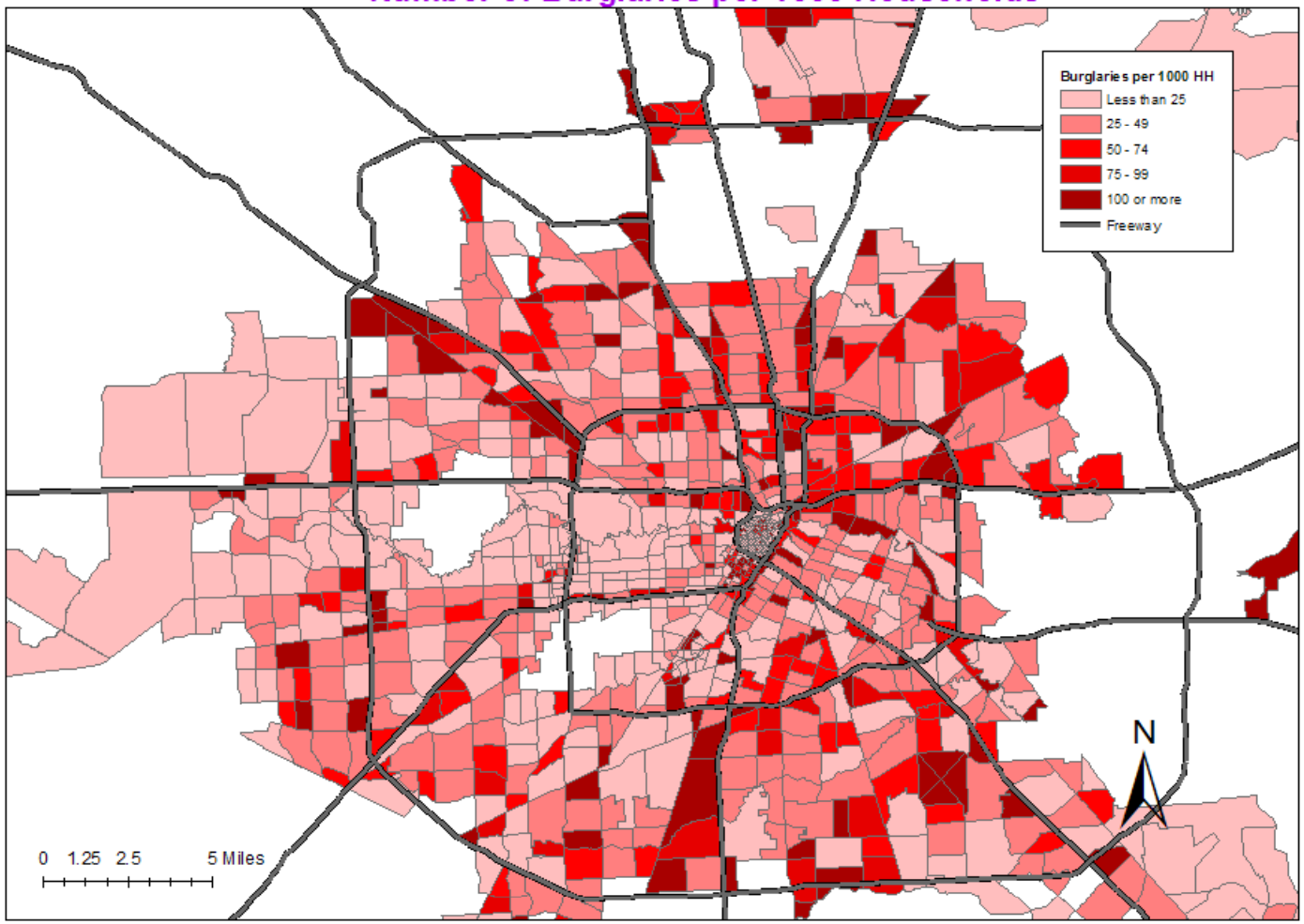
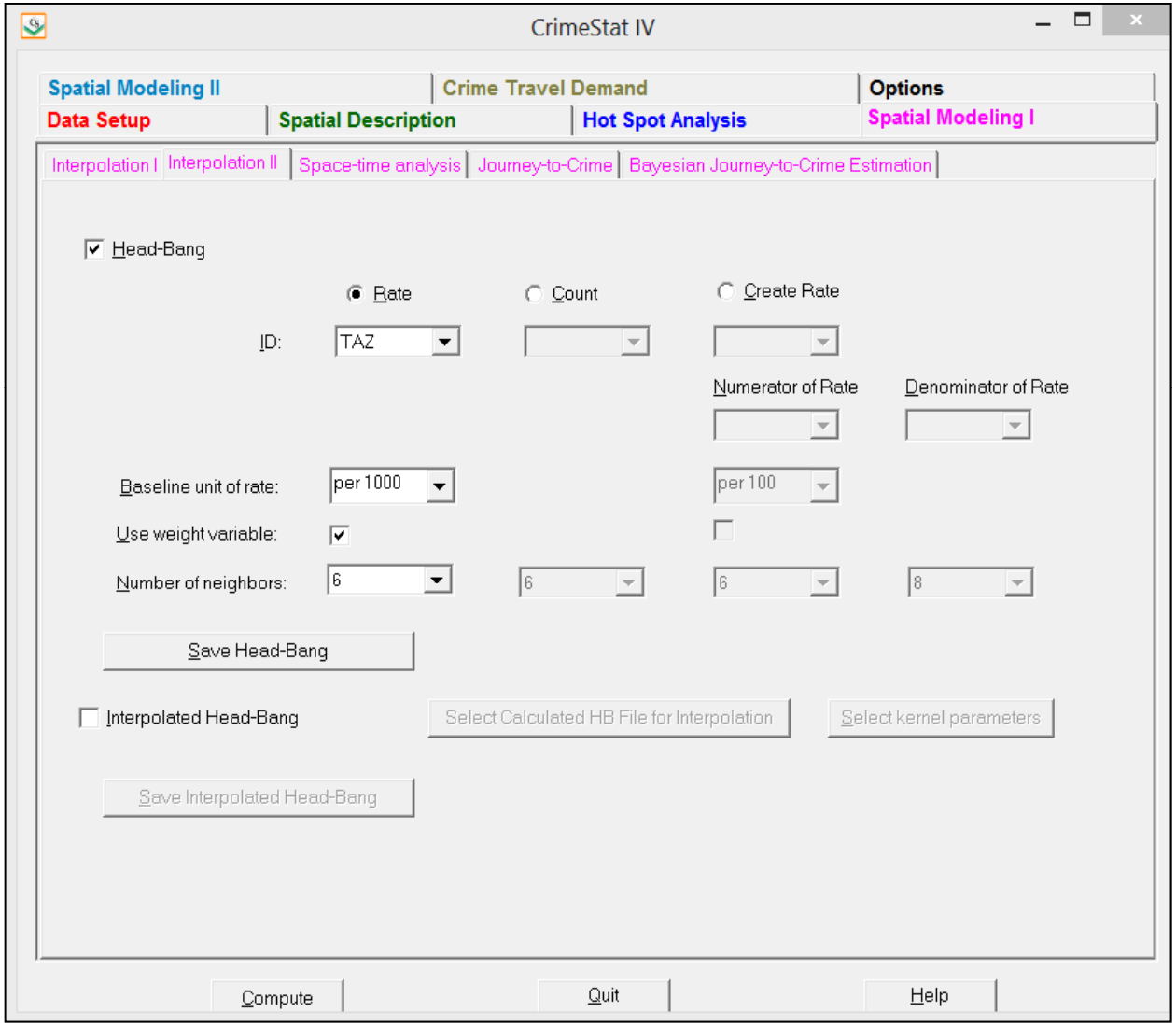


Figure 11.5:

Defining Rates with Head-Bang Routine



On the Primary file page, the number of households was chosen as the weight variable and the 'Use weight variable' box was checked under the Head-Bang routine.. The number of neighbors was left at the default 6. Finally, an output 'dbf' file was defined in the 'Save Head-Bang' dialogue.

The output 'dbf' file was linked to the input 'dbf' file using the ID field to allow the smoothed rates to be mapped. Figure 11.6 show the result of smoothing the burglary rate. As can be seen, the rates are more moderate than with the raw numbers (comparing Figure 11.4 with Figure 11.6). There are fewer zones in the highest rate category (100 or more burglaries per 1,000 households) for the Head-Bang estimate compared to the raw data (64 compared to 185) but there are also more zones in the lowest rate category (0-24 burglaries per 1,000 households) for the Head-Bang compared to the raw data (585 compared to 520). In short, the Head-Bang routine reduced the rates throughout the map.

Example 3: Using the Head-Bang Routine to Create Burglary Rates from Separate Counts of Burglaries and Households

The third example illustrates using the Head-Bang routine to create smoothed rates. In the Houston burglary data set, there are two variables that can be used to create a rate. First, there is the number of burglaries per traffic analysis zone. Second, there is the number of households that live in each zone. By dividing the number of burglaries by the number of households, an exposure index can be calculated. Of course, this index is not perfect because some of the burglaries occur on commercial properties, rather than residential units. But, without separating residential from non-residential burglaries, this index can be considered a rough exposure measure.

On the Head-Bang interface, the 'Create Rate' box is checked (Figure 11.7). The ID variable is selected (which is TAZ03 in the example). The numerator variable is selected which, in the example is the number of burglaries (BURG2006). Next, the denominator variable is selected. In the example, the denominator variable is the number of households (HH2006). The baseline units must be chosen and, unlike the rate routine, are used in the calculations. For the example, the rate is 'per 1,000' which means that the routine will calculate the rate (burglaries divided by households) but then will multiply by 1,000. On the Head-Bang page, the 'Use weight variable' box under the 'Create rate' column is checked.

Next, the number of neighbors are chosen, both for the numerator and for the denominator. One has to be careful about the denominator especially if some zones have very few households. The result would be an extreme rate estimate. To avoid this, it is recommended that a larger number of neighbors be used for the denominator than for the numerator. In the

Figure 11.6:

Head-Bang of Burglary Rate in Houston: 2006

Smoothed Estimate of Number of Burglaries per 1000 Households

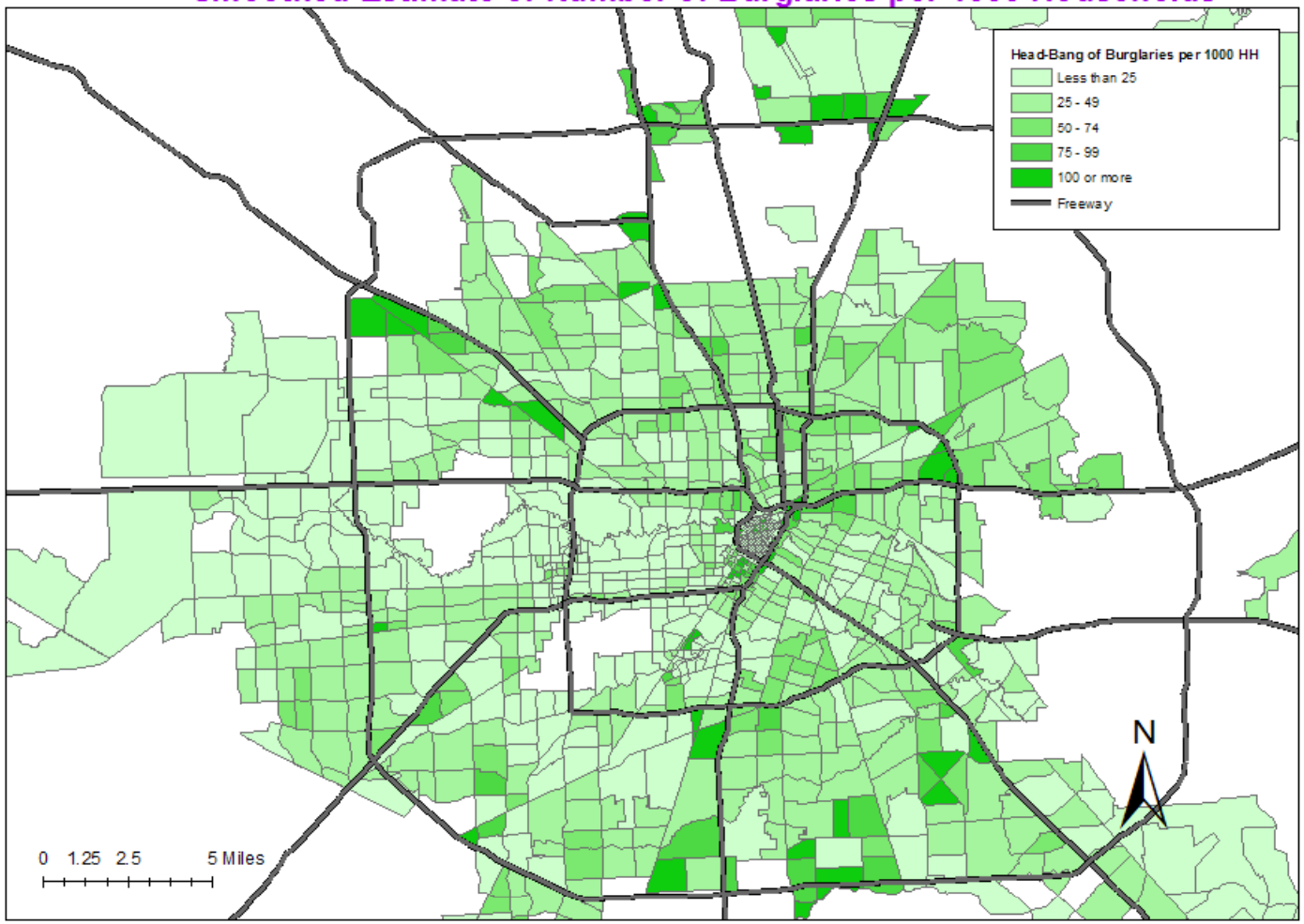
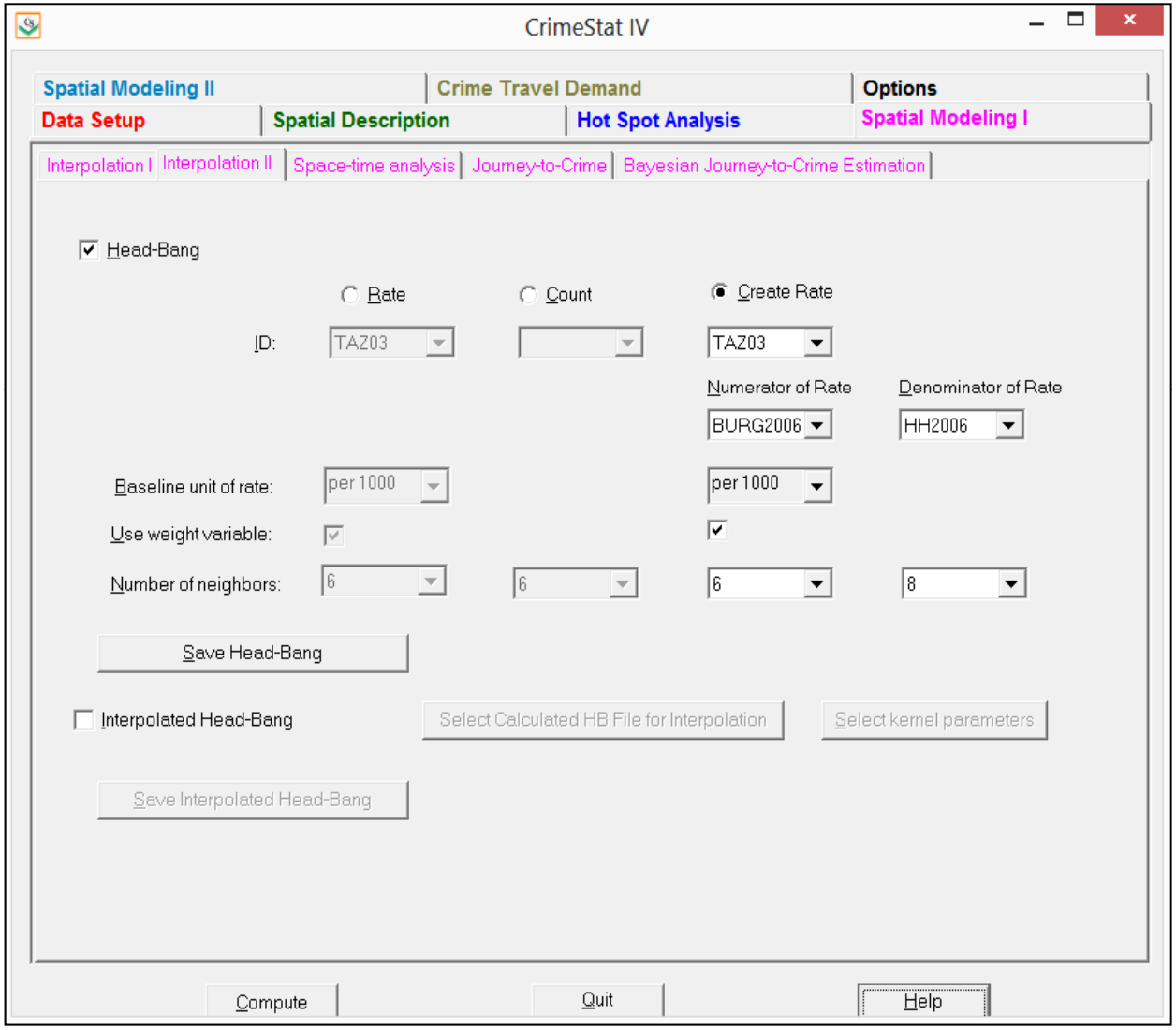


Figure 11.7:

Creating Rates with Head-Bang Routine



example, the default of 6 neighbors is chosen for the numerator variable (burglaries) while 8 neighbors are chosen for the denominator variable (households).

Finally, a 'dbf' output file was defined and the routine was run. The output 'dbf' file was then linked to the input 'dbf' file using the ID field to allow the smoothed rates to be mapped. Figure 11.8 show the results. Compared to the raw burglary rate (Figure 11.2), there are fewer zones in the highest category (36 compared to 185) but also more zones in the lowest category (607 compared to 520). Like the rate smoother, the rate that is created has reduced the rates throughout the map.

Uses of the Head-Bang Routine

The Head-Bang routine is useful for several purposes. First, it eliminates extreme measures, particularly very high ones ('peaks'). For a rate, in particular, it will produce more stable estimates. For zones with a small population, a few events can cause dramatic increases in the rate. The Head-Bang smoother will eliminate those extreme fluctuations. The use of population weights for estimating rates ensures that unusually high or low proportions that are reliable due to large populations are not modified whereas values based on small populations are modified to be more like those of the surrounding counties. Similarly, for counts (counts), the method will produce values that are more moderate.

Limitations of the Head-Bang Routine

On the other hand, the Head-Bang methodology does distort data. Because the extreme values are eliminated, the routine aims for more moderate estimates. However, those extremes may be real. Consequently, the Head-Bang routine should not be used to interpret the results for any one zone but more for the general pattern within the area.

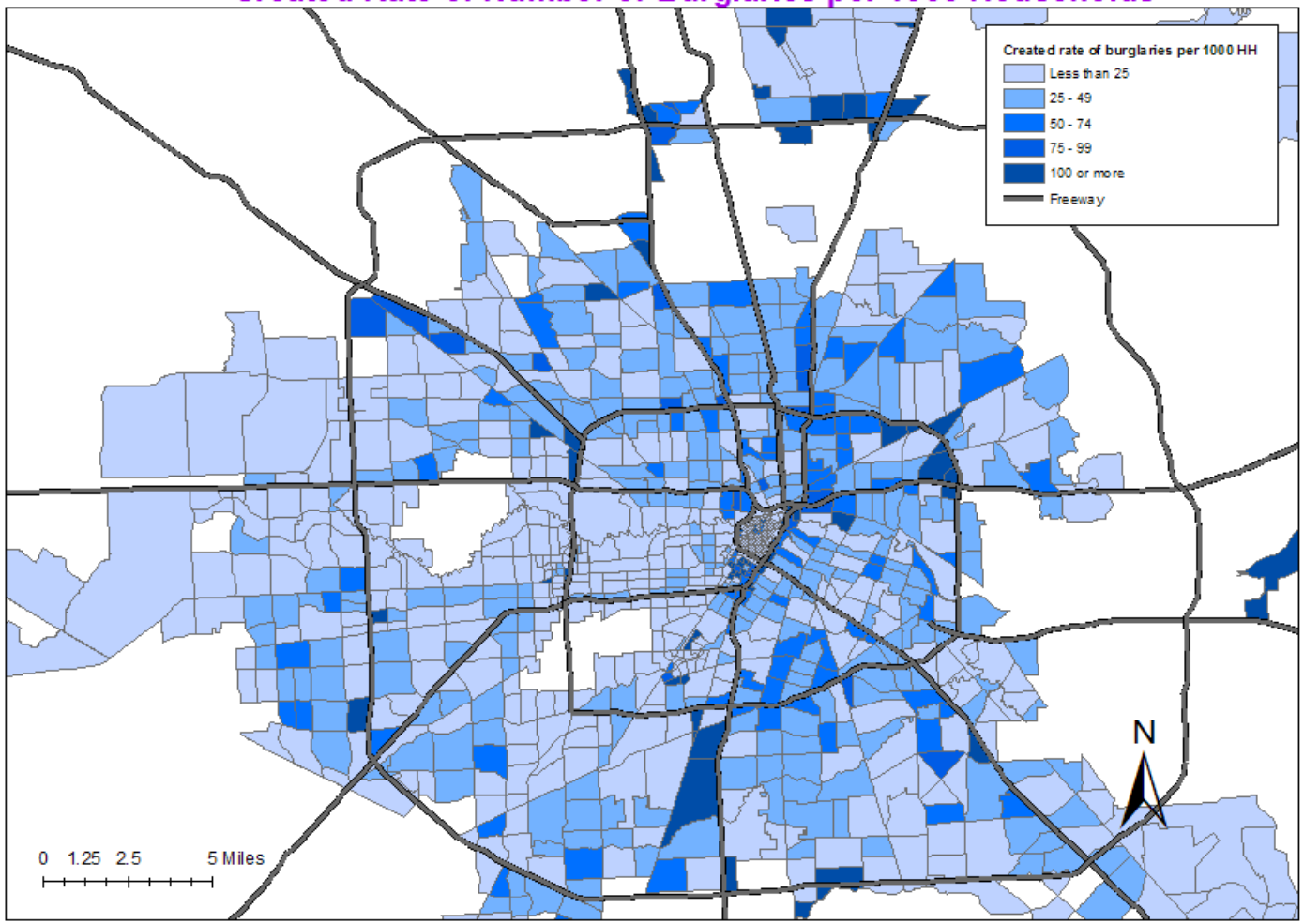
Also, there is often a trade-off that the user will have to make between the geographical size of the zone (smaller is better) with the stability of the estimates (larger population is better). Choosing zones that are very small (e.g., blocks or block groups) can produce unreliable estimates while choosing zones that are too large (e.g., districts or even counties) can eliminate meaningful local variations. The choice of zones is critical.

However, if used carefully, Head-Bang smoothing is a powerful tool for examining risk within a study area and for identifying changes in risk over time.

Figure 11.8:

Head-Bang of Burglary Rate in Houston: 2006

Created Rate of Number of Burglaries per 1000 Households



Interpolated Head-Bang Statistic

The Head-Bang calculations can be interpolated to a grid. If the user checks this box, then the routine will also interpolate the calculations to a grid using kernel density estimation. An output file from the Head-Bang routine is required. Also, a reference file is required to be defined on the Reference File page.

Essentially, the routine takes a Head-Bang output and interpolates it to a grid using a kernel density function. The same results can be obtained by inputting the Head-Bang output on the Primary file page and using the single kernel density routine on the Interpolations I page. The user must then define the parameters of the interpolation. However, there is no intensity variable in the Interpolated Head-Bang because the intensity has already been incorporated in the Head-Bang output. Also, there is no weighting of the Interpolated Head-Bang estimate.

The user must then define the parameters of the interpolation. Chapter 10 discussed these in more detail and provided guidelines, which will not be repeated here.

Method of Interpolation

There are five types of kernel distributions to interpolate the Head-Bang to the grid:

1. The **normal** kernel overlays a three-dimensional normal distribution over each point that then extends over the area defined by the reference file. This is the default kernel function. However, the normal kernel tends to over-smooth. One of the other kernel functions may produce a more differentiated map;

2. The **uniform** kernel overlays a uniform function (disk) over each point that only extends for a limited distance;

3. The **quartic** kernel overlays a quartic function (inverse sphere) over each point that only extends for a limited distance;

4. The **triangular** kernel overlays a three-dimensional triangle (cone) over each point that only extends for a limited distance; and

5. The **negative exponential** kernel overlays a three dimensional negative exponential function over each point that only extends for a limited distance.

The different kernel functions produce similar results though the normal is generally smoother for any given bandwidth.

Choice of Bandwidth

The kernels are applied to a limited search distance, called 'bandwidth'. For the normal kernel, bandwidth is the standard deviation of the normal distribution. For the uniform, quartic, triangular and negative exponential kernels, bandwidth is the radius of a circle defined by the surface. For all types, a larger bandwidth will produce smoother density estimates and both adaptive and fixed bandwidth intervals can be selected.

Adaptive bandwidth

An adaptive bandwidth distance is identified by the minimum number of other points found within a circle drawn around a single point. A circle is placed around each point, in turn, and the radius is increased until the minimum sample size is reached. Thus, each point has a different bandwidth interval. The user can modify the minimum sample size. The default is 100 points. If there is a small sample size (e.g., less than 500), then a smaller minimum sample size would be more appropriate).

Fixed bandwidth

A fixed bandwidth distance is a fixed search radius for each point. The user must define the radius and its distance units (miles, nautical miles, feet, kilometers, meters.)

Output (areal) units

Specify the areal density units as points per square mile, per squared nautical miles, per square feet, per square kilometers, or per square meters. The default is points per square mile.

Calculate Densities or Probabilities

The density estimate for each cell can be calculated in one of three ways:

Absolute densities

This is the number of points per grid cell and is scaled so that the sum of all grid cells equals the sample size. This is the default.

Relative densities

For each grid cell, this is the absolute density divided by the grid cell area and is expressed in the output units (e.g., points per square mile).

Probabilities

This is the proportion of all incidents that occur in each grid cell. The sum of all grid cells equals 1. Select whether absolute densities, relative densities, or probabilities are to be output for each cell. The default is absolute densities.

Output

The results can be output as a *Surfer for Windows* file (for both an external or generated reference file) or as an *ArcGIS* '.shp', *MapInfo* '.mif', *ArcGIS Spatial Analyst* 'asc', or ASCII grid 'grd' file (only if the reference file is generated by *CrimeStat*.) The output file is saved as IHB<root name> with the root name being provided by the user.

Example: Using the Interpolated Head-Bang to Visualize Houston Burglaries

The Houston burglary data set was, first, smoothed using the Head-Bang routine (Figure 11.3 above) and, second, interpolated to a grid using the Interpolated Head-Bang routine. The kernel chosen was the default normal distribution but with a fixed bandwidth of 1 mile. Figure 11.9 show the results of the interpolation.

To compare this to an interpolation of the original data, the number of burglaries in each zone was interpolated using the single kernel density routine. The kernel used was also the normal distribution with a fixed bandwidth of 1 mile. Figure 11.10 show the results of interpolating the raw burglary numbers.

A comparison of these two figures shows that they both capture the areas with the highest burglary density. However, the Interpolated Head-Bang produced fewer high density cells which, in turn, allowed the moderately high cells to stand out. For example, in southwest Houston, the Interpolated Head-Bang showed two small areas of moderately high density of burglaries whereas the raw interpolation merged these together.

Figure 11.9:

Interpolated Head-Bang Estimate of Burglaries in Houston: 2006

One Mile Bandwidth for Traffic Analysis Zones

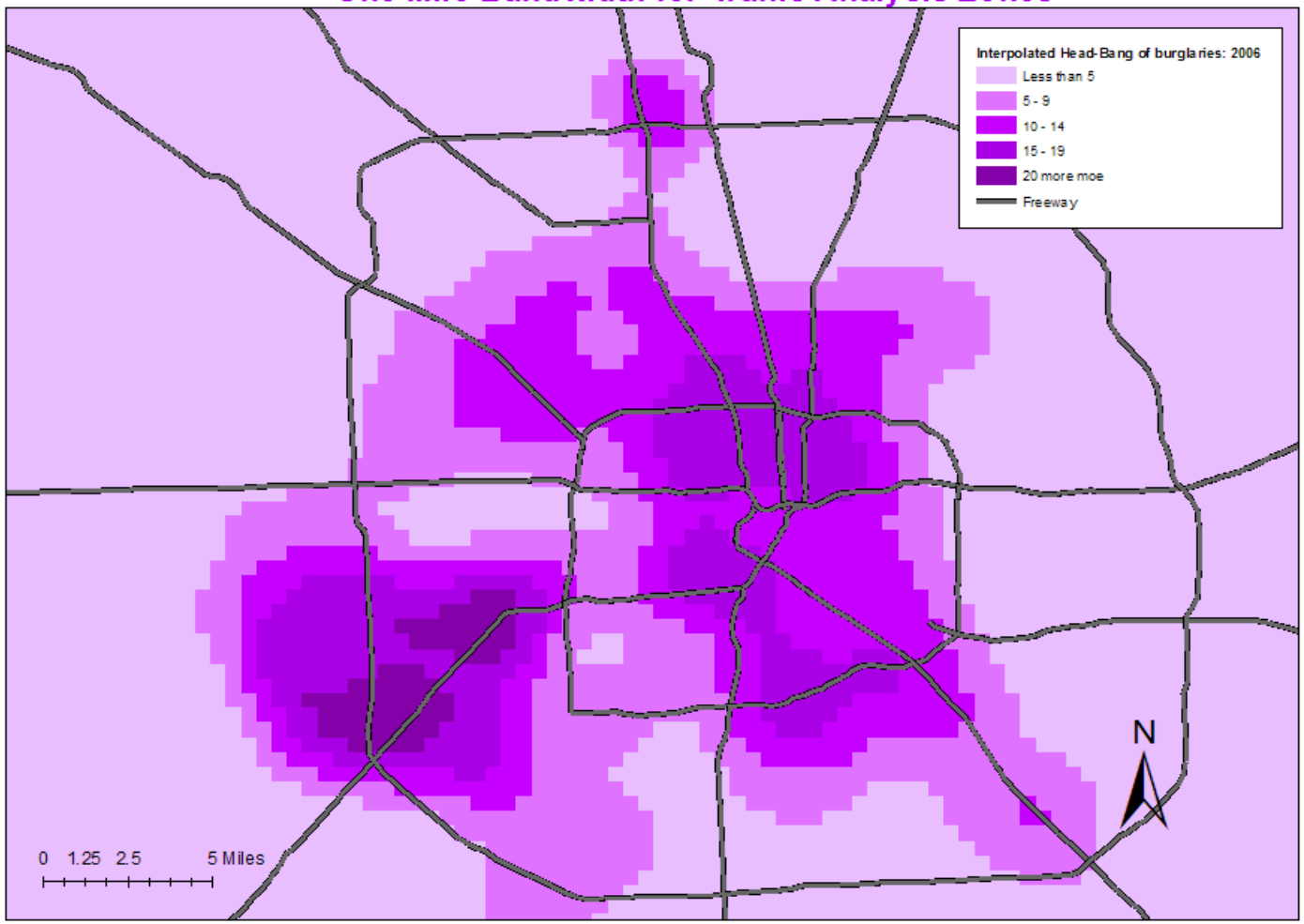
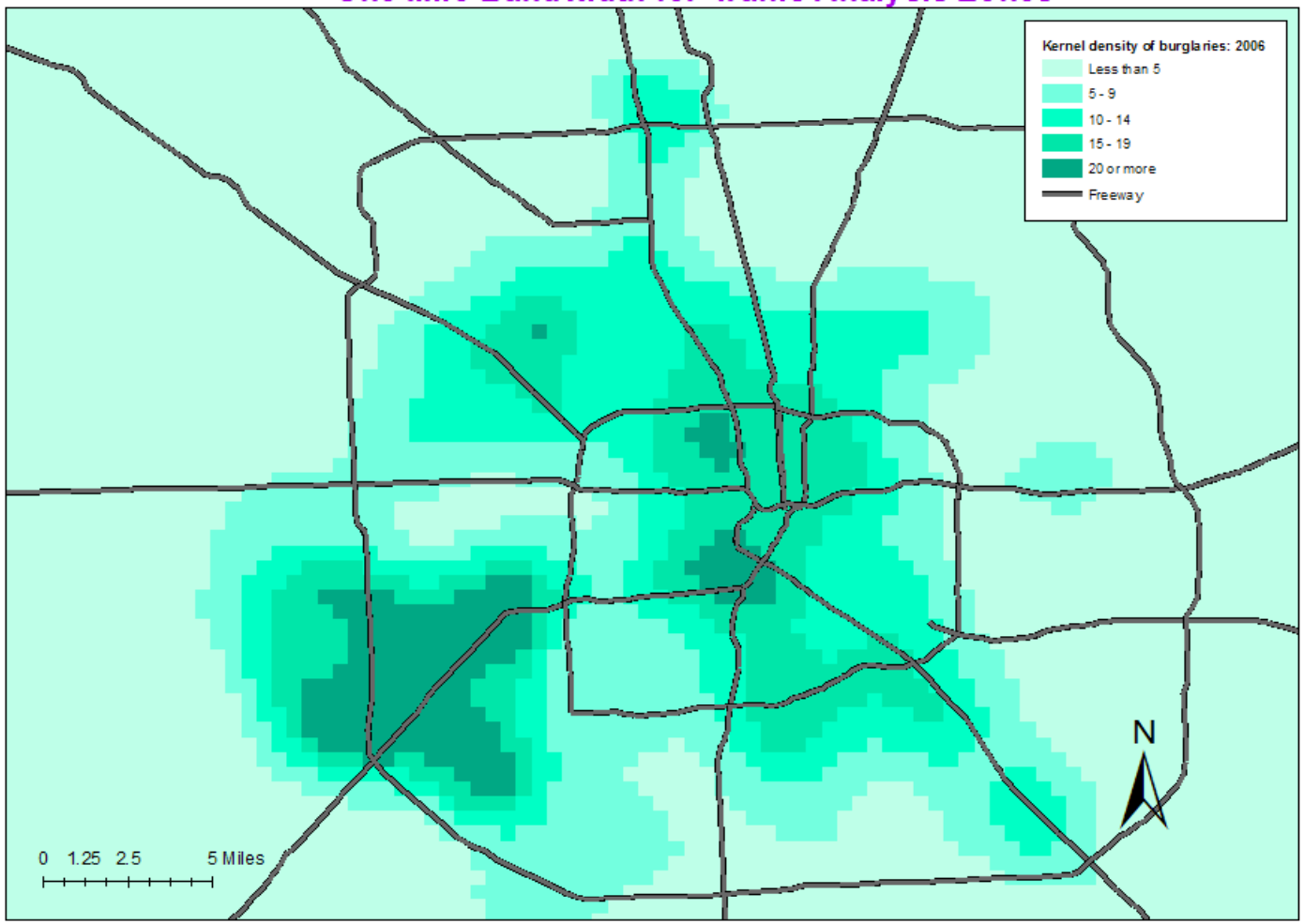


Figure 11.10:

Kernel Density Interpolation of Burglaries in Houston: 2006

One Mile Bandwidth for Traffic Analysis Zones



Advantages and Disadvantages of the Interpolated Head-Bang

The Interpolated Head-Bang routine has advantages and disadvantages over the regular Head-Bang. Its advantages are that it, like the Head-Bang, captures the strongest tendencies by eliminating 'peaks' and 'valleys'. But, it allows a smoother representation of the data. Zones are usually of unequal size with those in the center of a metropolitan area being much smaller than those in the periphery (the MAUP problem; see Wikipedia, 2012; Hipp, 2007; Wooldridge, 2002; Openshaw, 1984).). There is a visual distortion that occurs with large areas simply due to the larger area that they cover. The Head-bang will mute the effect of extreme low or high values in the periphery, but it will not eliminate the visual distortion that one sees in looking a map.

On the other hand, the interpolated Head-Bang routine does this by smoothing the data even more than the Head-Bang itself. There is a danger that it could over-smooth the data. The user has to determine whether the elimination of areas with very high or very low density values is real and not just due to small number of events.

For law enforcement applications, this may or may not be an advantage. Some hot spots, for example, are small areas where there are a many crime events. Smoothing the data may eliminate the visibility of these. On the other hand, large hot spots will generally survive the smoothing process because the number of events is large and will usually spread to adjacent grid cells. As usual, the user has to be aware of the advantages and disadvantages in order to decide whether a particular tool, such as the interpolated Head-Bang, is useful or not.

References

Fink, A. M. (1988). How to polish off Median Polish. *SIAM J. Sci. and Stat. Comput.*, 9(5), 932-940.

Hansen, K.M., Simonson, K. H. and Statistical Methodology and Applications Branch, NCI (2010). *Head-Bang PC Software (version 3.0)*. Surveillance Research, Canter Control and Population Sciences, National Cancer Institute. http://surveillance.cancer.gov/headbang/

Hansen, K.M., Simonson, K. H. and Statistical Methodology and Applications Branch, NCI (2010). *Head-Bang PC Software (version 3.0)*. Surveillance Research, Canter Control and Population Sciences, National Cancer Institute. http://surveillance.cancer.gov/headbang/

Hansen, K. M. (1991). Head-banging: robust smoothing in the plane, *IEEE Transactions on Geoscience and Remote Sensing*, 29, 369-378.

Hipp, J. R. (2007). Block, Tract, and Levels of Aggregation: Neighborhood Structure and Crime and Disorder as a Case in Point. *American Sociological Review* 72:659-680.

Mungiole, M., Pickle, L. W. & Simonson, K. H. (2002). Application of a weighted Head-Banging algorithm to Mortality data maps", *Statistics in Medicine*, 18, 3201-3209.

Mungiole, M. & Pickle, L. W. (1999). "Determining the optimal degree of smoothing using the weighted head-banging algorithm on mapped mortality data", In ASC '99 - Leading Survey & Statistical Computing into the New Millennium, Proceedings of the ASC International Conference, September. Available at http://srab.cancer.gov/headbang.

Openshaw, S. (1984). *The Modifiable Areal Unit Problem.* Norwich: Geo Books. ISBN 0-86094-134-5.

Pickle, L. W. & Su, Y. (2002). Within-State geographic patterns of health insurance coverage and health risk factors in the United States, *American Journal of Preventive Medicine*, 22 (2), 75-83.

Pickle, L. W., Mungiole, M., Jones, G. K., & White, A. A. (1996). *Atlas of United States Mortality*. National Center for Health Statistics: Hyattsville, MD.

Tukey, P. A. & Tukey, J. W. (1981). Graphical display of data sets in 3 or more dimensions, Barnett, V. (ed.), *Interpreting Multivariate Data*, Wiley, New York.

References (continued)

Wikipedia (2012). Modifiable Area Unit Problem. Wikipedia. http://en.wikipedia.org/wiki/Modifiable_areal_unit_problem. Accessed May 7, 2012.

Wooldridge, J. (2002). Examining the (Ir)Relevance of Aggregation Bias for Multilevel Studies of Neighborhoods and Crime with an Example Comparing Census Tracts to Official Neighborhoods in Cincinnati. *Criminology* 40:681-710.

Chapter 12:

Space-Time Analysis

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 12:
Space-Time Analysis

In this chapter, we discuss three techniques that are used to analyze the relationship between space and time. Up to this point, we have analyzed the distribution of incidents irrespective of the order in which they appeared or in which the time frame in which they appeared. The only temporal analysis that was conducted was in Chapter 4 where several spatial description indices, including the standard deviational ellipse, were compared for different time periods.

As police departments usually know, however, the spatial patterning of incidents does not occur uniformly throughout the year, but instead are often clustered together during short time periods. At certain times, a rash of incidents will occur in certain neighborhoods and the police often have to respond quickly to these events. In other words, there is both clustering in time as well clustering in space. This area of research has been developed mostly in the field of epidemiology (Knox, 1963, 1988; Mantel, 1967; Mantel and Bailar, 1970; Besag and Newell, 1991; Kulldorf and Nargawalla, 1995; Bailey and Gattrell, 1995). However, most of these techniques are applicable to crime analysis and criminal justice research as well.

CrimeStat includes four space-time techniques: the Knox index, the Mantel index, the Spatial-temporal moving average, and Correlated Walk Analysis. Figure 12.1 shows the Space-Time Analysis screen.

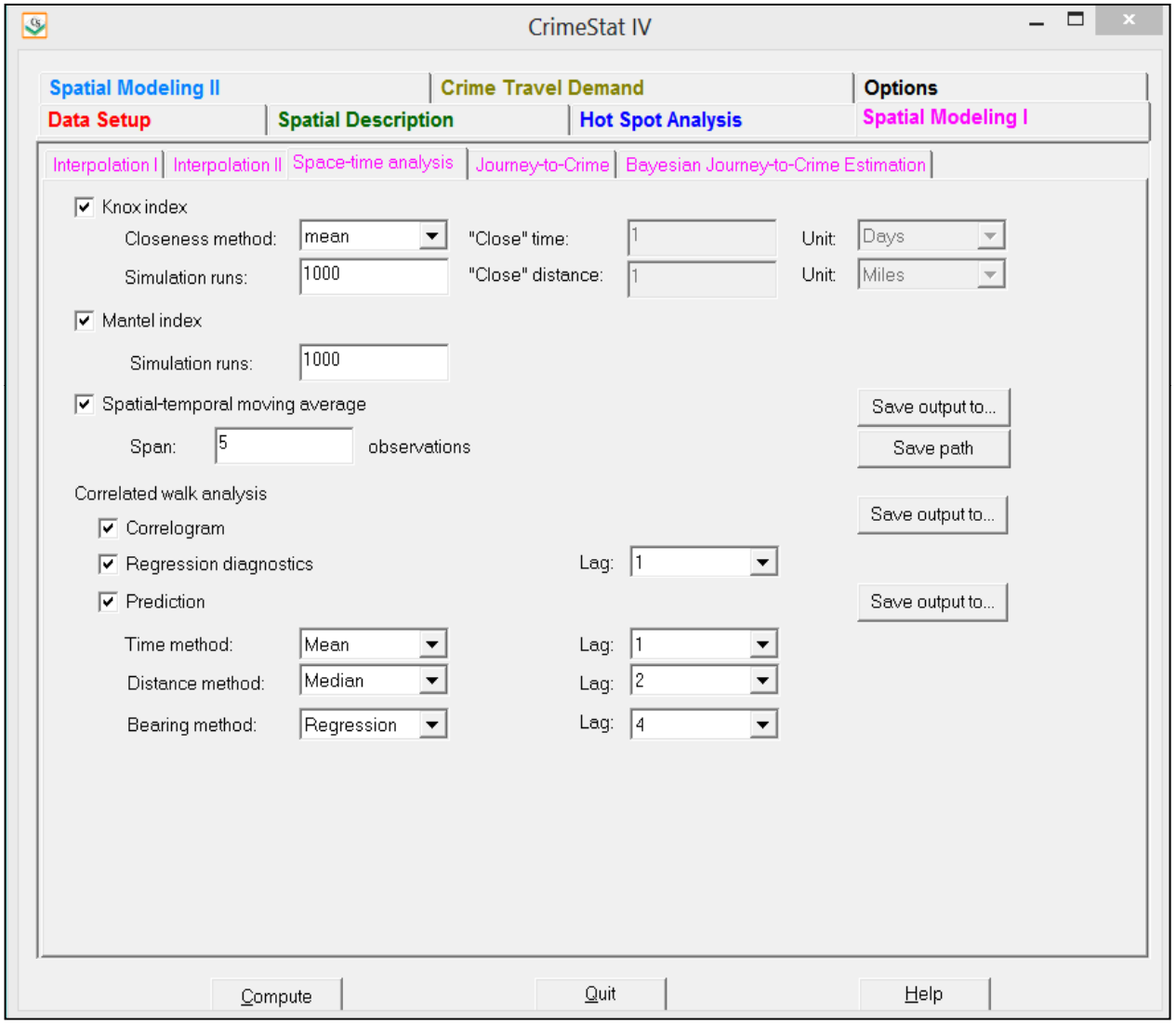
Measurement of Time in *CrimeStat*

Time can be defined as hours, days, weeks, months, or years. The default is days. However, please note that for any of these techniques, in *CrimeStat* time must be measured as an ***integer*** (or *real*) variable, as mentioned in Chapter 3. Time **cannot** be defined by a formatted date code (e.g., 11/06/01, July 30, 2002). Each of the three space-time routines require that time be an integer or real variable (e.g., 1, 2, 34527, 2.8). If given formatted dates, the routines will calculate an answer, but the result will not be correct.

If the time unit is days, a simple transformation is to use the number of days since January 1, 1900. Most spreadsheet and data base programs usually assign an integer number from this reference point. For example, November 12, 2001 has the integer value of 37207 while January 30, 2002 has the integer value of 37286. These are the number of days since January 1, 1900. Any spreadsheet program (e.g., Excel) can convert a date format into a real number with the Value function. Also, any arbitrary numbering system will work (e.g., 1, 2, 3).

Figure 12.1:

Space-Time Analysis Screen



Space-Time Interaction

There are different types of interaction that could occur between space and time. Four distinctions can be made. First, there could be ***spatial clustering all the time***. Certain communities are prone to certain events. For example, robberies often are concentrated in particular locations as are vehicle thefts. The hot spot methods that were discussed in chapters 7, 8 and 9 are useful for identifying these concentrations. In this case, there is no space-time interaction since the clustering occurs all the time.

Second, there could be ***spatial clustering within a specific time period***. Hot spots could occur during certain time periods. For example, motor vehicle crashes tend to occur with much higher frequencies in the late afternoon and early evening, often as a by-product of congestion on the roads. Crash hot spots will tend to appear at certain times because of the congestion. At most other times, the concentration does not occur because the congestion levels are lower.

Third, there could be ***episodic space-time clustering***. A number of events could occur within a short time period within a concentrated area. This type of effect is very common with motor vehicle thefts. A car thief gang may decide to attack a particular neighborhood. After a binge of car thefts, they move on to another neighborhood. In this instance, there are a number of theft incidents that are occurring within a limited period in a limited location. The cluster moves from one location to another. In this case, there is an interaction between space and time in that spatial hot spots appear at particular times, but are temporary. The ability to detect this type of shift is very important to police departments since it affects their ability to respond.

Fourth, there could be ***periodic space-time interaction*** in which the relationship between space and time occurs at certain times but not others and is somewhat predictable. The interaction could be concentrated, as in the spatial clustering mentioned above, or it could follow a more complex pattern. For example, there could be a diffusion of drug sales from a central location to a more dispersed area. Whereas initially, the drug dealing is concentrated in a few locations, it starts to diffuse to other areas. However, the diffusion may occur at different times of the year (e.g., Christmas and New Years). Alternatively, vehicle thefts may shift towards seaside communities during the summer months when the number of vacationers increases and then shift back to the city at other times of the year. We saw an example of this in Chapter 4 where the ellipse of motor vehicle thefts shifted between June and July to the communities along the Chesapeake River near Baltimore. This type of diffusion is not clustering *per se*, in that it may be spread over a very large coastline. But it is a distinct space-time interaction.

The importance of these distinctions is that many space-time tests that exist only measure gross space-time interaction, rather than space-time clustering. For example, the Knox and Mantel tests that are discussed below test for spatial interaction. The interaction could be the result of spatial clustering, but does not necessarily have to be. The interaction could occur in a very complex way that would not easily lend itself to more focused intervention by the police. Still, the ability to identify the interaction is an important first step in planning an intervention strategy.

Knox Index

The Knox Index is a simple comparison of the relationship between incidents in terms of distance (space) and time (Knox, 1963; 1964). That is, each individual pair is compared in terms of distance and in terms of time interval. Since each pair of points is being compared, there are N*(N-1)/2 pairs. The distance between points is divided into two groups - Close in distance and Not close in distance, and the time interval between points is also divided into two groups - Close in time and Not close in time. The definitions of 'close' and 'Not close' are left to the user.

A simple 2 x 2 table is produced that compares closeness in distance with closeness in time. The number of pairs that fall in each of the four cells is compared (Table 12.1).

Table 12.1:
Logical Structure of Knox Index

	Close in time	**Not close in time**	**TOTAL**
Close in distance	O_1	O_2	S_1
Not close in distance	O_3	O_4	S_2
TOTAL	S_3	S_4	N

where $N = O_1 + O_2 + O_3 + O_4$

$S_1 = O_1 + O_2$

$S_2 = O_3 + O_4$

$S_3 = O_1 + O_3$

$S_4 = O_2 + O_4$

The actual number of pairs that falls into each of the four cells are then compared to the expected number if there was no relationship between closeness in distance and closeness in time. The expected number of pairs in each cell under strict independence between distance and the time interval is obtained by the cross-products of the columns and row totals (Table 12.2).

Table 12.2:
Expected Frequencies for Knox Index

	Close in time	**Not close in time**
Close in distance	E_1	E_2
Not close in distance	E_3	E_4

where $\mathbf{E_1 = S_1 * S_3 / N}$
$\mathbf{E_2 = S_1 * S_4 / N}$
$\mathbf{E_3 = S_2 * S_3 / N}$
$\mathbf{E_4 = S_2 * S_4 / N}$

The difference between the actual (observed) number of pairs in each cell and the expected number is measured with a Chi-square statistic (equation 12.1):

$$\chi^2 = \sum_{i=1}^{4} \frac{(O_i - E_i)^2}{E_i} \qquad (12.1)$$

Monte Carlo Simulation of Critical Chi-square Values

Unfortunately, the usual probability test associated with the Chi-square statistic cannot be applied since the observations are not independent. The interaction between space and time tends to be compounded when calculating the Chi-square statistic. For example, we have noticed that the Chi-square statistic tends to get larger with increasing sample size, a condition that would normally not be true with independent observations.

To handle the issue of interdependency, there is a Monte Carlo simulation of the Chi-square value for the Knox Index under spatial randomness (Dwass, 1957; Barnard, 1963). This is known as *randomization* since it assumes that any location within the study area could be available for an event. If the user selects a simulation, the routine randomly selects M pairs of a

distance and a time interval where M is the number of pairs in the data set $M = N\frac{(N-1)}{2}$ and calculates the Knox Index and the Chi-square test. Each pair of a distance and a time interval are selected from the range between the minimum and maximum values for distance and time interval in the data set using a uniform random generator.

An alternative simulation is to assume that the spatial location of the events are fixed and cannot change. This would occur, for example, if one was measuring a unique set of individuals who do not change or certain neighborhoods only or even applying the statistic to grouped data where the groups do not change In this case, a *permutation* simulation would be appropriate, similar to the simulations used in the spatial autocorrelation indices (see Chapters 5 and 9). For this version of *CrimeStat*, we only use a randomization simulation.

Output of simulation

The randomization simulation is repeated *K* times, where *K* is specified by the user. Usually, it is wise to run the simulation 1000 or more times. The output includes:

1. The sample size
2. The number of pairs
3. The calculated chi-square value of the Knox Index from the data
4. The minimum chi-square value of the Knox Index from the simulation
5. The maximum chi-square value of the Knox Index from the simulation
6. Ten percentiles from the simulation:

    a. 0.5%
    b. 1%
    c. 2.5%
    d. 5%
    e. 10%
    f. 90%
    g. 95%
    h. 97.5%
    i. 99%
    j. 99.5%

Methods for Dividing Distance and Time

In the *CrimeStat* implementation of the Knox Index, the user can divide distance and time interval based on the three criteria:

1. The mean (mean distance and mean time interval). This is the default.

2. The median (median distance and median time interval)

3. User-defined criterion for distance and time separately.

There are advantages to each of these methods. The mean is the center of the distribution; it denotes a balance point. The median will divide both distance and time interval into approximately equal numbers of pairs. The division is approximate since the data may not easily divide into two equal numbered groups. A user-defined criterion can fit a particular need of an analyst. For example, a police department may only be interested in incidents that occur within two miles of each other within a one week period. Those criteria would be the basis for dividing the sample into 'Close' and 'Not close' distance and time intervals.

Example of the Knox Index

For an example, vehicle thefts in Baltimore County for 1996 were taken. There were 1855 vehicle thefts for which a date was recorded in the data base. The data base was further broken down into twelve separate monthly subsets. Using the median as the criterion for dividing the data into 'Close' and 'Not close' for both distance and time interval, the Knox Index was calculated for the entire set of 1855 incidents. Then, using the median distance for the entire year but a month-specific median time interval, the Knox Index was calculated for each of the twelve months. Table 12.3 presents the Chi-square values and their pseudo-significance levels.

To produce a better test of the significance of the results, 1000 random simulations were calculated for the vehicle theft for the entire year. Table 12.3 below shows the results. Because an extreme value could be obtained by chance with a random distribution, reasonable cut-off points are usually selected from the simulation. In this case, we want a cut-off point that approximates a 5% significance level. Since the Knox Index is a one-tailed test (i.e., only a high chi-square value is indicative of spatial interaction), we adopt an upper threshold of the 95 percentile. In other words, only if the observed Chi-square test for the Knox Index is larger than the 95^{th} percentile will the null hypothesis of a random distribution between space and time be rejected.

Table 12.3:

Knox Index for Baltimore County Vehicle Thefts

Median Split

(N = 1,855 with 1,719,585 comparisons)

Month	Actual Chi-square	95 Percentile Simulation Chi-square	Approx. p
January	0.26	6.95	n.s.
February	0.00	6.61	n.s.
March	0.00	6.86	n.s.
April	0.50	6.56	n.s.
May	1.04	7.25	n.s.
June	0.01	6.02	n.s.
July	9.96	9.05	.05
August	5.91	5.55	.05
September	0.27	5.41	n.s.
October	3.33	6.43	n.s.
November	10.79	8.91	.01
December	0.00	6.87	n.s.
All of 1996	**8.69**	**41.89**	**n.s.**

For the entire year, there was not a significant clustering between space and time. Approximately, 26.7% of the incidents were both close in distance (i.e., closer than the median distance between pairs of incidents) and close in time (i.e., closer than the median time interval between pairs of incidents). However, when individual months are examined, three show significant relationships: July, August, and November. During these months, there is an interaction between space and time. Typically, this indicates that, during those months, incidents that cluster together spatially tend also to cluster together temporally. However, it could be the opposite (i.e., events that cluster together temporally tend to be far apart spatially).

The next step would to identify whether there are particular clusters that occur within a short time period. Using one of the 'hot spot' analysis methods discussed in Chapters 7 and 8, an analyst could take the events for the three months and try to identify whether there is spatial clustering during those three months that does not normally occur. We did not do that here, but the point is that the Knox Index is useful to identify *when* there is spatial clustering.

Problems with the Knox Index

The Knox Index is a simple measure of space-time clustering. But there are potential problems with it. First, because it is only a 2 x 2 table, different results can be obtained by varying the cut-off points for distance or time. For example, using the mean as the cut-off, the overall Chi-square statistic for all vehicle thefts was 8.67, reasonably close. However, when a cut-off point for distance of 1000 meters and a cut-off point for time of 80 days was used, the Chi-square statistic dropped to 3.16. In other words, the Knox Index will produce different results for different cut-off points.

A second problem has to do with the interpretation. As with any Chi-square test, differences between the observed and expected frequencies could occur in any cell or any combination of cells. Finding a significant relationship does not automatically mean that events that were close in distance were also close in time; it could have been the opposite relationship. However, a simple inspection of the table can indicate whether the relationship is as expected or not. In the above example, all the significant relationships showed a higher proportion of events that were both close in distance and close in time.

Mantel Index

The Mantel Index resolves some of the problems of the Knox Index. Essentially, it is a correlation between distance and time interval for pairs of incidents (Mantel, 1967). More formally, it is a general test for the correlation between two *dissimilarity* matrices that summarizes comparisons between pairs of points (Mantel and Bailar, 1970). It is based on a simple cross-product of two interval variables (e.g., distance and time interval):

$$T = \sum_{i=1}^{N} \sum_{j \neq i=1}^{N-1} \left(X_{ij} - \bar{X}\right)\left(Y_{ij} - \bar{Y}\right) \qquad (12.2)$$

where X_{ij} is an index of similarity between two observations, i and j, for one variable (e.g., distance) while Y_{ij} is an index of similarity between the same two observations, i and j, for another variable (e.g., time interval). The comparison is between two observations and does not include a comparison of an observation with itself. Hence, *j* is incremented up to *N-1*.

The cross-product is then normalized by dividing each deviation by its standard deviation:

$$r = \sum_{i=1}^{N} \sum_{j \neq i=1}^{N-1} \frac{(X_{ij} - \bar{X})}{S_X} \frac{(Y_{ij} - \bar{Y})}{S_Y} \qquad (12.3)$$

$$= \frac{1}{(N-1)} \sum_{i=1}^{N} \sum_{i \neq j=1}^{N-1} Z_X Z_y \quad (12.4)$$

where X_{ij} and Y_{ij} are the original variables for comparing two observations, i and j, and Z_x and Z_y are the normalized variables.

Monte Carlo Simulation of Confidence Intervals

Even though the Mantel Index is a Pearson product-moment correlation between distance and time interval, the observations are not independent and, in fact, are highly interdependent. That is, the correlation is between observations for the same distance and time variable rather than between the two variables by themselves. Thus, the values of the Mantel *r* tend to be very low.

Further, the usual significance test for a correlation coefficient is not appropriate. Instead, the Mantel routine offers a simulation of the confidence intervals around the index. If the user selects a simulation, the routine randomly selects M pairs of a distance and a time interval where M is the number of pairs in the data set $M = N\frac{(N-1)}{2}$ and calculates the Mantel Index. Each pair of a distance and a time interval are selected from the range between the minimum and maximum values for distance and time interval in the data set using a uniform random generator. As with the Knox Index simulation discussed above, the simulation is a randomization where every location within the study area is possible for an event to occur, compared to a permutation simulation where the spatial locations are fixed. For this version of *CrimeStat*, we only use a randomization simulation.

The random simulation is repeated *K* times, where *K* is specified by the user. Usually, it is wise to run the simulation 1000 or more times. The output includes:

1. The sample size
2. The number of pairs
3. The calculated Mantel Index from the data
4. The minimum Mantel value from the simulation
5. The maximum Mantel value from the simulation
6. Ten percentiles from the simulation:
    a. 0.5%
    b. 1%
    c. 2.5%
    d. 5%
    e. 10%
    f. 90%

g. 95%
h. 97.5%
i. 99%
j. 99.5%

Example of the Mantel Index

In *CrimeStat*, the Mantel Index routine calculates the correlation between distance and time interval. To illustrate, Table 12.4 examines the Mantel correlation for the 1996 vehicle thefts in Baltimore County that was illustrated above. As seen, the correlations are all low. However, as with the Knox Index, July, August and November produce relatively higher correlations. As mentioned above, the correlations tend to be very low because the test is between observations for the same variables, rather than between variables.

Table 12.4:
Mantel Index for Baltimore County Vehicle Thefts
Median Split
(N = 1,855 and 1,719,585 Comparisons)

Simulation Month	Simulation r	Approx. 2.5%	97.5%	p-level
January	-.0047	-0.033	0.033	n.s.
February	-.0023	-0.037	0.042	n.s.
March	-.0245	-0.032	0.039	n.s.
April	0.0077	-0.040	0.041	n.s.
May	0.0018	-0.038	0.043	n.s.
June	0.0043	-0.035	0.041	n.s.
July	0.0348	-0.034	0.033	.025
August	0.0544	-0.034	0.035	.01
September	0.0013	-0.044	0.046	n.s.
October	0.0409	-0.037	0.043	n.s.
November	0.0630	-0.042	0.040	.001
December	0.0086	-0.035	0.038	n.s.
All of 1996	**0.0015**	**-0.009**	**0.010**	**n.s.**

To test whether these correlations are significant or not, 1000 random simulations were calculated for each month using the same sample size as the monthly vehicle theft totals. Table 12.4 above shows the results. Because an extreme value could be obtained by chance with a random distribution, reasonable cut-off points are usually selected from the simulation. In this

case, we want cut-off points that approximate a 5% significance level. Since the Mantel Index is a two-tailed test (i.e., one could just as easily get dispersion between space and time as clustering), we adopt a lower threshold of the 2.5 percentile and an upper threshold of 97.5 percentile. Combined, the two cut-off points ensure that approximately 5% of the cases will be either lower than the lower threshold or higher than the upper threshold under random conditions.[1] In other words, only if the observed Mantel Index is smaller than the lower threshold or larger than the upper threshold will the null hypothesis of a random distribution between space and time be rejected.

In Table 12.4, for the entire year, the observed Mantel Index (correlation between space and time) was 0.0015. The 2.5^{th} percentile was -.009 and the 97.5^{th} percentile was 0.01. Since the observed value is between these two cut-off points, we cannot reject the null hypothesis of no relationship between space and time. However, for the individual months, again, July, August and November have correlations above the upper cut-off threshold. Thus, for those three months *only*, the amount of space-time clustering in the vehicle theft data is most likely greater than what would be expected on the basis of a chance distribution. One would, then, have to explore the data further to find out where those vehicle thefts were occurring, using one of the hot spot routines in Chapters 7 or 8.

Limitations of the Mantel Index

The Mantel Index is a useful measure of the relationship between space and time. But it does have limitations. First, because it is a Pearson-type correlation coefficient, it is prone to the same types of problems that befall correlations. Extreme values of either space or time could distort the relationship, either positively, if there are one or two observations that are extreme in *both* distance in time interval, or negatively, if there are only one or two observations that are extreme in *either* distance or in time interval.

Second, because the test is a comparison of all pairs of observations, the correlations tend to be very small, as noted above. This makes it less intuitive as a measure than a traditional correlation coefficient that varies between -1 and +1 and in which high values are expected. For most analysts, it is not very intuitive to have an index where 0.05 is a high value. This does not fault the statistic as much make it a little non-intuitive for users.

[1] It would be possible to make a one-tailed test with the simulation. For example, if one is only interested in the degree of clustering, one could adopt the 95 percentile as the threshold. An observed Mantel value that was lower than this threshold would be consistent with the null hypothesis.

Third, as with any correlation coefficient, the sample size needs to be fairly large to produce a stable estimate. In the above, example, one could further break down monthly vehicle thefts by week or, even, day. However, the number of cases will decrease considerably. In the above example, with 1,855 vehicle thefts over a year, the weekly average would be around 36, which is a small sample. Intuitively, a crime analyst wants to know when space-time clustering is occurring and a short time frame is critical for detection. A week would be the largest time interval that would be useful.

However, as the sample size gets small, the index becomes unstable. The sample size makes the index volatile. While the Monte Carlo simulation will adjust for the sample size, the range of the cut-off thresholds will vary considerably from one week to another with small sample sizes. The analyst will have to run the simulation a large number of times to adjust for the varying sample sizes. Also, the shortened time frame allows fewer distinctions in time. If one takes a very narrow time frame (e.g., a day), there will be virtually no time differences observed because there is not enough data to produce reliable estimates.

One way to get around this is to have a moving average where the time frame is adjusted to fit a constant number of days (e.g., a 14 day moving average). The advantage is that the sample size tends to remain constant; one could therefore reduce the number of recalculations of the cut-off thresholds since they would not vary much from one day to another. To make this work, however, the data base must be set up to produce the appropriate number of incidents for a moving average analysis.

Nevertheless, the Mantel Index remains a useful tool for analysts. It is still widely used for space-time analysis and it has been generalized to many other types of dissimilarity analyses than just space and time. If used carefully, the index can be a powerful tool for detection of clusters that are also concentrated in time.

Spatial-Temporal Moving Average

The Spatial-Temporal Moving Average is a simple statistic. It is the moving mean center of *M* observations where *M* is a sub-set of the total sample, *N*. By 'moving', the observations are sequenced in order of occurrence. Hence, there is a time dimension associated with the sequence. The *M* observations are called the *span* and the default span is 5 observations. The span is centered on each observation so that there are an equal number on both sides. Because there are no data points prior to the first event and after the last event, the first few mean centers will have fewer observations than the rest of the sequence. For example, with a span of 5, the first and last mean centers will have only three observations, the second and next-to-last will have 4 observations, while all others will have 5. In general, it is a good idea to choose an odd

number since the middle of the span will be centered on a real observation rather than having to fall between two in the case of an even span.

Though simple, the Spatial-Temporal Moving Average is very useful for detecting changes in behavior by serial offenders. In the next chapter, we will examine journey-to-crime models that attempts to estimate the likely origin location of a serial offender based on the distribution of incidents committed by the offender. However, if the serial offender has either moved residences or else moved the field of operation, then the technique will error because it is assuming a stable field of operations when, in fact, it is not. The moving average can suggest whether the offender's behavior is stable or not.

As an example, figure 12.2 below shows the Spatial-Temporal Moving Average of an offender who committed 12 offenses before being arrested. The individual committed eight thefts from vehicles, two thefts from stores, one residential burglary and one highway robbery. The actual incidents are shown in red circles with the sequence number displayed. The moving average is shown in blue squares with the sequence number displayed. The path of the moving average is shown as a green line.

As seen, there is a definite shift in the field of operation by this offender. The mean center moved about a mile during this period but the consistency of the trend suggests that something fundamental changed by the offender, either the person moved residences or the nature of the committed crimes changed. In using the Journey-to-crime tools, an analyst would probably want to focus on the latter events since these are more geographically circumscribed. Notice that the last two moving averages are relatively close to the actual residence location of the offender when arrested (less than three-quarters of a mile away).

In short, the Spatial-Temporal Moving Average simply plots the changes in the mean center of the span and is useful for detecting changes in the behavior pattern of serial offenders.

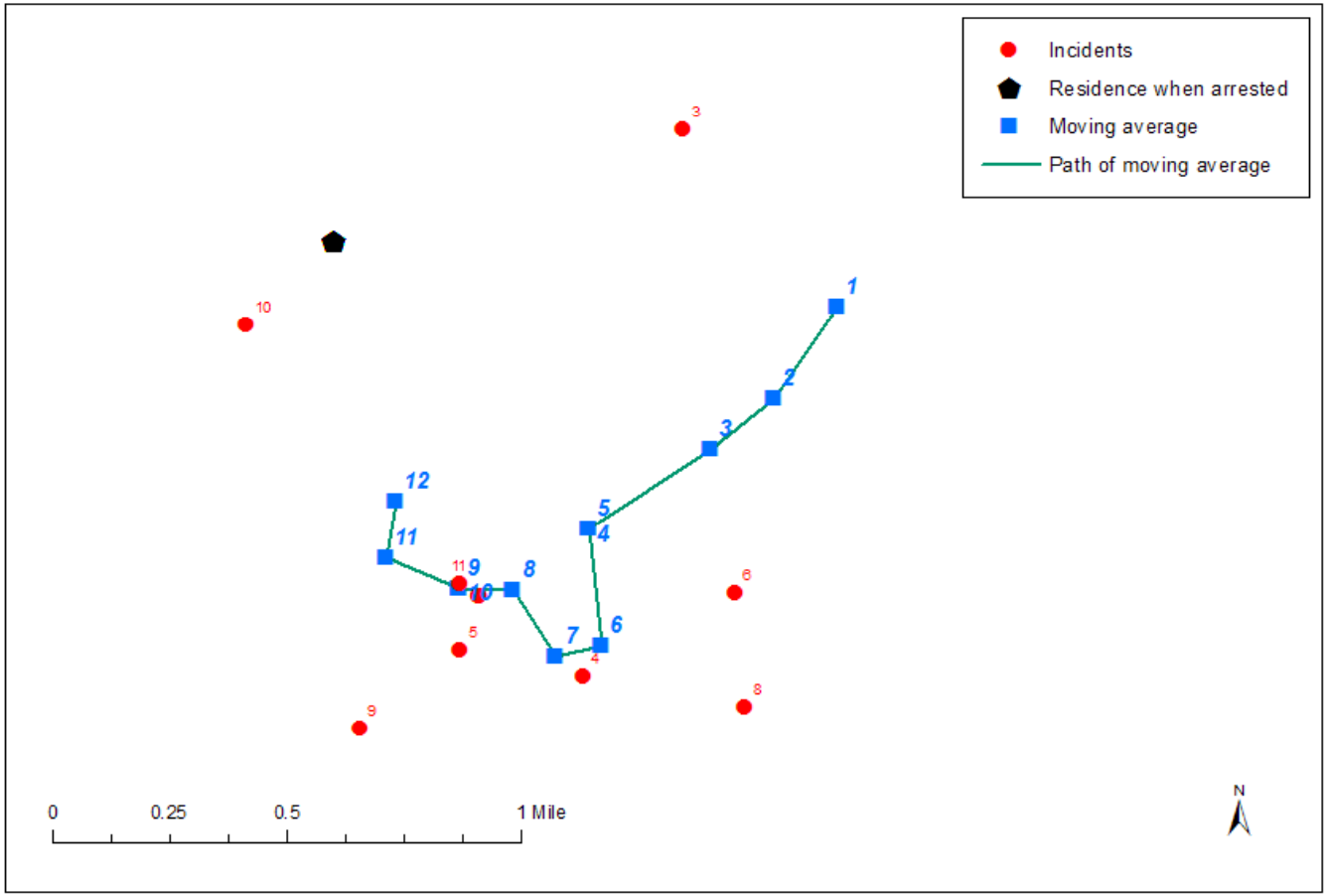
Correlated Walk Analysis

Correlated Walk Analysis (CWA) is a tool that is aimed at analyzing the spatial and temporal *sequencing* of incidents committed by a single serial offender. In this sense, it is the 'flip side' of Journey to crime analysis (see Chapter 13). Whereas journey to crime analysis makes guesses about the likely origin location for a serial offender, based on the spatial distribution of the incidents committed by the offender, the CWA routine makes guesses about the time and location of a next event, based on both the spatial distribution of the incidents and the temporal sequencing of them. In effect, it is a Spatial-Temporal Moving Average with a prediction of a next event.

Figure 12.2:

Moving Path of Serial Offender:

Sequence of 12 Crimes



The statistical origin of CWA is Random Walk Theory. Random Walk Theory has been developed by physicists to explain the distribution of molecules in a rapidly changing environment (e.g., the movements of a particle in a gas which is diffusing - Brownian movement). Sometimes called a 'drunkard's walk', the theory starts with the premise that movement is random in all directions. From an arbitrary starting point, a particle (or person) moves in any direction in a series of steps. The direction of each step is independent of the previous steps. After each step, a random decision is made and the person moves in a random direction. This process is repeated *ad infinitum* until an arbitrary stopping point is selected (i.e., the observer quits looking). It has been shown mathematically that all one and two dimensional random walks must eventually return to their original starting point (Spitzer, 1963; Henderson, Renshaw, & Ford, 1983; see endnote *i*). This is called a *recurrent random walk*. On the other hand, independent random walks in more than two dimensions are not necessarily recurrent, a state called *transient random walk*.

Figure 12.3 illustrates a random walk of 2000 steps. For a large number of steps in a two-dimensional walk, the likely distance of a person (or particle) from the starting point is:

$$E(d) = d_{RMS}\sqrt{N} \tag{12.4}$$

where d_{RMS} is the *root mean square* of distance.

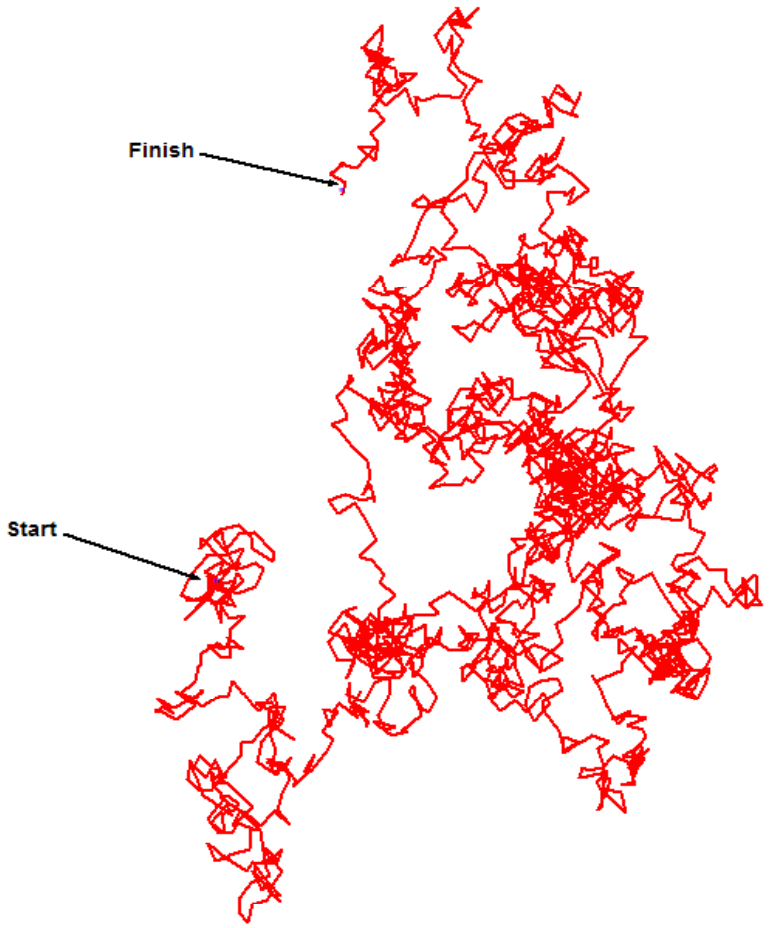
There are a number of different types of random walks. The simplest is a movement of uniform distance only along a grid cell (i.e., a Manhattan geometry). The person can only move North, South, East or West for a unit distance of 1. A more complex random walk allows angular distances and an even more complex random walk allows varying distances (e.g., normally distributed random distances, uniformly random distances). The walk in Figure 12.3 was of this latter type. X and Y values were selected randomly from a range of -1 to +1 using a uniform random number generator. For a conceptual understanding of Random Walk Theory, see Chaitin (1990) and, for a mathematical treatment, see Spitzer (1976). Malkiel (1999) applied the concepts of Random Walk Theory to stock price fluctuations in a book that has now become a classic.

Henderson, Renshaw and Ford (1984) have introduced the concept of a *correlated random walk*. In a correlated random walk, momentum is maintained. If a person is moving in a certain direction, they are more likely to continue in that direction than to reverse direction or travel orthogonally. In other words, at any one decision point, the probabilities of traveling in any direction are not equal; the same direction has a higher probability than an orthogonal change (i.e., turning 90 degrees) and those, in turn, have a higher probability than completely reversing direction.

Figure 12.3:

A Random Walk

2000 Random Steps of -1.0 to +1.0 in X and Y Direction



By implication, the same is true for distance and distance. A longer step than average is likely to be followed by another longer step than average while a shorter step than average is likely to be followed by another short step. Similarly, there is consistency in the time interval between events; a short interval is also likely to be followed by a short interval. In other words, a correlated random walk is a random walk with momentum (Chen & Renshaw, 1992; 1994). These authors have applied the theory to the analysis of the branching of tree roots (Henderson, Ford, Renshaw, & Deans, 1983; Renshaw, 1985).

Correlated Walk Analysis Routine

Correlated Walk Analysis is a set of tools that can help an analyst understand the sequencing of sequential events in terms of time interval, distance and direction. In *CrimeStat*, there are three CWA routines. The first two help the analyst understand whether there are *repeating* patterns in time, distance or direction while the last routine allows the analyst to make a guess about the next likely event, when it will occur and where it will occur. The three routines are:

1. CWA - Correlogram
2. CWA - Diagnostics
3. CWA - Prediction

CWA - Correlogram

The CWA - *Correlogram* routine calculates the correlation in time interval, distance, and bearing (direction) between events. It does this through *lags*. A lag is a separation in the intervals between events. The difference between the first and second event is the first interval. The difference between the second and third events is the second interval. The difference between the third and fourth events is the third interval, and so forth. For each successive interval, there is a time difference; there is a distance and there is a direction. One could extend this to all the intervals, comparing each interval with the next one; that is, we compare the first interval with the second, the second interval with the third, the third interval with the fourth, and so on until the sample is complete. When comparing successive intervals, this is called a *lag of 1*. It is important to keep in mind the distinction between an event (e.g., an incident) and an interval. It takes two events to create an interval. Thus, for a lag of 1, there are M=N-1 intervals where *M* is the number of intervals and *N* is the number of events (e.g., for 3 incidents, there are 2 intervals).

A lag of two compares every other event. Thus, the first interval is compared to the third interval; the second interval is compared to the fourth; the third interval is compared to the fifth; and so on until there are no more intervals left in the sample. Again, the comparison is for time

difference, distance, and direction separately. We can extend this logic to a lag of 3 (every third event), a lag of 4 (every fourth event), and so forth.

The CWA - Correlogram routine calculates the Pearson Product-Moment correlation coefficient between successive events. For a lag of 1, it compares successive events and correlates the time interval, distance, and bearing separately for these successive events. For a lag of 2, it compares every other event and correlates the time interval, distance, and bearing separately for these successive events. The routine does this until it reaches a maximum of 7 lags (i.e., every seventh event). However, if the sample size is very small, it may not be able to calculate all lags. It will require 12 incidents (events) to calculate all seven lags since it requires at least four observations per lag (i.e., N - L - 4 where N is the number of events and L is the maximum number of lags calculated).

Adjusted Correlogram

The Correlogram calculates the raw Pearson correlation coefficient between intervals by lag for time, distance, and bearing. One of the problems that may appear, especially with small samples, is that the correlation with higher-order lags are very high, either positive or negative. There are probably two reasons for this. For one thing, with each lag, the sample size decreases by one; with a very small sample size, correlations can become very volatile, jumping from positive to negative, and from low to high. Another reason is that periodicity in the data set is compounded with higher-order lags in the form of 'echos'. For example, if a lag of 2 is high, then a lag of 4 will also be somewhat high since there is a compounding of the lag 2 effect. When combined with a small sample size, it is not uncommon to have higher-order lags with very high correlations, sometimes approaching +/- 1.0. The user must be careful in selecting a higher-order lag because there is an apparent effect that may be due to the above reasons, rather than any real predictability. One of the key signs for spurious higher-order effect is a sudden jump in the strength of the correlation from one lag to the next (though sometimes a high higher-order lag can be real; see examples below).

To minimize these effects, the output also includes an adjusted correlogram that adjusts for the loss of degrees of freedom. The formula is:

$$A = \frac{M-L-1}{M-1} \tag{12.5}$$

where M is the number of intervals (N-1) and L is the number of lags. For example, for a sample size of 13, there will be 12 intervals (M). For a lag of 1, the adjustment will be:

$$A = \frac{12-1-1}{12-1} = \frac{10}{11} = 0.909 \quad (12.6)$$

The effect of the adjustment is to reduce the correlation for higher-order lags. It will not completely eliminate the effect, but it should help minimize spurious effects. As will be shown below, however, sometimes high correlations for higher-order lags are real.

CWA - Correlogram Output

The CWA - Correlogram routine outputs 10 parameters:

1. The sample size (number of events);
2. Number of intervals;
3. Information on the units of time, distance, and bearing;
4. Final distance to origin in meters (distance between last and first event);
5. Expected random walk distance from origin (if sequence was strictly random);
6. Drift (the ratio of actual distance from origin to expected random walk distance);
7. Final bearing from origin (direction between last event and first event);
8. Expected random walk bearing. Defined as 0 because there is no expected direction.
9. Correlations by lag for time, distance, and bearing (up to 7 lags); and
10. Adjusted correlations by lag for time, distance, and bearing (up to 7 lags).

The aim of the CWA - Correlogram is to examine repetitive sequences, whether for time interval, distance or direction. It is possible to have separate repetitions for time, distance and direction. For example, an offender may commit crimes every 7 days or so, say, on the weekend. In this case, the individual is repeating himself/herself about once every week. Similarly, an individual may alternate directions, first going East then going West, then going back to the East, and so forth. In other words, what we're asking with the routine is whether there are any repetitions in the sequence of incidents committed by a serial offender. Does he/she repeat the crimes in time? If so, what is the *periodicity* (the repititious sequence? Does he/she repeat the crimes in distance? Is so, what is the periodicity? Finally, does he/she repeat the crimes in direction? If so, what is the periodicity? The CWA - Correlogram, therefore, analyzes the sequence of incidents committed by an individual and does this separately for time interval, distance, and direction.

Offender repetition

Why is this important? Most crime analysis is predicted on the assumption that offenders (people in general) repeat themselves, consciously or unconsciously. That is, individuals have

specific behavior patterns that tend to be repeated. If an individual acts in a certain way (e.g., committing a burglary), then, most likely, the person will repeat himself/herself again. There is no guarantee, of course. But, because human beings do not behave spatially or temporally random but tend to operate in somewhat consistent ways, there is a likelihood that the individual will act in a similar manner again.

This assumption is the basis of profiling which aims at understanding the MO of an offender. If offenders were totally random in their behavior, detection and apprehension would be made much more difficult than it already is. So, between the two extremes of a totally random individual (the 'random walk person') and a totally predictable individual (the 'algorithmic person'), we have the bulk of human behavior, at least in terms of time, distance and direction.

CWA - Diagnostics

The Diagnostics routine is similar to the CWA - Correlogram except that it calculates an Ordinary Least Squares autoregression for a particular lag. That is, for a variable the routine regresses each interval against a previous interval. The user enters the lag number (the default is 1) and the routine produces three regression models for the successive event as the dependent variable against the prior event as the independent variable. There are three equations, for time interval, distance, and bearing separately. The output includes:

1. The sample size (number of events);
2. The number of intervals;
3. Information on the units of time, distance, and bearing;
4. The multiple correlation coefficient;
5. The squared multiple correlation coefficient (i.e., R^2);
6. The overall standard error of estimate;
7. The regression coefficient for the constant and for the prior event;
8. The standard error of the regression coefficients;
9. The t-values for the regression coefficients;
10. The p-value (two-tail) for the regression coefficients;
11. An analysis of variance test for the full model. This includes sum of squares for the regression term and for the residual;
12. The ratio of the regression sum of squares to the residual sum of squares (the F-ratio); and
13. The p-value associated with the F-value.

What the regression diagnostics provides is an indicator of the amount of predictability in the lag. It has the same information as the Correlogram (since the square of the correlation, r^2, is

the same as R^2 for a single independent variable regression equation), but it is easier to interpret. Essentially, it is argued below that, unless the R^2 in the regression equation is sufficiently high, that one is better off using the mean or median lag for prediction. Conversely, if the R^2 is very high, then the user should be suspicious about the data.

CWA - Prediction

Finally, after having analyzed the sequential pattern of events, the user can make a prediction about the time and place of the next event. There are three methods for making a prediction, each with a separate lag:

1. Mean difference
2. Median difference
3. Regression equation

The method is applied to the last event in the data set. The *mean difference* applies the mean interval of the data for the specified lag to the last event. For example, for time interval and a lag of 1, the routine calculates the time interval between each event and takes the average. It then applies the mean time interval to the last time in the data set as the prediction. The *median difference* applies the median interval of the data for the specified lag to the last event. For example, for bearing and a lag of 1, the routine calculates the direction (bearing) between each event, calculates the median bearing, and applies that median to the location of the last event in the data set as the predicted value.

The *regression equation* calculates a regression coefficient and constant for the specified lag and uses the data value for the last *interval* as input into the regression equation; the result is the predicted value. For example, for distance and a lag of 1, the routine calculates the regression coefficient and constant for a regression equation in which each event is compared to the previous event. The last distance in the data set (i.e., between the last event and the previous event) is used as an input for the regression equation and the predicted distance is marked off from the coordinates of the last event.

In other words, the routine takes the time and location of the last event and adds a time interval, a direction, and a distance as a predicted next event (next time, next location). The method by which this prediction is made can be the mean interval, the median interval, or the regression equation. If the user species a lag other than 1, that lag is applied to the last event. For example, for time with a mean difference and a lag of 2, the routine calculates the time interval between each event and every other event, calculates the average, and applies that average to the last event in the data set.

CWA - Prediction Graphical Output

The CWA - Prediction routine outputs five graphical objects in 'shp', 'mif, 'kml' (if the coordinates are spherical) or various Ascii formats. The routine adds five prefixes to the file name of the output object:

1. Events - a line indicating the sequence of events. If the user also brings in the points in the data set, it will be possible to number each of these steps;
2. PredDest - the predicted location for the next event;
3. PW - a line from the last location in the data set to the predicted location;
4. POrigL - a point representing the center of minimum distance of the data set. The center of minimum distance is taken as a proxy for the origin location of the offender; and
5. Path - a line from the expected origin to the predicted destination

For example, if the user provides the file name 'NightRobberies' and specifies a 'shp' output, there will be five objects output:

EventsNightRobberies.shp
PredDestNightRobberies.shp
PathNightRobberies.shp
POrigLNightRobberies.shp
PWNightRobberies.shp

Example 1: A Completely Predictable Individual

The simplest way to illustrate the logic of the CWA is to start with a completely predictable individual. This individual commits crimes on a completely systematic basis. Table 12.5 illustrates the behavior of this individual.

Starting at an arbitrary origin with an X coordinate of 1 and a Y coordinate of 1 on day 1, the individual commits 13 incidents in total. In the table, these are numbered events 1 through 13. From the origin, the individual always travels in a Northeast direction of 45 degrees (clockwise from due North - 0 degrees). The individual's second incident is at coordinate X=2, Y=2. Thus, the individual traveled at 45 degrees from the previous incident and for a distance of 1.4142 (the hypotenuse of the right angle created by traveling one unit in the X direction and one unit in the Y direction). For the third incident, the individual commits this at X=4, Y=4. Thus, the direction is also at 45 degrees from the previous location but the distance is now 2.8284 (or the square root of 8 which comes from a step of 2 along the X axis and a step of 2 along the Y axis). For the fourth incident, the individual commits the crime at X=7, Y=7. Again, the

direction is 45 degrees, but the distance is 4.2426 (or the square root of 18 which comes from a step of 3 along the X axis and a step of 3 along the Y axis).

Table 12.5:

Example of a Predictable Serial Offender: 1

(N = 13 incidents)

Event	X	Y	Distance	Days	Time Interval
1	1	1	-	-	
2	2	2	1.4142	3	2
3	4	4	2.8284	7	4
4	7	7	4.2426	9	2
5	8	8	1.4142	13	4
6	10	10	2.8284	15	2
7	13	13	4.2426	19	4
8	14	14	1.4142	21	2
9	16	16	2.8284	25	4
10	19	19	4.2426	27	2
11	20	20	1.4142	31	4
12	22	22	2.8284	33	2
13	25	25	4.2426	37	4
Logical prediction for next event	14	26	26	1.4142 39	2

For the fifth incident, again the individual traveled at 45 degrees to the previous incident, but repeated himself/herself with a step of only 1 unit in both the X and Y directions. The individual then continued the sequence, always traveling in a 45 degree orientation due North. For distance, a step of 1 in both the X and Y directions is followed by a step of 2 in both directions, and is followed by a step of 3 in both directions. In other words, the individual repeats direction every time and repeats distance every third time. There is a periodicity of 1 for direction and 3 for distance.

For time interval, this individual repeats him/herself every other time. The second event occurs 2 days after the first event. The third event occurs 4 days after the second event; the fourth event occurs 2 days after the third event; the fifth events occurs 4 days after the fourth event; and so forth. In other words, for time interval, the individual repeats him/herself every other interval (i.e., the periodicity is 2).

Figure 12.4 illustrates the sequence; the number at each event location is the number of the day that the individual committed the offense (starting at an arbitrary day 1).

Since this fictitious individual is completely predictable, we can easily guess when and where the next event will occur (see Table 12.5 above). The direction will, of course, be at 45 degrees from the previous location. Looking at the last known event (event 13), the distance traveled was 4.2426. Thus, we predict that the individual will revert to a move of 1 in the X direction and 1 in the Y direction, or coordinates X=26, Y=26. Finally, for time interval, since the last known time interval was 4 days, then this individual will commit the next event 2 days later, or day number 39.

Example 1: Analysis

The first step is to analyze the sequencing of the events. There are 13 events and 12 intervals. The CWA - Correlogram produces the output shown in Table 12.6 below.

Table 12.6:
Correlogram of Predictable Serial Offender: 1
(N=13 Incidents and M=12 Intervals)

Correlated Walk Analysis -- Correlogram:

Sample size .....…....: 13
Measurement type ...: Direct
Input units …….....: Feet
Time units ....…….....: Days
Distance units ...…...: Feet
Bearing units ....…....: Degrees

Correlation

Lag	Time	Distance	Bearing
0	1.00000	1.00000	1.00000
1	-1.00000	-0.42105	1.00000
2	1.00000	-0.56522	1.00000
3	-1.00000	1.00000	1.00000
4	1.00000	-0.38462	1.00000
5	-1.00000	-0.58824	1.00000
6	1.00000	1.00000	1.00000
7	-1.00000	-0.28571	1.00000

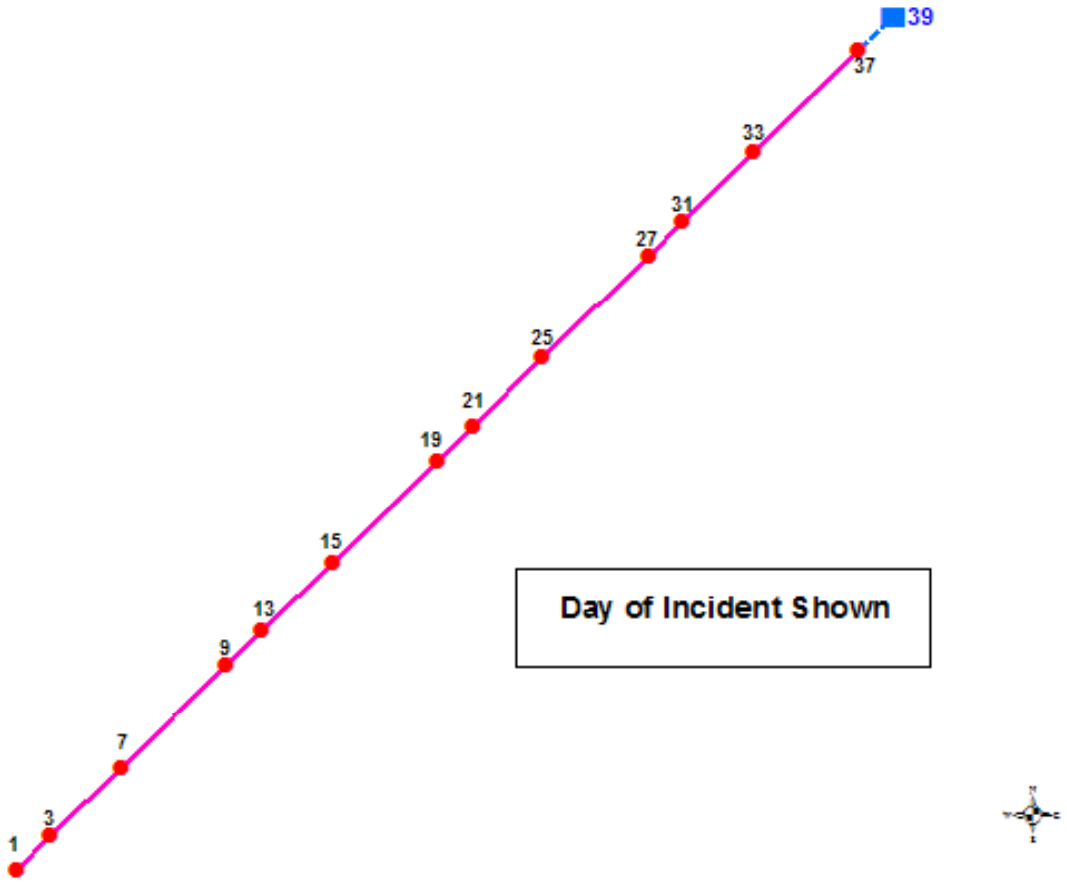
Adjusted Correlation

Lag	Time	Distance	Bearing
0	1.00000	1.00000	1.00000
1	-0.90909	-0.38278	0.90909
2	0.81818	-0.46245	0.81818
3	-0.72727	0.72727	0.72727
4	0.63636	-0.24476	0.63636
5	-0.54545	-0.32086	0.54545
6	0.45455	0.45455	0.45455
7	-0.36364	-0.10390	0.36364

Figure 12.4:

Example of a Predictable Serial Offender: I

(N=13 Incidents)



Looking at the unadjusted correlations, it can be seen that time shows an alternating pattern of perfect correlations. The first repeating positive 1.0 correlation is for lag 2, which is the exact periodicity that was specified in the example. This offender repeats the time sequence every other time. Thus, if the individual alternates between committing offenses 2 and 4 days after the last, then knowing the time interval for the last offense, it can be assumed that the next event will repeat the next-to-the-last time interval.

For distance, the highest correlation is for a lag of 3. This offender repeated himself/herself every third time, which is exactly what was programmed into the example. Knowing the location of the last event, it can be assumed that the individual will choose the same distance for the next interval as three earlier. Finally, all lags show a perfect 1.0 correlation for bearing. The lowest one is taken, which is a lag of 1. That is, this individual repeats the direction every single time (i.e., he/she always travels in the same direction). In summary, the CWA - Correlogram shows that the individual repeats the time interval every other time, the distance every third time, and the direction every time.

The CWA - Diagnostics routine merely confirms these correlations. The regression equations yield an R^2 of 1.0 (unadjusted) for each of three variables, for the appropriate lag. For example, Table 12.7 below shows the regression results for distance for a lag of 3

Table 12.7:
Regression Results for Serial Offender 1: Distance

Variable:	distance	Standard error of estimate:	0.00000
Multiple R:	1.00000	Squared multiple R:	1.00000

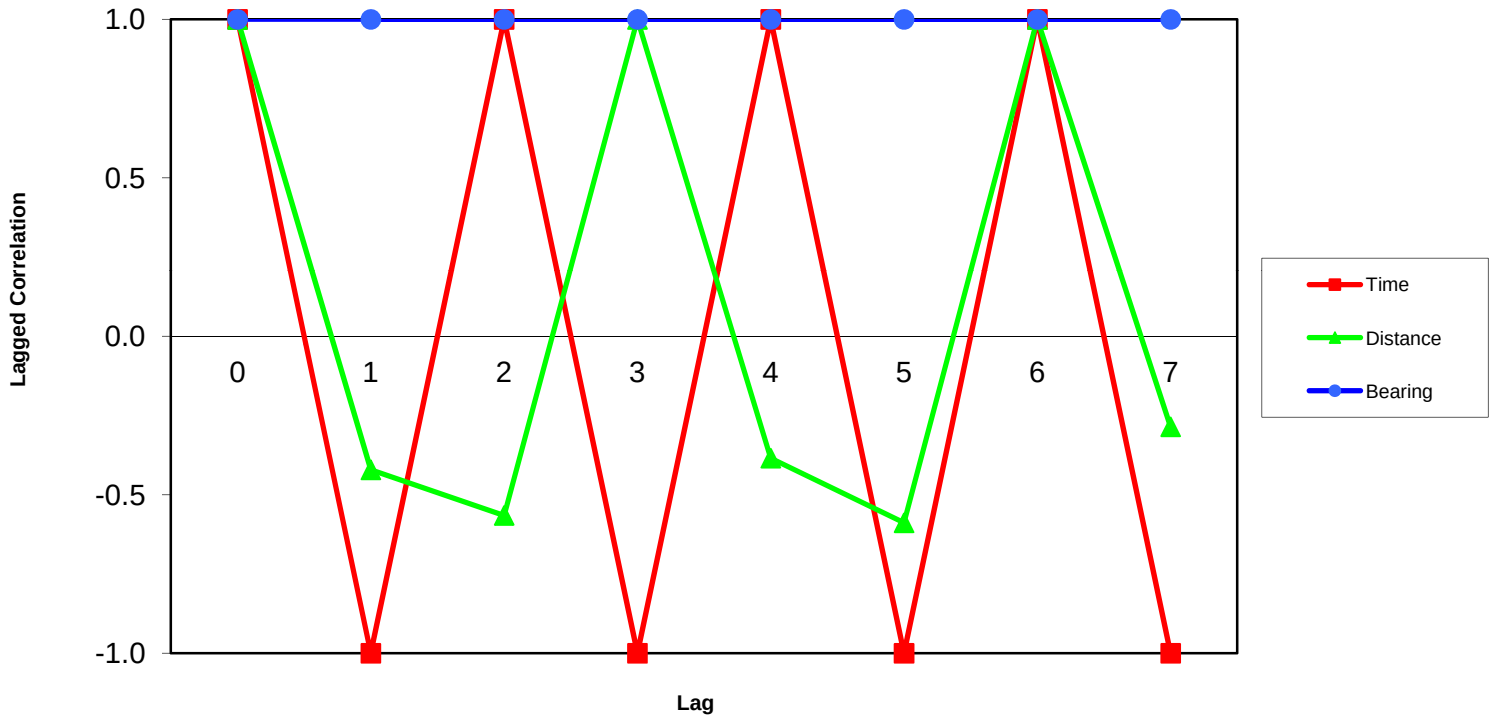
	Coefficient	***Std Error***	***t***	***p(2 Tail)***
Constant	0.000000	0.00000	0.00000	0.00000
Coefficient	1.000000	0.00000	0.00000	0.00000

Analysis of Variance

Source	Sum-of-Squares	df	Mean-Square	F-ratio	P
Regression	12.00000	1	12.00000	0.00000	0.00000
Residual	0.00000	8	0.00000		
Total	12.00000	9			

Figure 12.5:

Correlogram of a Predictable Offender: 1



The adjusted CWA - Correlogram shows a similar pattern, though the absolute correlations have been reduced. The best decision would still be for a lag of 2 for time, a lag of 3 for distance, and a lag of 1 for bearing. Figure 12.5 shows a graph of the correlogram. *CrimeStat* has a built-in graph function for the CWA - Correlogram and CWA-Adjusted correlogram.

Example 1: Prediction

Finally, for prediction, it is apparent that the best method would be to use a regression equation with lags of 2 for time, 3 for distance, and 1 for bearing. Table 12.8 shows the output. As can be seen, the routine predicts exactly the next time and location. The next event for this completely predictable serial offender will be on day 39 at the location with coordinates X=26, Y=26.

Table 12.8:
Predicted Results for Serial Offender 1
(Regression Equation with Lags of 2 for Time, 3 for Distance, 1 for Bearing)

Variable	Predicted Value	From Event	Method	Lag
Time interval	2.00000	13	Regression	2
Distance interval	1.41421	13	Regression	3
Bearing interval	44.99997	13	Regression	1
Predicted time	39.00000			
Predicted X coordinate	26.00000			
Predicted Y coordinate	26.00000			

The regression equation is the best model in this case. The other methods produce reasonably close approximations, however. Table 12.9 shows the results of using other methods for prediction. As seen, a model where all three components (time, distance, bearing) were lagged by 1 as well as a model where all three components were lagged by 3 also produces the expected correct answer. The mean interval and median interval methods also produce reasonably close, though not exact, answers. In this particular case, the regression method with the best lags produced the optimal solution.

Table 12.9:
Comparison of Methods for Predictable Serial Offender 1

	EVENT	X	Y	DISTANCE	DAYS	TIME INTERVAL
Logical Prediction for next event	**14**	**26**	**26**	**1.4142**	**39**	**2**
PREDICTION:						
Mean (lag=1)	14	27.0	27.0	2.8	40.0	3.0
Median (lag=1)	14	27.0	27.0	2.8	41.0	4.0
Regression:						
Lag=1	14	26.6	26.6	2.3	39.0	2.0
Lag=2	14	27.0	27.0	2.9	39.0	2.0
Lag=3	14	26.0	26.0	1.4	39.0	2.0
Optimal (t=2,d=3,b=1)	14	26.0	26.0	1.4	39.0	2.0

Example 2: Another Completely Predictable Individual

A second example is also a perfectly predictable individual. This time, the directional component changes from event to event. The directional trend is northward, but with changes in angle every third event. The time pattern is completely consistent with subsequent events occurring every two days. Table 12.10 presents the pattern and the logical next event while figure 12.6 displays the pattern.

The CWA - Correlogram reveals that both distance and bearing repeat themselves every third event while the time interval is repeated every time. The regression diagnostics show that there is perfect predictability for time and for distance, and high predictability for bearing (not shown). Finally, a regression model is used for prediction with lags of 1 for time, 3 for distance, and 3 for bearing. The model correctly predicts the expected time (days=25) and location (X=3, Y=25). Table 12.11 shows the results.

Table 12.10:

Example of a Predictable Serial Offender: 2

(N = 14 incidents)

Time Event	X	Y	Distance	Days	Interval
1	3	1	-	1	-
2	1	3	2.8284	3	2
3	1	5	2.0000	5	2
4	3	7	2.8284	7	2
5	1	9	2.8284	9	2
6	1	11	2.0000	11	2
7	3	13	2.8284	13	2
8	1	15	2.8284	15	2
9	1	17	2.0000	17	2
10	3	19	2.8284	19	2
11	1	21	2.8284	21	2
12	1	23	2.0000	23	2
Logical prediction for next event	**13**	**3**	**25**	**2.8284**	**25**	**2**

Methodology for CWA

These two examples illustrate what the CWA routine is doing. There are three steps. First, the sequential pattern is analyzed with the CWA - Correlogram. This analysis shows which lags have the strongest correlations between lags for time, distance, and bearing separately. Second, the pattern is tested with a regression model. The purpose is to determine how strong a relationship can be obtained for any particular model. As will be suggested below, if a model is too weak or, conversely, too strong, it most likely will not predict very well. Third, a prediction model is selected. The user can utilize the regression model or use the mean interval or median interval. Fourth, and finally, the prediction is made.

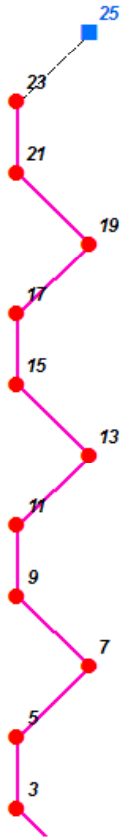
Example 3: A Real Serial Offender

How well does the CWA routine work with real serial offenders? People are not as predictable as these examples. The examples are algorithmic and people don't work like

Figure 12.6:

Example of a Predictable Serial Offender: 2

(N=12 Incidents)



Day of Incident Shown

Table 12.11:
Comparison of Methods for Predictable Serial Offender 2

	EVENT	X	Y	DISTANCE	DAYS	TIME INTERVAL	DIRECTION
Logical Prediction for next event	**13**	**3**	**25**	**2.8284**	**25**	**2**	**45**
PREDICTION:							
Mean (lag=1)	13	2.2	25.2	2.5	25.0	2.0	28.6
Median (lag=1)	13	3.0	25.0	2.8	25.0	2.0	45.0
Regression:							
Lag=1	13	3.0	25.0	2.8	25.0	2.0	45.0
Lag=2	13	1.9	25.2	2.4	25.2	2.0	22.5
Lag=3	13	3.0	25.0	2.8	25.0	2.0	45.0
Optimal (t=1,d=3,b=3)	13	3.0	25.0	2.8	25.0	2.0	45.0

algorithms. But, to the extent to which there is some predictability in human behavior, the CWA routine can be a useful tool for crime analysis, detection, and apprehension.

To illustrate this, a serial offender was identified from a large data set obtained from Baltimore County. The individual committed 16 offenses between 1992 and 1997 when he was eventually apprehended. The profile of crimes committed by this individual were quite diverse. There were 11 larceny incidents (shoplifting and bicycle theft), 1 residential burglary, 1 commercial burglary, 2 assaults, and 1 robbery.

To test the model, the first 15 incidents were used to predict the 16^{th}. This allowed the error between the observed and predicted values for time and location to be used for evaluation. Figure 12.7 shows the sequencing of actions of the first 15 incidents committed by this individual, most of which occurred in the eastern part of Baltimore County.

The CWA - Correlogram revealed a complicated pattern (Figure 12.8). The adjusted matrix was used because of the high correlations at higher-order lags. Nevertheless, the optimal lags appeared to be 1 for time, 3 for distance, and 6 for bearing. A regression model was used to test these parameters. Figure 12.7 also shows the predicted location for the next likely location (the red plus sign) and the location where the individual actually committed the 16^{th} event (green triangle). The error in prediction was good. The distance between the actual and predicted

Figure 12.7:

Likely Location for Next Crime Serial Offender in Baltimore County

N=16 Incidents

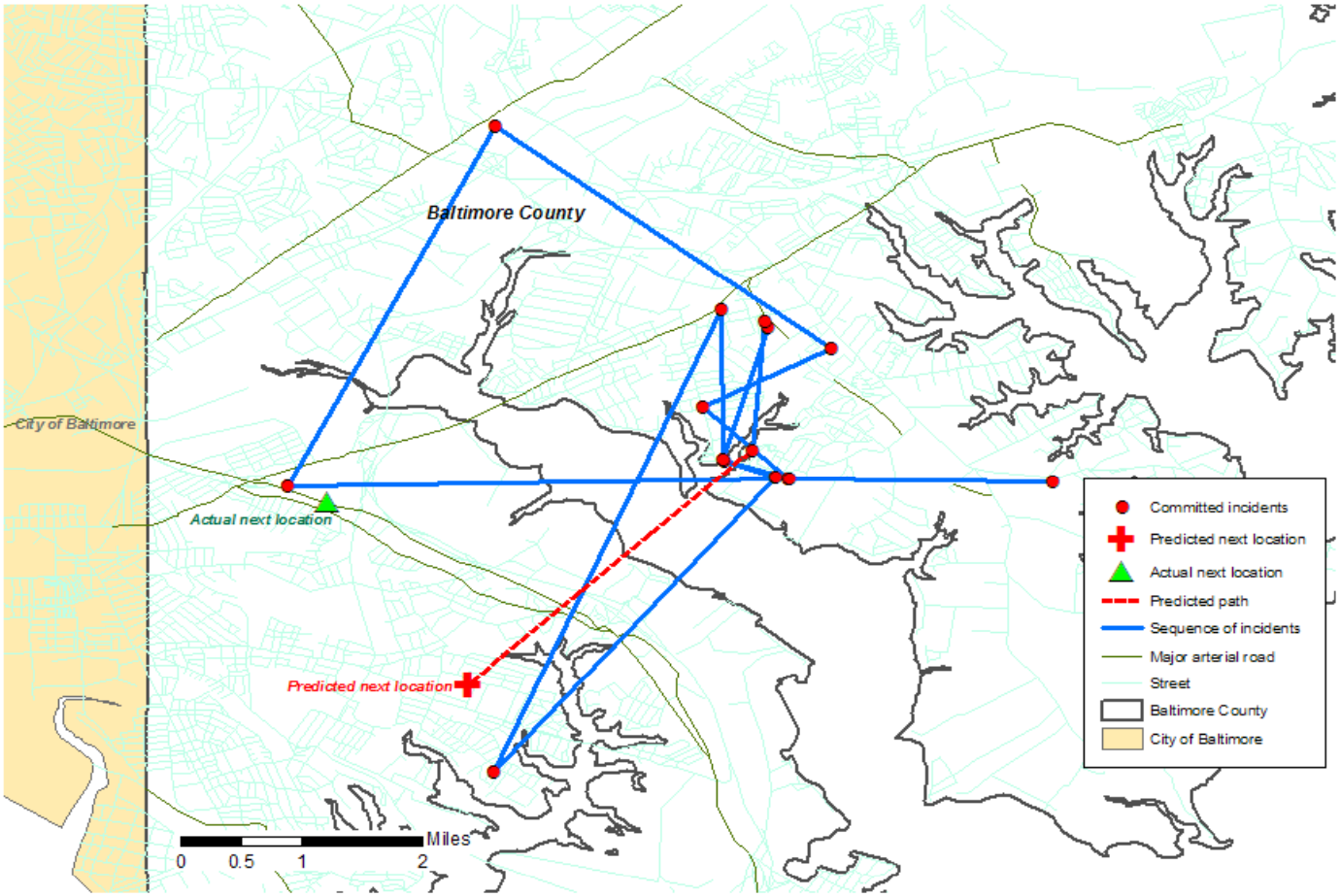
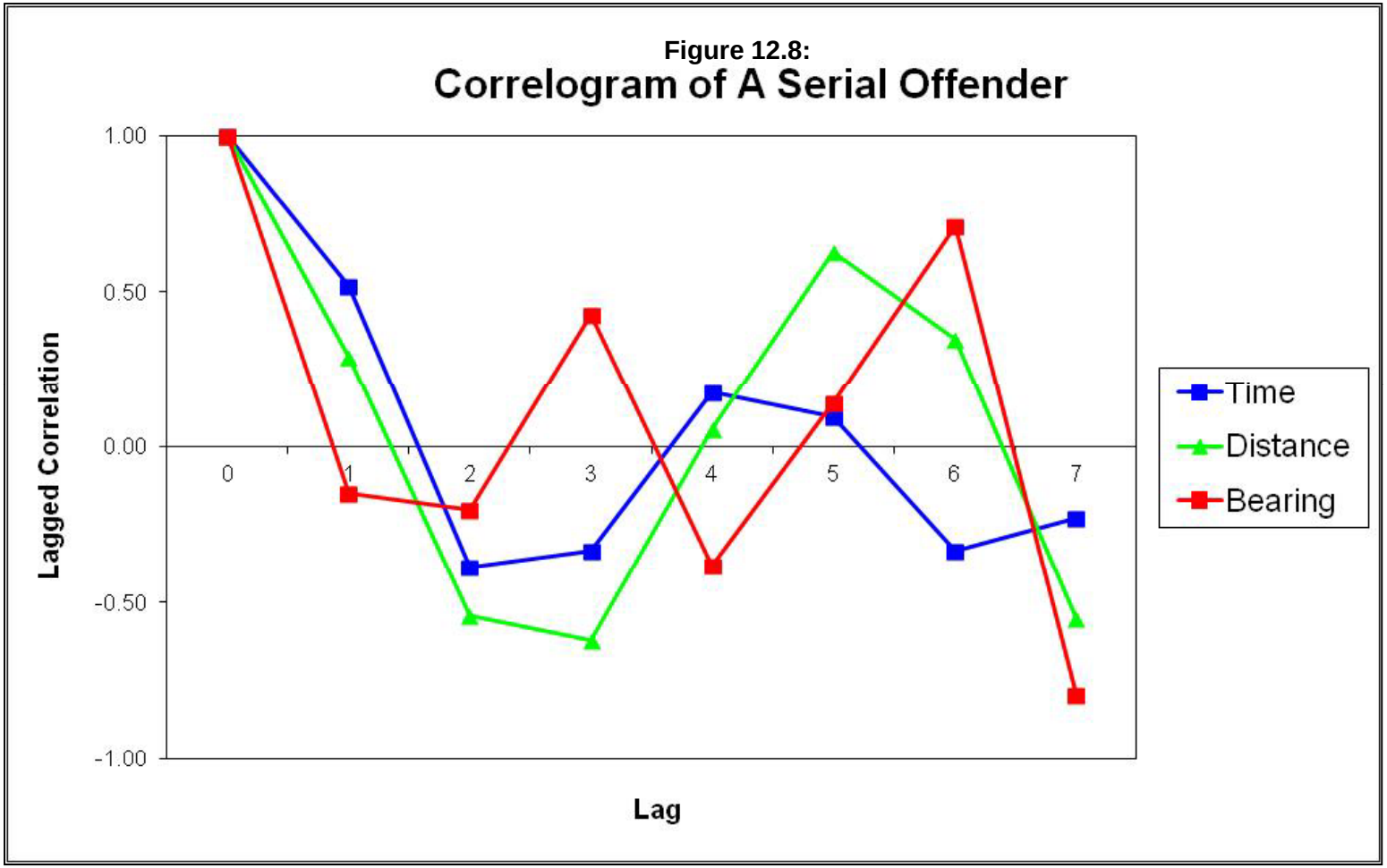


Figure 12.8:


Correlogram of A Serial Offender
1.00
0.50
0.00
-0.50
-1.00
Lagged Correlation
0
1
2
3
4
5
6
7
Lag
Time
Distance
Bearing

locations was 1.8 miles and the error in predicting the time of the next location was 3.9 days. Overall, the model did quite well for this individual.

Event Sequence as an Analogy to a Correlated Walk

Nevertheless, there are problems in the model for this case. First, this is not a true sequence of actions, but a pseudo-sequence. The individual doesn't go from the first event to the second event to the third event, and so forth. A considerable time may elapse between events. Similarly, distance and direction are conceptual only, not real. For example, in figure 12.7, the individual did not actually travel across the inlets of the Chesapeake Bay as the lines indicate. Distance between the events was actually much greater than estimated by the model and direction was more complex. Nevertheless, to the extent to which an individual makes a spatial decision about where to go, implicitly he or she is making a directional and distance decision. In other words, the decision making process may take into account prior locations. In this case, the CWA routines would be useful.

Example 4: A Second Real Serial Offender

A second real example confirms that the method can produce reasonably close predictions. An offender committed 13 crimes, including three incidents of shoplifting, eight incidents of theft from a vehicle, one residential burglary, and one highway robbery. The correlogram showed that a lag of 1 was strongest for time, distance, and bearing (figure 12.9). The R-squares were moderate (0.45 for time; 0.18 for distance; 0.18 for bearing). Using the regression method with a lag of 1 for each component, the likely location of the next event was predicted (Figure 12.10). The error between the predicted event and the actual event was, again, reasonable with a difference in time of 3.3 days and a difference in distance of 2.4 miles.

Accuracy of Predictions

However, it is important not to be overly optimistic about the technique. It is always possible to find cases that fit a method very well. The above mentioned cases appear to do that. Unfortunately, the method is not a magic elixir for predicting serial offenders. Like any method, it has error. It is also a fairly new tool in crime analysis so that we do not have a lot of experience with it. One example of its use was by Helms (2005), who was also is cautious about its utility.[2]

---

2 Personal communication from Dan Helms, National Law Enforcement Corrections and Technology Center, Denver, CO.

Figure 12.9:

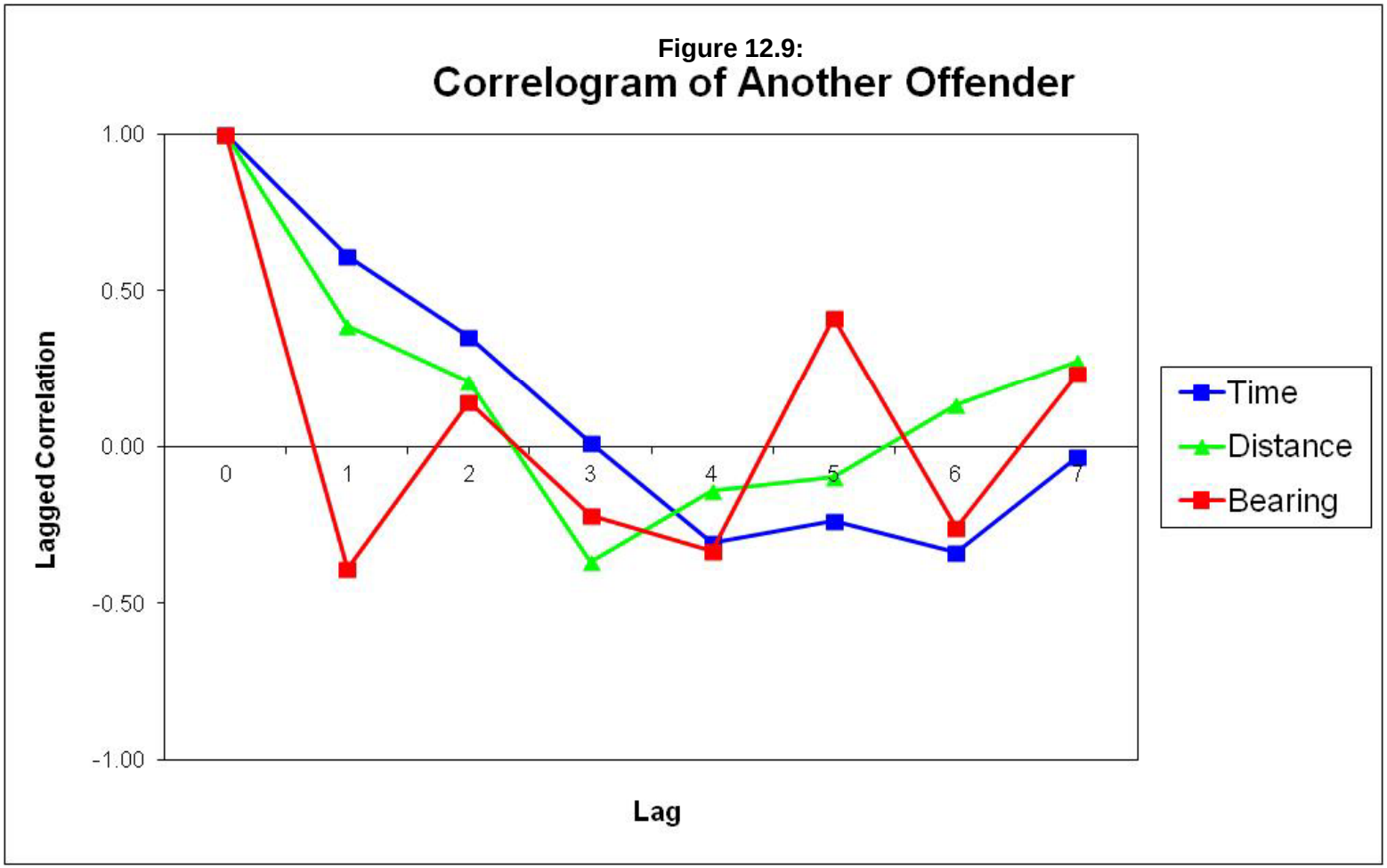
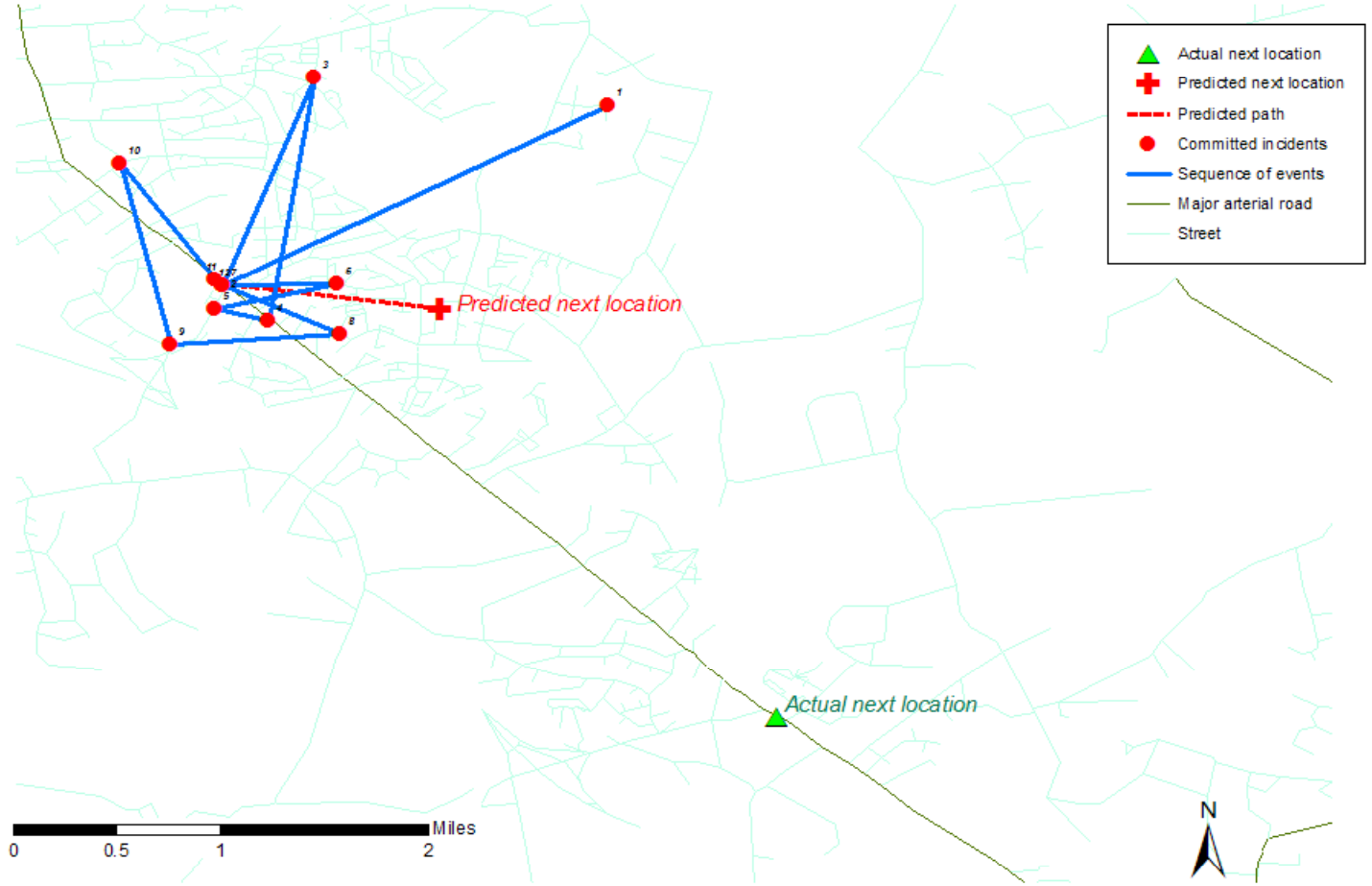

Correlogram of Another Offender
1.00
0.50
0.00
-0.50
-1.00
Lagged Correlation
0
1
2
3
4
5
6
7
Lag
Time
Distance
Bearing

Figure 12.10:

Likely Location for Next Crime
Another Serial Offender in Baltimore County
N=13 Incidents



To explore the accuracy of the method, 50 serial offenders were identified from a large data base of more than 41,000 incidents in Baltimore County between 1993 and 1997 (see Chapter 10 for details). The 50 offenders were identified based on knowing the dates on which they committed crimes, or at least on which they committed crimes for which they were charged and eventually tried. The number of incidents varied from a low of 7 incidents to a high of 38 incidents. An attempt was made to produce balance in the number of incidents, though the actual distribution of cases did reflect the availability of candidates in the data base. For the fifty individuals, the distribution of incidents was 7 (five individuals), 8 (four individuals), 9 (six individuals), 10 (two individuals), 11 (five individuals), 12 (five individuals), 13 (six individuals), 14 (three individuals), 15 (six individuals), 17 (two individuals), and one individual each for 20, 21, 24, 29 and 38 incidents.

To test the CWA model, the last event committed by these individuals was removed so that N-1 events could be used to predict event N. In this way, it is possible to evaluate the accuracy of the method.

Ten methods were compared:

1. The optimal regression method for time with the lag having the strongest relationship being selected;
2. The optimal regression method for location (distance and bearing) where the with the lags for distance and bearing having the strongest relationship being selected;
3. A regression model for time with a lag of 1;
4. A regression model for location with a lag of 1 (for both distance and bearing);
5. The mean interval for time;
6. The mean interval for location (distance and bearing);
7. The median interval for time;
8. The median interval for location (distance and bearing);
9. The mean center of the incidents (for location only); and
10. The center of minimum distance of the incidents (for location only).

The latter two methods were used for reference. For journey to crime estimation, the center of minimum distance is the best at predicting the origin location of serial offenders (see Chapter 13). The reason is because this statistic *minimizes the distance* to all incident locations. The mean center was close behind, though not quite as good. As an estimate, the center of minimum distance is a very good index when there is a single origin that is being predicted. On the other hand, where the purpose is to predict the location of a next event, the center of minimum distance and mean center may be less than useful since they will not generally predict the actual next location. They minimize error, but are rarely accurate. For example, in the above mentioned cases (two theoretical and two real), these statistics did not predict accurately the location of the

next event. Instead, they identified a point in the middle of the distribution where the sum of the distances to all incident locations was small.

Error analysis

Each of the models was compared to the actual time and location of the last, removed incident. For time, the error measure was in days (the absolute difference between the actual day and the predicted day). For location, the error measure was in miles (i.e., absolute distance between the actual and predicted location). The results were mixed. Overall, error was moderate. Table 12.12 summarizes the overall error.

Overall, the center of minimum distance and the mean center do produce, as expected, smaller errors for distance than any of the CWA methods; as noted above, locations in the middle of the distribution of incidents will minimize error, but they will not predict accurately the location of a next event nor indicate in which direction it will occur from the last event. On the other hand, the CWA methods are not particularly accurate, either. They work very well for a completely predictable offender, as was seen in the examples above, but not necessarily for real offenders.

Among the CWA methods, the mean interval, median interval and the lag 1 regression appears to give better results for time than the optimal regression. Overall, the median interval produces the lowest median error, which is about a month and half. In terms of location, the mean interval and median intervals produce slightly better results than the optimal regression, though the lag 1 regression was just as good.

Comparison of CWA Methods

At this point, it is unclear as when it is best to use this technique. Three variables seem to explain part of the error variation.

First, a larger sample size leads to better prediction, as would be expected (Table 12.13). For time, there is definitely an improvement in predictability with larger sample sizes. Among these methods, the mean interval and lag 1 regression show the smallest error for the largest samples (14 cases). For distance, on the other hand, generally, the error increases with increasing sample size. The one exception is for the optimal regression method where medium-sized samples (10-13 cases) produce the lowest error.

Table 12.12:
Average and Median Error for CWA Methods
(50 Serial Offenders)

Method	Average Error	Median Error
Time (days)		
Optimal regression: time	112.2	79.8
Lag 1 regression: time	88.1	70.0
Mean interval: time	89.7	64.9
Median interval: time	91.2	45.5
Distance (miles)		
Optimal regression: location	6.4	5.4
Lag 1 regression: location	5.7	4.2
Mean interval: location	5.8	4.7
Median interval: location	5.3	3.9
Reference Location (miles)		
Mean center	3.3	1.7
Center of minimum distance	3.1	1.2

Factors Affecting Predictability

Long time span

There are a variety of reasons for these results, but one reason may be the time span of the events. Some of these offenders committed crimes over a long period, up to five years. Sample size is intrinsically related to the time span (r=0.55). The longer the time span that an offender commits crimes, the more incidents he/she will perpetrate. With increasing time, the individual's behavior patterns may change (e.g., he/she may move residences).

For those offenders with many incidents, a separate analysis was conducted of the events occurring within the last year. Many of these individuals appeared to have moved their base of operation over time, so the isolation of the most recent events was done in order to produce a clearer behavior pattern. The results, while promising, were not dramatic. Accuracy was improved a little compared to using the full sequence, particularly spatial accuracy. However, even with the last few events, these frequently occurred over a long time period (up to two years).

Consequently, the idea of isolating a 'clean' set of events did not materialize, at least with these data. On the other hand, with a data set of only recent events, it may be possible to improve predictability.

Table 12.13:
Sample Size and Prediction Error
(Average Error)

Time (days)

Sample Size	Optimal Regression	Lag 1 Regression	Mean Interval	Median Interval
6-9	143.4	108.5	116.4	120.8
10-13	108.2	86.8	83.4	79.5
11+	79.8	65.1	65.7	71.2

Distance (miles)

Sample Size	Optimal Regression	Lag 1 Regression	Mean Interval	Median Interval
6-9	7.4	5.2	5.0	4.4
10-13	5.5	6.0	5.7	5.5
11+	6.1	5.9	6.8	6.1

Centrographic: Distance (miles)

Sample Size	Mean Center	Center of Minimum Distance
6-9	2.9	2.4
10-13	2.9	3.1
11+	4.3	4.1

Strength of predictability

A second variable that appears to have an effect is the strength of predictability, based on the first N-1 cases. For the diagnostics routine, as the overall R-square for the regression equation increases, the regression equation does better. However, with very high R-square coefficients, the error is worse. Table 12.14 shows the relationship.

The lowest error is obtained with moderate R-square coefficients, for both time and distance. This is why one has to be careful with very high lagged correlations in the correlogram and high R-squares in the diagnostics. Unless one is dealing with a perfectly predictable individual (as the two theoretical examples illustrated), high correlations may be a result of a very small sample size, rather than any inherent predictability.

Table 12.14:
Regression Diagnostics and Prediction Error
Comparison of CWA Regression Methods

	Time (days)		*Distance (miles)*	
R-Square	**Optimal Regression**	**Lag 1 Regression**	**Optimal Regression**	**Lag 1 Regression**
0-0.29	93.7	90.9	6.7	6.3
0.30-0.59	89.3	33.8	6.0	5.0
0.60+	164.3	122.7	6.3	5.2

Limitations of the Technique

In short, users should be careful about using the CWA technique. It can be useful for identifying repeating patterns by an offender, but it won't necessarily predict accurately the offender's next actions. There are a variety of reasons for the lack of predictability. First, there may be intermediate events that are unknown. With each of these offenders in the Baltimore County data base, there is always the possibility that the individuals committed other crimes for which they were not charged. The sequential analysis assumes that all the events are known. But this may not be the case.

A simulation on several cases was conducted by removing events and then re-running the correlogram and prediction models. Removing one event did not appreciably alter the relationship, but removing more than one event did. In other words, if there are unknown events, the true sequential behavior pattern of the offender may not be properly identified. Considering that most offenders commit fewer than 10 incidents before they get caught, the statistical effect of missing information may be critical.

A second reason has been alluded to already. In applying the model to crime events, it is not a true sequential model, but a *pseudo-sequential* model since much time may intervene between events. Distance and direction are conceptual in the sense that the individual doesn't directly orient from one event to the other, but returns to his/her living patterns. Thus, what may

appear to be a repeating pattern may not be. Here, the issue of sample size is critical. If there are only a few incidents on which to base an analysis, one could see a pattern which actually doesn't exist. One has to be careful about drawing inferences from very small samples.

A third reason is that people are inherently unpredictable. The two algorithmic examples produced excellent results, but few persons are that systematic about their behavior. Therefore, we must be cautious in expecting too much out of the model.

Conclusion

Nevertheless, the model has utility. First, it can help police identify whether there is a pattern in an offender's behavior. Knowing that there is a pattern can help in planning an arrest strategy. Even if the strategy does not pay off every time, it may improve police effectiveness. In short, the CWA can help a police department analyze the sequential behavior of an offender they are trying to catch. They may be able to anticipate a new event and may be able to warn people who are more likely to be attacked by this individual. If used carefully, the model can be useful for crime analysis and detection.

Second, it can encourage the development of additional predictor tools for individuals. As mentioned above, the center of minimum distance produces a 'best guess' estimate in the sense that it minimizes the distance to the next event. It usually doesn't predict the next event, but it does produce a minimal error. If used in conjunction with the CWA, it may be possible to narrow the search area for the next event.

Third, the CWA model can stimulate research into crime prediction. Police are always trying to predict the next event by an offender and will use multiple techniques and a lot of intuition in trying to 'out-guess' an offender. It is hoped that the CWA model will stimulate more research into predicting the sequence of offender behavior as well into how those sequences aggregate into a large spatial pattern. Most of this text has been devoted to analyzing the spatial patterns of a large number of events. The statistics have, perhaps naively assumed that each of those events were independent. In reality, they are not since many crimes are committed by the same individuals. In theory, a distribution of crime incidents could be disaggregated into a distribution of *sequences of events* committed by the same offenders, if we had enough information. Understanding how aggregate distributions is a by-product of the behavior of a limited number of individuals is an important research goal that needs to be addressed.

References

Bailey, T. C. & Gatrell, A. C. (1995). *Interactive Spatial Data Analysis*. Longman Scientific & Technical: Burnt Mill, Essex, England.

Barnard, G. A. (1963). Comment on 'The Spectral Analysis of Point Processes' by M. S. Bartlett, *Journal of the Royal Statistical Society*, Series B, 25, 294.

Besag, J. & Newell, J. (1991). The detection of clusters in rare diseases. Journal of the Royal Statistic Society A, 154, Part I, 143-155.

Chaitin, G. (1990). *Information, Randomness and Incompleteness* (second edition). World Scientific: Singapore.

Chen, A. & Renshaw, E. (1994) The general correlated random walk. *Journal of Applied Probability*, 31, 869-884.

Chen, A. & Renshaw, E. (1992). The Gillis-Domb-Fisher correlated random walk. *Journal of Applied Probability*, 29, 792-813.

Dwass, M (1957). Modified randomization tests for nonparametric hypotheses. *Annals of Mathematical Statistics*, 28, 181-187.

Henderson, R., E., Ford, D., Renshaw, E. & Deans, J. D. (1983). Morphology of the structural root system of Sitka Spruce 1. Analysis and Quantitative Description. *Forestry*, 56 (2), 121-135.

Henderson, R., Renshaw, E., & Ford, D. (1984). A correlated random walk model for two-dimensional diffusion. Journal of Applied Probability, 21, 233-246.

Henderson, R., Renshaw, E., & Ford, D. (1983). A note on the recurrence of a correlated random walk. Journal of Applied Probability, 20, 696-699.

Knox, E. G. (1988). Detection of clusters. In Elliott, P. (ed), *Methodology of Enquiries into Disease Clustering*, London School of Hygiene and Tropical Medicine: London.

Knox, E. G. (1964). The detection of space-time interactions. *Applied Statistics*, 13, 25-29.

Knox, E. G. (1963). Detection of low intensity epidemicity: application in cleft lip and palate. *British Journal of Preventive and Social Medicine*, 18, 17-24.

References (continued)

Kulldorff, M. & Nagarwalla, N. (1995). Spatial disease clusters: Detection and inference, *Statistics in Medicine*, 14, 799-810.

Malkiel, B. G. (1999). *A Random Walk Down Wall Street* (revised edition). W. W. Norton & Company: New York.

Mantel, N. (1967). The detection of disease clustering and a generalized regression approach. *Cancer Research*, 27, 209-220.

Mantel, N. & Bailar, J. C. (1970). A class of permutational and multinomial test arising in epidemiological research, *Biometrics*, 26, 687-700.

Renshaw, E. (1985). Computer simulation of sitka spruce: spatial branching models for canopy growth and root structure. *Journal of Mathematics Applied in Medicine and Biology*, 2, 183-200.

Spitzer, F. (1976). *Principles of Random Walk* (second edition). Springer: New York.

Endnotes

i. Henderson, Renshaw and Ford (1983) defined the correlated walk as a two-dimensional walk where the sum of the probabilities in four directions along a lattice are:

$$P = p + q + 2r = 1$$

where *P* is the total probability (1), *p* is the probability of continuing in the same direction, *q* is the probability of moving in an opposite direction, and *r* is the probability of moving one unit to the right or to the left. The advantage of this formulation is that the probabilities do not have to be equal (i.e., p could exceed q or r). Nevertheless, the individual steps can be considered a special case of a correlated random walk in the plane (Henderson, 1981).

The non-lattice two dimensional case can also be considered a recurrent random walk since a step in any direction (not just along a lattice) can be considered the result of two steps, one in the X direction and one in the Y (or, alternatively, a pairing of all steps in the X direction with all steps in the Y direction). Unfortunately, this logic does not apply to more than two dimensions. Such multi-dimensional walks do not have to return to their origin. However, Spitzer (1963) has shown that an independent walk is recurrent if the second moment around the origin is finite.

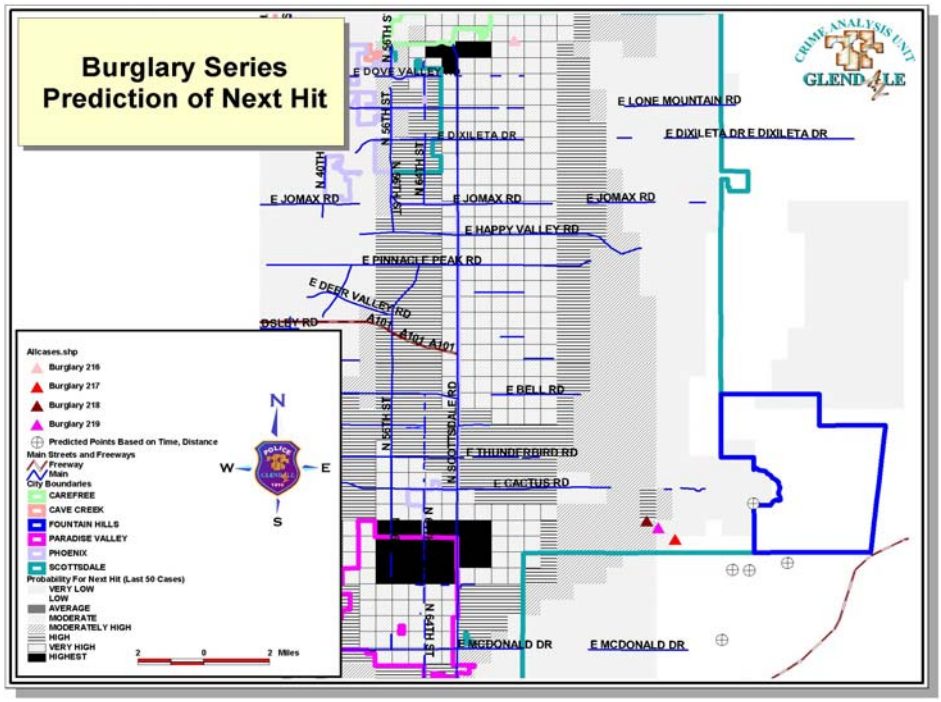
Attachments

Tracking a Burglary Gang with the Correlated Walk Analysis

Bryan Hill
Glendale Police Department
Glendale, AZ

The space-time analysis tools provided with *CrimeStat* add an important element to an analyst's review of a tactical prediction effort. Although the method for calculating the Correlated Walk Analysis (CWA) is still more experimental than proven, it allows the analyst to see potential patterns in relation to a suspect's crime travel in terms of time, distance, and direction. In a recent burglary series involving several jurisdictions in our county, the CWA technique was used as part of an aggregate process referred to as the Probability Grid Method. That method combines results from several models to predict the next likely area for a new hit in a crime series. One of the most confusing aspects of these burglaries was the fact that several jurisdictions were involved and the offenders seemed to bounce back and forth from one jurisdiction to the next.

There were also 219 offenses in the series, providing considerable complexity. Because there were so many events, the distances could be anywhere from 0.5 miles to 20 miles, I could never really put my finger on what direction or distance the offender would hit next, but was confident a pattern existed and was likely changing over time. The following map shows the probability grid areas predicted and the CWA points predicted. The triangles shown represent the last four hits. The first hit was near the probability grid prediction in the northern portion of the map; however the subsequent hits were all very close to where the CWA routine predicted they would be. This was also a brand new area for these offenders and was a surprise to the department investigating these incidents. This area was not what was expected based on the SD ellipses and other methods used to predict the next event. The CWA tool requires more testing to determine the accuracy of its predictions, however it may turn out to be a valuable tool in a crime analyst's arsenal.



Chapter 13:

Journey-to-Crime Estimation

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Chapter 13:

Journey-to-Crime Estimation

The *Journey-to-crime* (Jtc) routine is a distance-based method that makes estimates about the likely residential location of a serial offender. It is an application of *location theory*, a framework for identifying optimal locations from a distribution of markets, supply characteristics, prices, and events. The following discussion gives some background to the technique. Those wishing to skip this part can go to page 13.12 for the specifics of the Jtc routine.

Location Theory

Location theory is concerned with one of the central issues in geography. This theory attempts to find an optimal location for any particular distribution of activities, population, or events over a region (Haggett, Cliff & Frey, 1977; Krueckeberg & Silvers, 1974; Stopher & Meyburg, 1975; Oppenheim, 1980, Ch. 4; Bossard, 1993). In classic location theory, economic resources were allocated in relation to idealized representations (Anselin & Madden, 1990). Thus, von Thünen (1826) analyzed the distribution of agricultural land as a function of the accessibility to a single population center (which would be more expensive towards the center), the value of the product produced (which would vary by crop), and transportation costs (which would be more expensive farther from the center). In order to maximize profit and minimize costs, a distribution of agricultural land uses (or crop areas) emerges flowing out from the population center as a series of concentric rings. Weber (1909) analyzed the distribution of industrial locations as a function of the volume of materials to be shipped, the distance that the goods had to be shipped, and the unit distance cost of shipping; consequently, industries become located in particular concentric zones around a central city. Burgess (1925) analyzed the distribution of urban land uses in Chicago and described concentric zones of both industrial and residential uses. Their theory formed the backdrop for early studies on the ecology of criminal behavior and gangs (Thrasher, 1927; Shaw, 1929).

In more modern use, the location of persons with a certain need or behavior (the 'demand' side) is identified on a spatial plane and places are selected as to maximize value and minimize travel costs. For example, for a consumer faced with two retail shops selling the same product, one being closer but more expensive while the other being farther but less expensive, the consumer has to trade off the value to be gained against the increased travel time required. In designing facilities or places of attraction (the 'supply' side), the distance between each possible facility location and the location of the relevant population is compared to the cost of locating near the facility. For example, given a distribution of consumers and their propensity to spend,

such a theory attempts to locate the optimal placement of retail stores, or, given the distribution of patients, the theory attempts to locate the optimal placement of medical facilities.

Predicting Locations from a Distribution

One can also reverse the logic. Given the distribution of demand, the theory could be applied to estimate a central location from which travel distance or time is minimized. One of the earliest uses of this logic was that of John Snow, who was interested in the causes of cholera in the mid-19th century (Cliff and Haggett, 1988). He postulated the theory that water was the major vector transmitting the cholera bacteria. After investigating water sources in the London metropolitan area and concluding that there was a relationship between contaminated water and cholera cases, he was able to confirm his theory by an outbreak of cholera cases in the Soho district of London. By plotting the distribution of the cases and looking for water sources in the center of the distribution (essentially, the center of minimum distance; see Chapter 4), he found a well on Broad Street that was, in fact, contaminated by seepage from nearby sewers. The well was closed and the epidemic in Soho receded. Incidentally, in plotting the incidents on a map and looking for the center of the distribution, Snow applied the same logic that had been followed by the London Metropolitan Police Department who had developed the famous 'pin' map in the 1820s.

Theoretically, there is an optimal solution that minimizes the distance between demand and supply (Rushton, 1979). However, computationally, it is an almost impossible task to define, requiring the enumeration of every possible combination. Consequently in practice, approximate, though sub-optimal, solutions are obtained through a variety of methods (Everitt, 2011, Ch. 4).

Travel Demand Modeling

A sub-set of location theory models the travel behavior of individuals. It actually is the converse. If location theory attempts to allocate places or sites in relation to both a supply-side and demand-side, travel demand theory attempts to model how individuals travel between places, given a particular constellation of them. One concept that has been frequently used for this purpose is that of the *gravity function*, an application of Newton's fundamental law of attraction (Oppenheim, 1980). In the original Newtonian formulation, the attraction, F, between two bodies of respective masses M_i and M_j, separated by a distance d_{12}, will be equal to:

$$F = g\frac{M_1 M_2}{d_{ij}^2} \tag{13.1}$$

where *g* is a constant or scaling factor which ensures that the equation is balanced in terms of the measurement units (Oppenheim, 1980). As we all know, of course, *g* is the gravitational constant in the Newtonian formulation. The numerator of the function is the *attraction* term (or, alternatively, the attraction of M_2 for M_1) while the denominator of the equation, D^2, indicates that the attraction between the two bodies falls off as a function of their *squared* distance. It is an *impedance* term.

Social Applications of the Gravity Concept

The gravity model has been the basis of many applications to human societies and has been applied to social interactions since the 19th century. Ravenstein (1895) and Andersson (1897) applied the concept to the analysis of migration by arguing that the tendency to migrate between regions is inversely proportional to the squared distance between the regions. Reilly's 'law of retail gravitation' (1929) applied the Newtonian gravity model directly and suggested that retail travel between two centers would be proportional to the product of their populations and inversely proportional to the square of the distance separating them:

$$T_{ij} = \alpha \frac{P_i P_j}{d_{ij}^2} \tag{13.2}$$

where T_{ij} is the interaction between centers *i* and *j*, P_i and P_j are the respective populations, d_{ij} is the distance between them raised to the second power and α is a balancing constant. In the model, the initial population, P_i, is called a *production* while the second population, P_j, is called an *attraction*.

Stewart (1950) and Zipf (1949) applied the concept to a wide variety of phenomena (migration, freight traffic, exchange of information) using a simplified form of the gravity equation:

$$T_{ij} = \alpha \frac{P_i P_j}{d_{ij}} \tag{13.3}$$

where the terms are as in equation 13.2 but the exponent of distance is only 1. In doing so, they basically linked location theory with travel behavior theory. Given a particular pattern of interaction for any type of goods, service or human activity, an optimal location of facilities should be solvable.

In the Stewart/Zipf framework, the two P's were both population sizes and, therefore, their sums had to be equal. However, in modern use, it's not necessary for the productions and attractions to be identical units (e.g., P_i could be population while P_j could be employment).

The total volume of productions (trips) from a single location, i, is estimated by summing over all destination locations, j:

$$T_i = KP_i \sum_{j=1}^{L} \frac{P_j}{d_{ij}} \quad (13.4)$$

where T_i is the number of trip originating from zone i, K is a constant, and L is the number of zones.

Over time, the concept has been generalized and applied to many different types of travel behavior. For example, Huff (1963) applied the concept to retail trade between zones in an urban area using the general form of:

$$T_{ij} = \alpha \frac{A_j^{\beta}}{d_{ij}^{\lambda}} \quad (13.5)$$

where T_{ij} is the number of purchases in location j by residents of location i, A_j is the attractiveness of zone j (e.g., square footage of retail space), d_{ij} is the distance between zones i and j, β is the exponent of S_j, and λ is the exponent of distance, and α is a constant (Bossard, 1993). The distance component, $d_{ij}^{-\lambda}$, is sometimes called an *inverse distance* function. This is a *single constraint* model in that only the attractiveness of a commercial zone is constrained, that is the sum of all attractions for j must equal the total attraction in the region.

Again, it can be generalized to all zones by, first, estimating the total trips generated from one zone, i, to another zone, j:

$$T_{ij} = \alpha \frac{P_i^{\rho} A_j^{\beta}}{d_{ij}^{\lambda}} \quad (13.6)$$

where T_{ij} is the interaction between two locations (or zones), P_i is productions of trips from location/zone i, A_j is the attractiveness of location/zone j, D_{ij} is the distance between zones i and j, β is the exponent of S_j, ρ is the exponent of H_i, λ is the exponent of distance, and α is a constant.

Second, the total number of trips generated by a location, i, to all destinations is obtained by summing over all destination locations, j:

$$T_i = \alpha P_i^{\rho} \sum_{j=1}^{L} \frac{A_j^{\beta}}{d_{ij}^{\lambda}} \quad (13.7)$$

This differs from the traditional gravity function by allowing the exponents of the production from location i, the attraction from location j, and the distance between zones to vary. Typically, these exponents are calibrated on a known sample before being applied to a forecast sample and the locations are usually measured by zones. Thus, retailers in deciding on the location of a new store can use this type of model to choose a site location to optimize travel behavior of patrons. They will, typically, obtain data on actual shopping trips by customers and then calibrate the model on the data, estimating the exponents of attraction and distance. The model can then be used to predict future shopping trips if a facility is built at a particular location.

This type of function is called a *double constraint* model because the balancing constant, *K*, has to be constrained by the number of units in both the origin and destination locations; that is, the sum of P_i over all locations must be equal to the total number of productions while the sum of P_j over all locations must be equal to the total number of attractions. Adjustments are usually required to have the sum of individual productions and attractions equal the totals (usually estimated independently).

The equation can be generalized to other types of trips and different metrics can be substituted for distance, such as travel time, effort, or cost (Isard, 1960). For example, for commuting trips, usually employment is used for attractions, frequently sub-divided into retail and non-retail employment. In addition, for productions, median household income or car ownership percentage is used as an additional production variable. Equation 13.7 can be generalized to include any type of production or attraction variable (13.8 and 10.9):

$$T_{ij} = \alpha_1 P_i^{\rho} \alpha_2 \frac{A_j^{\beta}}{d_{ij}^{\lambda}} \tag{13.8}$$

$$T_i = \alpha_1 P_i^{\rho} \sum_{j=1}^{L} (\alpha_2 \frac{A_j^{\beta}}{d_{ij}^{\lambda}}) \tag{13.9}$$

where T_{ij} is the number of trips produced by location i that travel to location j, P_i is either a single variable associated with trips produced from a zone or the cross-product of two or more variables associated with trips produced from a zone, A_j is either a single variable associated with trips attracted to a zone or the cross-product of two or more variables associated with trips attracted to a zone, d_{ij} is either the distance between two locations or another variable measuring travel effort (e.g., travel time, travel cost), ρ, β, and λ are exponents of the respective terms, α_1 is a constant associated with the productions to ensure that the sum of trips produced by all zones equals the total number of trips for the region (usually estimated independently), and α_2 is a constant associated with the attractions to ensure that the sum of trips attracted to all zones equals the total number of trips for the region. Without having two constants in the equation,

usually conflicting estimates of K will be obtained by balancing the equation against productions or attractions. The summation over all destination locations, j (Equation 13.9), produces the total number of trips from zone i.

Intervening Opportunities

Stouffer (1940) modified the simple gravity function by arguing that the attraction between two locations was a function not only of the characteristics of the relative attractions of two locations, but of intervening opportunities between the locations. His hypothesis, ".assumes that there is no necessary relationship between mobility and distance.. that the number of persons going a given distance is directly proportional to the number of opportunities at that distance and inversely proportional to the number of intervening opportunities"(Stouffer, 1940, p. 846). This model was used in the 1940s to explain interstate and intercounty migration (Bright & Thomas, 1941; Isbell, 1944; Isard, 1979). Using the gravity type formulation, we can write this as:

$$T_{ij} = \alpha \frac{A_j^{\beta}}{\sum_{j=1}^{L} A_k^{\xi} d_{ij}^{\lambda}} \tag{13.10}$$

where T_{ji} is the attraction of location j by residents of location i, A_j is the attractiveness of zone j, A_k is the attractiveness of all other locations that are *intermediate* in distance between locations i and j, d_{ij} is the distance between zones *i* and *j*, β is the exponent of S_j, ξ is the exponent of S_k, λ is the exponent of distance, and α is a constant. While the intervening opportunities are implicit in Equation 13.5 in the exponents, β and λ, and coefficient, K, Equation 13.10 makes the intervening opportunities explicit. The importance of the concept is that the interaction between two locations becomes a complex function of the spatial environment of nearby areas and not just of the two locations.

Urban Transportation Modeling

This type of model is incorporated as a formal step in the urban transportation planning process, implemented by most regional planning organizations in the United States and elsewhere (Stopher & Meyburg, 1975; Krueckeberg & Silvers, 1974; Field & MacGregor, 1987).

The step, called *trip distribution*, is linked to a five step model. First, data are obtained on travel behavior for a variety of trip purposes. This is usually done by sampling households and asking each member to keep a travel diary documenting all their trips over a two or three day period. Trips are aggregated by individuals and by households. Frequently, trips by different purposes are separated. Second, the volume of trips produced by and attracted to zones (called traffic analysis zones) is estimated, usually on the basis of the number of households in the zone and some indicator of income or private vehicle ownership. Third, trips produced by each zone

are distributed to every other zone usually using a gravity-type function (Equation 13.9). That is, the number of trips produced by each origin zone and ending in each destination zone is estimated by a gravity model. The distribution is based on trip productions, trip attractions, and travel 'resistance' (measured by travel distance or travel time). Fourth, zone-to-zone trips are allocated by mode of travel (car, bus, walking, etc); and, fifth, trips are assigned to particular routes by travel mode (i.e., bus trips follow different routes than private vehicle trips). The advantage of this process is that trips are allocated according to origins, destinations, distances (or travel times), modes of travel and routes. Since all zones are modeled simultaneously, all intermediate destinations (i.e., intervening opportunities) are incorporated into the model. Chapters 25-31 present a crime travel demand model, an application of travel demand modeling to crime.

Alternative Distance Decay Functions

One of the problems with the traditional gravity formulation is in the measurement of travel resistance, either distance or time. For locations separated by sizeable distances in space, the gravity formulation can work properly. However, as the distance between locations decreases, the denominator approaches infinity. Consequently, an alternative expression for the interaction has been proposed which uses the negative exponential function (Hägerstrand, 1957; Wilson, 1970):

$$A_{JI} = s_J^{\beta} e^{-\alpha d_{ij}} \tag{13.11}$$

where A_{ji} is the attraction of location j for residents of location i, S_j is the attractiveness of location j, D_{ij} is the distance between locations *i* and *j*, β is the exponent of S_j, *e* is the base of the natural logarithm (i.e., 2.7183..), and α is an empirically-derived exponent. Sometimes known as *entropy maximization*, the latter part of the equation includes a negative exponential function which has a maximum value of 1 (i.e., $e^{-0} = 1$). This has the advantage of making the equation more stable for interactions between locations that are close together. For example, Cliff and Haggett (1988) used a negative exponential gravity-type model to describe the diffusion of measles into the United States from Canada and Mexico. It has also been argued that the negative exponential function generally gives a better fit to urban travel patterns, particularly by automobile (Foot, 1981; Bossard, 1993; NCHRP, 1995).

Other functions have also be used to describe the distance decay - negative linear, normal distribution, lognormal distribution, quadratic, Pareto function, square root exponential, and so forth (Haggett & Arnold, 1965; Taylor, 1970; Eldridge & Jones, 1991). Later in the chapter, we will explore several different mathematical formulations for describing the distance decay. One,

in fact, does not need to use a mathematical function at all, but could empirically describe the distance decay from a large data set and utilize the described values for predictions.

The use of mathematical functions has evolved out of both the Newtonian tradition of gravity as well as various location theories which used the gravity function. A mathematical function makes sense under two conditions: 1) if travel is uniform in all directions; and 2) as an approximation if there is inadequate data from which to calibrate an empirical function. The first assumption is usually wrong since physical geography (i.e., oceans, rivers, mountains) as well as asymmetrical street networks make travel easier in some directions than others. As we shall see below, the distance decay is quite irregular for Journey-to-crime trips and would be better described by an empirical, rather than mathematical function.

In short, there is a long history of research on both the location of places as well as the likelihood of interaction between these places, whether the interaction is freight movement, land prices or individual travel behavior. The gravity model and variations on it have been used to describe the interactions between these locations.

Travel Behavior of Criminals

Journey-to-crime Trips

The application of travel behavior theory to crime has a sizeable history as well. The analysis of distance for Journey-to-crime trips was applied in the 1930s by White (1932), who noted that property crime offenders generally traveled farther distances than offenders committing crimes against people, and by Lottier (1938), who analyzed the ratio of chain store burglaries to the number of chain stores by zone in Detroit. Turner (1969) analyzed delinquency behavior by a distance decay travel function showing how more crime trips tend to be close to the offender's home with the frequency dropping off with distance. Phillips (1980) is, apparently, the first to use the term *Journey-to-crime* is describing the travel distances that offenders make though Harries (1980) noted that the average distance traveled has evolved by that time into an analogy with the journey to work statistic.

Journey-to-crime trips by crime type

Rhodes and Conly (1981) expanded on the concept of a *criminal commute* and showed how robbery, burglary and rape patterns in the District of Columbia followed a distance decay pattern. LeBeau (1987a) analyzed travel distances of rape offenders in San Diego by victim-offender relationships and by method of approach. Boggs (1965) applied the intervening opportunities model in analyzing the distribution of crimes by area in relation to the distribution of offenders. Other empirical descriptions of Journey-to-crime distances and other travel

behavior parameters have been studied by Blumin (1973), Curtis (1974), Repetto (1974), Pyle (1974), Capone and Nichols (1975), Rengert (1975), Smith (1976), LeBeau (1987b), and Canter and Larkin (1993). It has generally been accepted that property crime trips are longer than personal crime trips (LeBeau, 1987a), though exceptions have been noted (Turner, 1969). Also, it would be expected that average trip distances will vary by a number of factors: crime type; method of operation; time of day; and, even, the value of the property realized (Capone & Nichols, 1975).

In more recent years, there have been more focused studies of travel behavior by types of crime: commercial robberies in the Netherlands (Van Koppen and Jansen, 1998); vehicle thefts in Baltimore County (Levine, 2005); robberies in Chicago and confrontations, burglaries, and vehicle thefts in Las Vegas (Block and Helms, 2005); residential burglaries in The Hague (Bernasco and Nieuwbeerta, 2005); homicides in Washington, DC (Groff and McEwen, 2005); bank robberies in Baltimore County (Levine, 2007); robberies in Chicago (Bernasco and Block, 2009); and the trips of drunk drivers involved in crashes in Baltimore County (Levine & Canter, 2011). These studies show substantial variability in crime trip lengths with many trips being long.

Personal characteristics and the journey-to-crime

In addition, there are several studies that have examined the how the personal characteristics of offenders effect their journey-to-crime. In terms of gender, Rengert (1975) found that female offenders were more likely to commit crimes within their own residential area than male offenders, hence making shorter trips, a result supported by Pettiway (1995) and by Groff and McEwen (2005). However, Phillips (1980) found that female offenders traveled longer distances, on average, than male offenders, a result supported by Fritzon (2001) who studied female arsonists.

In terms of age of the offender, several studies (Groff & McEwen, 2005; Snook, Cullen, Mokros, & Harbort, 2005; Bernasco & Nieuwbeerta, 2005; Snook, 2004; Warren, Reboussin, Hazelwood, Cummings, Gibbs, & Trumbetta, 1998) have shown that generally juveniles make shorter trips.

However, none of these studies attempted to control for myriad of factors that affect the journey-to-crime. In a more controlled study, Levine and Lee (2012) examined the interaction of gender and age group for offenders in Manchester, England and found distinct interactions between gender and age group. Juvenile male offenders had the shortest crime trips where adult male offenders had the longest. Female offenders, both juveniles and adults, had moderately long crime trips, though not as long as the adult males. However, a much higher proportion of crime trips by females went to commercial areas, in particular the town centre in Manchester.

Modeling the Offender Search Area

Conceptual work on the type of model have been made by Brantingham and Brantingham (1981) who analyzed the *geometry of crime* and conceptualized a criminal search area, a geographical area modified by the spatial distribution of potential offenders and potential targets, the awareness spaces of potential offenders, and the exchange of information between potential offenders. In this sense, their formulation is similar to that of Stouffer (1940), who described intervening opportunities, though their's is a behavioral framework. An important concept developed by the Brantingham's is that of decreased criminal activity near to an offender's home base, a sort of a safety area around their near neighborhood. Presumably, offenders, particularly those committing property crimes, go a little way from their home base so as to decrease the likelihood that they will get caught. This was noted by Turner (1969) in his study of delinquency in Philadelphia. Thus, the Brantingham's postulated that there would be a small safety area (or 'buffer' zone) of relatively little offender activity near to the offender's base location; beyond that zone, however, they postulated that the number of crime trips would decrease according to a distance decay model (the exact mathematical form was never specified, however).

Crime trips may not even begin at an offender's residence. Routine activity theory (Felson, 2002; Cohen & Felson, 1979) suggests that crime opportunities appear in the activities of everyday life. The routine patterns of work, shopping, and leisure affect the convergence in time and place of would be offenders, suitable targets, and absence of guardians. Many crimes may occur while an offender is traveling from one activity to another. Thus, modeling crime trips as if they are referenced relative to a residence is not necessarily going to lead to better prediction.

The mathematics of Journey-to-crime has been modeled by Rengert (1981) using a modified general opportunities model:

$$P_{ij} = KU_i V_j f(d_{ij}) \qquad (13.12)$$

where P_{ij} is the probability of an offender in location (or zone) i committing an offense at location j, U_i is a measure of the number of crime trips produced at location i (what Rengert called *emissiveness*), V_j is a measure of the number of crime targets (attractiveness) at location j, and $f(D_{ij})$ is an unspecified function of the cost or effort expended in traveling from location i to location j (distance, time, cost). He did not try to operationalize either the production side or the attraction side. Nevertheless, conceptually, a crime trip would be expected to involve both elements as well as the cost of the trip.

In short, there has been a great deal of research on the travel behavior of criminals in committing acts as well as a number of statistical formulations.

Predicting the Location of Serial Offenders

The Journey-to-crime formulation, as in Equation 13.9, has been used to estimate the origin location of a serial offender based on the distribution of crime incidents. The logic is to plot the distribution of the incidents and then use a property of that distribution to estimate a likely origin location for the offender. Inspecting a pattern of crimes for a central location is an intuitive idea that police departments have used for a long time. The distribution of incidents describes an activity area by an offender, who often lives somewhere in the center of the distribution. It is a *sample* from the offender's activity space. Using the Brantingham's terminology, there is a search area by an offender within which the crimes are committed; most likely, the offender also lives within the search area.

For example, Canter (1994) shows how the area defined by the distribution of the 'Jack the Ripper' murders in the east end of London in the 1880s included the key suspects in the case (though the case was never solved). Kind (1987) analyzed the incident locations of the 'Yorkshire Ripper' who committed thirteen murders and seven attempted murders in northeast England in the late 1970s and early 1980s. Kind applied two different geographical criteria to estimate the residential location of the offender. First, he estimated the center of minimum distance. Second, on the assumption that the locations of the murders and attempted murders that were committed late at night were closer to the offender's residence, he graphed the time of the offense on the Y axis against the month of the year (taken as a proxy for length of day) on the X axis and plotted a trend line through the data to account for seasonality. Both the center of minimum distance and the murders committed at a later time than the trend line pointed towards the Leeds/Bradford area, very close to where the offender actually lived (in Bradford).

There are several alternative models that have been proposed for Journey-to-crime modeling. The major ones are discussed in depth in Attachment A at the end of the chapter.

Geographic Profiling

Journey-to-crime estimation should be distinguished from *geographical profiling*. Geographical profiling involves understanding the geographical search pattern of criminals in relation to the spatial distribution of potential offenders and potential targets, the awareness spaces of potential offenders including the labeling of 'good' targets and crime areas, and the interchange of information between potential offenders who may modify their awareness space (Brantingham & Brantingham, 1981). According to Rossmo:

> "..Geographic profiling focuses on the probable spatial behaviour of the offender within the context of the locations of, and the spatial relationships between, the various crime sites. A psychological profile provides insights into an offender's likely motivation,

behaviour and lifestyle, and is therefore directly connected to his/her spatial activity. Psychological and geographic profiles thus act in tandem to help investigators develop a picture of the person responsible for the crimes in question" (Rossmo, 1997).

In other words, geographic profiling is a framework for understanding how an offender traverses an area in searching for victims or targets; this, of necessity, involves understanding the social environment of an area, the way that the offender understands this environment (the 'cognitive map') as well as the offender's motives.

On the other hand, Journey-to-crime estimation follows a much simpler logic involving the distance dimension of the spatial patterning of a criminal. It is a method aimed at estimating the distance that serial offenders will travel to commit a crime and, by implication, the likely location from which they started their crime 'trip'. In short, it is a strictly statistical approach to estimating the residential whereabouts of an offender compared to understanding the dynamics of serial offenders.

It remains an empirical question whether a conceptual framework, such as geographic profiling, can predict better than a strictly statistical framework. Understanding a phenomenon, such as serial murders, serial rapists, and so forth, is an important research area. We seek more than just statistical prediction in building a knowledge base. However, it does not necessarily follow that understanding produces better predictions. In many areas of human activity, strictly statistical models are better at predicting than explanatory models. I will return to this point later in the section.

The *CrimeStat* Journey-to-crime Routine

The Journey-to-crime (Jtc) routine is a diagnostic designed to aid police departments in their investigations of serial offenders. The aim is to estimate the likelihood that a serial offender lives at any particular location. Using the location of incidents committed by the serial offender, the program makes statistical guesses at where the offender is liable to live, based on the similarity in travel patterns to a known sample of serial offenders for the same type of crime. The *Jtc* routine builds on the Rossmo (1993a; 1993b; 1995) framework, but extends its modeling capability.

1. A grid is overlaid on top of the study area. This grid can be either imported or can be generated by *CrimeStat* (see Chapter 3). The grid represents the entire study area. There is no optimal study area. The technique will model that which is defined. Thus, the user has to select an area intelligently.

2. The routine calculates the distance between each incident location committed by a serial offender (or group of offenders working together) and each cell, defined by the centroid of the cell. Rossmo (1993a; 1995) used indirect (Manhattan) distances. However, this would be appropriate only when a city falls on a uniform grid. The *Jtc* routine allows direct, indirect or network distances. These are defined on the Measurement Parameters page (see Chapter 3) In most cases, direct distances would be the most appropriate choice as a police department would normally locate origin and destination locations rather than particular routes that are taken (see below).

3. A distance decay function is applied to each grid cell-incident pair and sums the values over all incidents. The user has a choice whether to model the travel distance by a mathematical function or an empirically-derived function.

4. The resultant of the distance decay function for each grid cell-incident pair is summed over all incidents to produce a likelihood (or density) estimate for each grid cell.

5. In both cases, the program outputs the two results: 1) the grid cell which has the peak likelihood estimate; and 2) the likelihood estimate for every cell. The latter output can be saved as a *Surfer® for Windows* 'dat', *ArcGIS Spatial Analyst©* 'asc', ASCII 'grd', *ArcGIS®* '.shp', *MapInfo®* '.mif' , or as an Ascii grid 'grd' file which can be read by many GIS packages (e.g., *Vertical Mapper©*). These files can also be read by other GIS packages (e.g., *Maptitude*).

Figure 13.1 shows the logic of the routine and Figure 13.2 shows the Journey-to-crime (Jtc) screen. There are two parts to the routine. First, there is a calibration model that is used in the empirically-derived distance function based on a large sample of crime trips by offenders. Second, there is the Journey-to-crime (Jtc) model for estimating the likely origin location of a single serial offender. To estimate the function, the user can select either the already-calibrated distance function or the mathematical function. The empirically-derived function is, by far, the easiest to use and is, consequently, the default choice in *CrimeStat*. It is discussed below. However, the mathematical function can be used if there is inadequate data to construct an empirical distance decay function or if a particular form is desired.

Figure 13.1:

Logic of Journey to Crime Interpolation Routine

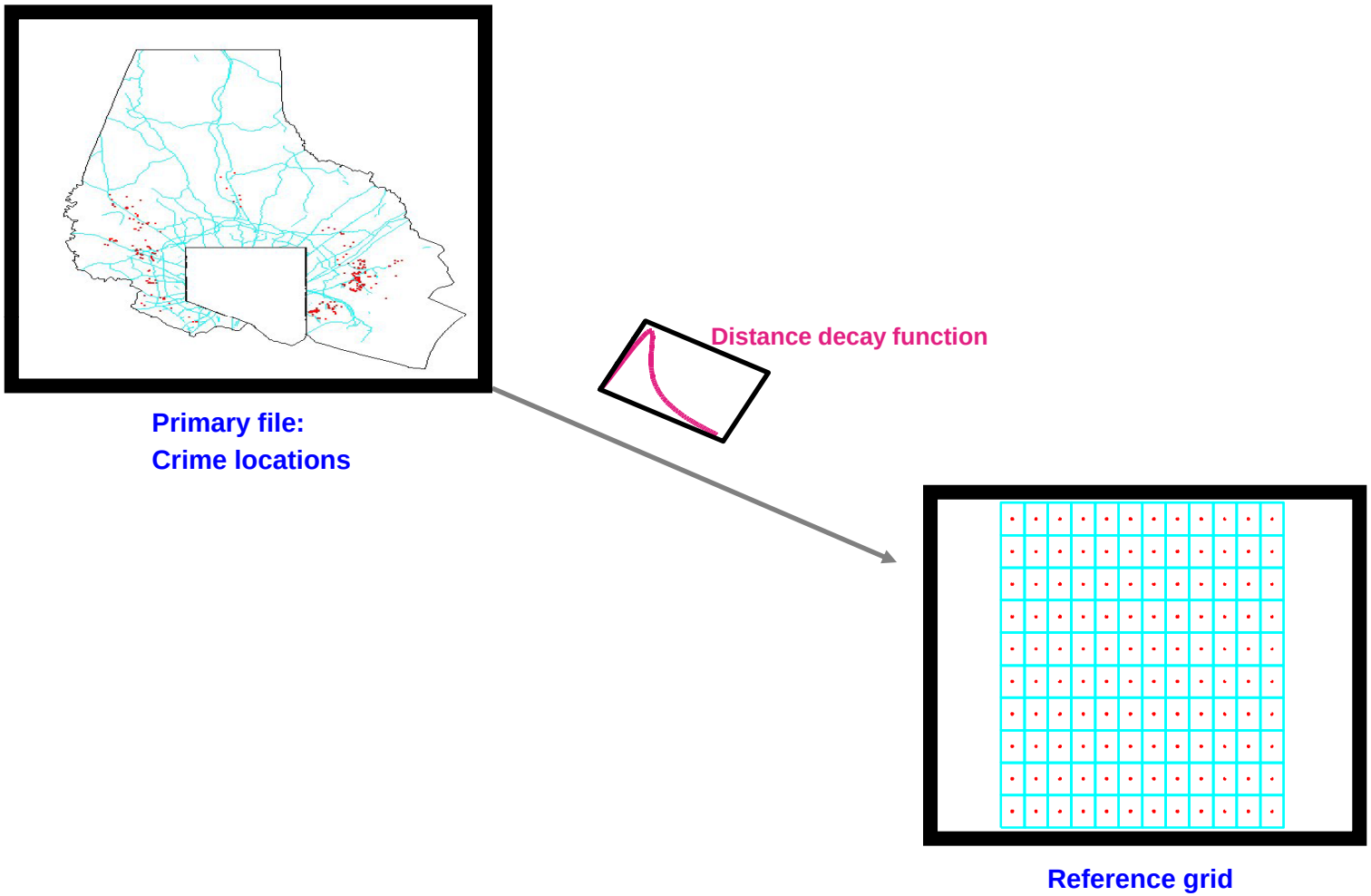
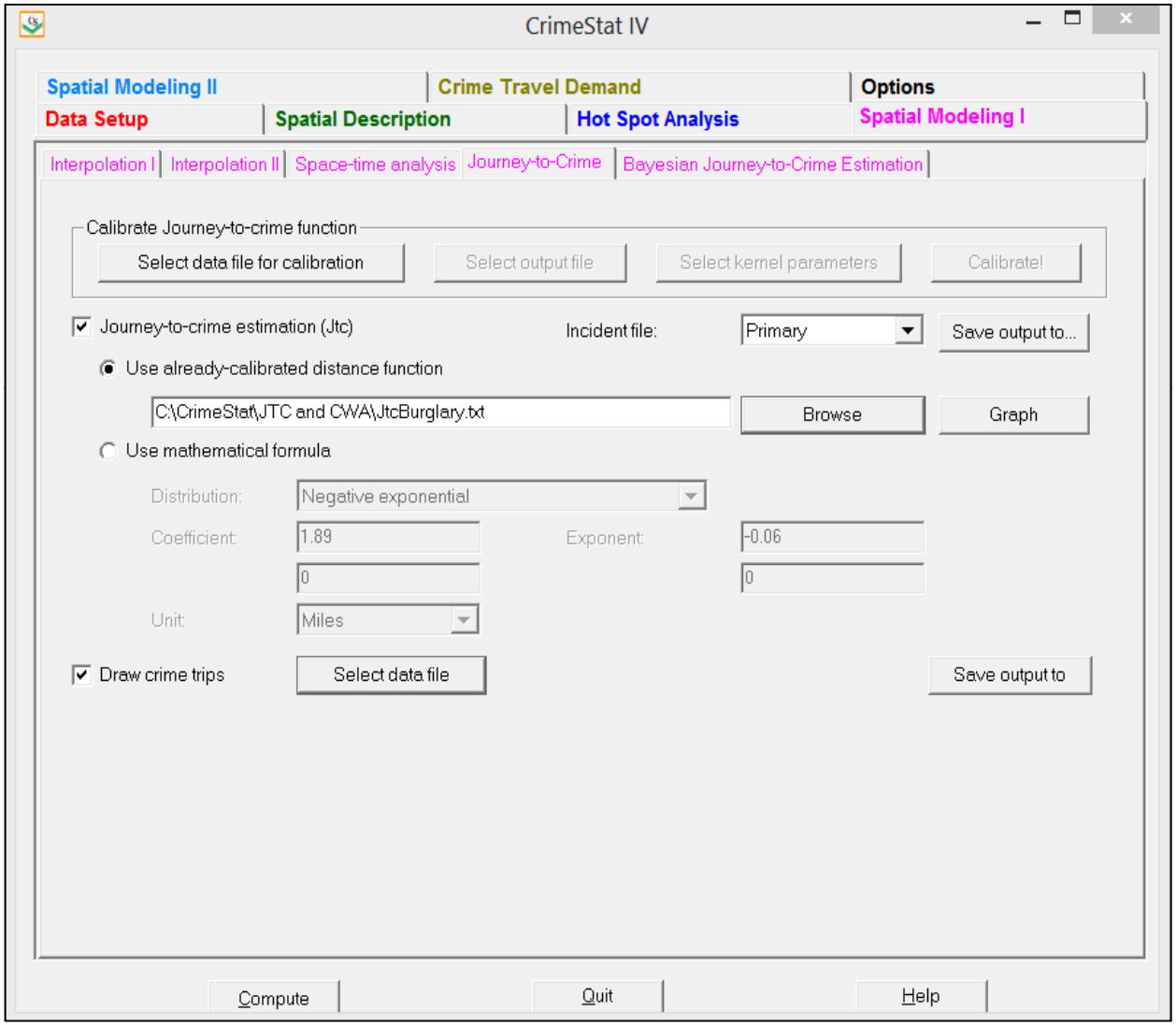


Figure 13.2:

Journey-to-crime Screen



Journey-to-crime Estimation Using Mathematical Functions

Let us start by illustrating the use of the mathematical functions because this has been the traditional way that distance decay has been examined. The *CrimeStat* Jtc routine allows the user to define distance decay by a mathematical function.

Probability Distance Functions

The user selects one of five **probability density distributions** to define the likelihood that the offender has traveled a particular distance to commit a crime. The advantage of having five functions, as opposed to only one, is that it provides more flexibility in describing travel behavior. The travel distance distribution followed will vary by crime type, time of day, method of operation, and numerous other variables. The five functions allow an approach that can simulate more accurately travel behavior under different conditions. Each of these has parameters that can be modified, allowing a very large number of possibilities for describing travel behavior of a criminal.

Figure 13.3 illustrates the five types.[1] Default values based on Baltimore County have been provided for each. The user, however, can change these as needed. Briefly, the five functions are:

Linear

The simplest type of distance model is a linear function. This model postulates that the likelihood of committing a crime at any particular location declines by a constant amount with distance from the offender's home. It is highest near the offender's home but drops off by a constant amount for each unit of distance until it falls to zero. The form of the linear equation is:

$$f(d_{ij}) = A + Bd_{ij} \tag{13.14}$$

---

[1] There are, of course, many other types of mathematical functions that can be used to describe a declining likelihood with distance. However, the five types of functions presented here are commonly used. We avoided the inverse distance function because of its potential to distort the likelihood relationship:

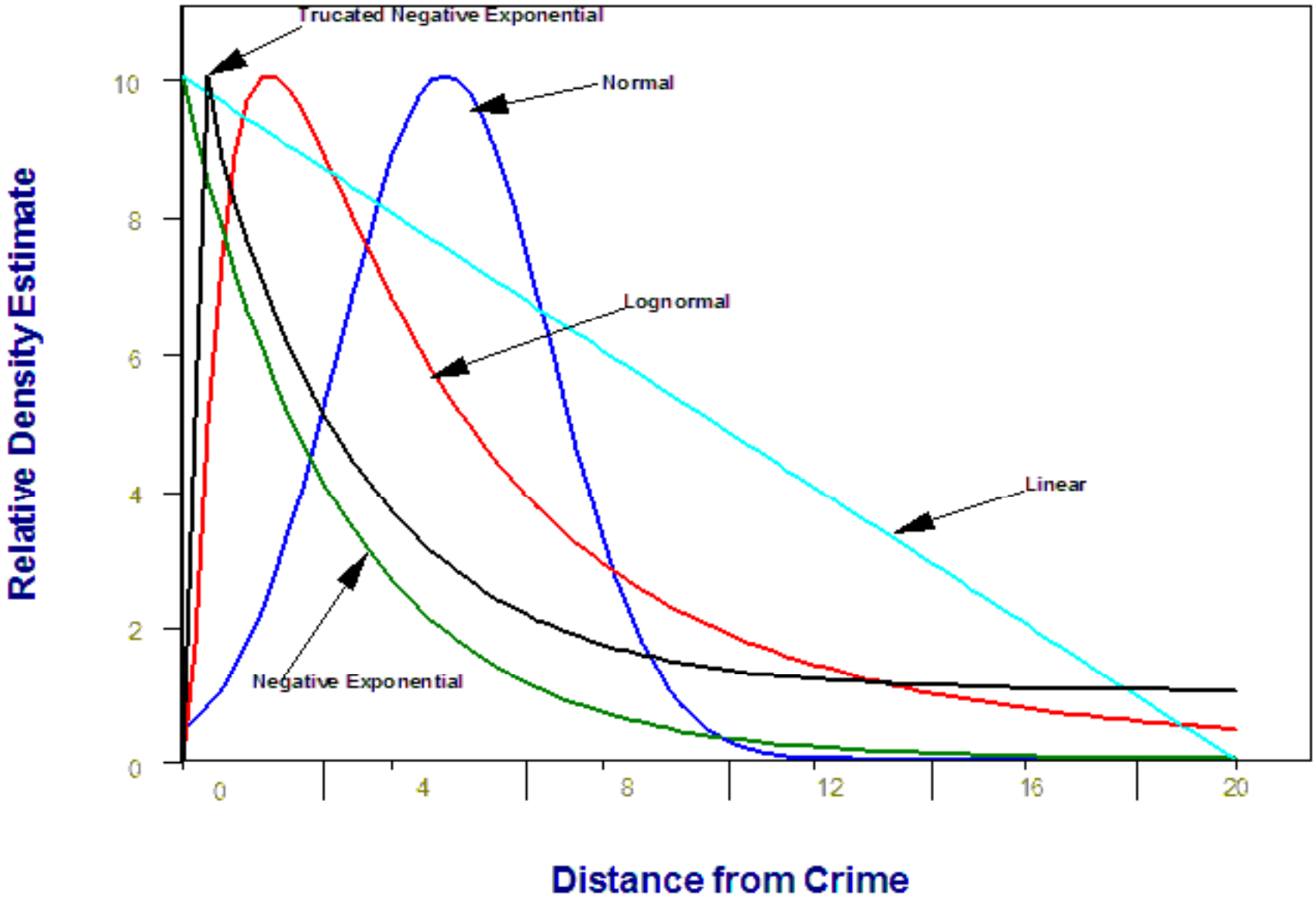
$$f(d) = \frac{1}{d_{ij}^{k}} \tag{13.13}$$

where *k* is a power (e.g., 1, 2, 2.5). For large distances, this function can be a useful approximation of the lessening travel interaction with distance. However, for short distances, the function goes towards infinity as the distance approaches zero. In fact, for $d_{ij} = 0$, the function is unsolvable. Since many distances between reference cells and incidents will be zero or close to zero, the function becomes unusable.

Figure 13.3:

Journey-to-crime Travel Demand Functions

Five Mathematical Functions



where $f(d_{ij})$ is the likelihood that the offender will commit a crime at a particular location, *i*, defined here as the center of a grid cell, d_{ij} is the distance between the offender's residence and location *i*, *A* is a slope coefficient which defines the fall off in distance, and *B* is a constant. It would be expected that the coefficient *B* would have a negative sign since the likelihood should decline with distance. The user must provide values for *A* and *B*. The default for A is 1.9 and for B is -0.06. This function assumes no buffer zone around the offender's residence. When the function reaches 0 (the X axis), the routine automatically substitutes a 0 for the function.

Negative Exponential

A slightly more complex function is the negative exponential. In this type of model, the likelihood is also highest near the offender's home and drops off with distance. However, the decline is at a ***constant rate*** of decline, thus dropping quickly near the offender's home until is approaches zero likelihood. The mathematical form of the negative exponential is

$$f(d_{ij}) = Ae^{-Bd_{ij}} \tag{13.15}$$

where $f(d_{ij})$ is the likelihood that the offender will commit a crime at a particular location, i, defined here as the center of a grid cell, d_{ij} is the distance between each reference location and each crime location, e is the base of the natural logarithm, A is the coefficient and B is an exponent of e. The user inputs values for A - the coefficient, and B - the exponent. The default for A is 1.89 and for B is -0.06. This function is similar to the Canter model (discussed in Attachment A) except that the coefficient is calibrated. Also, like the linear function, it assumes no buffer zone around the offender's residence.

Normal

A normal distribution assumes the peak likelihood is at some optimal distance from the offender's home base. Thus, the function rises to that distance and then declines. The rate of increase prior to the optimal distance and the rate of decrease from that distance is symmetrical in both directions. The mathematical form is:

$$Z_{ij} = \frac{(d_{ij} - \bar{d})}{S_d} \tag{13.16}$$

$$f(d_{ij}) = A\frac{1}{S_d\sqrt{2\pi}}e^{-\frac{Z_{ij}^2}{2}} \tag{13.17}$$

where $f(d_{ij})$ is the likelihood that the offender will commit a crime at a particular location, i (defined here as the center of a grid cell), d_{ij} is the distance between each reference location and each crime location, $\bar{d}$ is the mean distance input by the user, S_d is the standard deviation of distances, **e** is the base of the natural logarithm, and A is a coefficient. The user inputs values for $\bar{d}$, S_d, and A. The default values are 4.2 for the mean distance, $\bar{d}$, 4.6 for the standard deviation, S_d, and 29.5 for the coefficient, A.

By carefully scaling the parameters of the model, the normal distribution can be adapted to a distance decay function with an increasing likelihood for near distances and a decreasing likelihood for far distances. Choosing a standard deviation greater than the mean (e.g., $\bar{d}$ = 1,S_d = 2) will skew the distribution to the left. The function becomes similar to the model postulated by Brantingham and Brantingham (1981) in that it is a single function which describes travel behavior.

Lognormal

The lognormal function is similar to the normal except it is more skewed, either to the left or to the right. It has the potential of showing a very rapid increase near the offender's home base with a more gradual decline from a location of peak likelihood (see Figure 13.3). It is also similar to the Brantingham and Brantingham (1981) model. The mathematical form of the function is:

$$f(d_{ij}) = A\frac{1}{d_{ij}^2 S_d\sqrt{2\pi}}e^{-\frac{[Ln(d_{ij}^2-\bar{d},)^2}{2S_d^2}} \qquad (13.18)$$

where $f(d_{ij})$ is the likelihood that the offender will commit a crime at a particular location, i, defined here as the center of a grid cell, d_{ij} is the distance between each reference location and each crime location, $\bar{d}$ is the mean distance, S_d is the standard deviation of distances, ***e*** is the base of the natural logarithm, and A is a coefficient. The user inputs $\bar{d}$, S_d , and A. The default values are 4.2 for the mean distance, $\bar{d}$, 4.6 for the standard deviation, S_d, and 8.6 for the coefficient, A. They were calculated from the Baltimore County data (see Table 13.3).

Truncated Negative Exponential

The truncated negative exponential is a joined function made up of two distinct mathematical functions - the linear and the negative exponential. For the near distance, a positive linear function is defined, starting at zero likelihood for distance 0 and increasing to d_p, a location of peak likelihood. Thereupon, the function follows a negative exponential, declining quickly with distance. The two mathematical functions making up this spline function are

Linear: $f(d_{ij}) = 0 + Bd_{ij} = Bd_{ij}$ for $d_{ij} \geq 0$, $d_ij \leq d_p$ (13.19)

Negative Exponential: $f(d_{ij}) = Ae^{-Cd_{ij}}$ for $X_i > d_p$ (13.20)

where d_{ij} is the distance from the home base, B is the slope of the linear function and for the negative exponential function A is a coefficient and C is an exponent. Since the negative exponential only starts at a particular distance, d_p, A, is assumed to be the intercept *if* the Y-axis were transposed to that distance. Similarly, the slope of the linear function is estimated from the Cutoff distance, d_p, by a peak likelihood function. The default values are 0.4 for the Cutoff distance, d_p, 13.8 for the peak likelihood, and -0.2 for the exponent, C. Again, these were calculated with Baltimore County data.

This function is the closest approximation to the Rossmo model (see Attachment A). However, it differs in several mathematical properties. First, the 'near home base' function is linear (Equation 13.19), rather than a non-linear function. It assumes a simple increase in travel likelihoods by distance from the home base, up to the edge of the safety zone.[2] Second, the distance decay part of the function (Equation 13.20) is a negative exponential, rather than an inverse distance function (see Attachment A); consequently, it is more stable when distances are very close to zero (e.g., for a crime where there is no 'near home base' offset).

Calibrating an Appropriate Probability Distance Function

The mathematics are relatively straightforward. However, how does one know which distance function to use? The answer is to get some data and calibrate it. It is important to obtain data from a sample of known offenders where both their residence at the time they committed crimes as well as the crime locations are known. This is called the *calibration data set*. Each of the models are then tested against the calibration data set using an approach similar

---

2 There are, of course, many other types of mathematical functions that can be used to describe a declining likelihood with distance. However, the five types of functions presented here are commonly used. We avoided the inverse distance function because of its potential to distort the likelihood relationship:

$$f(d) = \frac{1}{d_{ij}^k} \quad (13.21)$$

where *k* is a power (e.g., 1, 2, 2.5). For large distances, this function can be a useful approximation of the lessening travel interaction with distance. However, as the distance between the reference cell location and an incident location becomes very small, approaching zero, then the likelihood estimate becomes very large, approaching infinity. In fact, for $d_{ij} = 0$, the function is unsolvable. Since many distances between reference cells and incidents will be zero or close to zero, the function becomes unusable.

to that explained below. An error analysis is conducted to determine which of the models best fits the data. Finally, the 'best fit' model is used to estimate the likelihood that a particular serial offender lives at any one location. Though the process is tedious, once the parameters are calculated they can be used repeatedly for predictions.

Because every jurisdiction is unique in terms of travel patterns, it is important to calibrate the parameters for the particular jurisdiction. While there may be some similarities between cities (e.g., Eastern "centralized" cities v. Western "automobile" cities), there are always unique travel patterns defined by the population size, historical road pattern, and physical geography. Consequently, it is necessary to calibrate the parameters anew for each new city. Ideally, the sample should be a large enough so that a reliable estimate of the parameters can be obtained. Further, the analyst should check the errors in each of the models to ensure that the best choice is used for the Jtc routine. However, once it has been completed, the parameters can be re-used for many years and only periodically re-checked.

Example of Calibrating a Journey-to-crime Estimate with a Mathematical Function

I will illustrate the calibration of a journey-to-crime probability estimate using a mathematical function with data from Baltimore County, MD. The steps in calibrating the *Jtc* parameters were as follows:

1. 49,083 matched arrest and incident records from 1992 through 1997 were obtained in order to provide data on where the offender lived in relation to the crime location for which they were arrested.[3]

2. The data set was checked to ensure that there were X and Y coordinates for both the arrested individual's residence location and the crime incident location for which the individual was being charged. The data were cleaned to eliminate duplicate records or entries for which either the offender's residence or the incident location were missing. The final data set had 41,424 records. There were many multiple records for the same offender since an individual can commit

3 There are several sources of error associated with the data set. First, these records were arrest records prior to a trial. Undoubtedly, some of the individuals were incorrectly arrested. Second, there are multiple offenses. In fact, more than half the records were for individuals who were listed two or more times in the database. The travel pattern of repeat offenders may be slightly different than for apparent first-time offenders (see Figure 13.22). Third, many of these individuals have lived in multiple locations. Considering that many are young and that most are socially not well adjusted, it would be expected that these individuals would have multiple homes. Thus, the distribution of incidents could reflect multiple home bases, rather than one. Unfortunately, the data we have only gives a single residential location, the place at which they were living when arrested.

more than one crime. In fact, more than half the records involved individuals who were listed two or more times. The distribution of offenders by the number of offenses for which they were charged is seen in Table 13.1. As would be expected, a small proportion of individuals account for a sizeable proportion of crimes; approximately 30% of the offenders in the database accounted for 56% of the incidents.

3. The data were imported into a spreadsheet, but a database program could equally have been used. For each record, the direct distance between the arrested individual's residence and the crime incident location was calculated. Chapter 3 presented the formulas for calculating direct distances between two locations and are repeated in endnote ***i***.

Table 13.1

Number of Offenders and Offenses in Baltimore County: 1993-97

(**Journey-to-crime Database)**

Number of Offenses	**Number of Individuals**	**Percent of Offenders**	**Number of Incidents**	**Percent of Incidents**
1	18,174	70.0%	18,174	43.9%
2	4,443	17.1%	8,886	21.5%
3	1,651	6.4%	4,953	12.0%
4	764	2.9%	3,056	7.4%
5	388	1.5%	1,940	4.7%
6-10	482	1.9%	3,383	8.2%
11-15	61	0.2%	757	1.8%
16-20	10	<0.0%	175	0.4%
21-25	3	<0.0%	67	0.2%
26-30	0	<0.0%	0	0.0%
30+	1	<0.0%	33	<0.0%
	25,977		41,424	

4. The records were sorted into sub-groups based on different types of crimes. Table 13.2 presents the categories with their respective sample sizes. Of course, other sub-groups could have been identified. Each sub-group was saved as a separate file. The same records can be part of multiple files (e.g., a record could be included in the 'all robberies' file as well as in the 'commercial robberies' file). All records were included in the 'all crimes' file.

Table 13.2:

Baltimore County Files Used for Calibration: 1993-97

Crime Type	Sample Size
All crimes	41,426
Homicide	137
Rape	444
Assault	8,045
Robbery (all)	3,787
Commercial robbery	1,193
Bank robbery	176
Burglary	4,694
Motor vehicle theft	2,548
Larceny	19,806
Arson	338

5. For each type of crime, the file was grouped into distance intervals of 0.25 miles each. This involved two steps. First, the distance between the offender's residence and the crime location was sorted in ascending order. Second, a frequency distribution was conducted on the distances and grouped into 0.25 mile intervals (often called *bins*). The degree of precision in distance would depend on the size of the data set. For 41,426 records, quarter mile bins were appropriate.

6. For each type of crime, a new file was created which included only the frequency distribution of the distances broken down into quarter mile distance intervals, d_i.

7. In order to compare different types of crimes, each of which will have different frequency distributions, two new variables were created. First, the frequency in the interval was converted into the percentage of all crimes of in each interval by dividing the frequency by the total number of incidents, N, and multiplying by 100. Second, the distance interval was adjusted. Since the interval is a range with a starting distance and an ending distance but has been identified by spreadsheet program as the beginning distance only, a small fraction, representing the midpoint of the interval, is added to the distance interval. In our case, since each interval is 0.25 miles wide, the adjustment is half of this, 0.125. Each new file, therefore, had four variables: the interval distance, the adjusted interval distance, the frequency of incidents within the interval (the number of cases falling into the interval), and the percentage of all crimes of that type within the interval.

8. Using the OLS regression program in the regression module (see Chapter 15), a series of regression equations was set up to model the frequency (or the percentage) as a function of distance. In this case, I used our routines, but other statistical packages could equally have been used. Again, because comparisons between different types of crimes were of interest, the percentage of crimes (by type) within an interval was used as the dependent variable (and was defined as a percentage, i.e., 11.51% was recorded as 11.51). Five equations testing each of the five models were set up.

Estimating Parameter Values Using Grouped Data

The parameters of the function can be estimated from the grouped data.

Linear

For the linear function, the test is:

$$Pct_i = A + Bd_i \tag{13.22}$$

where Pct_i is the percentage of all crimes of that type falling into interval i, d_i is the distance for interval i, A is the intercept, and B is the slope. A and B are estimated directly from the regression equation.

Negative Exponential

For the negative exponential function, the variables have to be transformed to estimate the parameters. The function is:

$$Pct_i = Ae^{-Bd_i} \tag{13.23}$$

A new variable is defined which is the natural logarithm of the percentage of all crimes of that type falling into the interval, $ln(Pct_i)$. This term was then regressed against the distance interval, d_i:

$$Ln(\mathrm{Pct_i}) = \mathrm{K} - \mathrm{Bd_i} \tag{13.24}$$

However, since the original equation has been transformed into a log function, B is the coefficient and A can be calculated directly from:

$$Ln(\mathrm{Pct_i}) = \mathrm{Ln(A)} - \mathrm{Bd_i} \tag{13.25}$$

$$A = e^{K} \tag{13.26}$$

If the percentage in any bin is 0 (i.e., $Pct_i = 0$), then a value of -16 is taken since the natural logarithm of 0 cannot be solved (it approximates -16 as the percentage approaches 0.0000001).

Normal

For the normal function, a more complex transformation must be used. The normal function in the model is:

$$Pct_i = A\frac{1}{S_d\sqrt{2\pi}}e^{-\frac{z_{ij}^2}{2}} \tag{13.27}$$

First, a standardized *Z* variable for the distance, d_i, is created:

$$Z_i = \frac{(d_i - \bar{d})}{S_d} \tag{13.28}$$

where $\bar{d}$ is the mean distance and S_d is the standard deviation of distance. These are calculated from the original data file (*before* creating the file of frequency distributions). Second, a normal transformation of Z is constructed with:

$$Normal(Z_i) = \frac{1}{S_d\sqrt{2\pi}}e^{-\frac{z_{ij}^2}{2}} \tag{13.29}$$

Finally, the normalized variable is regressed against the percentage of all crimes of that type falling into the interval, Pct_i with *no* constant

$$Pct_i = A * Normal(Z_i) \tag{13.30}$$

A is estimated by the regression coefficient.

Lognormal

For the lognormal function, another complex transformation must be done. The lognormal function for the percentage of all crimes of a type for a particular distance interval is:

$$Pct_i = A\frac{1}{d_{ij}^2 S_d\sqrt{2\pi}}e^{-\frac{(Ln(d_i^2)-\bar{d})^2}{2S_d^2}} \quad (13.31)$$

The transformation can be created in steps. First, create L:

$$L = Ln(d_i^2) \quad (13.32)$$

Second, create M:

$$M = (L - \bar{d})^2 \quad (13.33)$$

Third, create O:

$$O = \frac{M}{2S_d^2} \quad (13.34)$$

Fourth, create P by raising **e** to the Oth power.

$$P = e^{-O} \quad (13.35)$$

Fifth, create the lognormal conversion, Lnormal:

$$Lnormal(d_i) = A\frac{1}{d_i^2 S_d\sqrt{2\pi}}P \quad (13.36)$$

Finally, the lognormal variable is regressed against the percentage of all crimes of that type falling into the interval, Pct_i with *no* constant:

$$Pct_i = A * Lnormal(d_i) \quad (13.37)$$

A is estimated with the regression coefficient.

Truncated Negative Exponential

For the truncated negative exponential function, two models were set up. The first applied to the distance range from 0 to the distance at which the percentage (or frequency) is highest, $Maxd_i$. The second applied to all distances greater than this distance:

Linear: $Pct_i = A + Bd_i$ for $0 \leq d_i \leq Cutoff\ d_i$ (13.38)

Negative
Exponential: $Pct_i = Ae^{-Cd_i}$ for $d_i > Cutoff\ d$ (13.39)

To use this function, the user specifies the distance at which the peak likelihood occurs (*Cutoff d)* and the value for that peak likelihood, P (the *peak likelihood*). For the negative exponential function, the user specifies the exponent, C.

In order to splice the two equations together (the spline), the *CrimeStat* truncated negative exponential routine starts the linear equation at the origin and ends it at the highest value. Thus,

$$A = 0 \tag{13.40}$$

$$B = \frac{P}{Cutoff\ d} \tag{13.41}$$

where P is the peak likelihood and Cutoff d is the cutoff distance at which the probability is highest.

The exponent, C, can be estimated by transforming the dependent variable, Pct_i, as in the negative exponential above (Equation 13.23) and regressing the natural log of the percentage ($\ln(Pct_i)$ against the distance interval, d_i, *only* for those intervals that are greater than the Cutoff distance. I have found that estimating the transformed equation with a coefficient, A in:

$$Pct_i = Ae^{-Cd_i} \tag{13.42}$$

$$Ln(Pct_i) = Ln(A) - Cd_i \tag{13.43}$$

gives a better fit to the equation. However, the user need only input the exponent, C, in the Jtc routine as the coefficient, A, of the negative exponential is calculated internally to produce a distance value at which the peak likelihood occurs. The formula is:

$$A = e^{Ln(P)+C(Cutoff\ d-d_i)} \tag{13.44}$$

where P is the peak likelihood, d_p is the distance for the peak likelihood, C is an exponent (assumed to be positive) and d_i is the distance interval for the histogram.

9. Once the parameters for the five models have been estimated, they can be compared to see which one is best at predicting the travel behavior for a particular

type of crime. It is to be expected that different types of crimes will have different optimal models and that the parameters will also vary.

Example from Baltimore County, MD

Let us illustrate with the Baltimore County, MD data. Figure 13.4 shows the frequency distribution for all types of crime in Baltimore County. As can be seen, at the nearest distance interval (0 to 0.25 miles with an assigned 'adjusted' midpoint of 0.125 miles), about 6.9% of all crimes occur within a quarter mile of the offender's residence (it can be seen on the Y-axis). However, for the next interval (0.25 to 0.50 miles with an assigned midpoint of 0.375 miles), almost 10% of all crimes occur at that distance (9.8%). In subsequent intervals, however, the percentage decreases, a little less than 6% for 0.50 to 0.75 miles (with the midpoint being 0.625 miles), a little more than 4% for 0.75 to 1 mile (the midpoint is 0.875 miles), and so forth.

The best fitting statistical function was the negative exponential. The particular equation was:

$$Pct_i = 5.575e^{-0.229d_i} \tag{13.45}$$

This is shown with the solid line. As can be seen, the fit is good for most of the distances, though it underestimates at close to zero distance and overestimates from about a half mile to about four miles. There is only slight evidence of decreased activity near to the location of the offender.

However, the distribution varies by type of crime. With the Baltimore County data, property crimes, in general, occur farther away than personal crimes. The truncated negative exponential generally fit property crimes better, lending support for the Brantingham and Brantingham (1981) framework for these types. For example, larceny offenders have a definite safety zone around their residence (Figure 13.5). Fewer than 2% of larceny thefts occur within a quarter mile of the offender's residence. However, the percentage jumps to about 4.5% from a quarter mile to a half. The truncated negative exponential function fits the data reasonably well though it overestimates from about 1 to 3 miles and underestimates from about 4 to12 miles.

Similarly, motor vehicle thefts show decreased activity near the offender's resident, though it is less pronounced than larceny theft. Figure 13.6 shows the distribution of motor vehicle thefts and the truncated negative exponential function which was fit to the data. The fit is reasonably good though it tends to underestimate middle range distances (3-12 miles).

Figure 13.4:

Journey-to-crime Distances: All Crimes

Negative Exponential Distribution

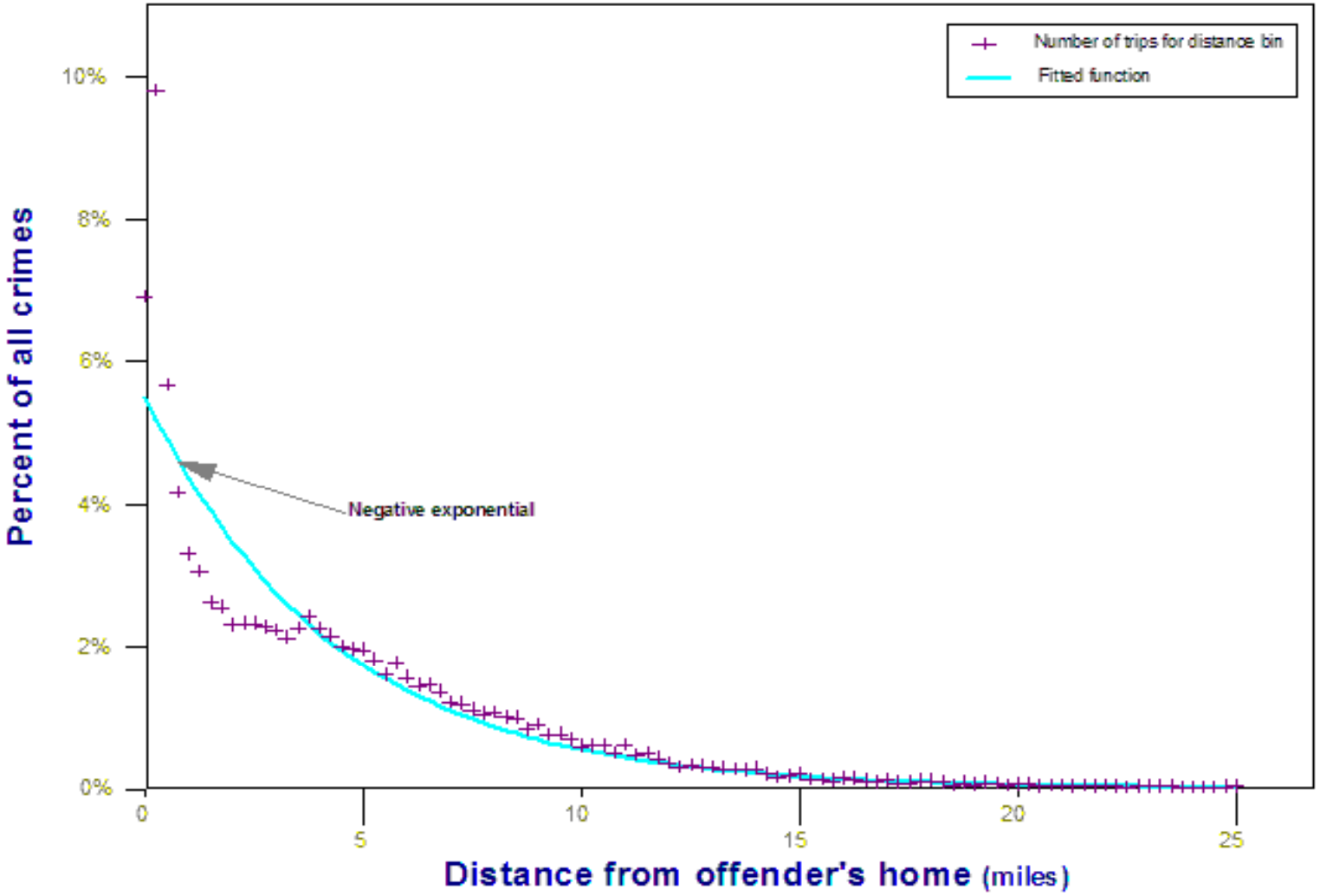


Figure 13.5:

Journey-to-crime Distances: Larceny

Truncated Negative Exponential Function

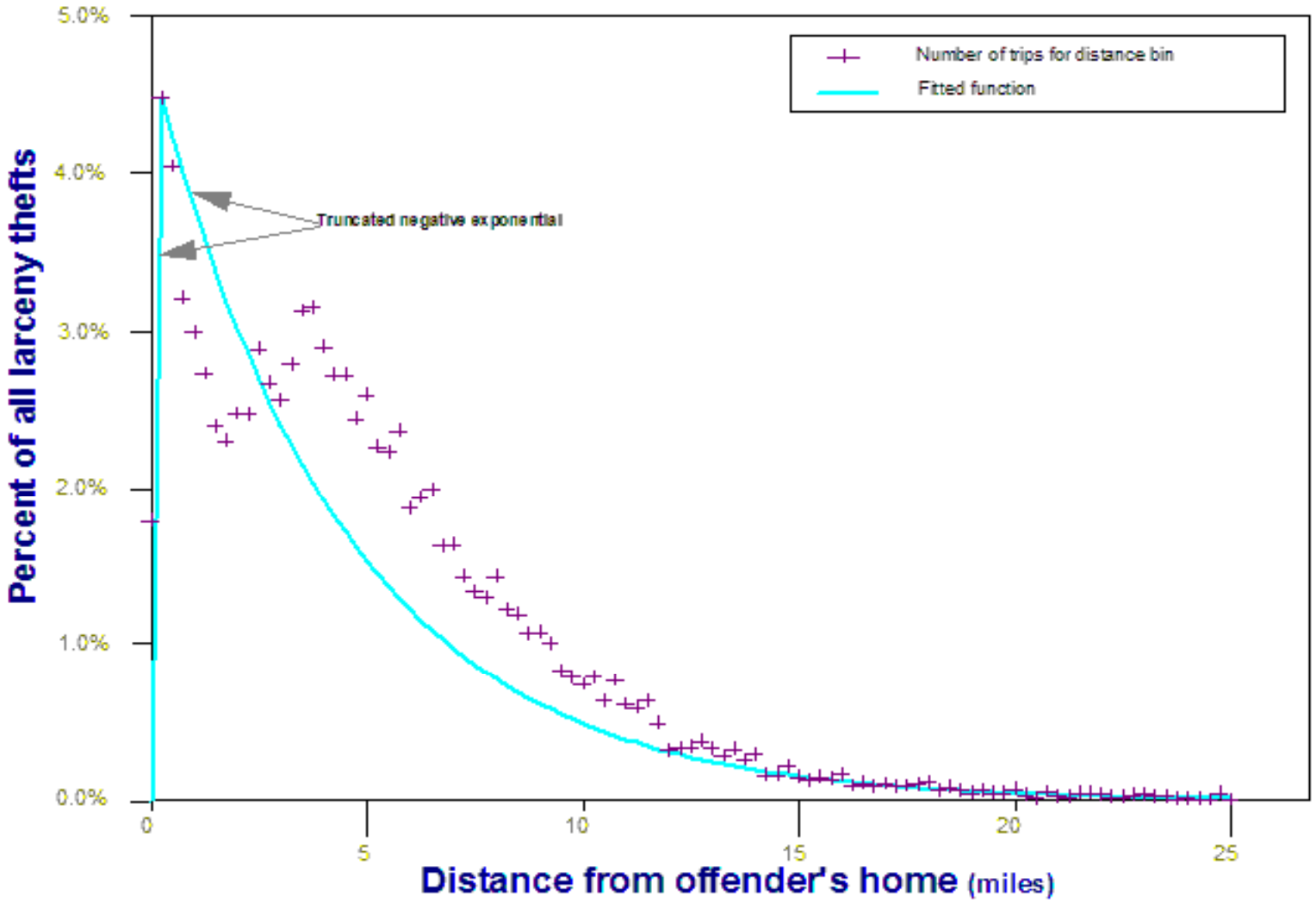
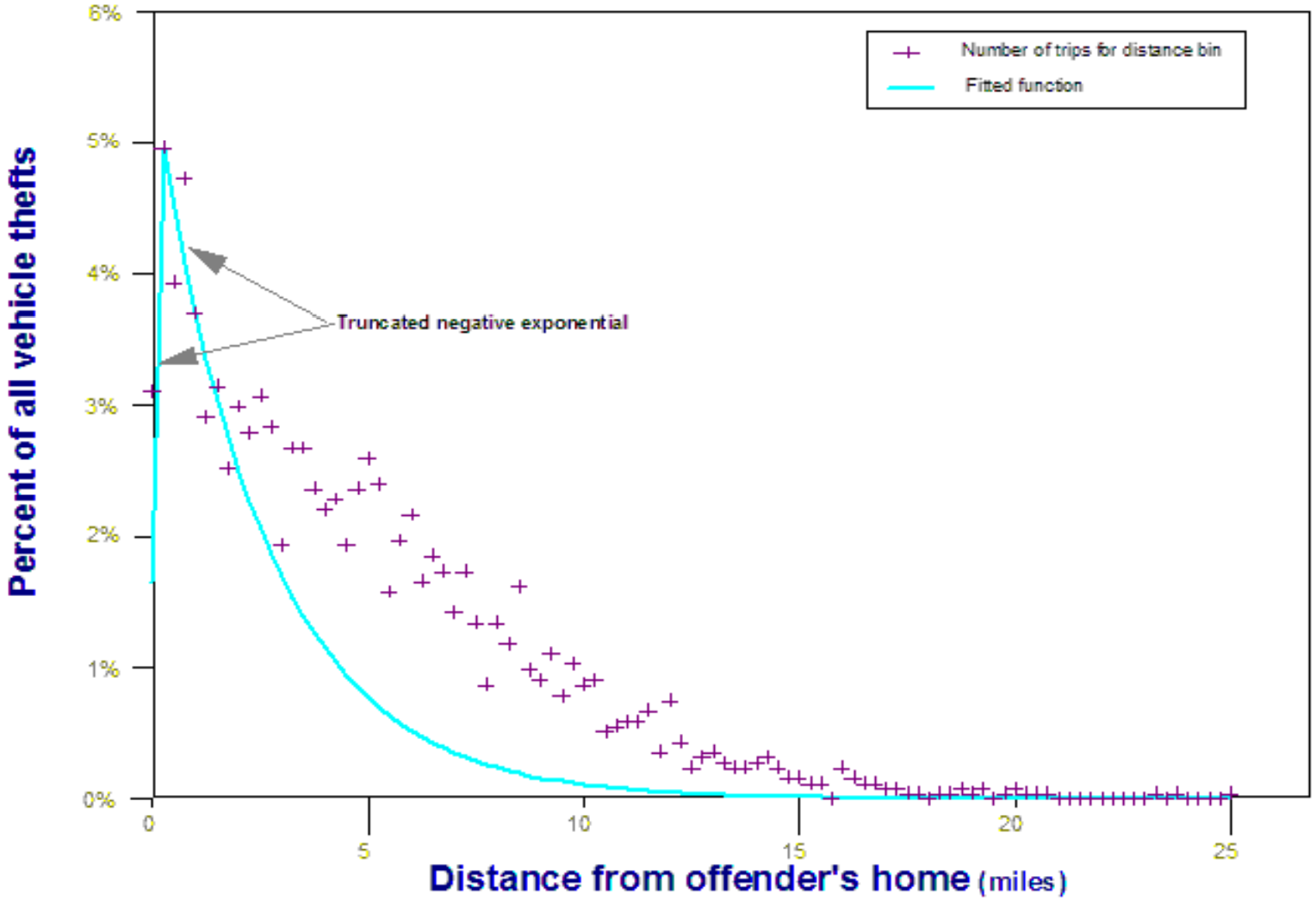


Figure 13.6:

Journey-to-crime Distances: Vehicle Theft

Truncated Negative Exponential Function



Some types of crime, on the other hand, are very difficult to fit. Figure 13.7 shows the distribution of bank robberies. Partly because there were a limited number of cases (N=176) and partly because it is a complex pattern, the truncated negative exponential gave the best fit, but not a particularly good one. As can be seen, the linear ('near home') function underestimates some of the near distance likelihoods while the negative exponential drops off too quickly; in fact, to make this function even plausible, the regression was run only up to 21 miles (otherwise, it underestimated even more).

For some crimes, it was very difficult to fit any single function. Figure 13.8 shows the frequency distribution of 137 homicides with three functions being fitted to the data - the truncated negative exponential, the lognormal, and the normal. As can be seen each function fits only some of the data, but not all of it.

Testing for Residual Errors in the Model

In short, the five mathematical functions allow a user to fit a variety of distance decay distributions. Each of the models will predict some parts of the distribution better than others. Consequently, it is important to conduct an error analysis to determine which model is 'best'. In an error analysis, the residual error is defined as:

$$Residual\ error_i = Y_i - E(Y_i) \tag{13.46}$$

where Y_i is the observed (actual) likelihood for distance i and $E(Y_i)$ is the likelihood predicted by the model. If raw numbers of incidents are used, then the likelihoods are the number of incidents for a particular distance. If the number of incidents are converted into proportions (i.e., probabilities), then the likelihoods are the proportions of incidents for a particular distance.

The choice of 'best model' will depend on what part of the distribution is considered most important. Figure 13.9, for example, shows the residual errors on vehicle theft for the five fitted models. That is, each of the five models was fit to the proportion of vehicle thefts by distance intervals (as explained above). For each distance, the discrepancy between the actual percentage of vehicle thefts in that interval and the predicted percentage was calculated. If there was a perfect fit, then the discrepancy (or residual) was 0%. If the actual percentage was greater than the predicted (i.e., the model underestimated), then the residual was positive; if the actual was smaller than the predicted (i.e., the model overestimated), then the residual was negative.

Figure 13.7:

Journey-to-crime Distances: Bank Robbery

Truncated Negative Exponential Function

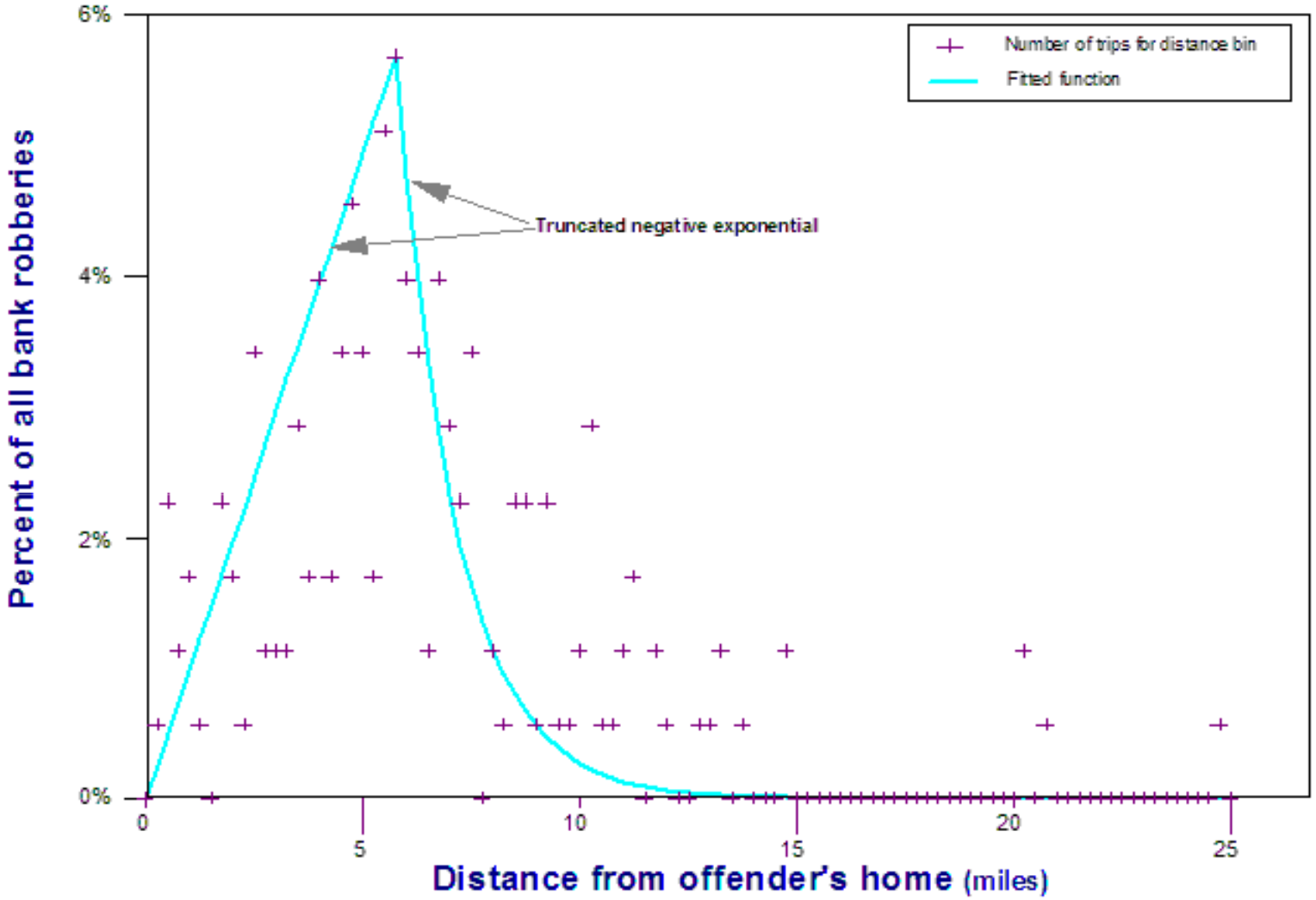


Figure 13.8:

Journey-to-crime Distances: Homicide

Normal, Lognormal, and Truncated Negative Exponential Functions

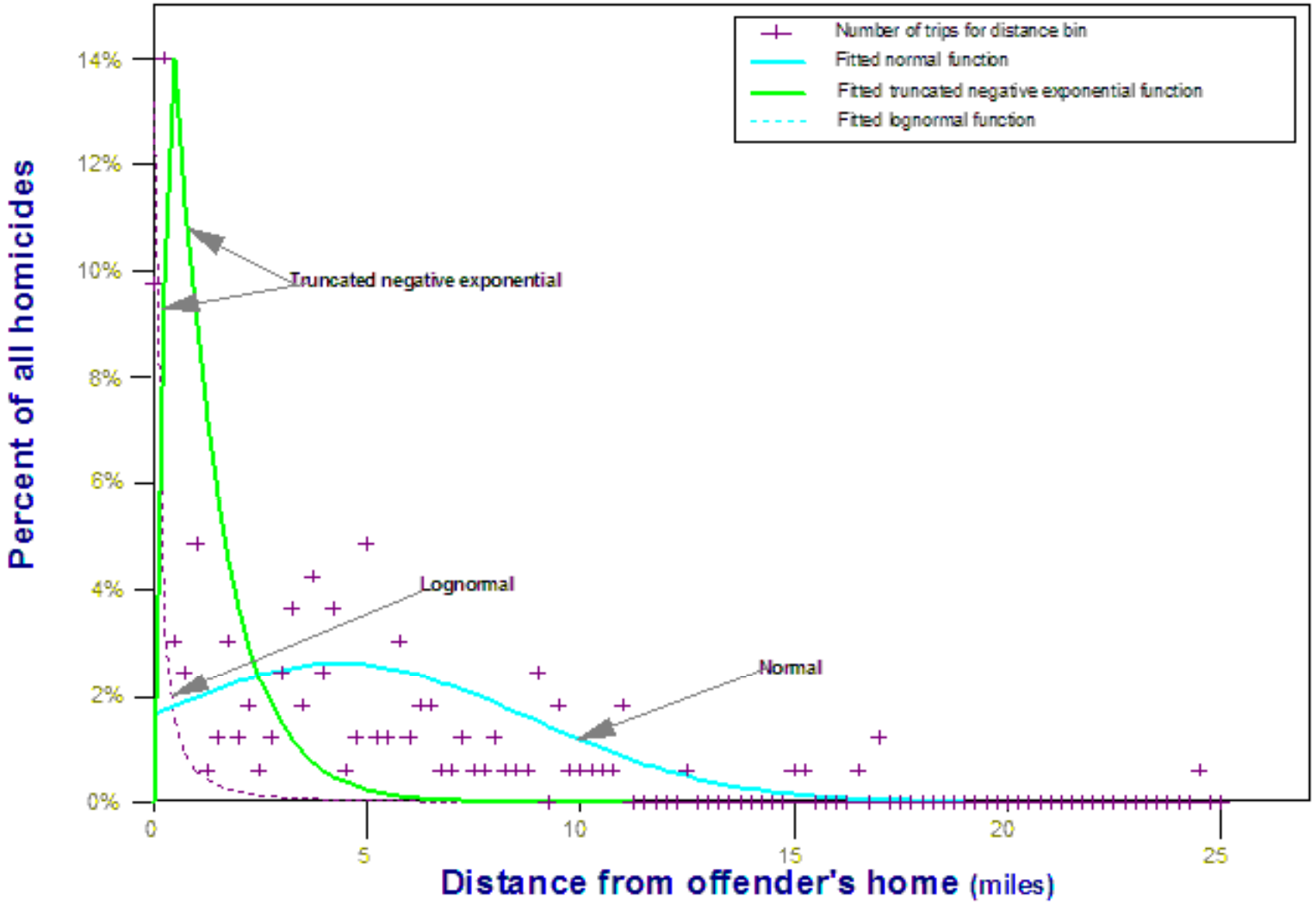
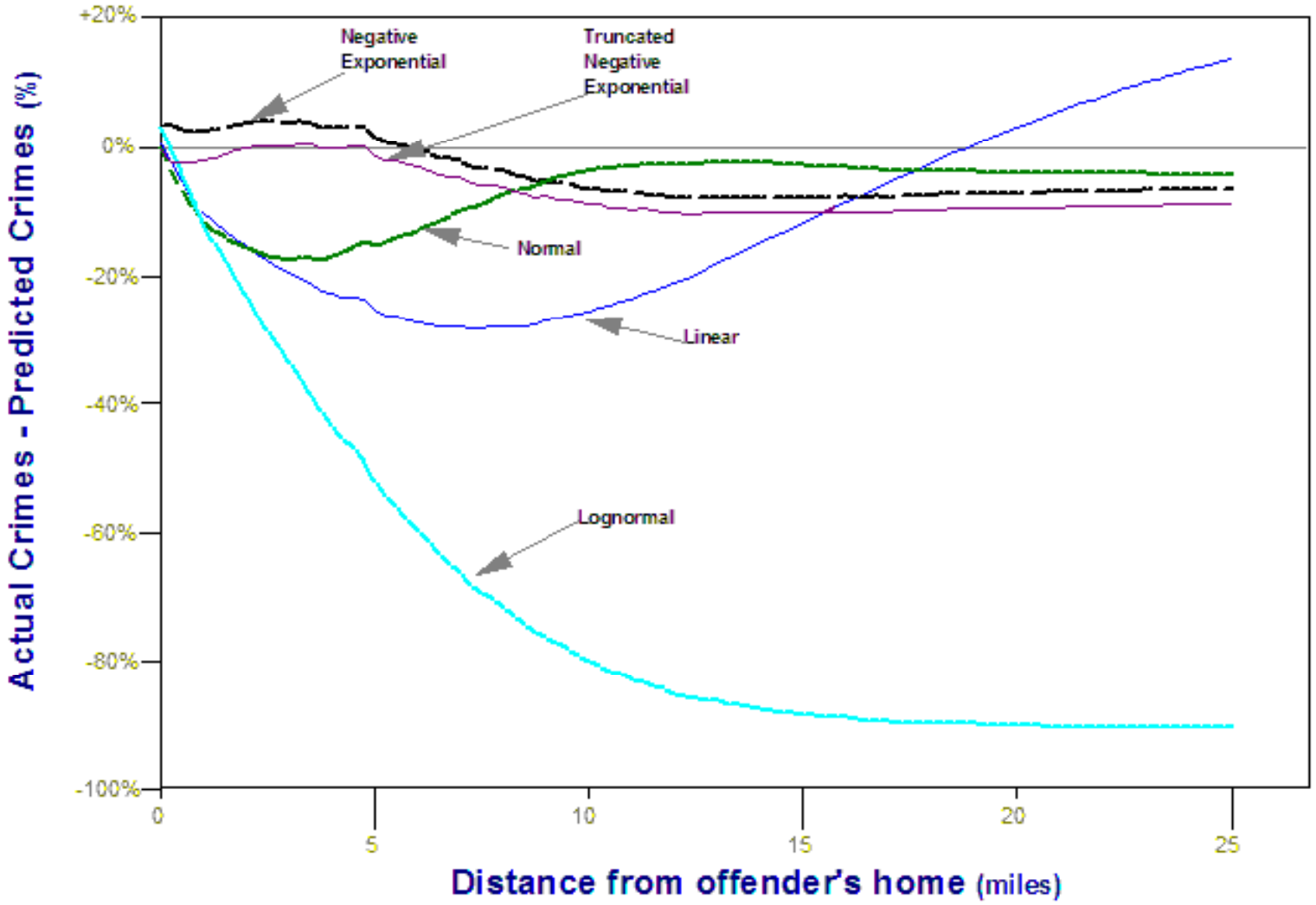


Figure 13.9:

Residual Error for Jtc Mathematical Models

Vehicle Theft



As can be seen in Figure 13.9, the truncated negative exponential fit the data well from 0 to about 5 miles, but then became poorer than other models for longer distances. The negative exponential model was not as good as the truncated for distances up to about 5 miles, but was better for distances beyond that point. The normal distribution was good for distances from about 10 miles and farther. The lognormal was not particularly good for any distances other than at 0 miles, nor was the linear.

The degree of predictability varied by type of crime. For some types, particularly property crimes, the fit was reasonably good. I obtained R^2 in the order of 0.86 to 0.96 for burglary, robbery, assault, larceny, and auto theft. For other types of crime, particularly violent crimes, the fit was not very good with R^2 values in the order of 0.53 (rape), 0.41 (arson) and 0.30 (homicide). These R^2 values were for the entire distance range; for any particular distance, however, the predictability varied from very high to very low.

In modeling distance decay with a mathematical function, a user has to decide which part of the distribution is the most important as no simple mathematical function will normally fit all the data (even approximately). In these cases, I assumed that the near distances were more important (up to, say, 5 miles) and, therefore, selected the model which 'best' fit those distances (see Table 13.2). However, it was not always clear which model was best, even with that limited criterion.

Problems with Mathematical Distance Decay Functions

There are several reasons that mathematical models of distance decay distributions, such as illustrated in the Jtc routine, do not fit data very well. First, as mentioned earlier, few cities have a completely symmetrical grid structure or even one that is approximately grid-like (there are exceptions, of course). Limitations of physical topography (mountains, oceans, rivers, lakes) as well as different historical development patterns make travel asymmetrical around most locations.

Second, there is population density. Since most metropolitan areas have much higher intensity of land use in the center (i.e., more activities and facilities), travel tends to be directed towards higher land use intensity than away from them. For origin locations that are not directly in the center, travel is more likely to go towards the center than away from it.

This would be true of an offender as well. If the person were looking for either persons or property as 'targets', then the offender would be more likely to travel towards the metropolitan center than away from it. Since most metropolitan centers have street networks that were laid out much earlier, the street network tends to be irregular. Consequently, trips will vary by location within a metropolitan area. One would expect shorter trips for offenders living close to

the metropolitan center than one living farther away, living in more built-up areas than in lower density areas, living in mixed use neighborhoods than in strictly residential neighborhoods; and so forth. Thus, the distribution of trips of any sort (in our case, crime trips from a residential location to a crime location), will tend to follow an irregular, distance decay type of distribution. Simple mathematical models will not fit the data very well and will make many errors.

Third, the selection of a best mathematical function is partly dependent on the interval size used for the bins. In the above examples, an interval size of 0.25 miles was used to calculate the frequency distribution. With a different interval size (e.g., 0.5 miles), however, a slightly different distribution is obtained. This affects the mathematical function that is selected as well as the parameters that are estimated. For example, the questoin of whether there is a safety zone near the offender's residence from which there is decreased activity or not is partly dependent on the interval size. With a small interval, the zone may be detected whereas with a slightly larger interval the subtle distinction in measured distances may be lost. On the other hand, having a smaller interval may lead to unreliable estimates since there may be few cases in the interval. Having a technique depend on the interval size makes it vulnerable to misspecification.

Uses of Mathematical Distance Decay Functions

Does this mean that one should not use mathematical distance functions? I would argue that under most circumstances, a mathematical function will give less precision than an empirically-derived one (see below). However, there are two cases when a mathematical model would be appropriate. First, if there is either no data or insufficient data to model the empirical travel distribution, the use of a mathematical model can serve as an approximation. If the user has a good sense of what the distribution looks like, then a mathematical model may be used to approximate the distribution. However, if a poorly defined function is selected, then the selected function may produce many errors.

A second case when mathematical models of distance decay would be appropriate is in theory development or application. Many models of travel behavior, for example, assume a simple distance decay type of function in order simplify the allocation of trips over a region. This is a common procedure in travel demand modeling where trips from each of many zones are assigned to every other zone using a gravity type of function (Stopher & Meyburg, 1975; Field & MacGregor, 1987). Even though the model produces errors because it assumes uniform travel behavior in all directions, the errors are corrected later in the modeling process by adjusting the coefficients for allocating trips to particular roads (traffic assignment). The model provides a simple device and the errors are corrected down the line. Still, I would argue that an empirically-derived distribution will produce fewer errors in allocation and, thus, require less adjustment later on. Errors can never help a model and it is better to get it more correct initially than to have to adjust it later on. Nevertheless, this is common practice in transportation planning.

Using the Routine with a Mathematical Function

The Jtc routine which allows mathematical modeling is simple to use. Figure 13.10 illustrates how the user specifies a mathematical function. The routine requires the use of a grid which is defined on the reference file tab of the program (see chapter 3). Then, the user must specify the mathematical function and the parameters. In the figure, the truncated negative exponential is being defined. The user must input values for the peak likelihood, the Cutoff distance, and the exponent (see equations 10.43 and 10.44 above). In the figure, since the serial offenses were a series of 18 robberies, the parameters for robbery have been entered into the program screen. The peak likelihood was 9.96% (entered as a whole number - i.e., 9.96); the distance at which this peak likelihood occurred was the second distance interval 0.25-0.50 miles (with a mid-point of 0.38 miles); and the estimated exponent was 0.177651. As mentioned above, the coefficient for the negative exponential part of the equation is estimated internally.

Table 13.3 gives the parameters for the 'best' models which fit the data for the 11 types of crime in Baltimore County. For several of these (e.g., bank robberies), two or more functions gave approximately equally good fits. Note that these parameters were estimated with the Baltimore County data. They will not fit any other jurisdiction. If a user wishes to apply this logic, then the parameters should be estimated anew from existing data. Nevertheless, once they have been calibrated, they can be used for predictions.

The routine can be output to *ArcGIS*, *MapInfo*, *Atlas*GIS*, *Surfer for Windows*, *Spatial Analyst*, and as an Ascii grid file which can be read by many other GIS packages. All but *Surfer for Windows* require that the reference grid be created by *CrimeStat*.

Empirically Estimating a Journey-to-crime Calibration Function

An alternative to mathematical modeling of distance decay is to empirically describe the Journey-to-crime distribution and then use this empirical function to estimate the residence location. *CrimeStat* has a two-dimensional kernel density routine that can calibrate the distance function if provided data on trip origins and destinations. The logic of kernel density estimation was described in chapter 10, and will not be repeated here. Essentially, a symmetrical function (the 'kernel') is placed over each point in a distribution. The distribution is then referenced relative to a scale (an equally-spaced line for two-dimensional kernels and a grid for three-dimensional kernels) and the values for each kernel are summed at each reference location. See chapter 8 for details.

Figure 13.10:

Jtc Mathematical Distance Decay Function

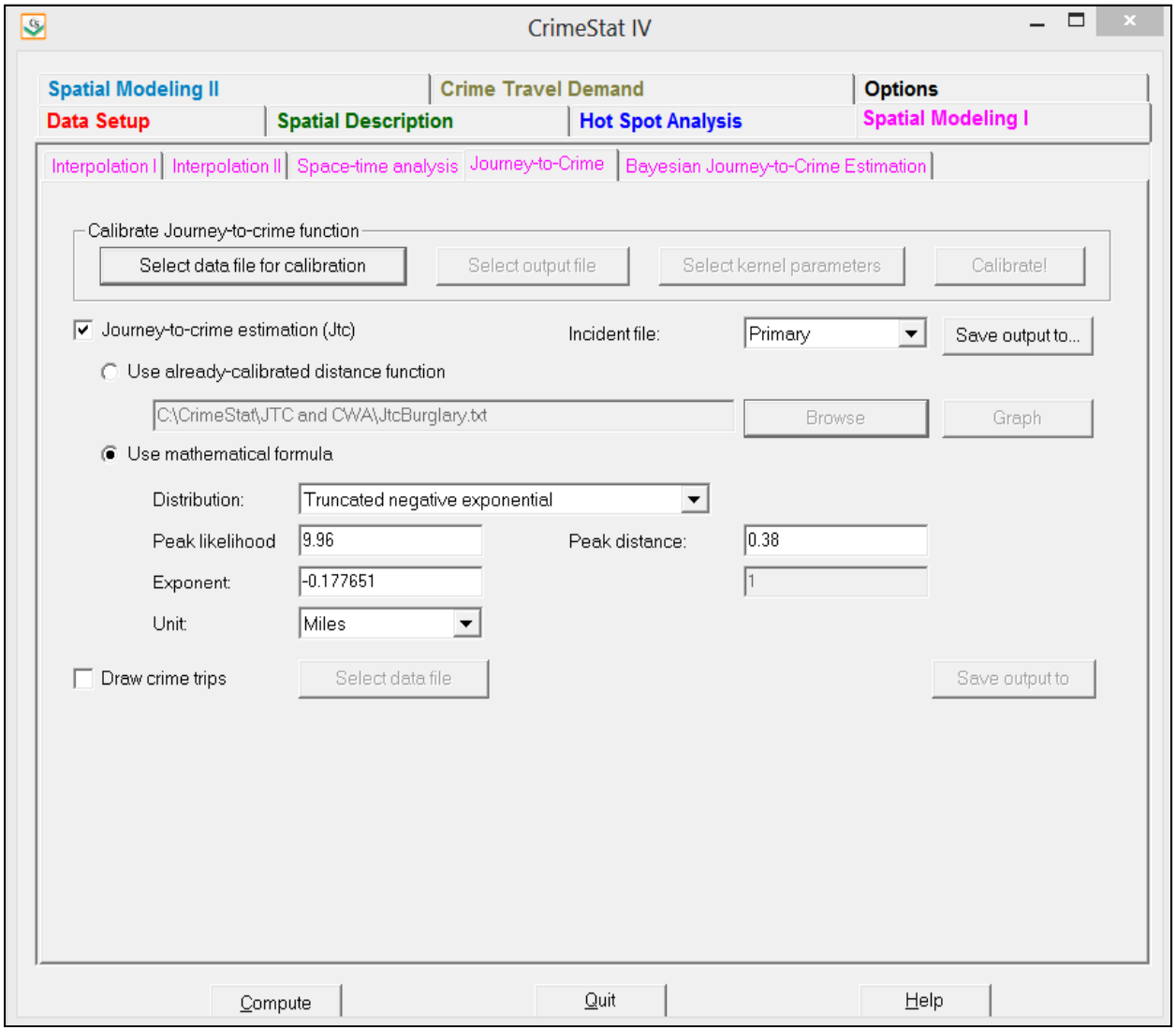


Table 13.3:
Journey-to-crime Mathematical Models for Baltimore County
Parameter Estimates for Percentage Distribution
(Sample Sizes in Parentheses)

ALL CRIMES

Negative Exponential:	Coefficient:	5.575107
	Exponent:	0.229466

HOMICIDE

Truncated		
Negative Exponential:	Peak likelihood	14.02%
	Cutoff distance	0.38 miles
	Exponent	0.064481

RAPE

Lognormal:	Mean	3.144959
	Standard Deviation	4.546872
	Coefficient	0.062791

ASSAULT

Truncated		
Negative Exponential:	Peak likelihood	27.40%
	Cutoff distance	0.38 miles
	Exponent	0.181738

ROBBERY

Truncated		
Negative Exponential:	Peak likelihood	9.96%
	Cutoff distance	0.38 miles
	Exponent	0.177651

COMMERCIAL ROBBERY

Truncated		
Negative Exponential:	Peak likelihood	4.9455%
	Cutoff distance	0.625 miles
	Exponent	0.151319

Table 13.3: (continued)

BANK ROBBERY

Truncated Negative Exponential:	Peak likelihood	9.96%
	Cutoff distance	5.75 miles
	Exponent	0.139536

BURGLARY

Truncated Negative Exponential:	Peak likelihood	20.55%
	Cutoff distance	0.38 miles
	Exponent	0.162907

AUTO THEFT

Truncated Negative Exponential:	Peak likelihood	4.81%
	Cutoff distance	0.63 miles
	Exponent	0.212508

LARCENY

Truncated Negative Exponential:	Peak likelihood	4.76%
	Cutoff distance	0.38 miles
	Exponent	0.193015

ARSON

Truncated Negative Exponential:	Peak likelihood	38.99%
	Cutoff distance	0.38 miles
	Exponent	0.093469

Calibrate Kernel Density Estimate

The *CrimeStat* calibration routine allows a user to describe the distance distribution for a sample of Journey-to-crime trips. The requirements are that:

1. The data set must have the coordinates of *both* an origin location and a destination location; and

2. The records of all origin and destination locations have been populated with legitimate coordinate values (i.e., no unmatched records are allowed).

Data set definition

The steps are relatively easy to run the routine. First, the user defines a calibration data set with both origin and destination locations. Figure 13.11 illustrates this process. As with the primary and secondary files, the routine reads *Excel 'xls' and 'xlsx', ArcGIS* 'shp', *dBase* 'dbf', Ascii 'txt', and *MapInfo* 'dat' files. For both the origin location (e.g., the home residence of the offender) and the destination location (i.e., the crime location), the names of the variables for the X and Y coordinates must be identified as well as the type of coordinate system and data unit (see Chapter 3). In the example, the origin locations has variable names of HomeX and HomeY and the destination locations has variable names of IncidentX and IncidentY for the X and Y coordinates of the two locations respectively. However, any name is acceptable as long as the two locations are distinguished.

The user should specify whether there are any missing values for these four fields (X and Y coordinates for both origin and destination locations). By default, *CrimeStat* will ignore records with blank values in any of the eligible fields or records with non-numeric values (e.g.,alphanumeric characters, #, *). Blanks will always be excluded unless the user selects <none>. There are 8 possible options:

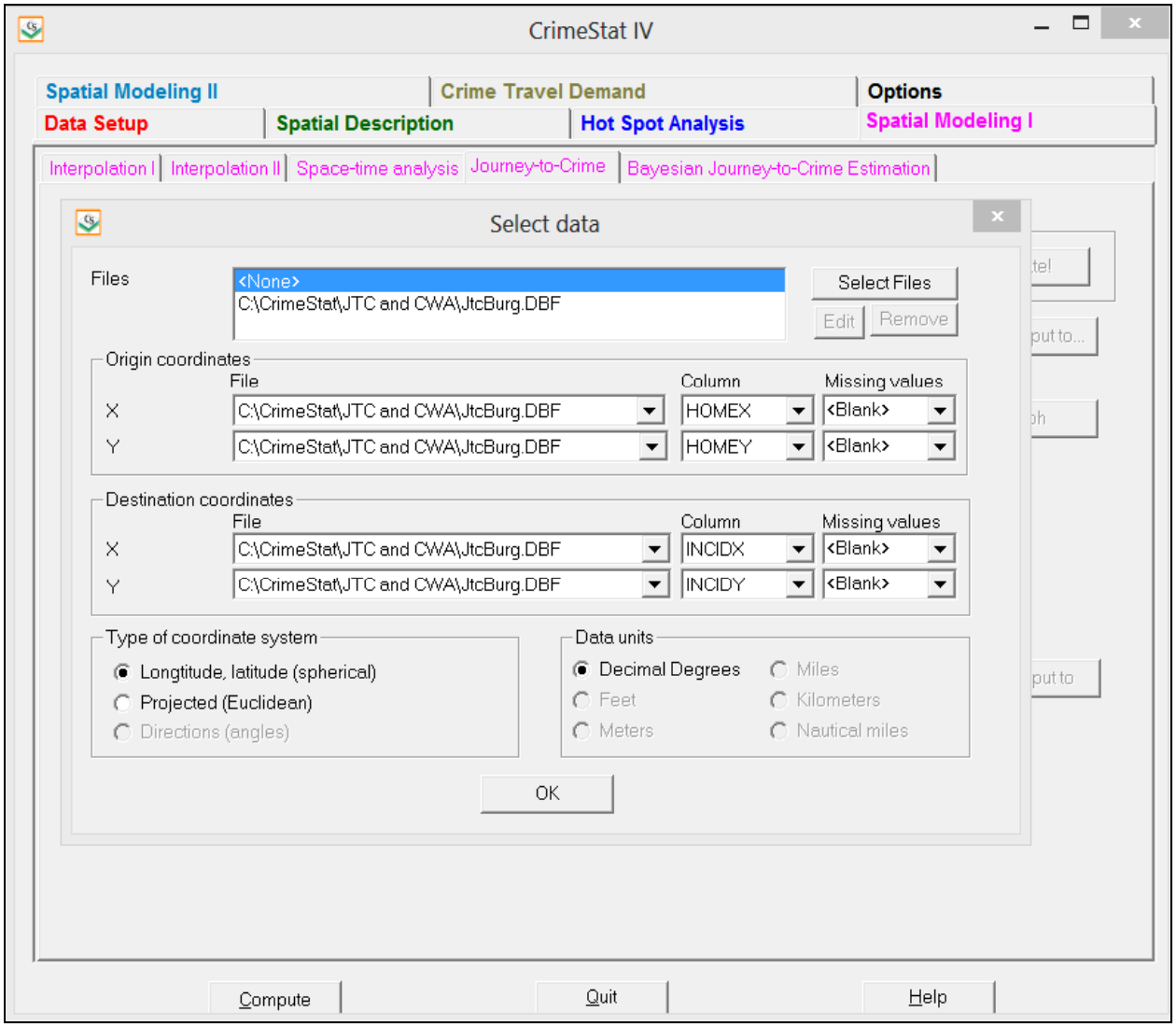
1. <blank> fields are automatically excluded. This is the default
2. <none> indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0
3. 0 is excluded
4. −1 is excluded
5. 0 and −1 indicates that both 0 and -1 will be excluded
6. 0, -1 and 9999 indicates that all three values (0, -1, 9999) will be excluded

Any other numerical value can be treated as a missing value by typing it (e.g., 99)Multiple numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99).

The program will calculate the distance between the origin location and the destination location for each record. If the units are spherical (i.e., lat/lon), then the calculations use spherical geometry; if the units are projected (either meters or feet), then the calculations are Euclidean (see chapter 3 for details).

Figure 13.11:

Jtc Calibration Data Input



Kernel Parameters

Second, the user must define the kernel parameters for calibration. There are five choices that have to be made (Figure 13.12):

1. The method of interpolation. As with the two-dimensional kernel technique described in Chapter 10, there are five possible kernel functions:

   A. Normal (the default);
   B. Quartic;
   C. Triangular (conical);
   D. A negative exponential (peaked); and
   E. A uniform (flat) distribution.

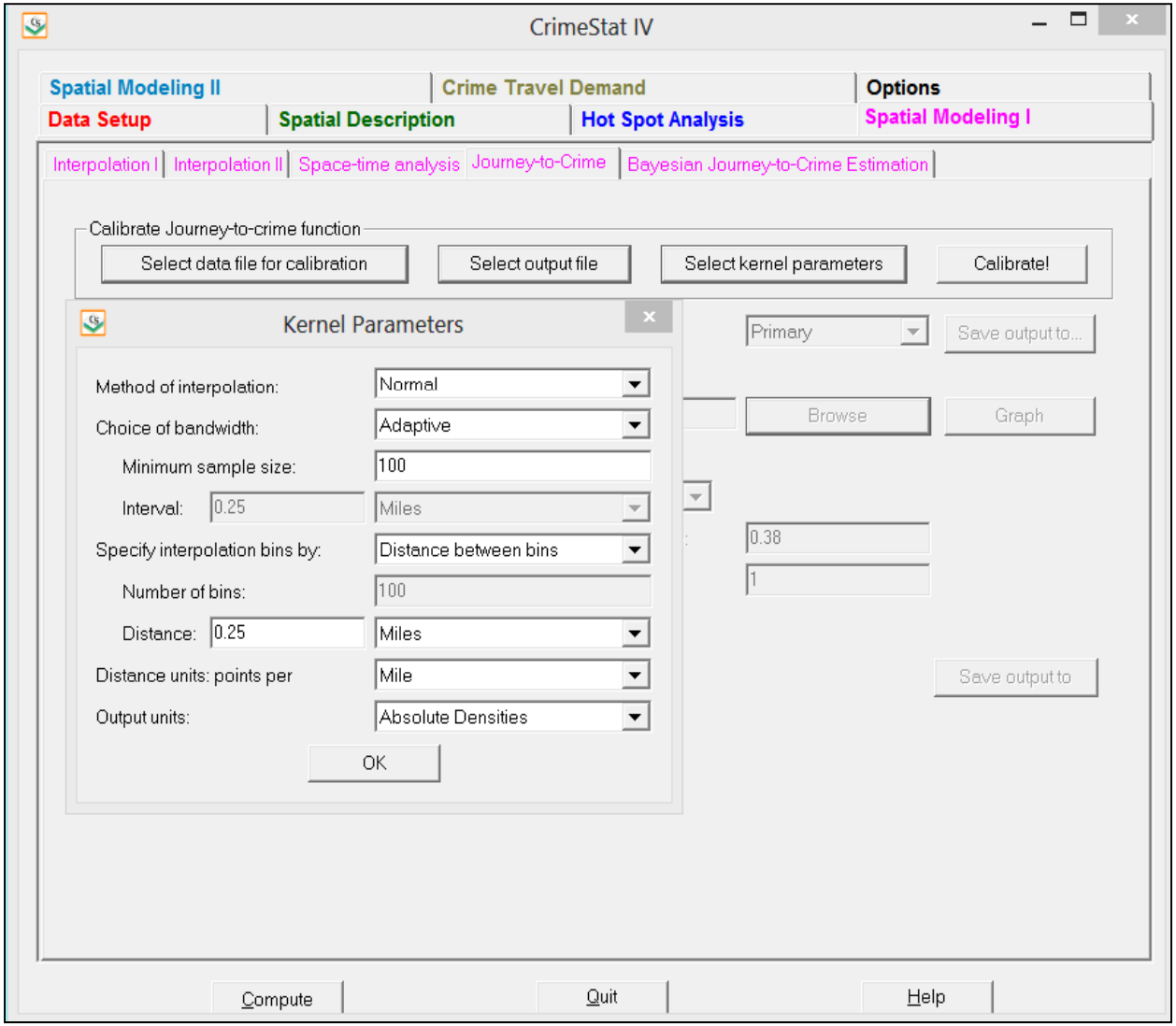
2. Choice of bandwidth. The bandwidth is the width of the kernel function. For a normal kernel, it is the standard deviation of the normal distribution whereas for the other four kernels (quartic, triangular, negative exponential, and uniform), it is the radius of the circle defined by the kernel. As with the two-dimension kernel technique, the bandwidth can be fixed in length or made adaptive (variable in length). However, for the one-dimensional kernel, the fixed bandwidth is the default since an even estimate over an equal number of intervals (bins) is desirable. If a fixed bandwidth is selected, the interval size must be specified and the units defined (in miles, kilometers, feet, meters, and nautical miles). The default is 0.25 mile intervals. If the adaptive bandwidth is selected, the user must identify the minimum sample size that the bandwidth should incorporate; in this case, the bandwidth is widened until the specified sample size is counted.

3. The number of interpolation bins. The bins are the intervals along the distance scale (from 0 up to the maximum distance for a Journey-to-crime trip) and are used to estimate the density function. There are two choices.

   A. The user can specify the number of intervals (the default choice with 100 intervals). In this case, the routine calculates the maximum distance (or longest trip) between the origin location and the destination location and divides it by the specified number of intervals (e.g., 100 equal-sized intervals). The interval size is dependent on the longest trip distance measured.

Figure 13.12:

Jtc Calibration Kernel Parameters



B. Alternatively, the user can specify the distance between bins (or the interval size). The default choice is 0.25 miles, but another value can be entered. In this case, the routine counts out intervals of the specified size until it reaches the maximum trip distance.

4. The output units. The user specifies the units for the density estimate (in units per mile, kilometer, feet, meters, and nautical miles).

5. The output calculations. The user specifies whether the output results are in probabilities (the default) or in densities. For probabilities, the sum of all kernel estimates will equal 1.0. For densities, the sum of all kernel estimates will equal the sample size.

Saved calibration file

Third, the user must define an output file to save the empirically determined function. The function is then used in estimating the likely home residence of a particular function. The choices are to save the file as a 'dbf' or Ascii text file. The saved file then can be used in the Jtc routine. Figure 13.13 illustrates the output file format.

Calibrate

Fourth, the calibrate button runs the routine. A calibration window appears and indicates the progress of the calculations. When it is finished, the user can view a graph illustrating the estimated distance decay function (Figure 13.14). The purpose is to provide quick diagnostics to the user on the function and selection of the kernel parameters. While the graph can be printed, it is not a high quality print. If a high quality graph is needed, the output calibration file should be imported into a graphics program.

Examples from Baltimore County, MD

I will illustrate this method by showing the results for the same data sets that were calculated above in the mathematical section (Figures 13.4-13.8). In all cases, the normal kernel function was used. The bandwidth was 0.25 miles except for the bank robbery data set, which had only 176 cases, and the homicide data set, which only had 137 cases; because of the small sample sizes, a bandwidth of 0.50 miles was used for these two data sets. The interval width selected was a distance of 0.25 miles between bins (0.5 miles for bank robberies and homicides) and probabilities were output.

Figure 13.13:

Jtc Calibration Output File

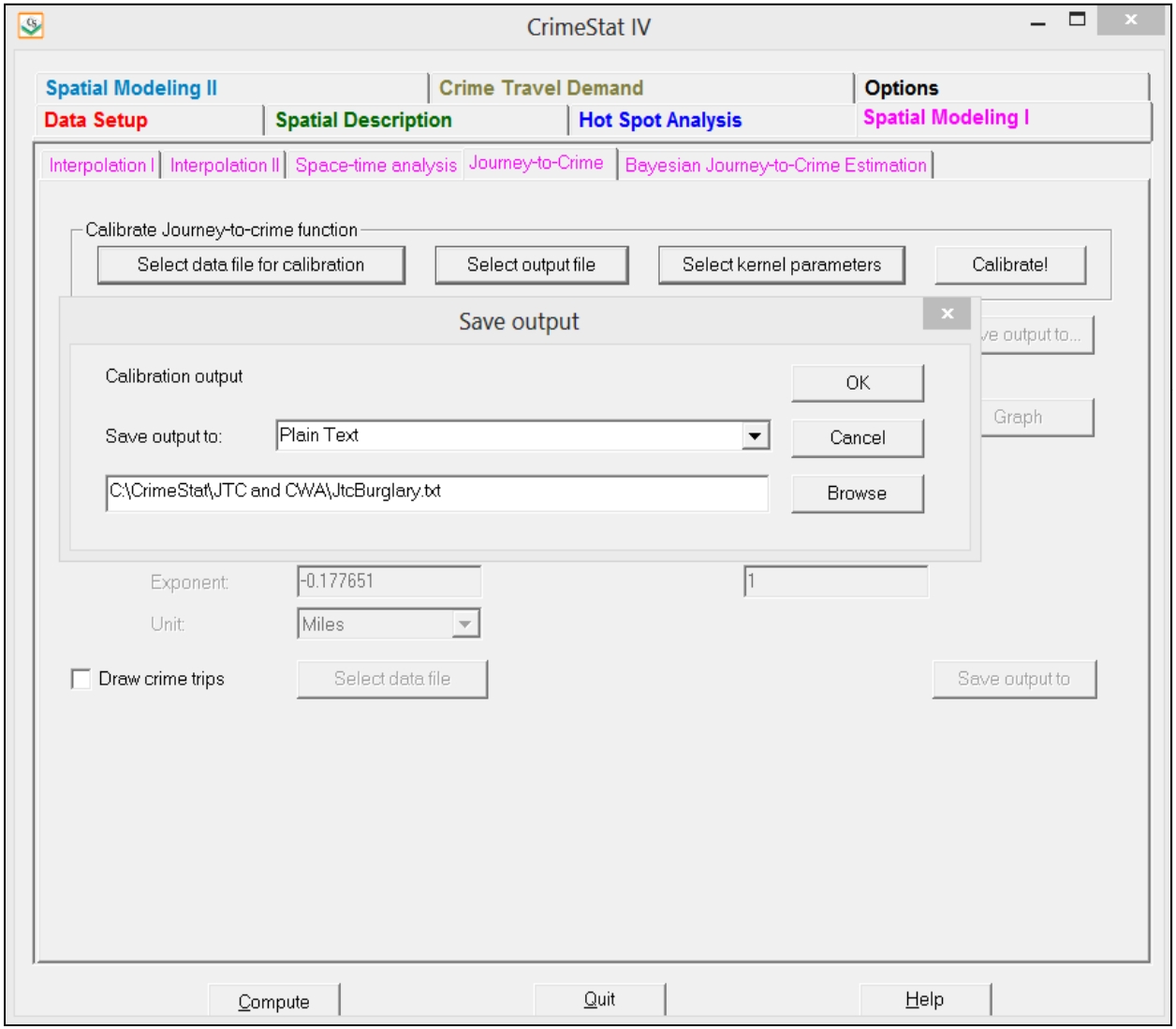


Figure 13.14:

Jtc Calibration Graphic Output

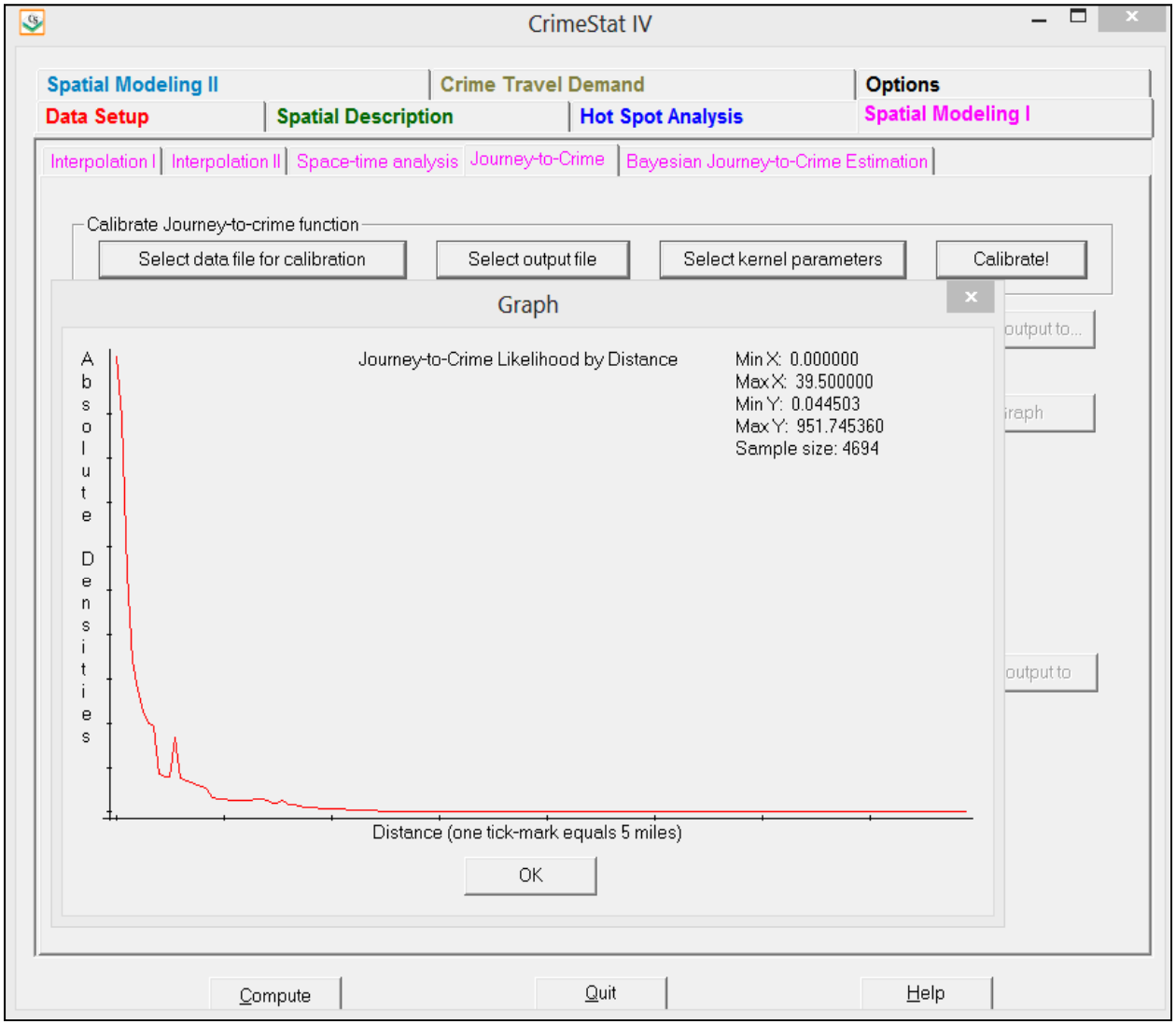


Figure 13.15 shows the kernel estimate for all crimes (41,426 trips). A frequency distribution was calculated for the same number of intervals and is overlaid on the graph. It was selected to be comparable to the mathematical function (see Figure 13.4). Note how closely the kernel estimate fits the data compared to the negative exponential mathematical function. The fit is good for every value but the peak value; that is because the kernel averages several intervals together to produce an estimate.

Figure 13.16 shows the kernel estimate for larceny thefts. Again, the kernel method produces a much closer fit as a comparison with Figure 13.5 will show. Figure 13.17 shows the kernel estimate for vehicle thefts. Figure 13.18 shows the kernel estimate for bank robberies and Figure 13.19 shows the kernel estimate for homicides. An inspection of these graphs shows how well the kernel function fits the data, compared to the mathematical function, even when the data are irregularly spaced (in vehicle thefts, bank robberies, and homicides). Figure 13.20 compares the distance decay functions for homicides committed against strangers compared to homicides committed against known victims.

In short, the Jtc calibration routine allows a much closer fit to the data than any of the simpler mathematical functions. While it's possible to produce a complex mathematical function that will fit the data more closely (e.g., higher order polynomials), the kernel method is much simpler to use and gives a good approximation to the data.

Journey-to-crime Estimation Using a Calibrated File

After the distance decay function has been calibrated and saved as a file, the file can be used to calculate the likelihood surface for a serial offender. The user specifies the name of the already-calibrated distance function (as a 'dbf' or an Ascii text file) and the output format. As with the mathematical routine, the output can be to *ArcGIS*, *MapInfo*, *Atlas*GIS*, *Surfer for Windows*, *Spatial Analyst*, and as an Ascii grid file which can be read by many other GIS packages. All but *Surfer for Windows* require that the reference grid be created by *CrimeStat*.

The result is produced in three steps:

1. The routine calculates the distance between each reference cell of the grid and each incident location;

2. For each distance measured, the routine looks up the calculated value from the saved calibration file; and

3. For each reference grid cell, it sums the values of all the incidents to produce a single likelihood estimate.

Figure 13.15:

Journey-to-crime Distances: All Crimes

Kernel Density Estimate by Percent of Crimes

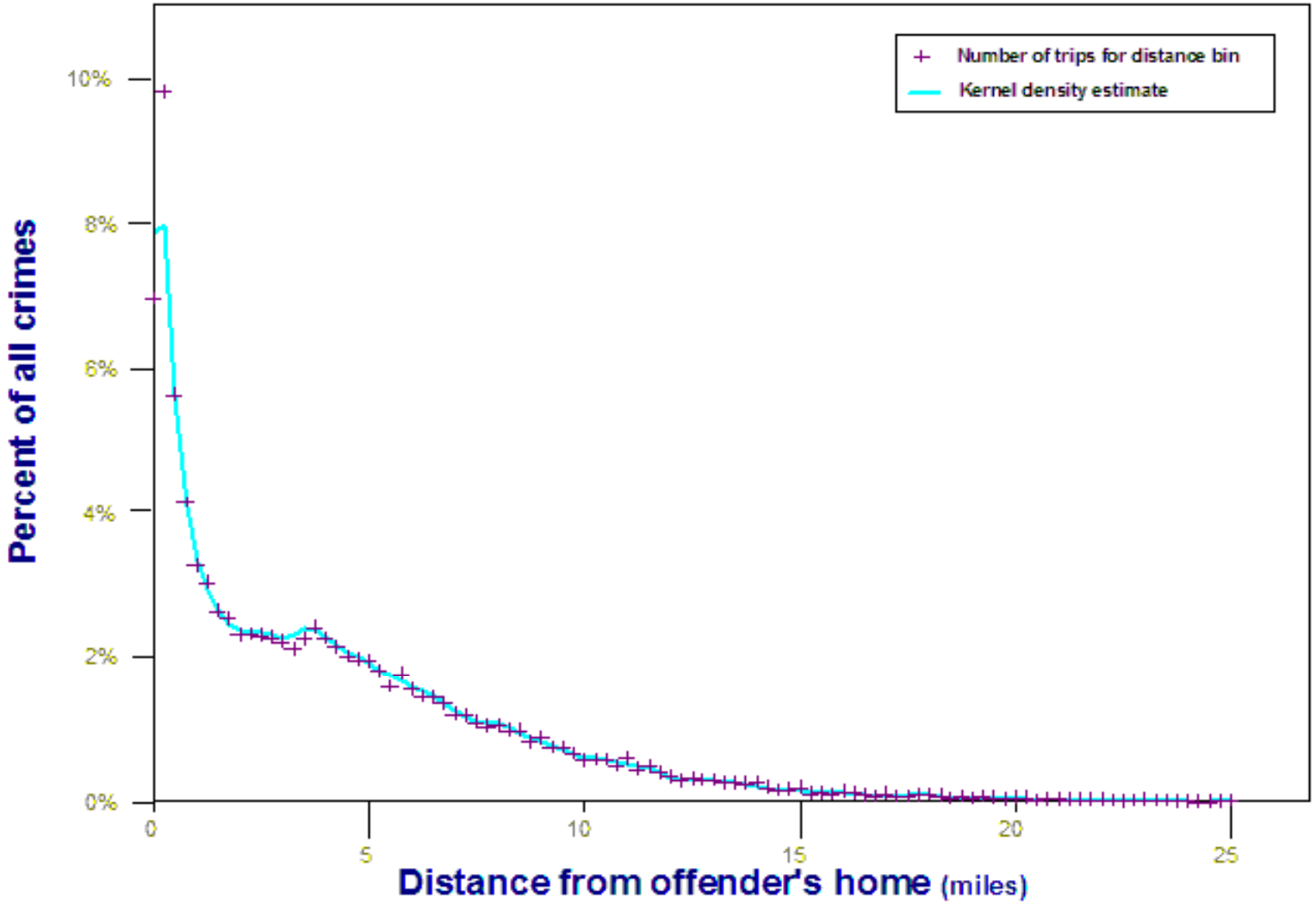


Figure 13.16:

Journey-to-crime Distances: Larceny

Kernel Density Estimate by Percent of Crimes

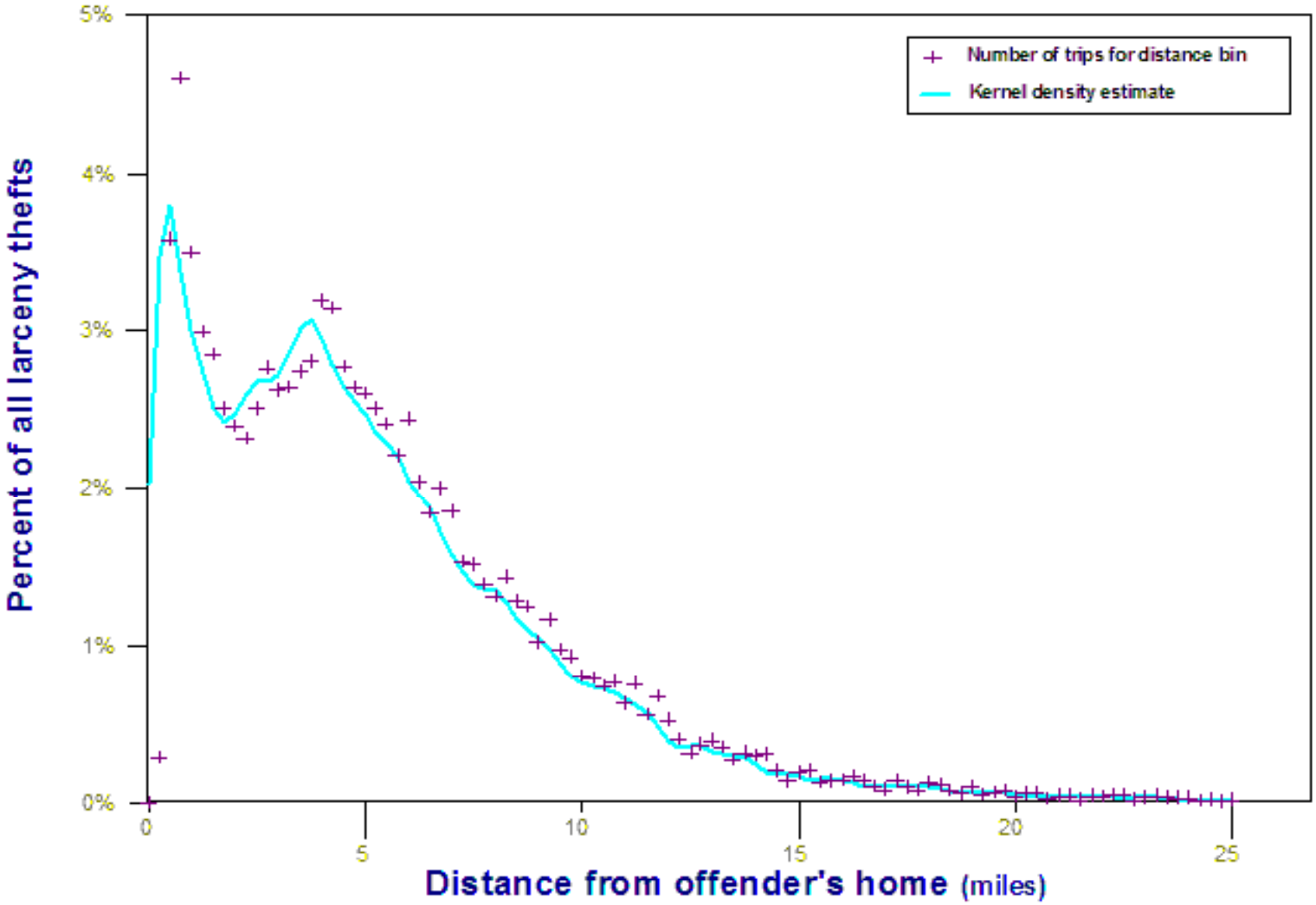


Figure 13.17:

Journey-to-crime Distances: Vehicle Theft

Kernel Density Estimate by Percent of Crimes

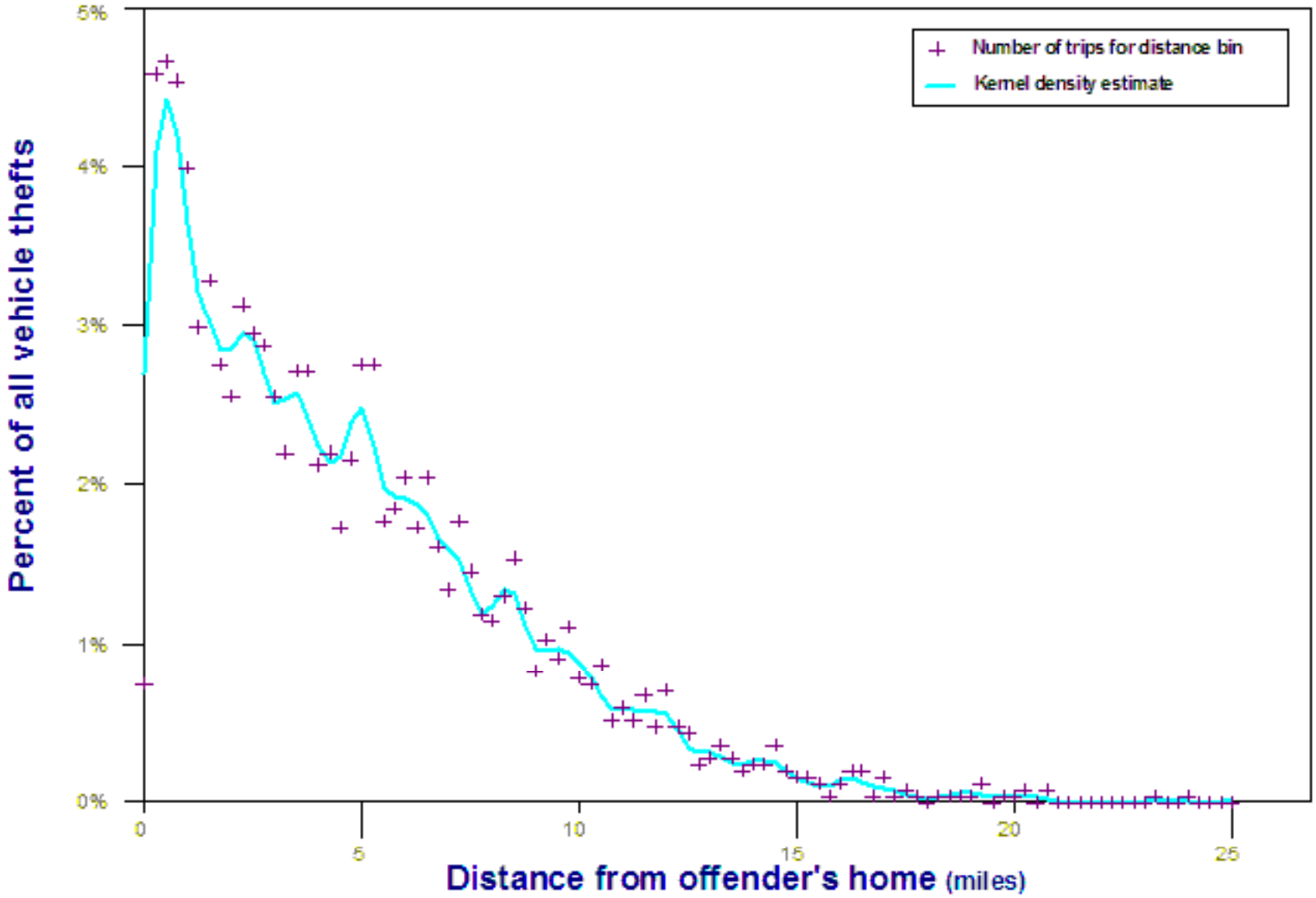


Figure 13.18:

Journey-to-crime Distances: Bank Robbery

Kernel Density Estimate by Percent of Crimes

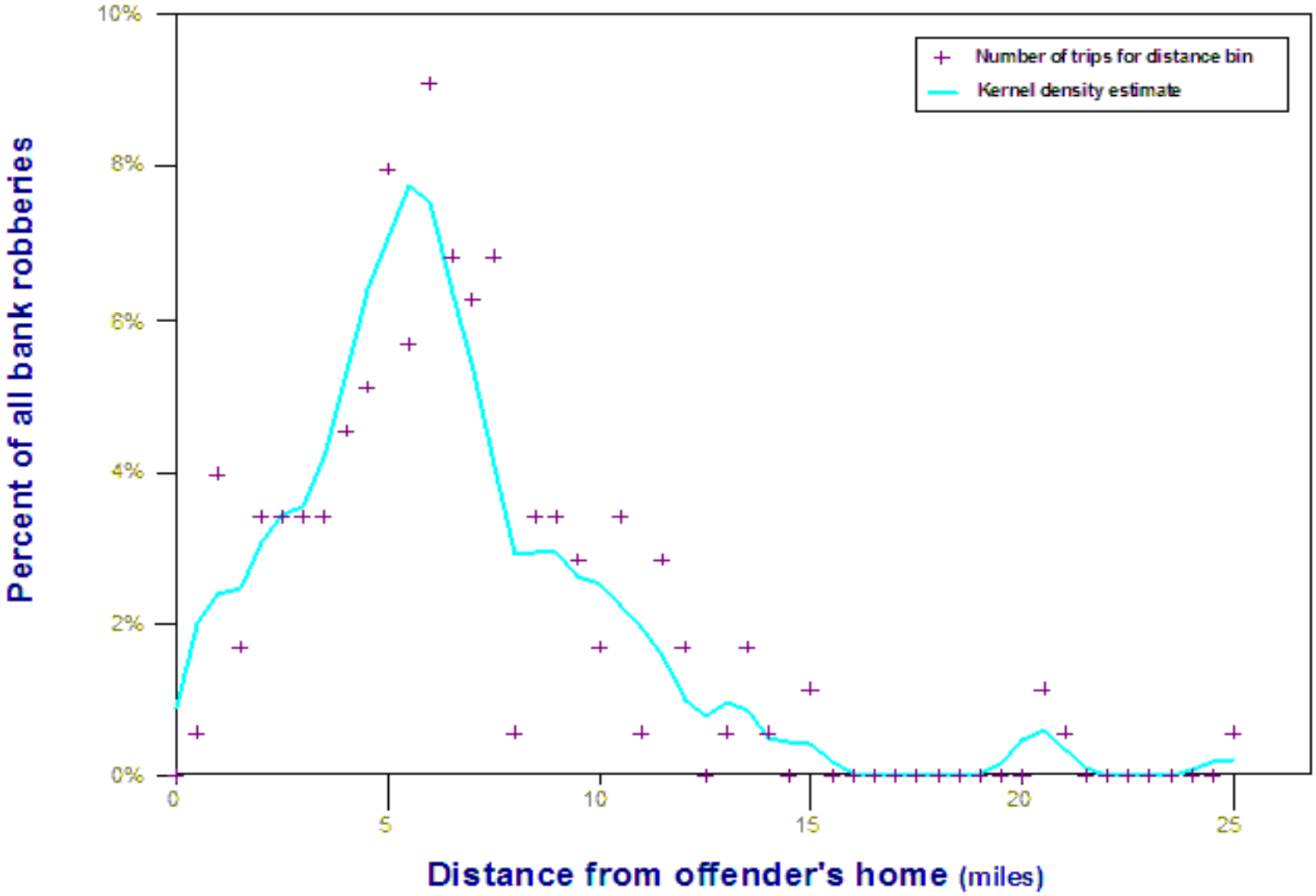


Figure 13.19:

Journey-to-crime Distances: Homicide

Kernel Density Estimate by Percent of Crimes

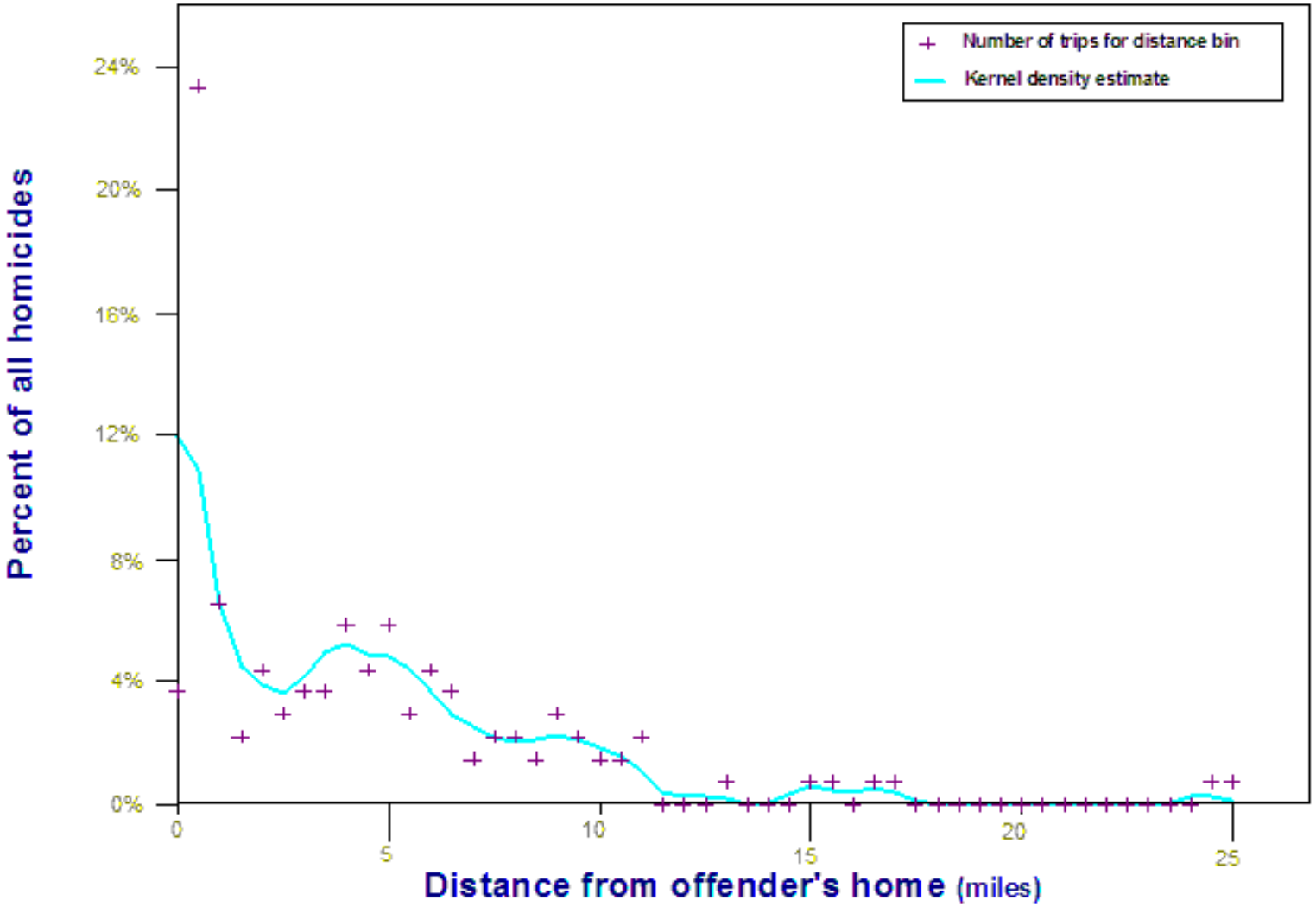
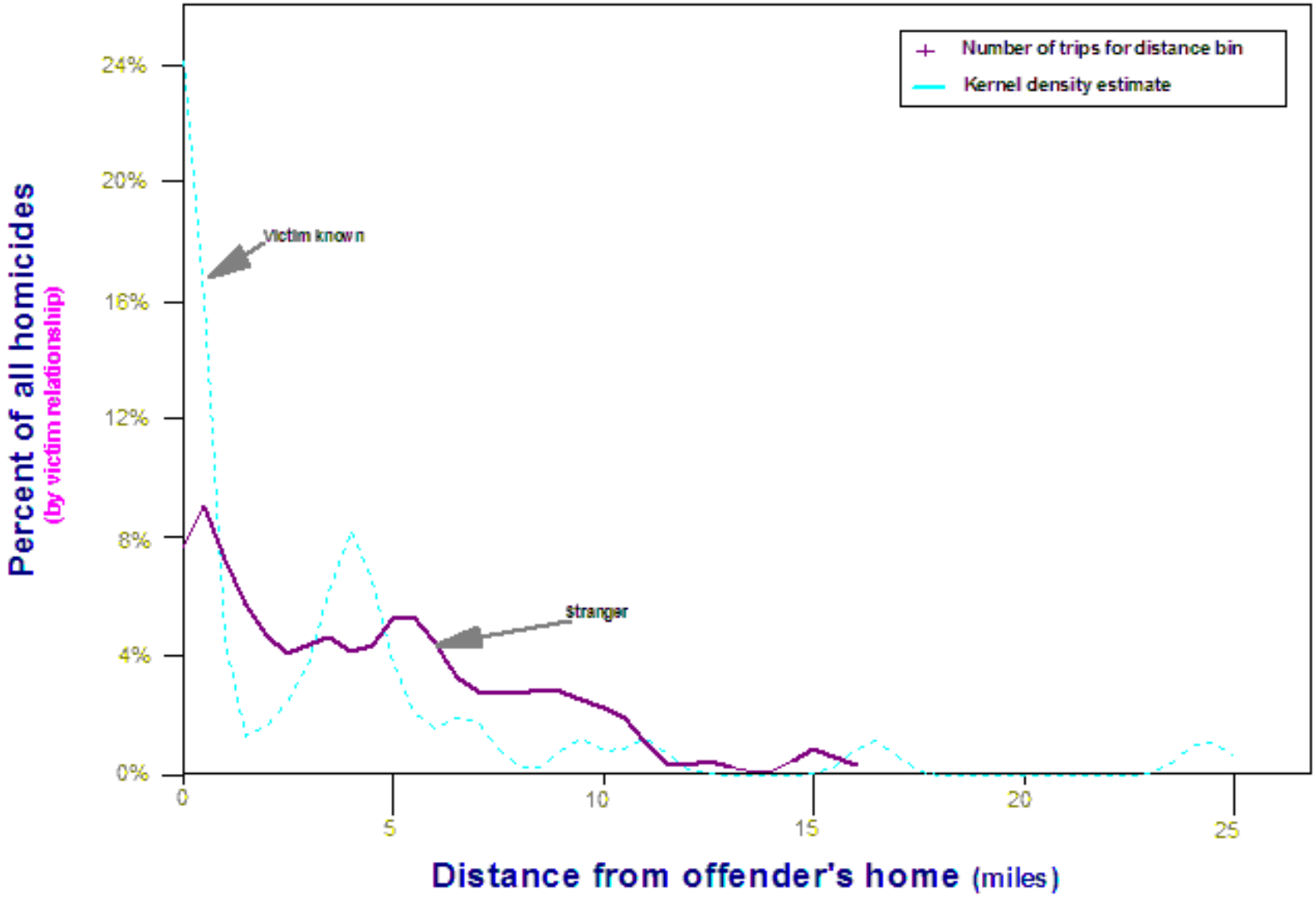


Figure 13.20:

Journey-to-crime Distances: Homicide by Victim Relationship

Kernel Density Estimate by Percent of Crimes



Application of the Routine

To illustrate the techniques, the results of the two methods on a single case are compared. The case has been selected because the routines accurately estimate the offender's residence. This was done to demonstrate how the techniques work. In the next section, I will ask the question about how accurate these methods are in general.

The case involved a man who had committed 24 offenses. These included 13 thefts, 5 burglaries, 5 assaults, and one rape. The spatial distribution was varied; many of the offenses were clustered but some were scattered. Since there were multiple types of crimes committed by this individual, a decision had to be made over which model to use to estimate the individual's residence. In this case, the theft (larceny) model was selected since that was the dominant type of crime for this individual.

For the mathematical function, the truncated negative exponential was chosen from Table 13.3 with the parameters being:

Peak likelihood	4.76%
Cutoff distance	0.38 miles
Exponent	0.193015

For the kernel density model, the calibrated function for larceny was selected (see Figure 13.16).

Figure 13.21 shows the results of the estimation for the two methods. The output is from *Surfer for Windows* (Golden Software, 2008). The left pane shows the results of the mathematical function while the right pane shows the results for the kernel density function. The incident locations are shown as circles while the actual residence location of the offender is shown as a square. Since this is a surface model, the highest location has the highest predicted likelihood.

In both cases, the models predicted quite accurately. The discrepancy (error) between the predicted peak location and the actual residence location was 0.66 miles for the mathematical function and 0.36 miles for the kernel density function. For the mathematical model, the actual residence location (square) is seen as slightly off from the peak of the surface whereas for the kernel density model the discrepancy from the peak cannot be seen.

Nevertheless, the differences in the two surfaces show distinctions. The mathematical model has a smooth decline from the peak likelihood location, almost like a cone. The kernel density model, on the other hand, shows a more irregular distribution with a peak location followed by a surrounding 'trough' followed a peak 'rim'. This is due to the irregular distance

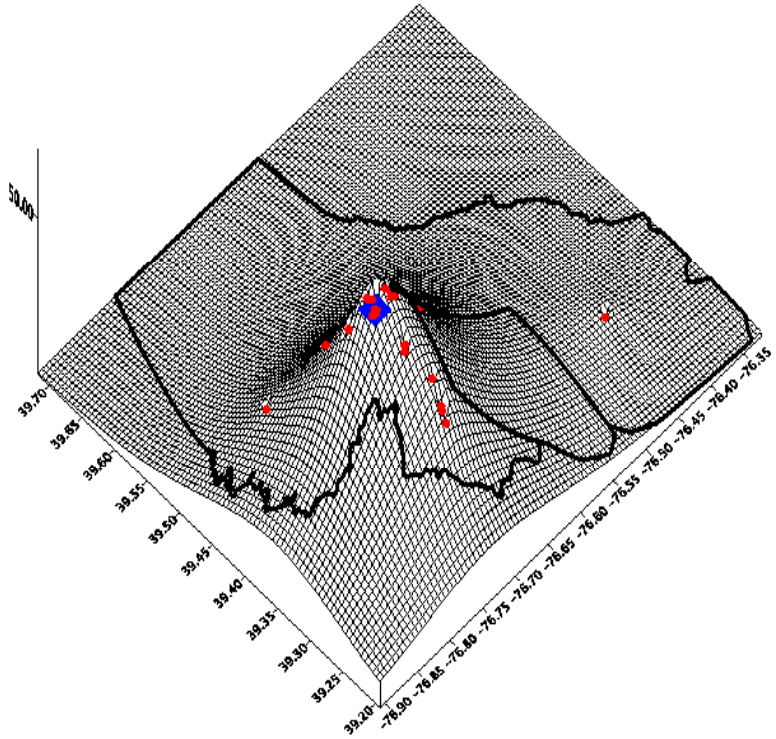
Figure 13.21:

Predicted and Actual Location of Serial Thief

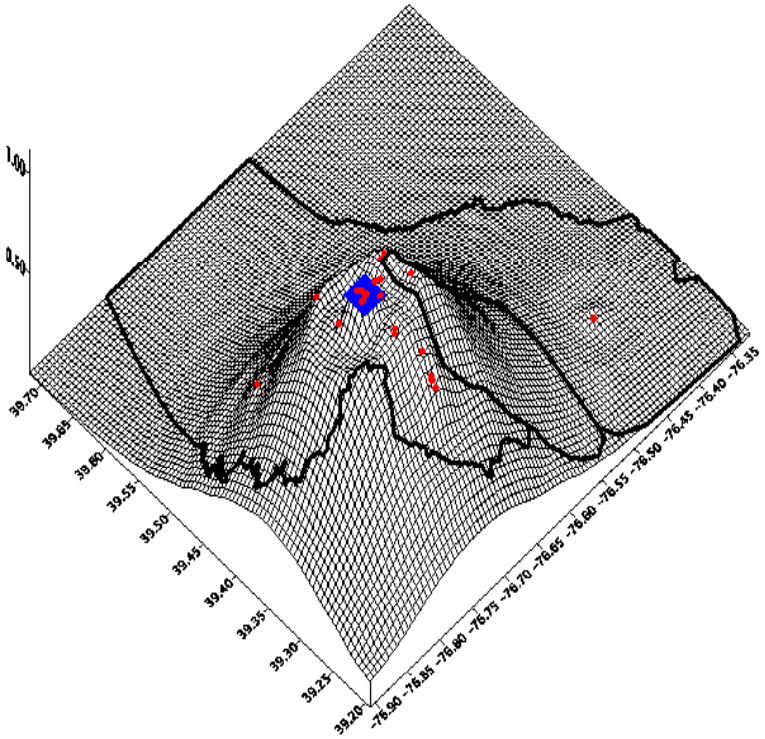
Man Charged with 24 Offenses in Baltimore County

Predicted with Mathematical and Kernel Density Models for Larceny

Residence = square
Crime locations=circles



Mathematical Model:
Truncated Negative Exponential



Kernel Density Model

decay function calibrated for larceny (see Figure 13.16). But, in both cases, they more or less identify the actual residence location of the offender.

Choice of Calibration Sample

The calibration sample is critical for either method. Each method assumes that the distribution of the serial offender will be similar to a sample of 'like' offenders. Obviously, distinctions can be made to make the calibration sample more or less similar to the particular case. For example, if a distance decay function of all crimes is selected, then a model (of either the mathematical or kernel density form) will have less differentiation than for a distance decay function from a specific type of crime. Similarly, breaking down the type of crime by, say, mode of operation or time of day will produce better differentiation than by grouping all offenders of the same type together. This process can be taken on indefinitely until there is too little data to make a reliable estimate. An analyst should try to match a calibration sample to the actual as is possible, given the limitations of the data.

For example, in our calibration data set, there were 4,694 burglary incidents where both the offender's home residence and the incident location were known. The approximate time of the offense for 2,620 of the burglaries was known and, of these, 1,531 occurred at night between 6 pm and 6 am. Thus, if a particular serial burglar for whom the police are interested in catching tends to commit most of his burglaries at night, then choosing a calibration sample of nighttime burglars will generally produce a better estimate than by grouping all burglars together. Similarly, of the 1,531 nighttime burglaries, 409 were committed by individuals who had a prior relationship with the victim. Again, if the analysts suspect that the burglar is robbing homes of people he knows or is acquainted with, then selecting the subset of nighttime burglaries committed against a known victim would produce even better differentiation in the model than taking all nighttime burglars. However, eventually, with further sub-groupings there will be insufficient data.

This point has been raised in a recent debate. Van Koppen and De Keijser (1997) argued that a distance decay function that combines multiple incidents committed by the same individuals could distort the estimated relationship compared to selecting incidents committed by different individuals.[4] This result has been supported by Smith, Bond and Townsley (2009) and

4 They also argued that the combination of incidents - which they called 'aggregation', would distort the relationship between distance and incidence likelihood because of the ecological fallacy. To my mind, they are incorrect on this point. Data on a distribution of incidents by distance traveled is an individual characteristic and is not 'ecological' in any way. An ecological inference occurs when data are aggregated with a *grouping* variable (e.g., state, county, city, census tract; see Langbein and Lichtman, 1978). A frequency distribution of individual crime trip distances is an individual probability distribution, similar, for example, to a distribution of individuals by height, weight, income or any other characteristic. Of course, there are sub-sets of the data that have been aggregated (similar to heights of men v. heights of women, for

Townsley and Sidebottom (2010). Rengert, Piquero and Jones (1999) argued that such a distribution is nevertheless meaningful. In our language, these are two different sub-groups - persons committing multiple offenses compared to persons committing only one offense. Combining these two sub-groups into a single calibration data set will only mean that the result will have less differentiation in prediction than if the sub-groups were separated out.

Actually, there is not much difference, at least in Baltimore County, a result that we also found in Baltimore County, MD (Levine & Lee, 2012). From the 41,426 cases, 18,174 were committed by persons who were only listed once in the database while 23,251 offenses were committed by persons who were listed two or more times (7,802 individuals). Categorizing the 18,174 crimes as committed by 'single incident offenders and the 23,251 crimes as committed by 'multiple incident offenders', the density distance decays functions were calculated using the kernel density method (Figure 13.22).

The distributions are remarkably similar. There are some subtle differences. The average Journey-to-crime trip distance made by a single incident offender is longer than for multiple incident offenders (4.6 miles compared to 4.0 miles, on average); the difference is highly significant ($p \leq .0001$), partly because of the very large sample sizes. However, a visual inspection of the distance decay functions shows they are similar. The single incident offenders tend to have slightly more trips near their home, slightly fewer for distances between about a mile up to three miles, and slightly more longer trips. But, the differences are not very large.

There are several reasons for the similarity. First, some of the 'single incident offenders' are actually multiple incident offenders who have not been charged with other incidents. Second, some of the single incident offenders are in the process of becoming multiple incident offenders so their behavior is probably similar. Third, there may not be a major difference in travel patterns by the number of offenses an individual commits, certainly compared to the major differences by type of crime (see graphs above). In other words, the distinction between a single offender crime trip and a multiple offender crime trip is just another sub-group comparison and, apparently, not that important. Nevertheless, it is important to choose an appropriate sample

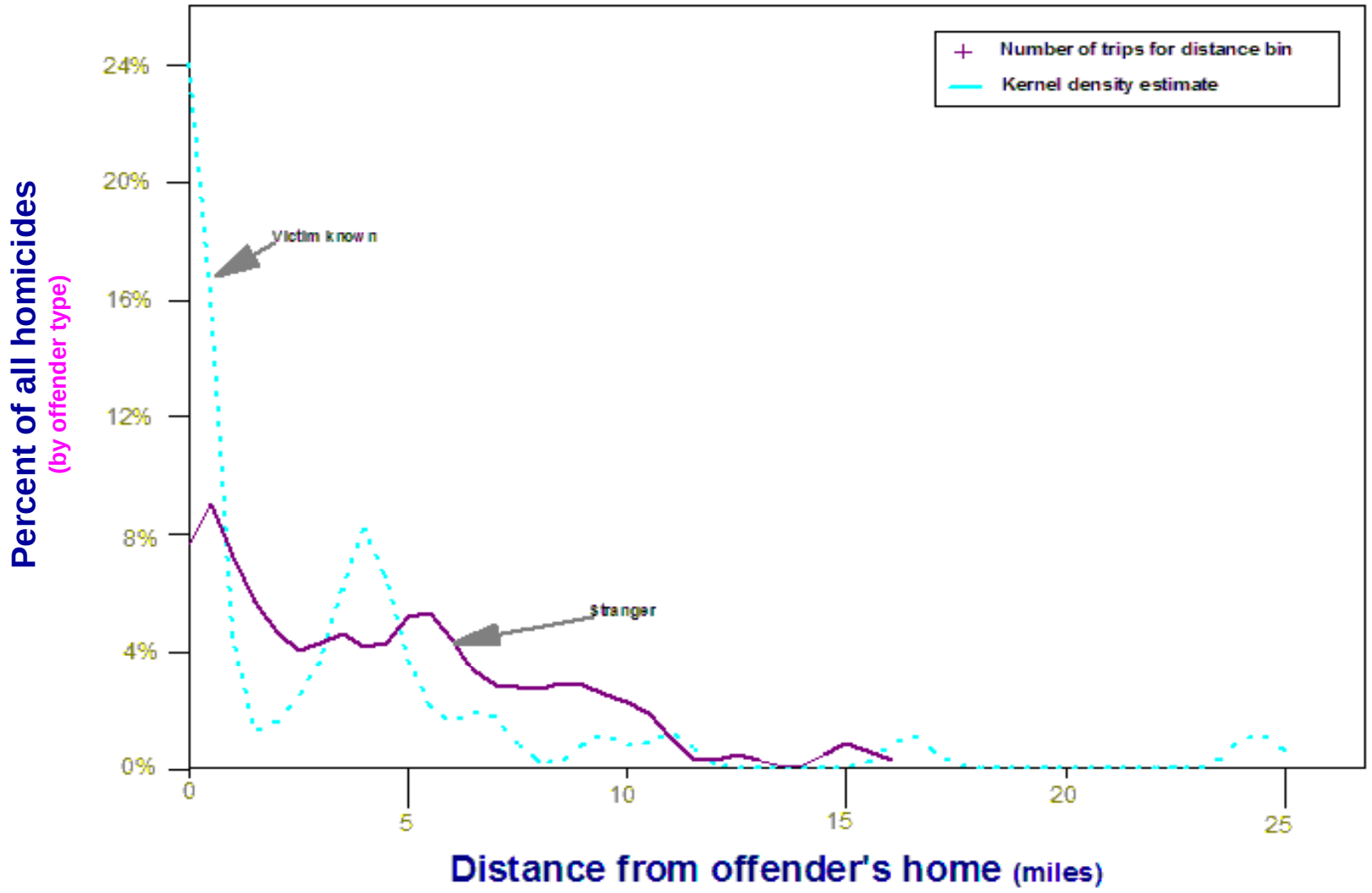
---

example). Clearly, identifying sub-groups can make better distinctions in a distribution. But, it is still an individual probability distribution and does not produce bias in estimating a parameter, only variability. For example if a particular distance decay function implies that 70% of the offenders live within, say, 5 miles of their committed incidents, then 30% do not live within 5 miles. In other words, because the data are individual level, then a distance decay function, whether estimated by a mathematical or a kernel density model, is an individual probability model (i.e., an attempt to describe the underlying distribution of individual travel distances for Journey-to-crime trips). See the following discussion (Wikipedia, 2010a; 2010b; Friedman, 1999).

Figure 13.22:

Journey-to-crime Distances

Kernel Density Estimate for Single and Multiple Incident Offenders



from which to estimate a likely home base location for a serial offender. The method depends on a similar sample of offenders for comparison.

Sample Data Sets for Journey-to-crime Routines

Three sample data sets from Baltimore County have been provided for the Journey-to-crime routine. The data sets are simulated and do not represent real data. The first file - JtcTest1.dbf, are 2000 simulated robberies while the second file - JtcTest2.dbf, are 2500 simulated burglaries. Both files have coordinates for an origin location (HomeX, HomeY) and a destination location (IncidentX, IncidentY). Users can use the calibration routine to calculate the travel distances between the origins and the destinations. A third data set - Serial1.dbf, are simulated incident locations for a serial offender. Users can use the Jtc estimation routine to identify the likely residence location for this individual. In running this routine, a reference grid needs to be overlaid (see chapter 3). For Baltimore County, appropriate coordinates for the lower-left corner are -76.91^0 longitude and 39.19^0 latitude and for the upper-right corner are -76.32^0 longitude and 39.72^0 latitude.

How Accurate are the Methods?

A critical question is how accurate are these methods? The Journey-to-crime model is just that, a model. Whether it involves using a mathematical function or an empirically-derived one, the assumption in the Jtc routine is that the distribution of incidents will provide information about the home base location of the offender. In this sense, it is not unlike the way most crime analysts will work when they are trying to find a serial offender. A typical approach will be to plot the distribution of incidents and routinely search a geographic area in and around a serial crime pattern, noting offenders who have an arrest history matching case attributes (MO, type weapon, suspect description, etc.). Because a high proportion of offenses are committed within a short distance of offender residence's, the method can frequently lead to their apprehension. But, in doing this method, the analysts are not using a sophisticated statistical model.

Test Sample of Serial Offenders

To explore the accuracy of the approach, a small sample of 50 serial offenders was isolated from the database and used as a target sample to test the accuracy of the methods. The 50 offenders accounted for 520 individual crime incidents in the database. To test the Jtc method systematically, the following distribution was selected (Table 13.4). The sample was not random, but was selected to produce a balance in the number of incidents committed and to, roughly, approximate the distribution of incidents by serial offenders. Each of the 50 offenders was isolated as a separate file so that each could be analyzed in *CrimeStat*.

Table 13.4:
Serial Offenders Used in Accuracy Evaluation

Number of Offenders	Number of Crimes Committed by Each Person
4	3
4	4
4	5
4	6
4	7
4	8
3	9
3	10
3	11
2	12
2	13
2	14
2	15
1	16
1	17
1	18
1	19
1	20
1	21
1	22
1	24
1	33
50	520

Identifying the Crime Type

Each of the 50 offenders was categorized by a crime type. Only two of the offenders committed the same crime for all their offenses; most committed two or more different types of crimes. Arbitrarily, each offender was typed according to the crime type that he/she most frequently committed; in the two cases where there was a tie between two crime types, the most severe was selected (i.e., personal crime over property crime). While I recognize that there is arbitrariness in the approach, it seemed a practical solution. Any error in categorizing an offender would be applicable to all the methods. The crime types for the 50 offenders

approximately mirrored the distribution of incidents: larceny (29); vehicle theft (7); burglary (5); robbery (5); assault (2); bank robbery (1); and arson (1).

Identifying the Home Base and Incident Locations

In the database, each of the offenders was listed as having a residence location. For the analysis, this was taken as the *origin* location of the Journey-to-crime trip. Similarly, the incident location was taken as the *destination* for the trip. Operationally, the crime trip is taken as the distance from the origin location to the destination location. However, it is very possible that some crime trips actually started from other locations. Further, many of these individuals have moved their residences over time; we only have the last known residence in the database. Unfortunately, there was no other information in the digital database to allow more accurate identification of the home location. In other words, there may be, and probably are, numerous errors in the estimation of the Journey-to-crime trip. However, these errors would be similar across all methods and should not affect their relative accuracy.

Evaluated Methods

Eleven methods were compared in estimating the likely residence location of the offenders. Four of the methods used the Jtc routines and seven were simple spatial distribution methods (Table 13.5).

The mean center and center of minimum distance are discussed in chapter 4. The center of minimum distance, in particular, is more or less the geographic center of distribution in that it ignores the values of particular locations; thus, locations that are far away from the cluster (extreme values) have no effect on the result. When the center of minimum distance is calculated on a road network in which each segment is weighted by travel time or speed, the result is the center of minimum travel time, the point at which travel time to each of the incidents is minimized. The directional mean, triangulated mean, geometric and harmonic means are discussed in chapter 4.

The Test

Each of these eleven methods were tested with the the files created for the serial offenders. For the seven 'means' (mean center, geometric mean, harmonic mean, directional mean, triangulated mean, center of minimum distance, center of minimum travel time), the mean was itself the best guess for the likely residence location of the offender. For the four Journey-to-crime functions, the grid cell with the highest likelihood estimate was the best guess for the likely residence location of the offender.

Table 13.5:

Comparison Methods for Estimating the Home Base of a Serial Offender

Journey-to-crime Methods

1. Mathematical model for all crimes
2. Mathematical model for specific crime type
3. Kernel density model for all crimes
4. Kernel density model for specific crime type

Spatial Distribution Methods

5. Mean center
6. Center of minimum distance
7. Center of minimum travel time
   (calculated on road network weighted by travel time)
8. Directional mean (weighted) calculated with 'lower left corner' as origin
9. Triangulated mean
10. Geometric mean
11. Harmonic mean

Measurement of Error

For each of the 50 offenders, error was defined as the distance in miles between the 'best guess' and the actual location. For each offender, the distance between the estimated home base (the 'best guess') and the actual residence location was calculated using direct distances. Table 13.6 presents the results. The data show the error by method for each of the 50 offenders. The three right columns show the average error of all methods and the minimum error and maximum errors obtained by a method. The method with the minimum error is boldfaced; for some cases, two methods are tied for the minimum. The bottom three rows show the median error, the average error and the standard deviation of the errors for each method across all 50 offenders.

Results of the Test

The results point to certain conclusions. First, the degree of precision for any of these methods varies considerably. The precision of the estimates vary from a low of 0.0466 miles (about 246 feet) to a high of 75.7 miles. The overall precision of the methods is not very high and is highly variable. There are a number of possible reasons for this, some of which have been discussed above. Each of the methods produces a single parameter from what is, essentially, a probability distribution whereas the distribution of many of these incidents are widely dispersed.

Table 13.6:

Accuracy of Methods for Estimating Serial Offender Residences

(N= 50 Serial Offenders)

Dataset	Number of Crimes	Primary Crime Type	Mean Center Error (miles)	Center of Minimum Distance Error (miles)	Triangulated Mean Error (miles)	Geometric Mean Error (miles)	Harmonic Mean Error (miles)	Jtc Kernel: All Crimes Error (miles)	Jtc Kernel: Crime Type Error (miles)	Jtc Math: All Crimes Error (miles)	Jtc Math: Crime Type Error (miles)	Average Error	All Methods Minimum Error	Maximum Error
3A	3	Larceny	**31.5991**	32.4477	32.4109	31.5995	31.6000	32.7824	32.7880	32.7824	32.7880	**32.3109**	**31.5991**	**32.7880**
3B	3	Larceny	13.2303	12.1683	24.1531	13.2311	13.2319	**10.7526**	14.4929	**10.7526**	11.2501	**13.6959**	**10.7526**	**24.1531**
3C	3	Bank robbery	2.8348	0.9137	2.7767	2.8335	2.8322	**0.6775**	5.8416	**0.6775**	6.0946	**2.8313**	**0.6775**	**6.0946**
3D	3	Burglary	2.9733	3.2603	6.1013	2.9728	**2.9724**	4.6038	3.3883	3.3882	3.7931	**3.7170**	**2.9724**	**6.1013**
4A	4	Vehicle theft	4.2436	4.2670	**3.8217**	4.2436	4.2436	4.2527	4.2364	4.2527	4.2590	**4.2022**	**3.8217**	**4.2670**
4B	4	Larceny	1.9618	0.3100	2.0563	1.9621	1.9623	0.3125	**0.2018**	0.3125	0.2784	**1.0397**	**0.2018**	**2.0563**
4C	4	Larceny	4.4733	4.4733	4.6789	4.4733	4.4733	4.9681	4.3563	**4.2637**	4.3563	**4.5018**	**4.2637**	**4.9681**
4D	4	Assault	0.2925	0.1905	**0.0466**	0.2925	0.2926	0.0703	0.0703	0.0703	0.4560	**0.1979**	**0.0466**	**0.4560**
5A	5	Larceny	17.3308	16.6459	17.8985	17.3292	17.3276	**15.9738**	17.8655	15.9739	16.4526	**16.9775**	**15.9738**	**17.8985**
5B	5	Larceny	1.3609	0.2481	1.7733	1.3586	1.3564	**0.2068**	0.6974	0.5140	0.6974	**0.9126**	**0.2068**	**1.7733**
5C	5	Larceny	2.2458	2.6832	16.4518	2.2450	2.2442	2.7886	2.4205	2.7886	3.0922	**4.1067**	**2.2442**	**16.4518**
5D	5	Larceny	0.9169	0.2250	0.2371	0.9171	0.9174	**0.1577**	0.4267	**0.1577**	0.4267	**0.4869**	**0.1577**	**0.9174**
6A	6	Larceny	5.1837	5.2081	7.9621	5.1837	5.1837	5.1271	**4.8554**	4.9393	5.2256	**5.4298**	**4.8554**	**7.9621**
6B	6	Vehicle theft	1.3720	1.1869	**0.9625**	1.3710	1.3700	3.1126	2.3800	1.3566	2.0831	**1.6883**	**0.9625**	**3.1126**
6C	6	Larceny	1.3199	0.3157	1.7928	1.3192	1.3184	**0.2580**	0.5272	**0.2580**	0.5272	**0.8485**	**0.2580**	**1.7928**
6D	6	Larceny	3.2458	2.3324	6.5209	3.2431	3.2405	**1.2506**	2.6253	1.9718	1.9718	**2.9336**	**1.2506**	**6.5209**
7A	7	Larceny	3.9023	3.4185	**2.3176**	3.9022	3.9021	2.7419	3.0532	3.1364	3.0532	**3.2697**	**2.3176**	**3.9023**
7B	7	Larceny	12.4100	9.2973	14.8293	12.4107	12.4115	**8.5357**	8.6148	**8.5357**	8.8275	**10.6525**	**8.5357**	**14.8293**
7C	7	Burglary	5.0501	7.1477	10.8567	5.0481	**5.0460**	7.9975	7.9975	7.9975	7.6274	**7.1965**	**5.0460**	**10.8567**
7D	7	Larceny	2.2686	0.7733	75.7424	2.2684	2.2682	**0.0892**	0.7191	**0.0892**	0.7191	**9.4375**	**0.0892**	**75.7424**
8A	8	Larceny	6.0298	**6.0165**	6.2653	6.0264	6.0229	8.4210	6.2962	6.2022	6.1166	**6.3774**	**6.0165**	**8.4210**
8B	8	Larceny	**1.0041**	1.1437	2.1776	1.0042	1.0042	1.7475	1.3510	1.5298	1.3510	**1.3681**	**1.0041**	**2.1776**
8C	8	Larceny	1.3059	1.6944	1.3684	1.3043	1.3027	2.1513	**1.2020**	2.1513	1.8707	**1.5946**	**1.2020**	**2.1513**
8D	8	Vehicle theft	3.5794	2.3780	5.5915	3.5809	3.5825	**0.5900**	1.3340	1.9133	1.3340	**2.6537**	**0.5900**	**5.5915**
9A	9	Robbery	5.2527	5.7156	**4.8574**	5.2529	5.2532	7.8257	7.1961	6.2520	5.9265	**5.9480**	**4.8574**	**7.8257**
9B	9	Larceny	8.1923	10.6555	**6.9916**	8.1886	8.1850	12.4578	10.3957	12.4578	12.0514	**9.9529**	**6.9916**	**12.4578**
9C	9	Robbery	3.7778	3.8454	11.0042	3.7758	**3.7738**	4.9015	5.1862	4.6206	4.3445	**5.0255**	**3.7738**	**11.0042**
10A	10	Larceny	0.9358	0.5159	1.1003	0.9355	0.9353	**0.0606**	0.3720	0.2601	0.7172	**0.6481**	**0.0606**	**1.1003**
10B	10	Larceny	2.8581	3.4940	14.2219	2.8536	**2.8491**	6.4051	6.5709	10.3095	6.4758	**6.2264**	**2.8491**	**14.2219**
10C	10	Larceny	0.8052	**0.7251**	5.5938	0.8050	0.8049	0.9059	0.8404	0.9060	1.2786	**1.4072**	**0.7251**	**5.5938**
11A	11	Vehicle theft	**2.9127**	3.2715	3.1192	2.9130	2.9134	3.6936	3.4335	3.4282	3.2087	**3.2104**	**2.9127**	**3.6936**
11B	11	Robbery	0.3250	0.3250	0.2513	0.3250	0.3250	0.4235	**0.2263**	0.4235	0.7011	**0.3695**	**0.2263**	**0.7011**
11C	11	Vehicle theft	1.2689	1.7157	1.4750	1.2709	1.2729	2.8945	**0.6984**	2.8945	2.2049	**1.7440**	**0.6984**	**2.8945**
12A	12	Larceny	3.3881	4.2334	10.9241	3.3867	3.3852	6.4050	**3.2639**	5.5843	5.2132	**5.0871**	**3.2639**	**10.9241**
12B	12	Larceny	0.5562	0.5361	2.8003	0.5562	0.5562	0.7897	0.6709	0.7897	0.9631	**0.9132**	**0.5361**	**2.8003**
13A	13	Larceny	6.3282	7.2857	**6.0244**	6.3248	6.3213	7.6438	7.4607	7.6438	7.9915	**7.0027**	**6.0244**	**7.9915**
13B	13	Assault	**1.4943**	**1.4943**	1.5279	1.4944	1.4944	1.6501	1.5954	1.6501	2.0824	**1.6092**	**1.4943**	**2.0824**
14A	14	Larceny	1.9363	0.8706	1.4498	1.9365	1.9368	0.3434	0.6058	**0.2596**	0.7631	**1.1224**	**0.2596**	**1.9368**
14B	14	Arson	0.6898	0.3727	0.8086	0.6899	0.6900	**0.3359**	**0.3359**	**0.3359**	0.6213	**0.5422**	**0.3359**	**0.8086**
15A	15	Vehicle theft	0.7282	0.7189	**0.3362**	0.7277	0.7271	0.8155	0.4855	0.8155	1.5128	**0.7630**	**0.3362**	**1.5128**
15B	15	Robbery	0.4914	0.4914	0.8254	0.4914	0.4914	0.6468	0.5693	0.6468	0.6546	**0.5898**	**0.4914**	**0.8254**
16A	16	Vehicle theft	2.1107	2.0995	8.2311	2.1107	2.1107	**1.5957**	1.6404	2.5911	2.4033	**2.7659**	**1.5957**	**8.2311**
17A	17	Burglary	1.6484	0.3093	1.0227	1.6461	1.6438	**0.2879**	**0.2879**	**0.2879**	0.5268	**0.8512**	**0.2879**	**1.6484**
18A	18	Larceny	0.6308	0.4196	1.0876	0.6329	0.6349	**0.2132**	0.3383	**0.2132**	0.6985	**0.5410**	**0.2132**	**1.0876**
19A	19	Larceny	**8.6462**	9.4195	8.6772	8.6486	8.6511	10.2869	9.2708	9.7022	9.5548	**9.2064**	**8.6462**	**10.2869**
20A	20	Burglary	6.3520	5.7969	28.3094	6.3486	6.3452	**0.5934**	0.8673	**0.5934**	0.7945	**6.2223**	**0.5934**	**28.3094**
21A	21	Burglary	1.2396	0.8861	1.2776	1.2393	1.2390	0.5243	0.5243	1.0253	**0.4965**	**0.9391**	**0.4965**	**1.2776**
22A	22	Larceny	3.6828	2.6232	**2.0949**	3.6803	3.6777	2.4937	2.8944	2.4937	2.8944	**2.9484**	**2.0949**	**3.6828**
24A	24	Larceny	1.7959	0.5892	2.3033	1.7975	1.7991	**0.2658**	0.3574	0.4222	0.6587	**1.1099**	**0.2658**	**2.3033**
33A	33	Robbery	3.9901	5.0481	7.2505	3.9940	3.9979	7.9485	7.6939	8.1907	7.9439	**6.2286**	**3.9901**	**8.1907**
		Median Error =	**2.5517**	**2.2159**	**3.4704**	**2.5509**	**2.5502**	**1.9494**	**2.0102**	**2.0615**	**2.1440**			
		Mean Error =	**4.0434**	**3.8441**	**7.6472**	**4.0429**	**4.0424**	**4.0395**	**4.0305**	**4.0163**	**4.1467**			
		SD Error =	**5.2166**	**5.3845**	**12.0642**	**5.2166**	**5.2166**	**5.5678**	**5.6237**	**5.5398**	**5.4177**			

Few of the offenders had such a concentrated pattern that only a single location was possible. Since these are probability distributions, not everyone follows the 'central tendency'. Also, some of these offenders may have moved during the period indicated by the incidents, thereby shifting the spatial pattern of incidents and making it difficult to identify the last residence.

A second conclusion is that, for any one offender, the methods produce similar results. For many of the offenders the difference between the best estimate (the minimum error) and the worst estimate (the maximum error) is not great. Thus, the simple methods are generally as good (or bad) as the more sophisticated methods.

Third, across all methods, the center of minimum travel time, which is calculated on a road network (see chapters 3 and 30), and it's distance-based 'cousin' - the center of minimum travel time, had the lowest average error. Thus, the approximate geographic center of the distribution where travel time to each of the incidents was minimal produced as good an estimate as the more sophisticated methods. However, it was not particularly close (3.84 miles, on average). The worst method was the triangulated mean which had an average error of 7.6472 miles. The triangulated mean is produced by vector geometry and will not necessarily capture the center of the distribution. Other than this, there were not great differences. This reinforces the point above that the methods are all, more or less, describing the central tendency of the distribution. For offenders that don't live in the center of their distribution, the error of a method will necessary be high.

Looking at each of the 50 offenders, the methods vary in their efficacy. For example, the Jtc kernel function for all crimes was the best or tied for best for 17 of the offenders, but was also the worst or tied for worst for 9. Similarly, the Jtc kernel function for the specific crimes was best or tied for best for 8 of the offenders, but worse for 4. Even the most consistent was best for 4 offenders, but also worst for one. On the other hand, the triangulated mean, which had the worst overall error, produced the best estimate for 9 of the individuals while it produced the worst estimate for 25 of the individuals. Thus, the triangulated mean tends to be very accurate or very inaccurate; it had the highest variance, by far.

Fourth, the median error is smaller than the average error. That is, the median is the point at which 50% of the cases had a smaller error and 50% had a larger error. Overall, most of the cases were found within a shorter distance than the average would indicate. This indicates that several cases had very large errors whereas most had smaller errors; that is, they were *outliers*. Over all methods, the Jtc kernel approach for all crimes had the lowest median error (1.95 miles). In fact, all four Jtc methods had smaller median errors than the simple centrographic methods. In other words, they are more accurate than the centrographic methods most of the time. The problem in applying this logic in practice, however, is that one would not know if the case being

studied is typical of most cases (in which case, the error would be relatively small) or whether it was an outlier. In other words, the median would define a search area that captured about 50% of the cases, but would be very wrong in the other 50%. If we could somehow develop a method for identifying when a case is 'typical' and when it isn't, increased accuracy will emerge from the Jtc methods. But, until then, the simple center of minimum travel time will be the most accurate method.

Fifth, the amount of error varies by the number of incidents. Table 13.7 below shows the average error for each method as a function of three size classes: 1-5 incidents; 6-9 incidents; and 10 or more incidents. As can be seen, for each of the ten methods, the error decreases with increasing number of incidents. In this sense, the measured error is responsive to the sample size from which it is based. It is, perhaps, not surprising that with only a handful of incidents no method can be very precise.

Sixth, the relative accuracy of each of these methods varies by sample size. The method or methods with the minimum error are boldfaced. For a limited number of incidents (1-5), the Jtc mathematical function for all crimes (i.e., the negative exponential with the parameters from Table 13.5) produced the estimate with the least error, followed by the Jtc kernel function for all crimes; the was the third best. The differences in error between these were not very great. For the middle category (6-9 incidents), the center of minimum distance produced the least error followed by the Jtc mathematical function for the specific crime type. For those offenders who had committed ten or more crimes, the Jtc kernel function for the specific crime type produced the best estimate, followed by the center of minimum distance. The two mathematical functions produced the least accuracy for this sub-group, though again the differences in error are not very big (2.2 miles for the best compared to 2.7 miles for the worst). In other words, only with a sizeable number of incidents does the Jtc kernel density approach for specific crimes produce a good estimate. It is better than the other approaches, but only slightly better than the simple measure of the center of minimum distance.

Search Area for a Serial Offender?

A number of researchers have been interested in the concept of a search area for the police (Rossmo, 2000; Canter, 2003). The concept is that the Journey-to-crime method can define a small search area within which there is a higher probability of finding the offender. The average or median error discussed above can be used to define such a search area if treated as a radius of a circle. While intuitive, this does not necessarily represent a meaningful statistic. For example, taking the average error of the center of minimum distance (3.84 miles) would produce a search area of 46.4 square miles, not exactly a small area in which to find a serial offender. Even if we take the median error of 1.94 miles from the Jtc kernel approach for all crimes (1.94

Table 13.7:

Method Estimation Error and Sample Size

Average Error of Method by Number of Incidents (miles)

Number of Incidents	*	Mean Center	Center of Minimum Distance	Triangulated Mean	Geometric Mean	Harmonic Mean	Jtc Kernel: All	Jtc Kernel: Crime types	Jtc Math: All	Jtc Math: Crime types	*	All Methods Average Error	Minimum Error
3-5	*	6.9553	6.4861	9.3672	6.9160	6.9545	6.4622	7.2321	**6.3278**	6.9954	*	**7.0774**	**6.3278**
6-9	*	4.2596	**4.0753**	10.6160	4.3331	4.2576	4.4805	4.2489	4.2274	4.2020	*	**4.9667**	**4.0753**
10+	*	2.3832	2.3149	4.8136	2.4575	2.3827	2.4880	**2.2176**	2.6725	2.6243	*	**2.7060**	**2.2176**

miles) will still produce a search area of 11.9 square miles, and it would be correct only half the time. These methods are still very imprecise.

Confirmation of These Results

This analysis was first conducted in 2000 with version 1.1 of *CrimeStat* (Levine, 2000). Since then, it has been confirmed with several studies. Snook, Zito, Bennell, and Taylor (2005) found a similar result with 16 serial burglars who had committed 10 or more incidents in the United Kingdom. Simple measures did as well as the complex measures.

Snook, Canter and Bennell (2002) compared the journey-to-crime method with the judgment of student volunteers and found that the journey-to-crime method was not significantly more accurate than the student judgments. A subsequent study found that simple training of geographic principles improved the predictive accuracy of police officers in predicting the residence location of 36 serial offenders and were as accurate as the journey-to-crime method (Bennell, Snook, Taylor, Corey, & Keyton, 2007).

Paulsen (2006) conducted an analysis of 247 serial offenders from Baltimore County and found that simple centrographic measures were more accurate than the more complex journey-to-crime method. He also compared four different software packages in terms of their accuracy. He found that all packages had about the same degree of accuracy, that simple centrographic approaches were generally more accurate, that there were substantial differences in the accuracy by different crime types, but, most importantly, none of the methods were very accurate.

Bennell, Taylor , and Snook (2007) examined a number of studies of geographic profiling and argued that simple heuristics can provide as much accuracy as more sophisticated methods, with much less effort and cost to a police department.

In other words, in several independent tests of the accuracy of the journey-to-crime approach to geographical profiling, simple measures, particularly the center of minimum distance, do as well as, if not better than, the journey-to-crime approach.

Theoretical Limitations

There are also some theoretical problems in journey-to-crime analysis which limits the method's ability to predict the origin location of a serial offender. First, the method is entirely based on distance traveled from a theoretical origin that will be estimated by the method. The means for assigning the distance is the journey-to-crime function that has been chosen (normal, quartic, exponential etc). However, transportation modelers usually conceptualize travel

distance not as an independent variable but the result of predispositions, attractions, and networks (Domencich & McFadden, 1975; Ortuzar & Willumsen, 2001; Culp, 2002). Different individuals have predispositions to travel that vary by gender as well as by age (Levine & Lee, 2012).

Second, the distance function in a journey-to-crime model is assumed to operate in any direction. In reality, there is a large amount of asymmetry in the direction of travel because attractions are more concentrated towards the center of a metropolitan area (FCCDR, 1994; Bruegmann, R.,2008; Bertaud, 2009; SCTLC 2009). For example, an offender who lives in a suburb is more likely to travel towards the center of a metropolitan area than away from it because there are more opportunities in the center than farther away. Similarly, offenders in a high crime neighborhood of a metropolitan area are more likely to travel to other high crime neighborhoods and not just in any direction. In addition travel is restrained by physical and social barriers (Bernasco & Block, 2009). The journey-to-crime approach assumes a uniform cost function that applies to everyone.

Third, criminal opportunities (or attractions) are never measured, but are inferred from the pattern of crime incidents. That is, the crime location is assumed to represent the opportunity for the offender, but the attraction for the offender is never measured. Therefore, the distance traveled is assumed to represent the likelihood of travel by the offender without any differentiation by place, crime type, type of person, or environment. As a pragmatic tool for informing a police search, one could argue that this is not important. However, in a different location or crime set, the distance function is liable to differ substantially.

Fourth, it is not clear whether knowing an offender's 'cognitive map' will help in prediction. There have been no evaluations that have compared a strictly statistical approach with an approach that utilizes information about the offender as he or she understands the environment. It cannot be assumed that integrating information about the perception of the environment will aid prediction. In most travel demand forecasts that transportation engineers and planners make, cognitive information about the environment is not utilized except in the definition of trip purpose (i.e., what the purpose of the trip was). The models use the actual trips by origin and destination as the basis for formulating predictions, not the understanding of the trip by the individual. Understanding is important from the viewpoint of developing theory or for ways to communicate with people. But, it is not necessarily useful for prediction. In short, understanding and prediction are not the same thing.

In short, journey-to-crime methodology is limited both theoretically and empirically. Theoretically, it ignores the distribution of opportunities and focuses only on the cost of travel. Empirically, the method has a substantial amount of error and cannot even do as well as simple measures in terms of prediction. Finally, existing journey-to-crime methodologies assume that

the awareness space of serial offenders surrounds the offender's anchor point. But, there are offenders who commit crimes far from where they reside or from their anchor point, so called 'Commuter' offenders (Paulson, 2006).

Cautionary Notes

There are certain cautions that must be considered in using either of these Journey-to-crime methods (the mathematical or the empirical). First, a simple technique, such as the center of minimum distance, may be as good as a more sophisticated technique. It does not always follow that a sophisticated method will produce any more accuracy than a simple one.

Second, there are other limitations to the technique. The model must be calibrated for each individual jurisdiction. Further, it must be periodically re-calibrated to account for changes in crime patterns. For example, in using the mathematical model, one cannot take the parameters estimated for Baltimore County (Table 13.3) and apply them to another city or if using the kernel density method take the results found at one time period and assume that they will remain indefinitely. The model is a probability model, not a guarantee of certainty. It provides guesses based on the similarity to other offenders of the same type of crime. In this sense, a particular serial offender may not be typical and the model could actually orient police wrongly if the offender is different from the calibration sample. It will take insight by the investigating officers to know whether the pattern is typical or not.

Third, as a theoretical model, the Journey-to-crime approach is quite simple. It is based on a distribution of incidents and an assumed travel distance decay function. As mentioned above, the method does not utilize information on the distribution of target opportunities nor does it utilize information on the travel mode and route that an offender takes. It is purely a statistical model.

The research area of geographic profiling attempts to go beyond statistical description and understand the cognitive maps that offenders use as well as how these interact with their motives. This is good and should clearly guide future research. But it has to be understood that the theory of offender travel behavior is not very well developed, certainly compared to other types of travel behavior. Further, some types of crime trips may not even start from an offender's residence, but may be referenced from another location, such as vehicle thefts occurring near disposal locations. Routine activity theory would suggest multiple origins for crimes (Cohen & Felson, 1979).

The existing models of travel demand used by transportation planners (which have themselves been criticized for being too simple) measure a variety of factors that have only been marginally included in the crime travel literature - the availability of opportunities, the

concentration of offender types in certain areas, the mode of travel (i.e., auto, bus, walk), the specific routes that are taken, the interaction between travel time and travel route, and other factors. It will be important to incorporate these elements into the understanding of Journey-to-crime trips to build a much more comprehensive theory of how offenders operate. Travel behavior is very complicated and we need more than a statistical distance model to adequately understand it.

In the next chapter, a Bayesian approach to journey-to-crime modeling will be discussed in which additional information about the origin location for the offender is introduced into the model in order to improve the distance estimate. As we shall see, the method is more accurate and more precise than the journey-to-crime function.

Draw Crime Trips

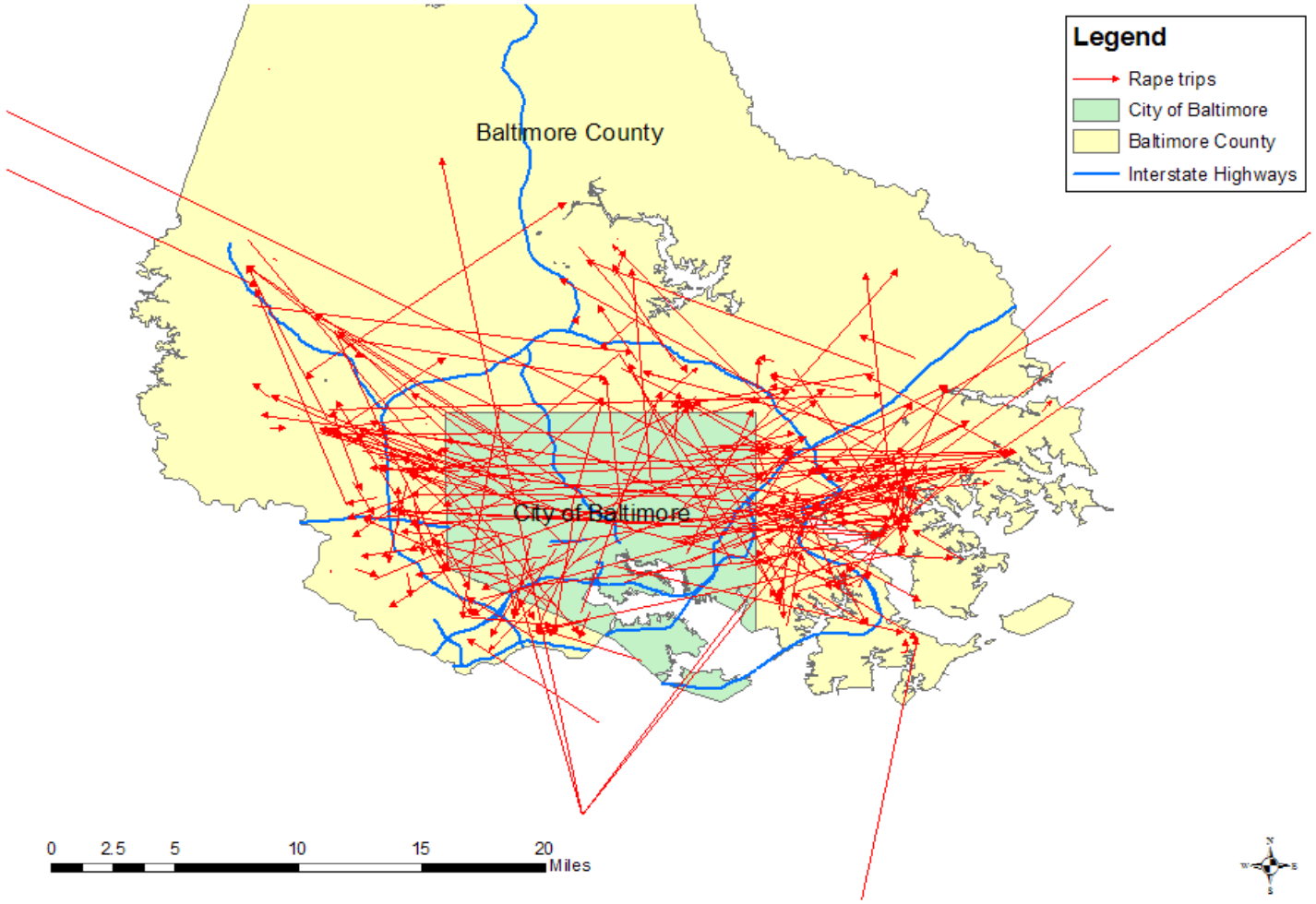
The Journey-to-crime module also includes one utility that can help visualize the pattern before selecting a particular estimation model. This is a Draw Crime Trips routine that simply draws lines between the origin and destination of individual crime trips. The X and Y coordinates of an origin and destination location are input and the routine draws a line in *ArcGIS* 'shp', *MapInfo* 'mif', *Google Earth 'kml'* or various Ascii formats.

Figure 13.23 illustrates the drawing of the known travel distances for 444 rape cases for which the residence location of the rapist was known. Of the 444 cases, 113 (or 25.5%) occurred in the residence of the rapist. However, for the remaining 331 cases, the rape location was not the residence location. As seen, many of the trips are of quite long distances. This would suggest the use of a Journey-to-crime function that has many trips at zero distance but with a more gradual decay function.

Figure 13.23:

Journey-to-rape Trips

Distance from Residence to Rape Location for 444 Known Offenders



References

Andersson, T. (1897). *Den Inre Omflyttningen*. Norrland: Mälmo.

Anselin, L. & Madden, M. (1990). *New Directions in Regional Analysis*. Belhaven Press: New York.

Bennell, C., Snook, B., Taylor, P. J., Corey, S. & Keyton, J. (2007). It's no riddle, choose the middle: The effect of number of crimes and topographical detail on police officer predictions of serial burglars' Home Locations. *Criminal Justice And Behavior*, 34 (1), 119-132.

Bennell, C., Taylor, P. J., &Snook, B. (2007). Clinical versus actuarial geographic profiling approaches: A review of the research. *Police Practice and Research*, 8(4), 335-345.

Bernasco, W. & Block, R. (2009). Where offenders choose to attack: A discrete choice model of robberies in Chicago. *Criminology* 47(1): 93-130.

Bernasco, W. & Nieuwbeerta, P. (2005). How do residential burglars select target areas?. *British Journal of Criminology* 44: 296-315.

Bertaud, A. (2009). *The Spatial Structure of Cities: International Examples of the Interaction of Government, Topography, and Markets*. Alain-Bertaud.com.
http://AB_China_course_part3_PPT.ppt

Blumin, D. (1973). *Victims: A Study of Crime in a Boston Housing Project*. City of Boston, Mayor's Safe Street Act, Advisory Committee: Boston.

Boggs, S. L. (1965). Urban crime patterns, *American Sociological Review*, 30, 899-908.

Bossard, E. G. (1993). RETAIL: Retail trade spatial interaction. In Klosterman, R. E. Brail, R. & Bossard, E. G. *Spreadsheet Models for Urban and Regional Analysis*. Center for Urban Policy Research, Rutgers University: New Brunswick, NJ, 419-448.

Brantingham, P. L. & Brantingham, P. J. (1981). Notes on the geometry of crime. In Brantingham, P. J. & Brantingham, P. L, *Environmental Criminology*. Waveland Press, Inc.: Prospect Heights, IL, 27-54.

Bright, M. L. & Thomas, D. S. (1941). Interstate migration and intervening opportunities, *American Sociological Review*, 6, 773-783.

References (continued

Bruegmann, R. (2008). Driving works. *Forbes Magazine*. July 29, http://www.forbes.com/2008/07/29/commuting-suburbs-future-lead-commuting08-cx_rb_0729bruegmann.html

Burgess, E. W. (1925). The growth of the city: an introduction to a research project. In R. E. Park, R. E. Burgess, E. W. & Mackensie, R. D. (ed), *The City*. University of Chicago Press: Chicago, 47-62.

Canter, D. (2003). *Dragnet: A Geographical Prioritisation Package*. Center for Investigative Psychology, Department of Psychology, The University of Liverpool: Liverpool, UK. http://www.i-psy.com/publications/publications_dragnet.php.

Canter, D. (1994). *Criminal Shadows: Inside the Mind of the Serial Killer*. Harper Collins Publishers: London.

Canter, D. V, Coffey, T., Huntley, M., & Missen, C. (2000). Predicting serial killers' home base using a decision support system. Journal of Quantitative Criminology, 16, 457-478.

Canter, D. & A. Gregory (1994). Identifying the residential location of rapists, *Journal of the Forensic Science Society*, 34 (3), 169-175.

Canter, D. & Larkin, P. (1993). The environmental range of serial rapists, *Journal of Environmental Psychology*, 13, 63-69.

Canter, D. V., & Snook, B. (1999). *Modelling the home location of serial offenders*. Paper presented at the meeting of the Crime Mapping Research Center, Orlando, FL. December.

Canter, D. & Tagg, S. (1975). Distance estimation in cities, *Environment and Behaviour*, 7, 59-80.

Capone, D. L. & Nichols Jr, W. W. (1975). Crime and distance: an analysis of offender behaviour in space, *Proceedings, Association of American Geographers*, 7, 45-49.

Cliff, A. D. & Haggett, P. (1988). *Atlas of Disease Distributions*. Blackwell Reference: Oxford.

Cohen, L.E. & Felson, M. (1979) Social change and crime rate trends: a routine activity approach, *American Sociological Review*, 44: 588-608.

References (continued

Culp, M. (2005). Travel Model Improvement Program Annual Report FY04. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/planning/tmip/about/annual_reports/fy2004/. Access April 1, 2012.

Curtis, L. A. (1974). *Criminal Violence*. Lexington Books: Lexington, MA.

Demographia (1999). *U.S. Central Cities and Suburban Crime Rates Ranked: 1999*. Wendell Cox Consultancy: Belleville, IL. http://www.demographia.com/db-crime99r.htm.

Demographia (1998). *U. S. Metropolitan Areas: 1998 Central City and Suburban Population*. Wendell Cox Consultancy: Belleville, IL. http://www.demographia.com/db-usmsacc98.htm.

Domencich, T. & McFadden, D. (1975). *Urban Travel Demand: A Behavioral Analysis*. North Holland Publishing Company: Amsterdam and Oxford (republished in 1996). Also found at http://emlab.berkeley.edu/users/mcfadden/travel.html. Accessed April 1, 2012.

Eldridge, J. D. & Jones, J. P. (1991). Warped space: a geography of distance decay, *Professional Geographer*, 43 (4), 500-511.

Everitt, B. S. (2011). *Cluster Analysis* (5th edition). J. Wiley: London.

FCCDR (1994). *Sustainable Community Design Principles*. Tampa, FL: Florida Center for Community Design and Research. http://www.fccdr.usf.edu/upload/projects/tlushtml/tlus100.htm. Accessed April 1, 2012.

Felson, M. (2002). *Crime & Everyday Life* (3rd Ed). Sage: Thousand Oaks, CA.

Field, B. & MacGregor, B. (1987). *Forecasting Techniques for Urban and Regional Planning*. UCL Press, Ltd: London.

Foot, D. (1981). *Operational Urban Models*. Methuen: London.

Freedman, D. A. (1999). Ecological inference and ecological fallacy. *International Encyclopedia of the Social and Behavioral Sciences*, Technical Report No. 549, October. http://www.stanford.edu/class/ed260/freedman549.pdf. Accessed March 26, 2012.

Fritzon, K. (2001). An Examination of the Relationship between Distance Travelled and Motivational Aspects of Firesetting Behaviour. *Journal of Environmental Psychology*, 21, 45-60.

References (continued

Golden Software. 2008. *Surfer® for Windows (Ver. 10)*. Golden Software, Inc.: Golden, CO.

Groff, E. R. & McEwen, J. T (2005). Disaggregating the Journey to Homicide. In Wang, F. (ed.), *Geographic Information Systems and Crime Analysis*. Idea Group Publishing: Hershey, PA.

Hägerstrand, T. (1957). Migration and area: survey of a sample of Swedish migration fields and hypothetical considerations on their genesis. *Lund Studies in Geography, Series B, Human Geography*, 4, 3-19.

Haggett, P. & Arnold, E. (1965). *Locational Analysis in Human Geography* (1st edition). Edward Arnold: London.

Haggett, P., Cliff, A. D. & Frey, A. (1977). *Locational Analysis in Human Geography* (2nd edition). Edward Arnold: London.

Harries, K. (1980). *Crime and the Environment*. Charles C. Thomas Press: Springfield.

Hodge, S. & Canter, D. (1998) Victims and Perpetrators of Male Sexual Assault. *Journal of Interpersonal Violence*, 1 (April), 222-239.

Huff, D. L. (1963). A probabilistic analysis of shopping center trade areas. *Land Economics*, 39, 81-90.

Isard, W. (1979). *Location and Space-Economy: A General Theory Relating to Industrial Location, Market Areas, Land Use, Trade, and Urban Structure* (originally published 1956). Program in Urban and Regional Studies, Cornell University: Ithaca, NY.

Isard, W. (1960). *Methods in Regional Analysis*. John Wiley & Sons: New York.

Isbel, E. C. (1944). Internal migration in Sweden and intervening opportunities, *American Sociological Review*, 9, 627-639.

Kind, S. S. (1987). Navigational ideas and the Yorkshire Ripper investigation. *Journal of Navigation*, 40 (3), 385-393.

Krueckeberg, D. A. & Silvers, A. L. (1974). *Urban Planning Analysis: Methods and Models*. John Wiley & Sons: New York.

References (continued

Langbein, L. I. & Lichtman, A. J. (1978). *Ecological Inference*. Sage University Paper series on Quantitative Applications in the Social Sciences, series no. 07-010. Beverly Hills and London: Sage Publications.

LeBeau, J. L. (1987a). The journey to rape: geographic distance and the rapist's method of approaching the victim, *Journal of Police Science and Administration*, 15 (2), 129-136.

LeBeau, J. L. (1987b). The methods and measures of centrography and the spatial dynamics of rape, *Journal of Quantitative Criminology*, 3 (2), 125-141.

Levine, N. (2007). Crime travel demand and bank robberies: Using CrimeStat III to model bank robbery trips. *Social Science Computer Review*, 25(2), 239-258.

Levine, N. (2000). *CrimeStat: A Spatial Statistics Program for the Analysis of Crime Incident Locations* (version 1.1). Ned Levine and Associates, Annandale, VA.; National Institute of Justice, Washington, DC. August 2000. Update chapter.

Levine, N. & Lee, P. (2012). Journey-to-crime by Gender and Age Group in Manchester, England. In Leitner, Michael (ed), *Crime Modeling and Mapping Using Geospatial Technologies,* Springer. In press.

Levine, N. & Canter, P. (2011). Linking origins with destinations for DWI motor vehicle crashes: An application of crime travel demand modeling. *Crime Mapping*, 3, 7-41.

Lottier, S. (1938). Distribution of criminal offences in metropolitan regions, *Journal of Criminal Law, Criminology, and Police Science*, 29, 37-50.

NCHRP (1995). *Travel Estimation Techniques for Urban Planning*. Project 8-29(2). National Cooperative Highway Research Program, Transportation Research Board: Washington, DC. http://www4.trb.org/trb/crp.nsf/0/647c1cb3a6b6bfe285256748005619fa?OpenDocument

Oppenheim, N. (1980). *Applied Models in Urban and Regional Analysis*. Prentice-Hall, Inc.: Englewood Cliffs, NJ.

Ortuzar, J. D. & Willumsen, L. G. (2001). *Modeling Transport* (3rd edition). New York: John Wiley and Sons.

References (continued

Paulsen, D. (2006). Connecting the dots: assessing the accuracy of geographic profiling software. *Policing: An International Journal of Police Strategies and Management*. 29 (2), 306-334.

Pettiway, L. E. (1995). Copping crack: The travel behaviour of crack users. *Justice Quarterly, 12*(3), 499-524.

Phillips, P. D. (1980) Characteristics and typology of the journey to crime. In Georges-Abeyie, D. E. & Harries, K. D. (eds), *Crime: A Spatial Perspective*, Columbia Univ. Press: New York, 156-166.

Pyle, G. F. (1974). *The Spatial Dynamics of Crime*. Department of Geography Research Paper No. 159, University of Chicago: Chicago.

Ravenstein, E. G. (1885). The laws of migration. *Journal of the Royal Statistical Society*. 48.

Reilly, W. J. (1929). Methods for the study of retail relationships. *University of Texas Bulletin*, 2944.

Rengert, G., Piquero, A. R., & Jones, P. R. (1999). Distance decay re-examined, *Criminology*, 37 (2), 427-445.

Rengert, G. F (1981). Burglary in Philadelphia: a critique of the opportunity structure model. In Brantingham, P. J. & Brantingham, P. L., *Environmental Criminology*. Waveland Press, Inc.: Prospect Heights, IL, 189-202.

Rengert, G. F. (1975). Some effects of being female on criminal spatial behavior. *The Pennsylvania Geographer*, 13 (2), 10-18.

Repetto, T. A. (1974). *Residential Crime*. Ballinger: Cambridge, MA.

Rhodes, W. M. & Conly, C. (1981). Crime and mobility: an empirical study. In Brantingham, P. J. & Brantingham, P. L., *Environmental Criminology*. Waveland Press, Inc.: Prospect Heights, IL, 167-188.

Rossmo, D. K. (2000). *Geographic Profiling*. CRC Press: Boca Raton Fl.

References (continued

Rossmo, D. K. (1997). Geographic profiling. In Jackson, J. L. & Bekerian, D. A., *Offender Profiling: Theory, Research and Practice*. John Wiley & Sons: Chicester, 159-175.

Rossmo, D. K. (1995). Overview: multivariate spatial profiles as a tool in crime investigation. In Block, C. R., Dabdoub, M & Fregly, S., *Crime Analysis Through Computer Mapping*. Police Executive Research Forum:Washington, DC. 65-97.

Rossmo, D. K. (1993a). Multivariate spatial profiles as a tool in crime investigation. In Block, C. R. & Dabdoub, M. (eds), *Workshop on Crime Analysis Through Computer Mapping: Proceedings*. Illinois Criminal Justice Information Authority and Loyola University Sociology Department: Chicago. (Library of Congress HV7936.C88 W67 1993).

Rossmo, D. K. (1993b). Target patterns of serial murderers: a methodological model. *American Journal of Criminal Justice*, 17, 1-21.

Rushton, G. (1979). *Optimal Location of Facilities*. COMPress: Wentworth, NH.
SPSS, Inc. (1999). *SPSS 9.0 for Windows*. SPSS, Inc.: Chicago.

SCTLC. (2009). *Patterns of Development*. Sonoma County, CA: Sonoma County Transportation and Land Use Coalition, http://www.sonomatlc.org/LandUse/Patterns.htm. Accessed April 1, 2012.

Shaw, C. R. (1929). *Delinquency Areas*. University of Chicago Press: Chicago.

Smith, T. S. (1976). Inverse distance variations for the flow of crime in urban areas. *Social Forces*, 25(4), 804-815.

Smith, W. Bond, J. W., & Townsley, M. (2009). Determining how journeys-to-crime vary: Measuring inter- and intra-offender crime trip distributions. In Weisburd, D., Bernasco, W., & Bruinsma, G. (eds.), *Putting Crime in its Place: Units of Analysis in Spatial Crime Research*. New York: Springer.

Snook, B. (2004). Individual differences in distance travelled by serial burglars. *Journal of Investigative Psychology and Offender Profiling, 1*, 53-66.

Snook, B., Cullen, R. M., Mokros, A., & Harbort, S. (2005). Serial murderers' spatial decisions: factors that influence crime location choice. *Journal of Investigative Psychology and Offender Profiling, 2*, 147-164.

References (continued

Snook, B., Zito, M., Bennell, C. & Taylor, P. J. (2005). On the complexity and accuracy of geographic profiling strategies. *Journal of Quantitative Criminology*, 21 (1), 1-26.

Snook, B., Canter, D. V., & Bennell, C. (2002). Predicting the home location of serial offenders: A preliminary comparison of the accuracy of human judges with a geographic profiling system. *Behavioural Sciences and The Law*, 20, 109-118.

Snyder, J. P. (1987). *Map Projections - A Working Manual*. U.S. Geological Survey Professional Paper 1395. U. S. Government Printing Office: Washington, DC.

Stewart, J. Q. (1950). The development of social physics. *American Journal of Physics*, 18, 239-53.

Stopher, P. R. & Meyburg, A. H. (1975). *Urban Transportation Modeling and Planning.* Lexington, MA: Lexington Books.

Stouffer, S. A. (1940). Intervening opportunities: a theory relating mobility and distance. *American Sociological Review*, 5, 845-67.

Taylor, P. J. (1970). *Interaction and Distance: An Investigation into Distance Decay Functions and a Study of Migration at a Microscale*. PhD thesis, University of Liverpool: Liverpool.

Thrasher, F. M. (1927). *The Gang*, University of Chicago Press: Chicago.

Townsley, M. & Sidebottom, A.. (2010). All offenders are equal, but some are more equal than others: Variations in Journeys to Crime between offenders. *Criminology*, 48 (3), 897-917.

Turner, S. (1969). Delinquency and distance. In Wolfgang, M. E. & Sellin, T. (eds), *Delinquency: Selected Studies*. John Wiley & Sons: New York.

U.S. Census Bureau (2000). All across the USA: Population distribution, 1999, In *Population Profile of the United States: 1999.* Bureau of the Census, U. S. Department of Commerce: Washington, DC., chapter 2.

van Koppen, P. J. & de Keijser, J. W. (1997). Desisting distance decay: on the aggregation of individual crime trips. *Criminology*, 35 (3), 505-516.

References (continued

von Thünen, J. (1826). *The Isolated State in Relation to Agriculture and Political Economy*. English edition, van Suntum, Ulrich. Palgrave Macmillan:Houndsmills, Basingstoke, Hampshire, England, 2009.

Warren, J., Reboussin, R., Hazelwood, R. R., Cummings, A., Gibbs, N., & Trumbetta, S. (1998). Crime scene and distance correlates of serial rape. *Journal of Quantitative Criminology, 14*(1), 35-59.

Weber, A. (1909). *Über den Standort der Industrien* (Theory of Location of Industries).

White, R. C. (1932). The relationship of felonies to environmental factors in Indianapolis. *Social Forces*, 10 (4), 488-509.

Wikipedia (2010a). *Ecological Correlation*. http://en.wikipedia.org/wiki/Ecological_correlation. Accessed March 26, 2012.

Wikipedia (2010b). *Ecological Fallacy*. http://en.wikipedia.org/wiki/Ecological_fallacy. Accessed March 26, 2012.

Wilson, A. G. (1970). *Entropy in Urban and Regional Planning*. Leonard Hill Books: Buckinghamshire.

Zipf, G. K. (1949). *Human Behaviour and the Principle of Least Effort*. Cambridge.

Endnotes

i. If the coordinate system is projected with the distance units in feet, meters or miles, then the distance between two points is the hypotenuse of a right triangle using Euclidean geometry:

$$d_{AB} = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2} \qquad \text{repeat (3.1)}$$

where each location is defined by an X and Y coordinate in feet, meters, or miles. If the coordinate system is spherical with units in latitudes and longitudes, then the distance between two points is the Great Circle distance. All latitudes and longitudes are converted into radians using:

$$Radians\ for\ latitude(\varphi) = \frac{2\pi\varphi}{360} \qquad \text{repeat (3.2)}$$

$$Radians\ for\ longitude(\lambda) = \frac{2\pi\lambda}{360} \qquad \text{repeat (3.3)}$$

Then, the distance between the two points is determined from:

$$d_{AB} = 2Arcsin\{Sin^2\left[\frac{(\varphi_B - \varphi_A)}{2}\right] + Cos\varphi_A Cos\varphi_B Sin^2[\frac{(\lambda_B - \lambda_A)}{2}]^{\frac{1}{2}}\} \qquad \text{repeat (3.4)}$$

with all angles being defined in radians (Snyder, 1987, p. 30, 5-3a).

Attachments

A Note on Alternative Journey-to-crime Models

Ned Levine
Ned Levine & Associates
Houston, TX

There are several alternative journey-to-crime models that have been developed in addition to the *CrimeStat* model. This is a brief note on two of them, the Rossmo model and the Canter model. The citations are listed in the reference section above.

Rossmo Model

Rossmo (1993a; 1995) has adapted location theory, particularly travel behavior modeling, to serial offenders. In a series of papers (Rossmo, 1993a; 1993b; 1995; 1997) he outlined a mathematical approach to identifying the home base location of a serial offender, given the distribution of the incidents. The mathematics represent a formulation of the Brantingham and Brantingham (1981) search area model, discussed above in which the search behavior of an offender is seen as following a distance decay function with decreased activity near the offender's home base. He has produced examples showing how the model can be applied to serial offenders (Rossmo, 1993a; 1993b; 1997).

The model has four steps (what he called *criminal geographic targeting*):

1. First, a rectangular study area is defined that extends beyond the area of the incidents committed by the serial offender. The average distance between points is taken in both the Y and X direction. Half the Y coordinate inter-point distance is added to the maximum Y value and subtracted from the minimum Y value. Half the X coordinate inter-point distance is added to the maximum X value and subtracted from the minimum X value. These are based on projected coordinates; presumably, the directions would have to be adjusted if spherical coordinates were used. The rectangular study defines a grid from which columns and rows can be defined.

2. For each grid cell, the Manhattan distance to each incident location is taken (see Chapter 3 for definition).

3. For each Manhattan distance from a grid cell to an incident location, MD_{ij}, one of two functions is evaluated:

A. If the Manhattan distance, Md_{ij}, is less than a specified buffer zone radius, B, then:

$$P_{ij} = \prod_{j=1}^{T}\{\frac{k(1-\varphi)(B^{g-f})}{(2B-|X_i-X_c|+|Y_i-Y_c|)^g}\} \quad (13.47)$$

where P_{ij} is the resultant of offender interaction for grid cell *i*; with incident *j*, *c* is the incident number, summing to T; $\varphi = 0$; *k* is an empirically determined constant; *g* is an empirically determined exponent; and *f* is an empirically determined exponent.

The Greek letter, Π, is the product sign, indicating that the results for each grid cell-incident distance, Md_{ij}, are *multiplied* together across all incidents, c. This equation reduces to:

$$P_{ij} = \prod_{j=1}^{T}\{\frac{k(1-0)(B^{g-f})}{(2B-|X_i-X_c|+|Y_i-Y_c|)^g}\} \quad (13.48)$$

$$P_{ij} = \prod_{j=1}^{T}\{\frac{k(B^{g-f})}{(2B-|X_i-X_c|+|Y_i-Y_c|)^g}\} \quad (13.49)$$

Within the buffer region, the function is the ratio of a constant, *k*, times the radius of the buffer, B, raised to another constant, *g-f*, divided by the difference between the diameter of the circle, *2B*, and the critical Manhattan distance, Md_{ij}, raised to a constant, g. This is a *non-linear* function that is increasing within the buffer zone.

4. If the Manhattan distance, Md_{ij}, is greater than a specified buffer zone radius, B, then

$$P_{ij} = \prod_{c=1}^{T}\{k\frac{\varphi}{(|X_i-X_c|+|Y_i-Y_c|)^f}\} \quad (13.50)$$

where P_{ij} is the resultant of offender interaction for grid cell, i, and incident location, j; *c* is the incident number, summing to T; $\varphi = 1$; *k* is an empirically determined constant (the same as in Equation 13.47 above); and *f* is an empirically determined exponent (the same as in Equation 13.47 above).

Again, the Greek letter, Π, indicates that the results for each grid cell-incident distance, Md_{ij}, are multiplied together across all incidents, c. This equation reduces to:

$$P_{ij} = \prod_{c=1}^{T}\{k \frac{1}{(|X_i - X_c| + |Y_i - Y_c|)^f}\} \quad (13.51)$$

$$P_{ij} = \prod_{c=1}^{T}\{k \frac{k}{(|X_i - X_c| + |Y_i - Y_c|)^f}\} \quad (13.52)$$

Outside of the buffer region, the function is a constant, k, divided by the distance, Md_{ij}, raised to an exponent, f. It is an inverse distance function and drops off rapidly with distance.

4. Finally, for each grid cell, i, the functions evaluated in step 3 above are summed over all incidents.

For both the 'within buffer zone' (near to home base) and 'outside buffer zone' (far from home base) functions, the coefficient, k, and exponents, f and g, are empirically determined. Though he does not discuss how these are calculated, they are presumably estimated from a sample of known offender locations where the distance to each incident is known (e.g., arrest records).

The result is a surface model indicating a likelihood of the offender residing at that location. He describes it as a probability surface, but it is actually a *density* surface. Since the probability of interaction between any one grid cell, i, and any one incident, j, cannot be greater than 1, the surface actually indicates the product of individual likelihoods that the offender uses that location as the home base. To be an actual probability function, it would have to be re-scaled so that the sum of the grid cells was equal to 1.

The second function - 'outside the buffer zone' (Equation 13.52) is a classic gravity function, similar to Equation 13.5 except there is no attraction definition. It is the distance decay part of the gravity function. The first function, Equation 13.49, is an increasing curvilinear function designed to model the area of decreased activity near the offender's home base.

Strengths and Weaknesses of the Rossmo model

The Rossmo model has both strengths and weaknesses. First, the model has some theoretical basis utilizing the Brantingham and Brantingham (1981) framework for an offender search area as well as the mathematics of the gravity model and distinguishes two types of travel behavior - near to home and farther from home. Second, the model does represent a systematic

approach towards identifying a likely home base location for an offender. By evaluating each grid cell in the study area, an independent estimate of the likelihood is obtained, which can then be integrated into a continuous surface with an interpolation graphics routine.

There are problems with the particular formulation, however. First, the exclusive use of Manhattan distances is questionable. Unless the study area has a street network that follows a uniform grid, measuring distances horizontally and vertically can lead to overestimation of travel distances; further, the more the layout differs from a north-south and east-west orientation, the greater the distortion. Since many urban areas do not have a uniform grid street layout, the method will necessarily lead to overestimation of travel distances in places where there are diagonal or irregular streets.[5]

Second, the use of a product term, Π, complicates the mathematics. That is, the technique evaluates the distance from a particular grid cell, i, to a particular incident location, j. It then *multiplies* this result by all other results. Since the P values are actually densities, which can be greater than 1.0, the process, if strictly applied, would be a compounding of probabilities with overestimation of the likelihood for grid cells close to incident locations and underestimation of the likelihood for grid cells farther away. In the description of the method, however, Rossmo actually mentions summing the terms. Thus, the substitution of a summation sign, Σ, for the product sign would help the mathematics.

A third problem is in the distance decay function (Equation 13.52). The use of an inverse distance term has problems as the distance between the grid cell location, i, and the incident location, j, decreases. For some types of crimes, there will be little or no buffer zone around the offender's home base (e.g., rapes by acquaintances). Consequently, the buffer zone radius, B, would approach 0. However, this would cause the model to become unstable since the inverse distance term will approach infinity.

Fourth, the use of a mathematical function to describe the distance decay, while easy to define, probably oversimplifies actual travel behavior. A mathematical function to describe distance decay is an approximation to actual travel behavior. It assumes that travel is equally likely in each direction, that travel distance is uniformly easy (or difficult) in each direction, and that, similarly, opportunities are uniformly distributed. For most urban areas, these conditions would not be true. Few cities form a perfect grid (there are exception, such as Salt Lake City), though most cities have sections that are grided. Both physical geography limit travel in certain directions as does the historical street structure, which is often derived from earlier communities.

---

5 It should also be pointed out that the use of direct distances will underestimate travel distances particularly if the street network follows a grid.

A mathematical function does not consider this structure, but rather assumes that the 'impedance' in all directions is uniform.

This latter criticism, of course, would be true for all mathematical formulations of travel distance. There are corrections that can be made to adjust for this. For example, in the urban travel demand type model, trip distribution between locations is estimated by a gravity model, but then the distributed trips are constrained by, first, the total number of trips in the region (estimated separately), second, by mode of travel (bus v. single driver v. drivers plus passengers v. walk, etc.), and, third, by the route structure upon which the trips are eventually assigned (Krueckeberg & Silvers, 1974; Stopher & Meyburg, 1975; Field & MacGregor, 1987). Calibration at all stages against known data sets ensures that the coefficients and exponents fit 'real world' data as closely as possible. It would take these types of modifications to make the travel distribution type of model postulated by Rossmo and others be a more realistic representation.

Fifth, the model imposes mathematical rigidity on the data. While there are two different functions that could vary from place to place, the particular type of distance decay function might also vary. Specifying a strict form for the two equations limits the flexibility of applying the model to different types of crime or to places where the distance decay does not follow the form specified by Rossmo.

A sixth problem is that opportunities for committing crimes - the attractiveness of locations, are never measured. That is, there is no enumeration of the opportunities that would exist for an offender nor is there an attempt to measure the strength of this attraction. Instead, the search area is inferred strictly from the distribution of incidents. Because the distribution of offender opportunities would be expected to vary from place to place, the model would need to be re-calibrated at each location. In this sense, both the Canter model (described below) and my Journey-to-crime model (described in the chapter) also share this weakness. It is understandable in that victim/target opportunities are difficult to define *a priori* since they can be interpreted differently by individuals. Nevertheless, a more complete theory of Journey-to-crime behavior would have to incorporate some measure of opportunities, a point that both Brantingham and Brantingham (1981) and Rengert (1981) have made.

Finally, the 'buffer zone' concept is but one interpretation of the tendency of many crimes not to be committed close to the home location. There are other interpretations that are applicable. For example, the distribution of crime opportunities is often not close to the home location, either. Many crimes occur in commercial areas. In most American and British cities, residential areas are not located in commercial areas. Thus, there will usually be a distance between a residential location and a nearby crime opportunity. This does not imply anything about a 'safety zone' for the offender but, instead, may illustrate the distribution of crime

opportunities. If we could map the travel distance of, say, shopping trips, we would probably find a similar distribution to that seen in most of Journey-to-crime studies (and illustrated below).

The concept of a 'buffer zone' is a hypothesis, not a certainty. The language of it is so appealing that many people believe it to be true. But, to demonstrate the existence of a 'buffer zone' would require interviewing offenders (or offenders who have been arrested) and demonstrating that they did not commit crimes near their residence even though there were opportunities (i.e., they valued safety over opportunity). Otherwise, one cannot distinguish between the 'buffer zone' hypothesis and the distribution of available opportunities. They may very well be the same thing.

Canter Model

Canter's group in Liverpool and, more recently, Huddersfield (Canter & Tagg, 1975; Canter & Larkin, 1993; Canter & Snook, 1999; Canter, Coffey, Huntley, & Missen, 2000) have modified the distance decay function for Journey-to-crime trips by using a negative exponential term, instead of the inverse distance. Their *Dragnet* program uses the negative exponential function:

$$Y = \alpha e^{\frac{-\beta d_{ij}}{P}} \tag{13.53}$$

where Y is the likelihood of an offender traveling a certain distance to commit a crime,, d_{ij} is the distance (from a home base location to an incident site), α is an arbitrary constant, β is the coefficient of the distance, P is a normalization constant, and e is the base of the natural logarithm. The model is similar to Equation 13.52 except, like Rossmo, it does not include the attractiveness of the location.

Using the logic that most crimes are committed near the offender's home base, Canter, Coffey, Huntley, and Missen (2000) use a five step process to estimate a search strategy:

2. The study area is defined by a rectangle that is 20% larger in area than that defined by the minimum and maximum X/Y points. A grid cell structure of 13, 300 cells is imposed over the rectangle. Each grid cell is a reference location, i.

3. A decay coefficient is selected. In Equation 13.53, this would be the coefficient, β, for the distance term, d_{ij}, both of which are exponents of e. Unlike Rossmo, Canter uses a series of decay coefficients from 0.1 to 10 to estimate the sensitivity

of the model. The equation indicates the likelihood with which any location is likely to be the home base of the offender based on one incident.

4. Because different offenders have different search areas, the measured distances for each cell are divided by a normalization coefficient, *P*, that adjusts all offenses to a comparable range. Canter uses two different types of normalization function: 1) Mean inter-point distance between all offenses (across a group of offenders); and 2) The QRange, which is an index that takes into account asymmetry in the orientation of the incidents.

5. For each reference cell, i, the distance between each grid cell and each incident location is evaluated with the function and the standardized likelihoods are summed to yield an estimate of location potential.

6. A *search cost* index is defined by the proportion of the study area that has to be searched to find the offender. By calibrating the model against known cases, an estimate of search efficiency is obtained.

Additional modifications can be added to the functions to make them more flexible (Canter, Coffey, Huntley & Missen, 2000). For example, 'steps' are distances near to home where offenders are not likely to act while 'plateaus' are constant distances near to home where there is the highest likelihood of acting. For example, Canter and Larkin (1993) found an area around serial offenders' homes of about 0.61 mile in radius within which they were less likely to commit crimes.

Canter and Snook (1999) provide estimates of the search cost (or efficiency) associated with various distance coefficients. For example, with the known home base locations of 32 burglars, a β of 1.0 yielded a mean search cost of 18.06%; that is, on average, only 18.06% of the study area had to be searched to find the location of 32 burglars in the calibration sample. Clearly, for some of them, a larger area had to be searched while for others a smaller area; the average was 18.06%. Conversely, the mean search cost index for 24 rapists was 21.10% and for 37 murderers 28.28%. They further explored the marginal increase in locating offenders by increasing the percentage of the study area that had to be searched. They found for their three samples (burglary, rape, homicide) that more than half the offenders could be located within 15% of the area searched.

The Canter model is different from the Rossmo model is that it suggests a search strategy by the police for a serial offender rather than a particular location. The strength of it is to indicate how narrow an area the police should concentrate on in order to optimize finding an offender. Clearly, in most cases, only a small area needs be searched.

Strengths and Weaknesses of the Canter model

The model has both strengths and weaknesses. First, the model provides a search strategy for law enforcement. By examining which type of function best fits a certain type of crime, police can target their search efforts more efficiently. The model is relatively easy to implement and is practical. Second, the mathematical formulation is stable. Unlike the inverse distance function in the Rossmo model, Equation 13.49 will not have problems associated with distances that are close to 0. Further, the model does provide a search strategy for identifying an offender. It is a useful tool for law enforcement officers, particularly as they frame a search for a serial offender.

There are also weaknesses to the model. First, it lacks a theoretical basis. Canter's research has provided a great deal in terms of understanding the activity spaces of serial offenders (Canter & Larkin, 1993; Canter & Gregory, 1994; Canter, 1994; Hodge & Canter, 1998). However, the empirical model used is strictly pragmatic. Second, mathematically, it imposes the negative exponential function without considering other distance decay models. In the *Dragnet* program, the decay function is a string of 20 numbers so that, in theory, any function can be explored. However, the default is a negative exponential. The negative exponential has been used in many travel behavior studies (Foot, 1981; Bossard, 1993), but it does not always produce the best fit. While the model can be adapted to be more flexible by different exponents and including steps and plateaus, for example, it is still tied to the negative exponential form. Thus, the model might work in some locations, but may fail in others; a user can't easily adjust the model to make it fit new data.

Third, the coefficient of the negative exponential, α, is defined arbitrarily. In the *Dragnet* program, it is usually set as 0.5. While this ensures that the result never exceeds 1.0 for any one incident, there is a limit on the location potential summation since the total potential is a function of the number of incidents (i.e., it will be higher for more incidents). It would have been better if the coefficient were calibrated against a known sample.

Fourth, and finally, also similar to the Rossmo model (and to my journey-to-crime model), criminal opportunities (or attractions) are never measured, but are inferred from the pattern of crime incidents. As a pragmatic tool for informing a police search, one could argue that this is not important. However, in a different location, the distance coefficient is liable to differ as is the search cost index. It would need to be re-calibrated each time.

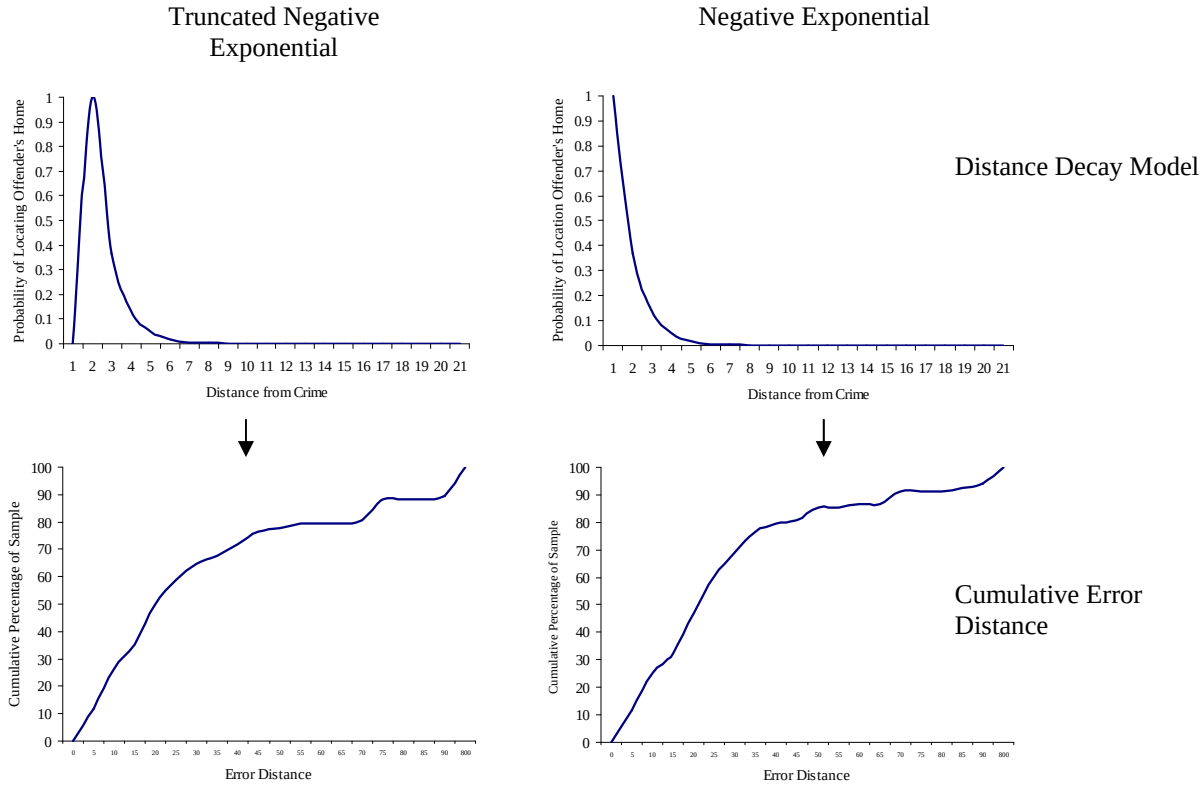
Nevertheless, the Canter model is a useful tool for police department and can help shape a search strategy. It is different from the other location models in that it is not focused so much on the best prediction for a location of an offender (though the summation discussed above in step 4 can yield that) as it does in defining where the search should be optimized.

Using *CrimeStat* for Geographic Profiling

Brent Snook, Memorial University of Newfoundland,
Paul J. Taylor, University of Liverpool, Liverpool
Craig Bennell, Carleton University, Ottawa

A challenge for researchers providing investigative support is to use information about crime locations to prioritize geographic areas according to how likely they are to contain the offender's residence. One prescient solution to this problem uses *probability distance functions* to assign a likelihood value to the activity space around each crime location. A research goal is to identify the function that assigns the highest likelihood to the offender's actual residence, since this should prove more efficient in future investigations.

CrimeStat was used to test of the effectiveness of two functions for a sample of 68 German serial murder cases, using a measure known as *error distance*. The top figures below illustrate the two functions used and the bottom figures portray the corresponding effectiveness of the functions by plotting the percentage of the sample 'located' by error distance. A steeper effectiveness curve indicates that home locations were closer to the point of highest probability and that, consequently, the probability distance function was more efficient. In this particular test, no difference was found between the two functions in their ability to classify geographic areas.



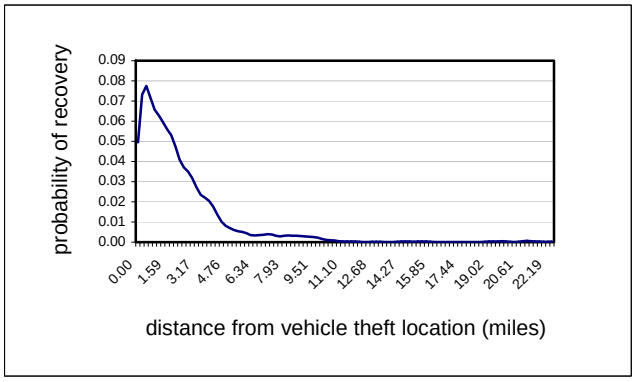
Original Article: Taylor, P.J., Bennell, C., & Snook B. (2002) *Problems of Classification in Investigative Psychology.* Proceedings of the 8th Conference of the International Federation of Classification Societies, Krakow, Poland

Using Journey-to-crime Routine for Journey-after-crime Analysis

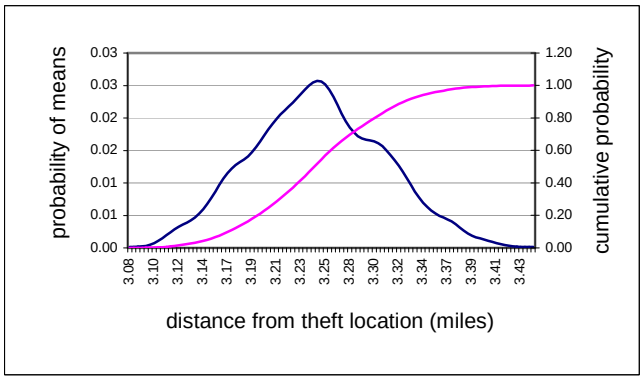
Yongmei Lu
Department of Geography
Southwest Texas State University
San Marcos, TX

The study of vehicle theft recovery locations can fill a gap in the knowledge about criminal travel patterns. Although the journey-to-crime routine of *CrimeStat* was designed to analyze the distance between offense location and offender's residential location, it can be used to describe the distance between vehicle theft location and the corresponding recovery location.

There were more than 3000 vehicle thefts in the City of Buffalo in 1998. Matching the offenses with vehicle recoveries in the same year, 1600 location pairs were identified for a journey-after-vehicle-theft analysis. To evaluate the randomness of the distances, 1000 groups of simulations were conducted. Every group contains 1600 simulated trips of journey-after-vehicle-theft. The results indicate that 1) short distances dominate journey-after-vehicle-theft, and 2) the observed trips are significantly shorter than the random trips given the distribution of possible vehicle theft and recovery locations.



Probability of recovering a stolen vehicle by distance from vehicle theft location



Distribution of mean distances of simulated vehicle theft-recovery location pairs.

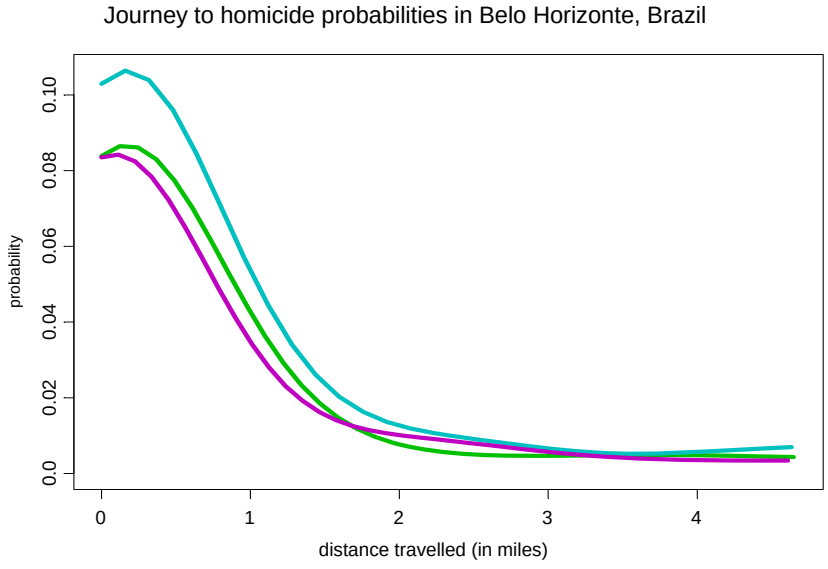
Using Journey-to-crime Analysis for Different Age Groups of Offenders

Renato Assunção, Cláudio Beato, Bráulio Silva
CRISP, Universidade Federal de Minas Gerais , Brazil

CrimeStat offers a method for analysing the distance between the crime scene and the residence of the offender within the spatial modeling module. We analysed homicide incidents in Belo Horizonte, a Brazilian city of 2 million inhabitants, for the period January 1996 – December 2000. We used 496 homicide cases for which the police identified an offender who was living in Belo Horizonte, and for which both the crime location and offender residence could be identified. The cases were divided into three groups according to the offender's age: 1) 14 to 24 (N=201); 2) 25 to 34 (N=176); and 3) 35 or older (N=119). The journey-to-crime calibration routine was used to produce a probability curve *P(d)* that gives the approximate chance of an offender travelling approximately distance *d* to commit the crime.

We used the normal kernel, a fixed bandwidth of 1000 meters, 100 output bins, and the probability (or proportion of all points) option, rather than densities. This is to allow comparisons between the three age groups since they have different number of homicides. We tested for each age group separately and directed the output to a text file to analyse the three groups simultaneously.

The green, blue, and purple curves are associated with the 14-24, 25-34, 35+ year olds respectively. There are more similarities than differences between the groups. Most homicides are committed near to the residence of the offenders with between 60% t o 70% closer than one mile from their home. However, the curve does not vanish totally even for large distances because there are around 15% of offenders, of any age group, travelling longer than 3 miles to commit the crime. The oldest offenders travel longer distances, on average, followed by the youngest group, with the 25-34 year olds travelling the shortest distances.

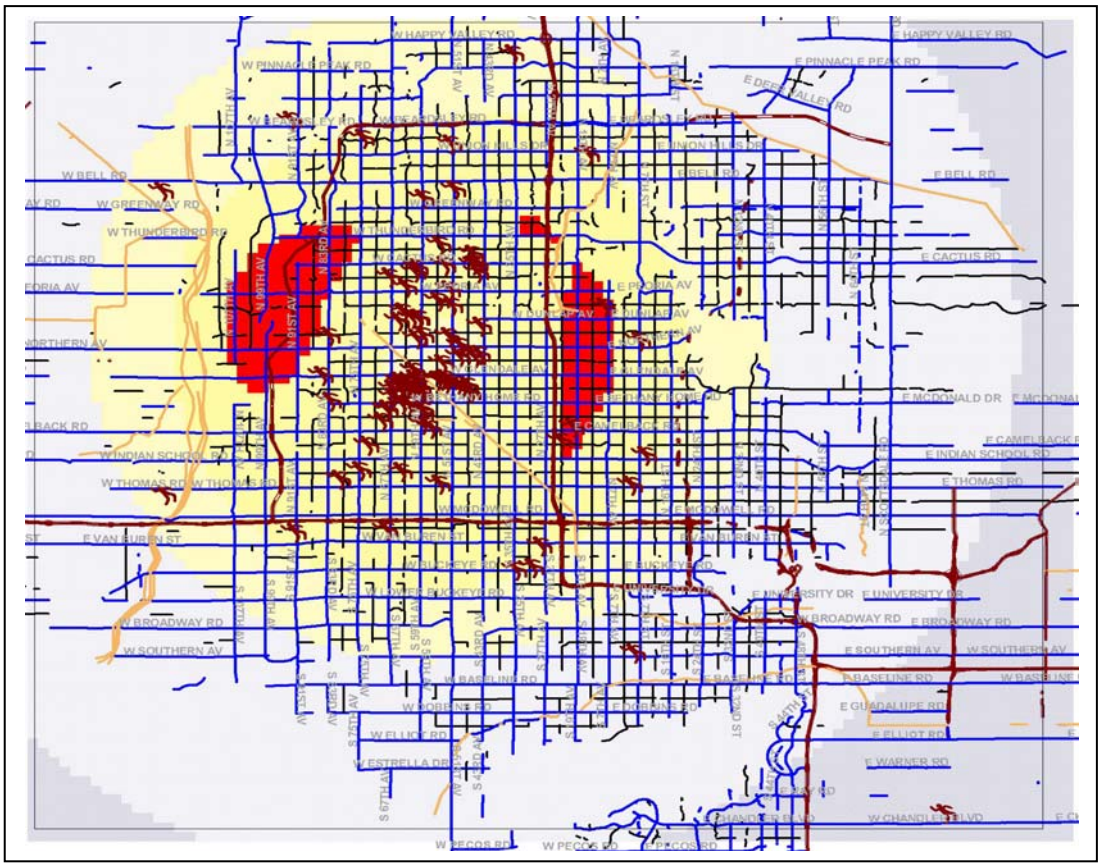


Catching the Bad Guy

Bryan Hill
Glendale Police Department
Glendale, AZ

The City of Glendale, Arizona recently had a string of auto thefts committed by the same individual. The map shows known auto theft suspects and their home address. The red area in the map shows the most probable home address. Prior to the analysis, the Phoenix Police Department's Crime Analysis Unit was able to calibrate the *CrimeStat* Jtc routine with known offender robbery suspect data.

Monthly citation data was used to search for anyone that lived within the area identified by the routine who also drove a red Saturn. A suspect with a felony warrant was identified and proved to be also the suspect in a series of armed robberies and a homicide that occurred in the Phoenix and Glendale jurisdictions. When he was arrested for the felony warrant at his home, evidence of the robberies and homicide were found.



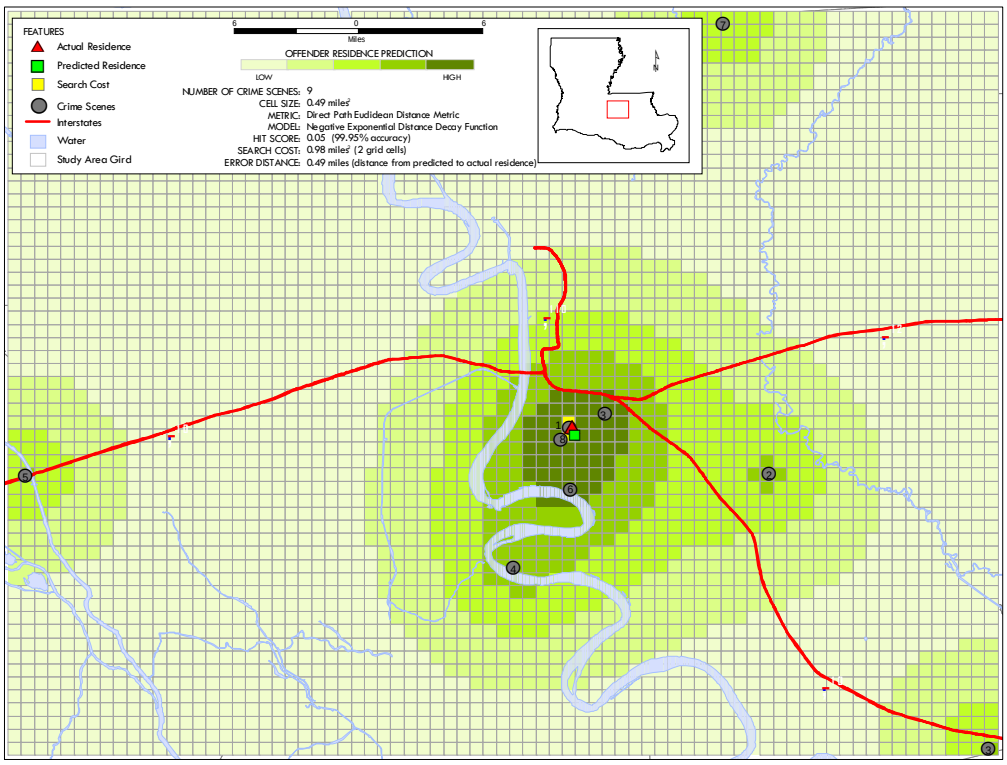
Constructing Geographic Profiles
Using the *CrimeStat* Journey-to-crime Routine

Josh Kent,
Michael Leitner,
Louisiana State University
Baton Rouge, LA

The map below shows a geographic profile constructed from nine crime sites associated with a Baton Rouge serial killer, Sean Vincent Gillis, who was apprehended on April 29, 2004 at his residence in Baton Rouge. Eight of the nine are body dump sites and the ninth is a point of fatal encounter. All crime sites were located in the City of Baton Rouge and surrounding parishes. Gillis's hunting style can best be described as that of a typical 'localized marauder'.

The Journey-to-crime routine, implemented in *CrimeStat* , was applied to simulate the travel characteristics of Gillis to and from the known crime sites. Gillis's travel behavior was calibrated with different mathematical functions that were derived from the known travel patterns of 301 homicide cases in Baton Rouge.

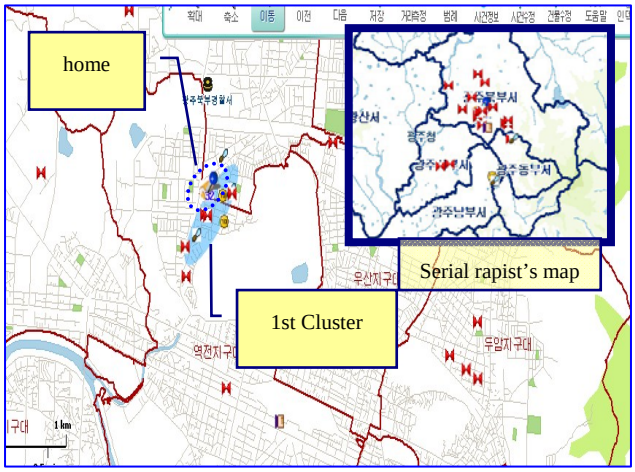
The profile was estimated using Euclidean distance and the negative exponential distance decay function. It predicts the actual residence of Gillis extremely accurately. The straight-line error distance between the predicted and the actual residence is only 0.49 miles. The proportion of the entire study area that must be searched in order to successfully identify the serial offender's residence is 0.05% (approximately 0.98 square miles out of a 2094.75 square miles study area).



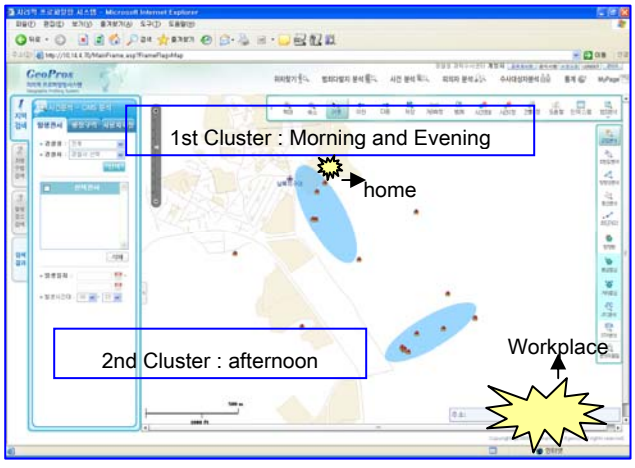
Predicting Serial Offender Residence by Cluster in Korea

Kang Eun Kyoung
Scientific Investigation Center, Korea National Police Agency

Since April, 2009, the Korea National Police Agency have been operating a Geographic Profiling system called GeoPros. GeoPros automatically links three information systems: information the police have collected, CrimeStat, and electronic mapping. Police are able to select areas that need crime prevention and decide which need CCTV. Profilers also use this system to predict the residence of serial offenders using CrimeStat routines linked with GeoPros. Specifically, the Nnh and Jtc routines were useful in predicting offender residence locations. We analyzed multiple serial offenders' data and found that offenders typically have 2 or 3 clusters related to their activities (home, workplace, and possibly evening entertainment).



For example, one offender committed 32 crimes including rape, robbery, and theft. With the Nnh, we found 3 clusters. The first was associated with the criminal's residence; the second was associated with his father's residence, and the third with his lover's house. The criminal's residence was only 10 meters away from the first cluster.



Another case involved unsolved arson cases over ten years (101 offences). There were two clusters that emerged. One consisted of activities in the morning or the evening while the second cluster consisted of activities in the afternoon. We hypothesized that the first cluster was related to the offender's home while the second with the offender's workplace. It turned out that the criminal's residence was only 60 meters away from the first cluster while his workplace was near the second cluster. We gave the detectives involved the information and within two weeks were able to catch the criminal.

One important finding is that offender's residence is typically not inside of the cluster areas, but nearby, which may relate to a buffer zone. In addition to these cases, we have predicted the residence of many cases of serial rape and arson by cluster analysis.

Chapter 14:[1]

Bayesian Journey-to-Crime Modeling

Ned Levine
Ned Levine & Associates
Houston, TX

Richard Block
Loyola University
Chicago, IL

[1] The authors would like to thank Ms. Haiyan Teng, Dr. Wim Bernasco, Dr. Michael Leitner, Dr. Josh Kent, Dr. Craig Bennell, Dr. Brent Snook, Dr. Paul Taylor, and Ms. Patsy Lee for extensively testing the Bayesian Journey-to-crime module.

Table of Contents

Table of Contents (continued)

Chapter 14:

Bayesian Journey-to-crime Modeling

The Bayesian Journey-to-crime module (Bayesian Jtc) includes a set of tools for estimating the likely residence location of a serial offender. It is an extension of the Journey-to-crime routine (Jtc) that uses a travel distance function to make an estimate about the likely residence location of a serial offender. The Bayesian Jtc routine adds information about specific origins of offenders who committed crimes in the same locations to the Jtc to update the estimate. Before proceeding with this chapter, users should be thoroughly familiar with the material on Jtc modeling discussed in Chapter 13.

First, the theory behind the Bayesian Jtc routine will be described. While this material is not essential for running the routine, it does provide the background behind the routine. Users who want to go immediately into the routine should skip to the data section on p. 14.10.

Second, data requirements will be discussed. Third, the routine will be illustrated with data from Baltimore County and from Chicago. Fourth, the use of probability filters as extensions will be illustrated. Fifth, and finally, some guidelines are provided for analysts.

Bayesian Probability

Bayes Theorem is a formulation that relates the conditional and marginal probability distributions of random variables. The *marginal probability* distribution is a probability independent of any other conditions. Hence, P(A) and P(B) is the marginal probability (or just plain probability) of A and B respectively.

The *conditional probability* is the probability of an event given that some other event has occurred. It is written in the form of P(A|B) (i.e., event A given that event B has occurred). In probability theory, it is defined as:

$$P(A|B) = \frac{P(A \text{ and } B)}{P(B)} \qquad (14.1)$$

Conditional probabilities can be best be seen in contingency tables. Table 14.1 below shows a possible sequence of counts for two variables (e.g., taking a sample of persons and counting their gender - male = 1 v. female = 0, and their age - older than 30 = 1 v. 30 or younger = 0). The probabilities can be obtained just by counting:

P(A) = 30/50 = 0.6
P(B) = 35/50 = 0.7
P(A and B) = 25/50 = 0.5
P(A or B) = (30+35-25)/50 = 0.8
P(A|B) = 25/35 = 0.71
P(B|A) = 25/30 = 0.83

However, if four of these six calculations are known, Bayes Theorem can be used to solve for the other two. Two logical terms in probability are the 'and' condition and the 'or' condition. Usually, the symbol ∩ is used for 'and' ∪ is used for 'or', but writing it in words might make it easier to understand.

Table 14.1:
Example of Determining Probabilities by Counting

	A has NOT occurred	A has occurred	**TOTAL**
B has NOT occurred	10	5	**15**
B has occurred	10	25	**35**
TOTAL	**20**	**30**	**50**

The following two theorems define these.

1. The probability that ***either*** A ***or*** B will occur is:

$$P(A\ or\ B) = P(A) + P(B) - P(A\ and\ B) \tag{14.2}$$

2. The probability that ***both*** A ***and*** B will occur is:

$$P(A\ and\ B) = P(A) * P(B|A) = P(B) * P(A|B) \tag{14.3}$$

Bayes Theorem relates the two equivalents of the '***and***' condition together:

$$P(B) * P(A|B) = P(A) * P(B|A) \tag{14.4}$$

$$P(A|B) = \frac{P(A)*P(B|A)}{P(B)} \quad (14.5)$$

The theorem is sometimes called the 'inverse probability' in that it can invert two conditional probabilities:

$$P(B|A) = \frac{P(B)*P(A|B)}{P(A)} \quad (14.6)$$

By plugging in the values from the example in Table 14.1, the reader can verify that Bayes Theorem produces the correct results, for example:

$$P(B|A) = \frac{0.7*0.71}{0.6} = 0.83 \quad (14.7)$$

Bayesian Inference

In the statistical interpretation of Bayes Theorem, the probabilities are estimates of a random variable. Let θ be a parameter of interest and let X be some data. Thus, Bayes Theorem can be expressed as:

$$P(\Theta|X) = \frac{P(X|\Theta)*P(\Theta)}{P(X)} \quad (14.8)$$

Interpreting this equation, $P(\theta|X)$ is the probability of θ given the data, X, and is called the *posterior probability* (or posterior distribution). $P(\theta)$ is the probability that θ has a certain distribution and is often called the *prior probability*. $P(X|\theta)$ is the probability that the data would be obtained given that θ is true and is often called the *likelihood function* (i.e., it is the likelihood that the data will be obtained given the distribution of θ). Finally, $P(X)$ is the marginal probability of the data, the probability of obtaining the data under all possible scenarios; essentially, it is the data.

The equation can be rephrased in words:

$$\text{Posterior probability that } \theta \text{ is true given the data, X} = \frac{\text{Likelihood of obtaining the data given } \theta \text{ is true} * \text{Prior probability of } \theta}{\text{Marginal probability of X}} \quad (14.9)$$

In other words, this formulation allows an estimate of the probability of a particular parameter, θ, to be updated given new information. Since θ is the prior probability of an event,

given some new data, X, Bayes Theorem can be used to update the estimate of θ. The prior probability of θ can come from prior studies, an assumption of no difference between any of the conditions affecting θ, or an assumed mathematical distribution. The likelihood function can also come from empirical studies or an assumed mathematical function. Irrespective of how these are interpreted, the result is an estimate of the parameter, θ, given the evidence, X.

A point that is often made is that the prior probability of obtaining the data (the denominator of the above equation) is not known or cannot easily be evaluated. The data are what was obtained from some data gathering exercise (either experimental or from observations). Thus, it is not easy to estimate it. Consequently, often the numerator only is used for estimate the posterior probability since:

$$P(\Theta|X) \propto P(X|\Theta) * P(\Theta) \qquad (14.10)$$

where $\propto$ means 'proportional to'. In some statistical methods (e.g., the Markov Chain Monte Carlo simulation, or MCMC, discussed in Chapters 17, 18 &19), the parameter of interest is estimated by thousands of random simulations using approximations to $P(X|\theta)$ and $P(\theta)$ respectively.

The key point is that estimates of parameters can be systematically *updated* by additional information. The formula requires that a prior probability for the estimate be given with new information being added which is *conditional* on the prior estimate, meaning that it takes into account information from the prior. Bayesian approaches are increasingly being used to provide estimates for complex calculations that previously were intractable (Denison, Holmes, Mallilck, & Smith, 2002; Lee, 2004; Gelman, Carlin, Stern, & Rubin, 2004). Our regression module includes the use of the MCMC algorithm to estimate complex equations.

Application of Bayesian Inference to Journey-to-crime Analysis

Bayes Theorem can be applied to the Journey-to-crime methodology. In the Journey-to-crime (Jtc) method, an estimate is made about where a serial offender is living. The Jtc method produces probability estimates based on an assumed travel distance function (or, in more refined uses of the method, travel time). That is, it is assumed that an offender follows a typical travel distance/time function. This function can be estimated from prior studies (Canter & Gregory, 1994; Canter, 2003) or from creating a sample of known offenders - a calibration sample (see Chapter 13; Levine, 2000) or from assuming that every offender follows a particular mathematical function (Rossmo, 1995; 2000). Essentially, it is a prior probability for a particular location, $P(\theta)$. That is, it is a guess about where the offender lives on the assumption that the offender of interest is following an existing travel distance model.

However, additional information from a sample of known offenders where both the crime location and the residence location are known can be added. This information would be obtained from arrest records, each of which will have a crime location defined (a 'destination') and a residence location (an 'origin'). If these locations are then assigned to a set of zones, a matrix that relates the origin zones to the destination zones can be created (Figure 14.1). This is called an ***origin-destination*** matrix (also known as a *trip distribution* or *O-D* matrix, for short).

In this figure, the numbers indicate crimes committed in each destination zone which originated from each origin zone (i.e., where the offender lived). For example, taking the first row in Figure 14.1, there were 37 crimes that were committed in zone 1 and in which the offender also lived in zone 1; there were 15 crimes committed in zone 2 in which the offender lived in zone 1; however, there were only 7 crimes committed in zone 1 in which the offender lived in zone 2; and so forth.

Note two things about the matrix. First, the number of origin zones can be (and usually is) greater than the number of destination zones because crimes can originate outside the study area. Second, the marginal totals have to be **equal**. That is, the number of crimes committed in all destination zones must equal the number of crimes originating in all origin zones.

This information can be treated as the likelihood estimate for the Journey-to-crime framework. That is, if a certain distribution of incidents committed by a particular serial offender is known, then this matrix can be used to estimate the likely origin zones from which offenders came, independent of any assumption about travel distance. In other words, this matrix is equivalent to the likelihood function in Equation 14.8, which is repeated below:

$$P(\Theta|X) = \frac{P(X|\Theta)*P(\Theta)}{P(X)} \qquad \text{repeat (14.8)}$$

The estimate of the likely origin location of a serial offender can be improved by updating the Jtc estimate, $P(\theta)$, with information from an empirically-derived likelihood estimate, $P(X|\theta)$.

Figure 14.2 illustrates the process. Suppose a serial offender committed crimes in three zones. These are shown in terms of grid cell zones. In reality, most zones are not grid cells but are irregular. However, illustrating with grid cells makes the process more understandable. Using an O-D matrix based on those cells, only the destination zones corresponding to those cells are selected (Figure 14.3). This process is repeated for all serial offenders in the calibration file. Destination zones that are repeated by different serial offenders are counted multiple times, once for each occurrence. This results in marginal totals that correspond to frequencies for those serial offenders who committed crimes in the selected zones. The marginal totals are then

Figure 14.1:

Crime Origin-Destination Matrix

Crime destination zone

Crime origin zone	1	2	3	4	5	.	.	.	.	N	Σ
1	**37**	15	21	4	3					12	346
2	7	**53**	14	0	4					15	1050
3	12	9	**81**	7	6					33	711
4	4	10	6	**12**	1					0	84
5	8	7	28	2	**24**					14	178
.											
.											
.											
M	12	5	43	3	10					**92**	1466
Σ	153	276	1245	99	110					812	43,240

converted to probabilities. In other words, the distribution of crimes is *conditioned* on the locations that correspond to where the serial offender of interest committed his or her crimes. It is a conditional probability.

But, what about the denominator of the Bayesian formula, P(X)? Essentially, it is the spatial distribution of all crimes irrespective of which particular model or scenario we are exploring. In practice, it is very difficult, if not impossible, to estimate the probability of obtaining the data under all circumstances. Therefore, only the numerator in equation 14.8 is estimated and the final probabilities are re-scaled so that they sum to 1.0 over the study area:

$$P(\Theta|X) \propto \mathrm{k} * P(X|\Theta) * P(\Theta) \qquad (14.11)$$

where *k* is a scaling constant.

We are going to change the symbols at this point so the *Jtc* represents the distance-based Journey-to-crime estimate, *O* represents an estimate based on an origin-destination matrix, and *O*|*Jtc* represents the particular origins associated with crimes committed in the same zones as that identified in the *Jtc* estimate. Therefore, there are three different probability estimates of where an offender lives:

1. A probability estimate of the residence location of a single offender based on the location of the incidents that this person committed and an assumed travel distance function, *P(Jtc)*;

2. A probability estimate of the residence location of a single offender based on a general distribution of all offenders, irrespective of any particular destinations for incidents, *P(O)*. Essentially, this is the distribution of origins irrespective of the destinations; and

3. A probability estimate of the residence location of a single offender based on the distribution of offenders given the distribution of incidents committed by other offenders who committed crimes in the same location, *P(O|Jtc)*.

Therefore, Bayes Theorem can be used to create an estimate that combines information both from a travel distance function and an origin-destination matrix in which the posterior probability of the Journey-to-crime location taking into account the origin-destination matrix is proportional to the product of the prior probability of the Journey-to-crime function, *P(Jtc)*, and the conditional probability of the origins for other offenders who committed crimes in the same locations, *P(O)*. This will be called the ***product*** probability.

Figure 14.2:

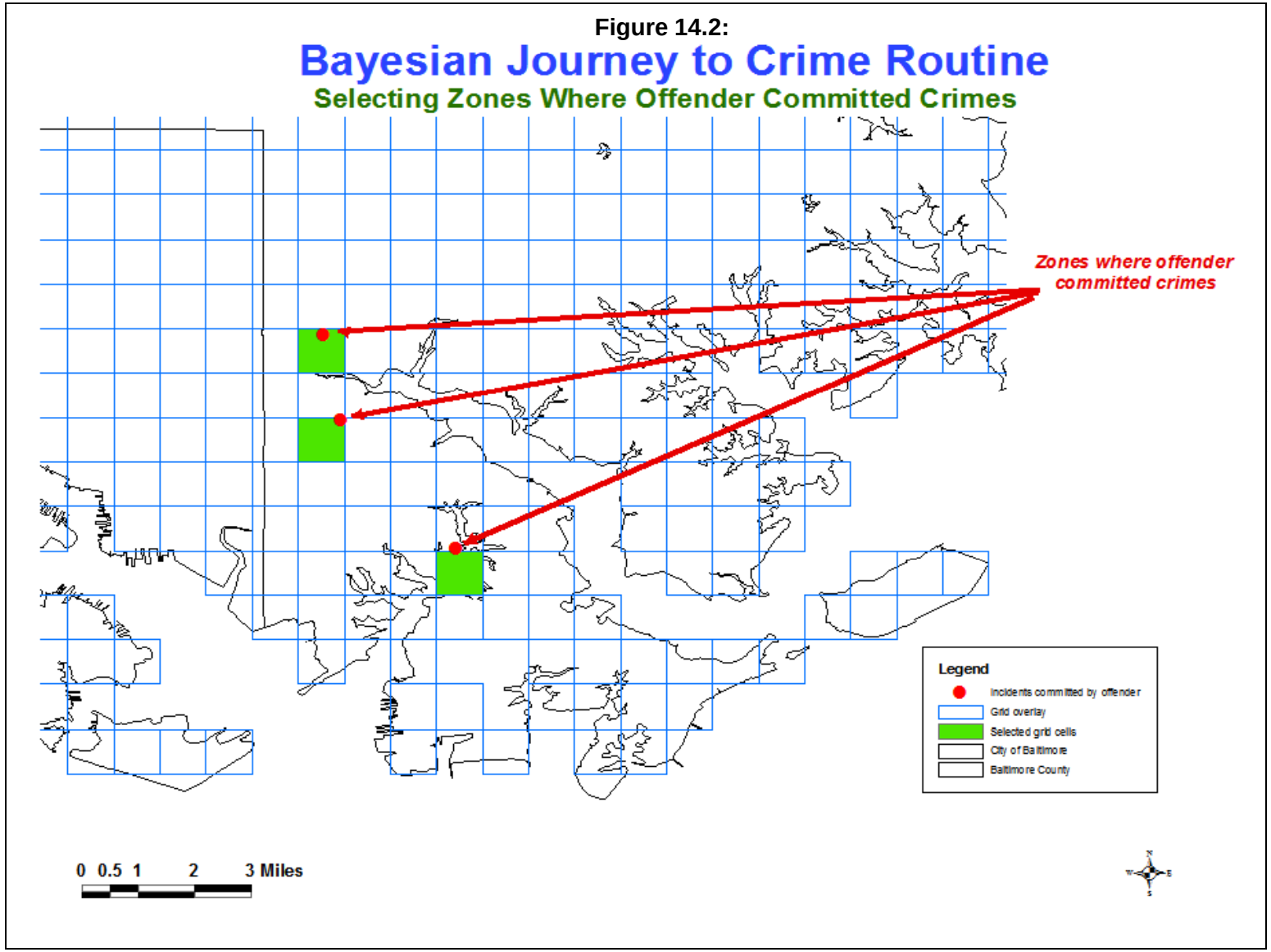
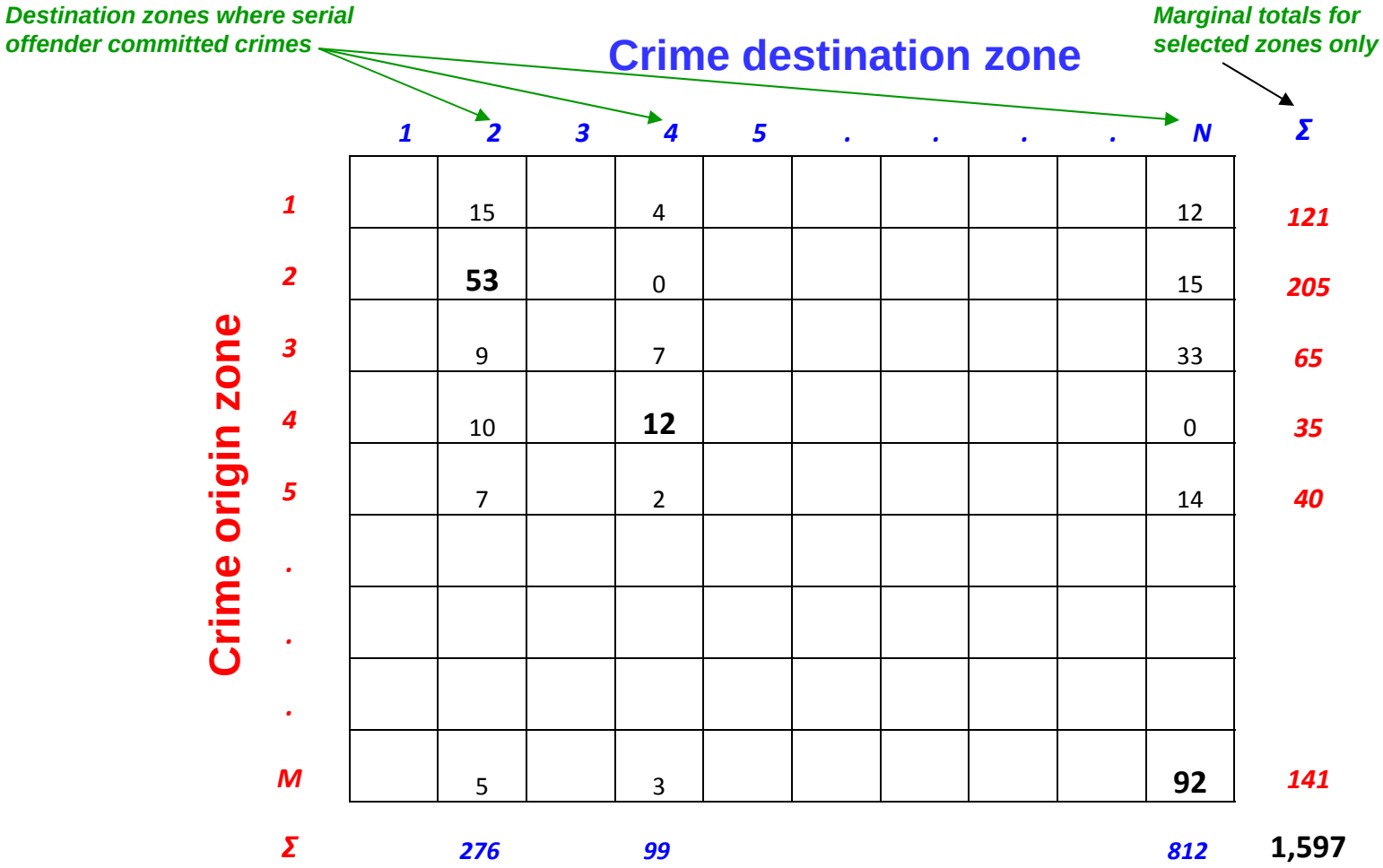

Bayesian Journey to Crime Routine
Selecting Zones Where Offender Committed Crimes
Zones where offender committed crimes
Legend
Incidents committed by offender
Grid overlay
Selected grid cells
City of Baltimore
Baltimore County
0 0.5 1 2 3 Miles

Figure 14.3:

Conditional Origin-Destination Matrix



As mentioned above, it is very difficult to determine the probability of obtaining the data under any circumstance, P(O). Consequently, the Bayesian estimate is usually calculated only with respect to the numerator, the product of the prior probability and the likelihood function, and the result re-scaled so that the probabilities over the study area sum to 1.0.

A very rough approximation to the full Bayesian probability can be obtained by taking the product probability and dividing it by the general probability: It relates the product term (the numerator) to the general distribution of crimes. This will produce a relative risk measure, which is called ***Bayesian Risk***:

$$P(Jtc|O) = \frac{P(O|Jtc)*P(Jtc)}{P(O)} \quad (14.12)$$

In this case, the product probability is being compared to the general distribution of the origins of all offenders irrespective of where they committed their crimes. Note that this measure will correlate with the product term because they both have the same numerator.

The Bayesian Journey-to-crime Estimation Module

The Bayesian Journey-to-crime estimation module is made up of two routines, one for diagnosing which Journey-to-crime method is best and one for applying that method to a particular serial offender. Figure 14.4 show the layout of the module.

Data Preparation for Bayesian Journey-to-crime Estimation

There are four data sets that are required:

1. The incidents committed by a single offender for which an estimate will be made of where that individual lives;

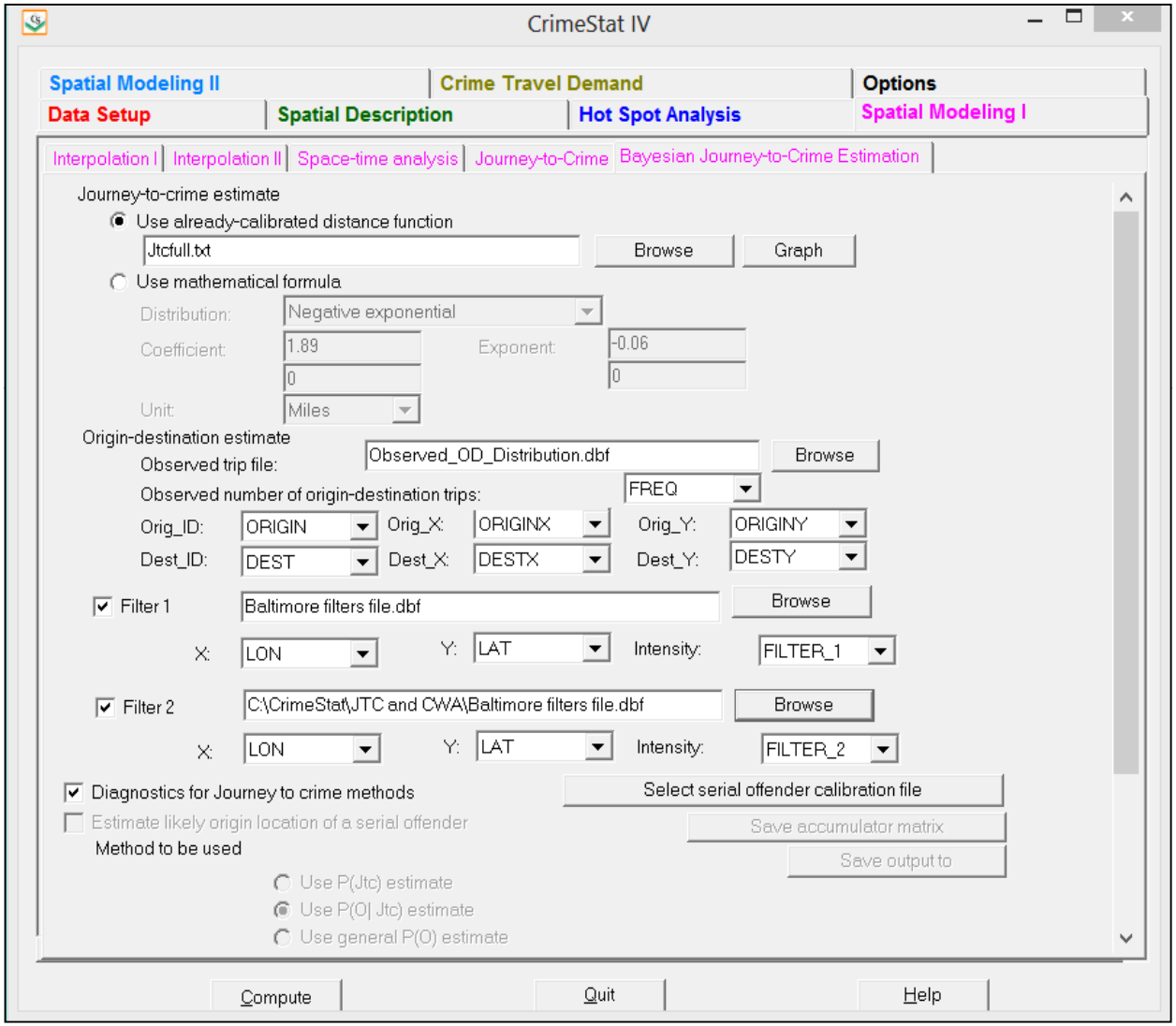
2. A Journey-to-crime travel distance function that estimates the likelihood of an offender committing crimes at a certain distance (or travel time if a network is used);

3. An origin-destination matrix; and

4. A diagnostics file of multiple known serial offenders for which both their residence and crime locations are known (optional for use in diagnostics routine).

Figure 14.4:

Bayesian Journey-to-crime Screen



Serial Offender Data

The first required data set is information on the location of crimes committed by a single serial offender. For each serial offender for whom an estimate will be made of where that person lives, the data set must include the location of the incidents committed by the offender. The data are a series of records in which each represents a single event. On each record, there are X and Y coordinates identifying the location of the incidents this person has committed (Table 14.2). There may be other data on the records, but the X and Y coordinates are essential.

Table 14.2:
Minimum Information Required for Serial Offenders:
Example for Offender Who Committed Seven Incidents

ID	UCR	INCIDX	INCIDY
TS7C	430.00	-76.494300	39.2846
TS7C	440.00	-76.450900	39.3185
TS7C	630.00	-76.460600	39.3157
TS7C	430.00	-76.450700	39.3181
TS7C	311.00	-76.449700	39.3162
TS7C	440.00	-76.450300	39.3178
TS7C	341.00	-76.448200	39.3123

Journey-to-crime Travel Function

The second data set that is required is a journey-to-crime (Jtc) function. The Journey-to-crime travel function is an estimate of the likelihood of an offender traveling a certain distance. Typically, it represents a frequency distribution of distances traveled, though it could be a frequency distribution of travel times if a network was used to calibrate the function with the Journey-to-crime estimation routine. It can come from an *a priori* assumption about travel distances, prior research, or a calibration data set of offenders who have already been caught. The "Calibrate Journey-to-crime function" routine (on the Journey-to-crime page under Spatial modeling) can be used to estimate this function (see Chapter 13).

The BJtc routine can use two different travel distance functions:

1. An already-calibrated distance function; and

2. A mathematical formula.

Either direct or indirect (Manhattan) distances can be used though the default is direct distance (see Measurement Parameters in Chapter 3, p. 3.29). In practice, an empirically-derived

travel function is often as accurate, if not better, than a mathematically-defined one. Given that an origin-destination matrix is also needed, it is easy for the user to estimate the travel function using the "Calibrate Journey-to-crime function".

If the user does not have data to calibrate a journey-to-crime travel function, then a mathematical model should be used. Typically, the negative exponential function is used for this purpose; the default values will work for many distributions.

Origin-destination Matrix

The third required data set is an origin-destination matrix. The origin-destination matrix relates the number of offenders who commit crimes in one of N zones who live (originate) in one of M zones, similar to Figure 14.1 above. It can be created from the "Calculate observed origin-destination trips" routine (on the 'Describe origin-destination trips' page under the Trip distribution module of the Crime Travel Demand model; see Chapter 28).

How many incidents are needed where the origin and destination location are known? While there is no simple answer to this, the numbers ideally should be in the tens of thousands. If there are N destinations and M rows, ideally one would want an average of 30 cases for each cell to produce a reliable estimate. Obviously, that is a huge amount of data that cannot easily found with any real database. For example, if there are 325 destination zones and 532 origin zones (for the Baltimore County example given below), that would be 172,900 individual cells. If the 30 cases or more rule is applied, then that would require 5,187,000 records or more to produce a barely reliable estimate for most cells.

The task becomes even more daunting when it is realized that most of these links (cells) have few or no cases in them as offenders typically travel along certain pathways. Obviously, such a demand for data is impractical even in the largest jurisdictions. Therefore, we recommend that as much data as possible be used to produce the origin-destination (O-D) matrix, at least several years worth. The matrix can be built with what data is available and then periodically updated to produce better estimates.

Diagnostics File for Bayesian Jtc Routine

The fourth data set is an optional diagnostics file. It is used for estimating which of several alternative parameters is best at predicting the residence location of serial offenders. Essentially, it is a set of serial offenders, each record of which has the X and Y coordinates of both the residence location and the crime location. For example, offender T7B committed seven incidents while offender S8A committed eight incidents. The records of both offenders are placed in the same file along with the records for all other offenders in the diagnostics file.

The diagnostics file provides information about which parameter (to be described below) is best at guessing where an offender lives. The assumption is that if a particular parameter was best with the *K* offenders in a diagnostics file in which the residence location was known, then it also will be best for a serial offender for whom the residence location is not known.

How many serial offenders are needed to make up a diagnostics file? Again, there is no simple answer to this although the number is much less than for the O-D matrix. Clearly, the more, the better since the aim is to identify which parameter is most sensitive with a certain level of precision and accuracy. We used 88 offenders in the diagnostics file for Baltimore County (see below). Certainly, a minimum of 10 would be necessary. But, more would certainly be more accurate. Further, the offender records used in the diagnostics file should be similar in other dimensions to the offender that is being tracked. However, this may be impractical. In the example data set, we combined offenders who committed different types of crimes. The results may be different if offenders who had committed only one type of crimes were tested (though Leitner and Kent, 2009, found that using records for all crimes produced more accurate measures than using crime-specific records).

Once the data sets have been collected, they need to be placed in an appended file, with one serial offender on top of another. Each record has to represent a single incident. Further, the records have to be arranged sequentially with all the records for a single offender being grouped together. The routine automatically sorts the data by the offender ID. But, to be sure that the result is consistent, the data should be prepared in this way.

The structure of the records is similar to the example in Table 14.3 below. At the minimum, there needs to be an ID field and the X and Y coordinates of both crime location and the residence location. Thus, in the example, all the records for the first offender (Num 1) are together; all the records for the second offender (Num 2) are together; and so forth. The ID field is any numeric or string variable. In Table 14.3, the ID field is labeled "ID", but any label would be acceptable as long as it is consistent (i.e., all the records of a single offender are together).

In addition to the ID field, the X and Y coordinates of both the crime and residence location must be included on each record. In the example table, the ID variable is called OffenderID, the crime location coordinates are called IncidX and IncidY while the residence location coordinates are called HomeX and HomeY. Again, any label is acceptable as long as the column locations in each record are consistent. As with the Journey-to-crime calibration file, other fields can be included.

Table 14.3:
Example Records in Bayesian Journey-to-crime Diagnostics File

OffenderID	HomeX	HomeY	IncidX	IncidY
Num 1	-77.1496	39.3762	-76.6101	39.3729
Num 1	-77.1496	39.3762	-76.5385	39.3790
Num 1	-77.1496	39.3762	-76.5240	39.3944
Num 2	-76.3098	39.4696	-76.5427	39.3989
Num 2	-76.3098	39.4696	-76.5140	39.2940
Num 2	-76.3098	39.4696	-76.4710	39.3741
Num 3	-76.7104	39.3619	-76.7195	39.3704
Num 3	-76.7104	39.3619	-76.8091	39.4428
Num 3	-76.7104	39.3619	-76.7114	39.3625
Num 4	-76.5179	39.2501	-76.5144	39.3177
Num 4	-76.5179	39.2501	-76.4804	39.2609
Num 4	-76.5179	39.2501	-76.5099	39.2952
Num 5	-76.3793	39.3524	-76.4684	39.3526
Num 5	-76.3793	39.3524	-76.4579	39.3590
Num 5	-76.3793	39.3524	-76.4576	39.3590
Num 5	-76.3793	39.3524	-76.4512	39.3347
Num 6	-76.5920	39.3719	-76.5867	39.3745
Num 6	-76.5920	39.3719	-76.5879	39.3730
Num 6	-76.5920	39.3719	-76.7166	39.2757
Num 6	-76.5920	39.3719	-76.6015	39.4042
Num 7	-76.7152	39.3468	-76.7542	39.2815
Num 7	-76.7152	39.3468	-76.7516	39.2832
Num 7	-76.7152	39.3468	-76.7331	39.2878
Num 7	-76.7152	39.3468	-76.7281	39.2889
.				
.				
.				
.				
Num Last	-76.4320	39.3182	-76.4297	39.3172
Num Last	-76.4880	39.3372	-76.4297	39.3172
Num Last	-76.4437	39.3300	-76.4297	39.3172
Num Last	-76.4085	39.3342	-76.4297	39.3172
Num Last	-76.4083	39.3332	-76.4297	39.3172
Num Last	-76.4082	39.3324	-76.4297	39.3172
Num Last	-76.4081	39.3335	-76.4297	39.3172

Logic of the Routine

The module is divided into two parts (under the "Bayesian Journey-to-crime Estimation" page of "Spatial Modeling"):

1. Diagnostics for Journey-to-crime methods; and

2. Estimate likely origin location of a serial offender.

The "diagnostics" routine takes the diagnostics calibration file and estimates a number of methods for each serial offender in the file and tests the accuracy of each parameter against the known residence location. The result is a comparison of the different methods in terms of accuracy in predicting both where the offender lives as well as minimizing the distance between where the method predicts the most likely location for the offender and where the offender actually lives.

The "estimate" routine allows the user to choose one method and to apply it to the data for a ***single*** serial offender. The result is a probability surface showing the results of the method in predicting where the offender is liable to be living.

Bayesian Journey-to-crime Diagnostics

The following applies to the Bayesian Journey-to-crime (BJtc) Diagnostics routine only.

Data Input

The user inputs the four required data sets.

1. Any primary file with an X and Y location. A suggestion is to use the file for the one of the serial offenders, but this is not essential;

2. A grid that will be overlaid on the study area. Use the Reference File under Data Setup to define the X and Y coordinates of the lower-left and upper-right corners of the grid as well as the number of columns (see Chapter 3, p. 3.21);

3. A Journey-to-crime travel function (Jtc) that estimates the likelihood of an offender committing crimes at various distances or travel times if a network is used (see Chapter 13).

4. An observed origin-destination matrix (see Chapter 28, p. 28.17); and

5. A diagnostics file of known serial offenders in which both their residence and crime locations are known (BJtc Diagnostics)

Methods Tested

The BJtc Diagnostics routine compares six methods for estimating the likely location of a serial offender and will include up to four additional methods if filters are used (see below).

1. The Jtc distance method, P(Jtc);

2. The general crime distribution based on the origin-destination matrix, P(O). Essentially, this is the distribution of origins irrespective of the destinations;

3. The distribution of origins in the O-D matrix based only on the incidents in zones that are identical to those committed by the serial offender, P(O|Jtc);

4. The product of the Jtc estimate (1 above) and the distribution of origins based only on those incidents committed in zones identical to those by the serial offender (3 above), P(Jtc)*P(O|Jtc). This is the numerator of the Bayesian function (Equation 14.8), the product of the prior probability times the likelihood estimate;

5. The Bayesian Risk estimate as indicated in Equation 14.8 above (method 4 above divided by method 2 above), P(Bayesian). This is a rough approximation to the full Bayesian function in Equation 14.12 above; and

6. The center of minimum distance, Cmd. Previous research has shown that the center of minimum of distance has the least error in minimizing the distance between the most likely location for the offender and where the offender actually lives (Paulsen, 2006a; Snook, Zito, Bennell, & Taylor, 2005; Levine, 2000).

7. If filters are used: The product of the Jtc estimate and the filters, P(Jtc)*F1*F2.

8. If filters are used: The product of the Conditional estimate and the filters, P(O|Jtc)*F1*F2.

9. If filters are used: The product of the "Product" estimate and the filters, P(Jtc)*P(O|Jtc)*F1*F2.

10. If filters are used: The product of the Bayesian Risk estimate and the filters, P(Bayesian)*F1*F2.

Interpolated Grid

For each serial offender in turn in the BJtc Diagnostics file and for each method, the routine overlays a grid over the study area. The grid is defined by the Reference File parameters (see Chapter 3). The routine then interpolates each input data set into a probability estimate for each grid cell with the sum of the cells equaling 1.0 (within three decimal places).

The manner in which the interpolation is done varies by the method:

1. For the Jtc method, P(Jtc), the routine interpolates the selected distance function to each grid cell to produce a density estimate. The densities are then re-scaled so that the sum of the grid cells equals 1.0 (see Chapter 10 on kernel density interpolation);

2. For the general crime distribution method, P(O), the routine sums up the incidents by each origin zone from the origin-destination matrix and interpolates that using the normal distribution method of the single kernel density routine (see Chapter 10 on kernel density interpolation). The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0;

3. For the distribution of origins based only on the incidents committed by the serial offender, from the origin-destination matrix the routine identifies the zone in which the incidents occurred and reads only those origins associated with those destination zones. Multiple incidents committed in the same origin zone are counted multiple times. The routine adds up the number of incidents counted for each zone and uses the single kernel density routine to interpolate the distribution to the grid (see Chapter 10 on kernel density interpolation). The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0;

4. For the product of the Jtc estimate and the distribution of origins based only on the incidents committed by the serial offender, the routine multiples the probability estimate obtained in 1 above by the probability estimate obtained in 3 above. The probabilities are then re-scaled so that the sum of the grid cells equals 1.0;

5. For the Bayesian Risk estimate, the routine takes the product estimate (4 above) and divides it by the general crime distribution estimate (2 above). The resulting probabilities are then re-scaled so that the sum of the grid cells equals 1.0; and

6. For the center of minimum distance estimate, the routine calculates the center of minimum distance for each serial offender in the "diagnostics" file and calculates the distance between this statistic and the location where the offender is actually residing. This is used only for the distance error comparisons.

7. For the interaction of the Jtc, Conditional, "Product" and Bayesian Risk estimates with the filters, the estimates are obtained by multiplying the filters times these terms and then re-scaling so that the sum of the grid cells equals 1.0.

Note in all of the probability estimates (excluding 6), the cells are converted to probabilities prior to any multiplication or division. The results are then re-scaled so that the resulting grid is a probability (i.e., all cells sum to 1.0).

Output

For each offender in the BJtc Diagnostics file, the routine calculates three different statistics for **each** of the methods:

1. The estimated probability in the cell where the offender actually lives. It does this by, first, identifying the grid cell in which the offender lives (i.e., the grid cell where the offender's residence X and Y coordinate is found) and, second, by noting the probability associated with that grid cell;

2. The percentile of all grid cells in the entire grid that have to be searched to find the cell where the offender lives based on the probability estimate from 1 above, ranked from those with the highest probability to the lowest. Obviously, this percentage will vary by how large a reference grid is used (e.g., with a very large reference grid, the percentile where the offender actually lives will be small whereas with a small reference grid, the percentile will be larger). But, since the purpose is to compare methods, the actual percentage should be treated as a relative index.

   The result is sorted from low to high so that the smaller the percentile, the better. For example, a percentile of 1% indicates that the probability estimate for the cell where the offender lives is within the top 1% of all grid cells. Conversely, a percentile of 30% indicates that the probability estimate for the cell where the offender lives in within the top 30% of all grid cells; and

3. The distance between the cell with the highest probability and the cell where the offender lives.

These three indices provide information about the accuracy and precision of the method. Table 14.4 illustrates a typical probability output for four of the methods. Only five serial offenders are shown in the table.

Output matrices

The BJtc Diagnostics routine outputs two separate matrices. The probability estimates (numbers 1 and 2 above) are presented in a separate matrix from the distance estimates (number

Table 14.4:
Sample Output of Probability Matrix

Offender	P(Jtc)	Percentile for P(Jtc)	P(O|Jtc)	Percentile for P(O|Jtc)	P(O)	Percentile for P(O)	P(Jtc)*P(O|Jtc)	Percentile for P(Jtc)*P(O|Jtc)
1	0.001169	0.01%	0.000663	0.01%	0.0003	11.38%	0.002587	0.01%
2	0.000292	5.68%	0.000483	0.12%	0.000377	0.33%	0.000673	0.40%
3	0.000838	0.14%	0.000409	0.18%	0.0002	30.28%	0.00172	0.10%
4	0.000611	1.56%	0.000525	1.47%	0.0004	2.37%	0.000993	1.37%
5	0.001619	0.04%	0.000943	0.03%	0.000266	11.98%	0.004286	0.04%

Table 14.5 illustrates a typical distance output for four of the methods. Only five serial offenders are shown in the table.

Table 14.5:
Sample Output of Distance Matrix

Offender	Distance(Jtc)	Distance(O|Jtc)	Distance(O)	Distance for P(Jtc)*P(O|Jtc)
1	0.060644	0.060644	7.510158	0.060644
2	6.406375	0.673807	2.23202	0.840291
3	0.906104	0.407762	11.53447	0.407762
4	3.694369	3.672257	2.20705	3.672257
5	0.423577	0.405526	6.772228	0.423577

3 above). The user can save the total output as a text file or can copy and paste each of the two output matrices into a spreadsheet separately. We recommend the copying-and-pasting method into a spreadsheet as it will be difficult to line up differing column widths for the two matrices and summary tables in a text file.

Which is the Most Accurate and Precise Method?

Accuracy and precision are two different criteria for evaluating a method. With accuracy, one wants to know how close a method comes to a target. The target can be an exact location (e.g., the residence of a serial offender) or it can be a zone (e.g., a high probability area within which the serial offender lives). Precision, on the other hand, refers to the consistency of the method, irrespective of how accurate it is. A more precise measure is one in which the method has a limited variability in estimating the central location whereas a less precise measure has a higher degree of variability. These two criteria - accuracy and precision, often conflict.

The following example is from Jessen (1979). Consider a target that one is trying to 'hit' (Figure 14.5A). The target can be a physical target, such as a dart board, or it can be a location in space, such as the residence of a serial offender. One can think of three different 'throwers' or methods attempting to hit the center of target, the Bulls Eye. The throwers make repeated attempts to hit the target and the 'throws' (or estimates from the method) can be evaluated in terms of accuracy and precision. In Figure 14.5B, the thrower is all over the dartboard. There is no consistency at all. However, if the center of minimum distance (Cmd) is calculated, it is very close to the actual center of the target, the Bulls Eye. In this case, the thrower is accurate but not precise. That is, there is no systematic bias in the thrower's throws, but they are not reliable. This thrower is accurate (or unbiased) but not precise.

In Figure 14.5C, there is an opposite condition. In this case, the thrower is precise but not accurate. That is, there is a systematic bias in the throws even though the throws (or method) are relatively consistent. Finally in Figure 14.5D, the thrower is both relatively precise and accurate as the Cmd of the throws is almost exactly on the Bulls Eye.

One can apply this analogy to a method. A method produces estimates from a sample. For each sample, one can evaluate how accurate is the method (i.e., how close to the target did it come) and how consistent is it (how much of variability does it produce). Perhaps the analogy is not perfect because the thrower makes multiple throws whereas the method produces a single estimate. But, clearly, we want a method that is both accurate and precise.

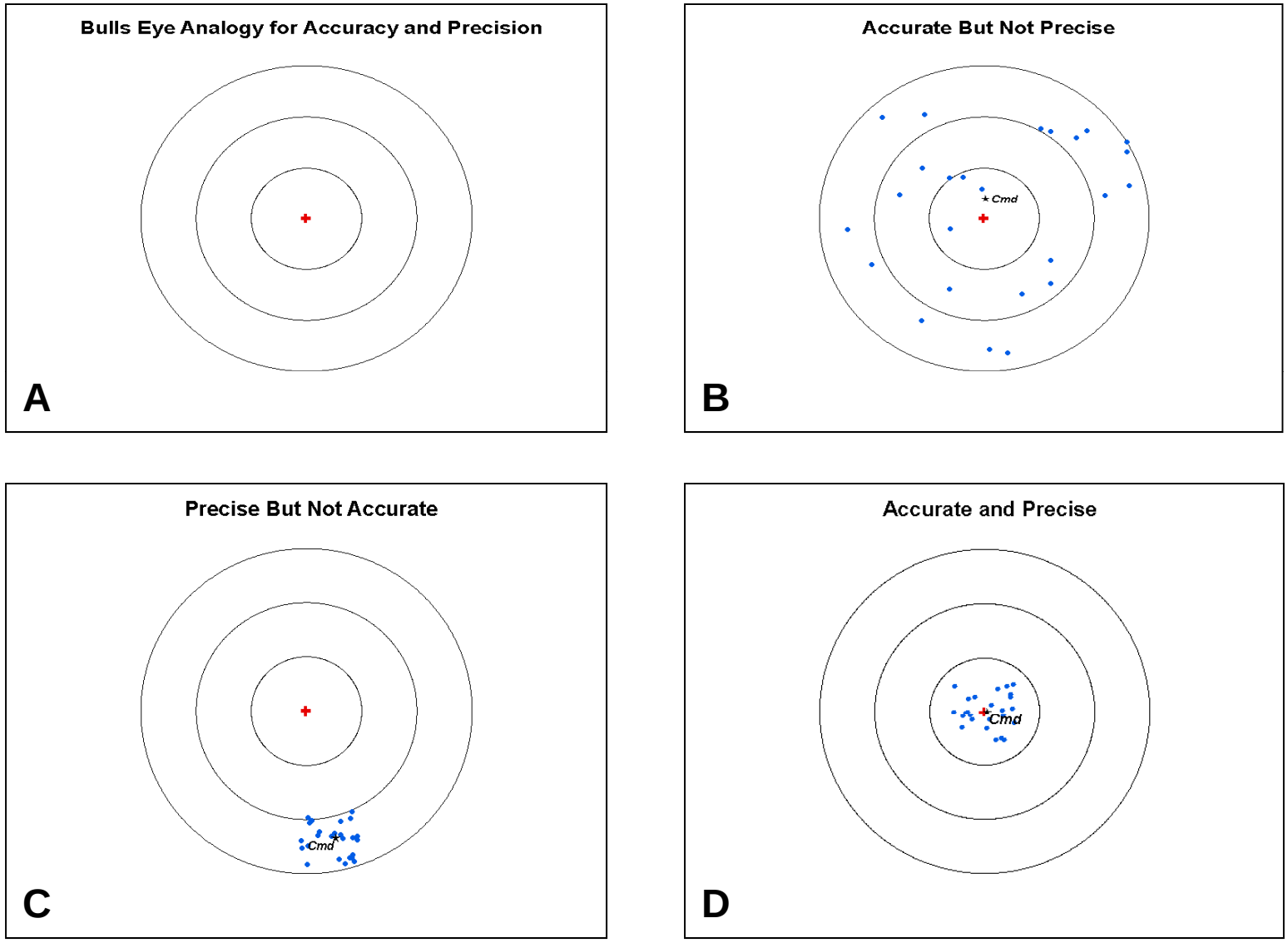
Measures of Accuracy and Precision

Much of the debate in the area of Journey-to-crime estimation has revolved around arguments about the accuracy and precision of the method. Levine (2000) first raised the issue of accuracy by proposing distance from the location with the highest probability to the location where the offender lived as a measure of accuracy, and suggested that simple, centrographic measures were as accurate as more precise Journey-to-crime methods in estimating this. Paulsen (2006a; 2006b) confirmed that centrgraphic methods were more accurate than journey-to-crime method. Snook and colleagues also confirmed this conclusion and showed that human subjects could do as well as any of the algorithms (Snook, Zito, Bennell, & Taylor, 2005; Snook, Taylor & Bennell, 2004).

On the other hand, Canter, Coffey and Missen (2000), Canter (2003), and Rossmo (2000) have argued for an area of highest probability being the criterion for evaluating accuracy, indicating a 'search cost' or a 'hit score' with the aim being to narrow the search area to as small

Figure 14.5:

Accuracy and Precision in Estimates



as possible. Rich and Shivley (2004) compared different Journey-to-crime/geographic profiling software packages and concluded that there were at least five different criteria for evaluating accuracy and precision - error distance, search cost/hit score, profile error distance, top profile area, and profile accuracy.

Rossmo (2005a; b) and Rossmo and Filer (2005) have critiqued these measures as being too simple and have rejected error distance. Levine (2005) justified the use of error distance as being fundamental to statistical error while acknowledging that an area measure is necessary, too.

While the debate continues to develop, practically a distinction can be made between measures of accuracy and measures of precision. Accuracy is measured by how close to the target is the estimate while precision refers to how large or small an area the method produces. The two become identical when the precision is extremely small, similar to a variance converging into a mean as the distance between observations and the mean approach zero.

In evaluating the methods, five different measures are used:

Accuracy Measures

1. **True accuracy** - the probability in the cell where the offender actually lives. The Bayesian Jtc diagnostics routine evaluates the six above mentioned methods on a sample of serial offenders with known residence address. Each of the methods (except for the center of minimum distance, Cmd) has a probability distribution. That method which has the highest probability in the cell where the offender lives is the most accurate.

2. **Diagnostic accuracy** - the distance between the cell with the highest probability estimate and the cell where the offender lives. Each of the methods produces probability estimates for each cell. The cell with the highest probability is the best guess for where the offender lives. The distance from this location to where the offender lives is an indicator of the diagnostic accuracy of the method.

3. **Neighborhood accuracy** - the percent of offenders who reside within the cell with the highest probability. Since the grid cell is the smallest unit of resolution, this measures the percent of all offenders who live at the highest probability cell. This was estimated by those cases where the error distance was smaller than half the grid cell size.

Precision Measures

4. **Search cost/hit score** - the percent of the total study area that has to be searched to find the cell where the offender actually lived after having sorted the output cells from the highest probability to the lowest

5. **Potential search cost** - the percent of offenders who live within a specified distance of the cell with the highest probability. In this evaluation, two distances are used though others can certainly be used:

    A. The percent of offender who live within one mile of the cell with the highest probability.

    B. The percent of offenders who live within one-half mile of the cell with the highest probability ("Probable search area in miles").

Summary Statistics

The "diagnostics" routine will also provide summary information at the bottom of each matrix. There are summary measures and counts of the number of times a method had the highest probability or the closest distance from the cell with the highest probability to the cell where the offender actually lived; ties between methods are counted as fractions (e.g., two tied methods are given 0.5 each; three tied methods are give 0.33 each). For the probability matrix, these statistics include:

1. The mean (probability or percentile);
2. The median (probability or percentile);
3. The standard deviation (probability or percentile);
4. The number of times the P(Jtc) estimate produces the highest probility;
5. The number of times the P(O|Jtc) estimate produces the highest probability;
6. The number of times the P(O) estimate produces the highest probability;
7. The number of times the product term estimate produces the highest probability;
8. The number of times the Bayesian estimate produces the highest probability.
9. If filters are used: The number of times the Jtc, Conditional, Product, and Bayesian Risk estimates times the filters produce the highest probability.

For the distance matrix, these statistics include:

1. The mean distance;
2. The median distance;
3. The standard deviation distance;
4. The number of times the P(Jtc) estimate produces the closest distance;
5. The number of times the P(O|Jtc) estimate produces the closest distance;
6. The number of times the P(O) estimate produces the closest distance;
7. The number of times the product term estimate produces the closest distance;
8. The number of times the Bayesian Risk estimate produces the closest distance; and
9. The number of times the CMD produces the closest distance.
10. If filters are used: The number of times the Jtc, Conditional, Product, and Bayesian Risk estimates times the filters produce the closest distance.

Testing the Routine with Serial Offenders from Baltimore County

To illustrate the use of the Bayesian Jtc diagnostics routine, the records of 88 serial offenders who had committed crimes in Baltimore County, MD, between 1993 and 1997 were compiled into a diagnostics file. The number of incidents committed by these offenders varied from 3 to 33 and included a range of different crime types (larceny, burglary, robbery, vehicle theft, arson, and bank robbery).

Because the methods are interdependent, traditional parametric statistical tests cannot be used. Instead, non-parametric tests have been applied. For the probability and distance measures, two tests were used. First, the Friedman two-way analysis of variance test examines differences in the overall rank orders of multiple measures (treatments) for a group of subjects (Kanji, 1993, 115; Siegel, 1956). This is a Chi-square test and measures whether there are significant differences in the rank orders across all measures (treatments). Second, differences between specific pairs of measures can be tested using the Wilcoxon matched pairs signed-ranks test (Siegel, 1956, 75-83). This examines pairs of methods by not only their rank, but also by the difference in the values of the measurements.

For the percentage of offenders who lived in the same grid cell, within one mile, and within one half-mile of the cell with the peak likelihood, the Cochran Q test for *k* related samples was used to test differences among the methods (Kanji, 1993, 74; Siegel, 1956, 161-166). This is a Chi-square test of whether there are overall differences among the methods in the percentages, but cannot indicate whether any one method has a statistically higher percentage. Consequently, the method with the highest percentage was tested against the method with the second highest percentage using the Cochran Q test to see whether the best method stood out.

Results: Accuracy

Table 14.6 presents the results three accuracy measures. For the first measure, the probability estimate in the cell where the offender actually lived, the product probability is far superior to any of the others. It has the highest mean probability of any of the measures and is more than double the probability of the Journey-to-crime method. The Friedman test indicates that these differences are significant and the Wilcoxon matched pairs test indicates that the product has a significantly higher probability than the second best measure, the Bayesian Risk, which in turn is significantly higher than the Journey-to-crime measure. At the low end, the general probability has the lowest average and is significantly lower than the other measures.

In terms of the individual offenders, the product probability had the highest probability for 74 of the 88 offenders. For the other methods, the Bayesian Risk measure had the highest probability for 10 offenders, the Journey-to-crime measure for one offender, the conditional probability for two offenders and the general probability for one offender.

Table 14.6:
Accuracy Measures of Total Sample

Method	**Mean probability in offender cell**[a]	**Mean distance from highest probability cell to offender cell (mi)**[b]	**Percent of offenders whose residence is in highest prob. cell**[c]
Journey-to-crime	0.00082	2.78	12.5%
General	0.00025	8.21	0.0%
Conditional	0.00052	3.22	3.4%
Product	**0.00170**	2.65	13.6%
Bayesian Risk	0.00131	3.15	10.2%
Cmd	n.a.	**2.62**	**18.2%**

a Friedman χ^2 =236.0 ; d.f. = 4; p≤.001; Wilcoxon signed-ranks test at p≤.05: Product > Bayesian Risk > JTC = Conditional > General

b Friedman χ^2 = 114.2; d.f. = 5; p≤.001; Wilcoxon signed-ranks test at p≤.05: CMD = Product = JTC > Bayesian Risk = Conditional < General

c Cochran Q=33.9, d.f. =5, p≤.001; Cochran Q of difference between best & second best=1.14, n.s.

Finally, for the third accuracy measure, the percent of offenders residing in the area covered by the cell with the highest probability estimate, the Cmd has the highest percentage (18.2%) followed by the product probability (13.6%), and the Journey-to-crime probability (12.5%). The Cochran Q shows significant differences over all these measures. However, the difference between the measure with the highest percentage in the same grid cell (the Cmd) and the measure with the second highest percentage (the product probability) is not significant.

For accuracy, the product probability appears to be better than the Journey-to-crime estimate and almost as accurate as the Cmd. It has the highest probability in the cell where the offender lived and a lower error distance than the Journey-to-crime method (though not significantly so). Finally, it had a slightly higher percentage of offenders living in the area covered by cell with the highest probability than for the Journey-to-crime.

The Cmd, on the other hand, which had been shown to be the most accurate in previous studies (Paulsen, 2006a; 2006b; Snook, Zito, Bennell, and Taylor, 2005; Snook, Taylor and Bennell, 2004; Levine, 2000), does not appear to be more accurate than the product probability. It has only a slightly lower error distance and a slightly higher percentage of offenders residing in the area covered by the cell with the highest probability. Thus, the product term has equaled the Cmd in terms of accuracy. Both, however, are more accurate than the Journey-to-crime estimate.

For the measure of diagnostic accuracy (the distance from the cell with the highest probability to the cell where the offender lived), the center of minimum distance (Cmd) had the lowest error distance followed closely by the product term. The Journey-to-crime method had a slightly larger error. Again, the general probability had the greatest error, as might be expected. The Friedman test indicates there are overall differences among the six measures in the mean distance. The Wilcoxon signed-ranks test, however, showed that the Cmd, the product, and the Journey-to-crime estimates are not significantly different, though all are significantly lower than the Bayesian Risk measure and the conditional probability which, in turn, are significantly lower than the general probability.

In terms of individual cases, the Cmd produced the lowest average error distance for 30 of the 88 cases while the conditional term (O|Jtc) had the lowest error distance in 17.9 cases (including ties). The product term produced a lower average error distance for 9.5 cases (including ties) and the Jtc estimate produced lower average distance errors in 8.2 cases (again, including ties). In other words, the Cmd will either be very accurate or very inaccurate, which is not surprising given that it is only a point estimate.

Results: Precision

Table 14.7 presents the three precision measures used to evaluate the six different measures. For the first measure, the mean percent of the study area with a higher probability (what Canter called 'search cost' and Rossmo called 'hit score'; Canter, 2003; Rossmo, 2005a, 2005b), the Bayesian Risk measure had the lowest percentage followed closely by the product term. The conditional probability was third followed by the Journey-to-crime probability followed by the general probability. The Friedman test indicates that these differences are significant overall and the Wilcoxon test shows that the Bayesian Risk, product term, conditional probability and Journey-to-crime estimates are not significantly different from each other. The general probability estimate, however, is much worse.

In terms of individual cases, the product probability had either the lowest percentage or was tied with other measures for the lowest percentage in 36 of the 88 cases. The Bayesian Risk and Journey-to-crime measures had the lowest percentage or were tied with other measures for the lowest percentage in 34 of the 88 cases. The conditional probability had the lowest percentage or was tied with other measures for the lowest percentage in 23 of the cases. Finally, the general probability had the lowest percentage or was tied with other measures for the lowest percentage in only 7 of the cases.

Similar results are seen for the percent of offenders living within one mile of the cell with the highest probability and also for the percent living within a half mile. For the percent within one mile, the product term had the highest percentage followed closely by the Journey-to-crime measure and the Cmd. Again, at the low end is the general probability. The Cochran Q test indicates that these differences are significant over all measures though the difference between the best method (the product) and the second best (the Journey-to-crime) is not significant.

Conclusion of the Evaluation

In conclusion, the product method appears to be an improvement over the Journey-to-crime method, at least with these data from Baltimore County. It is substantially more accurate and about as precise. Further, the product probability appears to be, on average, almost as accurate as the Cmd, though the Cmd still is more accurate in assessing the exact location of offenders. That is, the Cmd will identify about one-sixth of all offenders exactly. For a single guess of where a serial offender is living, the center of minimum distance produced the lowest distance error. But, since it is only a point estimate, it cannot point to a search area where the offender might be living. The product term, on the other hand, produced an average distance error almost as small as the center of minimum distance, but produced estimates for other grid cells too. Among all the probability measures, it had the highest probability in the cell where the offender lived and was among the most efficient in terms of reducing the search area.

Table 14.7:
Precision Measures of Total Sample

Method	Mean percent of study area with higher probability[a]	Percent of offenders living within distance of highest probability cell: 1 mile[b]	0.5 miles[c]
Journey-to-crime	4.7%	56.8%	44.3%
General	16.8%	2.3%	0.0%
Conditional	4.6%	47.7%	31.8%
Product	4.2%	**59.1%**	**48.9%**
Bayesian Risk	**4.1%**	51.1%	42.0%
Cmd	n.a.	54.5%	42.0%

a Friedman χ^2 =115.4 ; d.f. = 4; p<.001; Wilcoxon signed-ranks test at p<.05: Bayesian Risk =Product= JTC = Conditional> General

b Cochran Q = 141.0, d.f. = 5, p<.001; Cochran Q of difference between best and second best = 0.7, n.s.

c Cochran Q = 112.2, d.f. = 5, p<.001; Cochran Q of difference between best and second best = 2.0, n.s.

In other words, using information about the origin location of other offenders appears to improve the accuracy of the Jtc method. The result is an index (the product term) that is almost as good as the center of minimum distance, but one that is more useful since the center of minimum distance is only a single point.

Of course, each jurisdiction should re-run these diagnostics to determine the most appropriate measure. It is very possible that other jurisdictions will have different results due to the uniqueness of their land uses, street layout, and location in relation to the center of the metropolitan area. Baltimore County surrounds the City of Baltimore on three sides. It has a mixture of neighborhoods including parts of the central city, older suburbs, newer suburbs, separate communities and rural areas. The model results which fit Baltimore County might not fit other places.

Tests with Other Data Sets

The Bayesian Journey-to-crime model was tested in 2009 in several jurisdictions:

1. In Baltimore County with 850 serial offenders (Leitner & Kent, 2009);

2. In the Hague, Netherlands with 62 serial burglars (Block & Bernasco, 2009);

3. In Chicago, with 103 serial robbers (Levine & Block, 2011); and

4. In Manchester, England with 171 serial offenders (Levine & Lee, 2009).

In all cases, the product probability measure was both more accurate and more precise than the Journey-to-crime measure. In two of the studies (Chicago and the Hague), the product term was also more accurate than the Center of Minimum Distance. In the other two studies (Baltimore County and Manchester), the Center of Minimum Distance was slightly more accurate than the product term.

Among the probability methods, the product term was more accurate than all other measures for three of the studies (Baltimore County, Chicago, Manchester). For the Hague study, however, the conditional estimate was more accurate. This was because the journey-to-crime estimate was very inaccurate due to the small size of the Hague (Block & Bernasco, 2009).

The mathematics of these models has been explored by O'Leary (2009). These studies are presented in a special issue of the *Journal of Investigative Psychology and Offender Profiling*. Introductions are provided by Canter (2009) and Levine (2009).

In short, the product term appears to be almost as good a method as the Center of Minimum Distance and generally the best of the probability methods. However, users should first test whether this conclusion holds for their jurisdiction.

Estimate Likely Origin Location of a Serial Offender

The following applies to the Bayesian Jtc Estimate Likely Origin Location (BJtc) of a Serial Offender routine. Once the BJtc Diagnostic routine has been run and a preferred method selected, the next routine allows the application of that method to a ***single*** serial offender.

Data Input

The user inputs the three required data sets and a reference file grid:

1. The incidents committed by a single offender that we are interested in catching. This ***must*** be the Primary File;

2. A Jtc function that estimates the likelihood of an offender committing crimes at a certain distance (or travel time if a network is used). This can be either a mathematically-defined function or an empirically-derived one (see Chapter 13 on Journey-to-crime Estimation). In general, the empirically-derived function is slightly more accurate than the mathematically-defined one though the differences are not large;

3. An origin-destination matrix; and

4. The reference file also needs to be defined and should include all locations where crimes have been committed (see Reference File).

Selected Method

The BJtc routine interpolates the incidents committed by the serial offender to a grid, yielding an estimate of where the offender is liable to live. There are five standard methods that can be used and ten additional methods if filters are used. However, the user has to choose ***one*** of these:

1. The Jtc distance method, P(Jtc);

2. The general crime distribution based on the origin-destination matrix, P(O). Essentially, this is the distribution of origins irrespective of the destinations;

3. The conditional Jtc distance. This is the distribution of origins based only on the incidents committed by other offenders in the same zones as those committed by the serial offender, P(O|Jtc). This is extracted from the O-D matrix;

4. The product of the Jtc estimate (1 above) and the distribution of origins based only on the incidents committed by the serial offender (3 above), P(Jtc)*P(O|Jtc). This is the numerator of the Bayesian function (Equation 14.8), the product of the prior probability times the likelihood estimate; and

4. The Bayesian Risk estimate as indicated in Equation 14.12 above (method 4 above divided by method 2 above), P(Bayesian).

5. If one filter is used: the interaction between the Jtc, the Conditional, the Product, and the Bayesian Risk measures with the one filter. Also, the filter by itself can be checked.

6. If two filters are used: the interaction between the Jtc, the Conditional, the Product, and the Bayesian measures with the two filters. Also, the two filters by themselves can be checked.

As mentioned, however, the user must choose only one of these for estimation.

Interpolated Grid

For the BJtc method that is selected, the routine overlays a grid on the study area. The grid is defined by the reference file parameters (see Chapter 3). The routine then interpolates the input data set (the primary file) into a probability estimate for each grid cell with the sum of the cells equaling 1.0 (within three decimal places). The manner in which the interpolation is done varies by the method chosen:

1. For the Jtc method, P(Jtc), the routine interpolates the selected distance function to each grid cell to produce a density estimate. The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0 (see Chapter 10 on kernel density interpolation);

2. For the general crime distribution method, P(O), the routine sums up the incidents by each origin zone and interpolates this to the grid using the normal distribution method of the single kernel density routine (see Chapter 10 on kernel density interpolation). The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

3. For the distribution of origins based only on the incident committed by the serial offender, the routine identifies the zone in which the incident occurs and reads only those origins associated with those destination zones in the origin-destination matrix. Multiple incidents committed in the same origin zone are counted multiple times. The routine then uses the single kernel density routine to interpolate the distribution to the grid (see Chapter 10). The density estimates are converted to probabilities so that the sum of the grid cells equals 1.0;

4. For the product of the Jtc estimate and the distribution of origins based only on the incidents committed by the serial offender, the routine multiples the probability estimate obtained in 1 above by the probability estimate obtained in 3 above. The product probabilities are then re-scaled so that the sum of the grid cells equals 1.0; and

5. For the full Bayesian estimate as indicated in equation 14.12 above, the routine takes the product estimate (4 above) and divides it by the general crime distribution estimate (2 above). The resulting density estimates are converted to probabilities so that the sum of the grid cells equals 1.0.

6. For any of the interactions with filters, the routine takes the filters and interpolates them to a grid and converts the estimates to probabilities. If there is only filter, then this layer is interpolated to the grid and converted into probabilities. If there are two filters, each is first interpolated to the grid and converted into probabilities. Then the two probability interpolations are multiplied by each other. The routine then multiplies the resulting filter probability by the probability estimate of the selected measure (Jtc, Conditional, Product, or Bayesian Risk). The resulting multiplication product is then re-scaled so that sum of the grid cells equals 1.0.

Note in all estimates, the results are then re-scaled so that the resulting grid is a probability (i.e., all cells sum to 1.0).

Output

Once the method has been selected, the routine interpolates the data to the grid cell and outputs it as a 'shp', 'mif/mid', or Ascii file for display in a GIS program. The tabular output shows the probability values for each cell in the matrix and also indicates which grid cell has the highest probability estimate.

Accumulator Matrix

There is also an intermediate output, called the *accumulator matrix* which the user can save. This lists the number of origins identified in each origin zone for the specific pattern of incidents committed by the offender, prior to the interpolation to grid cells. That is, in reading the origin-destination file, the routine first identifies which zone each incident committed by the offender falls within. Second, it reads the origin-destination matrix and identifies which origin zones are associated with incidents committed in the particular destination zones. Finally, it sums up the number of origins by zone ID associated with the incident distribution of the

offender. This can be useful for examining the distribution of origins by zones prior to interpolating these to the grid.

Two Examples of Using the Bayesian Journey-to-crime Routine

Two examples will illustrate the routine. Figure 14.6 presents the probability output for the general origin model, that is for the origins of all offenders irrespective of where they committed their crimes. It is a probability surface in that all the grid cells sum to 1.0. The map is scaled so that each bin covers a probability of 0.0001. The cell with the highest probability is highlighted in light blue.

As seen, the distribution is heavily weighted towards the center of the metropolitan area, particularly in the City of Baltimore. For the crimes committed in Baltimore County between 1993 and 1997 in which both the crime location and the residence location was known, about 40% of the offenders resided within the City of Baltimore and the bulk of those living within Baltimore County lived close to the border with City. In other words, as a general condition, most offenders in Baltimore County live relatively close to the center.

Offender S14A

The general probability output does not take into consideration information about the particular pattern of an offender. Therefore, we will examine specifically a particular offender. Figure 14.7 maps the distribution of an offender who committed 14 offenses between 1993 and 1997 (offender S14A) before being caught and the residence location where the individual lived when arrested

Of the 14 offenses, seven were thefts (larceny), four were assaults, two were robberies, and one was a burglary. As seen, most of the incidents occurred in the southeast corner of Baltimore County though two incidents were committed more than five miles away from the offender's residence.

The general probability model is not very precise since it assigns the same probability to all grid cells for all offenders. In the case of offender S14A, the error distance between the cell with the highest probability and the cell where the offender actually lived was 7.4 miles.

On the other hand, the Jtc method uses the distribution of the incidents committed by a particular offender and a model of a typical travel distance distribution to estimate the likely origin of the offender's residence. A travel distance estimate based on the distribution of 41,424 offenders from Baltimore County was created using the *CrimeStat* Journey-to-crime calibration routine (see Chapter 13 on Journey-to-crime Estimation).

Figure 14.6:

Bayesian Journey-to-crime Routine

General Distribution of Offenders by Residence Location

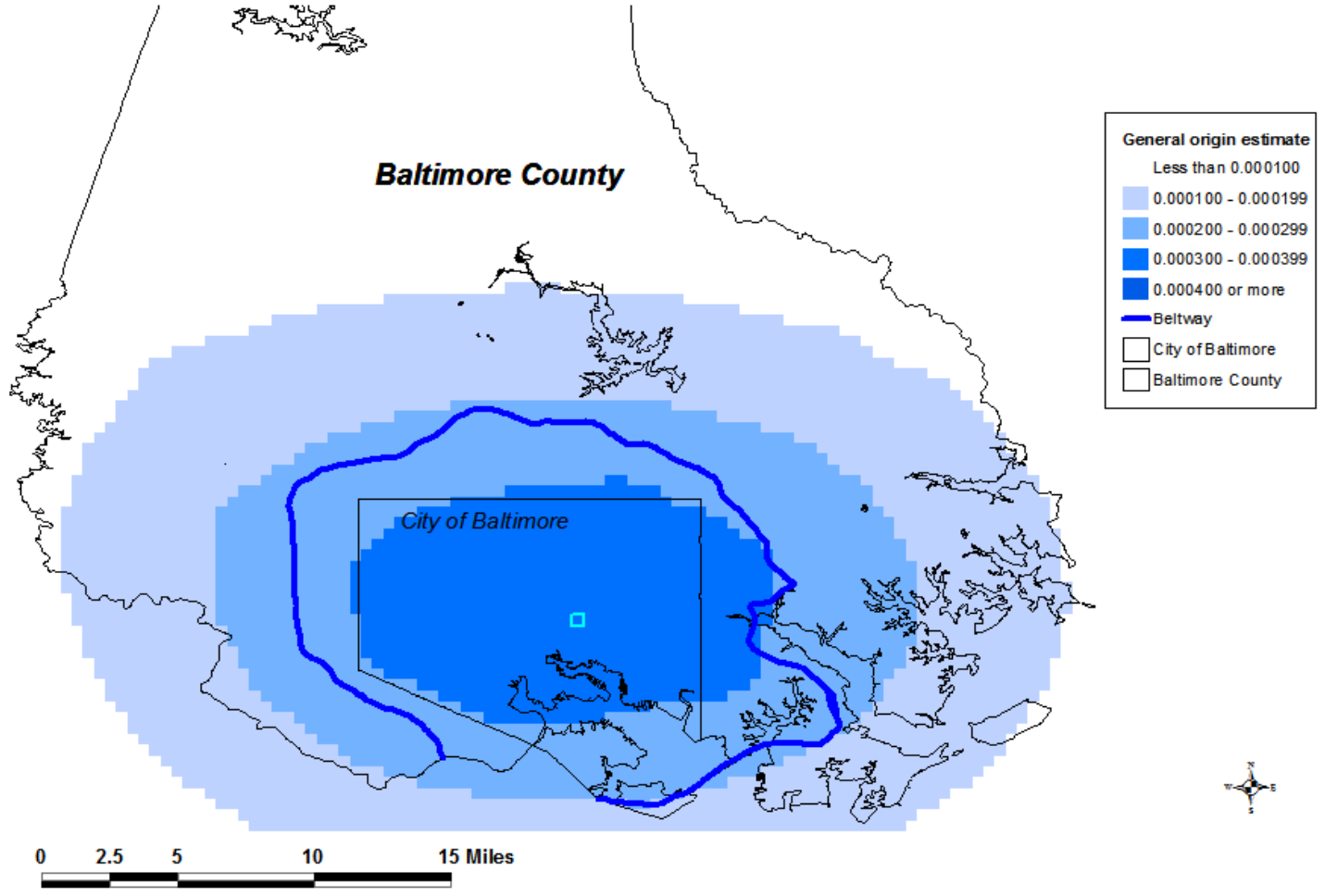


Figure 14.7:

Bayesian Journey-to-crime Routine

Location of Incidents and Residence of Offender S14A

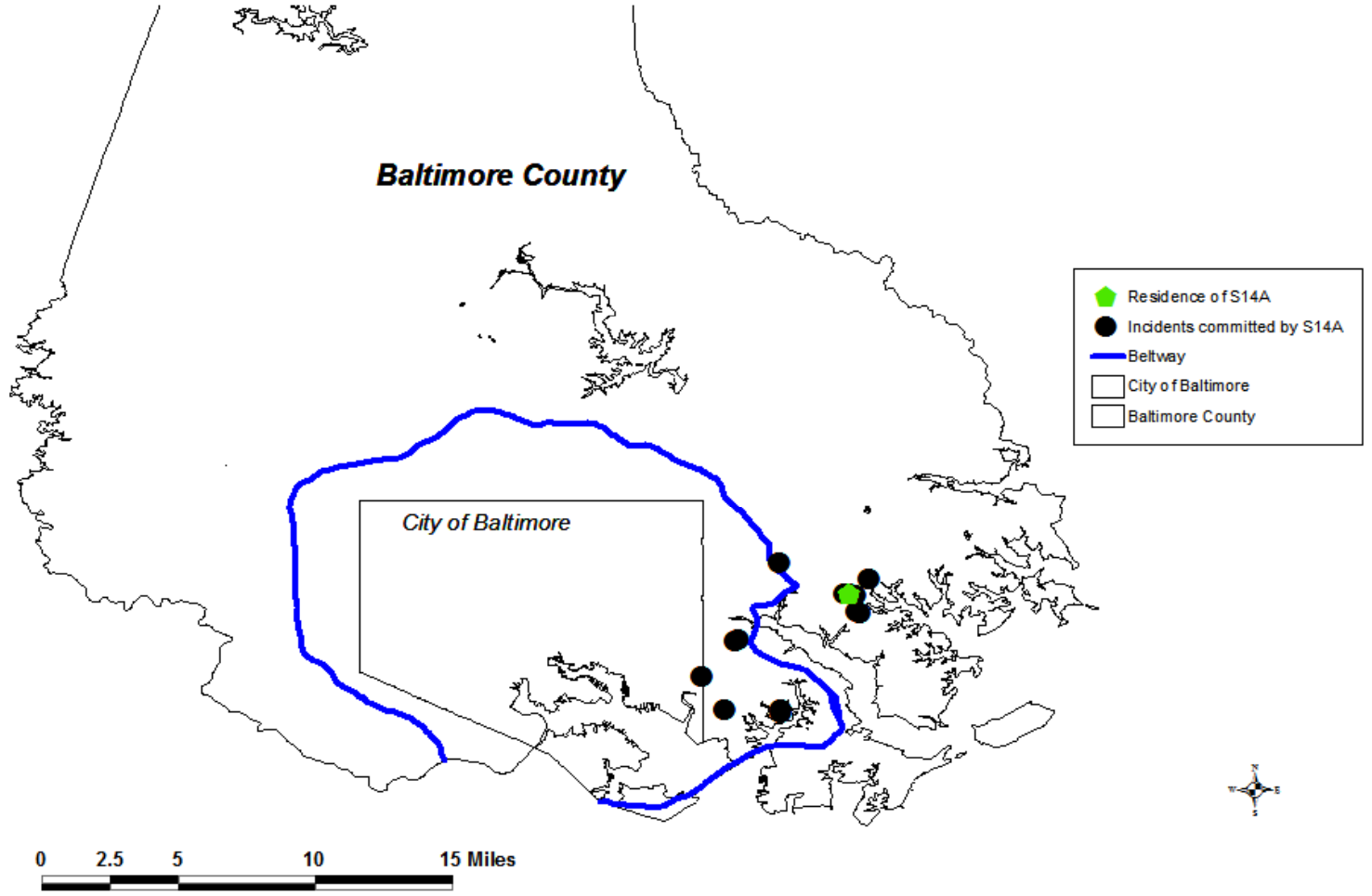


Figure 14.8 shows the results of the Jtc probability output. In this map and the following maps, the bins represent probability ranges of 0.0001. The cell with the highest likelihood is highlighted in light blue. As seen, this cell is very close to the cell where the actual offender lived. The distance between the two cells was 0.34 miles. With the Jtc probability estimate, however, the area with a higher probability (dark red) covers a fairly large area. However, the precision of the Jtc estimate is good since only 0.03% of the cells have higher probabilities that the cell associated with the area where the offender lived. In other words, the Jtc estimate has produced a very good estimate of the location of the offender, as might be expected given the concentration of the incidents committed by this person.

For this same offender, Figure 14.9 show the results of the conditional probability estimate of the offender's residence location, that is the distribution of the likely origin based on the origins of offenders who committed crimes in the same locations as that by S14A. Again, the cell with the highest probability is highlighted (in light green). As seen, this method has also produced a fairly close estimate, with the distance between the cell with the highest probability and the cell where the offender actually lived being 0.18 miles, about half the error distance of the Jtc method. Further, the conditional estimate is more precise than the Jtc with only 0.01% of the cells having a higher probability than the cell associated with the residence of hte offender. Thus, the conditional probability estimate is not only more accurate than the Jtc method, but also more precise (i.e., more efficient in terms of search area).

For this same offender, Figure 14.10 shows the results of the Bayesian product estimate, the product of the Jtc probability and the conditional probability re-scaled to be a single probability (i.e., with the sum of the grid cells equal to 1.0). It is a Bayesian estimate because it updates the Jtc probability estimate with the information on the likely origins of offenders who committed crimes in the same locations (the conditional estimate). Again, the cell with the highest probability is highlighted (in dark tan). The distance error for this method is 0.26 miles, not as accurate as the conditional probability estimate but more accurate than the Jtc estimate. Further, this method is about as precise as the Jtc since 0.03% of the cells having probabilities higher than that associated with the location where the offender lived.

Figure 14.11 shows the results of the Bayesian Risk probability estimate. This method takes the Bayesian product estimate and divides it by the general origin probability estimate. It is analogous to a *risk* measure that relates the number of events to a baseline population. In this case, it is the estimate of the probability of the updated Jtc estimate relative to the probability of where offenders live in general. Again, the cell with the highest likelihood is highlighted (in dark yellow). The Bayesian Risk estimate produces an error of 0.34 miles, the same as the Jtc estimate, with 0.04% of the cells having probabilities higher than that associated with the residence of the offender.

Figure 14.8:

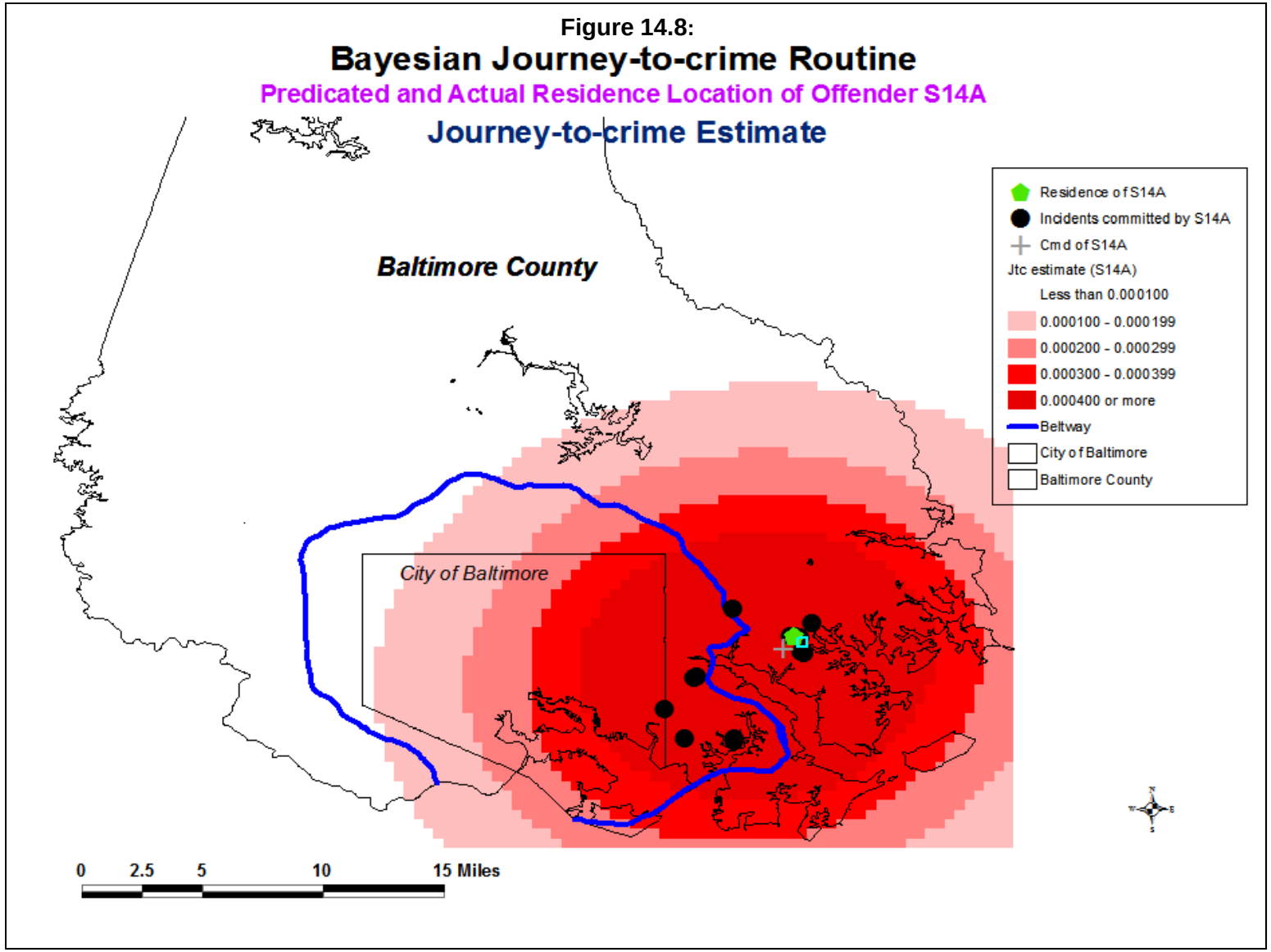

Bayesian Journey-to-crime Routine
Predicated and Actual Residence Location of Offender S14A
Journey-to-crime Estimate
Baltimore County
City of Baltimore
Residence of S14A
Incidents committed by S14A
Cmd of S14A
Jtc estimate (S14A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
Beltway
City of Baltimore
Baltimore County
0
2.5
5
10
15 Miles

Figure 14.9:

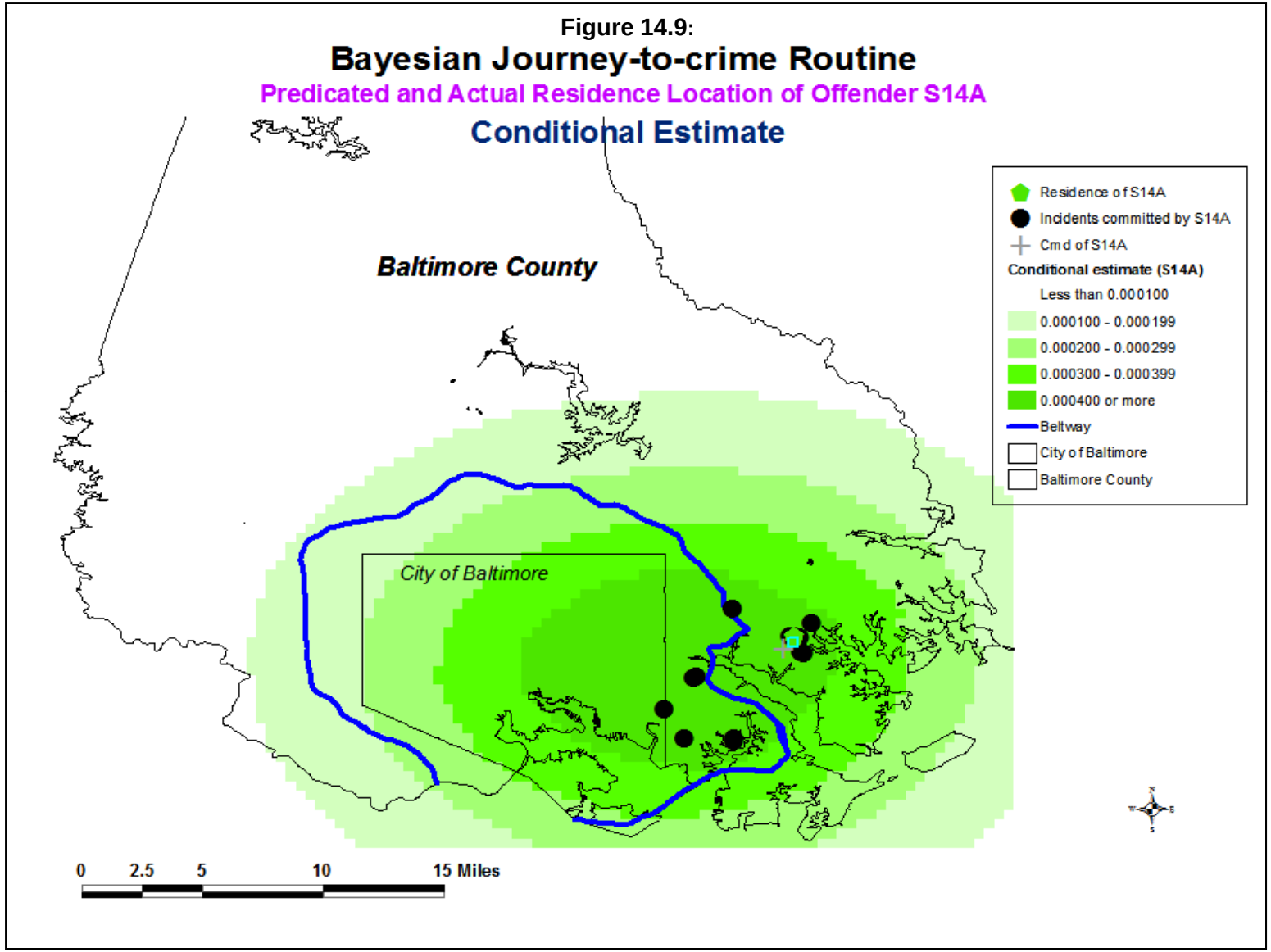

Bayesian Journey-to-crime Routine
Predicated and Actual Residence Location of Offender S14A
Conditional Estimate
Baltimore County
City of Baltimore
Residence of S14A
Incidents committed by S14A
Cmd of S14A
Conditional estimate (S14A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
Beltway
City of Baltimore
Baltimore County
0
2.5
5
10
15 Miles

Figure 14.10:

Bayesian Journey-to-crime Routine

Predicated and Actual Residence Location of Offender S14A

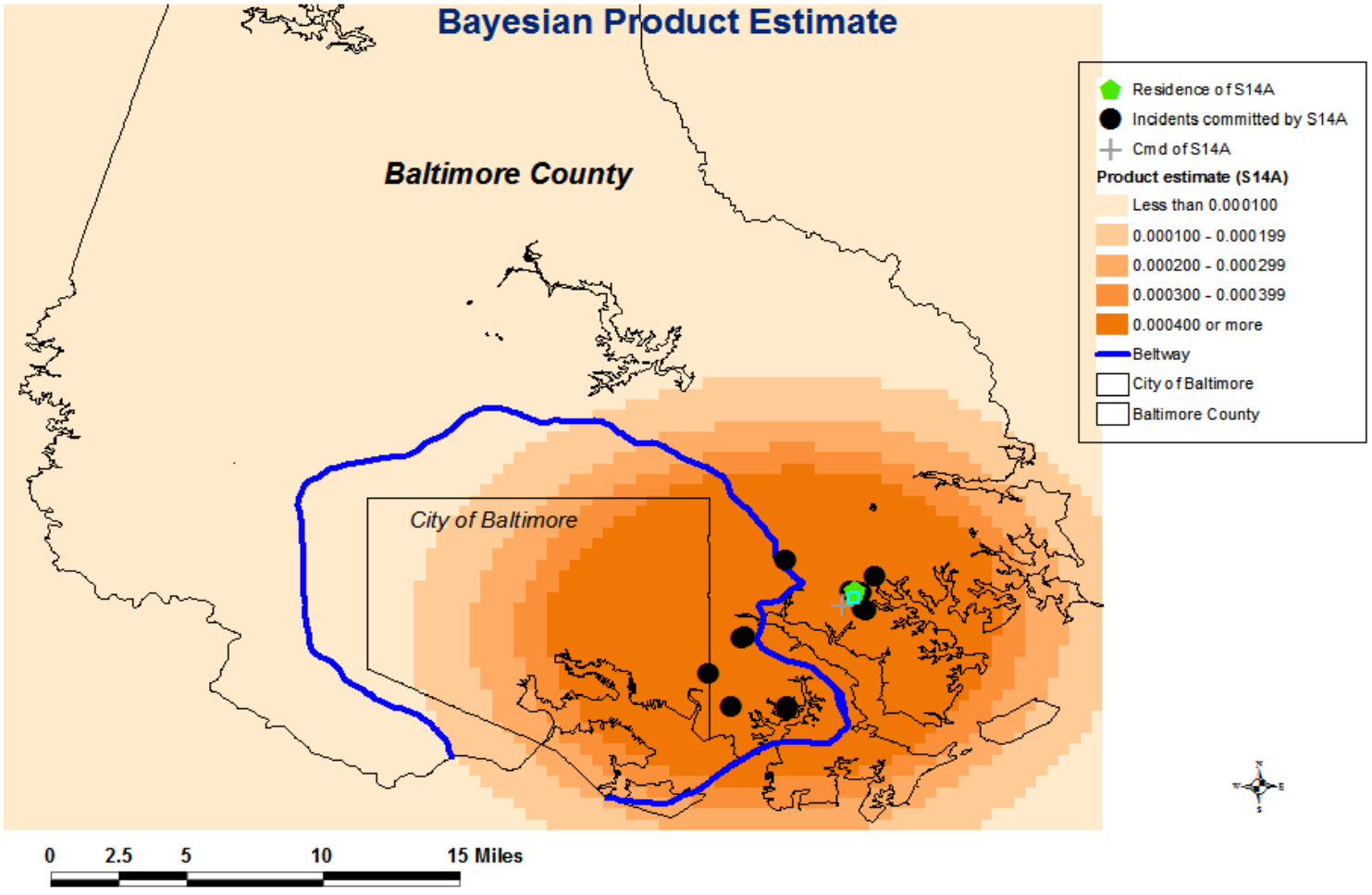
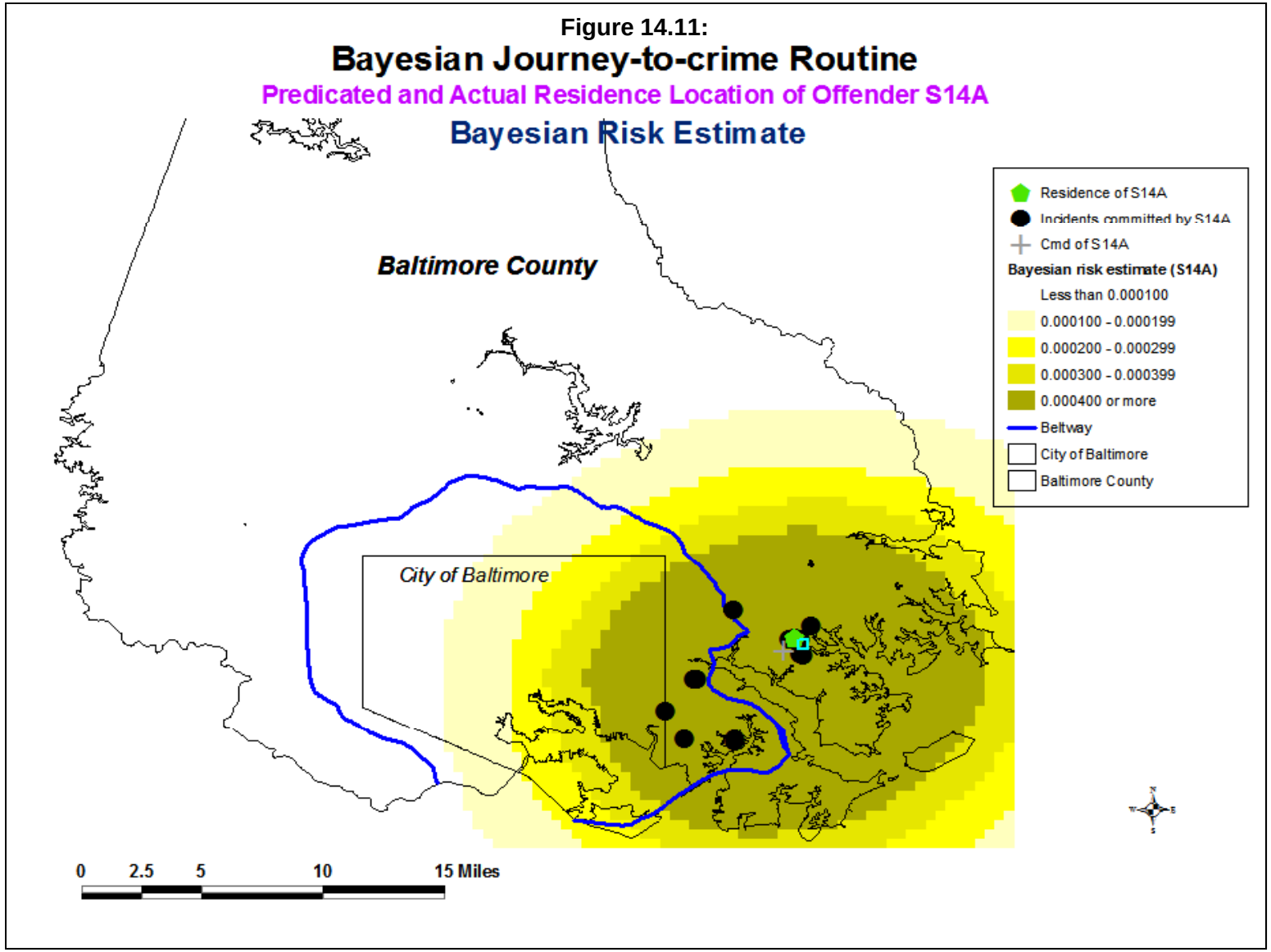


Figure 14.11:


Bayesian Journey-to-crime Routine
Predicated and Actual Residence Location of Offender S14A
Bayesian Risk Estimate
Baltimore County
City of Baltimore
Residence of S14A
Incidents committed by S14A
Cmd of S14A
Bayesian risk estimate (S14A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
Beltway
City of Baltimore
Baltimore County
0
2.5
5
10
15 Miles

Finally, the center of minimum distance (Cmd) is indicated on each of the maps with a grey cross. In this case, the Cmd is not as accurate as any of the other methods since it has an error distance of 0.58 miles.

In summary, all of the Journey-to-crime estimate methods produced relatively accurate estimates of the location of the offender (S14A). Given that the incidents committed by this person were within a fairly concentrated pattern, it is not surprising that each of the method produced reasonable accuracy. In Canter and Larkin's (1994) terminology, this offender is a 'marauder'.

Offender TS15A

But what happens if an offender who did not commit crimes in the same part of town is selected, what Canter and Larkin (1994) call a 'commuter'? Figure 14.12 shows the distribution of an offender who committed 15 offenses (TS15A). Of the 15 offenses committed by this individual, there were six larceny thefts, two assaults, two vehicle thefts, one robbery, one burglary, and three incidents of arson. Twelve of the offenses are relatively concentrated but two are more than eight miles away.

Only three of the estimates will be shown. The general method produces an error of 4.6 miles. Figure 14.13 show the results of the Jtc method. Again, the map bins are in ranges of 0.0001 and the cell with the highest probability is highlighted. As seen, the cell with the highest probability is located north and west of the actual offender's residence. The error distance is 1.89 miles. The precision of this estimate is good with only 0.08% of the cells having higher probabilities than the cell where the offender lived.

Figure 14.14 show the result of the conditional probability estimate for this offender. In this case, the conditional probability method is less accurate than the Jtc method with an error distance between the cell with the highest probability and the cell where the offender lived being 2.39 miles. However, this method is less precise than the Jtc method with 1.6% of the study area having probabilities higher than that in the cell where the offender lived.

Finally, Figure 14.15 shows the results of the product probability estimate. For this method, the error distance is only 0.47 miles, much less than the Jtc method. Further, it is smaller than the CMD which has an error distance of 1.33 miles. Again, updating the Jtc estimate with information from the conditional estimate produces a more accurate guess where the offender lives. Further, the product estimate is more precise with only 0.02% of the study area having probabilities higher than the cell covering the area where the offender lived.

Figure 14.12:

Bayesian Journey-to-crime Routine

Predicated and Actual Residence Location of Offender TS15A

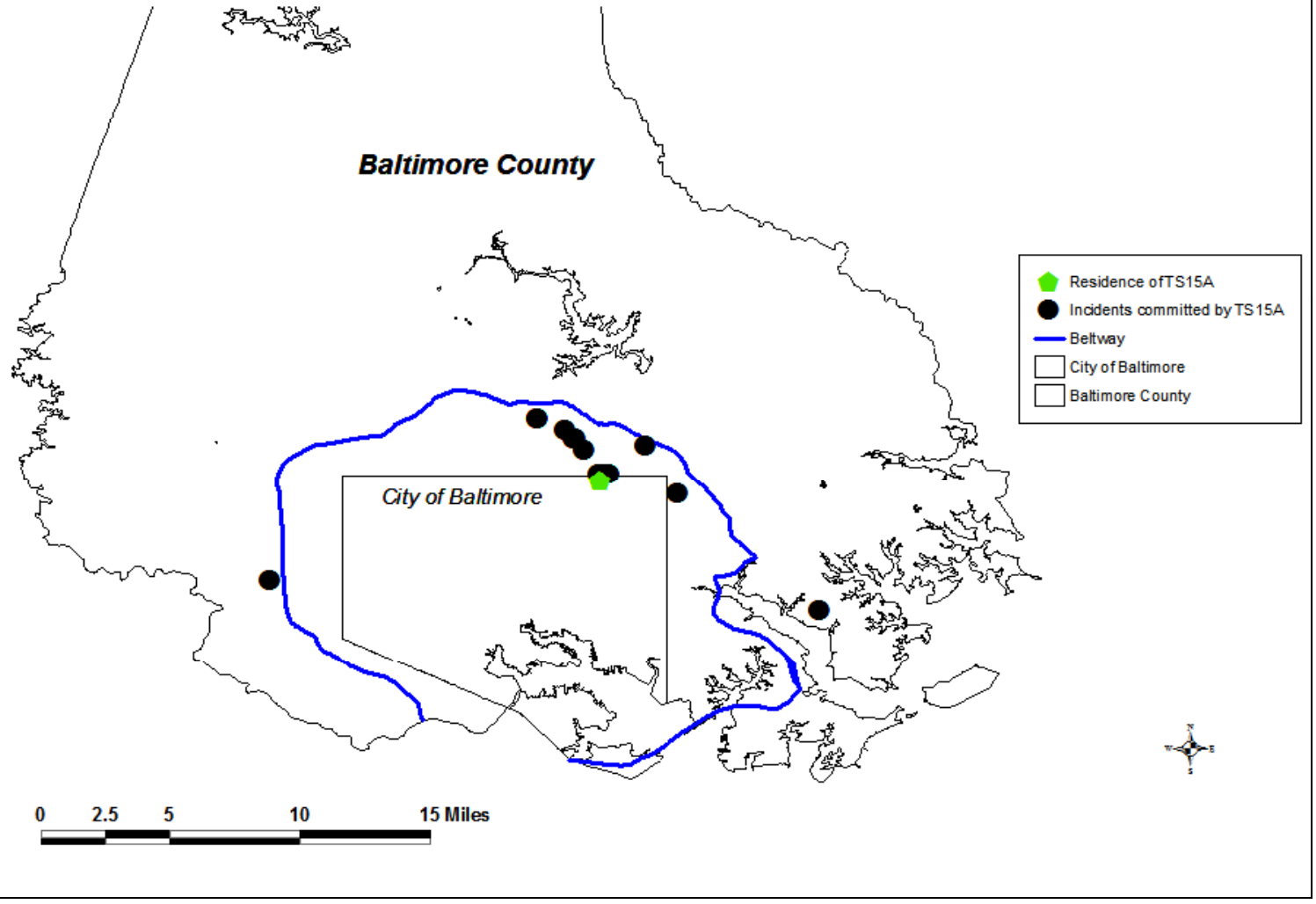


Figure 14.13:

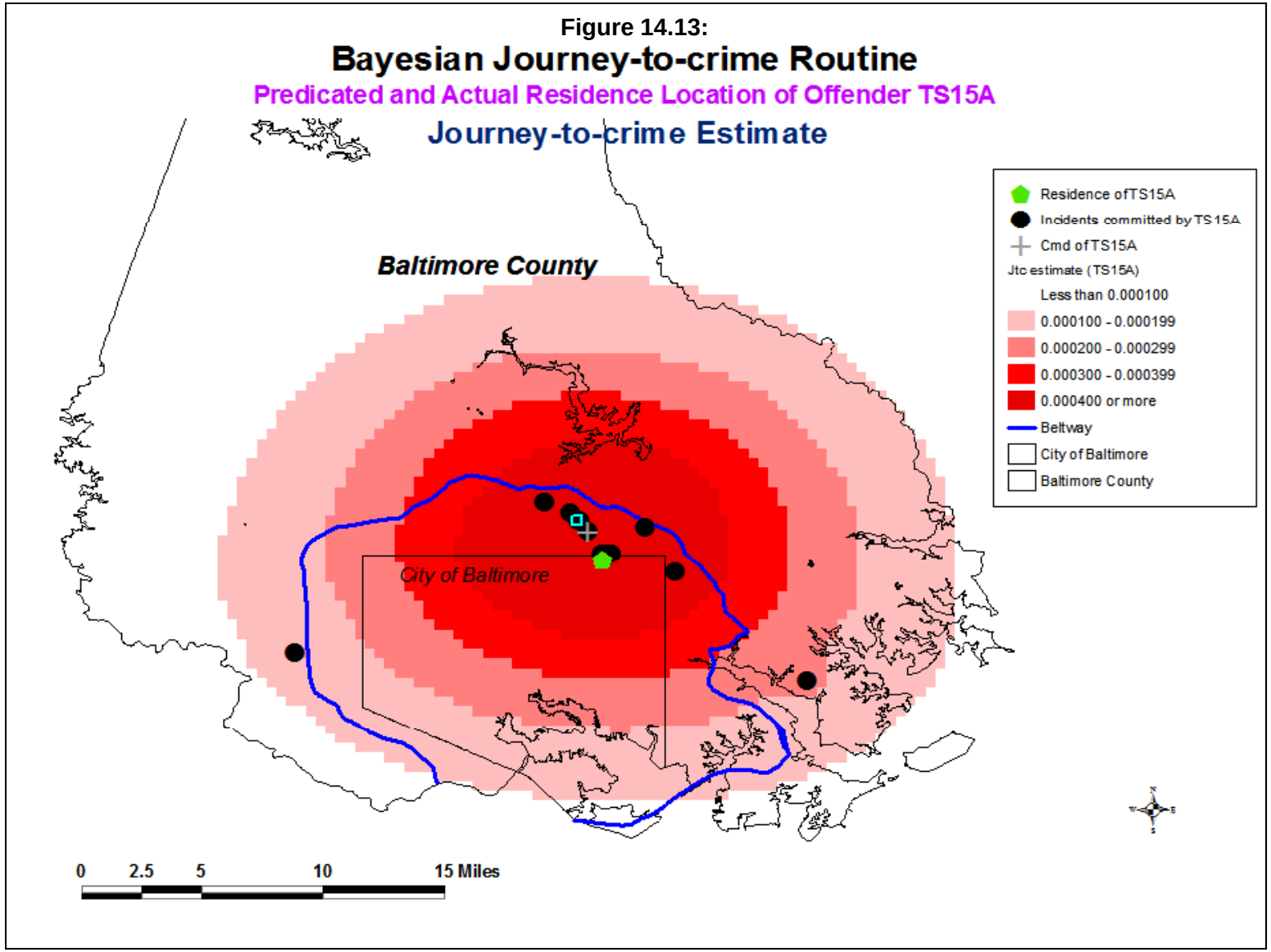

Bayesian Journey-to-crime Routine
Predicated and Actual Residence Location of Offender TS15A
Journey-to-crime Estimate
Baltimore County
City of Baltimore
Residence of TS15A
Incidents committed by TS15A
Cmd of TS15A
Jtc estimate (TS15A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
Beltway
City of Baltimore
Baltimore County
0
2.5
5
10
15 Miles

Figure 14.14:

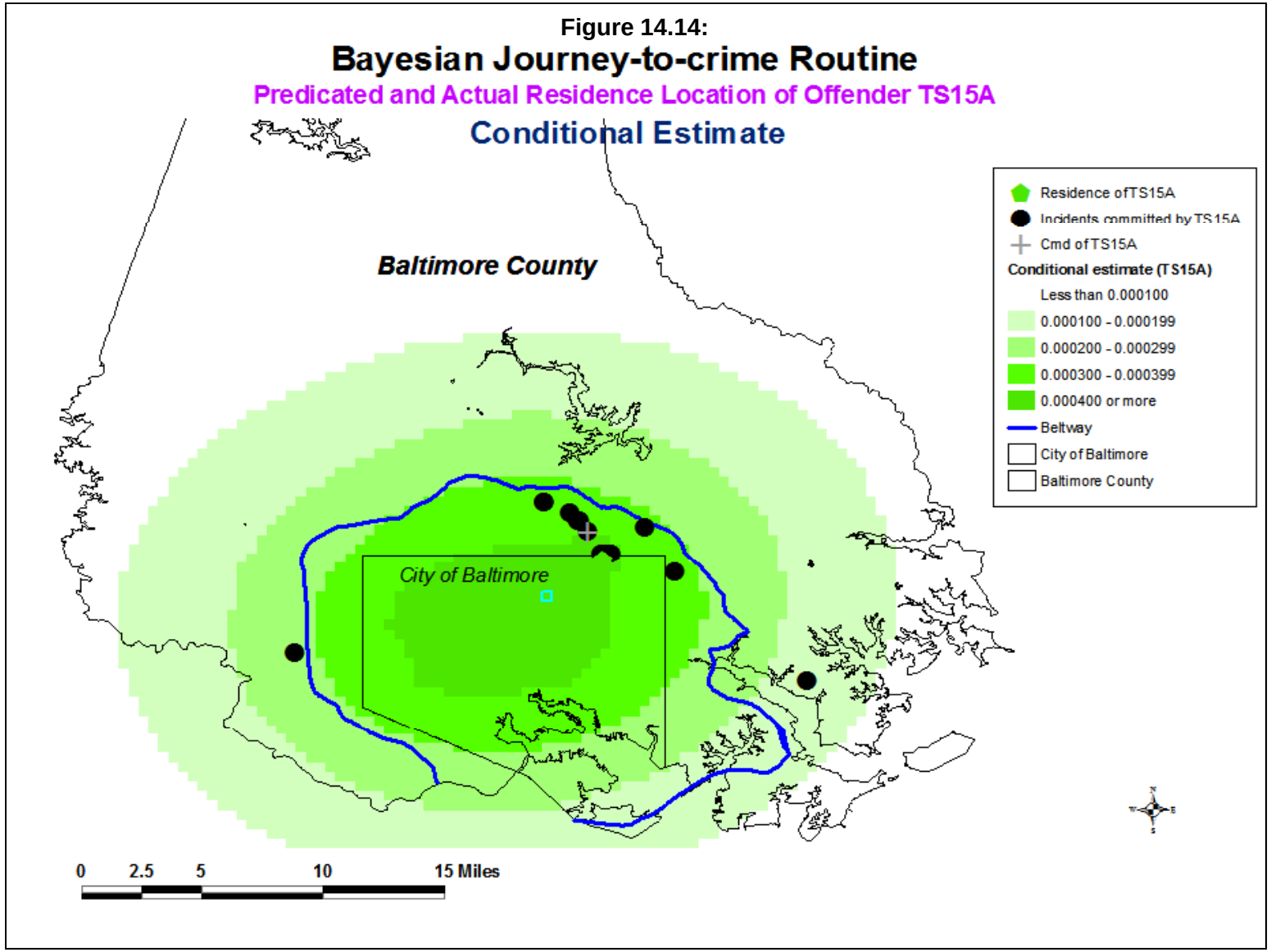
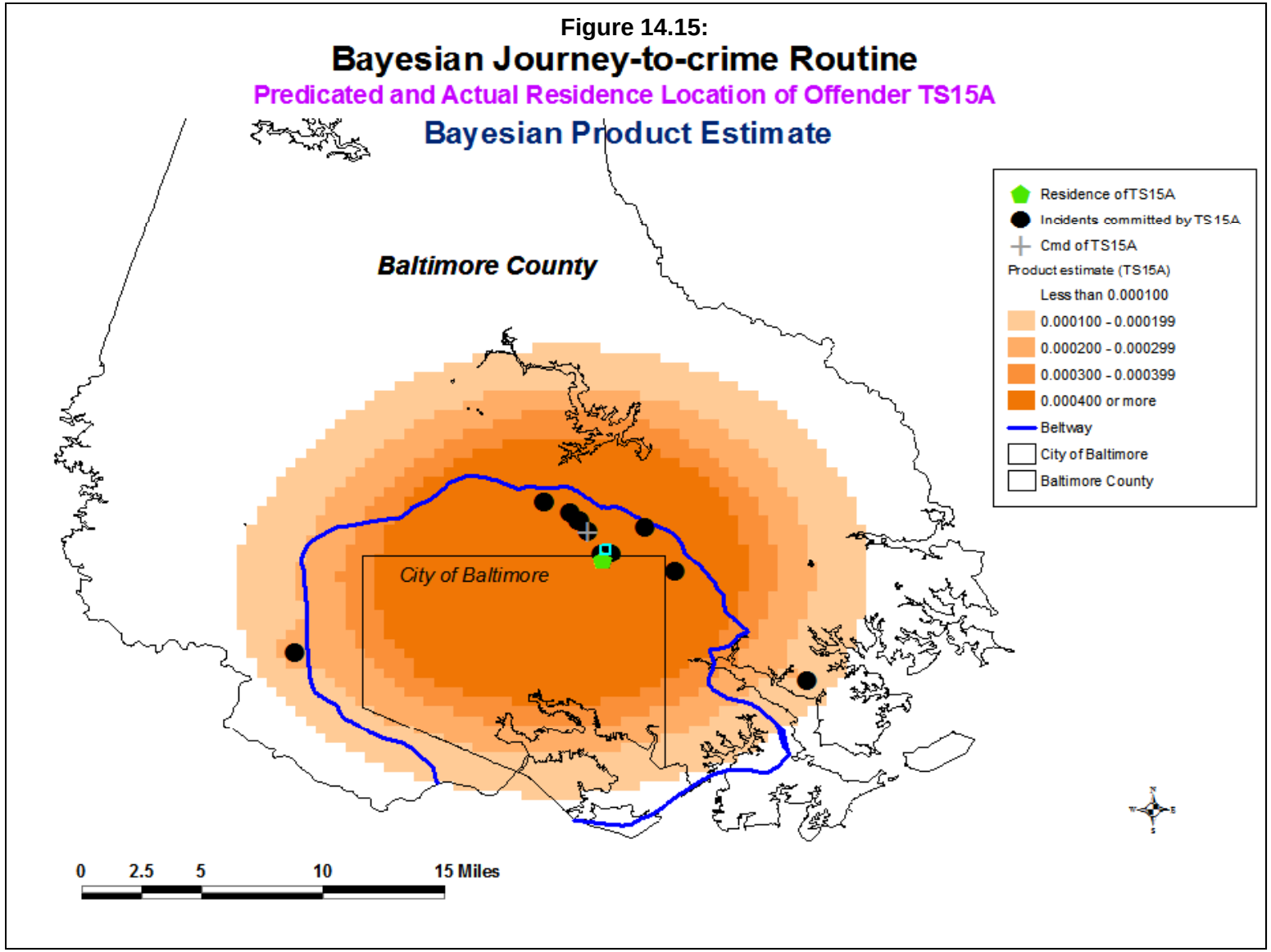

Bayesian Journey-to-crime Routine
Predicated and Actual Residence Location of Offender TS15A
Conditional Estimate
Baltimore County
City of Baltimore
Residence of TS15A
Incidents committed by TS15A
Cmd of TS15A
Conditional estimate (TS15A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
Beltway
City of Baltimore
Baltimore County
0
2.5
5
10
15 Miles

Figure 14.15:


Bayesian Journey-to-crime Routine
Predicated and Actual Residence Location of Offender TS15A
Bayesian Product Estimate
Baltimore County
City of Baltimore
Residence of TS15A
Incidents committed by TS15A
Cmd of TS15A
Product estimate (TS15A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
Beltway
City of Baltimore
Baltimore County
0
2.5
5
10
15 Miles

In other words, the BJtc routine allows the estimation of a probability grid based on a **single** selected method. The user must decide which probability method to select and the routine then calculates that estimate and assigns it to a grid. As mentioned above, the BJtc Diagnostics routine should be first run to decide on which method is most appropriate for the jurisdiction in question. In these 88 cases, the Bayesian product estimate was the most accurate of all the probability methods. However, differences in the balance between central-city and suburbs, the road network, and land uses may change the travel patterns of offenders. So far, as mentioned above, in tests in four cities (Baltimore County, Chicago, the Hague, Manchester), the product estimate has consistently been better than the Journey-to-crime estimate and almost as good, if not better, than the center of minimum distance. Further, the product term appears to be more precise than the Journey-to-crime method though in the Hague study, the conditional estimate was more accurate than the product estimate. The center of minimum distance, while generally more accurate than other methods, has no probability distribution; it is simply a point. Consequently, one cannot select a search area from the estimate.

Potential to Add New Information to Improve the Methodology

Further, it should be possible to add more information to this framework to improve the accuracy and precision of the estimates. One obvious dimension that should be added is an opportunity matrix, a distribution of targets that are crime attractions for offenders. Among these are convenience stores, shopping malls, parking lots, and other types of land use that attract offenders. It will be necessary to create a probability matrix for quantifying these attractions. Further, the opportunity matrix would have to be conditional on the distribution of the crimes and on the distribution of origins of offenders who committed crimes in the same location. The Bayesian framework is a conditional one where factors are added to the framework but conditioned on the distribution of earlier factors:

$$P(Jtc|O) \propto P(Jtc) * P(O|Jtc) * P(A|O,Jtc) \qquad (14.13)$$

where A is the attractions (or opportunities), Jtc is the distribution of incidents, and O is the distribution of other offender origins. It will not be an easy task to estimate an opportunity matrix that is conditioned (dependent) upon both the distribution of offences (Jtc) and the origin of other offenders who committed crimes in the same location (O|Jtc) and it may be necessary to approximate this through a series of filters.

Probability Filters

A filter is a probability matrix that is applied to the estimate but is not conditioned on the existing variables in the model. For example, an opportunity matrix that was independent of

the distribution of offences by a single serial offender or the origins of other offenders who committed crimes in the same locations could be applied as an alternative (equation 14.14):

$$P(Jtc|O) \propto P(Jtc) * P(O|Jtc) * P(A) \quad (14.14)$$

In this case, P(A) is an independent matrix. Another filter that could be applied is residential land use. The vast majority of offenders are going to live in residential areas. Thus, a residential land use filter estimates the probability of a residential land use for every cell, P(R), could be applied to screen out cells that are not residential, such as

$$P(Jtc|O) \propto P(Jtc) * P(O|Jtc) * P(A) \quad (14.15)$$

In this way, additional information can be integrated into the Journey-to-crime methodology to improve the accuracy and precision of the estimates. Clearly, having additional variables be conditioned upon existing variables in the model would be ideal since that would fit the true Bayesian approach. But, even if independent filters were brought in, the model might be improved.

Defining Filters in the Bayesian Journey-to-crime Routine

The Bayesian Journey-to-crime routine allows filters to be applied. The routine can be run with or without filters and the user has a choice of running the routine with no filters, one filter (called 'Filter 1') and two filters (called 'Filter 1' and 'Filter 2'). See Figure 14.4 above that illustrates how to define the filters on the Bayesian Journey-to-crime page.

For example, one filter could be whether the grid cell is residential or not. Each zone in the filter variable could have a dummy variable indicating whether it is primarily residential (1) or not (0). The criteria for defining residential could be having a minimum number of residential units but also having less than a specified number of persons employed in the zone. A 'pure' residential zone would have only residences and no employment.

Kent and Leitner (2009) showed that the use of residential land covers improved accuracy for Jtc estimates, but did not improve estimates for the Bayesian approach. Nevertheless, it is likely that a subset of residential land cover might improve the precision of an estimate.

Another filter could be the amount of employment in a zone. Zones with many employees (e.g., commercial areas) would have high values on the filter variable while zones with few, if any, employees would have low values on the filter.

A third filter could be the number of businesses of a certain type for crimes of a particular type. For example, to model liquor store robberies, the filter variable could be the number of liquor stores in each zone. Or, to model bank robberies, the filter variable could be the number of banks in each zone.

Whichever variable is used for the filter, the routine interpolates this to the same grid as the Journey-to-crime function, P(O), the conditional probability function, P(O|Jtc), the general function, P(O), the production probability function, P(Jtc)*P(O|Jtc), and the Bayesian Risk function, $\frac{P(Jtc)*P(O|Jtc)}{P(O))}$.

The interpolated filter grid is then multiplied by four functions - the Journey-to-crime grid, P(Jtc), the conditional probability function, P(O|Jtc), the product probability function, P(Jtc)*P(O|Jtc), and the Bayesian Risk function, $\frac{P(Jtc)*P(O|Jtc)}{P(O))}$.

Example of the Use of a Probability Filter

To illustrate this, Figure 14.16 shows the location of 22 crimes committed by a single offender, S22A, and the offender's residence when arrested. The incidents are shown in blue and the residence location in black. The crimes committed were 6 commercial burglaries, 1 residential burglary, 11 vehicle break-ins and 4 vehicle thefts.

Figure 14.17 shows the result of the Journey-to-crime probability estimate. As with the other maps, the center of minimum distance (CMD) is shown as a gray cross. Notice that neither the center of minimum distance nor the journey-to-crime estimates were particularly accurate as the cell with the peak probability was 2.5 miles and the CMD was 2.6 miles respectively away from the actual home location of the offender.

Figure 14.18 shows the conditional probability estimate. With this estimate, the cell with the peak probability was much more accurate, being about 0.5 miles away.

Figure 14.19 shows the product probability estimate which multiplies the Journey-to-crime estimate by the conditional probability estimate and then re-scales the grid to sum to 1.0. This estimate was not particularly accurate as well with the cell having the peak probability being 2.5 miles away. The reason is that the inaccurate Journey-to-crime estimate also made the product estimate inaccurate. In some cases, a more accurate conditional estimate will improve the product probability but in other cases it will not. Block and Bernasco (2009) found that journey-to-crime estimates were inaccurate with serial burglars in The Hague, Netherlands, and that the conditional probability estimate was more accurate than the product probability estimate because the poor journey-to-crime estimates degraded the product estimates. That is why it is

Figure 14.16:

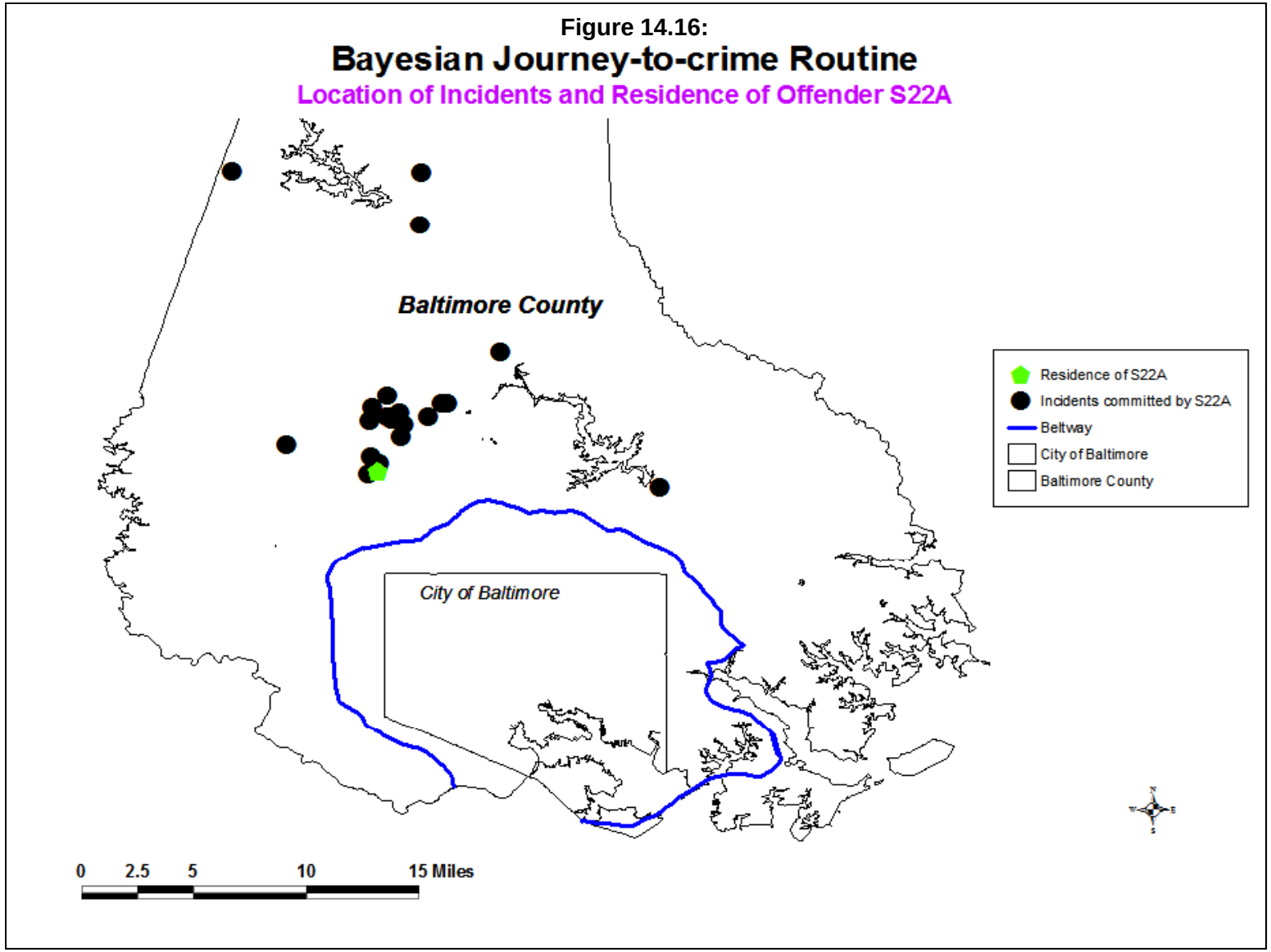

Bayesian Journey-to-crime Routine
Location of Incidents and Residence of Offender S22A
Baltimore County
City of Baltimore
Residence of S22A
Incidents committed by S22A
Beltway
City of Baltimore
Baltimore County
0 2.5 5 10 15 Miles

Figure 14.17:

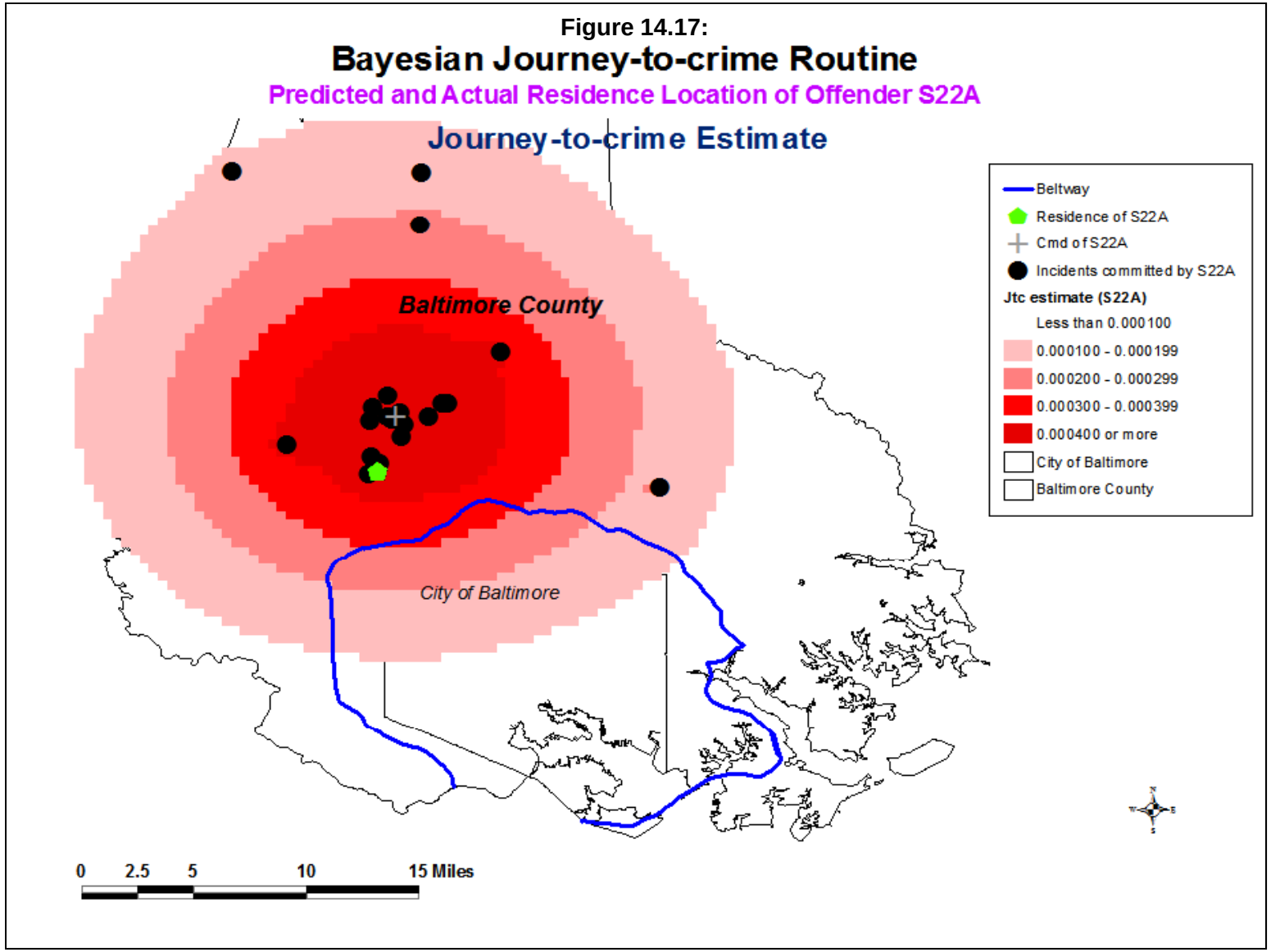

Bayesian Journey-to-crime Routine
Predicted and Actual Residence Location of Offender S22A
Journey-to-crime Estimate
Baltimore County
City of Baltimore
Beltway
Residence of S22A
Cmd of S22A
Incidents committed by S22A
Jtc estimate (S22A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
City of Baltimore
Baltimore County
0
2.5
5
10
15 Miles

Figure 14.18:

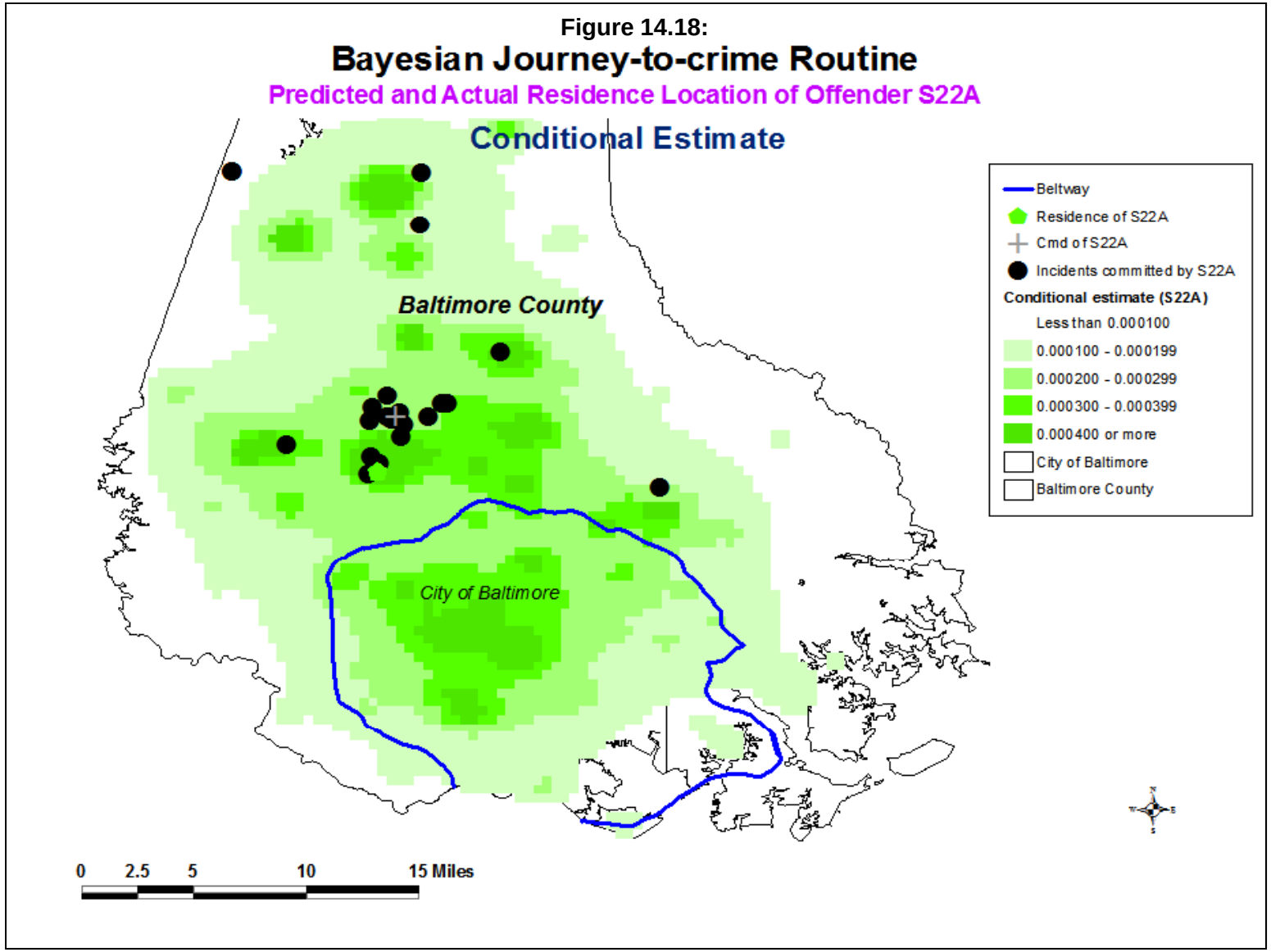
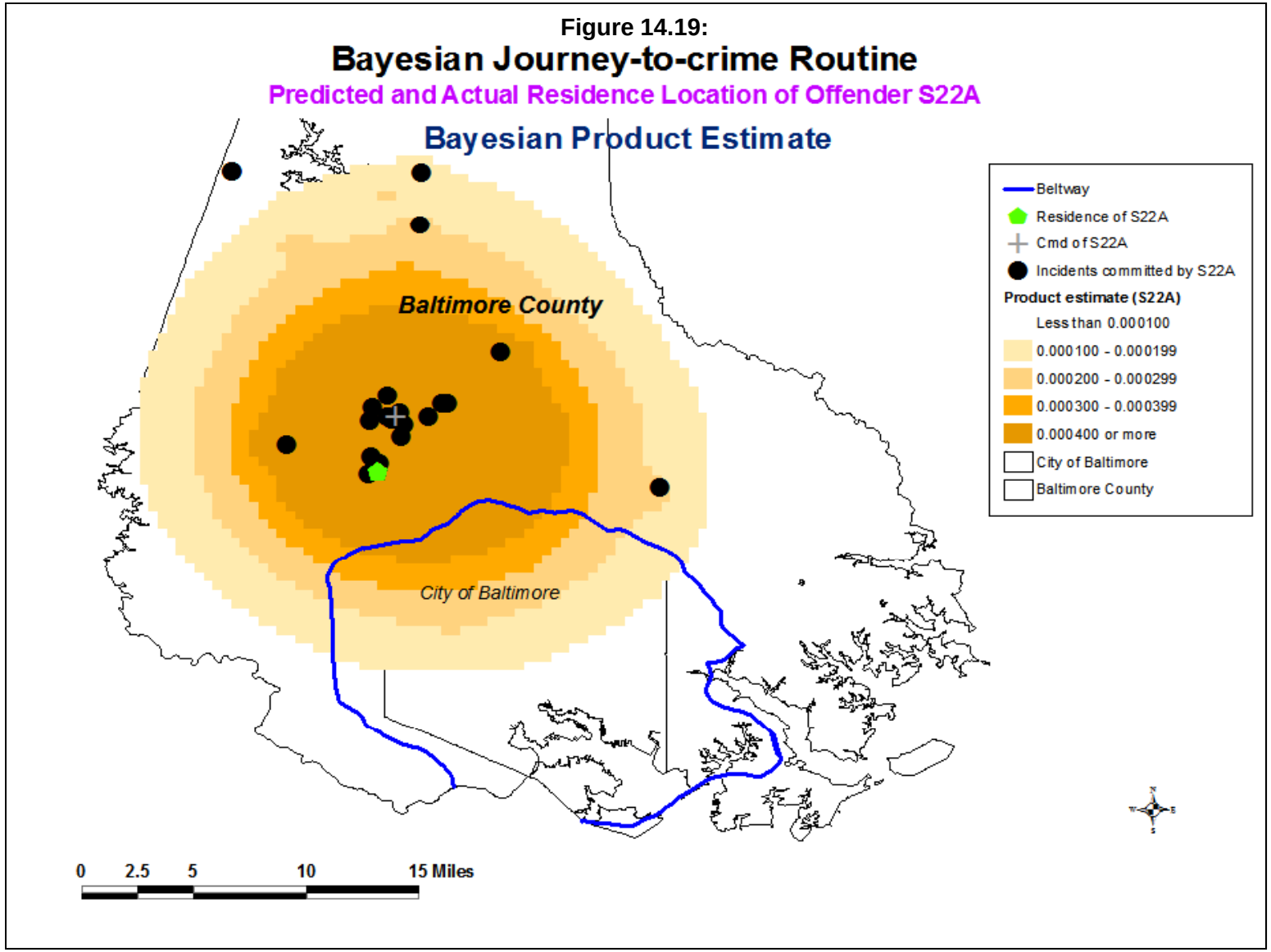

Bayesian Journey-to-crime Routine
Predicted and Actual Residence Location of Offender S22A
Conditional Estimate
Baltimore County
City of Baltimore
Beltway
Residence of S22A
Cmd of S22A
Incidents committed by S22A
Conditional estimate (S22A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
City of Baltimore
Baltimore County
0 2.5 5 10 15 Miles

Figure 14.19:


Bayesian Journey-to-crime Routine
Predicted and Actual Residence Location of Offender S22A
Bayesian Product Estimate
Baltimore County
City of Baltimore
Beltway
Residence of S22A
Cmd of S22A
Incidents committed by S22A
Product estimate (S22A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
City of Baltimore
Baltimore County
0 2.5 5 10 15 Miles

important to analyze which method is best for a single jurisdiction using the Bayesian journey-to-crime diagnostics routine before applying a particular method a single serial offender.

With offender S22A, a residential land use probability filter was defined by a zonal data base of Traffic Analysis Zones (TAZ) in Baltimore County that included both residential population variables and employment variables. A dummy variable was created by defining TAZ's that had 100 or more persons living in them but 200 or fewer employees working in them. Thus, these TAZ's were primarily residential. When the TAZ layer was interpolated to the grid in the routine, each grid cell had a probability value that varied from 0 to 1 and which indicated the likelihood of the cell residential.

Figure 14.20 shows the result of combining the product probability estimate with this residential land use filter. The results were as accurate as the conditional probability estimate in distance as the cell with the peak probability was about 0.5 miles away. But, more important is the probability estimate in the cell with the peak probability was much higher than the probability estimated for the conditional (0.008842 compared to 0.000345, a ratio that was 25.6 times higher).

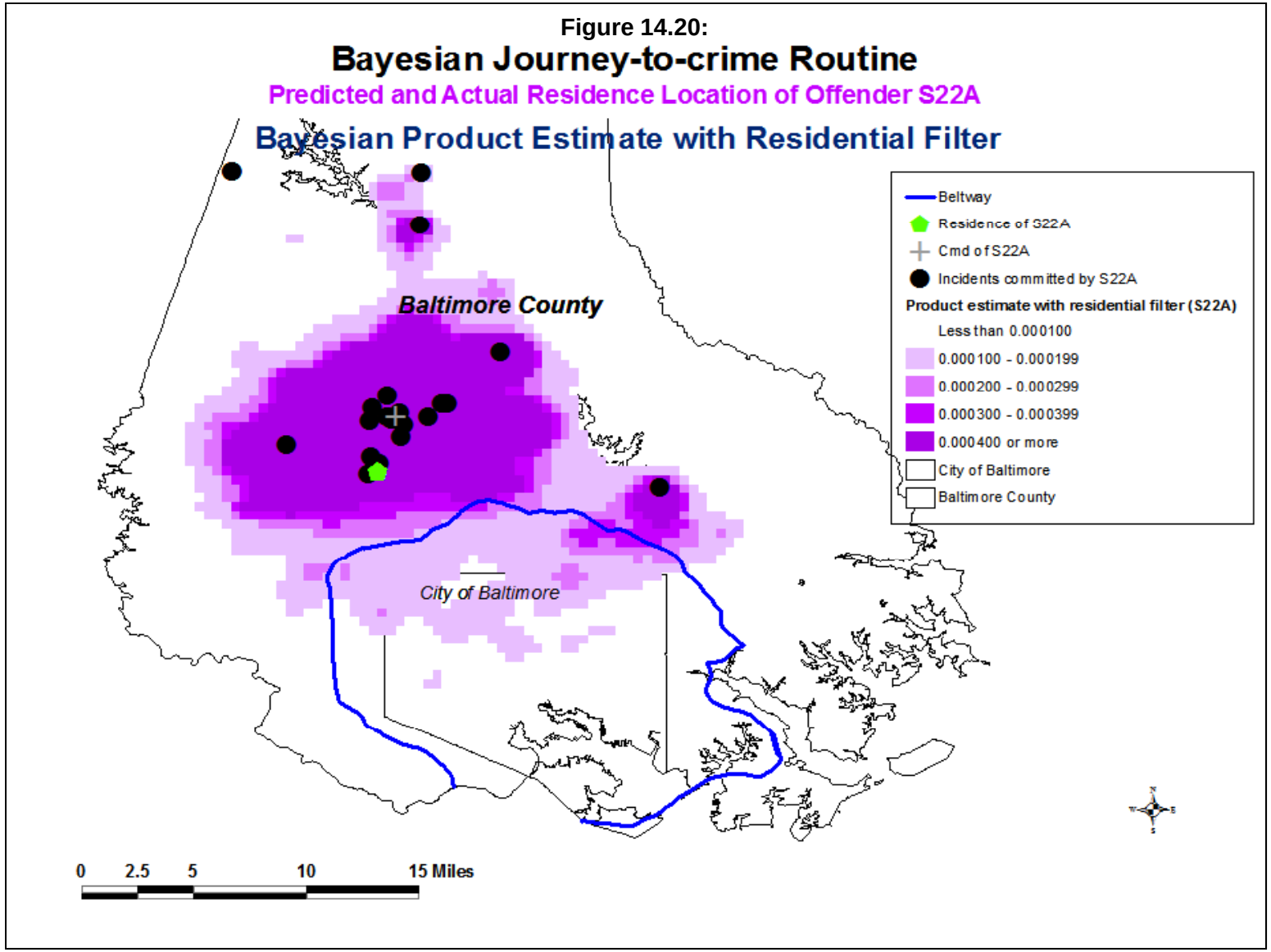
In other words, the effect of narrowing the probability estimates of the product probability by discounting cells that were not residential actually improved the accuracy of the product probability estimate. We do not yet know whether using a residential filter will always improve accuracy since we have not tested it on a number of cases yet. It is possible that these filters will improve accuracy but it is also possible that they will make precision worse since they multiply a conditional probability by a matrix that is constant for all offenders.

Until a thorough evaluation is conducted, the filters are provided as tools for users to experiment with in modeling the likely residence location of a serial offender.

Guidelines for Analysts

The following discussion is for analysts wishing to utilize the technique to try to narrow down the geographic areas for particular serial offenders. The hardest part of using the technique is collecting the data and constructing a journey-to-crime (Jtc) function and an origin-destination (O-D) matrix. However, once the data have been assembled and the Jtc function and the O-D matrix constructed, the technique can be used for multiple serial offenders. These estimates need to be only updated every few years in order to account for changes in travel patterns by offenders.

Figure 14.20:


Bayesian Journey-to-crime Routine
Predicted and Actual Residence Location of Offender S22A
Bayesian Product Estimate with Residential Filter
Baltimore County
City of Baltimore
Beltway
Residence of S22A
Cmd of S22A
Incidents committed by S22A
Product estimate with residential filter (S22A)
Less than 0.000100
0.000100 - 0.000199
0.000200 - 0.000299
0.000300 - 0.000399
0.000400 or more
City of Baltimore
Baltimore County
0 2.5 5 10 15 Miles

We have argued that an analyst should test which of the different methods produces the best estimate for a particular jurisdiction. However, if an analyst wants to choose a single best technique without testing which method works best in the jurisdiction, we recommend sticking with the Product probability by itself (without filters). We have found that the Product estimate (the product of the Jtc and Conditional probability estimates) generally produces more accurate results than the Jtc function by itself or the Conditional probability by itself, though some exceptions have been noted. The use of filters to improve estimates is still too new a technique and needs to be evaluated further.

To simplify, seven basic steps are required to run the "Estimate likely location of a serial offender" routine and one additional step if the analyst wants to test which method works best for the jurisdiction.

Steps

1. Obtain Required Data

First, the data that will be needed is a large number of records where both the residence location and the crime location are known. Most likely, these will come from arrest records. By large, we mean at least 10,000 cases.

2. Construct Journey-to-crime Function

Second, once these data have been assembled, the user should create a journey-to-crime function using the "Calibrate journey-to-crime function" routine (discussed in Chapter 13).

3. Define Zonal Framework

Third, to construct an origin-destination matrix, the analyst will need a zonal framework for allocating the incidents to both origin and destination locations. Commonly used zones are census tracts or traffic analysis zones, though others can be used. Also, we have found good results by using a grid as the zone structure, especially with small-sized grid cells (e.g., a 100 column x 100 row grid). The single kernel density interpolation tool (discussed in Chapter 10) is a useful tool for creating a grid overlay that can be used as a zone framework.

4. Construct Origin-Destination Matrix

Fourth, using the data on offenders where both the residence location and the crime location are known and the zonal framework, the origin-destination matrix can be constructed using the "Calculate observed origin-destination trips" routine that is discussed in Chapter 28.

The routine reads in an origin file (the zonal framework) and a destination file (also the zonal framework) and a data file (the set of records of offenders where both residence and crime location are known) and outputs the O-D matrix. The user should save the matrix as a dbf file.

5. Input Jtc Function and O-D Matrix

Fifth, the Jtc function and the O-D matrix are input on the Bayesian Journey-to-crime page.

6. Input Records of Single Serial Offender

Sixth, to estimate the likely origin (residence) location of a serial offender, the records for that serial offender need to be input as the Primary File. As with all Primary File inputs, the coordinate system and the data metrics need to be defined.

7. Estimate Likely Origin Location of a Serial Offender

Seventh, and finally, the user selects one estimation method on the Bayesian Journey-to-crime page and runs the routine. As mentioned above, unless there is contrary information, we recommend using the Product estimate by itself ("Use product of P(Jtc) and P(O|Jtc) estimate").

8. (Optional) Evaluate which Method Produces the Best Results for the Jurisdiction

The Product estimate will generally produce good results for medium-to-large cities. However, for small cities, it may not work well and other measures may work better (e.g., the Conditional or the Bayesian Risk estimate). Therefore, an optional strategy is to evaluate which of the methods works best in the jurisdiction.

To do this, the analyst will have to assemble a diagnostics file on multiple serial offenders where the offender has committed multiple offences (e.g., 5 or more) and where both the residence and crime locations are known. The number of offenders included should be as large as possible, at least 50. The four studies mentioned on page 14.30 all used sizeable data sets (60 or more). The reason is that there needs to be sufficient variability to allow the routine to properly estimate the accuracy and precision of each of the methods.

While this requires even more data to be collected, the advantage is that the best estimation method can be determined for the jurisdiction. As mentioned, the use of the Product estimate may or may not produce the best estimates.

Summary

In summary, the Bayesian Jtc methodology is an improvement over the current Journey-to-crime method and appears to be as good, and more useful, than the center of minimum distance. First, it adds new information to the Journey-to-crime function to yield a more accurate and precise estimate. Second, it can sometimes predict the origin of 'commuter'-type serial offenders, those individuals who do not commit crimes in their neighborhoods (Paulsen, 2007; Canter & Larkin, 1994). The traditional Journey-to-crime function cannot predict the origin location of a 'commuter'-type. Of course, this will only work if there are prior offenders who lived in the same location as the serial offender of interest. If the offender lived in a neighborhood where there were no previous serial offenders that were documented in the origin-destination matrix, the Bayesian approach would not detect that location, either.

Caveat

A caveat should be noted, however. The Bayesian method still has a substantial amount of error. Much of this error reflects the inherent mobility of offenders, especially those living in in suburbs outside of central cities. While adolescent offenders, especially juvenile males, tend to commit crimes within a more circumscribed area (Levine & Lee, 2013), the almost-universal ability adults to own automobiles and to travel outside their residential neighborhoods is turning crime into a much more mobile phenomena than it was, say, 50 years ago when only about half of American households owned an automobile.

Thus, the Bayesian approach to Journey-to-crime estimation must be seen as a tool that produces an incremental improvement in accuracy and precision. Geographic profiling is but one tool in the arsenal of methods that police must use to catch serial offenders.

References

Block, R. & Bernasco, W. (2009). Finding a serial burglar's home using distance decay and conditional origin-destination patterns: A test of Empirical Bayes Journey-to-crime estimation in The Hague. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 187-211.

Canter, D. (2009). Developments in geographical offender profiling: Commentary on Bayesian journey-to-crime modeling. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 161-166.

Canter, D. (2003). *Dragnet: A Geographical Prioritisation Package*. Center for Investigative Psychology, Department of Psychology, The University of Liverpool: Liverpool, UK. http://www.i-psy.com/publications/publications_dragnet.php.

Canter, D., Coffey, T., Huntley, M., & Missen, C. (2000). Predicting serial killers' home base using a decision support system. *Journal of Quantitative Criminology, 16* (4), 457 -- 478.

Canter, D. & Gregory, A. (1994). Identifying the residential location of rapists, *Journal of the Forensic Science Society*, 34 (3), 169-175.

Canter, D. & Larkin, P. (1993). The environmental range of serial rapists, *Journal of Environmental Psychology*, 13, 63-69.

Denison, D.G.T., Holmes, C.C. Mallilck, B. K. & Smith, A. F. M. (2002). *Bayesian Methods for Nonlinear Classification and Regression*. John Wiley & Sons, Ltd: New York.

Gelman, A., Carlin, J. B. Stern, H. S. & Rubin, D. B. (2004). *Bayesian Data Analysis* (second edition). Chapman & Hall/CRC: Boca Raton, FL.

Jessen, R. J. (1979). *Statistical Survey Techniques*. John Wiley & Sons: New York.

Kanji, G. K. (1993). *100 Statistical Tests*. Sage Publications: Thousand Oaks, CA.

Kent, J. & Leitner, M. (2009). Utilizing land cover characteristics to enhance journey-to-crime estimation models. *Crime Mapping*, 1.

Lee, P. M. (2004). *Bayesian Statistics: An Introduction* (third edition). Hodder Arnold: London.

References (continued)

Leitner, M. & Kent, J. (2009). Bayesian Journey-to-crime modeling of single- and multiple crime type series in Baltimore County, MD. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 213-236.

Levine, N. (2009). Introduction to the special issue on Bayesian Journey-to-crime modeling. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 167-185.

Levine, N. (2005). *"The evaluation of geographic profiling software: Response to Kim Rossmo's critique of the NIJ methodology".* http://www.nedlevine.com/Response to Kim Rossmo Critique of the GP Evaluation Methodology.May 8 2005.doc

Levine, N. (2000). Journey-to-crime Estimation. Chapter 10 of Levine, N. (ed), *CrimeStat III: A Spatial Statistics Program for the Analysis of Crime Incident Locations* (version 1.1). Ned Levine & Associates, Annandale, VA; National Institute of Justice, Washington, DC. August. See archived files at http://www.icpsr.umich.edu/CrimeStat.

Levine, N. & Lee, P. (2013). Crime travel of offenders by gender and age in Manchester, England. Leitner, M. (ed), *Crime Modeling and Mapping Using Geospatial Technologies,* Springer. 145-178.

Levine, N., & Block, R. (2010). Bayesian Journey-to-Crime Estimation: An Improvement in Geographic Profiling Methodology. *The Professional Geographer*. 63(2), 213-229.

Levine, N. & Lee, P. (2009). Bayesian Journey-to-crime modeling of juvenile and adult offenders by gender in Manchester. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 237-251.

O'Leary, M. (2009). The mathematics of geographical profiling. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 253-265.

Paulsen, D. (2007). Improving geographic profiling through commuter/marauder prediction.. *Police Practice and Research*, 8: 347-357

Paulsen, D. (2006a). Connecting the dots: assessing the accuracy of geographic profiling software. *Policing: An International Journal of Police Strategies and Management*. 20 (2), 306-334.

References (continued)

Paulsen, D. (2006b). Human versus machine: A comparison of the accuracy of geographic profiling methods. *Journal of Investigative Psychology and Offender Profiling* 3: 77-89.

Rich, T., & Shively, M. (2004). *A Methodology for Evaluating Geographic Profiling Software*. Final Report for the National Institute of Justice, Abt Associates: Cambridge, MA. http://www.ojp.usdoj.gov/nij/maps/gp.pdf.

Rossmo, D. K. (2005a). Geographic heuristics or shortcuts to failure?: Response to Snook et al. *Applied Cognitive Psychology* 19: 651-654.

Rossmo, D. K. (2005b). Response to NIJ's methodology for evaluating geographic profiling software. http://www.ojp.usdoj.gov/nij/maps/gp.htm.

Rossmo, D. K., & Filer, S. (2005). Analysis versus guesswork. *Blue Line Magazine*, , August / September, 24:26.

Rossmo, D. K. (2000). *Geographic Profiling*. CRC Press: Boca Raton Fl.

Rossmo, D. K. (1995). Overview: multivariate spatial profiles as a tool in crime investigation. In Block, C. R., Dabdoub, M. & Fregly, S., *Crime Analysis Through Computer Mapping*. Police Executive Research Forum: Washington, DC. 65-97.

Siegel, S. (1956). *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill: New York.

Snook, B., Zito, M., Bennell, C. & Taylor, P. J. (2005). On the complexity and accuracy of geographic profiling strategies. *Journal of Quantitative Criminology*, 21 (1), 1-26.

Snook, B., Taylor, P. J. & Bennell, C. (2004). Geographic profiling; the fast, fugal and accurate way. *Applied Cognitive Psychology* 18: 105-121.

CrimeStat IV

Part V: Spatial Modeling II

Chapter 15:

OLS Regression Modeling

Ned Levine
Ned Levine & Associates
Houston, TX

Dominique Lord
Zachry Dept. of
Civil Engineering
Texas A & M University
College Station, TX

Table of Contents

Chapter 15:
OLS Regression Modeling[1]

The Regression I and Regression II modules are a series of routines for regression modeling and prediction. This chapter will lay out the basics of regression modeling and prediction and will discuss the Ordinary Least Squares (OLS) model in *CrimeStat*.

Functional Relationships

The aim of a regression model is to estimate a functional relationship between a dependent variable (call it y_i) and one or more independent variables (call them $x_{1i}, \cdots x_{Ki}$). In an actual database, these variables have unique names (e.g., ROBBERIES, POPULATION), but we will use general symbols to describe these variables. The functional relationship can be specified by an equation (15.1):

$$y_i = f(x_{1i}, \cdots, x_{Ki}) + \varepsilon_i \tag{15.1}$$

where Y is the dependent variable, $x_{1i}, \cdots x_{Ki}$ are the independent variables, $f(\,)$ is a functional relationship between the dependent variable and the independent variables, and ε_i is an error term (essentially, the difference between the actual value of the dependent variable and that predicted by the relationship).

Normal Linear Relationships

The simplest relationship between the dependent variable and the independent variables is *linear* with the dependent variable being normally distributed,

$$y_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_K x_{Ki} + \varepsilon_i \tag{15.2}$$

---

[1] The regression chapters are the result of the effort of many persons. The maximum likelihood routines were produced by Ian Cahill of Cahill Software in Edmonton, Alberta as part of his MLE++ software package. We are grateful to him for providing these routines and for conducting quality control tests on them. The basic MCMC algorithm in *CrimeStat* for the Poisson-Gamma and Poisson-Gamma-CAR models was designed by Dr. Shaw-Pin Miaou of College Station, TX. We are grateful for Dr. Miaou for this effort. Improvements to the algorithm were made by us, including the block sampling strategy and the calculation of summary statistics. Dr. Dominique Lord of Texas A & M University provided technical advice on the Poisson-based models. Dr. Byung-Jung Park of the Korea Transport Institute expanded the MCMC algorithms to include various dispersion functions and a Simultaneous Autoregressive function. Dr. Ned Levine developed the block sampling methodology and provided overall project management. The programmer for the routines was Ms. Haiyan Teng of Houston, TX. We are also grateful to Dr. Richard Block of Loyola University in Chicago (IL) for testing the MCMC and MLE routines.

This equation can be written in a simple matrix notation: $y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i$ where $\mathbf{x}_i^T = (1, x_{1i}, \cdots, x_{Ki})$ and $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots, \beta_K)^T$. The number one in the first element of $\mathbf{x}_i^T$ represents an intercept while T denotes that the matrix $\mathbf{x}_i^T$ is transposed.

This function says that a unit change in each independent variable, x_{ki}, for every observation, is associated with a unit change in the dependent variable, y_i. The coefficient of each variable, β_k, specifies the amount of change in y_i associated with that independent variable while keeping all other independent variables in the equation constant. The first term, β_0, is the intercept, a constant that is added to all observations. The error term, ε_i, is assumed to be *identically and independently* distributed (***iid***) across all observations, normally distributed with an expected mean of 0 and a constant standard deviation. If each of the independent variables has been standardized by

$$z_k = \frac{x_k - \bar{x}_k}{std(x_k)} \tag{15.3}$$

then the standard deviation of the error term will be 1.0 and the coefficients will be standardized, b_1, b_2, b_3, and so forth.

The equation is estimated by one of two methods, ordinary least squares (OLS) and maximum likelihood estimation (MLE). Both solutions produce the same results. The OLS method minimizes the sum of the squares of the residual errors while the maximum likelihood approach maximizes a joint probability density function.

Ordinary Least Squares

Appendix B by Luc Anselin discusses the method in more depth. Briefly, the intercept and coefficients are estimated by choosing a function that minimizes the residual errors by setting:

$$\sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{k=1}^{K} \beta_k x_{ki} \right) x_{ki} = 0 \tag{15.4}$$

for *k=1* to *K* independent variables or, in matrix notation:

$$\mathbf{X}^T (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = 0 \tag{15.5}$$

$$\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} = \mathbf{X}^T\mathbf{y} \tag{15.6}$$

where $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N)^T$ and $\mathbf{y} = (y_1, y_2, \cdots, y_N)^T$.

The solution to this system of equations yields the familiar matrix expression for $\mathbf{b}_{OLS} = (b_0, b_1, \cdots, b_K)^T$

$$\mathbf{b}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{15.7}$$

An estimate for the error variance follows as

$$s^2_{OLS} = \sum_{i=1}^{N}\left(y_i - b_0 - \sum_{k=1}^{K} b_k x_{ki}\right)^2 /(N-K-1) \tag{15.8}$$

or, in matrix notation,

$$s^2_{OLS} = \mathbf{e}^T\mathbf{e}/(N-K-1) \tag{15.9}$$

Maximum Likelihood Estimation

For the maximum likelihood method, the *likelihood* of a function is the joint probability density of a series of observations (Wikipedia, 2010; Myers, 1990). Suppose there is a sample of *n* independent observations $(x_1, x_2, \cdots, x_N)$ that are drawn from an unknown *probability density* distribution but from a known family of distributions, for example the single-parameter exponential family. This is specified as $f(\cdot \mid \boldsymbol{\theta})$ where $\boldsymbol{\theta}$ is the parameter (or parameters if there are more than one) that define the uniqueness of the family. The joint density function will be:

$$f(x_1, x_2, \cdots, x_N \mid \boldsymbol{\theta}) = f(x_1 \mid \boldsymbol{\theta}) \times f(x_2 \mid \boldsymbol{\theta}) \times \cdots \times f(x_N \mid \boldsymbol{\theta}) \tag{15.10}$$

and is called the ***likelihood*** function:

$$L(\boldsymbol{\theta} \mid x_1, x_2, \cdots, x_N) = f(x_1, x_2, \cdots, x_N \mid \boldsymbol{\theta}) = \prod_{i=1}^{N} f(x_i \mid \boldsymbol{\theta}) \tag{15.11}$$

where L is the likelihood and $\prod$ is the product term.

Typically, the likelihood function is interpreted in term of natural logarithms since the logarithm of a product is a sum of the logarithms of the individual terms. That is,

$$\ln\left\{\prod_{i=1}^{N} f(x_i \mid \boldsymbol{\theta})\right\} = \ln\left[f(x_1 \mid \boldsymbol{\theta})\right] + \ln\left[f(x_2 \mid \boldsymbol{\theta})\right] + \cdots + \ln\left[f(x_n \mid \boldsymbol{\theta})\right] \tag{15.12}$$

This is called the ***Log likelihood*** function and is written as:

$$\ln L(\boldsymbol{\theta} \mid x_1, x_2, \cdots, x_N) = \sum_{i=1}^{N} \ln\left[f(x_i \mid \boldsymbol{\theta})\right] \tag{15.13}$$

For the OLS model, the log likelihood is:

$$\ln L = -\frac{N}{2}\ln(2\pi) - \frac{N}{2}\ln(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \mathbf{x}_i^T\boldsymbol{\beta})^2 \tag{15.14}$$

where N is the sample size and σ^2 is the variance. As a comparison, in Chapter 16 we discuss the Poisson model in which the log likelihood is:

$$\ln L = \sum_{i=1}^{N}\left[-\lambda_i + y_i \ln(\lambda_i) - \ln y_i!\right] \tag{15.15}$$

where $\lambda_i = \exp(\mathbf{x}_i^T\boldsymbol{\beta})$ is the conditional mean for zone i, and y_i is the observed number of events for zone i. As mentioned, Anselin provides a more detailed discussion of these models in Appendix B.

The MLE approach estimates the value of θ that maximizes the log likelihood of the data coming from this family. Because they are all part of the same mathematical family and are distributed as a concave function, the maximum of a joint probability density distribution can be easily estimated. The approach is to, first, define a probability function from this family, second, create a joint probability density function for each of the observations (the Likelihood function); third, convert the likelihood function to a log likelihood; and, fourth, estimate the value of parameters that maximize the joint probability through an approximation method (e.g., Newton-Raphson or Fisher scores). Because the function is regular and known, the solution is relatively easy. Anselin discusses the approach in detail in Appendix B of the *CrimeStat* manual. More detail can be found in Hilbe (2008) or in Train (2009).

In *CrimeStat*, we use the MLE method. Because the OLS method is the most commonly used, a normal linear model is sometimes called an Ordinary Least Squares (OLS) regression. If the equation is correctly specified (i.e., all relevant variables are included), the error term, ε, will be normally distributed with a mean of 0 and a constant variance, σ^2.

The OLS normal estimate is sometimes known as a *Best Linear Unbiased Estimate* (BLUE) since it minimizes the sum of squares of the residuals errors (the difference between the

observed and predicted values of y). In other words, the overall fit of the normal model estimated through OLS or maximum likelihood will produce the best overall fit for a *linear* model. However, keep in mind that because a normal function has the best overall fit does not mean that it fits any particular section of the dependent variable better. In particular, for count data, the normal model usually does a poor job of modeling the observations with the greatest number of events. We will demonstrate this with an example below.

Assumptions of Normal Linear Regression

The normal linear model has some assumptions. When these assumptions are violated, problems can emerge in the model, sometimes easily correctable and other times introducing substantial bias.

Normal Distribution of Dependent Variable

First, the normal linear model assumes that the dependent variable is normally distributed. If the dependent variable is not exactly normally distributed, it has to have its peak somewhere in the middle of the data range and be somewhat symmetrical (e.g., a quartic distribution; see Chapter 10).

For some variables, this assumption is reasonable (e.g., with height or weight of individuals). However, for most variables that crime researchers work with (e.g., number of robberies, number of homicides, journey-to-crime distances), this assumption is usually violated. Most variables that are *counts* (i.e., number of discrete events) are highly skewed. Consequently, when it comes to counts and other extremely skewed variables, the normal (OLS) model will produce distorted results.

Errors are Independent, Constant, and Normally-distributed

Second, the errors in the model, the ε in equation 15.2, must be independent of each other, constant, and normally distributed. This fits the ***iid*** assumption mentioned above. Independence means that the estimation error for any one observation cannot be related to the error for any other observation. Constancy means that the amount of error should be more or less the same for every observation; there will be natural variability in the errors, but this variability should be distributed normally with the mean error being the expected value.

Unfortunately, for most variables that crime researchers and analysts work with, this assumption is usually violated. With count variables, the errors increase with the count and are much higher for observations with large counts than for observation with few counts. Thus, the assumption of constancy is violated. In other words, the variance of the error term is a function

of the count. The shape of the error distribution is also sometimes not normal either but may be more skewed. Also, if there is spatial autocorrelation among the error terms (which would be expected in a spatial distribution), then the error term may be quite irregular in shape; in this latter case, the assumption of independent observations would also be violated.

Independence of Independent Variables

Third, an assumption of the normal model (and any model, for that matter) is that the independent variables are truly independent. In theory, there should be zero correlation between any of the independent variables. In practice, however, many variables are related, sometimes quite highly. This condition, which is called *multicollinearity*, can produce distorted coefficients and overall model effects. The higher the degree of multicollinearity among the independent variables, the greater the distortion in the coefficients. This problem affects all types of models, not just the normal, and it is important to minimize the effects. We will discuss diagnostic methods for identifying multicollinearity later in the chapter.

Adequate Model Specification

Fourth, the normal model assumes that the independent variables have been correctly *specified*. That is, the independent variables are the correct ones to include in the equation and that they have been measured adequately. By 'correct ones', we mean that the independent variable chosen should be a true predictor of the dependent variable, not an extraneous one. With any model, the more independent variables that are added to the equation, in general the greater will be the overall fit. This will be true even if the independent variables are highly correlated with independent variables already in the equation or are mostly irrelevant (but may be slightly correlated due to sampling error). When too many variables are added to an equation, strange effects can occur. *Overfitting* of a model is a serious problem that must be seriously evaluated. Including too many variables will also artificially increase the model's variance (Myers, 1990).

Conversely, a correct specification implies that all the important variables have been included and that none have been left out. When important variables are not included, this is called *underfit*ting a model. Also, not including important variables lead to a biased model (known as the *omitted variables* bias). A large bias means that the model is unreliable for prediction (Myers, 1990). Also, the left out variables can be shown to have irregular effects on the error terms. For example, if there is spatial autocorrelation in the dependent variable (which there usually is), then the error terms will be correlated. Without modeling the spatial autocorrelation (either through a proxy variable that captures much of its effect or through a parameter adjustment), the error can be biased and even the coefficients can be biased.

In other words, adequate specification involves choosing the correct number of independent variables that are appropriate, neither overfitting nor underfitting of the model. Also, it is assumed that the variables have been correctly measured and that the amount of measurement error is very small.

Unfortunately, we often do not know whether a model is correctly specified or not, nor whether the variables have been properly measured. Consequently, there are a number of diagnostics tests that can be brought to bear to reveal whether the specification is adequate. For overfitting, there are tolerance statistics and adjusted summary values. For underfitting, we analyze the error distribution to see if there is a pattern that might indicate *lurking* variables that are not included in the model. In other words, examining violations of the assumptions of a model is an important task in assessing whether there are too many variables included or whether there are variables that should be included but are not, or whether the specification of the model is correct or not.

Example of Modeling Burglaries by Zones

For many problems, normal regression is an appropriate tool. However, for many others, it is not. Let us illustrate this point. A note of caution is warranted here. This example is used to illustrate the application of the normal model in CrimeStat and, as discussed further below, the normal model with a normal error distribution is not appropriate for this kind of dataset. For example, figure 15.1 shows the number of residential burglaries that occurred in 2006 within 1,179 Traffic Analysis Zones (TAZ) inside the City of Houston. The data on burglaries came from the Houston Police Department. There were 26,480 burglaries that occurred in 2006. They were then allocated to the 1,179 TAZ's within the City of Houston. As can be seen, there is a large concentration of residential burglaries in southwest Houston with small concentrations in southeast Houston and in parts of north Houston.

The distribution of burglaries by zones is quite skewed. Figure 15.2 shows a graph of the number of burglaries per zone. Of the 1,179 traffic analysis zones, 250 had no burglaries occur within them in 2006. On the other hand, one zone had 284 burglaries occur within it. The graph shows the number of burglaries up to 59; there were 107 zones with 60 or more burglaries that occurred in them. About 58% of the burglaries occurred in 10% of the zones. In general, a small percentage of the zones have the majority of the burglaries.

Example of Normal Linear Model

We can set up a normal linear model to try to predict the number of burglaries that occurred in each zone in 2006. We obtained estimates of population, employment and income from the transportation modeling group within the Houston-Galveston Area Council, the

Figure 15.1:

Burglaries in the City of Houston

Number in Each Traffic Analysis Zone: 2006

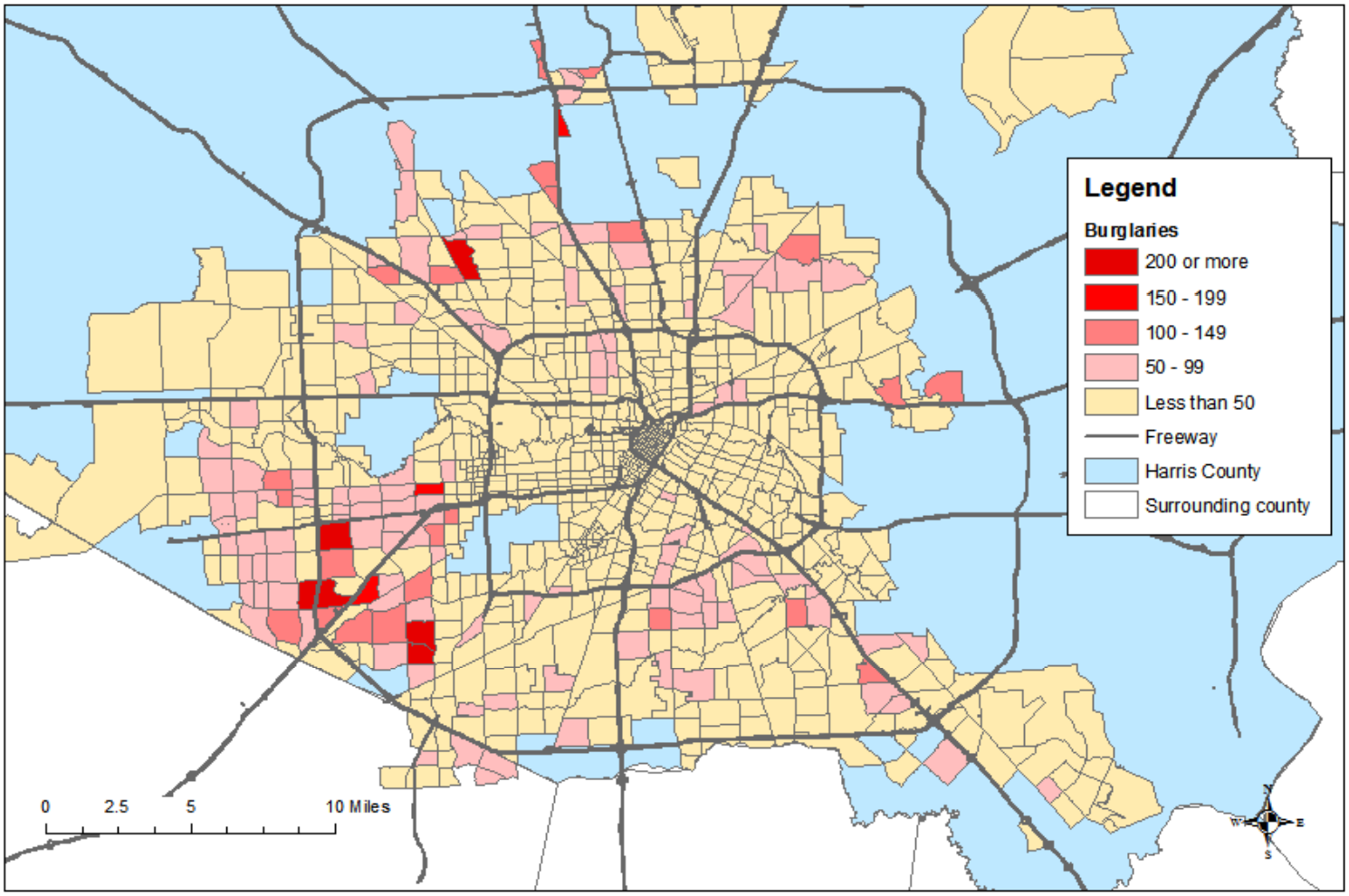
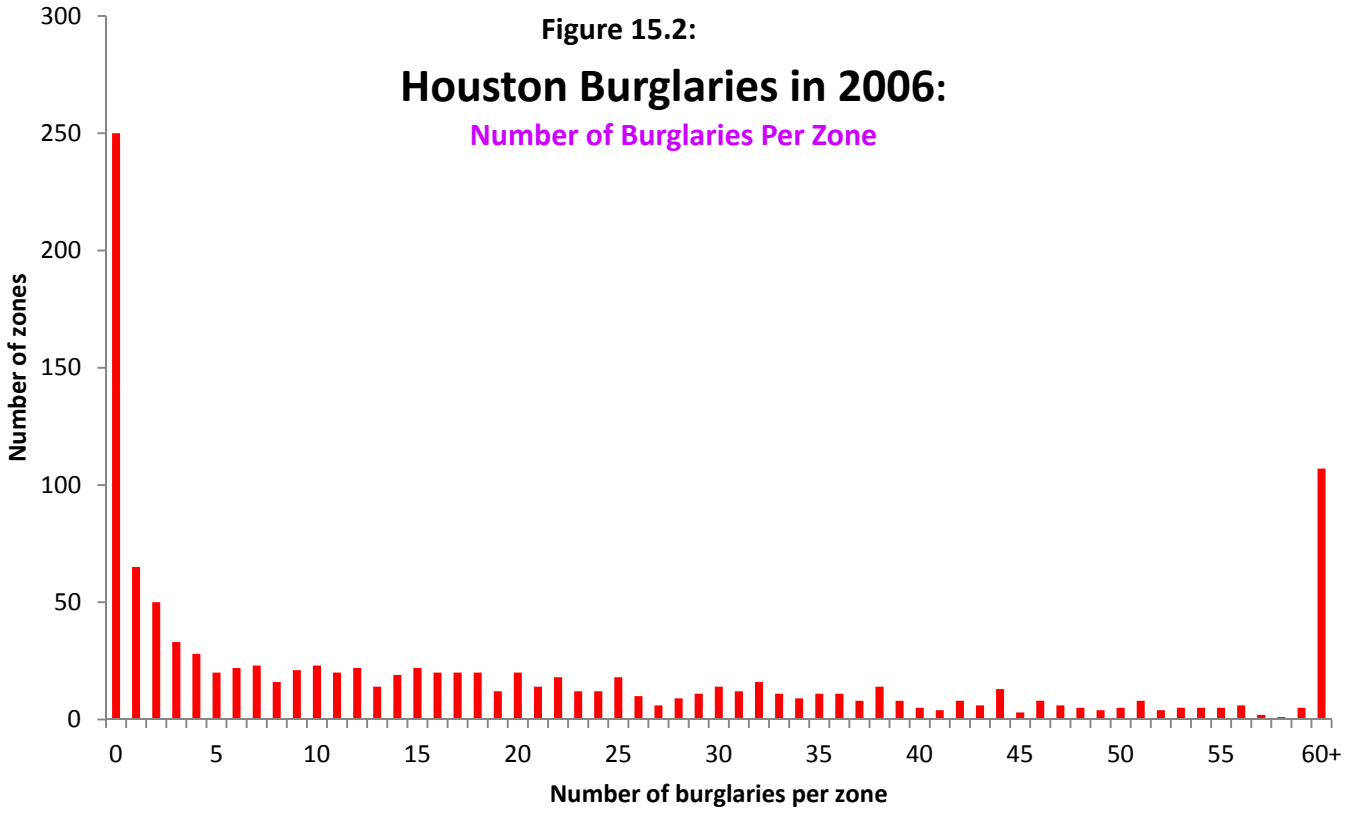


Figure 15.2:


Houston Burglaries in 2006:
Number of Burglaries Per Zone
Number of zones
300
250
200
150
100
50
0
0
5
10
15
20
25
30
35
40
45
50
55
60+
Number of burglaries per zone

Metropolitan Planning Organization for the area (H-GAC, 2010). Specifically, the model relates the number of 2006 burglaries to the number of households, number of jobs (employment), and median income of each zone. The estimates for the number of households and jobs were for 2006 while the median income was that measured by the 2000 census. Table 15.1 present the results of the normal (OLS) model.

Table 15.1:

Predicting Burglaries in the City of Houston: 2006
Ordinary Least Squares: Full Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1,179
Df:	1,174
Type of regression model:	Ordinary Least Squares
F-test of model:	357.2 $p \leq .0001$
R-square:	0.48
Adjusted r-square:	0.48
Mean absolute deviation:	13.5
1st (highest) quartile:	26.4
2nd quartile:	10.6
3rd quartile:	8.3
4th (lowest) quartile:	8.8
Mean squared predictive error:	505.1
1st (highest) quartile:	1,497.5
2nd quartile:	270.4
3rd quartile:	134.3
4th (lowest) quartile:	120.9

Predictor	DF	Coefficient	Stand Error	Tolerance	VIF	t-value	p
INTERCEPT	1	12.9320	1.269	-	-	10.19	0.001
HOUSEHOLDS	1	0.0256	0.0008	0.923	1.083	31.37	0.001
JOBS	1	-0.0002	0.0005	0.903	1.107	-0.453	n.s.
MEDIAN HOUSEHOLD INCOME	1	-0.0002	0.00003	0.970	1.031	-6.88	0.001

Summary Statistics for the Goodness-of-Fit

The table presents two types of results. First, there is summary information. Information on the size of the sample (in this case, 1,179) and the degrees of freedom (the sample size less one for each parameter estimated including the intercept and one for the mean of the dependent variable); in the example, there are 1,174 degrees of freedom (1,179 – 1 for the intercept, 1 for HOUSEHOLDS, 1 for JOBS, 1 for MEDIAN HOUSEHOLD INCOME, and 1 for the mean of the dependent variable, 2006 BURGLARIES).

The F-test presents an Analysis of Variance test of the ratio of the *mean square error* (MSE) of the model compared to the total mean square error (Kanji, 1993, 131; Abraham & Ledolter, 2006, 41-51). Next, there is the R-square (or R^2) statistic, which is the most common type of overall fit test. This is the percent of the total variance of the dependent variable accounted for by the model. More formally, it is defined as:

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \tag{15.16}$$

where y_i is the observed number of events for a zone, i, $\hat{y}_i$ is the predicted number of events given a set of *K* independent variables, and Mean $\bar{y}$ is the mean number of events across zones. The R-square value is a number from 0 to 1; 0 indicates no predictability while 1 indicates perfect predictability.

For a normal (OLS) model, R-square is a very consistent estimate. It increases in a linear manner with predictability and is a good indicator of how effective a model has fit the data. As with all diagnostic statistics, the value of the R-square increases with more independent variables. Consequently, an R-square adjusted for degrees of freedom is also calculated - the *adjusted r-square* in the table. This is defined as:

$$R_a^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2 / (N - K - 1)}{\sum (y_i - \bar{y})^2 / (N - 1)} \tag{15.17}$$

where N is the sample size and K is the number of independent variables.

The R^2 value is sometimes called the *coefficient of determination*. It is an indicator of the extent to which the independent variables in the model *predict* (or explain) the dependent variable. One interpretation of the R^2 is the percent of the variance of Y accounted for by the variance of the independent variables (plus the intercept and any other constraints added to the model). The *unexplained* variance is 1 - R^2 or the extent to which the model does not explain the

variance of the dependent variable. For a normal linear model, the R^2 is relatively straightforward. In the example, both the F-test is highly significant and the R^2 is substantial (48% of the variance of the dependent variable is explained by the independent variables). However, for non-linear models, it is not at all an intuitive measure and has been shown to be unreliable (Miaou, 1996).

The final two summary measures are *Mean Squared Predictive Error* (MSPE), which is the average of the squared residual errors, and the *Mean Absolute Deviation* (MAD), which is the average of the absolute value of the residual errors (Oh, Lyon, Washington, Persaud, & Bared, 2003). The lower the values of these measures, the better the model fits the data.

These measures are also calculated for specific quartiles. The 1st quartile represents the error associated with the 25% of the observations that have the highest values of the dependent variable while the 4th quartile represents the error associated with the 25% of the observations with the lowest value of the dependent variable. These percentiles are useful for examining how well a model fits the data and whether the fit is better for any particular section of the dependent variable. In the example, the fit is better for the low end of the distribution (the zones with zero or few burglaries) and less good for the higher end. We will use these values in comparing the normal model to other models.

It is important to point out that the summary measures are more useful when several models with a different number of variables are compared with each other than for evaluating a single model.

Statistics on Individual Coefficients

The second type of information presented is about each of the coefficients. The table lists the independent variables plus the intercept. For each coefficient, the degrees of freedom associated are presented (one per variable) plus the estimated linear coefficient. For each coefficient, there is an estimated standard error, a t-test of the coefficient (the coefficient divided by the standard error), and the approximate two-tailed probability level associated with the t-test (essentially, an estimate of the probability that the null hypothesis of zero coefficient is correct). Usually, if the probability level is smaller than 5% (.05), then we reject the null hypothesis of a zero coefficient though frequently 1% (.01) or even 0.1% (0.001) have been used to reduce the likelihood that a false alternative hypothesis has been selected (called a *Type I error*).

The last two parameters included in the table are the *tolerance* of the coefficient and the *VIF* (or Variance *Inflation Factor*). They are measures of multicollinearity (or one type of overfitting). Basically, they measure the extent to which each independent variable correlates with the other dependent variables in the equation. The traditional tolerance test is a normal

model relating each independent variable to the *other* independent variables (StatSoft, 2010; Berk, 1977). It is defined as:

$$Tol_i = 1 - R^2_{j \neq i} \tag{15.18}$$

where $R^2_{j \neq i}$ is the R-square associated with the prediction of one independent variable with the remaining independent variables in the model using an OLS model. The VIF is simply the reciprocal of tolerance:

$$VIF_i = 1 / Tol_i \tag{15.19}$$

In other words, the tolerance of each independent variable is the unexplained variance of a model that relates the variable to the other independent variables. If an independent variable is highly related (correlated with) to the other independent variables in the equation, then it will have a low tolerance. Conversely, if an independent variable is independent of the other independent variables in the equation, then it will have a high tolerance. In theory, the higher the tolerance, the better since each independent variable should be unrelated to the other independent variables. In practice, there is always some degree of overlap between the independent variables so that a tolerance of 1.0 is rarely, if ever, achieved. However, if the tolerance is low (e.g., 0.70 or below), this suggests that there is too much overlap in the independent variables and that the interpretation will be unclear. In Chapter 17, we will discuss multicollinearity and the general problem of overfitting in more detail.

Note that the statistic is labeled as *pseudo-tolerance* in the CrimeStat output. The reason is that this statistic is only approximate when the independent variable is skewed, a situation that we will discuss shortly. For a normally-distributed independent variable (or approximately normally-distributed), however, the tolerance test is exact.

Looking at the output in Table 15.1, we see that the number of burglaries is positively associated with the intercept and the number of households and negatively associated with the median household income. The relationship to the number of jobs is also negative, but not significant. Essentially, zones with larger numbers of households but lower household incomes are associated with more residential burglaries. Because the model is linear, each of the coefficients contributes to the prediction in an additive manner. The intercept is 12.93 and indicates that, on average, each zone had 12.93 burglaries. For every household in the zone, there was a contribution of 0.0256 burglaries. For every job in the zone, there was a contribution of -0.0002 burglaries. For every dollar increase in median household income, there is a decrease of -0.0002 burglaries. Thus, to predict the number of burglaries with the full model in any one zone, i, we would take the intercept – 12.93, and add in each of these components:

$$(BURGLARIES)_i = 12.93 + 0.0256(HOUSEHOLDS)_i - 0.0002(JOBS)_i - 0.0002(MEDIAN\ HOUSEHOLD\ INCOME)_i \quad (15.20)$$

To illustrate, TAZ 833 had 1762 households in 2006, 2,698 jobs in 2006, and had a median household income of $27,500 in 2000. The model's prediction for the number of burglaries in TAZ 833 is:

Number of burglaries (TAZ833) = 12.93 + 0.0256*1762 – 0.0002*2,698 – 0.0002*27,500
= 52.0

The actual number of burglaries that occurred in TAZ 833 was 78.

Estimated Error in the Model for Individual Coefficients

In *CrimeStat*, and in most statistical packages, there is additional information that can be output as a file. There is the *predicted* value for each observation. Essentially, this is the linear prediction from the model. There is also the *residual* error, which is the difference between the actual (observed) value for each observation, i, and that predicted by the model. It is defined as:

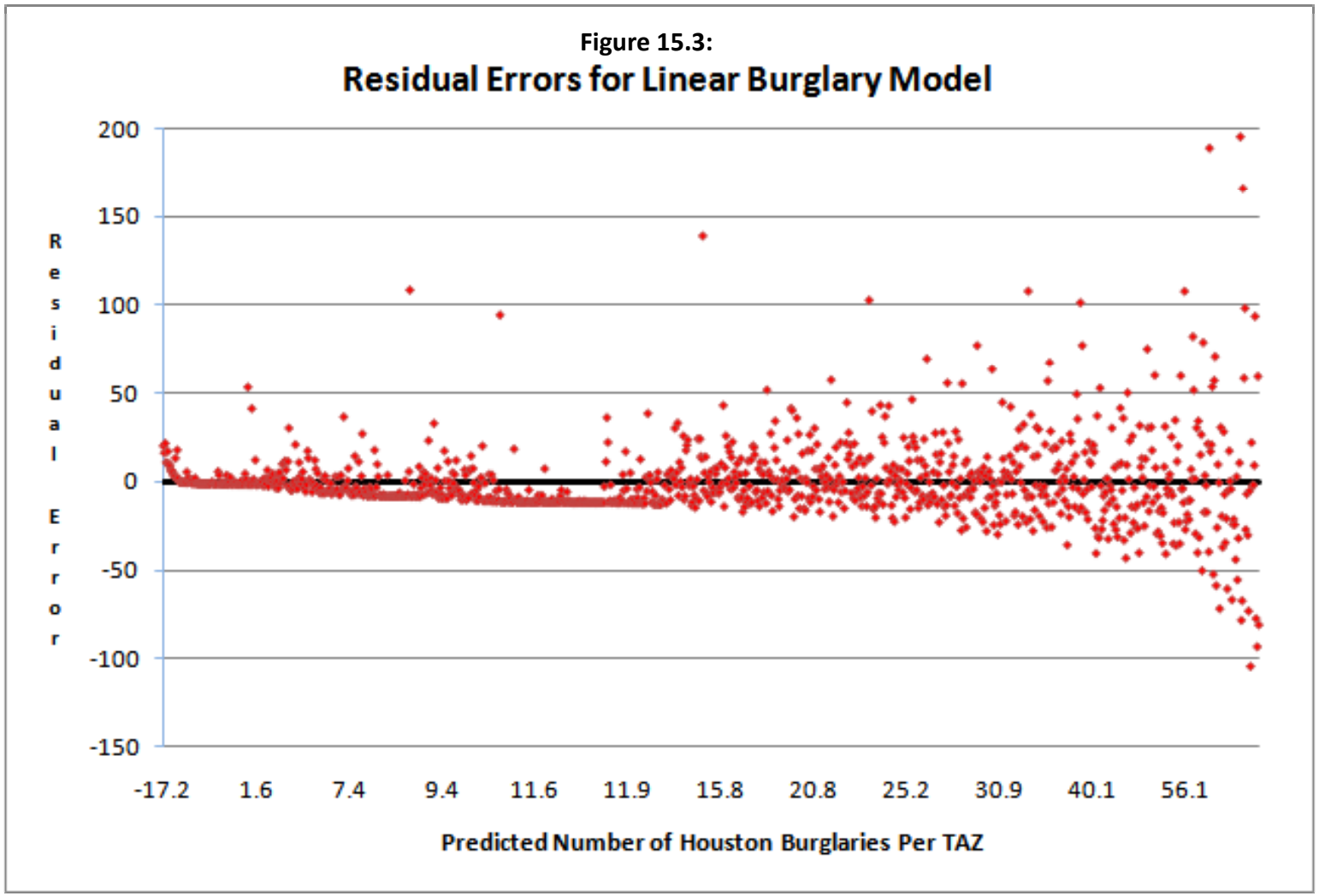
$$\text{Residual error}_i = \text{Observed Value}_i - \text{Predicted value}_i \quad (15.21)$$

Table 15.2 below gives predicted values and residual errors for five of the observations from the Houston burglary data set. Analysis of the residual errors is one of the best tools for diagnosing problems with the model. A plot of residual errors against predicted values indicate whether the prediction is consistent across all values of the dependent variable and whether the underlying assumptions of the normal model are valid (see below). Figure 15.3 show a graph of the residual errors of the full model against the predicted values for the model estimated in table 1. As can be seen, the model fits quite well for zones with few burglaries, up to about 12 burglaries per zone.

Table 15.2:
Predicted Values and Residual Error for Houston Burglaries: 2006
(5 Traffic Analysis Zones)

Zone (TAZ)	Actual value	Predicted value	Residual error
833	78	52.0	26.0
831	46	35.9	10.1
911	89	67.6	21.4
2173	30	42.3	-12.3
2940	3	10.2	-7.2

Figure 15.3:


Residual Errors for Linear Burglary Model
Residual Error
200
150
100
50
0
-50
-100
-150
-17.2
1.6
7.4
9.4
11.6
11.9
15.8
20.8
25.2
30.9
40.1
56.1
Predicted Number of Houston Burglaries Per TAZ

However, for the zones with many predicted burglaries (the ones that we are most likely interested in), the model does quite poorly. First, the errors increase the greater the number of predicted burglaries. Sometimes the errors are positive, meaning that the actual number of burglaries is much higher than predicted and sometimes the errors are negative, meaning that we are predicting more burglaries than actually occurred. More importantly, the residual errors indicate that the model has violated one of the basic assumptions of the normal model, namely that the errors are independent, constant, and identically-distributed. It is clear that they are not.

Because there are errors in predicting the zones with the highest number of burglaries and because the zones with the highest number of burglaries were somewhat concentrated, there are spatial distortions from the prediction. Figure 15.4 show a map of the residual errors of the normal model. As can be seen by comparing this map with the map of burglaries (figure 15.1), typically the zones with the highest number of burglaries (mostly in southwest Houston) were under-estimated by the normal model (shown in red) whereas some zones with few burglaries ended up being over-estimated by the normal model (e.g., in far southeast Houston).

In other words, the normal linear model is not necessarily good for predicting Houston burglaries. It tends to underestimate zones with a large number of burglaries but overestimates zones with few.

Violations of Assumptions for Normal Linear Regression

There are several deficiencies with the normal (OLS) model. First, normal models are not good at describing skewed dependent variables, as we have shown. Since crime distributions are usually skewed, this is a serious deficiency for multivariate crime analysis. Second, a normal model can have negative predictions. With a count variable, such as the number of burglaries committed in a zone, the minimum number is zero. That is, the count variable is always *positive*, being bounded by 0 on the lower limit and some large number on the upper limit. The normal model, on the other hand, can produce negative predicted values since it is additive in the independent variables. This clearly is illogical and is a major problem with data that are highly skewed. If most records have values close to zero, it is very possible for a normal model to predict a negative value.

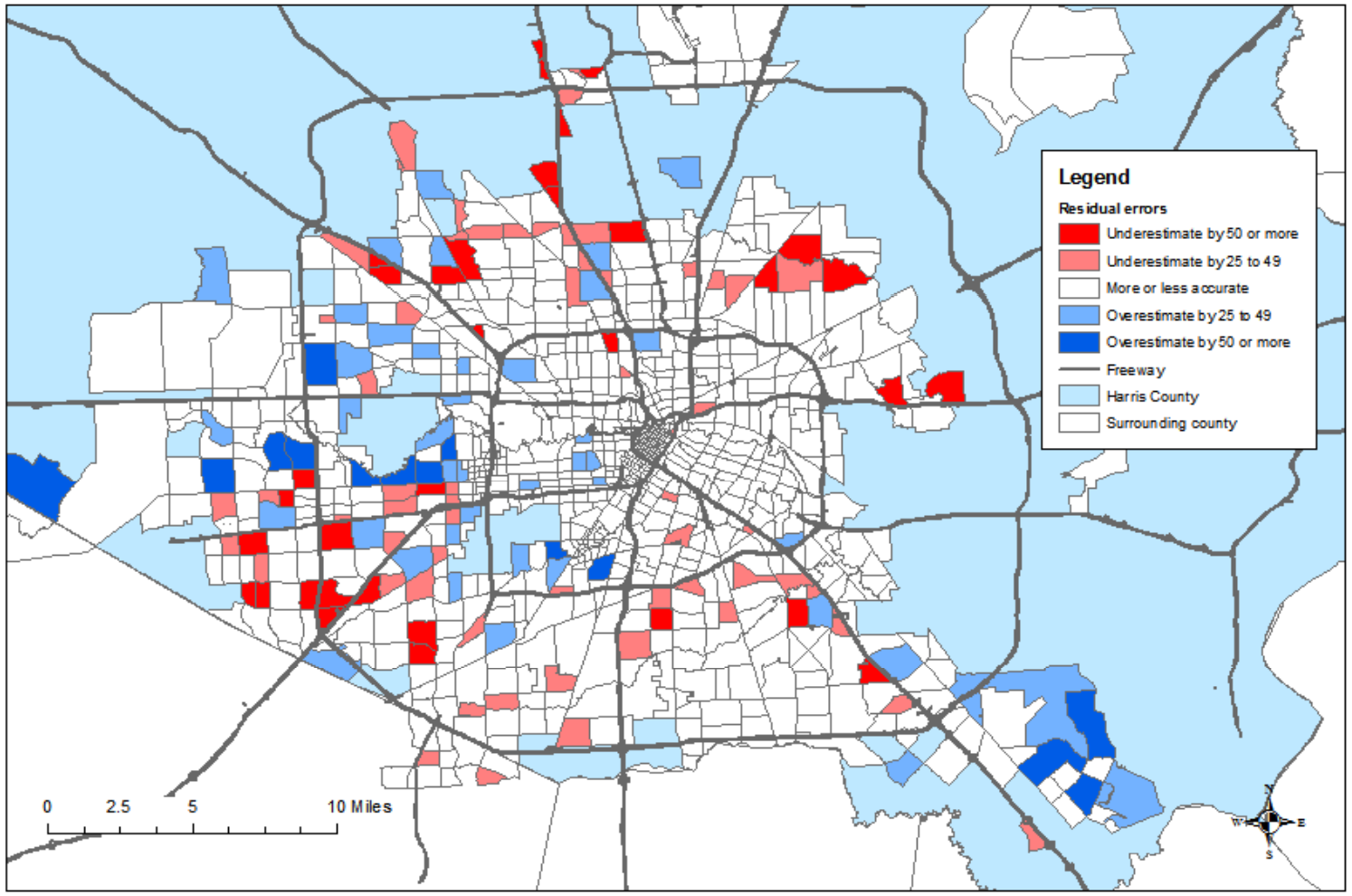
Non-consistent Summation

A third problem with the normal model is that the sum of the observed values does not necessarily equal the sum of the predicted values. Since the estimates of the intercept and coefficients are obtained by minimizing the sum of the squared residual errors (or maximizing the joint probability distribution, which leads to the same result), there is no balancing mechanism to require that they add up to the same as the input values. In calibrating the model,

Figure 15.4:

Predicting Burglaries in the City of Houston: 2006

Residual Errors from Linear Model



adjustments can be made to the intercept term to force the sum of the predicted values to be equal to the sum of the input values. But in applying that intercept and coefficients to another data set, there is no guarantee that the consistency of summation will hold. In other words, the normal method cannot guarantee a consistent set of predicted values.

Non-linear Effects

A fourth problem with the normal model is that it assumes the independent variables are normal in their effect. If the dependent variable was normal or relatively balanced, then a normal model would be appropriate. But, when the dependent variable is highly skewed, as is seen with these data, typically the additive effects of each component cannot usually account for the non-linearity. Independent variables have to be transformed to account for the non-linearity and the result is often a complex equation with non-intuitive relationships.[2] It is far better to use a non-linear model for a highly skewed dependent variable.

Greater Residual Errors

The final problem with a normal model and a skewed dependent variable is that the model tends to over- or under-predict the correct values, but rarely comes up with the correct estimate. As we saw with the example above, typically a normal equation produces non-constant residual errors with skewed data. In theory, errors in prediction should be uncorrelated with the predicted value of the dependent variable. Violation of this condition is called *heteroscedasticity* because it indicates that the residual variance is not constant. The most common type is an increase in the residual errors with higher values of the predicted dependent variable. That is, the residual errors are greater at the higher values of the predicted dependent variable than at lower values (Draper and Smith, 1981, 147).

A highly skewed distribution tends to exacerbate this. Because the least squares procedure minimizes the sum of the squared residuals, the regression line balances the lower residuals with the higher residuals. The result is a regression line that neither fits the low values nor the high values. For example, motor vehicle crashes tend to concentrate at a few locations (crash hot spots). In estimating the relationship between traffic volume and crashes, the hot spots tend to unduly influence the regression line. The result is a line that neither fits the number of expected crashes at most locations (which is low) nor the number of expected crashes at the hot spot locations (which are high).

---

[2] For example, to account for the skewed dependent variable, one or more of the independent variables have to be transformed with a non-linear operator (e.g., log or exponential term). When more than one independent variable is non-linear in an equation, the model is no longer easily understood. It may end up making reasonable predictions for the dependent variable, but is not intuitive nor easily explained to non-specialists.

Corrections to Violated Assumptions in Normal Linear Regression

Some of the violations in the assumptions of an OLS normal model can be corrected.

Eliminating Unimportant Variables

One good way to improve a normal model is to eliminate variables that are not important. Including variables in the equation that do not contribute very much adds 'noise' (variability) to the estimate. In the above example, the variable, JOBS, was not statistically significant and, hence, did not contribute any real effect to the final prediction. This is an example of overfitting a model. Whether we use the criteria of statistical significance to eliminate non-essential variables or simply drop those with a very small effect is less important than the need to reduce the model to only those variables that truly predict the dependent variable. We will discuss the 'pros' and 'cons' of dropping variables in Chapter 17, but for now we argue that a good model - one that will be good not just for description but for prediction, is usually a simple model with only the strongest variables included.

To illustrate, we reduce the burglary model further by dropping the non-significant variable (JOBS). Table 15.3 show the results. Comparing the results with those from Table 15.1, we can see that the overall fit of the model is actually slightly better (an F-value of 536.0 compared to 357.2). The R^2 values are the same while the mean squared predictive error is slightly worse while the mean absolute deviation is slightly better. The coefficients for the two common independent variables are almost identical while that for the intercept is slightly less (which is good since it contributes less to the overall result).

In other words, dropping the non-significant variable has led to a slightly better fit. One will usually find that dropping non-significant or unimportant variables makes models more stable without much loss of predictability, and conceptually they become simpler to understand.

Eliminating Multicollinearity

Another way to improve the stability of a normal model is to eliminate variables that are substantially correlated with other independent variables in the equation. This is the *multicollinearity* problem that we discussed above. Even if a variable is statistically significant in a model, if it is also correlated with one or more of the other variables in the equation, then it is capturing some of the variance associated with those other variables. The results are ambiguity in the interpretation of the coefficients as well as error in trying to use the model for

Table 15.3:

Predicting Burglaries in the City of Houston: 2006
Ordinary Least Squares: Reduced Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1,179
Df:	1,175
Type of regression model:	Ordinary Least Squares
F-test of model:	536.0 $p \leq .0001$
R-square:	0.48
Adjusted r-square:	0.48
Mean absolute deviation:	13.5
1st (highest) quartile:	26.5
2nd quartile:	10.6
3rd quartile:	8.3
4th (lowest) quartile:	8.8
Mean squared predictive error:	505.1
1st (highest) quartile:	1498.8
2nd quartile:	269.5
3rd quartile:	135.1
4th (lowest) quartile:	120.2

Predictor	DF	Coefficient	Stand Error	Tolerance	VIF	t-value	p
INTERCEPT	1	12.8099	1.240	-	-	10.33	0.001
HOUSEHOLDS	1	0.0255	0.0008	0.994	1.006	33.44	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.0002	0.00003	0.994	1.006	-7.03	0.001

prediction. Multicollinearity means that essentially there is overlap in the independent variables; they are measuring the same thing. It is better to drop a multicollinear variable even if it results in a loss in fit since it will usually result in a simpler and more stable model.

For the Houston burglary example, the two remaining independent variables in Table 15.3 are relatively independent; their tolerances are 0.994 respectively, which points to little overlap in the variance that they account for in the dependent variable. Therefore, we will keep

these variables. However, in the next chapter, we will present an example of how multicollinearity can lead to ambiguous coefficients.

Transforming the Dependent Variable

It may be possible to correct the normal model by transforming the dependent variable (in another program since *CrimeStat* does not currently do this). Typically, with a skewed dependent variable and one that has a large range in values, a natural log transformation of the dependent variable can be used to reduce the amount of skewness. The problem will occur for zones with 0 since the natural log of 0 cannot be calculated. Consequently, one takes:

$$\ln y_i = \log_e (y_i + 1) \tag{15.22}$$

where *e* is the base of the natural logarithm (2.718…) and regresses the transformed dependent variable against the linear predictors,

$$\ln y_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_K x_{Ki} + \varepsilon_i \tag{15.23}$$

This is equivalent to the equation

$$y_i = e^{\beta_0 + \beta_1 x_{1i} + \cdots + \beta_K x_{Ki} + \varepsilon_i} \tag{15.24}$$

with, again, *e* being the base of the natural logarithm.

In doing this, it is assumed that the log transformed dependent variable is consistent with the assumptions of the normal model, namely that it is normally distributed with an independent and constant error term, ε, that is also normally distributed.

One must be careful about transforming values that are zero since the natural log of 0 is unsolvable. Usually researchers will set the value of the log-transformed dependent variable to 0 or the value of the dependent variable to a small number (e.g., 1) for cases where the raw dependent variable actually has a value of 0 (e.g., equation 15.22 above). But, one must be careful that it does not distort relationships if there are many zeros in the data. For example, in the burglary data, there were 250 zones (out of 1,179, or 21%) that had zero burglaries!

Example of Transforming Dependent Variable on Houston Burglaries

Using the Houston burglary example from above, we transformed the dependent variable– number of 2006 burglaries per TAZ, by taking the natural logarithm of it. All zones with zero burglaries were automatically given the value of 0 for the transformed variable.

The transformed variable was then regressed against the two independent variables in the reduced form model (from Table 15.3 above). Table 15.4 present the results: The coefficients are similar in sign. The R^2 value is smaller than the untransformed model (0.42 compared to 0.48). Further, the mean squared predictive error is now much lower than the original raw values (1.47 compared to 505.14) and the mean absolute deviation is also much lower (1.05 compared to 13.50).[3] In other words, transforming the dependent variable into a logarithm has improved the fit of the estimate substantially.

Another type of transformation that is sometimes used is to convert the independent variables and, occasionally, the dependent variable into Z-scores. The Z-score of a variable is defined as:

$$z_k = \frac{x_k - \bar{x}_k}{std(x_k)} \tag{15.25}$$

But all this will do is to standardize the scale of the variable as standard deviations around an expected value of zero, but not alter the shape. If the dependent variable is skewed, taking the Z-score of it will not alter its skewness.

A third type of transformation takes the square root of the dependent variable and regress it in an OLS model. When we did this with the Houston burglary data, however, the fit was not as good as the log transformation (model not shown). The mean absolute deviation was more than 50% higher and the mean squared predictive error was three times higher. Again, the basic reason is that a count, such as the number of burglaries, is typically Poisson-distributed, meaning that it is exponential in form. Essentially, skewness is a fundamental property of a distribution and the normal model is poorly suited for modeling it.

Example of Modeling Skewed Variable with OLS

A simple example can illustrate this theoretically. Figure 15.5 shows an exponential distribution that relates a dependent variable, Y, to an independent variable, X. Think of these as any two variables that are positively related (e.g., crime & poverty; crime & unemployment). The data were created in a spreadsheet by the function $Y_i = e^X$ with a random error added to simulate randomness. However, the underlying curve is still exponential. In Figure 15.6, we fit a linear model to the data using the *CrimeStat* module. The result show that the model tended

[3] The errors were calculated by, first, transforming the dependent variable by taking its natural log; second, the natural log was then regressed against the independent variables; third, the predicted values were then calculated; and, fourth, the predicted values were then converted back into raw scores by taking them as the exponents of *e*, the base of the natural logarithm. The residual errors were calculated from the re-transformed predicted values.

Table 15.4:

Predicting Burglaries in the City of Houston: 2006

Log Transformed Dependent Variable

(N= 1,179 Traffic Analysis Zones)

DepVar:	**Natural log of 2006 BURGLARIES**
N:	1,179
Df:	1,175
Type of regression model:	Ordinary Least Squares
F-test of model:	417.4 $p \leq .0001$
R-square:	0.42
Adjusted r-square:	0.42
Mean absolute deviation:	1.05
1st (highest) quartile:	1.23
2nd quartile:	0.94
3rd quartile:	0.56
4th (lowest) quartile:	1.46
Mean squared predictive error:	1.47
1st (highest) quartile:	2.02
2nd quartile:	1.14
3rd quartile:	0.47
4th (lowest) quartile:	2.24

Predictor	DF	Coefficient	Stand Error	Tolerance	VIF	t-value	p
INTERCEPT	1	1.5674	0.067	-	-	23.44	0.001
HOUSEHOLDS	1	0.0012	0.00004	0.994	1.006	28.84	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.000006	0.000001	0.994	1.006	-4.09	0.001

to underestimate both the upper- and lower-ends of the distribution of X, especially the high end while overestimating the middle range.

Transforming the dependent variable into a natural log (i.e., Ln[X]) creates a better fit (Figure 15.6). Similarly, transforming the dependent variable into a square root (i.e., Sqrt[X]) is better than the linear though not as good as the log transformation (Figure 15.7). However, neither transformation are as good as fitting a true Poisson function (Figure 15.8). This can be

Figure 15.5:

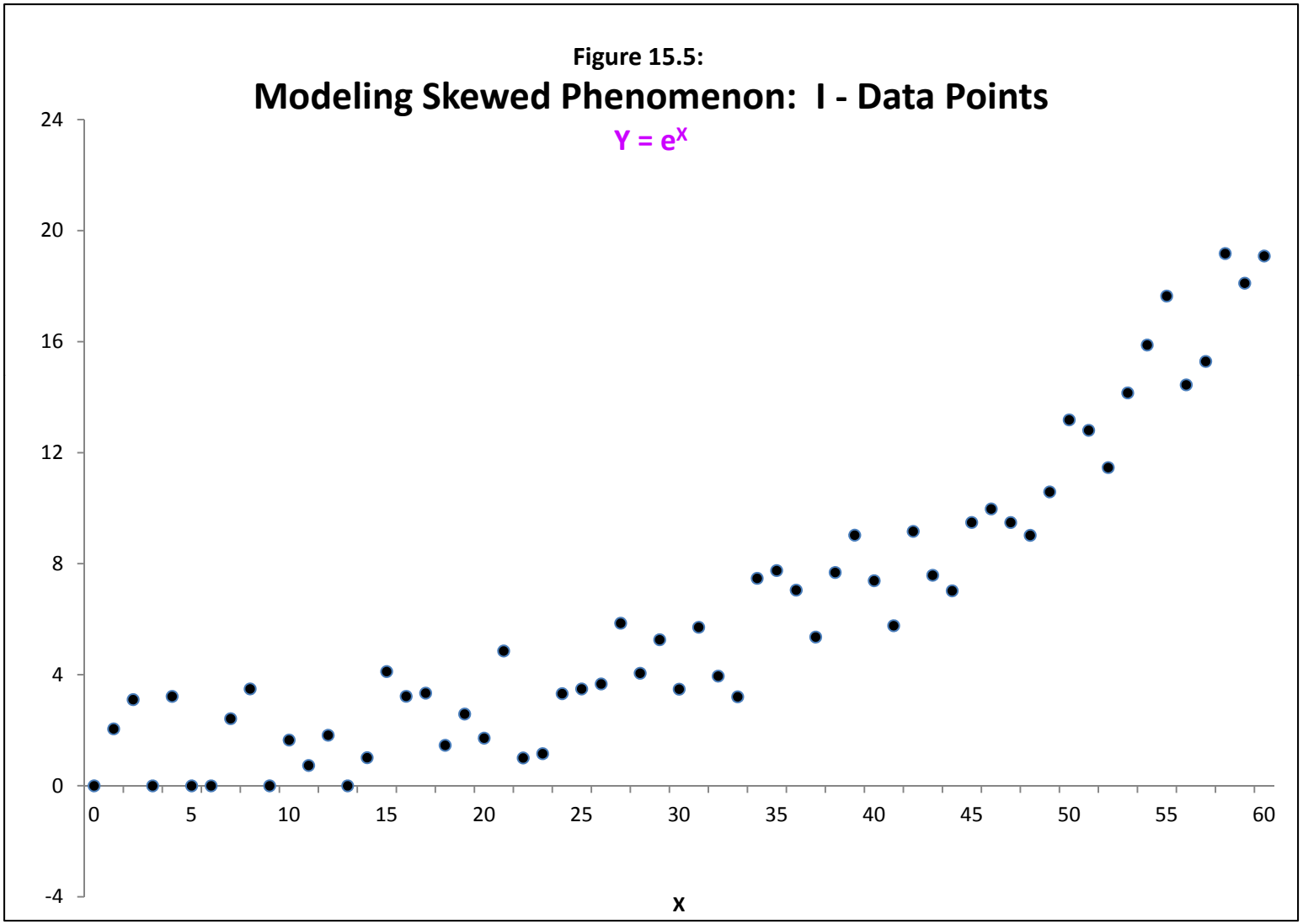
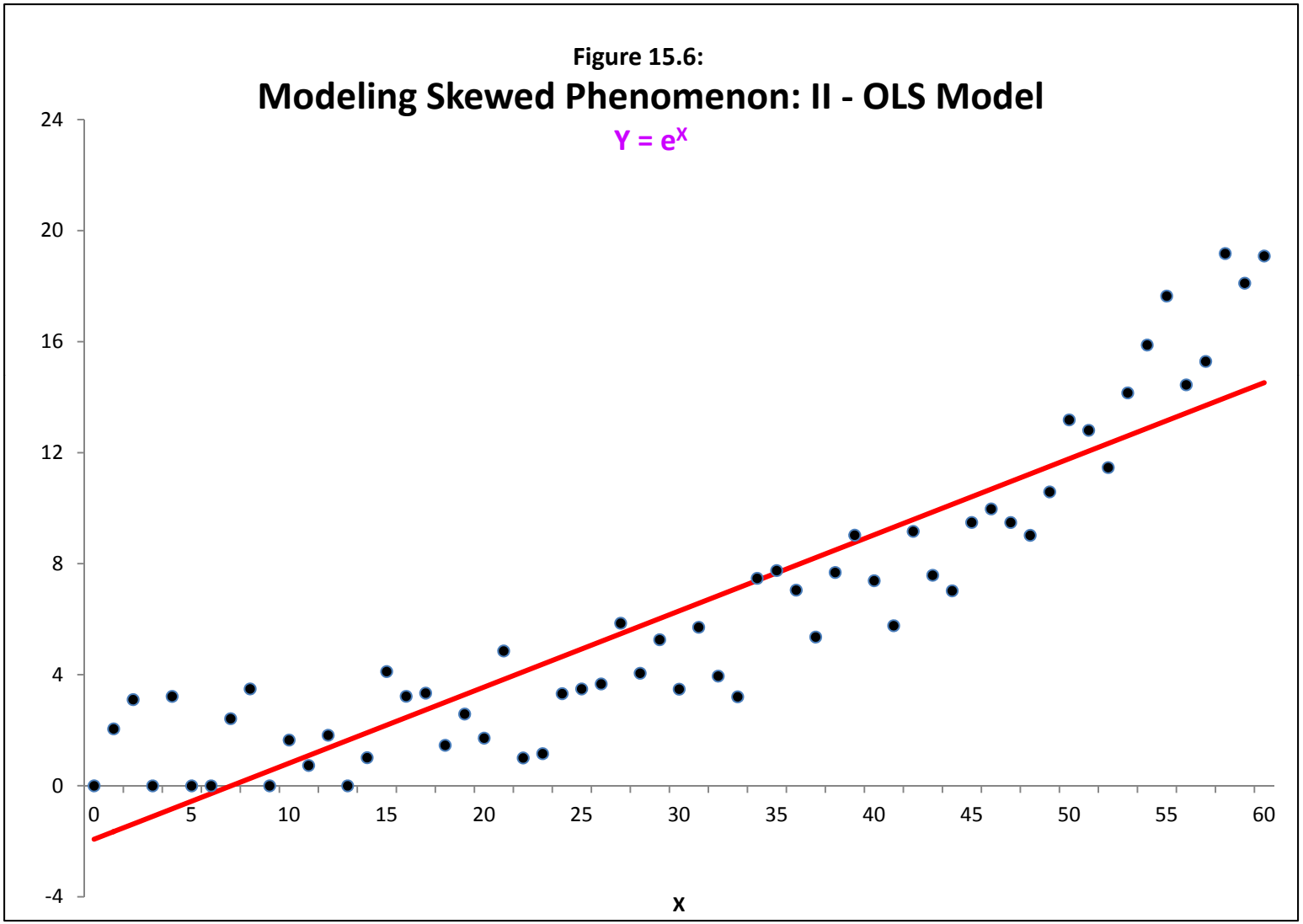

Modeling Skewed Phenomenon: I - Data Points
Y = e^X
24
20
16
12
8
4
0
-4
0
5
10
15
20
25
30
35
40
45
50
55
60
X

Figure 15.6:


Modeling Skewed Phenomenon: II - OLS Model
Y = e^X
24
20
16
12
8
4
0
-4
0
5
10
15
20
25
30
35
40
45
50
55
60
X

seen by comparing the Mean Square Predictive Error (MSPE) and the Mean Absolute Deviation (MAD) statistics including the quartiles for the MAD (Table 15.5).

Table 15.5:
Comparing Errors for Models Estimating Exponential Function
Mean Squared Predictive Error and Mean Absolute Deviation

	Model			
Error Statistic	**OLS**	**OLS w. Ln(Y)**	**OLS w. Sqrt(Y)**	**Poisson**
MSPE:	**4.96**	**1.94**	**2.57**	**1.80**
MAD:	**1.79**	**1.19**	**1.31**	**1.15**
1st quartile:	2.15	1.15	1.48	0.96
2nd quartile:	2.15	1.35	1.28	1.36
3rd quartile:	2.16	1.01	1.50	1.06
4th quartile:	1.50	1.21	0.93	1.21

As seen, the Poisson provides the best overall fit with both the MSPE and the MAD. While the OLS using the log-transformed dependent variable produces a reasonably good fit, certainly better than the OLS on the untransformed dependent variable, it still provides a poorer fit than a non-linear Poisson function, which is an exponential function. Further, the MAD for the first quartile (i.e., the data points with the highest actual values) is much worse for the OLS of the transformed dependent variable compared to the Poisson. Where the transformed dependent variable does as well if not better than the Poisson is in the last two quartiles, the low end of the X distribution.

With either the log transformation or the square root transformation, the fit is better for the low end of the dependent variable (i.e., those observations with fewer counts of the dependent variable) than for the high end. The reason is because the OLS minimizes the sum of the squared deviations of the predictions from the dependent variable. Since it assumes homoscedasticity in the residual errors across the ranges of independent variables, it cannot adjust the errors at the high end. In other words, no matter what transformation is used with an OLS, the result will always be worse than a Poisson-based model. Since we are usually interested in the high end of the dependent variable (i.e., those observations with many counts), that is a substantial deficiency of the OLS model.

Figure 15.7:

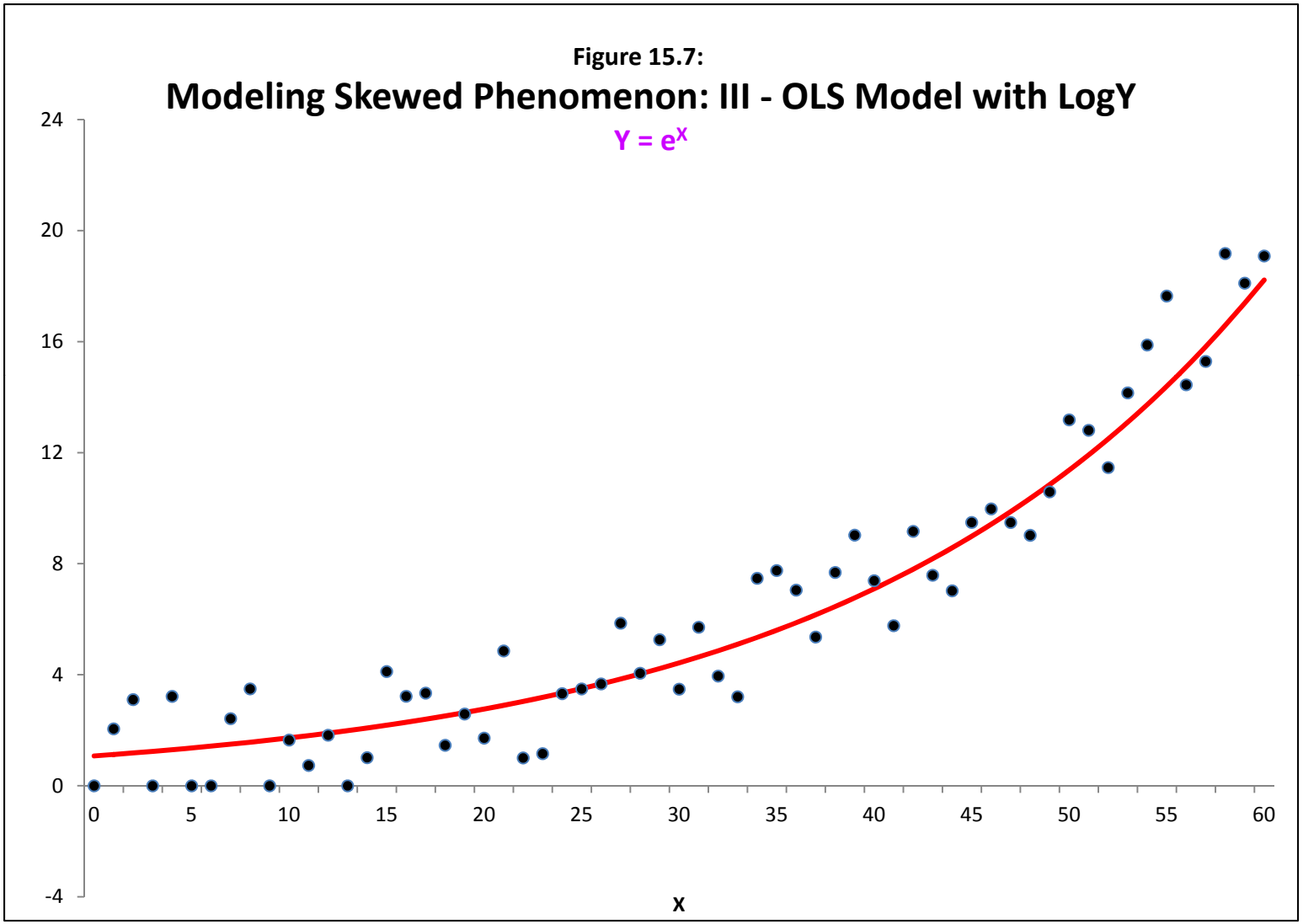

Modeling Skewed Phenomenon: III - OLS Model with LogY
Y = e^X
24
20
16
12
8
4
0
-4
0
5
10
15
20
25
30
35
40
45
50
55
60
X

Figure 15.8:

Modeling Skewed Phenomenon: IV - OLS Model with Square RootY

$Y = e^X$

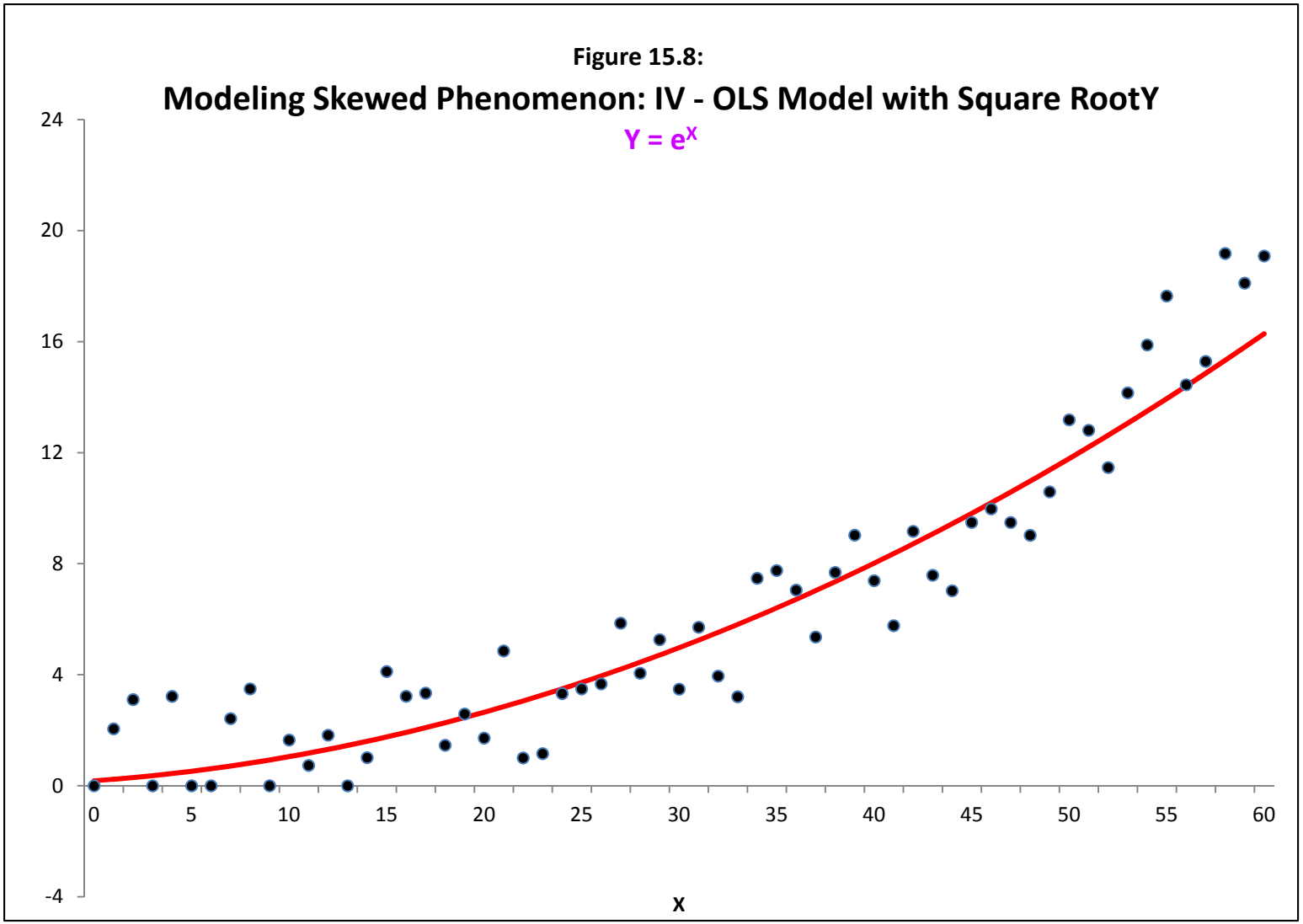
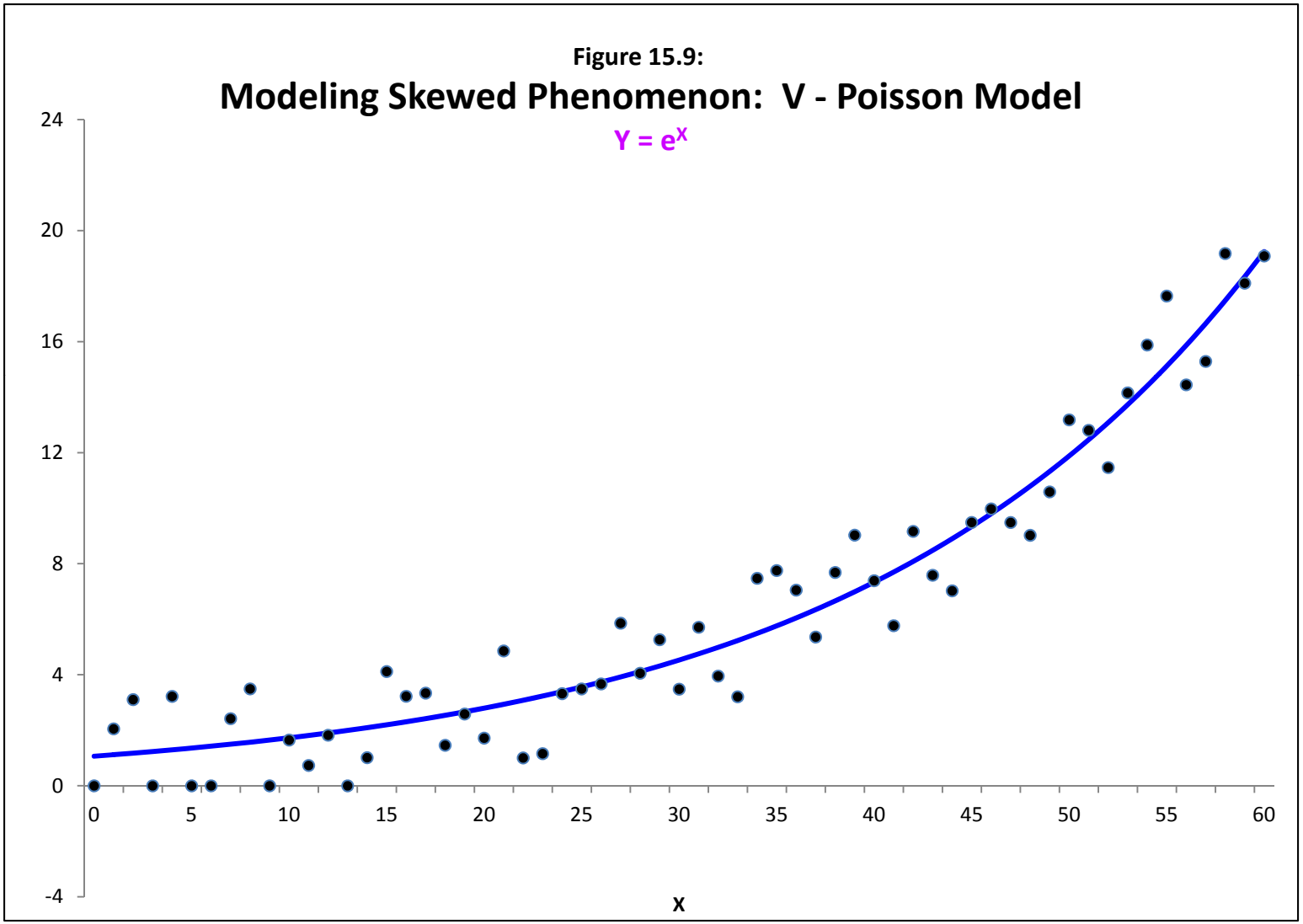


Figure 15.9:


Modeling Skewed Phenomenon: V - Poisson Model
Y = e^X
24
20
16
12
8
4
0
-4
0
5
10
15
20
25
30
35
40
45
50
55
60
X

Keep in mind that this was a created distribution where the data points were distributed equally across the X spectrum and where the errors were constant throughout (homosceadstic). With real data, a count variable (e.g., number of crimes, income) will usually be highly skewed with most observations having low values with a small percentage having high values (or vice versa such as with distance traveled) and the errors will typically increase with the value of the dependent variable.

Diagnostic Tests and OLS

To evaluate skewness and other violations of assumptions of a linear model, it is essential to examine various diagnostics about the dependent variable. The regression module has a set of diagnostic tests for evaluating the characteristics of the data and the most appropriate model to use. There is a diagnostics box on the Regression I page (see Figure 20.1 in chapter 20). Diagnostics are provided on:

1. The minimum and maximum values for the dependent and independent variables
2. Skewness in the dependent variable
3. Spatial autocorrelation in the dependent variable
4. Estimated values for the distance decay parameter – alpha, for use in the Poisson-Gamma-CAR model
5. Multicolinarity among the independent variables

Minimum and Maximum Values for the Variables

The minimum and maximum values of both the dependent and independent variables are listed. A user should look for ineligible values (e.g., -1) as well as variables that have a very high range. The MLE routines are sensitive to variables with very large ranges. To minimize the effect, variables are internally scaled when being run (by dividing by their mean) and then re-scaled for output. Nevertheless, variables with extreme ranges in values and especially variables where there are a few observations with extreme values can distort the results for models.[4] A user would be better choosing a more balanced variable than using one where one or two observations determines the relationship with the dependent variable.

[4] For example, in Excel, two columns of random numbers from 1 to 10 were listed in 99 rows to represent two variables X1 and X2. The correlation between these two variables over the 99 rows (observations) was -0.03. An additional row was added and the two variables given a value of 100 each for this row. Now, the correlation between these two variables increased to 0.89! The point is, one or two extreme values can distort a statistical relationship.

Skewness Tests

As we have discussed, skewness in a variable can distort a normal model by allowing high values to be underestimated while allowing low or middle-range values to be overestimated. For this reason, a Poisson-type model is preferred over the normal for highly skewed variables.

The diagnostics utility tests for skewness using two different measures. First, the utility outputs the "***g"*** statistic (Microsoft, 2003):

$$g = \frac{n}{(n-1)(n-2)} \sum_i [(X_i - \bar{X}) / s]^3 \tag{15.26}$$

where n is the sample size, X_i is observation i, $\bar{X}$ is the mean of X, and s is the sample standard deviation (corrected for degrees of freedom). The sample standard deviation is defined as:

$$s = \sqrt{\sum_i \frac{(X_i - \bar{X})^2}{(n-1)}} \tag{15.27}$$

The standard error of skewness (SES) can be approximated by (Tabachnick and Fidell, 1996):

$$SES = \sqrt{\frac{6}{n}} \tag{15.28}$$

An approximate Z-test can be obtained from:

$$Z(g) = \frac{g}{SES} \tag{15.29}$$

Thus, if Z is greater than +1.96 or smaller than -1.96, then the skewness is significant at the $p \leq .05$ level.

An example is the number of crimes originating in each traffic analysis zone within Baltimore County in 1996. The summary statistics were:

$\bar{X}$ = 75.108
s = 96.017
n = 325

$\sum_i [(X_i - \bar{X}) / s]^3 = 898.391$

Therefore,

$$g = \frac{325}{324 * 323} * 898.391 = 2.79$$

$$SES = \sqrt{\frac{6}{325}} = 0.136$$

$$Z(g) = \frac{2.79}{0.136} = 20.51$$

The Z of the g value shows the data are highly skewed.

The second skewness measure is a ratio of the simple variance to the simple mean. While this ratio had not been adjusted for any predictor variables, it is usually a good indicator of skewness. Ratios greater than about 2:1 should make the user cautious about using a normal model.

If either measure indicates skewness, *CrimeStat* prints out a message indicating the dependent variable appears to be skewed and that a Poisson-type model should be used.

Testing for Spatial Autocorrelation in the Dependent Variable

A fourth test that is available is a test for spatial autocorrelation in the dependent variable. It will be discussed in the spatial regression section (Chapter 19).

Multicollinearity Tests

The fifth type of diagnostic test is for multicollinearity among the independent predictors. As we have discussed in this chapter, one of the major problems with many regression models, whether MLE or MCMC, is multicollinearity among the independent variables.

To assess multicollinearity, the pseudo-tolerance test is presented for each independent variable. This was discussed above in the chapter (see equation 15.18).

MCMC Version of Normal (OLS)

There is also a Markov Chain Monte Carlo (MCMC) version of the OLS model which assumes the dependent variable is normally distributed. This will be discussed in chapter 17 on Markov Chain Monte Carlo estimation and in chapter 19 on spatial regression modeling.

References

Abraham, B. & Ledolter, J. (2006). *Introduction to Regression Modeling*. Thompson Brooks/Cole: Belmont, CA.

Berk, K. N. (1977). "Tolerance and condition in regression computations", *Journal of the American Statistical Association*, 72 (360), 863-866.

Draper, N. & Smith, H. (1981). *Applied Regression Analysis, Second Edition*. John Wiley & Sons: New York.

H-GAC (2010). Transportation and air quality program, *Houston-Galveston Area Council*. http://www.h-gac.com/taq/.

Hilbe, J. M. (2008). *Negative Binomial Regression (with corrections)*. Cambridge University Press: Cambridge.

Kanji, G. K. (1993). *100 Statistical Tests*. Sage Publications: Thousand Oaks, CA.

Miaou, S. P. (1996). *Measuring the Goodness-of-Fit of Accident Prediction Models*. FHWA-RD-96-040. Federal Highway Administration, U.S. Department of Transportation: Washington, DC.

Microsoft (2003). "SKEW - skewness function", *Microsoft Office Excel 2003, Microsoft: Redmond, WA*.

Myers, R. H. (1990) Classical and Modern Regression with Applications, 2nd edition, Duxbury Press, Belmont, CA.

Oh, J., Lyon, C., Washington, S., Persaud, B., & Bared, J. (2003). "Validation of FHWA crash models for rural intersections: lessons learned". *Transportation Research Record 1840*, 41-49.

StatSoft (2010). "Tolerance", *StatSoft Electronic StatisticsTextbook,* StatSoft:Tulsa, OK. http://www.statsoft.com/textbook/statistics-glossary/t/button/t/

Tabachnick, B. G. & Fidell, L. S. (1996). *Using Multivariate Statistics* (3rd ed). Harper Collins: New York.

Train, K. (2009). *Discrete Choice Methods with Simulation* (2nd edition). Cambridge University Press: Cambridge.

References (continued)

Venables, W.N. & Ripley, B. D. (1997). *Modern Applied Statistics with S-Plus (second edition)*. Springer-Verlag: New York.

Wikipedia (2010b). "Maximum likelihood", *Wikipedia*. http://en.wikipedia.org/wiki/Maximum_likelihood. Accessed March 12, 2010.

Chapter 16:

Poisson Regression Modeling[1]

Dominique Lord
Zachry Dept. of
Civil Engineering
Texas A & M University
College Station, TX

Byung-Jung Park
Korea Transport Institute
Goyang, South Korea

Ned Levine
Ned Levine & Associates
Houston, TX

[1] The code for the Poisson and Negative Binomial models was developed by Ian Cahill of Cahill Software, Edmonton, Alberta, based on his MLE++ software package (http://cahillsoftware.com/2122/index.html). The integration and stepwise procedures were developed by us with the programming by Ms. Haiyan Teng of Houston.

Table of Contents

Chapter 16:
Poisson Regression Modeling

In this chapter, we discuss Poisson models for estimating count variables.

Count Data Models

In chapter 15, we examined Ordinary Least Squares (OLS) regression models. We showed that these models were bound by some strong assumptions of a normally-distributed dependent variable and errors that were normal and constant. We then demonstrated that OLS models are inadequate for describing skewed distributions, particularly counts. Given that crime analysis usually involves the analysis of counts, this is a serious deficiency.

Poisson Regression

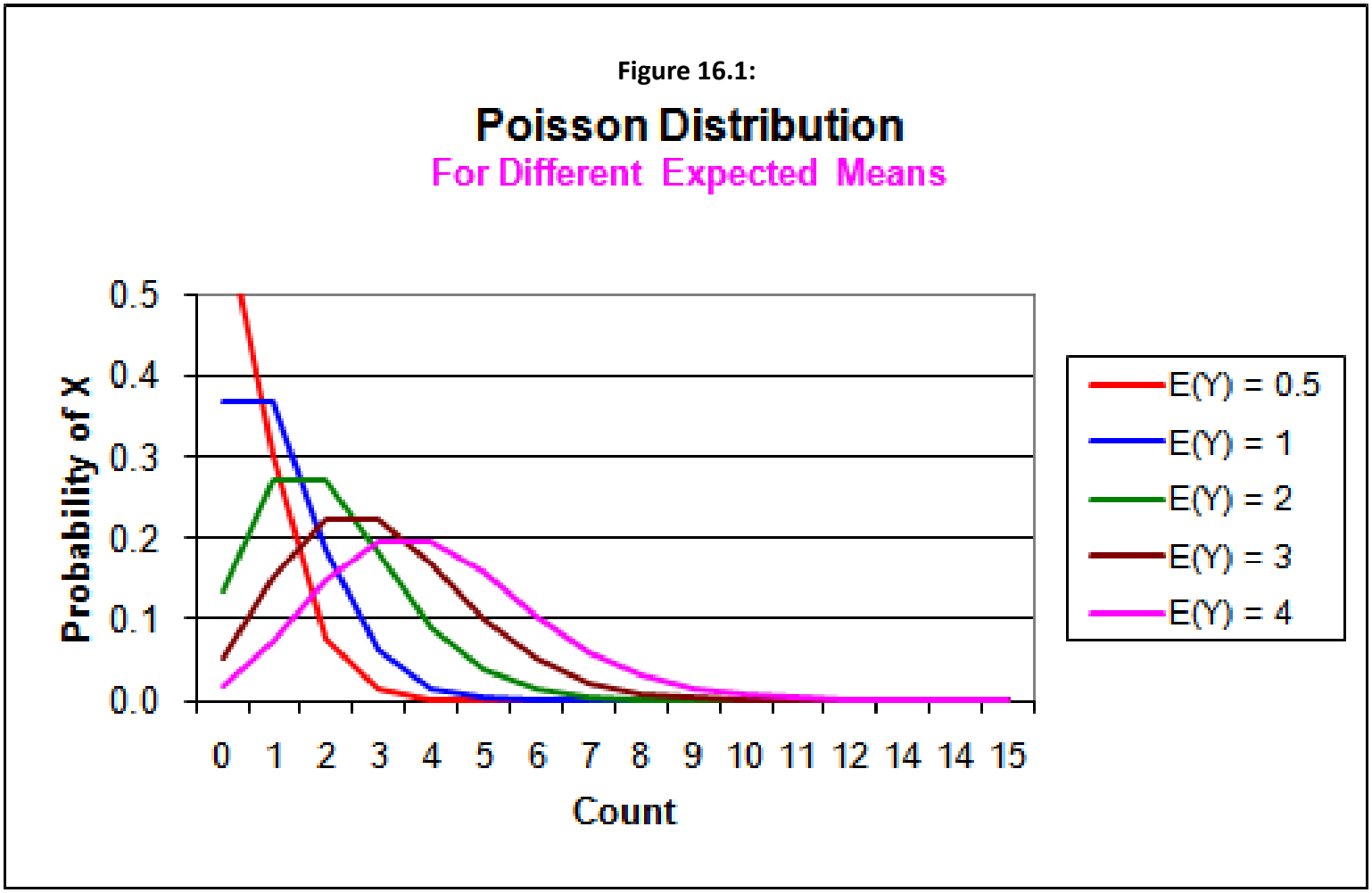
Consequently, we turn to count data models, in particular the Poisson family of models. This family is part of the generalized linear models (GLMs), in which the OLS normal model described above is a special case (McCullagh & Nelder, 1989). Poisson regression is a modeling method that overcomes some of the problems of traditional regression in which the errors are assumed to be normally distributed (Cameron & Trivedi, 1998). In the model, the number of events is modeled as a Poisson random variable with a probability of occurrence being:

$$\text{Prob}(y_i) = \frac{e^{-\lambda}\lambda^{y_i}}{y_i!} \tag{16.1}$$

where y_i is the count for one group or class, i, λ is the mean count over all groups, and e is the base of the natural logarithm. The distribution has a single parameter, λ, which is both the mean and the variance of the function.

The "law of rare events" assumes that the total number of events will approximate a Poisson distribution *if* an event occurs in any of a large number of trials but the probability of occurrence in any given trial is small and assumed to be constant (Cameron & Trivedi, 1998). Thus, the Poisson distribution is very appropriate for the analysis of rare events such as crime incidents (or motor vehicle crashes or uncommon diseases or any other rare event). The Poisson model is not particularly good if the probability of an event is more balanced; for that, the normal distribution is a better model as the sampling distribution will approximate normality with increasing sample size. Figure 16.1 illustrates the Poisson distribution for different expected means.

Figure 16.1:


Poisson Distribution
For Different Expected Means
Probability of X
0.5
0.4
0.3
0.2
0.1
0.0
0 1 2 3 4 5 6 7 8 9 10 11 12 14 14 15
Count
E(Y) = 0.5
E(Y) = 1
E(Y) = 2
E(Y) = 3
E(Y) = 4

The Poisson distribution is part of a large family known as the exponential family of distributions (McCullagh & Nelder, 1989). The probability distribution for this family is expressed as (Hilbe, 2008):

$$f(y_i: \mu, \Phi) = e^{\left\{\frac{y_i\theta_i - b(\theta_i)}{\alpha(\Phi)} + C(y_i:\Phi)\right\}} \tag{16.2}$$

where θ_i is the canonical parameter or *link* function for observation i, b(θ_i) is the cumulant for observation i, α(Φ) is the scale parameter which is set to one in discrete and count models, and C(y_i: Φ) is a normalization (scaling) term that guarantees that the probability function sums to 1. This family of functions is unique in that the first and second derivatives of the cumulant, with respect to θ, produce the mean and variance function (Hilbe, 2008). All members of the class of generalized linear models can be converted to the exponential form.

Since the Poisson family is a member of the exponential family, the mean can be modeled as a function of some other variables (the independent variables). Given a set of observations on one or more independent variables, $\mathbf{x}_i^T = (1, x_{1i}, \cdots, x_{Ki})$, the *conditional mean* of y_i can be specified as an exponential function of the *x*'s:

$$E(y_i \mid \mathbf{x}_i) = \lambda_i = e^{\mathbf{x}_i^T\boldsymbol{\beta}} \tag{16.3}$$

where *i* is an observation, $\mathbf{x}_i^T$ is a set of independent variables including an intercept, $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots, \beta_K)^T$ are a set of coefficients, and *e* is the base of the natural logarithm. Equation 16.3 can be also written as:

$$\ln(\lambda_i) = \mathbf{x}_i^T\boldsymbol{\beta} = \beta_0 + \sum_{k=1}^{K} \beta_k x_{ki} \tag{16.4}$$

where each independent variable, k, is multiplied by a coefficient, β_k, and is added to a constant, β_0. In expressing the equation in this form, we have transformed it using a ***link*** function, the link being the log-linear relationship. As discussed above, the Poisson model is part of the GLM framework in which the functional relationship is expressed as a linear combination of predictive variables. This type of model is sometimes known as a ***loglinear*** model as the natural log of the mean is a linear function of *K* independent variables and an intercept.

However, we will refer to it as a *Poisson model*. In more familiar notation, this is

$$\ln(\lambda_i) = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_K x_{Ki} \tag{16.5}$$

For the Poisson model, the log-likelihood is:

$$\ln L = \sum_{i=1}^{N} \left[-\lambda_i + y_i \ln(\lambda_i) - \ln y_i! \right] \tag{16.6}$$

where $\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta})$ is the conditional mean for zone i, and y_i is the observed number of events for observation i. Anselin provides a more detailed discussion of these functions in Appendix B. The data are assumed to reflect the Poisson model and the variance equals the mean. Therefore, it is expected that the residual errors should increase with the conditional mean. That is, there is inherent heteroscedasticity in a Poisson model (Cameron & Trivedi, 1998). This is different than a normal model where the residual errors are expected to be constant.

The model is estimated using a maximum likelihood (MLE) procedure, typically the Newton-Raphson method or, occasionally, using Fisher scores (Wikipedia, 2010; Cameron & Trivedi, 1998). In Appendix B, Anselin presents a more formal treatment of both the normal and Poisson regression models including the methods by which they are estimated.

Advantages of the Poisson Regression Model

The Poisson model overcomes some of the problems of the normal model. First, the Poisson model has a minimum value of 0. It will not predict negative values. This makes it ideal for a distribution in which the mean or the most typical value is close to 0. Second, the Poisson is a fundamentally skewed model; that is, it is data characterized with a long 'right tail'. Again, this model is appropriate for counts of rare events, such as crime incidents.

Third, because the Poisson model is estimated by the maximum likelihood method, the estimates are adapted to the actual data. In practice, this means that the sum of the predicted values is virtually identical to the sum of the input values, with the exception of a very slight rounding off error.

Fourth, compared to the normal model, the Poisson model generally gives a better estimate of the counts for each record. The problem of over- or underestimating the number of incidents for most records with the normal model is usually lessened with the Poisson. When the residual errors are calculated, generally the Poisson has a lower total error than the normal model, as was illustrated in chapter 15.

In short, the Poisson model has some desirable statistical properties that make it very useful for predicting crime incidents.

Example of Poisson Regression

Using the same Houston burglary database as in chapter 15, we estimate a Poisson model of the two independent predictors of burglaries (Table 16.1).

Likelihood Statistics

Log-likelihood

The summary statistics are quite different from the normal model. In the *CrimeStat* implementation, there are five separate statistics about the likelihood, representing a joint probability function that is maximized. First, there is the log-likelihood (L). The likelihood function is the joint (product) density of all the observations given values for the coefficients and the error variance. The log-likelihood is the log of this product or the sum of the individual densities. Because the function maximizes a probability, which is always between 0 and 1, the log-likelihood is ***always*** negative with a Poisson model.

Note that in comparing two models, the model with the **smallest** log-likehood will fit the data better assuming that the data set and the dependent variable are the same. For example, if one model has a log-likelihood of -4,000 and a second model on the same data set and dependent variable has a log-likelihood of -5,000, the first model is better because it has a *smaller* log-likelihood than the second model. While this is unintuitive, it makes sense in terms of probability theory. If the probability of the first model is 0.6 and that of the second 0.4, then the log-likelhood of the first model will be -0.51 and that of the second -.91. Since a likelihood is the product of the densities of each individual case (and, therefore, the log-likelihood is the sum of the individual logarithms), in practice the log-likelihood is proportional to the probability.

Aikaike Information Criterion (AIC)

Second, the Aikaike Information Criterion (AIC) adjusts the log-likelihood for degrees of freedom since adding more variables will always increase the log-likelihood. It is defined as:

$$AIC = -2L + 2(K+1) \quad (16.7)$$

where L is the log-likelihood and K is the number of independent variables. The model with the lowest AIC is 'best'.

Table 16.1:

Predicting Burglaries in the City of Houston: 2006

Poisson Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**		
N:	1,179		
Df:	1,175		
Type of regression model:	Poisson		
Method of estimation:	Maximum likelihood		
Likelihood statistics			
Log-likelihood:	-13,639.5		
AIC:	27,287.1		
BIC/SC:	27,307.4		
Deviance:	23,021.4	p:	0.0001
Pearson Chi-square:	24,804.4	p:	0.0001
Model error estimates			
Mean absolute deviation:	16.0		
1st (highest) quartile:	33.9		
2nd quartile:	7.3		
3rd quartile:	8.8		
4th (lowest) quartile:	13.9		
Mean squared predicted error:	714.2		
1st (highest) quartile:	2,351.8		
2nd quartile:	203.7		
3rd quartile:	99.8		
4th (lowest) quartile:	206.7		
Dispersion tests			
Adjusted deviance:	19.6	p: 0.0001	
Adjusted Pearson Chi-Square:	21.1	p: 0.0001	
Dispersion multiplier:	21.1	p: 0.0001	Inverse dispersion multiplier: 0.05

Predictor	DF	Coefficient	Stand Error	Tolerance	VIF	Z-value	p
INTERCEPT	1	2.8745	0.014	-	-	212.47	0.001
HOUSEHOLDS	1	0.0006	0.000004	0.994	1.006	146.24	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.000009	0.00000	0.994	1.006	-28.68	0.001

Bayes Information Criterion (BIC/SC)

Third, another measure which is very similar is the *Bayes Information Criterion* (BIC/SC, sometimes called *Schwartz Criterion*), which is defined as:

$$\text{BIC/SC} = -2L+[(K+1)\ln(N)] \tag{16.8}$$

These two measures penalize the number of parameters added in the model, and reverse the sign of the log-likelihood (L) so that the statistics are more intuitive. The model with the lowest BIC/SC value is 'best'.

Deviance

Fourth, a decision about whether the Poisson model is appropriate can be based on the statistic called the *deviance* which is defined as:

$$Dev = 2(L_F - L_M) = 2\sum_{i=1}^{N}\left[y_i \ln\left(\frac{y_i}{\hat{\lambda}_i}\right) - y_i - \hat{\lambda}_i \right] \tag{16.9}$$

where L_F is the log-likelihood that would be achieved if the model gave a perfect fit and L_M is the log-likelihood of the model under consideration. If the latter model is correct, the deviance (*Dev*) is approximately χ^2 distributed with degrees of freedom equal to $N-(K+1)$. A value of the deviance greatly in excess of $N-(K+1)$ suggests that the model is over-dispersed due to missing variables or non-Poisson form. This statistic is sometimes called the G^2 statistic (Bishop, Feinberg, & Holland, 1975). The deviance has N-*K*-1 degrees of freedom where *K* is the number of parameters estimated (including the constant).

Pearson Chi-square

Fifth, there is the Pearson Chi-square statistic which is defined by

$$Pearson-\chi^2 = \sum_{i=1}^{N}\frac{(y_i-\hat{\mu}_i)^2}{VAR(y_i)} \tag{16.10}$$

If the mean and the variance are properly specified, then $E\left[\sum_{i=1}^{N}(y_i - \frac{\mu_i)^2}{VAR(y_i)})\right] = N$ (Cameron and Trivedi, 1998). Values closer to N (the sample size) show a better fit. The Pearson Chi-square has N-*K*-1 degrees of freedom where *K* is the number of parameters

estimated (including the constant). Note, that the expected value depends on the variance function, which we will discuss below.

Model Error Estimates

Next, there are two statistics that measure how well the model fits the data (goodness-of-fit). Mean Absolute Deviation (MAD) and Mean Squared Predicted Error (MSPE) were defined in Chapter 15. Comparing these with the results of the normal model (Table 15.1), it can be seen that the overall MAD and MSPE are slightly worse than for the normal model, though much better than with the log transformed linear model (Table 15.4). Comparing the four quartiles, it can be seen that for three of the four quartiles the normal model had slightly better MAD and MSPE scores than for the Poisson but the differences were not great.

Dispersion Tests

The remaining four summary statistics measure *dispersion*. A more extensive discussion of dispersion is given later in the chapter. But, very simply, in the Poisson framework, the variance should equal the mean. These statistics indicate the extent to which the variance exceeds the mean.

First, the *adjusted deviance* is defined as the deviance divided by the degrees of freedom (N-K-1); a value closer to 1 indicates a satisfactory goodness-of-fit. Usually, values greater than 1 indicate signs of over-dispersion.

Second, the *adjusted Pearson Chi-square* is defined as the Pearson Chi-square divided by the degrees of freedom; again, a value closer to 1 indicates a satisfactory goodness-of-fit.

Third, the *dispersion* multiplier, γ, measures the extent to which the conditional variance exceeds the conditional mean (conditional on the independent variables and the intercept term) and is defined by $Var(y_i) = \lambda_i + \gamma\lambda_i^2$. The Z-test of the dispersion multiplier indicates whether the amount of dispersion is significantly greater than that assumed by the Poisson model (Hilbe, 2008). The test is:

$$Z = \frac{(\sum(y_i - \mu_i)^2 - y_i)}{\sum \mu_i \sqrt{2}} \qquad (16.11)$$

where y_i is the observed value of Y and μ_i is the predicted value of Y. The statistic is a test of *over-dispersion*, that the conditional variance is *greater* than the conditional mean. A significant value for Z indicates that the assumption of equi-dispersion of the conditional variance is

rejected and the model should be estimated as a negative binomial or lognormal for over-dispersion.

In some cases, there may be *under-dispersion*, that is where the conditional variance is less than the conditional mean. In this case, a Poisson with linear correction should be used. Unfortunately, the Z-test will identify that as being not significant. We are not aware of a good test for under-dispersion and the user will have to use judgment.

Fourth, the *inverse dispersion multiplier* (ψ) is simply the reciprocal of the dispersion multiplier ($\psi = 1/\gamma$); some users are more familiar with it in this form.

As seen in Table 16.1, the four dispersion statistics are much greater than 1 and indicate *over-dispersion*. In other words, the conditional variance is greater – in this case, much greater, than the conditional mean. The 'pure' Poisson model (in which the variance is supposed to equal the mean) is not an appropriate model for these data.

Individual Coefficient Statistics

Finally, the signs of the coefficients are the same as for the normal and transformed normal models, as would be expected. The relative strengths of the variables, as seen through the Z-values, are also approximately the same.

In short, the Poisson model has produced results that are an alternative to the normal model. While the likelihood statistics indicate that, in this instance, the normal model is slightly better, the Poisson model has the advantage of being theoretically sounder. In particular, it is not possible to get a minimum predicted value less than zero (which is possible with the normal model) and the sum of the predicted values will always equal the sum of the input values (which is rarely true with the normal model). With a more skewed dependent variable, the Poisson model will usually fit the data better than the normal as well.

Problems with the Poisson Regression Model

On the other hand, the Poisson model is not perfect. The primary problem is that count data are usually *over-dispersed*.

Over-dispersion in the Residual Errors

In the Poisson distribution, the mean equals the variance. In a Poisson regression model, the mathematical function, therefore, equates the conditional mean (the mean controlling for all the predictor variables) with the conditional variance. However, most actual distributions have a

high degree of skewness, much more than are assumed by the Poisson distribution (Cameron & Trivedi, 1998; Mitra & Washington, 2007).

As an example, figure 16.2 shows the distribution of Baltimore County and Baltimore City crime origins and Baltimore County crime destinations by TAZ. For the origin distribution, the ratio of the variance to the mean is 14.7; that is, the variance is 14.7 times that of the mean! For the destination distribution, the ratio is 401.5!

In other words, the simple variance is many times greater than the mean. We have not yet estimated some predictor variables for these variables, but it is probable that even when this is done the conditional variance will far exceed the conditional mean. Many real-world count data are similar to this; the variance will usually be much greater than the mean (Lord, 2006) although, occasionally, the variance can be smaller than the conditional mean (Lord, 2010). What this means in practice is that the residual errors - the difference between the observed and predicted values for each zone, will be greater than what is expected. The Poisson model calculates a standard error as if the variance equals the mean. Thus, the standard error will be underestimated using a Poisson model and, therefore, the significance tests (the coefficient divided by the standard error) will be greater than they really should be. In a Poisson multiple regression model, we might end up selecting variables that really should not be selected because we think they are statistically significant when, in fact, they are not (Park & Lord, 2007).

Under-dispersion in the Residual Errors

There are also cases where the conditional variance is less than the conditional mean (under-dispersion). This happens sometimes with crime data. For example, in an analysis of drunk driving crashes in Baltimore County, we found that the modeled variance was substantially less than the modeled mean (Levine & Canter, 2011). In both cases, one needs to correct the estimated standard error from the Poisson model.

To visualize over- and under-dispersion, Figure 16.3 shows three different skewed distributions, over-dispersed, equi-dispersed (Poisson), and under-dispersed. These are based on the variance-to-mean ratios of the raw data. Note that the over-dispersed distribution is extremely skewed while the under-dispersed distribution is mildly skewed. Still, with under-distribution, one cannot assume a normal distribution because it will still underestimate the high values of the dependent variable.

Also, the actual dispersion is conditional on the independent variables (i.e., after the model has been run). However, Cameron and Trivedi (1998) suggest that if the raw variance-to-mean ratio is less than 2.0, most likely the conditional variance will be less than the conditional mean.

Figure 16.2:

Distribution of Crime Origins and Destinations: Baltimore County, MD: 1993-1997

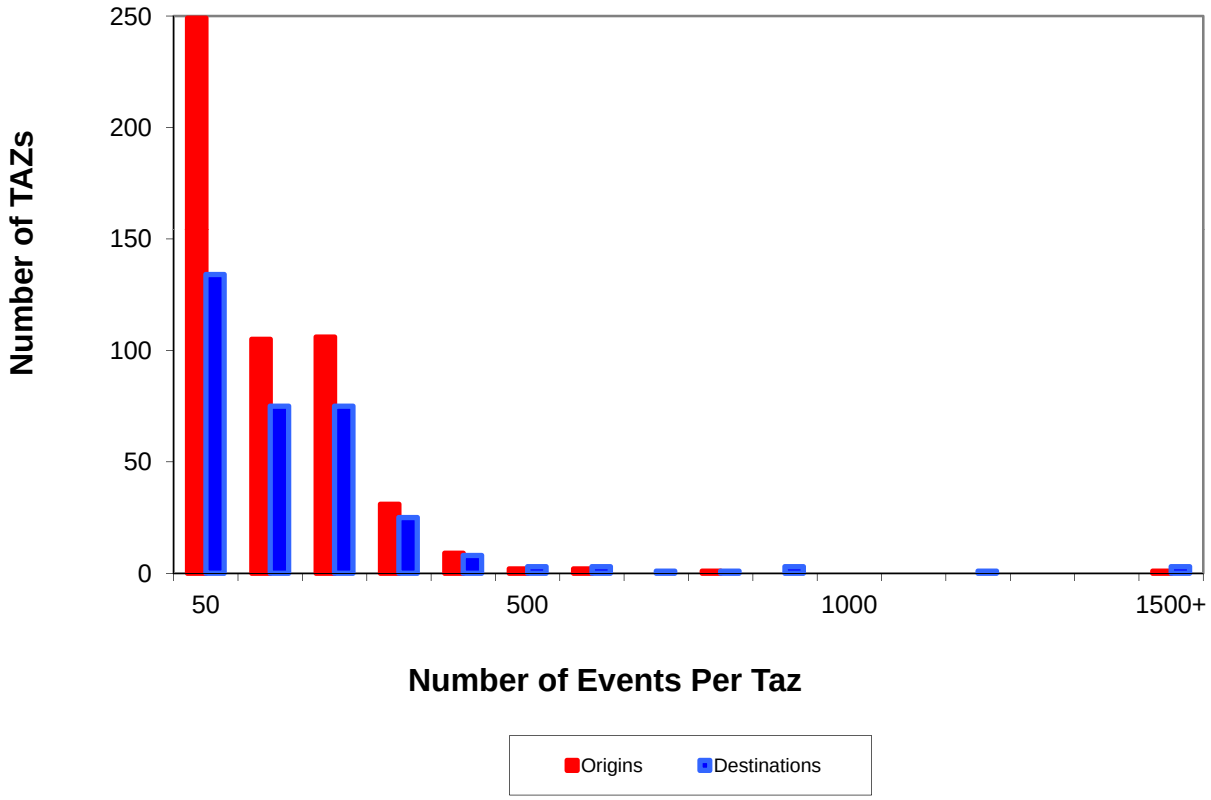
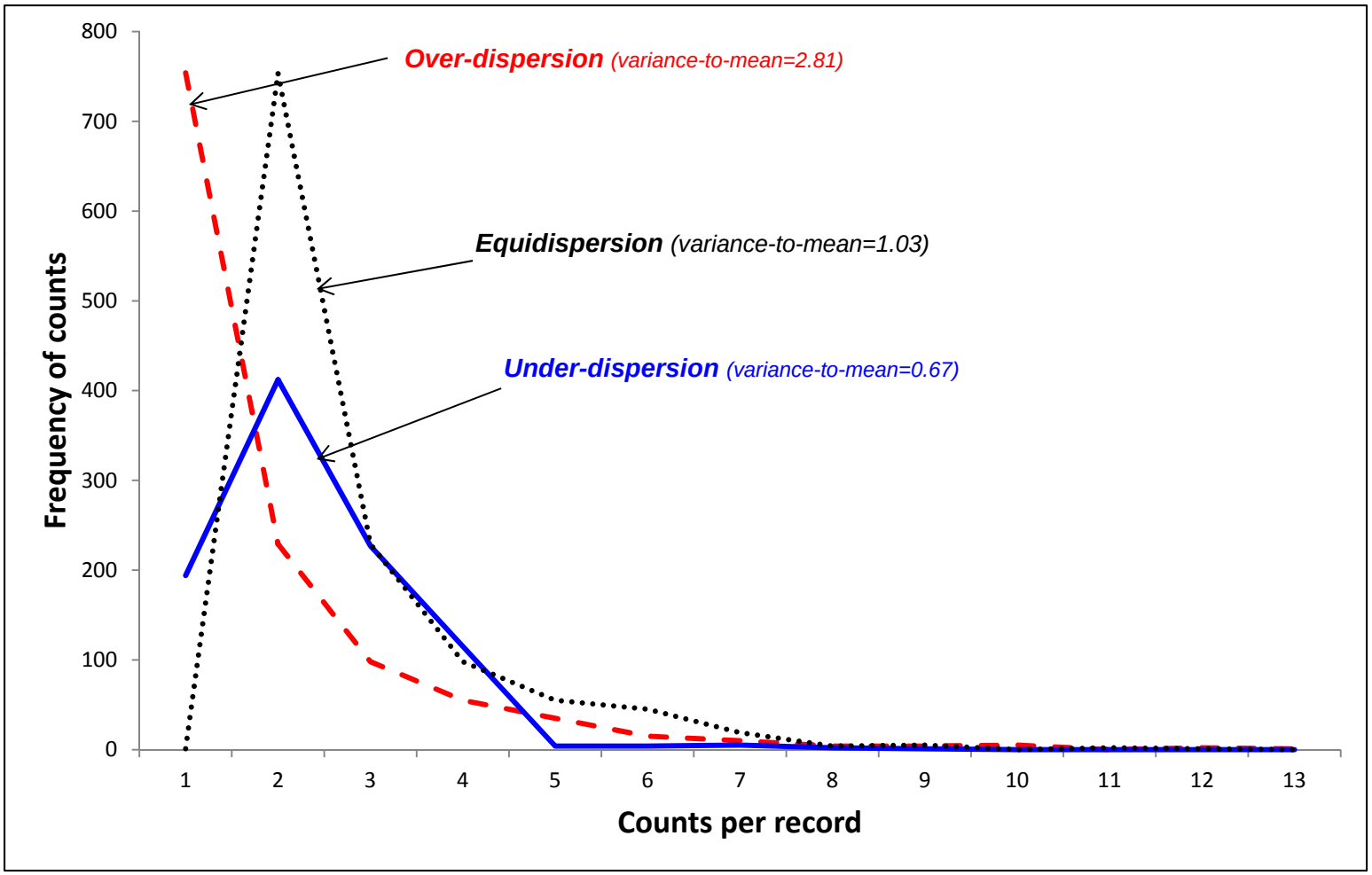


Figure 16.3:

Skewed Distributions and Type of Dispersion



Poisson Regression with Linear Dispersion Correction

There are a number of methods for correcting the over-dispersion in a count model. Most of them involve modifying the assumption of the conditional variance equal to the conditional mean. The first is a simple linear correction known as the *linear negative binomial* (or NB1 model; Cameron & Trivedi, 1998, 63-65). The variance of the function is assumed to be a linear multiplier of the mean. The conditional variance is defined as:

$$\omega_i = V[y_i \mid \mathbf{x}_i] \tag{16.12}$$

where $V[y_i \mid \mathbf{x}_i]$ is the variance of y_i given the independent variables.

The conditional variance is then a function of the mean:

$$\omega_i = \lambda_i + \tau\lambda_i^p \tag{16.13}$$

where τ is the *dispersion parameter* and p is a constant (usually 1 or 2). In the case where p is 1, the equation simplifies to:

$$\omega_i = \lambda_i + \tau\lambda_i \tag{16.14}$$

This is the NB1 correction. In the special case where $\tau = 0$, the variance becomes equal to the mean (the Poisson model). The model is estimated in two steps. First, the Poisson model is fitted to the data and the degree of over- (or under) dispersion is estimated. The dispersion parameter is defined as:

$$\hat{\tau} = 1/\hat{\psi} = \frac{1}{N-K-1}\sum_{i=1}^{N}\frac{(y_i - \hat{\lambda}_i)^2}{\hat{\lambda}_i} - 1 \tag{16.15}$$

where *N* is the sample size, *K* is the number of independent variables, Y_i is the observed number of events that occur in observation (or zone) *i*, and $\hat{\lambda}_i$ is the predicted number of events for observation (or zone) *i*. The test is similar to an average chi-square in that it takes the square of the residuals $(y_i - \hat{\lambda}_i)^2$ and divides it by the predicted values, and then averages it by the degrees of freedom. The dispersion parameter is a standardized number. A value greater than 0 indicates over-dispersion while a value less than 0 indicates under-dispersion. A value of 0 indicates *equidispersion* (or the variance equals the mean). The dispersion parameter can also be estimated based on the deviance.

In the second step, the Poisson standard error is multiplied by the square root of the dispersion parameter to produce an *adjusted standard error*:

$$SE_{adj} = SE \times \sqrt{\hat{\tau}} \tag{16.16}$$

The new standard error is then used with the t-test to produce an adjusted t-value. This adjustment is found in most Poisson regression packages using a Generalized Linear Model (GLM) approaches (McCullagh and Nelder, 1989, 200). Cameron & Trivedi (1998) have shown that this adjustment produces results that are virtually identical to that of the negative binomial, but involving fewer assumptions. *CrimeStat* includes an NB1 correction and is called *Poisson with linear correction*.

Example of Poisson Model with Linear Dispersion Correction (NB1)

Table 16.2 shows the results of running the Poisson model with the linear dispersion correction. The likelihood statistics are the same as for the simple Poisson model (Table 16.1) and the coefficients are identical. The dispersion parameter, however, has now been adjusted to be 1.0. This affects the standard errors, which are now greater. In the example, the two independent variables are still statistically significant, but the Z-values are smaller.

Poisson-Gamma (Negative Binomial) Regression

A second type of dispersion correction involves a mixed function model. Instead of simply adjusting the standard error by a dispersion correction, different assumptions are made for the mean and the variance (dispersion) of the dependent variable. In the *negative binomial* model, the number of observations (Y_i) is assumed to follow a Poisson distribution but the mean (λ_i) follows a Gamma distribution (Lord, 2006; Cameron & Trivedi, 1998, 62-63; Venables & Ripley, 1997, 242-245). This is frequently called an NB2 model.

Mathematically, the negative binomial distribution is one derivation of the binomial distribution in which the sign of the function is negative, hence the term *negative binomial* (for more information on the derivation, see Wikipedia, 2010). For our purposes, it is defined as a mixed distribution with a Poisson mean and a one parameter Gamma dispersion function having the form:

$$f(y_i / \theta_i) = \frac{e^{-\theta_i} \theta_i^{y_i}}{y_i!} \tag{16.17}$$

Table 16.2:

Predicting Burglaries in the City of Houston: 2006
Poisson with Linear Dispersion Correction Model (NB1)

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**			
N:	1,179			
Df:	1,175			
Type of regression model:	Poisson with linear dispersion correction			
Method of estimation:	Maximum likelihood			
Likelihood statistics				
Log-likelihood:	-13,639.5			
AIC:	27,287.1			
BIC/SC :	27,307.4			
Deviance:	12,382.5	p: 0.0001		
Pearson Chi-square:	12,402.2	p: 0.0001		
Model error estimates				
Mean absolute deviation:	16.0			
1st (highest) quartile:	33.9			
2nd quartile:	7.3			
3rd quartile:	8.8			
4th (lowest) quartile:	13.9			
Mean squared predicted error:	714.2			
1st (highest) quartile:	2,351.8			
2nd quartile:	203.7			
3rd quartile:	99.8			
4th (lowest) quartile:	206.7			
Dispersion tests				
Adjusted deviance:	10.5	P: 0.001		
Adjusted Pearson Chi-Square:	10.6	p: 0.001		
Dispersion multiplier:	1.0	p: n.s.	Inverse dispersion multiplier:	1.0

Predictor	DF	Coefficient	Stand Error	Tolerance	VIF	Z-value	p
INTERCEPT	1	2.87452	0.062	-	-	46.26	0.001
HOUSEHOLDS	1	0.00059	0.00002	0.994	1.006	31.84	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.000009	0.000001	0.994	1.006	-6.24	0.001

where

$$\theta_i = e^{\beta_0 + (\Sigma \beta_i x_i) + \varepsilon_i} \tag{16.18}$$

$$\theta_i = e^{\beta_0 + (\Sigma \beta_i x_i)} e^{\varepsilon_i} \tag{16.19}$$

$$\theta_i = \mu_i \nu_i \tag{16.20}$$

and where θ_i is a function of a one-parameter gamma distribution where the parameter, τ, is greater than 0 (ignoring the subscripts):

$$h(y/\mu,\tau) = \frac{\Gamma(\tau^{-1}+y)}{\Gamma(\tau^{-1})\Gamma(y+1)}\left(\frac{(\tau^{-1})}{\tau^{-1}+\mu}\right)^{\tau^{-1}}\left(\frac{\mu}{\tau^{-1}+\mu}\right)^{y} \tag{16.21}$$

The model is used traditionally with integer (count) data though it can also be applied to continuous (real) data. Sometimes the integer model is called a *Pascal* model while the real model is called a *Polya* model (Wikipedia, 2010; Springer, 2010). Boswell and Patil (1970) argued that there are at least 12 distinct probabilistic processes that can give rise to the negative binomial function including heterogeneity in the Poisson intensity parameter, cluster sampling from a population which is itself clustered, and the probabilities that change as a function of the process history (i.e., the occurrence of an event breeds more events). The interpretation we adopt here is that of a heterogeneous population with different observations coming from different sub-populations, and the Gamma distribution is the mixing variable.

Because both the Poisson and Gamma functions belong to the single-parameter exponential family of functions and are convex in shape (increasing smoothly up to a peak and then decreasing smoothly), they can be solved by the maximum likelihood method. The mean is always estimated as a Poisson function. However, there are slightly different parameterizations of the variance function (Hilbe, 2008). In the original derivation by Greenwood and Yule (1920), the conditional variance was defined as:

$$\omega_i = \mu_i + \mu_i^2/\psi \tag{16.22}$$

whereupon ψ (Psi) became known as the *inverse dispersion parameter* (McCullagh & Nelder, 1989).

However, in more recent years, the conditional variance was defined within the Generalized Linear Models tradition as a direct adjustment of the squared Poisson mean, namely:

$$\omega_i = \mu_i + \tau \mu_i^2 \quad (16.23)$$

where the variance is now a quadratic function of the Poisson mean (i.e., p is 2 in formula 16.13) and τ is called the *dispersion multiplier*. This is the formulation proposed by Cameron & Trivedi (1998; pp. 62-63). That is, it is assumed that there is an unobserved variable that affects the distribution of the count so that some observations come from a population with higher expected counts whereas others come from a population with lower expected counts. The model then has a Poisson mean but with a 'longer tail' variance function. The dispersion parameter, τ, is directly related to the amount of dispersion. This is the interpretation that we will use in the chapter and in *CrimeStat*.

Formally, we can write the negative binomial model as a Poisson-gamma mixture form:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \quad (16.24)$$

The Poisson mean λ_i is organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i) \quad (16.25)$$

where exp() is an exponential function, $\boldsymbol{\beta}$ is a vector of unknown coefficients for the *k* covariates plus an intercept, and ε_i is the model error independent of all covariates. The $\exp(\varepsilon_i)$ is assumed to follow the gamma distribution with a mean equal to 1 and a variance equal to $\tau = 1/\psi$ where ψ is a parameter that is greater than 0 (Lord, 2006; Cameron & Trivedi, 1998).

For a negative binomial generalized linear model, the deviance can be computed the following way:

$$D = \sum_{i=1}^{N} \left[y_i \ln\left(\frac{y_i}{\hat{\lambda}_i}\right) - (y_i + \hat{\psi}) \ln\left(\frac{y_i + \hat{\psi}}{\hat{\lambda}_i + \hat{\psi}}\right) \right] \quad (16.26)$$

For a well-fitted model the deviance should be approximately χ^2 distributed with $N - K - 1$ degrees of freedom (McCullagh and Nelder, 1987). If $D/(N - K - 1)$ is close to 1, we generally conclude that the model's fit is satisfactory.

Example 1 of Negative Binomial Regression

To illustrate, Table 16.3 presents the results of the negative binomial model for Houston burglaries. Even though the individual coefficients are similar, the likelihood statistics indicate

that the model fit the data better than the Poisson with linear correction for over-dispersion. The log-likelihood is higher, the AIC and BIC/SC statistics are lower as are the deviance and the Pearson Chi-square statistics.

On the other hand, the model error is higher than for the Poisson and Poisson NB1 models, both for the mean absolute deviation (MAD) and the mean squared predicted error (MSPE). Accuracy and precision need to be seen as two different dimensions for any method, including a regression model (Jessen, 1979, 13-16). Accuracy is 'hitting the target', in this case maximizing the likelihood function. Precision is the consistency in the estimates, again in this case the ability to replicate individual data values. A normal model will often produce lower overall error because it minimizes the sum of squared residual errors though it rarely will replicate the values of the records with high values and often does poorly at the low end.

For this reason, we say that the negative binomial is a more accurate model though not necessarily a more precise one. To improve the precision of the negative binomial, we would have to introduce additional variables to reduce the conditional variance further. Clearly, residential burglaries are associated with more variables than just the number of households and the median household income (e.g., ease of access into buildings, lack of surveillance on the street, having easy contact with individuals willing to distribute stolen goods).

Nevertheless, the negative binomial is a better model than the Poisson and certainly the normal, Ordinary Least Squares. It is theoretically sounder and does better with highly skewed (over-dispersed) data. In Appendix C, Lord and Park present a more formal presentation of the model.

Example 2 of Negative Binomial Regression with Highly Skewed Data

To illustrate further, the negative binomial is very useful when the dependent variable is extremely skewed. Figure 16.4 show the number of crimes committed (and charged for) by individual offenders in Manchester, England in 2006. The X-axis plots the number of crimes committed while the Y-axis plots the number of offenders. Of the 56,367 offenders, 40,755 committed one offence during that year, 7,500 committed two offences, and 3,283 committed three offences. At the high end, 26 individuals committed 30 or more offences in 2006 with one individual committing 79 offences. The distribution is very skewed.

A negative binomial regression model was set up to model the number of offences committed by these individuals as a function of conviction for previous offence (prior to 2006), age, and distance that the individual lived from the city center. Table 16.4 shows the results.

Table 16.3:
Predicting Burglaries in the City of Houston: 2006
MLE Negative Binomial Model
(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**			
N:	1,179			
Df:	1,175			
Type of regression model:	Poisson with Gamma dispersion			
Method of estimation:	Maximum likelihood			
Likelihood statistics				
Log-likelihood:	-4,430.8			
AIC:	8,869.6			
BIC/SC :	8,889.9			
Deviance:	1,390.1	p: 0.0001		
Pearson Chi-square:	1,112.7	p: n.s.		
Model error estimates				
Mean absolute deviation:	39.6			
1st (highest) quartile:	124.1			
2nd quartile:	19.4			
3rd quartile:	6.2			
4th (lowest) quartile:	8.9			
Mean squared predicted error:	62,031.2			
1st (highest) quartile:	242,037.1			
2nd quartile:	6,445.8			
3rd quartile:	118.3			
4th (lowest) quartile:	154.9			
Dispersion tests				
Adjusted deviance:	1.2	p: n.s		
Adjusted Pearson Chi-Square:	0.9	p: n.s.		
Dispersion multiplier:	1.5	p: n.s.	Inverse dispersion multiplier: 0.7	

Predictor	DF	Coefficient	Stand Error	Tolerance	VIF	Z-value	p
INTERCEPT	1	2.3210	0.083	-	-	27.94	0.001
HOUSEHOLDS	1	0.0012	0.00007	0.994	1.006	17.66	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.00001	0.000002	0.994	1.006	-5.13	0.001

Figure 16.4:

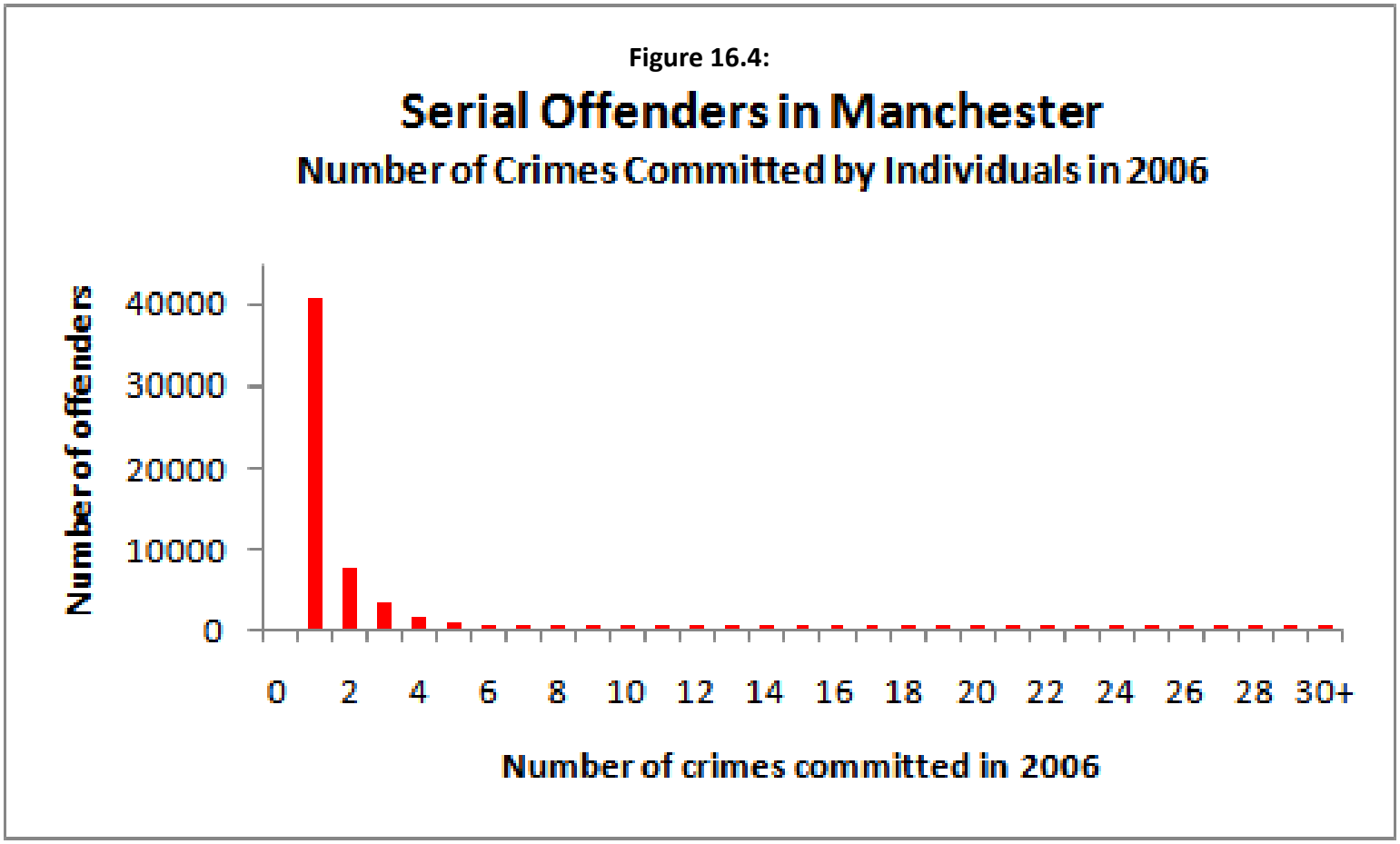

Serial Offenders in Manchester
Number of Crimes Committed by Individuals in 2006
Number of offenders
40000
30000
20000
10000
0
0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30+
Number of crimes committed in 2006

Table 16.4:

Number of Crimes Committed in Manchester in 2006
Negative Binomial Model

(N= 56,367 Offenders)

DepVar:	**NUMBER OF CRIMES COMMITTED IN 2006**				
N:	56,367				
Df:	56,362				
Type of regression model:	Poisson with Gamma dispersion				
Method of estimation:	Maximum likelihood				
Likelihood statistics					
Log-likelihood:	-89,103.7				
AIC:	178,217.4				
BIC/SC:	178,262.1				
Deviance:	36,616.6	p: n.s.			
Pearson Chi-square:	80,950.2	p: 0.0001			
Model error estimates					
Mean absolute deviation:	0.93				
1st (highest) quartile:	1.9				
2nd quartile:	0.7				
3rd quartile:	0.6				
4th (lowest) quartile:	0.6				
Mean squared predicted error:	3.90				
1st (highest) quartile:	13.8				
2nd quartile:	0.7				
3rd quartile:	0.6				
4th (lowest) quartile:	0.6				
Dispersion tests					
Adjusted deviance:	0.6	p: n.s.			
Adjusted Pearson Chi-Square:	1.4	p: n.s.			
Dispersion multiplier:	0.2	p : n.s.	Inverse dispersion multiplier:	6.2	

Predictor	DF	Coefficient	Stand Error	Tolerance	Z-value	p
INTERCEPT	1	0.509	0.012	-	41.90	0.001
DISTANCE FROM CITY CENTER	1	-0.022	0.003	0.999	-6.74	0.001
PRIOR OFFENCE	1	0.629	0.008	0.982	80.24	0.001
AGE OF OFFENDER	1	-0.012	0.0003	0.981	-35.09	0.001

The model was discussed in a recent article (Levine & Lee, 2013). The closer an offender lives to the city center, the greater than number of crimes committed. Also, younger offenders committed more offences than older offenders. However, the strongest variable is whether the individual had an earlier conviction for another crime. Offenders who have committed previous offences are more likely to commit more of them again. Crime is a very repetitive behavior!

The likelihood statistics indicates that the model was reasonably closely. The likelihood statistics were better than that of a normal OLS and a Poisson NB1 models (not shown). The model error was also slightly better for the negative binomial. For example, the MAD for this model was 0.93 compared to 0.95 for the normal and 0.93 for the Poisson NB1. The MSPE for this model was 3.90 compared to 3.93 for the normal and also 3.90 for the Poisson NB1. The negative binomial and Poisson models produce very similar results because, in both cases, the means are modeled as Poisson variables. The differences are in the dispersion statistics. For example, the standard error of the four parameters (intercept plus three independent variables was 0.012, 0.003, 0.008, and 0.0003 respectively for the negative binomial compared to 0.015, 0.004, 0.010, and 0.0004 for the Poisson NB1 model. In general, the negative binomial will fit the data better when the dependent variable is highly skewed and will usually produce lower model error.

Advantages of the Negative Binomial Model

The main advantage of the negative binomial model over the Poisson and Poisson with linear dispersion correction (NB 1) is that it incorporates the theory of Poisson but allows more flexibility in that multiple underlying distributions may be operating. Further, mathematically it separates out the assumptions of the mean (Poisson) from that of the dispersion (Gamma) whereas the Poisson with linear dispersion correction only adjust the dispersion after the fact (i.e., it determines that there is over- or under-dispersion and then adjusts it). This is neater from a mathematical perspective. Separating the mean from the dispersion can also allow alternative dispersion estimates to be modeled, such as the lognormal (Lord, 2006). This is very useful for modeling highly skewed data.

Disadvantages of the Negative Binomial Model

The biggest disadvantage is that the constancy of sums is not maintained. Whereas the Poisson model (both "pure" and with the linear dispersion correction) maintains the constancy of the sums (i.e., the sum of the predicted values equals the sum of the input values), the negative binomial does not. Usually, the degree of error in the sum of the predicted values is not far from the sum of the input values. But, occasionally substantial distortions are seen.

A second disadvantage is that the negative binomial model cannot handle under-dispersion. There are crime data sets that we have seen which show under-dispersion. For those, one needs another type of model. In Levine and Canter (2011), a Poisson with linear correction was used to adjust the standard errors (essentially, making them smaller). But, better methods need to be developed.

A final disadvantage of the negative binomial is related to the small sample size and low sample mean bias. It has been shown that the dispersion parameter of NB2 models can be significantly biased or misestimated when not enough data are available for estimating the model (Lord, 2006). For that, a Poisson-lognormal model is a better solution.

Alternative Poisson Regression Models

There are a number of variations of these involving different assumptions about the dispersion term, such as a lognormal function. There are also a number of different Poisson-type models including the zero-inflated Poisson (or ZIP; Hall, 2000), the Generalized Extreme Value family (Weibul, Gumbel and Fréchet), the lognormal function (see NIST 2004 for a list of common non-linear functions), and the Negative binomial-Lindley (Lord and Greedipally, 2011).

There are also alternative methods than maximum likelihood for estimating the likely value of a count given a set of independent predictors. In Chapter 17, we will examine several other approaches to estimating the Poisson model and will develop several alternative Poisson models.

Likelihood Ratios

One test that we have not implemented in the regression I module is the *likelihood ratio* because it is so simple. A likelihood ratio is the ratio of the log-likelihood of one model to that of another. For example, a Poisson-Gamma model run with three independent variables can be compared with a Poisson-Gamma model with two independent variables to see if the third independent variable significantly adds to the prediction.

The test is very simple. Let L_C be the log-likelihood of the comparison model and let L_B be the log-likelihood of the baseline model (the model to which the comparison model is being compared). Then,

$$LR = 2(L_C - L_B) \tag{16.27}$$

LR is distributed as a χ^2 statistic with K degrees of freedom where K is the difference in the number of parameters estimated between the two models including the intercepts. In the

example above, *K* is 1 since a model with three independent variables plus an intercept (d.f. = 4) is being compared with a model with two independent variables plus an intercept (d.f.=3).

Limitations of the Maximum Likelihood Approach

The functions considered up to this point are part of the single-parameter exponential family of functions where the function is smooth and convex. Because of this, maximum likelihood estimation (MLE) can be used. However, there are more complex functions that are not part of this family. Also, some functions come from multiple families and are, therefore, too complex to solve for a single maximum. They may have multiple 'peaks' for which there is not a single optimal solution. For these functions, a different approach has to be used.

Also, one of the criticisms leveled against maximum likelihood estimation (MLE) in general is that the approach *overfits* data. That is, it finds the values of the parameters that maximize the joint probability function. This is similar to the old approach of fitting a curve to data points with higher-order polynomials. While one can find some combination of higher-order terms to fit the data almost perfectly, such an equation has no theoretical basis nor cannot easily be explained. Further, such an equation does not usually do very well as a predictive tool when applied to a new data set.

MLE has been seen as analogous to this approach. By finding parameters that maximize the joint probability density distribution, the approach may be fitting the data too tightly. The original logic behind the AIC and BIC/SC criteria were to penalize models that included too many variables (Findley, 1993). However, these corrections only partially adjust the model. It is still possible to overfit a model with MLE. Radford (2006) has suggested that, in addition to a penalty for too many variables, that the gradient assent in a maximum likelihood algorithm be stopped before reaching the peak. This would require modifying the MLE algorithm substantially.

Further, Nannen (2003) has argued that overfitting creates a paradox because as a model fits the data better and better, it will do worse on other datasets to which it is applied for prediction purposes. In other words, it is better to have a simpler, but more robust, model than one that closely models one data set. Probably the biggest criticism against the MLE approach is that it underestimates the sampling errors by, again, overfitting the parameters (Husmeier & McGuire, 2002).

Instead, we will now examine a method that overcomes some of these difficulties, the Markov Chain Monte Carlo (MCMC) approach. Because the algorithm samples from a larger space rather than maximizes a function *per se*, it has the ability to find solutions to very complex problems for which the MLE approach is not appropriate. Chapter 17 presents this approach.

References

Bishop, Y. M. M., Feinberg, S. E. & Holland, P. W. (1975). *Discrete Multivariate Analysis: Theory and Practice*. MIT Press: Cambridge, MA.

Boswell, M. T. & Patil, G. P. (1970). "Chance mechanisms generating negative binomial distributions". In *Random Counts in Scientific Work*, Vol. 1, G. P. Patil, ed., Pennsylvania State University Press:University Park, PA, 3-22.

Cameron, A. Colin & Trivedi, Pravin K. (1998). *Regression Analysis of Count Data*. Cambridge University Press: Cambridge, U.K.

Findley, D. F. (1993). *The Overfitting Principles Supporting AIC*. Statistical Research Division Report Series, SRD Research Report no. CENSUS/SRD/ RR-93/04, U.S. Bureau of the Census: Washington, DC. http://www.census.gov/srd/papers/pdf/rr93-04.pdf.

Geedipally, S.R., D. Lord, S.S. Dhavala (2012) The Negative Binomial-Lindley Generalized Linear Model: characteristics and Application using Crash Data. Accident Analysis & Prevention, in press.

Greenwood, M. & Yule, G. U. (1920). "An inquiry into the nature of frequency distributions of multiple happenings, with particular reference to the occurrence of multiple attacks of disease or repeated accidents". *Journal of the Royal Statistical Society*, 83, 255-279.

Hall, D. B. (2000). "Zero-inflated Poisson and binomial regression with random effects: a case study". *Biometrics*, 56, 1030-1039.

Hilbe, J. M. (2008). *Negative Binomial Regression (with corrections)*. Cambridge University Press: Cambridge.

Husmeier, D. & McGuire, G. (2002). "Detecting recombination in DNA sequence alignments: A comparison between maximum likelihood and Markov Chain Monte Carlo". Biomathematics and Statistics Scotland, SCRI: Dundee.
http://www.bioss.ac.uk/~dirk/software/BARCEtdh/Manual/em/em.html

Jessen, R.J. (1979). *Statistical Survey Techniques*. John Wiley & Sons: New York.

Levine, N. & Lee, P. (2013). Crime travel of offenders by gender and age in Manchester, England. Leitner, M. (ed), *Crime Modeling and Mapping Using Geospatial Technologies,* Springer. 145-178.

References (continued)

Levine, N. & Canter, P. (2010). "Linking origins with destinations for DWI motor vehicle crashes: An application of Crime Travel Demand modeling". *Crime Mapping*, 3, 7-41.

Lord, D. (2006). "Modeling motor vehicle crashes using Poisson-gamma models: Examining the effects of low sample mean values and small sample size on the estimation of the fixed dispersion parameter". *Accident Analysis and Prevention*, 38, 751-766.

Lord, D. & Geedipally, S. R. (2011) The Negative Binomial-Lindley Distribution as a Tool for Analyzing Crash Data Characterized by a Large Amount of Zeros. Accident Analysis & Prevention, Vol. 43, No. 5, pp. 1738-1742.

Lord, D., Geedipally, S. R., & Guikema, S. (2010) Extension of the Application of Conway-Maxwell-Poisson Models: Analyzing Traffic Crash Data Exhibiting Under-Dispersion. Risk Analysis, Vol. 30, No. 8, pp. 1268-1276.

McCullagh, P. & Nelder, J. A. (1989). *Generalized Linear Models* (2nd edition). Chapman & Hall/CRC: Boca Raton, FL.

Mitra, S. & Washington, S. (2007). "On the nature of over-dispersion in motor vehicle crash prediction models", *Accident Analysis and Prevention*, 39, 459-468.

Nannen, V. (2003). *The Paradox of Overfitting*. Artificial Intelligence, Rijksuniversitat: Groningen, Netherlands. http://volker.nannen.com/pdf/the_paradox_of_overfitting.pdf. Accessed March 11, 2010.

NIST (2004). "Gallery of distributions". *Engineering Statistics Handbook*. National Institute of Standards and Technology: Washington, DC. http://www.itl.nist.gov/div898/handbook/eda/section3/eda366.htm.

Park, E.S., and Lord, D. (2007) Multivariate Poisson-Lognormal Models for Jointly Modeling Crash Frequency by Severity. In Transportation Research Record 2019: Journal of the Transportation Research Board, TRB, National Research Council, Washington, D.C., pp. 1-6.

Radford, N. (2006). "The problem of overfitting with maximum liklihood" . CSC 411: Machine Learning and Data Mining, University of Toronto: Toronto, CA. http://www.cs.utoronto.ca/~radford/csc411.F06/10-nn-early-nup.pdf Accessed March 11, 2010.

References (continued)

Springer (2010). "Polya distribution", *Encyclopedia of Mathematics*, Springerlink: London, http://eom.springer.de/p/p073540.htm.

Venables, W.N. & Ripley, B. D. (1997). *Modern Applied Statistics with S-Plus (second edition)*. Springer-Verlag: New York.

Wikipedia (2010). "Negative binomial distribution", *Wikipedia*, http://en.wikipedia.org/wiki/Negative_binomial_distribution Accessed February 24, 2010.

Chapter 17:

Estimating Complex Models with Markov Chain Monte Carlo Simulation[1]

Dominique Lord
Zachry Dept. of Civil Engineering
Texas A & M University
College Station, TX

Ned Levine
Ned Levine & Associates
Houston, TX

Byung-Jung Park
Korea Transport Institute
Goyang, South Korea

Srinivas Geedipally
Texas Transportation Institute
Arlington, TX

Haiyan Teng
Houston, TX

Li Sheng
Houston, TX

[1] This chapter was the result of the efforts of several people. Dr. Shaw-pin Miaou of College Station, TX designed the MCMC algorithm for the Poisson-Gamma model. Dr. Byung-Jung Park modified the algorithm to incorporate Poisson-Lognormal and the MCMC binomial model. Dr. Srinivas Geeidpally added the MCMC Normal model. Dr. Dominique Lord provided technical consulting on the dispersion parameters in these models. Dr. Ned Levine developed the block sampling scheme and provided overall project management. Ms. Haiyan Teng and Dr. Li Sheng programmed the routines and added numerous technical improvements to the algorithms.

Table of Contents

Table of Contents (continued)

Chapter 17:
Estimating Complex Models with Markov Chain Monte Carlo Simulation

In this chapter, we examine the Markov Chain Monte Carlo (MCMC) method for estimating complex models. We apply it to the family of Poisson models for modeling count data.

Markov Chain Monte Carlo (MCMC) Simulation of Regression Functions

To estimate a regression model from a complex function, we use a simulation approach called *Markov Chain Monte Carlo* (or MCMC). Chapter 12 of the *CrimeStat* manual discussed the Correlated Walk Analysis (CWA) routines. This was an example of a *random walk* whereby each step follows from the previous step. That is, a new position is defined only with respect to the previous position. This is an example of a Markov Chain.

In recent years, there have been numerous attempts to utilize this methodology for simulating regression and other models using a Bayesian approach (Lynch, 2007; Gelman, Carlin, Stern, & Rubin, 2004; Lee, 2004; Denison, Holmes, Mallick & Smith, 2002; Carlin & Louis, 2000; Leonard & Hsu, 1999).

Hill Climbing Analogy

To understand the MCMC approach, let us use a 'hill climbing' analogy. Imagine a mountain climber who wants to climb the highest mountain in a mountain range (for example, Mt. Everest in the Himalaya mountain range). However, suppose a cloud cover has descended on the range such that the tops of mountains cannot be seen; in fact, assume that only the bases of the mountains can be seen. Without a map, how does the climber find the mountain with the highest peak and then climb it? Realistically, of course, no climber is going to try to climb without a map and, certainly, without good visibility. But, for the sake of the exercise, think of how this could be done.

First, the climber could adopt a gradient approach with a systematic walking pattern. For example, he/she takes a step. If the step is higher than the current elevation (i.e., it is uphill), the climber then accepts the new position and moves to it. On the other hand, if the step is at the same or a lower elevation as the current elevation, the step is rejected. After each iteration (accepting or rejecting the new step), the procedure continues. Such a procedure is sometimes called a *greedy algorithm* because it optimizes the decision in incremental steps (local

optimization; Wikipedia, 2010a; Cormen, Leiserson, Rivest, & Stein; 2009; So, Ye, & Zhang, 2007; Dijkstra, 1959).

This strategy can be useful if there is a single mountain to climb (i.e., it is convex throughout or at least in the vicinity of the highest peak). Because generally moving uphill means moving towards the peak of the mountain, this approach will often lead the climber to get to the peak if the mountain is smooth. For a single mountain, a greedy algorithm such as our hill climbing example often works fine. The Maximum Likelihood Estimation (MLE) method is similar to this in that it requires a smooth convex function for which each step upward is assumed to be climbing the mountain. For functions that are smooth and convex, such as the single-parameter exponential family, this algorithm will work very well. The algorithm goes under different names but a common one is the *method of steepest ascent* (Goldfield, Quandt, & Trotter, 1966).

But, if there are multiple mountains (i.e., a range of mountains), how can we be sure that the peak that is climbed is really that of the highest mountain? In other words, again, without a map, for a range of mountains where there are multiple peaks but with only one being the highest, there is no guarantee that this greedy algorithm will find the single highest peak. Greedy algorithms work for simple problems but not necessarily for complex ones. Because they optimize the local decision process, they will not necessarily see the best approach for the whole problem - the global decision process (Goldfield, Quandt, & Trotter, 1966).

In other words, there are two problems that the climber faces. First, he/she does not know where to start. For this a 'map' would be ideal. Second, the search strategy of always choosing the step that goes up does not allow the climber to find alternative routes. Hills or mountains, as we all know, are rarely perfectly smooth; there are crevices and ridges and undulations in the gradient so that a climber will not always be going up in scaling a mountain. Instead, a climber needs to search a larger area in order to find a path that really does go up to the peak (sampling, if you wish).

This is the main reason why the MLE approach cannot estimate the parameters of a complex function since the approach works only for functions that are part of the single-parameter exponential family; they are closed-form functions for which there is a simple maxima that can be estimated. For these functions, which are very common, the MLE is a good approach. These functions are perfectly smooth which will allow a greedy algorithm to work. All of the generalized linear model functions – Ordinary Least Squares (OLS), Poisson, negative binomial, binomial probit, and others, can be solved with the MLE approach.

However, for a two or higher-parameter family, the approach will not work because there may be multiple peaks and a simple optimization approach will not necessarily discover the

highest likelihood. In fact, for a complex surface, MLE may get stuck on a local peak (a local optimum) and not have a way to backtrack in order to find another peak which is truly the highest.

For these, one needs a map for a good starting location and a sampling strategy that allows the exploration of a larger area than just that defined by a greedy algorithm. The 'map' comes from a Bayesian approach to the problem and the alternative search strategy comes from a sampling approach. This is essentially the logic behind the MCMC method.

Bayesian Probability

Let us start with the 'map' and briefly review the information that was discussed in Chapter 14. Bayes Theorem is a formulation that relates the conditional and marginal probability distributions of random variables. The *marginal probability* distribution is a probability independent of any other conditions. Hence, P(A) and P(B) is the marginal probability (or just plain probability) of A and B respectively.

The *conditional probability* is the probability of an event given that some other event has occurred. It is written in the form of P(A|B) (i.e., event A given that event B has occurred). In probability theory, it is defined as:

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} \tag{17.1}$$

or

$$P(B \mid A) = \frac{P(A \cap B)}{P(A)} \tag{17.2}$$

where the symbol $\cap$ represents the logical concept of 'and' (the Boolean intersection of A and B), which we expressed in words in Chapter 14. We will use the mathematical symbol now.

Bayes Theorem relates the two equivalents of the 'and' condition together.

$$P(B) \times P(A \mid B) = P(A) \times P(B \mid A) \tag{17.3}$$

$$P(A \mid B) = \frac{P(A) \times P(B \mid A)}{P(B)} \tag{17.4}$$

or

$$P(B \mid A) = \frac{P(B) \times P(A \mid B)}{P(A)} \tag{17.5}$$

Bayesian Inference

In the statistical interpretation of Bayes Theorem, the probabilities are estimates of a random variable. Let θ be a parameter of interest and let X be some data. Thus, Bayes Theorem can be expressed as:

$$P(\theta \mid X) = \frac{P(X \mid \theta) \times P(\theta)}{P(X)} \tag{17.6}$$

Interpreting this equation, $P(\theta \mid X)$ is the probability of θ given the data, X. $P(\theta)$ is the probability that θ has a certain distribution and is usually called the *prior probability*. $P(X \mid \theta)$ is the probability that the data would be obtained given that θ is true and is usually called the *likelihood function* (i.e., it is the likelihood that the data will be obtained given θ. Finally, $P(X)$ is the marginal probability of the data, the probability of obtaining the data under all possible scenarios of θ's.

The data are what was obtained from some data gathering exercise (either from experiments or observations). Since the prior probability of obtaining the data (the denominator of the above equation) is not known or cannot easily be evaluated, it is not easy to estimate it. Consequently, often the numerator only is used for estimating the posterior probability since

$$P(\theta \mid X) \propto P(X \mid \theta) \times P(\theta) \tag{17.7}$$

where $\propto$ means 'proportional to'. Because probabilities must sum to 1.0, the final result can be re-scaled so that the probabilities of all entities do sum to 1.0. The prior probability, $P(\theta)$, essentially is the 'map' in the hill climbing analogy discussed above! It points the way towards the correct solution.

The key point behind this logic is that an estimate of a parameter can be updated by additional information. The formula requires that a prior value for the estimate be given with new information being added that is *conditional* on the prior estimate, meaning that it factors in information from the prior. Bayesian approaches are increasingly being used to provide estimates for complex calculations that previously were intractable (Denison, Holmes, Mallilck, & Smith, 2002; Lee, 2004; Gelman, Carlin, Stern, & Rubin, 2004).

Markov Chain Sequences

Now, let us look at an alternative search strategy, the MCMC strategy. Unlike a conventional random number generator that generates independent samples from the distribution

of a random variable, the MCMC technique simulates a Markov chain with a limiting distribution equal to a specified target distribution. In other words, a Markov chain is a sequence of samples generated from a random variable in which the probability of occurrence of each sample depends only on the previous one. More specifically, a conventional random number generator draws a sample of size N and stops. It is non-iterative and there is no notion of the generator converging. We simply require N to be sufficiently large to produce reliable statistics.

An MCMC algorithm, on the other hand, is iterative with the generation of the next sample dependent on the value of the current sample. The algorithm requires us to sample until convergence has been obtained. The initial values of an MCMC algorithm are usually chosen arbitrarily and samples generated from one iteration to the next are correlated (autocorrelation). Consequently, the question of when we can safely accept the output from the algorithm as coming from the target distribution gets complicated and is an important topic in MCMC (convergence monitoring and diagnosis).

The MCMC algorithm involves five conceptual steps for estimating the parameter:

1. The user specifies a functional model and sets up the model parameters.

2. A likelihood function is set up and prior distributions for each parameter are assumed.

3. A joint posterior distribution for all unknown parameters is defined by multiplying the likelihood and the priors as in equation 17.7.

4. Repeated samples are drawn from this joint posterior distribution. However, it is difficult to directly sample from the joint distribution since the joint distribution is usually multi-dimensional. The parameters are, instead, sampled sequentially from their full conditional distributions, one at a time holding all existing parameters constant. This is the *Markov Chain* part of the MCMC algorithm. Typically, because it takes the chain a while to reach an *equilibrium* state, the early samples are thrown out ('burn-in') and the results are summarized based on the M-L samples where M is the total number of iterations and L are the discarded ('burn-in') samples (Miaou, 2006).

5. The estimates for all coefficients are based on the results of the M-L samples, for example the mean, the standard deviation, the median and various percentiles. Similarly, the overall model fit is based on the M-L samples.

MCMC Simulation

Each of these conceptual steps is complex, of course, and involves some detail. The following represents a brief discussion of the steps. In Appendix C, Dominique Lord and Byung-Jung Park presents a more formal discussion of the MCMC method in the context of the Poisson-Gamma-CAR model.

Step 1: Specifying a Model

The MCMC algorithm can be used for many different types of models. In this version of *CrimeStat*, we examine four types of MCMC model: the normal model, two non-spatial Poisson regression models (plus a Logit model that will be discussed in Chapter 18).

The normal model is an MCMC variant on the MLE Ordinary Least Squares. The two Poisson models (Poisson-Gamma and Poisson-Lognormal) are used to test over-dispersion while the NB1 model, discussed in Chapter 16, can be used to test under-dispersion. Figure 17.1 (which is a repeat of Figure 16.3) illustrates three types of dispersion. Note that over-dispersion is more extreme than under-dispersion though both are skewed. One has to use one of the Poisson family models with skewed count data to avoid introducing bias (see Chapter 15 for a discussion of bias from the use of an Ordinary Least Squares model).

Irrespective of the model used, in the Bayesian approach, prior probabilities have to be assigned to all unknown parameters, β, ψ, τ, ν. It is usually assumed that the β_k coefficients follow a *multivariate normal* distribution with $k+1$ dimensions:

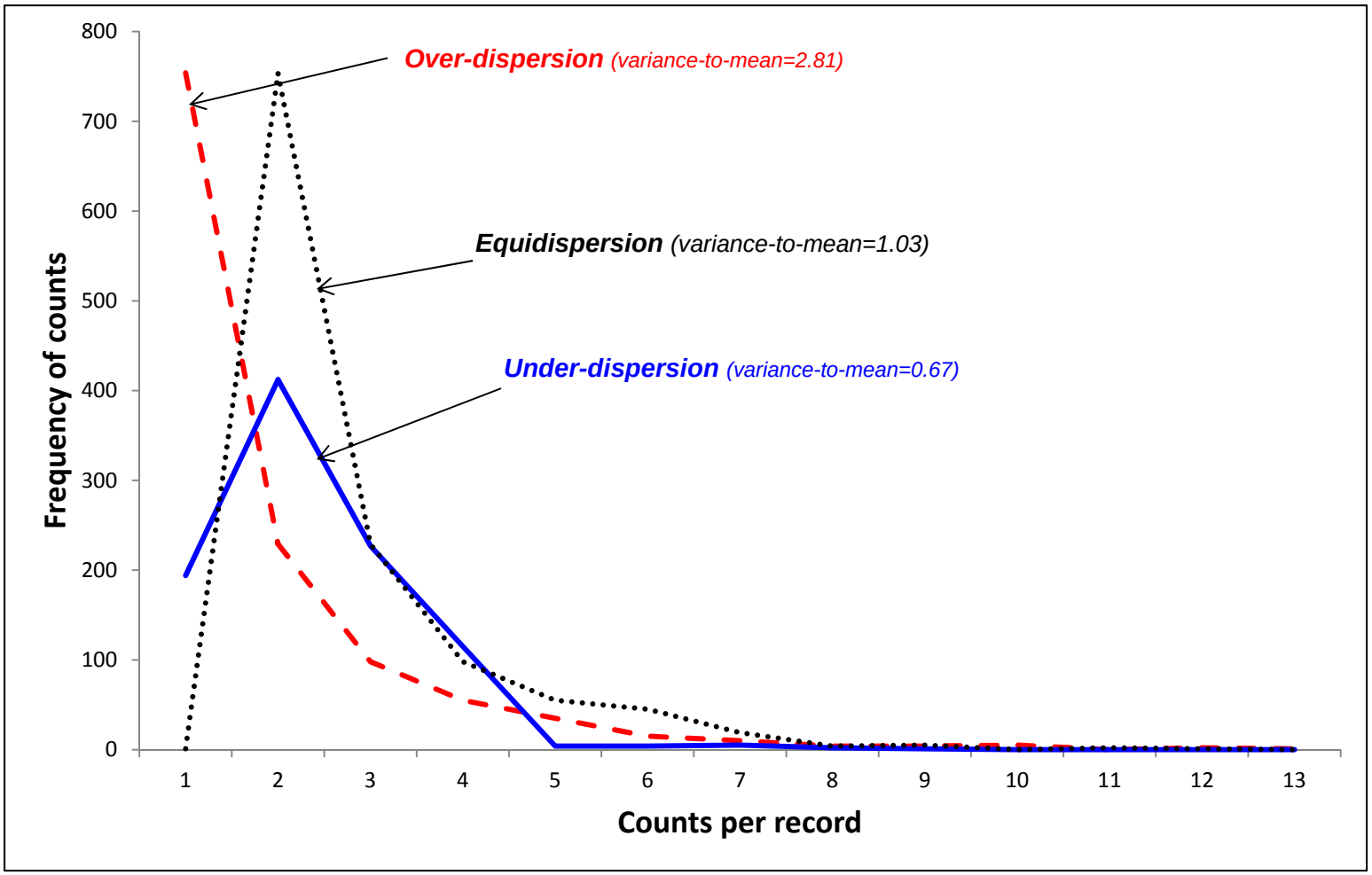
$$\boldsymbol{\beta} \sim MVN_{k+1}(\mathbf{b}_0, \mathbf{B}_0) \tag{17.8}$$

where MVN_{k+1} indicates a multivariate normal distribution with $k+1$ dimensions, and $\mathbf{b}_0$ and $\mathbf{B}_0$ are *hyperparameters* (parameters that define the multivariate normal distribution). For a non-informative prior specification, we usually assume $\mathbf{b}_0 = (0, \cdots, 0)^T$ and a large variance $\mathbf{B}_0 = 100\mathbf{I}_{k+1}$, where $\mathbf{I}_{k+1}$ denotes the $(k+1)$-dimensional identity matrix. Alternatively, independent normal priors can be placed on each of the regression parameters, e.g. $\beta_k \sim N(0, 100)$. If no prior information is known about $\boldsymbol{\beta}$, then sometimes a *flat* uniform prior is also used, $\beta_j \sim U(-\infty, \infty)$.

1. **Normal Model.** This is similar to the Ordinary Least Squares model discussed in Chapter 15 in that it assumes the dependent variable is normally-distributed. However, it is estimated by the MCMC algorithm rather than by MLE.

Figure 17.1:

Skewed Distributions and Type of Dispersion



The dependent variable *y* is a function of an expected mean for observation i and an error term, ε_i:

$$y_i = \lambda_i + \varepsilon_i \tag{17.9}$$

where λ_i is the predicted value of *y* and is a function of *k* independent variables (covariates),

$$\lambda_i = \mathrm{x}_i^T \beta \tag{17.10}$$

$\boldsymbol{\beta}$ is a vector of unknown coefficients for the *k* covariates plus an intercept. The error terms ε are independently and identically distributed as normal. Formally, it is defined as:

$$\varepsilon_i \sim Normal\ (0, \tau) \tag{17.11}$$

with τ being the variance The model error, ε_i, is independent of all covariates. The variance, τ, is assumed to follow a gamma distribution with a mean equal to 1 and a variance equal to $\tau = 1/\psi$ where ψ is a parameter that is greater than 0. The assumption on the uncorrelated error term ε_i is that it is constant for all observations. From equation 17.9, it follows that

$$y_i \sim Normal\ (\lambda_i, \tau) \tag{17.12}$$

2. **Poisson-Gamma Model**. This is similar to the negative binomial model discussed in Chapter 16 except that it is estimated by MCMC rather than by MLE. The Poisson-Gamma model is used when there is over-dispersion in the dependent variable. Formally, it is defined as:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \tag{17.13}$$

The Poisson mean λ_i is organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i) \tag{17.14}$$

where exp() is an exponential function, $\boldsymbol{\beta}$ is a vector of unknown coefficients for the *k* covariates plus an intercept, and ε_i is the model error independent of all covariates. The error, $\exp(\varepsilon_i)$, is assumed to follow a gamma distribution with a mean equal to 1 and a variance equal to $\tau = 1/\psi$ where ψ is a parameter that is greater than 0 (Lord, 2006; Cameron & Trivedi, 1998).

3. **Poisson-Lognormal Model.** The Poisson-Lognormal model is an alternative to the Poisson-Gamma. It is useful when there is over-dispersion and when there is a small sample size (less than 50) and the sample mean is low (<1.0; Park & Lord, 2007). It has been used in a number of transportation studies to model motor vehicle crashes (El-Basyouny & Sayed, 2009) and has been adapted to the Bayesian approach by Ma, Kockelman and Damien (2008). Like the Poisson-Gamma model, the Poisson-Lognormal model is defined as:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \tag{17.15}$$

The Poisson mean λ_i is organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i) \tag{17.16}$$

where exp() is an exponential function, $\boldsymbol{\beta}$ is a vector of unknown coefficients for the *k* covariates plus an intercept, and ε_i is the model error independent of all covariates. Unlike the Poisson-Gamma model, the error, $\exp(\varepsilon_i)$, is assumed to follow the ***lognormal distribution*** with a mean equal to 0 and a variance equal to $\sigma_\varepsilon^{-2} = \tau_\varepsilon \sim Gamma\ (a_\varepsilon, b_\varepsilon)$.

The reader is referred to Lord and Miranda-Moreno (2008) for additional details about the parameterization of the Poisson-lognormal model.

4. **Logit Model.** *CrimeStat* also includes an MCMC Logit model, but this will be discussed in Chapter 18.

How to Choose a Model

A key issue is how to choose among these alternatives. Overall, the two Poisson-based MCMC models give similar coefficients because the expected value is always estimated with a Poisson function. They differ primarily in the dispersion terms. The user is advised to first run an MLE Poisson model and check the diagnostics box. The diagnostics routine provides information on whether the dependent variable (the count) is significantly skewed while the dispersion parameter from the MLE Poisson model provides information on whether the conditional mean (the mean after controlling for the independent predictors) is still skewed. Further, for a spatial model (discussed in Chapter 19), the diagnostics routine will provide guidelines for the distance decay parameter (alpha).

While more research is clearly needed, a simple set of guidelines are as follows:

A. If the dependent variable is not significantly skewed (as indicated by the significance level of the "*g"* skewness test in the diagnostics routine), then run an OLS model.

B. If the "g" test of the dependent variable shows significant skewness and the ratio of the sample variance to the sample is greater than 2.0, then run an MLE or MCMC negative binomial (Poisson-Gamma) model since the negative binomial is a robust version of the Poisson. This is particularly true when the data set is larger than 50 cases and when the sample mean is 1.0 or greater. Note that the Poisson-lognormal model will provide similar results. However, the negative binomial is the usual model used with skewed data.

C. If the dispersion parameter in the negative binomial model is very close to 0 and is not significant, then the MLE Poisson model can be used. This is a case of equi-dispersion. However, in our experience very few data sets will show actual equi-dispersion. The vast majority are over-dispersed while some are under-dispersed.

D. If the "g" test of the dependent variable shows significant skewness and the ratio of the sample variance to the sample mean is greater than 2.0 but the sample size is less than 50 and the sample mean is less than 1.0, then use the MCMC Poisson-Lognormal model because it is a more robust model than the Poisson-Gamma with small samples and low sample means.

E. Finally, if the "g" test of the dependent variable shows significant skewness but the dispersion parameter in the negative binomial is less than 0, then use the NB1 model that was discussed in Chapter 16. This is a case of under-dispersion where the conditional variance is less than the conditional mean.

F. For all of these tests, the user should be aware of extreme outliers and multicollinearity among the independent variables (i.e., eliminate overlapping, multicollinear variables) as this can cause instability in the coefficients as well as cause models to shift from over-dispersion to under-dispersion, or vice versa.

Data with a Large Number of Zeros

The available Poisson models will handle the vast majority of data sets with count data. However, very occasionally, a data set with an extreme number of zeros will be found (e.g., 70%

or more of the cases have zero for the dependent variable). In cases where the dataset contains a large amount of zeros, traditional models, such as the Poisson-gamma or the Poisson-lognormal, can provide biased estimates or have difficulties converging. To overcome this problem, Poisson and negative binomial zero-inflated (ZI) models could be used (Lambert, 1992), as long as the model properly characterizes the data generating process (Lord et al., 2005). More recently, the Negative Binomial-Lindley (NB-L) distribution has been proposed to model datasets with a large number of zeros (Ghitany et al., 2008; Lord and Geedipally, 2011). The NB-L distribution is, as the name implies, a mixture of the NB and the Lindley distributions (Lindley, 1958; Ghitany et al., 2008). This two-parameter distribution has interesting and thorough theoretical properties in which the distribution is characterized by a single long-term mean that is never equal to zero and a single variance function, similar to the traditional NB distribution. This year, Geedipally et al. (2012) were able to fully develop the NB-L generalized linear model. The model has, in fact, been found to perform much better than the ZI models. The NB-L may be incorporated in a future version of *CrimeStat*.

Step 2: Setting up a Likelihood Function

For any of these types of non-spatial Poisson model, the log likelihood function is set up as a sum of individual logarithms of the model. In the case of the Poisson-Gamma model, the log likelihood function is:

$$L = \sum_{i=1}^{n} \left\{ \left(\sum_{j=0}^{y-1} \ln\left(j+\psi\right) \right) - \ln y_i ! - \left(y_i + \psi\right) \ln\left(1+\psi^{-1} \nu_i e^{\theta_i}\right) + y_i \ln \psi^{-1} + y_i \ln\left(\nu_i + \theta_i\right) \right\} \qquad (17.17)$$

with y_i being the observed (actual) value of the dependent variable, λ_i being the posterior *mean* of each site, $\theta_i = \ln \lambda_i$, ψ is the inverse dispersion parameter, and ν_i is an offset ('at risk') variable.

For the Poisson-Lognormal model, the log likelihood function is:

$$L = \ln\left(\prod_{i=1}^{n} \frac{e^{-(\nu_i \lambda_i)} (\nu_i \lambda_i)^{y_i}}{y_i !} \right) = \sum_{i=1}^{n} \left\{ \left[\nu_i e^{\theta_i} + y_i (\ln \nu_i + \theta_i) - \log \Gamma(y_i + 1) \right] \right\} \qquad (17.18)$$

with y_i being the observed (actual) value of the dependent variable, λ_i being the posterior *mean* of each site, $\theta_i = \ln \lambda_i$, and ν_i is an offset ('at risk') variable.

Step 3: Defining a Joint Posterior Distribution

In the case of the Poisson-Gamma model, the posterior probability, $p(\lambda, \beta, \psi \mid y, a_\omega, b_\omega)$, of the joint posterior distribution is defined as:

$$\pi(\lambda, \beta, \psi | y, a_\omega, b_\omega) \propto f(y|\nu\lambda) \cdot \pi(\lambda|\boldsymbol{\beta}, \psi) \cdot \pi(\beta_1) \cdots \pi(\beta_J) \cdot \pi(\psi|a_\omega, b_\omega) \qquad (17.19)$$

where y_i is the observed value of the dependent variable, **β** are the coefficients of each independent variable, ψ is the inverse dispersion parameter, while α_ω and b_ω are hyperparameters estimated internally in the routine. The equation is not in standard form (Park, 2009). Note that this is a general formulation. The parameters of interest are $(\lambda_1, \cdots \lambda_n), (\beta_1, \cdots \beta_J)$, and ψ.

For the Poisson-Lognormal, the posterior probability, $p(\lambda, \beta, \tau_\varepsilon \mid y, a_\varepsilon, b_\varepsilon)$, of the joint posterior distribution is defined as:

$$\pi(\lambda, \beta, \tau_\varepsilon | y, a_\varepsilon, b_\varepsilon) \propto f(y|\nu\lambda) \cdot \pi(\lambda|\boldsymbol{\beta}, \tau_\varepsilon) \cdot \pi(\beta_1) \cdots \pi(\beta_J) \cdot \pi(\tau_\varepsilon|a_\varepsilon, b_\varepsilon) \qquad (17.20)$$

where y_i is the observed value of the dependent variable, $\boldsymbol{\beta}$ are the coefficients of the independent variable, λ is the Poisson mean, τ_ε is the inverse of the variance and is Gamma distributed, and a and b are hyperparameters that are estimated internally in the routine.

In all the cases, since it is difficult to draw samples of the parameters from the joint posterior distribution, we usually draw samples for each parameter from its full conditional distribution sequentially. This is an iterative process (the Markov Chain part of the algorithm).

Prior distributions for these parameters have to be assigned. In the *CrimeStat* implementation, there is a parameter dialogue box that allows estimates for each of the parameters (including the intercept). On the other hand, if the user does not know which values to assign as prior probabilities, very vague values are used as default conditions to simulate what is known as *non-informative* priors (essentially, vague information). Sometimes these are known as *flat priors* if they assume all values are likely. In *CrimeStat*, we assign a default value for the expected coefficients of 0 and a very large variance. As mentioned, the user can substitute more precise values for the expected value of the coefficients or the variance (based on previous research, for example). Generally, having more precise prior values for the parameters will lead to quicker convergence and a more accurate estimate.

Step 4: Drawing Samples from the Full Conditional Distribution

Since the full conditional distribution itself is sometimes complicated (and becomes more so when the spatial components are added), the parameters are estimated by sampling from a distribution that represents the *target* distribution, either the target distribution itself if the function is standardized or a *proposal* distribution. While there are several approaches to sampling from a joint posterior distribution, the particular sampling algorithm used in *CrimeStat* is a *Metropolis-Hastings* (or MH) algorithm (Gelman, Carlin, Stern & Rubin, 2004; Denison, Holmes, Mallick, & Smith, 2002) with slice sampling of individual parameters (Radford, 2003).[2]

The MH algorithm is a general procedure for estimating the value of parameters of a complex function (Hastings, 1970; Metropolis, Rosenbluth, Rosenbluth, Teller & Teller, 1953). It was developed in the U. S. Hydrogen Bomb project by Rosenbluth and his colleagues and improved by Hastings.[3] Hence, it is known as the Metropolis-Hastings algorithm. With this algorithm, we do not need to sample directly from the target distribution but from an approximation called a *proposal* distribution (Lynch, 2007). The basic algorithm consists of six steps (Train, 2009; Lynch, 2007; Denison, Holmes, Mallick, & Smith, 2002).

1. Define the functional form of the target distribution and establish starting values for each parameter that is to be estimated, θ_0. For the first iteration, the existing value of the parameter, θ_E, will equal θ_0. Set t=1.

2. Draw a candidate parameter from a proposal density, θ_C.

3. Compute the posterior probability of the candidate parameter and divide it by the posterior probability of the existing parameter. Call this R.

4. If R is greater than 1, then accept the proposal density, θ_C.

---

[2] The Gibbs sampler utilizes the conditional probabilities of all parameters, which have to be specified. For a model such as the Poisson-Gamma, the Gibbs sampler could have been used. However, for a more complex model such as the Poisson-Gamma-CAR/SAR, the conditional probabilities are not easily defined. Consequently, we decided to utilize the MH algorithm in the routine. More information on the Gibbs sampler can be found in Lynch (2008); Gelman, Carlin, Stern & Rubin (2004); and Denison, Holmes, Mallick, & Smith (2002). Slice sampling is a way of drawing random samples from a distribution by sampling under the density distribution (Radford, 2003).

[3] It's called Metropolis-Hasting because Nicolas Metropolis was the first name listed on the paper. However, the math was developed mostly by Marshall Rosenbluth with the idea proposed by Edward Teller and the programming done by Arianna Rosenbluth (Wikipedia, 2012).

5. If R is not greater than 1, compare it to a random number drawn from a uniform distribution that varies from 0 to 1, u. If R is greater than u, accept the candidate parameter, θ_C. If R is not greater than u, keep the existing parameter θ_E.

6. Return to step 2 and keep drawing samples until sufficient draws are obtained.

Let us discuss these steps briefly. In the first step, an initial value of the parameter is taken. It is assumed that the functional form of the target population is known and has been defined (e.g., the target is a Poisson-Gamma function, a Poisson-Gamma-CAR, a Poisson-Lognormal-SAR function, a Binomial logit-CAR, etc.). The initial value should be consistent with this function. As mentioned above, a *non-informative* prior value can be selected.

Second, for each parameter in turn, a value is selected from a proposal density distribution. It is considered a 'candidate' since it is not automatically accepted as a draw from the target distribution. The proposal density can take any form that is easy to sample from, such as a normal distribution or a uniform distribution though usually the normal is used. Also, usually the distribution is symmetric though the algorithm can work for non-symmetric proposal distributions, too (Lynch, 2007, 109-112). In the *CrimeStat* implementation, we use a normal distribution. The proposal distribution does not have to be centered over the previous value of the parameter.

Third, the ratio of the posterior probability of the candidate parameter to the posterior probability of the existing parameter is calculated. This is called the *Acceptance* probability and is defined as:

$$\text{Acceptance probability} = R = \frac{f(\theta_C) * g(\theta_E)}{f(\theta_E) * g(\theta_C)} \tag{17.21}$$

The acceptance probability is made up of the product of two ratios. The function f is the target distribution and the function g is the proposal distribution. The first ratio, $f(\theta_C) * f(\theta_E)$, is the ratio of the densities of the target function using the candidate parameter in the numerator relative to the existing parameter in the denominator. That is, with the target function (the function for which we are trying to estimate the parameter values), we calculate the density using the candidate value and then divide this by the density using the existing value. Lynch (2007) calls it the *importance ratio* since the ratio will be greater than 1 if the candidate value yields a higher density (and, consequently, higher probability) than the existing one.

The second ratio, $g(\theta_E) * g(\theta_C)$, is the ratio of the proposal density using the existing value to the proposal density with the candidate value. This latter ratio adjusts for the fact that some candidate values may be selected more often than others (especially with asymmetrical

proposal functions). Note that the first ratio involves the target function densities whereas the second ratio involves the proposal function densities. If the proposal density is symmetric, then the second ratio will only have a very small effect.

Fourth, if R is greater than 1, meaning that the proposal density is greater than the original density, the candidate is accepted. However, if R is not greater than 1, this does not mean that the candidate is rejected but is instead compared to a random draw (otherwise we would have a 'greedy algorithm' that would only find local maxima).

Fifth, a random number, *u*, that varies from 0 to 1 is drawn from a uniform distribution and compared to R. If R is greater than *u*, then the value of the candidate parameter is accepted and becomes the new 'existing' parameter. Otherwise, if R is not greater than *u*, the existing parameter remains. Finally, in the sixth step, we repeat this algorithm and keep drawing samples until the desired sample size is reached.

Now what does this procedure do? Essentially, it draws values from the proposal distribution that increase the probability obtained from the target distribution. That is, generally only candidate values that increase the importance ratio will be accepted. But, this will not happen automatically (as, for example, in a greedy algorithm) since the ratio has to be compared to a random number, *u*, from 0 to 1. In the early steps of the algorithm, the random number may be higher than the existing R since it varies from 0 to 1. Thus, the candidate value is initially rejected more because it does not contribute to a high R ratio.

But, slowly, the acceptance probability will start to be accepted more often than the random draw since the candidate value will slowly approximate the true value of the parameter as it maximizes the target function's probability. Using the hill climbing analogy, the climber will wander around initially going in different directions but will slowly start to climb the hill and, most likely, the hill that is highest in the nearby vicinity. Each step that goes up will be accepted. But, each step that goes down will not necessarily be rejected since it is compared with a random 'step'. Thus, the climber explores other directions than just 'up'. But, over time, the climber will slowly move upward and, probably, more likely climb the highest hill nearby.

It is still possible for this algorithm to find a local 'peak' rather than the highest 'peak' since it explores in the vicinity of the starting location. To truly climb the highest peak, the algorithm needs a good starting value. Where does this 'good' starting value come from? Earlier research can be one basis for choosing a likely starting point. The more a researcher knows about a phenomenon, the better the researcher can utilize that information to ensure that the algorithm starts at a likely place. Without previous research to provide that value, however, Lynch (2007) proposes using the MLE approach to calculate parameters that are used as the initial values. That is, for a common distribution, such as the negative binomial, we use the

MLE negative binomial to estimate the values of the coefficients and intercept and then plug these into the MCMC routine as the initial values for that algorithm. *CrimeStat* allows the defining of initial values for the coefficients in the MCMC routine.

Step 5: Summarizing the Results from the Sample

Finally, after a sufficient number of samples have been drawn, the results can be summarized by analyzing the sample. That is, if a sample is drawn from a target population (using the MH approach or another one, such as the Gibbs method), then the distribution of the sample parameters is our best guess for the distribution of the parameters of the target function. The mean of each parameter would be the best guess for the coefficient value of the parameter in the target function. Similarly, the standard deviation of the sample values would be the best guess for the standard error of the parameter in the target distribution.

Credible intervals can be estimated by taking percentiles of the distribution. This is the Bayesian equivalent to a confidence interval in that it is estimated from a sample rather than from an asymptotic distribution. For example, the 95% credible interval can be calculated by taking the 2.5^{th} and 97.5^{th} percentiles of the sample while the 99% credible interval can be calculated by taking the 0.5^{th} and 99.5^{th} percentiles. There are also other statistics that can be calculated, for example the median (50^{th} percentile and the inter-quartile range (25^{th} and 75^{th} percentiles).

In other words, the entire MCMC sample is used to calculate statistics about the target distribution. Once the MCMC algorithm has reached 'equilibrium', meaning that it approximates the target distribution fairly closely, then a sample of values for each parameter from this algorithm yields an accurate representation of the target distribution.

MCMC Output

Let us discuss the statistics presented in the MCMC output.

Summary Statistics

First, there are the summary statistics represented by the log likelihood, the AIC, the BIC/SC, the Deviance, and Pearson Chi-square indices. Second, there are statistics for model error represented by the MAD and the MSPE; as with the MLE output, quartiles for these error statistics are presented. Third, there are the coefficients, the standard error, and a t-test based on the assumption that the distribution was normal and that the "t" is applicable (an assumption that is not necessarily correct). We present this because it allows a quick evaluation of the 'significance' of an independent variable.

Convergence Statistics

Fourth, in addition to these individual statistics, there are convergence statistics which indicate whether the algorithm converged (Spiegelhalter, Best, Carlin, & Van der Linde, 2002). It is essential for the user to evaluate whether the sequence converged; if it did not, then the coefficients and standard errors are not valid. These statistics are calculated by comparing chains of estimated values for parameters, either with themselves or with the complete series. When there is convergence, the estimates will be similar.

The first convergence statistic is the **Monte Carlo simulation error** (called *MC Error;* Ntzoufras, 2009, 30-40). Two estimates of the value of each parameter are calculated and their discrepancy is evaluated. The first estimate is the mean value of the parameter over all *M-L* iterations (total number of iterations minus the number of burn-in samples discarded). The second estimate is the mean value of the parameter after breaking the *M-L* iterations into *m* chains where *m* is the integer value of the square root of *M-L*.

Let:

$$Mean\theta_K = (\sum_i \theta_i) / K \tag{17.22}$$

$$Mean\theta_M = (\sum_m \theta_m) / m \tag{17.23}$$

and

$$MCError = \frac{\sqrt{Mean\theta_K - Mean\theta_M}}{m(m-1)} \tag{17.24}$$

Generally, the MC error is related to the standard deviation of the parameters. If the ratio is less than 0.05, then the sequence is considered to have converged after the 'burn in' samples have been discarded (Ntzourfras, 2009). As can be seen, the ratios are very low in Table 17.1.

The second convergence statistic is the **Gelman-Rubin convergence diagnostic** (*G-R*, sometimes called the *scale reduction factor*; Gelman, Carlin, Stern & Rubin, 2004; Gelman, 1996; Gelman & Rubin, 1992). Gelman and Rubin called it the *R* statistic, but we will call it the *G-R* statistic. The concept is, again, to break the larger chain into multiple smaller chains and calculate whether the variation within the chains for a parameter approximately equals the total variation across the chains (Carlin & Louis, 2008; Lynch, 2007). That is, when *m* chains are run, each of length *n*, the mean of a parameter θ_m can be calculated for each chain as well as the overall mean of all chains θ_G, the within-chain variance, and the between-chain variance. The G-R statistic is the square root of the total variance divided by the within-chain variance:

$$G-R=\sqrt{\left(\frac{m+1}{m}\right)*\left(\frac{n-1}{n}+\frac{B}{W}\right)-\left(\frac{n-1}{mn}\right)} \quad (17.25)$$

where B is the variance between the means from the m parallel chains, W is the average of the m within-chain variances, and n is the length of each chain (Lynch, 2007; Carlin & Louis, 2000).

The G-R statistic should generally be low for each parameter. If the G-R statistic is under approximately 1.2, then the posterior distribution is commonly considered to have converged (Mitra and Washington, 2007).

Example of Estimating Houston Burglaries with the MCMC Poisson-Gamma

Before we discuss some of the subtleties of the method, let us illustrate this with the Houston burglary example that we have been using in the previous two chapters (Table 17.1). The data came from the Houston Police Department. There were 26,480 burglaries that occurred in 2006 which were allocated to 1,179 Traffic Analysis Zones (TAZ) within the City of Houston. The independent variables were the number of households in 2006 (estimated by the Houston-Galveston Area Council, the metropolitan planning organization) and the median household income for 2000 (from the 2000 U.S. Census).

The MCMC algorithm for the Poisson-Gamma (negative binomial) model was run on the Houston burglary dataset. The total number of iterations was 25,000 with the initial 5,000 being discarded (the 'burn in' period). Thus, the results are based on the final 20,000 samples.

Comparison of MCMC Poisson-Gamma with MLE Poisson-Gamma

By comparing the results of the MCMC Poisson-Gamma estimate on the Houston burglary data set with that from the MLE Poisson-Gamma model from the previous chapter (Table 15.3), we can show that the MCMC method produces very similar results to the MLE when the estimated functions are identical. This is expected since the hyper-priors MCMC are very vague or have large variance. In Table 17.1, the two convergence statistics are very low for all three parameters as well as for the error term. In other words, the algorithm appears to have converged properly and the results are based on a good equilibrium chain.

Second, looking at the likelihood statistics, we see that they are very similar to that of the MLE negative binomial model. The log likelihood value is identical for the two models -4430.8. The AIC and BIC/SC statistics are also almost identical (8869.6 and 8869.8 compared to 8869.6 and 8889.9). The deviance statistic is very similar for the two models - 1,387.5 compared to 1,390.1, as is the Pearson Chi-square statistic – 1,106.4 compared to 1,112.7.

Table 17.1:

Predicting Burglaries in the City of Houston: 2006

MCMC Poisson-Gamma Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**		
N:	1,179		
Df:	1,175		
Type of regression model:	Poisson with Gamma dispersion		
Method of estimation:	MCMC		
Number of iterations:	25,000	Burn in: 5,000	
Likelihood statistics			
Log Likelihood:	-4,430.8		
AIC:	8,869.6		
BIC/SC:	8,889.9		
Deviance:	1,387.5	$p \leq 0.0001$	
Pearson Chi-Square:	1,106.4	$p \leq 0.0001$	
Model error estimates			
Mean absolute deviation:	40.0		
1st (highest) quartile:	124.9		
2nd quartile:	19.5		
3rd quartile:	6.2		
4th (lowest) quartile:	9.0		
Mean squared predicted error:	63,007.2		
1st (highest) quartile:	245,857.0		
2nd quartile:	6,527.5		
3rd quartile:	119.4		
4th (lowest) quartile:	156.2		
Dispersion tests			
Adjusted deviance:	1.2	$p \leq 0.0001$	
Adjusted Pearson Chi-Square:	0.9	$p \leq 0.0001$	
Dispersion multiplier:	1.5	$p \leq 0.0001$	Inverse dispersion multiplier: 0.7

Predictor	Mean	Std	t-value[p]	MC error	MC error/ std	G-R stat
INTERCEPT	2.3204	0.086	26.88***	0.002	0.019	1.002
HOUSEHOLDS	0.0012	0.00007	17.57***	0.0000009	0.013	1.001
MEDIAN HOUSEHOLD INCOME	-0.00001	0.00002	-4.92***	0.00000003	0.019	1.002

*** $p \leq .001$

Third, in terms of the model error statistics, the MAD and MSPE are also very similar (40.0 and 63,007.2 compared to 39.6 and 62,031.2; while the difference in the MSPE is 976.0, it is less than 2% of the MSPE for the MLE.[4] Fourth, the over-dispersion tests reveal identical values - adjusted deviance (1.2 for both), adjusted Pearson Chi-square (0.9 for both), and the Dispersion multiplier (both 1.5).

Fifth, the coefficients are identical with the MLE up through third decimal place. For example, for the intercept the MCMC gives 2.3204 compared to 2.3210; that of the two independent variables are identical within the precision of the table. This is not surprising since when we use non-informative priors, it is expected that the posterior estimates will be very close to those estimated by the MLE.

Sixth, the standard errors are identical for all three coefficients. In the MCMC, the standard errors are calculated by taking the standard deviation of the sample. In general, the MCMC will produce similar or slightly larger standard errors. The theoretical distribution assumes that the errors are normally distributed. This may or may not be true depending on the data set. Thus, the MCMC standard errors are non-parametric.

Seventh, a t-test (or more precisely a 'pseudo' t-test) is calculated by dividing the coefficient by the standard error. If the standard errors are normally distributed (or approximately normally distributed), then such a test is valid. On the other hand, if the standard errors are skewed, then the approximate t-test is not accurate. *CrimeStat* outputs additional statistics that list the percentiles of the distributions. These are more accurate indicators of the true confidence intervals and are known as *credible intervals*. We will illustrate these shortly with another example. In short, the pseudo t-test is an approximation to true statistical significance and should be seen as a guide, rather than a definitive answer.

Example of Estimating Houston Burglaries with the MCMC Normal

As an example of the MCMC Normal model, we ran the model on the Houston burglary data set. Keep in mind that this is a skewed data set and that the Normal model is not really appropriate. Table 17.2 presents the results. As a comparison, we repeat the MLE Normal/OLS model from Chapter 15 (Table 15.1).

[4] Frequently, the model error is greater for an MCMC model than an MLE model. Whether this represents true model error or overfitting by the MLE algorithm is not fully understood at this point.

Table 17.2:

Predicting Burglaries in the City of Houston: 2006

MCMC Normal Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1,179
Df:	1,175
Type of regression model:	Poisson with Lognormal dispersion
Method of estimation:	MCMC
Number of iterations:	25,000 Burn in: 5,000

Likelihood statistics

Log Likelihood:	-5342.6
AIC:	10,693.2
BIC/SC:	10,713.6
R^2:	0.48

Model error estimates

Mean absolute deviation:	13.5
1st (highest) quartile:	26.5
2nd quartile:	10.6
3rd quartile:	8.2
4th (lowest) quartile:	8.6
Mean squared predicted error:	505.1
1st (highest) quartile:	1,501.7
2nd quartile:	272.3
3rd quartile:	130.5
4th (lowest) quartile:	120.0

Predictor	Mean	Std	t-value[p]	MC error	MC error/ std	G-R stat
INTERCEPT	12.7804	1.235	10.35***	0.020	0.016	1.001
HOUSEHOLDS	0.0255	0.001	32.62***	0.000009	0.011	1.0005
MEDIAN HOUSEHOLD INCOME	-0.0002	0.00003	-7.00***	0.0000004	0.015	1.0004

** p≤.01

*** p≤.001

Table 15.3 (REPEAT):

Predicting Burglaries in the City of Houston: 2006

Ordinary Least Squares: Reduced Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1,179
Df:	1,175
Type of regression model:	Ordinary Least Squares
F-test of model:	536.0 $p \leq .0001$
R^2:	0.48
Adjusted R^2:	0.48
Mean absolute deviation:	13.5
1st (highest) quartile:	26.5
2nd quartile:	10.6
3rd quartile:	8.3
4th (lowest) quartile:	8.8
Mean squared predictive error:	505.1
1st (highest) quartile:	1498.8
2nd quartile:	269.5
3rd quartile:	135.1
4th (lowest) quartile:	120.2

Predictor	DF	Coefficient	Stand Error	Tolerance	VIF	t-value	p
INTERCEPT	1	12.8099	1.240	-	-	10.33	0.001
HOUSEHOLDS	1	0.0255	0.0008	0.994	1.006	33.44	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.0002	0.00003	0.994	1.006	-7.03	0.001

Comparison of MCMC Normal with MLE Normal

The MCMC Normal and the MLE Normal produce similar estimates. The log-likelihood statistics are unique to the MCMC model, but the R-squares are identical and the Mean Absolute Deviation and the Mean Squared Predictive Error values are very close to each other in both models. This means that the MCMC Normal converged on the function in a similar manner to the MLE normal. Also, the coefficients estimates for the MCMC Normal are quite close to those produced by the MLE.

Thus, it appears that the MCMC Normal can approximate the MLE Normal under some circumstances. Further, if the dependent variable is truly normally distributed, then the MCMC Normal will produce results that are almost identical.

Note that this is not always the case. When the dependent variable is highly skewed, we have frequently found that the MCMC Normal model will not produce identical results to that of the MLE even if a large number of iterations are run. We are not completely sure why this occurs, but the more skewed the distribution or the more complex the model, the less likely the MCMC Normal will yield the same solution as the MLE Normal. In short, the MCMC Normal is very sensitive to skewness in a data set and is most appropriate when the dependent variable is normally distributed.

Therefore, the user has to be careful in interpreting the MCMC Normal. Before running a spatial regression model using the MCMC Normal (see Chapter 19), users should confirm that the MCMC Normal can replicate an MLE Normal/OLS model. If it does not, they should run an alternative model such as the Poisson-Gamma.

Why Run an MCMC when MLE is So Easy to Estimate?

What we have seen is that the MCMC Poisson-Gamma (negative binomial) model and the MCMC Normal model produced results that were very similar to that of the MLE Poisson-Gamma and MLE Normal models respectively. In other words, simulating the distribution of the Poisson-Gamma function or the MCMC Normal function with the MCMC method has produced results that are completely consistent with a maximum likelihood estimate.

A key question, then, is why bother? The maximum likelihood algorithm works efficiently with functions from the single-parameter exponential family while the MCMC method takes time to calculate. Further, the larger the database, the greater the differential there will be in calculating time. For example in Chapter 16, Table 16.4 presented an MLE negative binomial model of the number of 2006 crimes committed by individual offenders in Manchester as a function of three independent variables – distance from the city center, prior conviction, and age of the offenders. With an Intel Duo core 2.44 GHz processor, the run took 6 seconds for the MLE while it took 86 minutes for the MCMC equivalent! Clearly, the MCMC algorithm is more calculation intensive than the MLE algorithm. If they produce essentially the same results, there is no obvious reason for choosing the slower method over the faster one.

The reason for preferring the MCMC method, however, has to do with the complexity of other models. The MLE approach works particularly well when all the individual functions in a mixed function model belong to the single-parameter exponential family of functions. For more complex functions, however, the method does not work very well. The likelihood functions

need to be worked out explicitly for the MLE approach to work. For example, if other functions for the dispersion were used, such as a Weibul or Gumbel or Cauchy or uniform distribution, the MLE approach would not easily be able to solve such equations since the mathematics are complex and there may not be a single optimal solution.

Further, if we start combining functions in different mixtures, such as Poisson mean, Gamma dispersion but Weibul shape function, the MLE is not easily adapted. An example is spatial regression where assumptions about the mean, the variance and spatial autocorrelation need to be specified exactly. This is a complex model and there is not a simple second derivative that can be calculated for such a function. The existing spatial models have tried to work around this by using a linear form but allowing a spatial autocorrelation term either as a predictive variable (the *spatial lag* model) or as part of the error term (the *spatial error* model; DeSmith, Goodchild, & Longley, 2007; Anselin, 2002). But, they all assume a normally-distributed dependent variable which is rarely found with crime data.

In short, the MCMC method has an advantage over MLE for complex functions. For simpler functions in which the functions are all part of the same exponential family and for which the mathematics has been worked out, MLE is clearly superior in terms of efficiency.

However, the more irregular and complex the function to be estimated, the more the simulation approach has an advantage over the MLE. For example, to estimate a Poisson-Gamma (negative binomial) function takes longer with the MCMC method than with the MLE method and there is no advantage for the MCMC over the MLE. On the other hand, the Poisson-Lognormal model (see below) or the Poisson-Gamma-CAR model (to be discussed in Chapter 19) cannot be estimated by MLE. An even more complex model is a spatial risk model where the 'at risk' variable is constrained to have a coefficient of 1.0 with spatial autocorrelation also being tested; this cannot be estimated with MLE.

Example of Estimating Houston Burglaries with the MCMC Poisson-Lognormal

For an example of a complex mixed function model, let us run the Houston burglary dataset with the Poisson-Lognormal. As mentioned above, the Poisson-Lognormal is an alternative model to the Poisson-Gamma. It is particularly useful when the sample mean is low and there are lots of zeros. The Poisson-lognormal is usually more stable than the Poisson-gamma for these kinds of data.

Table 17.3 shows the results. Compared to Table 17.1 for the Poisson-Gamma model, the log likelihood of the Poisson-Lognormal is more negative (weaker) than for the Poisson-Gamma while the AIC and BIC statistics are higher. In other words, the MCMC Poisson-Gamma fit the data slightly better than the MCMC Poisson-Lognormal though the differences are small.

Table 17.3:
Predicting Burglaries in the City of Houston: 2006
MCMC Poisson-Lognormal Model
(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**		
N:	1,179		
Df:	1,175		
Type of regression model:	Poisson with Lognormal dispersion		
Method of estimation:	MCMC		
Number of iterations:	25,000	Burn in: 5,000	
Likelihood statistics			
Log Likelihood:	-4,650.2		
AIC:	9,308.4		
BIC/SC:	9,328.7		
Deviance:	1,551.9	p≤ 0.0001	
Pearson Chi-Square:	4,685.6	p≤ 0.0001	
Model error estimates			
Mean absolute deviation:	37.5		
1st (highest) quartile:	122.5		
2nd quartile:	20.6		
3rd quartile:	3.3		
4th (lowest) quartile:	4.0		
Mean squared predicted error:	62,216.2		
1st (highest) quartile:	244,906.0		
2nd quartile:	4,489.4		
3rd quartile:	40.4		
4th (lowest) quartile:	63.4		
Dispersion tests			
Adjusted deviance:	1.3	p≤ 0.0001	
Adjusted Pearson Chi-Square:	4.0	p≤ 0.0001	
Dispersion multiplier:	2.0	p≤ 0.0001	Inverse dispersion multiplier: 0.5

Predictor	Mean	Std	t-value[p]	MC error	MC error/ std	G-R stat
INTERCEPT	1.3612	0.092	14.82***	0.002	0.022	1.002
HOUSEHOLDS	0.0013	0.00005	25.30***	0.0000007	0.014	1.000
MEDIAN HOUSEHOLD INCOME	-0.000005	0.00002	-2.92**	0.00000003	0.018	1.001

** p≤.01
*** p≤.001

However, the log-likelihood is the overall probability of the model, not particularly the best fit for the residual errors. Comparing Tables 17.1 and 17.2, we find that the MAD and the MSPE are smaller for the Poisson-Lognormal than for the Poisson-Gamma. The coefficients are very similar. The intercept is smaller in the Poisson-Lognormal while the coefficients for households and for median household income are virtually the same. In short, the Poisson-Lognormal will predict a slightly smaller expected count than the Poisson-Gamma due to the smaller intercept term, but the two sets of estimates are quite similar. In other words, with these data, the Poisson-Lognormal model produces a slightly lower probability but a better fit than the Poisson-Gamma. In this case, we would accept the Poisson-Gamma because the differences are not great. But, there are data sets where the Poisson-Lognormal is definitely better than the Poisson-Gamma (Lord & Miranda-Moreno, 2008).

Risk Analysis

One example of where the MCMC method is better than the MLE method is in ***risk analysis***. Sometimes a dependent variable is analyzed with respect to an exposure variable. For example, instead of modeling just burglaries, a user might want to model burglaries relative to the number of households. In our example in this chapter (Houston burglaries), we have included the number of households as a predictor variable but it is unstandardized, meaning that the estimated effect of households on burglaries cannot be easily compared to other studies that model burglaries relative to households.

For this, a different type of analysis has to be used. Frequently called a *risk analysis*, the dependent variable is related to an exposure measure. The formulation we use is that of Besag, Green, Higdon and Mengersen (1995). Like all the non-linear models that we have examined, the dependent variable, y_i, is modeled as a Poisson function of the mean, λ_i:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \tag{17.26}$$

In turn, the mean of the Poisson is modeled as:

$$\mu_i = \nu_i \lambda_i \tag{17.27}$$

where ν_i is an *exposure* measure and λ_i is the *rate* (or risk). The exposure variable is the baseline variable to which the number of events is related. For example, in motor vehicle crash analysis, the exposure variable is usually Vehicle Miles Traveled or Vehicle Kilometers Traveled (multiplied by a power of 10 to eliminate very small numbers, such as per 1000 or per 100 million). In epidemiology, the exposure variable is the population at risk, either the general population or the population of a specific age group perhaps broken down further into gender.

For crime analysis, the exposure variable might be the number of households for residential crimes or the number of businesses for commercial crimes. Choosing an appropriate exposure variable is not a trivial matter. In some cases, there are national standards for exposure (e.g., number of infants for analyzing child mortality; Vehicle Miles Traveled for analyzing motor vehicle crash rates). But, often there are not accepted exposure standards.

In some cases, the exposure variable may be non-linear in order to capture important missing variables. For instance, in highway safety, traffic flow (i.e., the number of vehicle traveling passing a given point in a unit of time) has been found to vary in a non-linear fashion. This characteristic can be explained by the fact vehicle occupancy (i.e., the number of vehicles per unit of length) and vehicle speed, which are directly linked to traffic flow, are variables that are not availabe or routinely collected. Hence, traffic flow tends to show non-linear relationships (see Lord, Manar, &Vizioli, 2005, for more details).

The rate is further structured in the Poisson-Gamma or Poisson-Lognormal models:

$$\mu_i = \nu_i \lambda_i = \nu_i \cdot \exp(\mathrm{x}_i^T \beta + \varepsilon_i) \tag{17.28}$$

where the symbols have the same definitions as in equation 17.18 with the error term, ε_i, being modeled either as a Gamma function (equation 17.9) or as a Lognormal function (equation 6.10).

With the exposure term, the full model is estimated as the same fashion,

$$y_i \sim Poisson(\nu_i \lambda_i) \tag{17.29}$$

$$\lambda_i \sim Gamma\ (\psi, \psi e^{-\mathbf{x}_i^T \boldsymbol{\beta} - \phi_i}) \tag{17.30}$$

$$\lambda_i \sim Lognormal\,[0, Gamma(a_\varepsilon, b_\varepsilon)] \tag{17.31}$$

Note that no coefficient for the exposure variable, ν_i, is estimated (i.e., it is 1.0). It is sometimes called an *offset* variable (or exposure offset). The model is then estimated either with an MLE or MCMC estimation algorithm.

An example is that of Levine (2011) who analyzed the number of motor vehicle crashes in which a male was the primary driver relative to the number of crashes in which a female was the primary driver for each major road segment in the Houston metropolitan area. In the risk model set up, the dependent variable was the number of crashes involving a male primary driver for each road segment while the exposure (offset) variable was the number of crashes involving a female primary driver. The independent variables in the equation were volume-to-capacity ratio

(an indicator of congestion on the road), the distance to downtown Houston, and several road categories (freeway, principal arterial, etc).

To illustrate this type of model, we ran a MCMC Poisson-Gamma model using the number of households as the exposure variable. There was, therefore, only one independent variable, median household income. Table 17.4 shows the results

.

Compared to the non-exposure burglary model (Table 17.1), the model does not fit the data as well. The log likelihood is lower while the AIC and BIC are higher. Further, the MAD and MSPE statistics for model error are much worse.

Further, the dispersion statistics indicate that there is more over-dispersion with the risk model than the simple Poisson-Gamma model. In other words, the exposure variable has not eliminated the dispersion as much as the random effects (non-exposure) model.

Looking at the coefficients, the offset variable (number of households) has a coefficient of 1.0 because it is defined as such. The coefficient for median household income is still negative, but is stronger than in Table 17.1. The effect of standardizing households as the baseline exposure variable has increased the importance of household income in predicting the number of burglaries, controlling for the number of households.

The second part of the table show percentiles for the coefficients, and is preferable for statistical testing than the asymptotic t-test. The reason is that the distribution of parameter values may not be normally distributed or may be very skewed, whereas the t- and other parametric significance tests assume that there is perfect normality. *CrimeStat* outputs a number of percentiles for distribution. We have shown only four of them, the 0.5^{th}, 2.5^{th}, 97.5^{th}, and 99.5^{th} percentiles. The 2.5^{th} and 97.5^{th} represent 95% credible intervals while the 0.5^{th} and 99.5^{th} represent 99% credible intervals.

The way to interpret the percentiles is to check whether a coefficient of 0 (the 'null hypothesis') or any other particular value is outside the 95% or 99% credible intervals. For example, with the intercept term, the 95% credible interval is defined by -2.4365 to -2.1292. For both the intercept and median household income, a coefficient of 0 is outside both the 95% and 99% credible intervals. In other words, both the intercept and median household income are *significantly* different than 0, though the use of the term 'significant' is different than with the usual asymptotic normality assumptions since it is based on the distribution of the parameter values from the MCMC simulation.

Table 17.4:

Predicting Burglaries in the City of Houston: 2006
MCMC Poisson-Gamma Model with Exposure Variable

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**		
N:	1,179		
Df:	1,176		
Type of regression model:	Poisson with Gamma dispersion		
Method of estimation:	MCMC		
Number of iterations:	25,000	Burn in:	5,000
Distance decay function:	Poisson-Gamma		
Likelihood statistics			
Log Likelihood:	-6,634.4		
AIC:	13,274.8		
BIC/SC:	13,290.0		
Deviance:	5,373.5	p≤ 0.0001	
Pearson Chi-square:	514.4	p≤ 0.0001	
Model error estimates			
Mean absolute deviation:	14,147.3		
Mean squared predicted error:	553,058,555.7		
Dispersion tests			
Adjusted deviance:	4.6	p≤ 0.0001	
Adjusted Pearson Chi-Square:	0.44	p≤ 0.0001	
Dispersion multiplier:	2.3	p≤ 0.0001	Inverse dispersion multiplier: 0.44

Predictor	Mean	Std	t-value[p]	MC error	MC error/ std	G-R stat
Exposure/offset variable:						
HOUSEHOLDS	1.0	-	-	-	-	-
Linear predictors:						
INTERCEPT	3.4624	0.0917	37.75***	0.002	0.020	1.002
MEDIAN HOUSEHOLD INCOME	-0.00009	0.000002	-4.57***	0.00000004	0.020	1.002

*** p≤.001

Percentiles	**0.5th**	**2.5th**	**97.5th**	**99.5th**
INTERCEPT	3.2242	3.2833	3.6389	3.6942
MEDIAN HOUSEHOLD INCOME	-0.00002	-0.00001	-0.00005	-0.00004

In other words, percentiles can be used as a non-parametric alternative to the t- or Z-test. Without making assumptions about the theoretical distribution of the parameter value (which the t- and Z-test do – they are assumed to be normal or near normal for "t"), significance can be assessed empirically.

In summary, in risk analysis, an exposure variable is defined and held constant in the model. Thus, the model is really a risk or rate model that relates the dependent variable to the baseline exposure. The independent variables are now predicting the rate, rather than the count by itself.

Issues in MCMC Modeling

We now turn to four issues in MCMC modeling. The first is the starting values of the MCMC algorithm. The second is the issue of *convergence* to an equilibrium state. The third issue is the statistical testing of parameters and the general problem of overfitting the data while the fourth issue is the performance of the MCMC algorithm with large datasets.

Starting Values of Each Parameter

The MCMC algorithm requires that initial values be provided for each parameter to be estimated. These are called *prior probabilities* even though they do not have to be standardized in terms of a number from 0 to 1. The *CrimeStat* routine allows the defining of initial starting values for each of the parameters and for the overall Φ coefficient in the various spatial regression models (see chapter 18). If the user does not define the initial starting values, then default values are used. Of necessity, these are vague. For the individual coefficients (and the intercept), the initial default values are 0. For the Φ coefficient, the initial default values are defined in terms of its hyperparameters, (Rho = 0.5; Tauphi = 1; alpha = -1). Essentially, these assume very little about the distribution and are, essentially, *non-informative priors*.

The problem with using vague starting values, however, is that the algorithm could get stuck on a local 'peak' and not actually find the highest probability. Even though the MCMC algorithm is not a greedy algorithm, it still explores a limited space. It will generally find the highest peak within its search radius. But, there is no guarantee that it will explore regions far away from its initial location. If the user has some basis for estimating a prior value, then this will usually be of benefit to the algorithm in that it can minimize the likelihood of finding local 'peaks' rather than the highest 'peak'.

Where do the prior values come from? They can come from other research, of course (see Miranda-Moreno et al., 2009). Alternatively, they can come from other methods that have attempted to analyze the same phenomena. Lynch (2007), for example proposes running an

MLE Poisson-Gamma (negative binomial) model and then using those estimates as the prior values for the MCMC Poisson-Gamma. Even if the user is going to run a spatial model (e.g., MCMC Poisson-Gamma-CAR/SAR), the estimates from an MLE model are probably good starting values.

Example of Defining Prior Values for Parameters

We can illustrate this with an example. A model was run on 325 Baltimore County traffic analysis zones (TAZ) predicting the number of crimes that occurred in each zone in 1996. There were four independent variables:

1. Population (1996)
2. Relative median household income index
3. Retail employment (1996)
4. Distance from the center of the metropolitan area (in the City of Baltimore)

The dataset was divided into two groups, group A with 163 TAZs and group B with 162 TAZs. The model was run as a spatial regression (Poisson-Gamma-CAR – see chapter 19) for each of the groups. Table 17.5 shows the results of the coefficients with the standard errors in brackets.

Column 1 shows the results of running the model on group A. Column 2 shows the results of running the model on group B while column 3 shows the results of running the model on group B but using the coefficient estimates from group A as prior values. With the exception of the relative income variable, the coefficients of column C generally fall between the results for group A and group B by themselves. Even the one exception – relative income, is very close to the 'non-informative' estimate for group B.

In other words, using prior values that are based on realistic estimates (in this case, the estimates from group A) have produced results that incorporate that information in estimating the information just from the data. Essentially, this is what equation 17.7, updating the probability estimate of the data given the likelihood based on the prior probability. In short, using prior estimates combines new information with the existing information to update the estimates. Aside from protecting against finding local optima in the MCMC algorithm, the prior information generally improves the knowledge base of the model.

Convergence

In theory, the MCMC algorithm should converge into a stable equilibrium state whereby the true probability distribution is being sampled. With the hill climbing analogy, the climber has found the highest mountain to be climbed and is simply sampling different locations on the

Table 17.5:
The Effects of Starting Values on Coefficient Estimates for Baltimore County Crimes:

Dependent Variable = Number of Crimes in 1996

	(1) **Group A** (N=163 TAZs)	(2) **Group B** (N=162 TAZs)	(3) **Group B** (N=162 TAZs)
Starting values:	**Default/ 'non-informative'**	**Default/ 'non-informative'**	**Group A estimates**
Independent variables			
INTERCEPT	4.3621 *(0.2674)*	4.7727 *(0.2434)*	4.7352 *(0.2489)*
POPULATION	0.00035 *(0.00004)*	0.00034 *(0.00004)*	0.00035 *(0.00004)*
RELATIVE INCOME	-0.0234 *(0.0047)*	-0.0226 *(0.0041)*	-0.0224 *(0.0043)*
RETAIL EMPLOYMENT	0.0021 *(0.0002)*	0.0017 *(0.0002)*	0.0017 *(0.0001)*
DISTANCE FROM CENTER	-0.0590 *(0.0160)*	-0.0898 *(0.0141)*	-0.0881 *(0.0142)*
AVERAGE PHI COEFFICIENT	0.0104 *(0.1117)*	-0.0020 *(0.0676)*	0.0077 *(0.0683)*

mountain to see which one will provide the best path up the mountain. The first iterations in a sequence are thrown away (the 'burn in') because the sequence is assumed to be looking for the true probability distribution. Put another way, the starting values of the MCMC sequence have a big effect on the early draws and it takes a while for the algorithm to move away from those initial values (remember, it is a random walk and the early steps are near the initial starting location).

A key question is how many samples to draw and a second, ancillary question is how many should be discarded as the 'burn in'? Unfortunately, there is not a simple answer to these questions. For some distributions, the algorithm quickly converges on the correct solution and a limited number of draws are needed to accurately estimate the parameters. In the Houston burglary example, the algorithm easily converged with 20,000 iterations after the first 5,000 had been discarded. We have been able to estimate the model accurately after only 4000 iterations with 1000 burn in samples being discarded. The dependent variable is well behaved because it is at the zonal level and the model is simple.

On the other hand, some models do not easily converge to an equilibrium stage. Models with individual level data are typically more volatile. Also, models with many independent variables are complex and do not easily converge. To illustrate, we estimate a model of the residence locations of drunk drivers (DWI) who were involved in crashes in Baltimore County between 1999 and 2001 (Levine & Canter, 2011). The drivers lived in 532 traffic analysis zones (TAZ) in both Baltimore County and the City of Baltimore. The dependent variable was the annual number of drivers involved in DWI crashes who lived in each TAZ and there were six independent variables:

1. Total population of the TAZ
2. The percent of the population who were non-Hispanic White
3. Whether the TAZ was in the designated rural part of Baltimore County (dummy variable: 1 – Yes; 0 – No)
4. The number of liquor stores in the TAZ
5. The number of bars in the TAZ
6. The area of the TAZ (a control variable).

Table 17.6 presents the results. The overall model fit was statistically significant and there was very little over-dispersion (as seen by the dispersion parameter). A "pure" Poisson model could have been used in this case. Of the parameters, the intercept and four of the six independent variables were statistically significant, based on the t-test. The results were consistent with expectations, namely zones (TAZs) with greater population, a greater percentage of non-Hispanic White persons, that were in the rural part of the county, that had more liquor stores, and that had more bars had a higher number of drunk drivers residing in those zones.

However, the convergence statistics were questionable. Two of the parameters had G-R values higher than the acceptable 1.2 level and five of the MC error/standard error values were higher than the acceptable 0.05 level. In other words, it appears that the model did not properly converge. Consequently, we ran the model again with 100,000 iterations and discarded the initial 10,000 'burn in' samples. Table 17.7 shows the results. Comparing tables 17.6 with 17.5, we can see that the overall likelihood statistics was approximately the same as were the dispersion

Table 17.6:

Number of Drivers Involved in DWI Crashes Living in Baltimore County: 1999-2001 MCMC Poisson-Gamma Model with 20,000 Iterations

(N= 532 Traffic Analysis Zones)

DepVar:	**Annual Number of Drivers in DWI Crashes Living in TAZ**		
N:	532		
Type of regression model:	Poisson with Gamma dispersion		
Method of estimation:	MCMC		
Total number of iterations:	25,000	Burn in:	5,000
Likelihood statistics			
Log Likelihood:	-278.7		
AIC:	573.4		
BIC/SC:	607.6		
Deviance:	316.6	p: 0.0001	
Pearson Chi-square:	475.6	p: 0.0001	
Model error estimates			
Mean absolute deviation:	0.32		
Mean squared predicted error:	0.25		
Dispersion tests			
Adjusted deviance:	0.60	p: 0.0001	
Adjusted Pearson Chi-Square:	0.91	p: 0.0001	
Dispersion multiplier:	0.15	p: 0.0001	Inverse dispersion multiplier: 6.77

Predictor	**Mean**	**Std**	**t-value[p]**	**MC error**	**MC error/ std**	**G-R stat**
INTERCEPT	-4.5954	0.476	-9.65***	0.0386	0.081	1.349
POPULATION	0.0004	0.00005	8.70***	0.000003	0.068	1.165
PERCENT WHITE	0.0237	0.005	4.81***	0.0004	0.079	1.283
RURAL	0.6721	0.329	2.04*	0.0184	0.056	1.042
LIQUOR STORES	0.2423	0.125	1.94n.s.	0.0059	0.047	1.028
BARS	0.1889	0.058	3.28**	0.0024	0.041	1.008
AREA	-0.0548	0.033	-1.68n.s.	0.0018	0.055	1.041

n.s. Not significant
** $p \leq .01$
*** $p \leq .001$

Table 17.7:

Number of Drivers Involved in DWI Crashes Living in Baltimore County: 1999-2001 MCMC Poisson-Gamma Model with 90,000 Iterations

(N= 532 Traffic Analysis Zones)

DepVar:	**Annual Number of Drivers in DWI Crashes Living in TAZ**		
N:	532		
Type of regression model:	Poisson with Gamma dispersion		
Method of estimation:	MCMC		
Total number of iterations:	100,000	Burn in:	10,000
Likelihood statistics			
Log Likelihood:	-278.6		
AIC:	573.2		
BIC/SC:	607.4		
Deviance:	317.9	p: 0.0001	
Pearson Chi-square:	479.5	p: 0.0001	
Model error estimates			
Mean absolute deviation:	0.32		
Mean squared predicted error:	0.25		
Dispersion tests			
Adjusted deviance:	0.61	p: n.s.	
Adjusted Pearson Chi-Square:	0.92	p: n.s.	
Dispersion multiplier:	0.14	p: n.s.	Inverse dispersion multiplier: 7.36

Predictor	**Mean**	**Std**	**t-value[p]**	**MC error**	**MC error/ std**	**G-R stat**
INTERCEPT	-4.6608	0.425	-10.96***	0.0222	0.052	1.085
POPULATION	0.0004	0.00005	8.78***	0.000002	0.041	1.041
PERCENT WHITE	0.0243	0.004	5.77***	0.0002	0.050	1.081
RURAL	0.6378	0.324	1.97*	0.0092	0.028	1.005
LIQUOR STORES	0.2431	0.123	1.98*	0.0033	0.027	1.002
BARS	0.1859	0.055	3.36***	0.0011	0.020	1.004
AREA	-0.0515	0.032	-1.63n.s.	0.0009	0.029	1.008

n.s. Not significant
* $p \leq .05$
*** $p \leq .001$

statistics. However, the convergence statistics indicate that the model with 90,000 iterations had better convergence than that with only 20,000. Of the parameters, none had a G-R value greater than 1.2 while only one had an MC Error/Standard error value greater than 0.05, and that only slightly.

This had an effect on both the coefficients and the significance levels. The coefficients were in the same direction for both models but were slightly different. Further, the standard deviations were generally smaller with more iterations and only one of the independent variables was not significant (area, which was a control variable).

In other words, increasing the number of burn-in samples as well as the number of iterations run improved the model. It apparently converged for the second run whereas it had not for the first run. The algorithm did this for two reasons. First, by taking a larger number of iterations, the model was more precise. Second, by dropping more initial iterations during the 'burn in' phase (10,000 compared to 5,000), the series apparently reached an equilibrium state before the sample iterations were calculated. The smaller standard errors suggest that there still was a trend when only 5,000 were dropped but had ceased by the time the first 10,000 iterations had been reached.

The point to remember is that one wants a stable series before drawing a sample. If in doubt, run more during the 'burn in' phase. This increases the calculating time, of course, but the results will be more reliable. Once the MCMC algorithm has reached 'equilibrium', it won't take that many additional samples to produce good estimates. We have estimated that 5,000-10,000 additional samples beyond the 'burn-in' sample will produce good results. One can implement this in stages. For example, run the model with the default 25,000 iterations with 5,000 for the 'burn in' (for a total of 20,000 sample iterations from which to base the conclusions). If the convergence statistics suggest that the series has not yet stabilized, run the model again with more 'burn in' samples and, perhaps, more sample iterations.

Monitoring Convergence

A second concern is how to monitor convergence. There appear to be two different approaches. One is a graphical approach whereby a plot of the parameter values is made against the number of iterations (often called *trace plots*). If the chain has converged, then there should be no visible trend in the data (i.e., the series should be flat). The *WinBugs* software package uses this approach, in addition to the MC Error and G-R statistics (BUGS, 2008). For the time being, we have not included a graphical plot of the parameters in this version of *CrimeStat* because of the difficulties in using this plot with the block sampling approach to be discussed shortly.

Also, graphical visualizations, while useful for informing readers, can be misinterpreted. A series that appears to be stable, such as the Baltimore County DWI crash example given above, may actually have a subtle trend. A series can look stable and yet summary statistics such as the G-R statistic and the MC Error relative to the standard error statistic do not indicate convergence.

On the other hand, summary convergence statistics, such as these two measures, are not completely reliable indicators either since a series may only temporarily be stable. This would be especially true for a simulation with a limited number of runs. Both the G-R and MC Error statistics require that at least 2500 iterations be run, with more being desirable. Further, these statistics are not without controversy. Flegal, Haran, & Jones (2008) argue that MCMC standard errors are needed to allow assessment of the accuracy of the estimate while Gelman (2007), in responding to their concerns, argues that a simulation need only be run sufficiently long so that the estimate is more accurate than its standard error. In other words, the precision defines the number of runs needed once the sequence has achieved equilibrium.

Some authors argue that one needs multiple approaches for monitoring convergence (Carlin and Louis, 2000, 182-183). While we would agree with this approach, for the time being we are utilizing primarily the convergence statistics approach.

Statistically Testing Parameters

With an MCMC model, there are two ways that statistical significance can be tested. The first is by assuming that the sampling errors of the algorithm approximate a normal distribution. Thereby, the t-test would be appropriate. In the output table, the t-value is shown, which is the coefficient divided by the standard error. With a simple model, a dependent variable with higher means and adequate sample, this might be a reasonable assumption for a regular Poisson or Poisson-Gamma function. However, for models with many variables and with low sample means, such an assumption is probably not valid (Lord & Miranda-Moreno, 2008). Further, with the addition of many predictor parameters added, the assumption becomes more questionable.

Consequently, MCMC models tend to be tested by looking at the sampling distribution of the parameter and calculating approximate 95% and 99% credible intervals based on the percentile distribution, as illustrated above in Table 17.4.

Proper Specification of a Model

But statistical testing does not just involve testing the significance of the coefficients, whether by asymptotic *t*- or *Z*-tests or by percentiles. A key issue is whether a model is properly specified. On the one hand, a model can be incomplete since there are other variables that could

predict the dependent variable. The Houston burglary model is clearly underspecified since there are additional factors that account for burglaries, as we suggested above.

But, there is also the problem of *overspecifying* a model, that is, including too many independent variables. While the algorithms – MLE or MCMC, can fit virtually any model that is defined, logically many of these models should have never been tested in the first place.

Multicollinearity

The phenomenon of multicollinearity among independent variables is well known, and most statistical texts discuss this. In chapter 15, we briefly discussed multicollinearity among the independent variables. Now, we will show why multicollinearity can be a problem.

In theory, each independent variable should be statistically independent of the other independent variables. Thus, the amount of variance for the dependent variable that is accounted for by each independent variable should be a unique contribution. In practice, however, it is rare to obtain completely independent predictive variables. More likely, two or more of the independent variables will be correlated. The effect is that the estimated standard error of a predictor variable is no longer unique since it shares some of the variance with other independent variables. If two variables are highly correlated, it is not clear what contribution each makes towards predicting the dependent variable. In effect, multicollinearity means that variables are measuring the same thing.

Multicollinearity among the independent variables can produce very strange effects in a regression model. Among these effects are: 1) if two independent variables are highly correlated, but one is more correlated with the dependent variable than the other, the stronger one will usually have a correct sign while the weaker one will sometimes get flipped around (e.g., from positive to negative, or the reverse); 2) two variables can cancel each other out; each coefficient is significant when it alone is included in a model but neither are significant when they are together; 3) one independent variable can inhibit the effect of another correlated independent variable so that the second variable is not significant when combined with the first one; and 4) if two independent variables are virtually perfectly correlated, many regression routines break down because the matrix cannot be inverted. All these effects indicate that there is non-independence among the independent variables.

Aside from producing confusing coefficients, multicollinearity can overstate the predictability of a model. Since every independent variable accounts for some of the variance of the dependent variable, multicollinearity can cause the overall model to ‘improve’ when it probably has not.

A good example of this is a model that we ran relating the number of 1996 crime trips that originated in each of 532 traffic analysis zones in Baltimore County and the City of Baltimore that culminated in a crime committed in Baltimore County. The dependent variable was, therefore, the number of 1996 crimes originating in the zone while there were six independent variables:

1. Population of the zone (1996)
2. An index of relative median household income of the zone (relative to the zone with the highest income)
3. Retail employment in the zone (1996)
4. Non-retail employment in the zone (1996)
5. The number of miles of the Baltimore Beltway (I-695) that passed through the zone
6. Dummy variable indicating whether the Baltimore Beltway passed through the zone.

The last two variables are clearly highly correlated. If a zone has the Baltimore Beltway passing through it, then it has some miles of that freeway assigned to it. The simple Pearson correlation between the two variables is 0.71. Logically, one should not include highly correlated variables in a model. But, what happens if we do this? Table 17.8 illustrates what can happen. Only the coefficients are shown. In the first model, the Beltway miles variable was used along with population, income, retail employment and non-retail employment. In the second model, the dummy variable for whether the Baltimore Beltway passed through the zone or not was used with the four other independent variables. In the third model, both the Beltway miles and the dummy variable for the Baltimore Beltway were both included along with the four other independent variables.

The coefficients for the intercept and the four other independent variables are very similar (and sometimes identical) across the three models. So, look at the two correlated variables. In the first model, the Beltway miles variable is positive, but not significant. In the second model, the Beltway dummy variable is positive and significant. In the third model, however, when both Beltway variables were included, the Beltway miles variable has become negative while the Beltway dummy variable remains positive and significant.

In other words, including two highly correlated variables has caused illogical results. That is, without realizing that the two variables are, essentially, measuring the same thing, one might conclude that the effect of the Beltway passing through a zone is to increase the likelihood that offenders live in that zone but that the effect of having Beltway miles in the zone decreases the likelihood! Any such conclusion is nonsense, of course. In short, do not include highly correlated variables in the same model.

Table 17.8:
Effects of Multicollinearity on Estimation
MLE Poisson-Gamma Model
(N= 532 Traffic Analysis Zones in Baltimore County)

Dependent variable: Number of 1996 crimes that originated in a zone

	(1)	(2)	(3)
Independent Variables	**Model 1**	**Model 2**	**Model 3**
Intercept	1.6437***	1.5932***	1.5964***
Population	0.00045***	0.00045***	0.00045***
Relative Income	-0.0184***	-0.0188***	-0.0188***
Retail Employment	-0.00024*	-0.00026*	-0.00026*
Non-retail Employment	-0.0001***	-0.00013***	-0.00013***
Beltway miles	0.1864n.s.	---	-0.0397n.s.
Beltway	---	0.3194*	0.3496*

n.s. Not significant
* $p \leq .05$
*** $p \leq .001$

How do we know if two or more variables are correlated? There is a simple tolerance test that is included in the MLE models and in the diagnostics utility for the regression module. Tolerance is defined as (repeating equation 15.18, from Chapter 15)

$$Tol_i = 1 - R^2_{j \neq i} \qquad (17.32)$$

where $R^2_{j \neq i}$ is the R-square associated with the prediction of one independent variable with the remaining independent variables in the model. In the example, the tolerance of both the Beltway miles variable and the Beltway dummy variable was 0.49 whereas when each were in the equation by themselves (models 1 and 2), the tolerance was 0.97. The tolerance test should be the first indicator in suspecting too much overlap in two or more independent variables.

The tolerance test is a simple one and is based on normal (OLS) regression. Consequently, it may be erroneous when one or more of the independent variables are highly

skewed. Nevertheless, it is a good indicator of potential problems. When the tolerance of a variable is low, then the variable should be excluded from the model. Typically, when this happens two or more variables will show a low tolerance and the user can choose which one to remove.

How 'low' is low? There is no simple answer to this, but variables with reasonably high tolerance values can have substantial multicollinearity. For example, if there are only two independent variables in a model and they are correlated 0.3, then the tolerance score is 0.91 ($100 - 0.3^2$). While 0.91 appears high, in fact it indicates that there is 9% of overlap between the two variables. *CrimeStat* prints out a warning message about the degree of multicollinearity based on the tolerance levels. But, the user needs to understand that overlapping independent variables can lead to ambiguous and unreliable results. The aim should be to have truly independent variables in a model since the results are more likely to be reliable over time.

Stepwise Variable Entry to Control Multicollinearity

One solution to limiting the number of variables in a model is to use a *stepwise* fitting procedure. There are three standard stepwise procedures (Der & Everitt, 2002, 88-89). In the first procedure, variables are added one at a time (a *forward selection* model). The independent variable having the strongest linear correlation with the dependent variable is added first. Next, the independent variable from the remaining list of independent variables having the highest correlation with the dependent variable *controlling for* the one variable already in the equation is added and the model is re-estimated. In each step, the independent variable remaining from the list having the highest correlation with the dependent variable controlling for the variables already in the equation is added to the model, and the model is re-estimated. This proceeds until either all the independent variables are added to the equation or else a stopping criterion is met. The usual criterion is only variables with a certain significance level are allowed to enter (called a *p-to-enter*).

Second, a *backward elimination* procedure works in reverse. All independent variables are initially added to the equation. The variable with the weakest coefficient (as defined by the significance level and the t- or Z-test) is removed, and the model is re-estimated. Next, the variable with the weakest coefficient in the second model is removed, and the model is re-estimated. This procedure is repeated until either there are no more independent variables left in the model or else a stopping criterion is met. The usual criterion is that all remaining variables pass a certain significance level (called a *p-to-remove*). This ensures that all variables in the model pass this significance level.

The third method is a combination of these procedures, first adding a variable in a forward selection manner but second removing any variables that are no longer significant or

using a backward elimination procedure but allowing new variables to enter the model if they suddenly become significant.

There are advantages to each approach. A fixed model allows specified variables to be included. If either theory or previous research has indicated that a particular combination of variables is important, then the fixed model allows that to be tested. A stepwise procedure might drop one of those variables. On the other hand, a stepwise procedure usually can obtain the same or higher predictability than a fixed procedure.

Within the stepwise procedures, there are also advantages and disadvantages to each method, though the differences are generally very small. A forward selection procedure adds variables one at a time. Thus, the contribution of each new variable can be seen. On the other hand, a variable that is significant at an early stage could become insignificant at a later stage because of the unique combinations of variables. Similarly, a backward elimination procedure will ensure that all variables in the equation meet a specified significance level. But, the contribution of each variable is not easily seen other than through the coefficients. In practice, one usually obtains the same model with either procedure, so the differences are not that critical.

A stepwise procedure will not guarantee that multicollinearity will be removed entirely. However, it is a good procedure for narrowing down the variables to those that are significant. Then, any co-linear variables can be dropped manually and the model re-estimated.

In the normal and MLE Poisson routines, there is a backward elimination procedure whereby variables are dropped from an equation if their coefficients are not significant.

Overfitting

Overfitting is a more general phenomenon of including too many variables in an equation (Radford, 2006; Nannen, 2003). With the development of Bayesian models, this has become an increasing occurrence because the models, usually estimated with the MCMC algorithm, can fit an enormous number of parameters. Many of these models estimate parameters that are properties of the functions used (called *hyperparameters*) rather than just the variables input as part of the data. In the Poisson-Gamma-CAR model, for example, we estimate the dispersion parameter (ψ) and a general Φ function. Phi (Φ), in turn, is a function of a global component (Rho, ρ), a local component (Tauphi τ_Φ), and a neighborhood component (Alpha -α).

These parameters are part of the functions and are not data. But, since they can vary and are often estimated from the data, there is always the potential that they could be highly correlated and, thereby, cause ambiguous results to occur. Unfortunately, there are not good diagnostics for multicollinearity among the hyperparameters, as there is with the tolerance test.

But, the problem is a real one and one that the user should be cognizant. Sometimes an MCMC or MLE model fails to converge properly, meaning that it either did not finish or else produced inconsistent results from one run to another. We usually assume that the probability structure of the space being modeled is too complex for the model that we are using. And, while that may be true, it is also possible that there is overlap in some of the hyperparameters. In this case, one would be better off choosing a simpler model – one with fewer hyperparameters, than a more complex one.

Condition Number of Matrix

In other words, a user should be very cautious about overfitting models with too many variables, both the data variables and those estimated from functions (the hyperparameters). We have included a condition matrix test for the distance matrix in the Poisson-Gamma-CAR/SAR model. The condition number of a matrix is an indicator of how amenable it is to digital solution (Wikipedia, 2010b). A matrix with a low condition number is said to be well conditioned whereas one with a high number is said to be ill-conditioned. With ill-conditioned matrices, the solutions are volatile and inconsistent from one run to another. How 'high' is high? Numbers higher than, say, 400 are generally ill-conditioned while low condition numbers (say, under 100) are well conditioned. Between 100 and 400 is an ambiguous area. For the Poisson-Gamma-CAR model, if you see a condition number higher than 100, be cautious. If you see one higher than 400, assume the results are completely unreliable with respect to the spatial component.

Overfitting and Poor Prediction

There is also a question about the extent to which a model that is fit is reliable and accurate for predicting a data set which is different. Without going into an extensive literature review, a few guidelines can be given. The Machine Learning computing community concentrates on *training* samples in order to estimate parameters and then using the estimated models to predict a *test* sample (another data set). In general, they have found that simple models do better for prediction than complicated models. One can always fit a particular data set by adding variables or adding complexity to the mathematical function. On the other hand, the more complex the model – the more independent variables in it and the more specified hyperparameters, generally the model will do worse when applied to a new data set. Nannen (2003) called this the *paradox of overfitting*, and it is a rule that a user would be well advised to follow. Try to keep your models simple and reliable. In the long run, simple models with well-defined independent variables will generally do better for prediction.

Improving the Performance of the MCMC Algorithm

Most medium- and large police departments use large datasets, such as calls for service, crime reports, motor vehicle crash reports and other data sets. The largest police departments have huge data sets, constituting millions of records. Further, these data are being collected on a continual basis. *CrimeStat* was developed to handle fairly large data sets and the routines are optimized for this.

However, large data sets pose a problem for multivariate modeling in a number of ways. First, they pose a computing problem in terms of the processing of information. As the number of records goes up, the demand for computer resources increases exponentially. For example, consider the problem of calculating a distance matrix for use in, say, the Poisson-Gamma-SAR model. If each number is represented by 64 bits (double precision), then the amount of memory space required is a function of K^2*64 where K is the number of records. For example, if there are 10,000 records (a relatively small database by police standards), then the amount of memory required will be 10,000*10,000*64 = 6.4 billion bits (or 800 Mb). On the other hand, if the number of records is 100,000, then the memory demand goes up to 80,000 Mb (or 80 Gb). That such databases take a long time to be analyzed is understandable.

Second, large data sets pose problems for interpretation. The 'gold standard' for testing of coefficients or even the overall fit of a model has been to compare the coefficients to 0. This follows from traditional statistics (whom the Bayesians call *frequentists*) whereby a particular statistic (in this case, a regression coefficient) is compared to a 'null hypothesis' which is usually 0. However, with large datasets, especially with extremely large datasets, virtually all coefficients will be significantly different from 0, no matter how they are tested (with t-tests or with percentiles). In this case, 'significance' does not necessarily mean 'importance'. For example, if you have a data set of one million records and plug in a model with 10 independent variables, the chances are that the majority of the variables will be significantly different than 0. This does not mean that the variables are important in any way, only that they account for some of the variance of the dependent variable greater than what would be expected on the basis of chance.

The two problems interact when a user works with a very large dataset. The routines may have difficulty calculating the solution and the results may not necessarily be very meaningful. This will be particularly true for complex models, such as the Poisson-Gamma-CAR which will be discussed in chapter 19. An example will illustrate this. With an Intel 2.4 Ghz computer with a dual core, we ran a model with three independent variables on a scalable dataset; that is, we took a large dataset and sampled smaller subsets of it. We then tested the MCMC Poisson-Gamma and MCMC Poisson-Gamma-CAR models with subsets of different size. Table 17.9 present the results.

As can be seen, the calculation time went up exponentially with the sample size. Further, with the spatial Poisson-Gamma-CAR model, a limit was reached. Because the routine was calculating the distance between each observation and every other observation as part of the spatial weight coefficients, the memory demands blow up very quickly. The non-spatial Poisson-Gamma model can be run on larger datasets (we have run them on sets as large as 100,000 records) but the spatial model cannot be. Even with the non-spatial model, the calculation time for a very large dataset goes up very substantially with the sample size.

Table 17.9:
Effects of Sample Size on Calculations
(Second to Complete)

Sample size	Poisson-Gamma	Poisson-Gamma-CAR
125	23	67
250	43	163
500	81	480
1,000	160	1,569
2,000	305	6,000
4,000	622	25,740
5,000	762	43,740
8,000	1,247	Unable to complete
12,000	1,869	Unable to complete
15,000	2,412	Unable to complete
20,000	3,278	Unable to complete

Scaling of the Data

There are several things that can be done to improve the performance of the MCMC algorithm with large datasets. The first is to scale the data, either by reducing the number of digits that represent each value or by standardizing by Z-scores. There are different ways to scale the data, but a simple one is to move the decimal places. For example, if one of the variables is median household income and is measured in tens of thousands (e.g., 55,000, 135,000), then these values can be divided by 1000 so that they represent ‘per 1000’ (i.e., 55.0 and 135.0 in the example).

To illustrate, we ran a single-family housing value model on a large data set of 588,297 single-family home parcels. The data came from the Harris County Appraisal District and the model related the 2007 assessed value against the square feet of the home, the square feet of the parcel, the distance from downtown Houston and two dummy variables - whether the home had

received a major remodeling between 1985 and 2007 and whether the parcel was within 200 feet of a freeway. The valuations were coded as true dollars and were then re-scaled into units of 'per 1000' (e.g., 45,000 became 45.0). When the data were in real units, the time to complete the run was 20.8 minutes for the MCMC Poisson-Gamma using the Block Sampling Method (see below). When the data were in units of thousandths, the time to complete the run was 15.3 minutes for the MCMC Poisson-Gamma.

In other words, scaling the data by reducing the number of decimal places led to an improvement in calculating time of around 25% for the MCMC model. The effects on an MLE model will be even more powerful due to the different algorithm used. The point is, scaling your data will pay in terms of improving the efficiency of runs.

Block Sampling Method for the MCMC

Another solution is to sample records from the full database and run the MCMC algorithm on that sample. In the MCMC literature, drawing a sub-sample is called 'thinning' the sample (Link & Eaton, 2011). Essentially, a sub-sample is drawn and the MCMC algorithm is run. It is clearly much faster to run a sub-sample than the entire database. However, the problem with this approach, as pointed out by McEachern & Berliner (1994) is that it will be less precise than by running the full database. The reason is that there is sampling error and that the results from any one sub-sample might deviate from the full database.

With the block sampling method, on the other hand, multiple subsamples are drawn with the overall statistics based on a summary of the individual samples. That is, a first sub-sample is drawn and run through the MCMC algorithm. The statistics from the run are calculated. Then, the process is repeated with another sample, and the statistics are calculated on this sample. Then, the process is repeated again and again. We call this the *block sampling method* and it has been implemented in *CrimeStat*. The advantage over a thinned sample is that, because of the Central Limit Theorem, the summary statistics for the repeated samples will converge towards the summary statistics of the full database with much smaller sampling error.

With the block sampling method, the user defines three parameters for controlling the sampling:

1. The block sampling threshold – the size of the database beyond which the block sampling method will be implemented. For example, the default block sampling threshold is set at 6,000 observations, though the user can change this. With this default, any dataset that has fewer than 6,000 records/observations will be analyzed with the full database. However, any dataset that has 6,000 records or more will cause the block sampling routine to be implemented. Note that the user run the entire

dataset, no matter how long it takes, by setting the block sampling threshold to be greater than the number of records in the dataset.

2. Average block size – the expected block size of a sample from the block sampling method. The default is 400 records thought the user can change this. The routine defines a sampling interval, based on *n/N* where *n* is the defined average block size and *N* is the total number of records. For drawing a sample, however, a uniform random number from 0 to 1 is drawn and compared to the ratio of *n/N*. If the number is equal to or less than this ratio (probability), then the record is accepted for the block sample; if the number is greater than this ratio, the record is not accepted for the block sample. Thus, any one sample may not have exactly the number of records defined by the user. But, on average, the average sample size over all runs will be very close to the defined average block size though the variability is high.

3. Number of samples – the number of samples drawn. The default is 25 though the user can change this. We have found that 20-30 samples produce very reasonable results.

The routine then proceeds to implement the block sampling method. For example, if the user keeps the default parameters, then the block sampling method will only be implemented for databases of 6,000 records or more. If the database passes the threshold, then each of the 25 samples are drawn with, approximately, 400 records per sample. The MCMC algorithm is run on each of the samples and the statistics are calculated. After all 25 samples have been run, the routine summarizes the results by averaging the summary statistics (likelihood, AIC, BIC/SC, etc), the coefficients, the standard errors, and the percentile distribution. The results that are printed represent the average over all 25 samples.

GUIDELINE:

Note that MCMC models can take a very long time to calculate. For large datasets, we recommend using the block sampling method. A rough rule-of-thumb is that for non-spatial MCMC models, the block sampling method should be used for 6,000 or more cases while for spatial MCMC models, the block sampling method should be used for 2,000 or more cases. Of course, this will depend on the amount of available RAM as well as the processing speed of the computer.

We have found that this method produces very good approximations to the full database. For several datasets, we have compared the results of the block sampling method with running the full database through the MCMC routine. The means of the coefficients appear to be unbiased estimates of the coefficients for the full database. Similarly, the percentiles appear to be very close, if not unbiased, estimates of the percentiles for the full database. On the other hand, the standard errors appear to be biased estimates of standard errors of the full database.

The reason is that they are calculated from a sample of n observations where the standard errors of the full database are calculated from N observations. An adjusted standard error is produced which approximates the true standard error of the full database. It is defined as;

$$AdjStd.Err = StdErr_{block} * \sqrt{\frac{\bar{n}}{N}} \tag{17.33}$$

where $StdErr_{block}$ is the average standard error from the k samples, N is the total number of records, and $\bar{n}$ is the average block size (the empirical average, not the expected sample size). This is only output when the block sampling method is used.

Comparison of Block Sampling Method with Full Dataset

Test 1

A test was constructed to compare the block sampling method with the full MCMC method on two datasets. The first dataset contained 4000 road segments in the Houston metropolitan area and the model that was run was a traffic model relating vehicle miles traveled (VMT - the dependent variable) against the number of lanes, the number of lane miles, and the volume-to-capacity ratio of the segment. It is not a very meaningful model but was used to test the algorithm.

The dataset was tested with the MCMC model using all records (the full dataset) and the block sampling method. For simplicity, the variables have been called $X_1 \ldots X_k$. The significance levels of the coefficients for the full dataset based on the t-test are shown, since these are based on the estimated standard errors rather than the adjusted standard errors.

Table 17.10 shows the results of the traffic dataset. Comparing the full sample results with the block sample results, the coefficients are very close to each other, within the second decimal place. Similarly, the adjusted standard errors are very close within the third decimal place. On the other hand, the block sampling method took 11.2 minutes to run compared to only

Table 17.10:
Comparing Block Sampling Method with Full Database
MCMC Poisson-Gamma Model
Houston Traffic Dataset
(Time to Complete)

Dependent variable = Vehicle Miles Traveled

	Full dataset		**Block Sampling method**		
	(N=4000)		(n = 402.9)		
Iterations:	20,000		20,000		
Burn in:	5,000		5,000		
Number of samples:	1		20		
Time to complete run:	7.7 minutes		11.2 minutes		
Variable	**Coefficient**	**Std. Error**	**Coefficient**	**Std. Error**	**Adj. Std. Error**
Intercept	4.5414***	0.045	4.5498***	0.140	0.044
X_1	0.6254***	0.022	0.6267***	0.066	0.021
X_2	0.8502***	0.020	0.8618***	0.064	0.020
X_3	2.4163***	0.049	2.3938***	0.154	0.049

Significance of block sampling method based on unadjusted standard error
*** $p \leq .001$

7.7 for the full dataset. With a dataset of this size (N=4000), there was no advantage for the block sampling method even though it produced very similar results.

Now, let's take a more complicated dataset. The second represented 97,429 crimes committed in Manchester, England. It is part of a study on gender differences in crime travel (Levine & Lee, 2012). The model related the journey to crime distance against 14 independent variables involving spatial location, land use, type of crime, ethnicity of the offender, prior conviction history, and gender.

Test 2

Table 17.11 shows the results of the journey to crime dataset. Not all of the variables were significant, according to the t-test of the full dataset. In this case, there were greater discrepancies in the coefficients between the full dataset and the block sampling method. The signs of the coefficients were identical for all parameters except X_{10}, which was not significant. For all parameters, though, the coefficient for the full dataset was within the 95% credible interval of the block sampling method. That is, since this is a sample, the sampling error of the block sampling method incorporates the coefficient for the full dataset for all 16 parameters.

The adjusted standard errors from the block sampling method were quite close to the standard errors of the full dataset; the biggest discrepancy was 0.004 for variable X_6 and is about 15% larger. Most of the adjusted standard errors are within 10% of the standard error for the full dataset, and three are exactly the same. Further, where there is a discrepancy, the adjusted standard errors were slightly larger, suggesting that this is a conservative adjustment.

In short, the block sampling method produced reasonably close results to that of the full dataset for both the coefficients and the standard errors. Given that this model was a very complex one (with 14 independent variables), the fit was good. The biggest advantage of the block sampling method, on the other hand, is the efficiency of it. The block sampling method took 222.7 minutes to run compared to 4,855.1 minutes for the full dataset, an improvement of more than 20 times! Running a large dataset through the MCMC algorithm is a very time consuming process. The block sampling approach produced reasonably close results in a much shorter period of time.

Statistical Testing with Block Sampling Method

Regarding statistical testing of the coefficients, however, we think that the modeled standard errors (or percentiles) be used rather than the adjusted errors. The adjusted standard error is an approximation to the full dataset *if* that dataset had been run. In most cases, it will not have been run. On the other hand, the standard errors estimated from the block sampling method and the percentile distribution were the products of running the individual samples. The errors are larger because the samples were much smaller. But, because this was the method used, statistical inferences should be based on the sample.

What to do if there is a discrepancy? For some datasets, the coefficients from the block sampling method will not be significant whereas they would be if the full dataset was run. In the Manchester example above, only 3 of the coefficients were significant using the block sampling method compared to 14 for the full dataset. This brings up a statistical dilemma. Does one adopt the adjusted standard errors and then re-test the coefficients using the asymptotic t-test or does

Table 17.11:

Comparing Block Sampling Method with Full Database MCMC Poisson-Gamma Model

Manchester Journey to Crime Dataset

(Time to Complete)

Dependent variable = Distance traveled

	Full dataset		**Block Sampling method**		
	(N = 97,429)		(n=402.8)		
Iterations:	100,000		100,000		
Burn in:	10,000		10,000		
Number of samples:	1		30		
Time to complete:	4,855.1 minutes		222.7 minutes		
Variable	**Coefficient**	**Std. Error**	**Coefficient**	**Std. Error**	**Adj. Std. Error**
Intercept	0.2096***	0.018	0.2103	0.321	0.021
X_1	0.8871***	0.025	1.0135*	0.430	0.028
X_2	0.3311***	0.018	0.3434	0.294	0.019
X_3	-0.2274***	0.012	-0.2751	0.199	0.013
X_4	-0.2820***	0.014	-0.3137	0.231	0.015
X_5	0.2525***	0.016	0.3099	0.256	0.016
X_6	0.3560***	0.027	0.3783	0.488	0.031
X_7	0.0753***	0.013	0.1092	0.214	0.014
X_8	0.1766***	0.021	-0.0030	0.374	0.024
X_9	0.1880***	0.023	0.1326	0.406	0.026
X_{10}	0.0135n.s.	0.016	-0.0070	0.268	0.017
X_{11}	-0.5697***	0.016	-0.6759	0.265	0.017
X_{12}	0.0042n.s.	0.014	0.0521	0.226	0.015
X_{13}	-0.2214***	0.016	-0.2755	0.262	0.017
X_{14}	0.0056***	0.001	-0.00004	0.016	0.001
Error	-0.7299***	0.008	-0.7062***	0.139	0.009

Based on asymptotic t-test:

n.s. Not significant

* $p \leq .05$

*** $p \leq .001$

one accept the estimated standard errors and the percentiles? Our opinion is to do the latter. The former is making an assumption (and a big one) that the adjusted standard errors will be a good approximation to the real ones. In these two datasets, this appears to be the case. But, we have no theoretical basis for assuming that. It has just worked out for these and a couple of other datasets that we have tested.

Therefore, the choice for a researcher is to do one of three things if some of the coefficients are not significant using the block sampling method when it appears that they might be if the full dataset would be used.

First, one could always run the full dataset through the MCMC algorithm. If the dataset is large, then it will take a long time to calculate. But, if it is important, then the user should do that. Note that it will be possible to do this only for the Poisson-Gamma model and not for the Poisson-Gamma-CAR/SAR spatial model.

Second, the researcher could try to tweak the MCMC algorithm to increase the likelihood of finding statistical significance for the coefficients increasing the number of iterations to improve the precision of the estimate and by increasing the average sample size of the block sample. If 400 samples were not sufficient, perhaps 600 would be? In doing this, the efficiency advantage of the block sampling method becomes less important compared to improving the accuracy of the estimates.

Third, the researcher can accept the results of the block sampling method and 'live' with the conclusions. If one or more variables was not significant using the block sampling method (which, after all, was based on 20 to 30 samples of around 400 records each), then the variables are probably not important. In other words, running the MCMC algorithm on the full dataset or increasing the sample size of the block samples may find statistical significance in one or more variables. But, the chances are that the variables are not very important, from a statistical perspective.

In our experience, the strongest variables are significant with the block sampling scheme. Perhaps the researcher or analyst should focus on those and build a model around them, rather than scouring for other variables that have very small effect? In short, our opinion is that a smaller, but more robust, model is better than a larger, more volatile one. In terms of understanding, the major variables need to be isolated because they contribute the most to the development of theory. In terms of prediction, the strongest variables will also have the biggest impact. Elegance in a model should be the aim, not a comprehensive list of variables that might be important but probably are not.

References

Anselin, L. (2002). Under the hood: Issues in the specification ad interpretation of spatial regression models, *Agricultural Economics*, 17(3), 247-267.

Besag, J., Green, P., Higdon, D. & Mengersen, K. (1995). Bayesian computation and stochastic systems (with discussion), *Statistical Science*, 10, 3-66.

BUGS (2008). *The BUGS (Bayesian Inference Using Gibbs Sampling) Project*. MRC Biostatistics Unit, University of Cambridge: Cambridge. http://www.mrc-bsu.cam.ac.uk/bugs/. Accessed March 23, 2010.

Cameron, A. Colin & Trivedi, Pravin K. (1998). *Regression Analysis of Count Data*. Cambridge University Press: Cambridge, U.K.

Carlin, B. P. & Louis, T. A. (2008). *Bayesian Methods for Data Analysis, Third Edition*, Chapman and Hall/CRC: Boca Raton.

Cormen, T. H., Leiserson, C. E., Rivest, R. L. & Stein, C. (2009). Ch. 16: Greedy algorithms, *Introduction to Algorithms*, MIT Press: Cambridge, MA.

Denison, D.G.T., Holmes, C. C., Mallick, B. K. & Smith, A. F. M (2002). *Bayesian Methods for Nonlinear Classification and Regression*. John Wiley & Sons, Ltd: Chichester, Sussex.

Der, G. & Everitt, B. S. (2002). *A Handbook of Statistical Analyses using SAS*. Chapman & Hall/CRC: London.

De Smith, M., Goodchild, M. F., & Longley, P. A. (2007). *Geospatial Analysis* (second edition). Matador: Leicester, U.K.

Dijkstra, E. W. (1959). A note on two problems in connection with graphs, *Numerische Mathematik*, 1, 269-271.

El-Basyouny K. & Sayed, T. (2009). Collision prediction models using multivariate Poisson-lognormal regression. *Accident Analysis & Prevention*, 41(4), 820-828.

Flegal, J. M., Haran, M., & Jones, G. L. (2008). Markov Chain Monte Carlo: Can we trust the third significant figure?, *Statistical Science*, 23 (2), 250-60.

References (continued)

Geedipally, S.R., Lord, D., & Dhavala, S.S. (2012). The Negative-Binomial-Generalized-Lindley Generalized Linear Model: Characteristics and application using crash data. *Accident Analysis & Prevention*, 45 (2), 258-265.

Gelman, A. (2007). Markov Chain Monte Carlo standard errors. Note on paper by Flegal, Haran & Jones. http://andrewgelman.com/2007/04/markov_chain_mo/.

Gelman, A. (1996). Inference and monitoring convergence. In Gilks, W. R., S. Richardson, & D. J. Spiegelhalter (eds), *Markov Chain Monte Carlo in Practice*, Chapman and Hall: London.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2004). *Bayesian Data Analysis* (second edition). Chapman and Hall/CRC: Boca Raton, FL.

Gelman, A. & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences (with discussion), *Statistical Science,* 7, 457-511.

Ghitany, M.E., Atieh, B., Nadarajah, S. (2008). Lindley distribution and its application. *Mathematics and Computers in Simulation*, (78), 39-49.

Goldfield, S. M., Quandt, R. E., & Trotter, H. F. (1966). Maximization by quadratic hill-climbing, *Econometrica*, 34 (3), 541-551.

Guikema, S.D. & Coffelt, J. P. (2008). A flexible count data regression model for risk analysis, *Risk Analysis*, 28 (1), 213–223.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov Chains and their applications, *Biometrika*, 57, 97-109.

Lambert, D. (1992). Zero-Inflated Poisson Regression, With an Application to Defects in Manufacturing. *Technometrics*, 34 (1), 1-14.

Lee, P. M. (2004). *Bayesian Statistics: An Introduction* (third edition). Holder Arnold: London.

Leonard, T. & Hsu, J.S.J. (1999). *Bayesian Methods: An Analysis for Statisticians and Interdisciplinary Researchers*. Cambridge University Press: Cambridge.

References (continued)

Levine, N. (2011). Spatial variation in motor vehicle crashes by gender in the Houston Metropolitan Area. *Proceedings of the 4th International Conference on Women's Issues in Transportation. Volume II: Technical Papers*, Transportation Research Board: Washington, DC. 12-25. http://onlinepubs.trb.org/onlinepubs/conf/cp46v2.pdf.

Levine, N. & Canter, P. (2011). Linking origins with destinations for DWI Motor Vehicle Crashes: An application of crime travel demand modeling. *Crime Mapping*, 3, 7-41.

Levine, N. & Lee, P. (2012).). "Crime travel of offenders by gender and age in Manchester, England". In press, Michael Leitner (ed), *Crime Modeling and Mapping Using Geospatial Technologies,* Springer.

Lindley, D.V. (1958). Fiducial distributions and Bayes' theorem. *J. R. Stat. Soc.*, (20), 102-107.

Link, W. A. & Eaton, M. J. (2011). On thinning of chains in MCMC. *Methods in Ecology and Evolution*, June. http://onlinelibrary.wiley.com/doi/10.1111/j.2041-210X.2011.00131.x/abstract.

Lord, D. & Geedipally, S.R. (2011). The negative binomial–Lindley distribution as a tool for analyzing crash data characterized by a large amount of zeros. *Accident Analysis & Prevention*, 43 (5), 1738-1742.

Lord, D. (2006). Modeling motor vehicle crashes using Poisson-gamma models: Examining the effects of low sample mean values and small sample size on the estimation of the fixed dispersion parameter. *Accident Analysis and Prevention*, 38, 751-766.

Lord, D. & Miranda-Moreno, L. F. (2008). Effects of low sample mean values and small sample size on the estimation of the fixed dispersion parameter of Poisson-gamma models for modeling motor vehicle crashes: A Bayesian Perspective. *Safety Science*, 46 (5), 751-770.

Lord, D., Manar, A., & Vizioli, A. (2005). Modeling Crash-Flow-Density and Crash-Flow-V/C Ratio for Rural and Urban Freeway Segments. *Accident Analysis & Prevention*. 37 (1), 185-199.

Lynch, Scott M. (2007). *Introduction to Applied Bayesian Statistics and Estimation for Social Scientists*. Springer: New York.

McEachern, S. N. & Berliner, L. M. (1994). Subsampling the Gibbs Sampler, *The American Statistician*, 48, 188–190.

References (continued)

Ma, J., Kockelman, K. M., & Damien, P. (2008). A multivariate Poisson-lognormal regression model for prediction of crash counts by severity, using Bayesian methods. *Accident Analysis & Prevention*, 40 (3), 964-975.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equations of state calculations by fast computing machines, *Journal of Chemical Physics*, 21, 1087-91.

Miaou, S. P. (2006). Coding instructions for the spatial regression models in CrimeStat. Unpublished manuscript. College Station, TX.

Mitra, S. & Washington, S. (2007). On the nature of over-dispersion in motor vehicle crash prediction models, *Accident Analysis and Prevention*, 39, 459-468.

Nannen, V. (2003). *The Paradox of Overfitting*. Artificial Intelligence, Rijksuniversitat: Groningen, Netherlands. http://volker.nannen.com/pdf/the_paradox_of_overfitting.pdf. Accessed March 11, 2010.

Ntzourfras, I. (2009). *Bayesian Modeling using WinBugs*. Wiley Series in Computation Statistics, Wiley: New York.

Park, B. J. (2009). Note on the Bayesian analysis of count data. From Park, Byung-Jung PhD thesis, Texas A & M University: College Station, TX.

Radford, N. (2006). The problem of overfitting with maximum likliehood . CSC 411: Machine Learning and Data Mining, University of Toronto: Toronto, CA. http://www.cs.utoronto.ca/~radford/csc411.F06/10-nn-early-nup.pdf Accessed March 11, 2010.

Radford, N. (2003). Slice sampling, *Annals of Statistics*, 31(3), 705-767.

Sellers, K. S. & Shmueli, G. (2010), A flexible regression model for count data, *Annals of Applied Statistics*, 4 (2), 943-961.

So, A. M., Ye, Y., & Zhang, J. (2007). Greedy algorithms for metric facility location problems. In Gonzalez, T. F. (Ed), *Handbook of Approximation Algorithms and Metaheuristics*, CRC Computer & Information Sciences Series, Chapman & Hall/CRC: Boca Raton, FL, Chapter 39.

References (continued)

Spiegelhalter, D.J., Best, N.G., Carlin, B.P., & Van der Linde, A. (2002) Bayesian measures of model complexity and fit. Journal of the Royal Statistical Society, Series B, Vol. 64, No. 4, pp. 583-639.

Train, K. (2009). *Discrete Choice Methods with Simulation* (2nd edition). Cambridge University Press: Cambridge.

Wikipedia (2012). Metropolis-Hastings algorithm, *Wikipedia*. http://en.wikipedia.org/wiki/Metropolis%E2%80%93Hastings_algorithm. Accessed July 2, 2012.

Wikipedia (2010a). Greedy algorithm, *Wikipedia*. http://en.wikipedia.org/wiki/Greedy_algorithm. Accessed March 12, 2010.

Wikipedia (2010b). Condition number. *Wikipedia.* *http://en.wikipedia.org/wiki/Condition_number*. Accessed March 19, 2010

Chapter 18:

Binomial Regression Modeling

Ned Levine
Ned Levine & Associates
Houston, TX

Dominique Lord
Zachry Dept. of
Civil Engineering
Texas A & M University
College Station, TX

Byung-Jung Park
Korea Transport Institute
Goyang, South Korea

Table of Contents

Chapter 18:
Binomial Regression Modeling

Introduction

In this chapter, we discuss binomial regression models as applied to ungrouped data. Users should be familiar with the materials in Chapter 15, 16, and 17 before attempting to read this chapter. A good background in statistics is necessary to understand the material.

These are models that are applied to individual cases (records) and where the dependent variable has only two responses, expressed as 0 and 1. They are part of a family of regression models called *limited dependent variables* where the range of possible values is restricted. They are sometimes called *restricted dependent variables* or, if the restriction is one side of the distribution only, *censored dependent variables* or even *truncated dependent variables*. In chapters16 and 17, we discussed the Poisson family of regression models. This is a limited dependent variable in that 0 is the minimum since the Poisson models counts (i.e., for which the minimum number is 0).

However, with binomial regression models, the limitations are on both sides of the distribution, namely a minimum value of 0 and a maximum value of 1. Such a model is useful when there is a discrete choice between two alternatives, for example 'yes' versus 'no' on a survey or 'males' versus 'females' as a demographic distinction or even 'under age 65' versus '65 or older' for an age group distinction. The key is that there can only be two alternatives and that they have to be identified as '1' or '0'.

The underlying model is that of a probability, which also varies from 0 to 1. The problem, however, is that with a binomial variable, the underlying probabilities are not measured but only inferred from a discrete, binomial choice. Thus, the models that have been proposed estimate the underlying probability using only the two alternative values for the dependent variable.

The two models that we will examine are the logistic (usually called logit) model and the probit model, the two most common forms for estimating the underlying probabilities. Binomial functions are also the basic building block for discrete choice models that comprise models for estimating probabilities when there are more than two alternatives. These will be discussed in chapters 21 and 22.

Generalized Linear Models

The Generalized Linear Model (GLM) is a family of functions for estimating the relationship of many functions to a set of linear predictors in a regression framework (Liao, 1994; McCullagh & Nelder, 1989). It relates the expected mean of a distribution, μ, to a *link function*, η, which, in turn, is related to a set of linear predictors,

$$\eta_i = \beta_0 + \sum_1^K \beta_K X_{iK} + \epsilon_i \quad (18.1)$$

where, for case i, β_0 is the intercept, β_K is the coefficient of each of the *K* independent variables, X_{iK}, and ε_i is an error term. The coefficients are applied to individual records, i. To simplify notation, we will drop the case letter but it will be understood that the parameters apply to individual cases.

Not all functions can be estimated this way, essentially only those that belong to the exponential family of functions and which have a concave, closed-form solution. In the classic linear form of the GLM model (Ordinary Least Squares, or OLS), which we examined in chapter 15, the link function is simply the mean itself,

$$\eta = \mu \quad (18.2)$$

In the Poisson form, which we examined in Chapters 16 and 17, the link function is the natural log of the mean,

$$\eta = Ln(\mu) \quad (18.3)$$

This brings us to binomial regression and the two forms which are also part of the GLM family. First, there is the ***logistic*** **(**or ***logit*****)** model where the link function is related to the *log of the odds* ,

$$\eta = Ln[(\mu/(1-\mu)] \quad (18.4)$$

Second, there is the ***probit*** model where the link function is related to the inverse of the standard normal cumulative distribution,

$$\eta = \Phi^{-1}(\mu) \quad (18.5)$$

There are other link functions that can be expressed by the GLM model, but we will concentrate on the logit and probit models. The logit is the most common way to relate a binomial outcome to a set of independent predictors with the probit used less often. In practice,

the logit and probit models produce more or less the same results (Greene, 2008). They differ primarily in the tails of the distribution with the probit approaching the limiting ends of the probability more quickly than the logit (Chen & Tsurumi, 2011; Hahn & Soyer, 2005).

Logistic Model

Logit

The logistic model is related to the binomial probability. It is usually called a logit model because it takes the log of the odds (*logit* and *log of the odds* are equivalent terms). If an event has two possible outcomes expressed as 0 and 1 (e.g. 'head' or 'tails', 'males' or 'females', 'A' or 'B', or any other binomial alternative), then its probability can be estimated for successive independent outcomes from *N* observations. Let *p* be the probability of obtaining one of the outcomes which takes the value 1 (call it A) with *1-p* (sometimes called *q*) being the probability of obtaining the outcome that takes the value 0 (call it B).

Binomial Distribution

The binomial distribution defines the distribution of alternative A in *O* successive samples by (Wikipedia, 2011a; Hosmer & Lemeshow, 2001):

$$P(Y = O) = \binom{N}{O} p^K (1-p)^{(N-K)} \tag{18.6}$$

where *P(Y=O)* is the probability of obtaining exactly *O* instances from *N* observations, *p* is the probability of obtaining A for one observation, and $\binom{N}{O}$ is the number of *combinations* for getting exactly *O* outcomes for A and *N-O* outcomes for B, and is expressed by

$$\binom{N}{K} = \frac{N!}{O!(N-O)!} = \frac{N(N-1)(N-2)\ldots(1)}{[O(O-1)(O-2)\ldots1][(N-O)(N-O-1)(N-O-2)\ldots1]} \tag{18.7}$$

where *!* is a factorial.

The probability is always estimated with respect to A (or the probability of achieving a 1). For example, if *p* for A is 0.4 (and, therefore, the probability for B is *1-p*, or 0.6) and there are 10 successive observations, each of which is independent, the probability of getting exactly 4 instances of *p* and 6 instances of *(1-p)* is:

$$\binom{100}{40} = \frac{10!}{4!(6)!} = \frac{(10)(9)(8)\ldots(1)}{[(4)(3)(2)(1)][(6)(5)(4)(3)(2)*(1)]} = (210)(.4)^4(.6)^6 = 0.2508$$

The probability is often called a Bernoulli trial, named after the Swiss mathematician Jacob Bernoulli (1654-1705; Wikipedia, 2011b; Hosmer & Lemeshow, 2001). Notice that the successive outcomes (sometimes called 'trials' or 'experiments') must be independent. That is, the probability of achieving either of the two outcomes in an observation (or trial) must be constant across observations and unrelated to prior observations. That is, the outcomes are random and independent. The assumption of independence of each observation (or trial or experiment) is different from the MCMC method that we discussed in Chapter 17 where the results of each sample depend on the value from the previous sample.

In a binomial experiment, there are exactly *N* observations and the function P(X=K) is called the *binomial distribution*. The binomial distribution, in turn, is a special case of the *Poisson distribution* which is a sum of *N* independent Bernoulli trials with a constant probability for each choice. The Poisson distribution expresses the probability of a given number of events occurring (in time or in space) if these events are independent and occur with a known probability. In other words, the Poisson distribution, which we examined in Chapters 16 and 17, is a more general case of the binomial distribution and, in turn, is part of the GLM family of models. The binomial distribution becomes the Poisson for very large samples (i.e., as N approaches infinity) and when p is very small (Lord, Washington, & Ivan, 2005).

Odds Ratio

Another way to look at the probability of obtaining alternative A compared to alternative B is through the *Odds Ratio* (or just Odds). This is the ratio of *p* to *1-p*, or

$$Odds\ ratio = \frac{p}{1-p} \tag{18.8}$$

and expresses the relative likelihood of obtaining outcome A relative to outcome B. For example, if p is 0.4 then 1-p is 0.6 and the odds ratio is 0.4/0.6 = 0.667. Alternatively, if p is 0.7 and 1-p is 0.3, then the odds ratio is 0.7/0.3 = 2.33. Finally, if p and 1-p are equal (i.e., both are 0.5), then the odds ratio is 0.5/0.5 = 1. Note that with the odds ratio, a value greater than 1 indicates that A is more likely to occur than B while a value less than 1 indicates that A is less likely to occur than B (or, conversely, B is more likely to occur than A). Thus, this means that A is about 2.3 times more likely to occur than B in the example.

Log of the Odds Ratio

Since the logit is the natural log of the odds ratio, if we let the probability, *p*, represent an estimate of the mean of the function, μ, then the logit model relates the logit of *p* to a linear set of predictors,

$$\eta = \text{Ln}\left(\frac{p}{1-p}\right) = \beta_0 + \sum_1^K \beta_K X_K + \varepsilon \quad (18.9)$$

This link function does three things that are useful. First, it relates the probability of a binomial outcome to a set of linear predictors. Second, taking the exponent of the logit relates the odds ratio to a set of linear predictors,

$$\left(\frac{p}{1-p}\right) = e^{\beta_0 + \sum_1^K \beta_K X_K + \varepsilon} \quad (18.10)$$

Therefore, the relative probability of obtaining outcome A relative to outcome B can be expressed as an exponential function of a linear set of predictors. This means that one can relate the odds ratio to a set of predictors that can account for the likelihood of A relative to that of B. Comparisons can then be made and linked to other variable. For example, suppose we categorize weapon use by robbers into two categories: 1) gun, knife or other weapon, and 2) using bodily force or threat. Then, the probability of using a physical weapon relative to bodily force can be expressed as a function of one or more independent variables.

Third, by taking the log of the odds ratio, the dependent variable is now a continuous variable that varies from minus infinity to plus infinity (though in practice between -3 and +3). In other words, the logit also eliminates the range restriction of a dependent binomial variable since the logit can have any value between minus and plus infinity.

Figure 18.1 shows the effect of transforming a probability into a logit. Notice how the function is fairly flat from about 0.2 to 0.8 beyond which the logit accelerates. When we reverse the axes and plot the effect of a logit on the probability, we have the classic S-shaped curve (Figure 18.2). The effect of a change in the logit on the probability is most pronounced in the middle of the probability range whereas there is less change at the low and high ends of the logit. In other words, the effect of the logit is to linearize the probability within the middle range of probability in order to allow a regression model to be tested.

Logistic Form

Equation 18.9 expresses the log of the odds as a function of linear predictors. Manipulating equation 18.9 leads to a solution for *p*,

$$P(Y=1) = \frac{e^{\beta_0 + \sum_1^K \beta_K X_K}}{1 + e^{\beta_0 + \sum_1^K \beta_K X_K}} = \frac{1}{1 + e^{-(\beta_0 + \sum_1^K \beta_I X_K)}} \quad (18.11)$$

Figure 18.1:

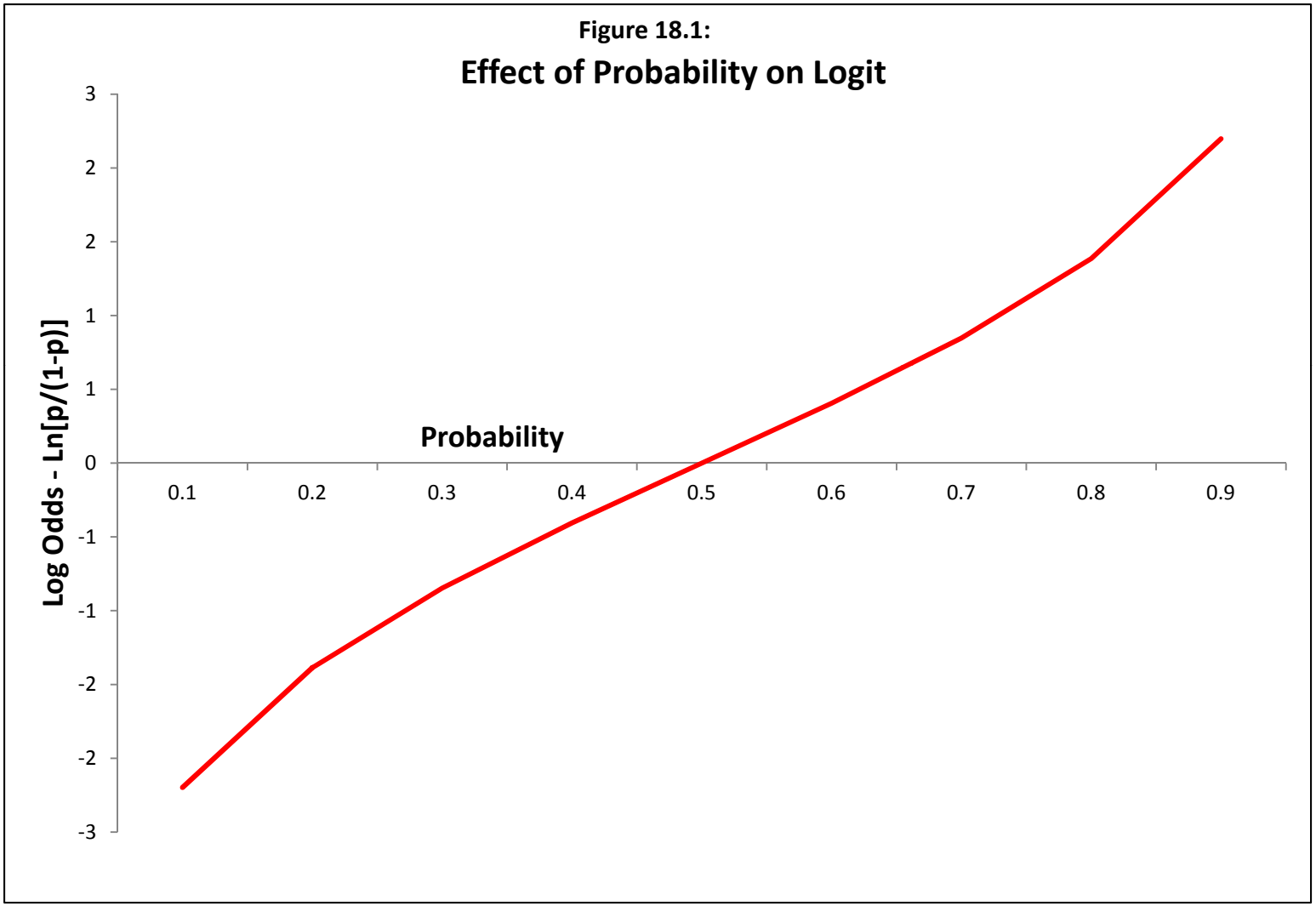
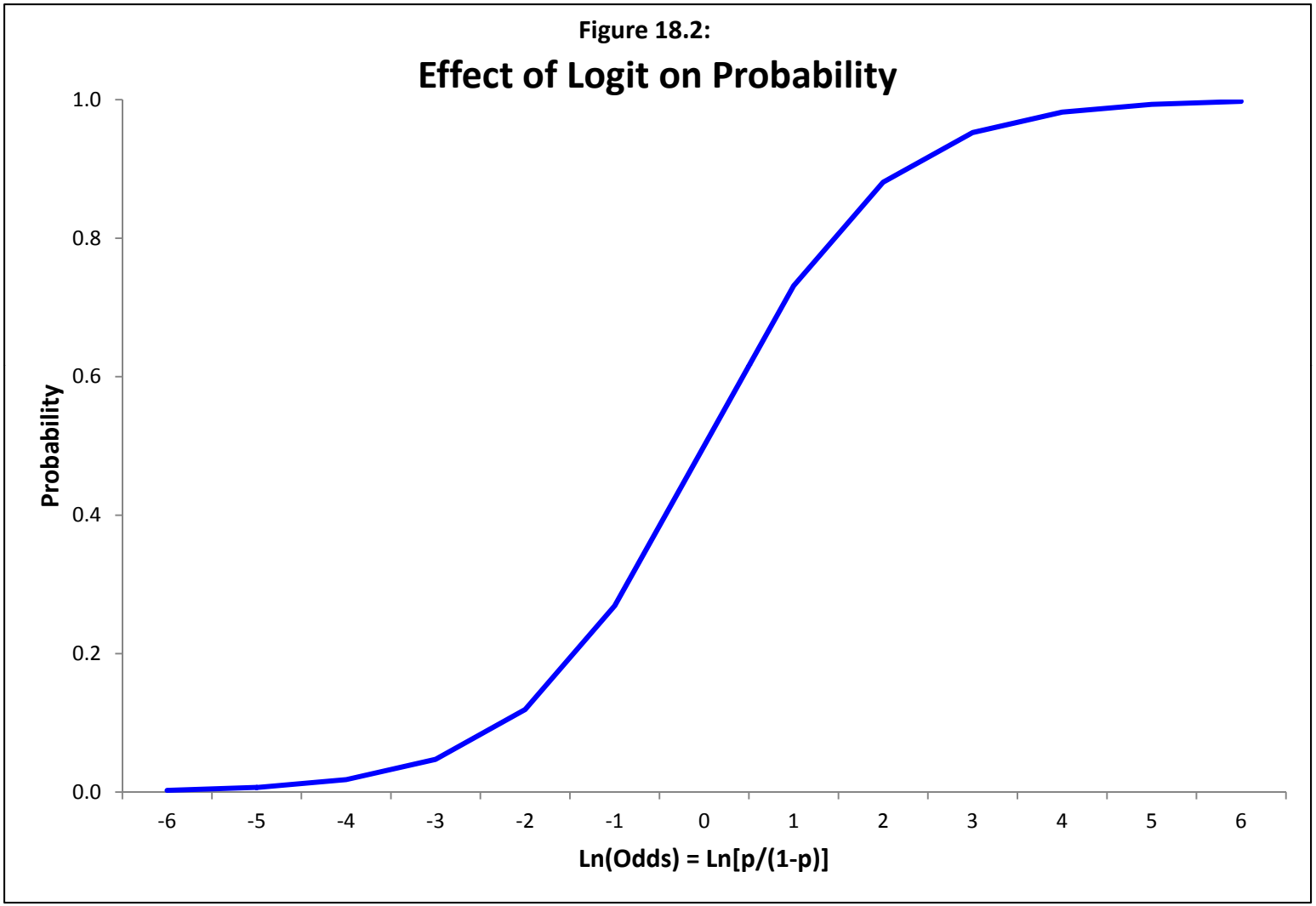

Effect of Probability on Logit
Log Odds - Ln[p/(1-p)]
Probability
3
2
2
1
1
0
-1
-1
-2
-2
-3
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

Figure 18.2:


Effect of Logit on Probability
1.0
0.8
0.6
0.4
0.2
0.0
Probability
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
Ln(Odds) = Ln[p/(1-p)]

which is a true logistic (S-shaped) function. Some references refer to equation 18.9 as a logit and 18.11 as a logistic. However, they are equivalent functions (Liao, 1994). The probability of a 0 is simply 1 minus the probability of a 1, or

$$P(Y=0) = \frac{1}{1+e^{\beta_0+\sum_1^K \beta_K X_K}} \tag{18.12}$$

As an example, figure 18.3 illustrates the probability that is obtained from a logit model that is estimated by

$$Ln\left[\frac{p}{1-p}\right] = -10 + X$$

where X is a simple variable that varies from 0 to 20. Note the coefficient for X is 1.0. At the low end, the effect of increasing X is minimal in effecting the probability. In the middle, the effect of X is the greatest while at the high end, again, the effect of increasing X on the probability is minimal. This is the nature of a probability function since it is bounded by 0 and 1. The logit simply allows the probability to be regressed against one or more independent variables.

The model is inherently non-linear and must be solved by an iterative method. For the normal logit function, maximum likelihood estimation (MLE) is used. For more complex logit functions, Markov Chain Monte Carlo (MCMC) methods can be used.

Interpretation of the Logit Model

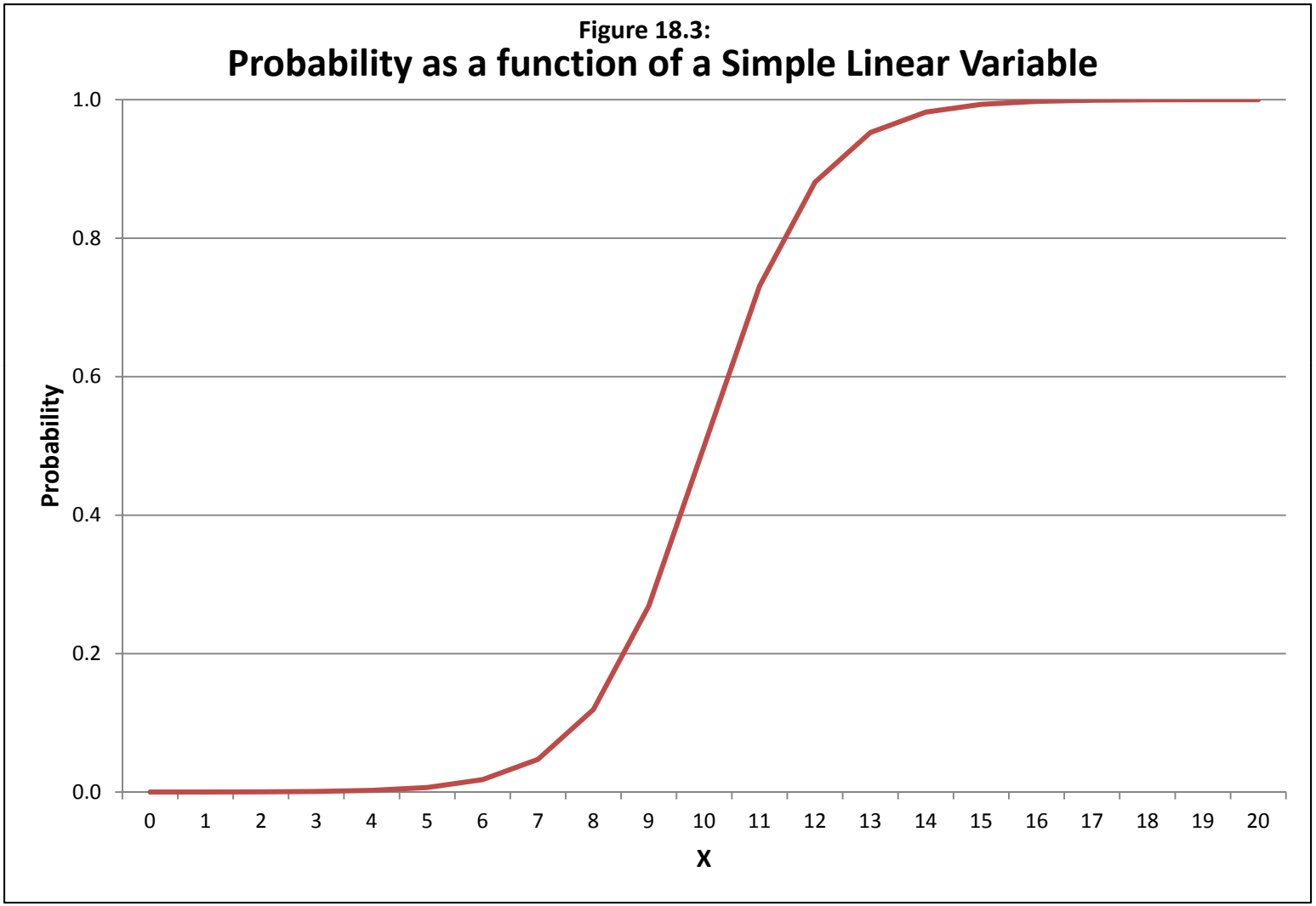
Sign of the Coefficient

Examples will be provided shortly but, there are several ways to interpret the logit model in equation 18.9 (Pampel, 2000). First, there is the sign of the coefficient. As in most regression models, a positive sign indicates that the independent variable increases the probability of the choice being made while a negative sign indicates that the independent variable decreases the probability of the choice being made. Whether we interpret the results in terms of the log of the odds ratio, the odds ratio itself, or the probability, the sign indicates the directional effect of the variable.

Log of the Odds Ratio

Second, there is the log of the odds ratio. Since the model is estimated as a log of the odds function, the interpretation of the coefficients is similar to other regression models, namely the coefficient of each independent variable expresses the change in the dependent variable from

Figure 18.3:


Probability as a function of a Simple Linear Variable
Probability
1.0
0.8
0.6
0.4
0.2
0.0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
X

a one unit change in that variable. However, since the dependent variable is a log of the odds, the coefficients do not have any intuitive meaning in this form other than indicating the sign of the relationship (increasing or decreasing) and the relative strength of the variable as indicated by a Z-test (coefficient divided by standard error).

The use of logged odds for interpretation does have the advantage of symmetry. For example, if the odds of obtaining one alternative (e.g., the odds that a robber will carry some type of weapon) is 9:1 (i.e., the probability of the alternative is 0.9 while the probability of other alternative is 0.1), then the log of the odds for the alternative is 2.1972. For the other alternative, the log of the odds is -2.1972. In other words, the log of the odds of the selected alternative (which takes the value 1) is the opposite of the log of the odds of the non-selected alternative (which takes the value 0).

Odds ratio

Third, a more intuitive way to interpret the logit model is through the odds ratio itself. Equation 18.10 above shows the odds as a function of the exponentiated linear equation. Since the exponent of a sum is equal to the product of the exponent of the parts, we have

$$\left(\frac{p}{1-p}\right) = e^{\beta_0 + \sum_1^K \beta_K X_K} = e^{\beta_0} e^{\beta_1 X_1} e^{\beta_2 X_2} \dots \dots e^{\beta_K X_K + \epsilon} \quad (18.13)$$

The odds ratio can be expressed as the product of the exponentiated coefficients times their variable values and including the error term, ε. In this case, the effect of a unit change in each independent variable on the odds ratio is the exponentiated coefficient. For example, if a coefficient was -0.2, then the effect of a one unit change in that variable on the odds ratio will be $e^{-0.2} = 0.8187$ (or a decreasing effect). Similarly, if a coefficient was 1.1, then the effect of a one unit change in that variable on the odds ratio will be $e^{1.1} = 3.0042$ (or an increasing effect). As mentioned above, the odds ratio has an intuitive meaning in that it indicates the relative likelihood of alternative A versus alternative B.

The percentage change for a one unit increment in the independent variable can be determined by (Pampel, 2000):

$$Percent\ change_K = \left(e^{\beta_K} - 1\right) * 100 \quad (18.14)$$

where β_K is the coefficient of an independent variable in the logit function in equation 18.9 while e^{β_K} is the odds ratio of the variable. To use the example above, if the coefficients was -0.2, then the percentage change from a one unit increase in that variable is -18.1% ($[e^{-.2}-1]*100 = [0.819-1)*100$).

Probability

Fourth, one can express the logit model through a probability itself, essentially solving equation 18.11. The result is a probability function. Unfortunately, the effect of a coefficient on the probability is non-linear and not constant and changes according to the level of the probability. That is, when the probability, *p*, is very low (e.g., 0.1), the effect of an independent variable is also very weak. Similarly, when *p* is very high (e.g., 0.9), the effect of an independent variable is similarly weak. The effects of an independent variable on the probability are strongest when the probability is in the middle range and the absolute strongest when the probability is exactly 0.5.

Variance

Fifth, an important component of a logit model is the variance. With logit models, as with Poisson models, the variance is a function of the mean. That is, the probability, *p*, is the expected value of the distribution:

$$E(Y) = p \tag{18.15}$$

where Y is a binary variable. The variance of a probability is, itself, a function of the mean:

$$Var(Y) = p(1-p) \tag{18.16}$$

This is the similar to the Poisson-based models where the variance of the Poisson is a function of mean and is always underdispersion (variance less than the mean).[1] With ungrouped data, it is not possible for the actual variance to exceed the predicted variance since they are measured exactly the same (McCullagh and Nelder, 1989). With grouped data, however, it is possible for the actual variance to exceed the expected variance. However, since the logit routines in CrimeStat only apply to individual records (i.e., there is no grouping), the variance is always that indicated by equation 18.16.

The Error Term

Finally, let us discuss briefly the error term in the model, ε. In the GLM interpretation (equation 18.1), the error, ε, is the difference between the observed and predicted values. With the OLS model discussed in Chapter 15, the errors are assumed to be normally distributed and

[1] Note that in an Ordinary Least Squares (OLS), the variance is estimated independently of the mean. Thus, there is no confounding of effects. This is one advantage of OLS compared to Poisson or binomial models. On the other hand, OLS does not model skewed distributions very well nor can it model a binary variable.

constant (a condition known as homoscasticity). With the Poisson family of models discussed in Chapters 16 and 17, the errors are normal but not constant (heteroscadstic). For the 'true' Poisson model, they are also Poisson but for the negative binomial model, they are Gamma distributed. We also discussed lognormal error terms in Chapter 17. In all cases, though, the errors are normally distributed.

However, for a probability, the error cannot be normally distributed except in the middle range of the probabilities. Take the example shown above in figure 18.3. At the two extremes – the low end and the high end, the error will be much smaller than in the middle range of the probabilities. In fact, the error will be greatest in the middle. But, also, the errors must be asymmetrical at the two extremes. The closer an estimated probability is to either 0 or to 1, the more likely the errors will be skewed and asymmetrical (meaning that they will fall on one side of the estimate rather than the other. This is just a function of the limits of a probability which have to fall between 0 and 1. In the middle range, however, the errors are generally symmetrical and normally distributed.

McFadden (1973) and Train (2009) have shown that the errors for a logit model are distributed *extreme value* distribution (sometimes called Gumbel or type I extreme value (see also Wikipedia, 2011c). It is part of a family that describes extreme distributions called the Generalized Extreme Value distribution (Wikipedia, 2011d). The extreme value distribution models the maximum or minimum at the extremes of a limited dependent variable, such as a probability. Train (2009) points out that the extreme value gives slightly higher proportions at the extremes of a probability than a normal distribution, and also allow for the asymmetry at the extremes. However, in the middle range, the extreme value distribution is virtually indistinguishable from a normal distribution. It is somewhat similar to a Student's t-distribution though the mathematics is different (Wikipedia, 2011e).

Logit Regression

In CrimeStat, there are three different logit models. One of these is estimated through maximum likelihood (MLE) while the other two are estimated through the Markov Chain Monte Carlo (MCMC) simulation methodology. If readers are unfamiliar with the MCMC method, we suggest that they review Chapters 16 and 17 before going forward in this chapter.

Logit Analysis of Weapon Use for 2007-09 Houston Robberies

MLE Logit

In an MLE logit, the logit model shown in equation 18.9 is estimated with a maximum likelihood estimator. As an example, we use data on 3,709 robberies that occurred within the

City of Houston from 2007-2009. Robberies were selected in which both the crime location and the offender's residence location were known. These came from suspect lists and are only 11% of the total robberies committed within the City for those years. They were selected because the suspect list included information on the age, gender and ethnicity of the offender, whether other suspects were involved, as well as the distance from the residence to the crime location. Additional information on the location of the crime was collected.

The dependent variable was whether a physical weapon had been used, either a firearm, a knife, a stick or another physical object, compared to physical force or the threat of force. Figure 18.4 show the distribution of the robberies and the type of weapon or threat used. Of these 3,709 robberies, 2,333 (or 63%) involved a physical weapon. These were coded as '1' (used a physical weapon) or '0' (did not use a physical weapon). The goal was to estimate the characteristics associated with the use of a physical weapon.

Table 18.1 shows the results of a regression model relating the use of a physical weapon to seven independent variables. The model was estimated with the maximum likelihood (MLE) Logit model in CrimeStat. Only variables that were significant at the $p \leq .05$ or smaller and which had very high tolerances were selected for the model (the process of eliminating non-significant and collinear variables is not shown). See Chapters 15 and 17 for a discussion of multicollinearity.

The log likelihood is substantially negative and the AIC and BIC, statistics used to correct the log likelihood for the number of independent variables (see Chapter 16, p. 16.5) are substantially positive, as would be expected. However, given that there are 3,709 records, we would expect the models to be significant.

Therefore, one has to look at other statistics. In terms of the overall probability, the deviance and the Pearson chi-square are both significant, indicating that the model is significantly different from a random model (which would be expected). On the other hand, when these are adjusted for degrees of freedom (adjusted deviance and adjusted Pearson Chi-square), the statistics are not significant. This indicates that fit of the model was. This is supported by the mean absolute deviation and the measured squared predicted error statistics which shows the model fit quite well (a discussion of these statistics are found in Chapter 16). Keep in mind, though, that the dependent variable is binary which means that there are only values of 0 or 1.

All six independent variables are highly significant. The tolerance statistics indicate that they are almost completely independent (note, this is not surprising since we eliminated collinear statistics while building the model). This is an important point that we keep re-iterating.

Figure 18.4:

Houston Robberies: 2007 to 2009

Location of Robberies by Weapon Use

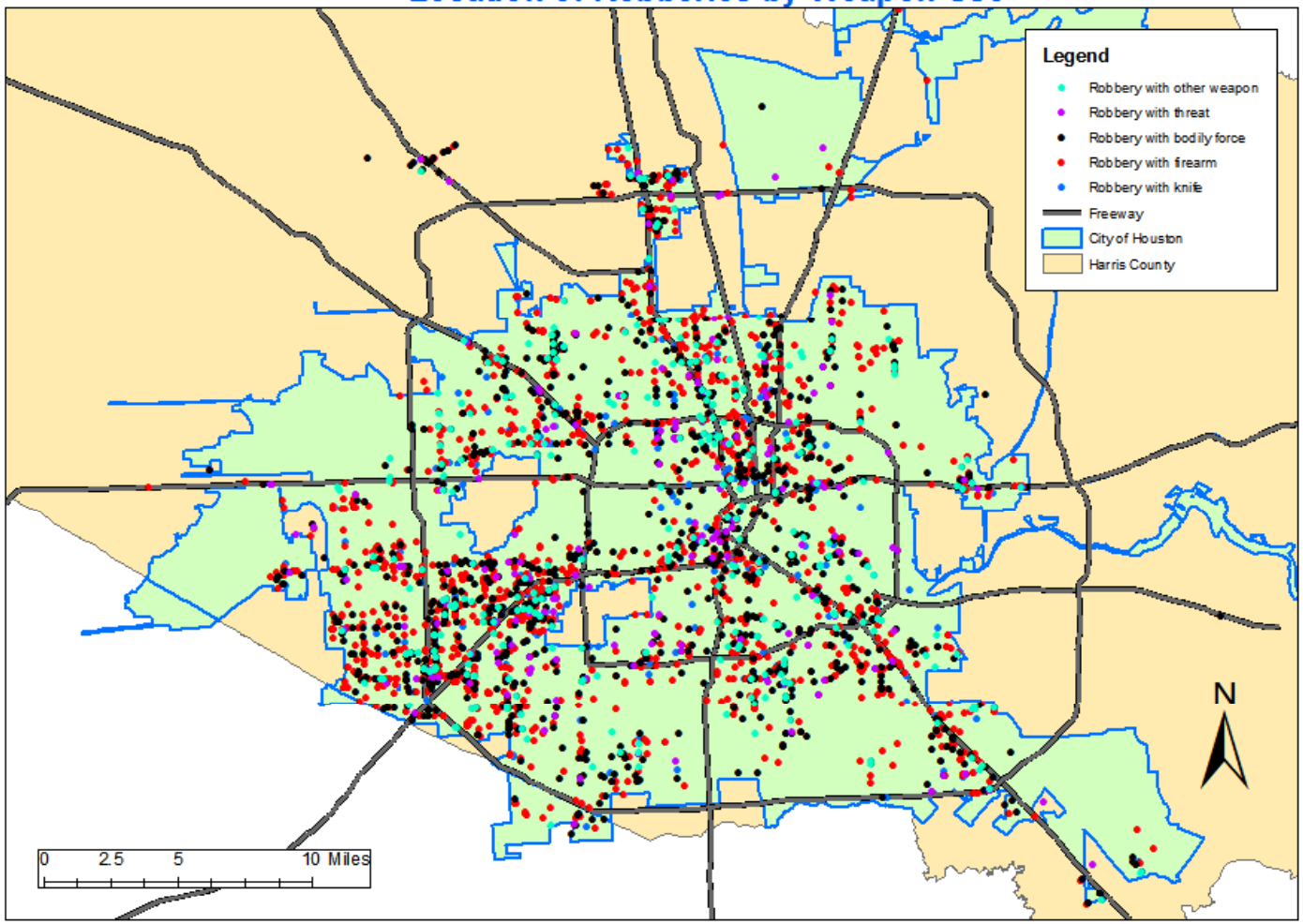


Table 18.1
Weapon Use by 2007-09 Houston Robbers:
MLE Binomial Logit Model

(N=3,709 Robberies with Known Origin & Destination Coordinates)

DepVar:	**WEAPON USE IN ROBBERIES**	
N:	3,709	
Df:	3,696	
Type of regression model:	Logit	
Method of estimation:	Maximum Likelihood	
Likelihood statistics		
Log Likelihood:	-2,345.7	
AIC:	4,707.3	
BIC/SC:	4,757.1	
Deviance:	2,086.1	p: 0.0001
Pearson Chi-Square:	1,373.3	p: 0.0001
Model error estimates		
Mean absolute deviation:	0.4	
1st (highest) quartile:	0.4	
2nd quartile:	0.4	
3rd quartile:	0.5	
4th (lowest) quartile:	0.6	
Mean squared predicted error:	0.2	
1st (highest) quartile:	0.1	
2nd quartile:	0.1	
3rd quartile:	0.3	
4th (lowest) quartile:	0.4	
Dispersion tests		
Adjusted deviance:	0.6	p: n.s.
Adjusted Pearson Chi-Square:	0.4	p: n.s.

Predictor	DF	Coefficient	Stand Error	Tolerance	Z-value	p	Odds ratio
INTERCEPT	1	0.7005	0.147	-	4.76	0.001	2.015
AGE	1	-0.0197	0.003	0.965	-5.67	0.001	0.981
GENDER	1	-0.6059	0.110	0.992	-5.53	0.001	0.546
# SUSPECTS	1	0.2981	0.043	0.979	6.89	0.001	1.347
NIGHT	1	0.5225	0.092	0.985	5.68	0.001	1.686
MEDIAN HOUSEHOLD INCOME	1	-0.000008	0.000002	0.981	-3.47	0.001	1.000
DISTANCE TO DOWNTOWN	1	0.0316	0.007	0.966	4.56	0.001	1.032

Typically, both an MLE and an MCMC model will converge more quickly and will produce cleaner estimates if the independent variables are truly independent.

Examining the effects of the individual variables, younger offenders and those who are male are more likely to use a physical weapon. Looking at the odds ratio of -0.0197 means that for each year of age for a robber, the likelihood of using a physical weapon decreases by about 2% ($[e^{0.0197}-1]*100$). Female robbers (those whose gender value is 1 in the model) are 45% less likely than males to use a physical weapon ($[e^{-0.6059} - 1]*100$).

On the other hand, the more suspects/co-offenders involved in the robbery, the more likely there will be a use of a physical weapon. With an odds ratio of 1.347, each additional co-offender increases the likelihood of using a physical weapon by 35% ($[e^{1.347}-1]*100$) compared to a robbery with only a single offender. Similarly, robberies committed at night time (Midnight to 6 am) are 69% more likely to involve a physical weapon ($[e^{1.686}-1]*100$).

The environmental variables suggest a small effect for income (decreasing) and a small effect for distance (increasing). Why robberies committed farther from downtown involve a greater likelihood of having a physical weapon involved is not clear. For the other variables, the effects are what we would expect.

Note that the odds ratio gives the relative likelihood of the independent variable on the dependent variable. For categorical independent variables, such as GENDER or NIGHT, the comparison is between the group with the value 1 (females and night time respectively) compared to the group with the value 0 (males and other time periods respectively). For continuous independent variables, such as AGE and #SUSPECTS, the odds ratio indicates the incremental effect of a one unit change in that variable.

MCMC Logit

CrimeStat includes both maximum likelihood and MCMC versions of the logit. For comparison, we ran the same model as in Table 18.1 using the MCMC algorithm. There were 25,000 iterations with 5,000 of these being discarded ('burn in'). Hence, the final results were from the 20,000 iterations beyond the 'burn in' sample. Table 18.2 shows the results.

The log likelihood value is stronger (more negative) than for the MLE logit while the AIC and BIC statistics are more positive. The deviance and Pearson chi-square statistics are very similar to the MLE logit and indicate that the model was significantly different than one fit by chance. The MCMC error relative to the standard deviation values are all below 0.05 and the G-R statistics are well below 1.2 (see Chapter 17 for explanation of these indices).

Table 18.2

Weapon Use by 2007-09 Houston Robbers: MCMC Binomial Logit Model

(N=3,709 Robberies with Known Origin & Destination Coordinates)

DepVar:	**WEAPON USE IN ROBBERY**		
N:	3,709		
Df:	3,701		
Type of regression model:	Logit		
Method of estimation:	MCMC		
Number of iterations:	25,000	Burn in: 5,000	
Likelihood statistics			
Log Likelihood:	-2,348.1		
AIC:	4,712.3		
BIC/SC:	4,762.0		
Deviance:	-587.3	p: 0.0001	
Pearson Chi-square:	1,373.6	p: 0.0001	
Model error estimates			
Mean absolute deviation:	0.4		
1st (highest) quartile:	0.3		
2nd quartile:	0.4		
3rd quartile:	0.5		
4th (lowest) quartile:	0.6		
Mean squared predicted error:	0.2		
1st (highest) quartile:	0.1		
2nd quartile:	0.1		
3rd quartile:	0.3		
4th (lowest) quartile:	0.4		
Dispersion tests			
Adjusted deviance:	-0.2	p: n.s.	
Adjusted Pearson Chi-Square:	0.4	p: n.s.	

Predictor	Mean	Std	t-value[p]	.	MC error	MC error/ std	G-R stat	Odds ratio
INTERCEPT	0.6923	0.150	4.60***		0.005	0.035	1.008	1.998
AGE	-0.0197	0.003	-5.65***		0.0001	0.030	1.004	0.981
GENDER	-0.6070	0.110	-5.50***		0.001	0.008	1.000	0.545
# SUSPECTS	0.3005	0.044	6.81***		0.001	0.022	1.003	1.350
NIGHT	0.5249	0.091	5.74***		0.001	0.009	1.000	1.690
MEDIAN HOUSEHOLD INCOME	-0.000008	0.0000	-3.29**		0.0000006	0.024	1.004	1.000
DISTANCE TO DOWNTOWN	0.0318	0.007	4.58***		0.0001	0.011	1.000	1.032

*** p≤.001

** p≤.01

Note, also, that the deviance statistic is negative in Table 18.2. This is because the posterior distribution of the dependent variable (weapon use in robberies) is not normal since it is constrained by the binomial variable to be between 0 and 1 and has a small standard deviation (Spiegelhalter, 2006). Thus, with an MCMC logit model, one might expect a negative deviance. This was not true with the MLE logit model in Table 18.1, however. In either case, the adjusted deviance is not significant, suggesting that the dispersion has been adequately accounted.

The coefficient estimates are almost identical. They differ only in the third decimal place for several values. Similarly, the standard error estimates are also quite similar up through the second decimal place. Finally, the odds ratios are almost identical for the two estimates, up through the second decimal place.

Note that there is no dispersion measure in the logit model. The reason is that the standard deviation of a binomial variable is always:

$$SD_{binomial} = \sqrt{(p)(1-p)} \tag{18.20}$$

In short, the MCMC logit has replicated the MLE logit model for Houston robbery weapon use. So, why run an MCMC model when an MLE will produce almost identical results in a fraction of the time? The reason has to do with running more complex models than a simple logit, particularly a binomial logit with an estimate of spatial autocorrelation. Chapter 19 will discuss that issue.

MCMC Logit-CAR/SAR

The final logit model is a spatial model. This will be discussed in Chapter 19.

Probit Model

MLE Probit

The logit is the most commonly used way to model a binary variable. But, there are other functions that can also linearize a binary dependent variable. One commonly used one is the probit function for which the link function was defined in equation 18.5. The probit expresses the inverse of the cumulative standard normal distribution as a linear function of independent variables (without an error term):

$$p(Y=1) = \Phi^{-1}(p_i) = \beta_0 + \sum_1^K \beta_K X_K \tag{18.21}$$

where Φ is the cumulative standard normal distribution,

$$\Phi(\mathbf{x}_i^T\boldsymbol{\beta}) = \int_{-\infty}^{\mathbf{x}_i^T\boldsymbol{\beta}} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}\, dx \tag{18.22}$$

The inverse of the cumulative standard normal distribution is a Z-score and, essentially, the probit is a cumulative Z-score for a one-tailed probability:

$$p(Y = 1) = \Phi\{\beta_0 + \sum_1^K \beta_K X_K\} \tag{18.23}$$

The area under the standard normal distribution is 1.0. Starting at minus infinity, the area under the curve can be expressed as a probability and the link function, η, is a linear regression of the Z score of the event probability (Liao, 1994). The probability of a non-event is 1 minus the probability, or

$$p(Y = 0) = 1 - \Phi\{\beta_0 + \sum_1^K \beta_K X_K\} \tag{18.24}$$

Interpreting the coefficients is not intuitive because it involves additive effects of the intercept and independent variables on the inverse of the cumulative standard normal distribution. Also, unlike the logit function, there is not an odds ratio. Nevertheless, the signs of the coefficients are in the same direction as for the logit model and the Z-values produced by coefficients divided by their standard errors are usually of the same magnitude.

To see this, we model weapon use among the Houston robbers (Table 18.3). Comparing this table with MLE logit model (Table 18.1), the likelihood statistics are virtually the same; the signs of the coefficients are identical and the Z-scores of the coefficients are of the same magnitude. The values of the coefficients are, of course, very different since they express the dependent binary variable in different units. The model is estimated in CrimeStat with maximum likelihood. At this point, there are no MCMC probit models though we may add them in later versions.

Utility of the Probit Model

With most datasets, the logit and probit models will produce almost identical conclusions. They differ primarily in the tails of the distribution with the probit approaching the limiting ends of the probability more quickly than the logit.

Table 18.3:
Weapon Use by 2007-09 Houston Robbers:
MLE Probit Model

(N=3,709 Robberies with Known Origin & Destination Coordinates)

DepVar:	**WEAPON USE IN ROBBERY**	
N:	3,709	
Df:	3,696	
Type of regression model:	Probit	
Method of estimation:	Maximum Likelihood	
Likelihood statistics		
Log Likelihood:	-2,347.4	
AIC:	4,710.9	
BIC/SC:	4,760.6	
Deviance:	4,479.6	p: 0.0001
Pearson Chi-Square:	2,472.9	p: 0.0001
Model error estimates		
Mean absolute deviation:	0.9	
1st (highest) quartile:	0.6	
2nd quartile:	0.6	
3rd quartile:	0.9	
4th (lowest) quartile:	1.4	
Mean squared predicted error:	1.2	
1st (highest) quartile:	0.6	
2nd quartile:	0.6	
3rd quartile:	1.2	
4th (lowest) quartile:	2.2	
Dispersion tests		
Adjusted deviance:	1.2	p: n.s.
Adjusted Pearson Chi-Square:	0.7	p: n.s.

Predictor	DF	Coefficient	Stand Error	Tolerance	Z-value	p
INTERCEPT	1	0.4550	0.089	-	5.10	0.001
AGE	1	-0.0121	0.002	0.965	-5.68	0.001
GENDER	1	-0.3706	0.068	0.992	-5.47	0.001
# SUSPECTS	1	0.1656	0.024	0.979	6.89	0.001
NIGHT	1	0.3181	0.055	0.985	5.78	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.000005	0.000001	0.981	-3.38	0.001
DISTANCE TO DOWNTOWN	1	0.0191	0.004	0.966	4.63	0.001

Using the example discussed in chapters 15, 16 and 17, we model 2006 Houston burglaries in 1,179 traffic analysis zones (TAZ). But, instead of modeling the number of burglaries per TAZ, we created a binomial variable for one or more burglaries. The dependent variable was whether the TAZ had one or more burglaries in 2006 and the two independent variables were the number of households in 2006 and the 2000 median household income. Table 18.4 shows the result of the probit model while table 18.5 shows the result of the logit model.

There are some subtle differences. The logit model has a higher log likelihood value (i.e., less negative) and lower AIC and BIC values, suggesting that it is a better probability model. The model error statistics (mean absolute deviation and mean squared predicted error) are similar though the logit does a better job in fitting the fourth (lowest) quartile.

The coefficients, however, are a little different. The intercept for the logit is significant while that of the probit is not. The coefficient for median household income is almost significant in the logit model ($p \leq 0.1$) while it not significant in the probit model. Whether these differences are meaningful would depend on what the researcher is willing to assume. As mentioned, the probit assumes an underlying normal distribution while the logit does not. If the transition from a measured null response (0) to a counted response (1) is assumed to be gradual, then the probit may make more theoretical sense.

Figure 18.5 graphs the results of the two models. As seen, the probit model levels off more quickly than the logit model. That is, at the low end, it approaches both the low and high asymptote more quickly than the logit. The probit shows a more gradual change than the logit, which could be a more realistic representation of the shift in probabilities from the null condition to the prevalence of the phenomenon.

Nevertheless, the two models are very highly correlated. Hahn and Soyer (2005) make the point that the two models will be different if the values at the ends are of interest. For most other tests, however, the estimated probabilities will be very similar.

Conclusion

We have examined two different models for estimating the effects of independent variables on a binary dependent variable, the logit and the probit. The logit is clearly more convenient to use given that the exponentiated coefficients can be expressed in terms of the odds ratio. That is the main reason that it more widely used. In Chapter 19, we will show how an MCMC version of the logit can be adapted to estimate spatial autocorrelation in the dependent variable.

Figure 18.5:

Logit and Probit Predictions of Houston Burglaries: 2005-2007

1,179 Traffic Analysis Zones with One or More Burglaries

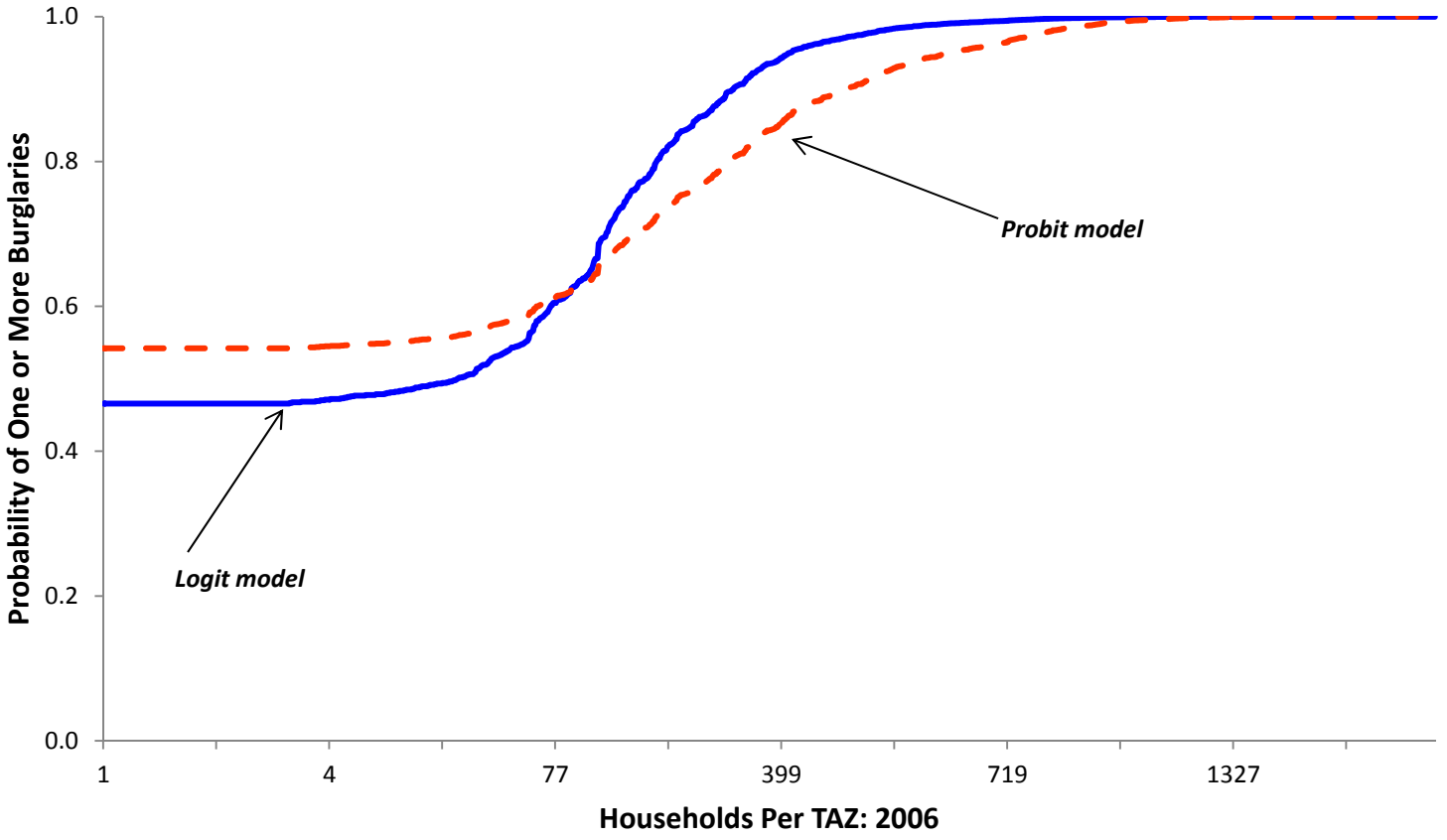


Table 18.4:

Predicting Burglaries in the City of Houston: 2006

MLE Probit Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**ONE OR MORE BURGLARIES**	
N:	1,179	
Df:	1,175	
Type of regression model:	Probit	
Method of estimation:	Maximum Likelihood	
Likelihood statistics		
Log Likelihood:	-427.5	
AIC:	863.0	
BIC/SC:	883.3	
Deviance:	347.4	p: 0.0001
Pearson Chi-Square:	220.4	p: 0.0001
Model error estimates		
Mean absolute deviation:	0.2	
1st (highest) quartile:	0.1	
2nd quartile:	0.2	
3rd quartile:	0.1	
4th (lowest) quartile:	1.5	
Mean squared predicted error:	0.1	
1st (highest) quartile:	0.0	
2nd quartile:	0.1	
3rd quartile:	0.0	
4th (lowest) quartile:	0.3	
Dispersion tests		
Adjusted deviance:	0.3	p: n.s.
Adjusted Pearson Chi-Square:	0.2	p: n.s.

Predictor	DF	Coefficient	Stand Error	Tolerance	t-value	p
INTERCEPT	1	0.0252	0.083	-	0.03	n.s.
HOUSEHOLDS	1	0.0023	0.0002	0.994	14.34	0.001
MEDIAN HOUSEHOLD INCOME	1	0.000002	0.000002	0.994	1.28	n.s.

Table 18.5:

Predicting Burglaries in the City of Houston: 2006

MLE Logit Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**ONE OR MORE BURGLARIES**	
N:	1,179	
Df:	1,175	
Type of regression model:	Probit	
Method of estimation:	Maximum Likelihood	
Likelihood statistics		
Log Likelihood:	-389.8	
AIC:	787.7	
BIC/SC:	807.9	
Deviance:	325.4	p: 0.0001
Pearson Chi-Square:	222.6	p: 0.0001
Model error estimates		
Mean absolute deviation:	0.2	
1st (highest) quartile:	0.1	
2nd quartile:	0.2	
3rd quartile:	0.1	
4th (lowest) quartile:	0.4	
Mean squared predicted error:	0.1	
1st (highest) quartile:	0.0	
2nd quartile:	0.1	
3rd quartile:	0.0	
4th (lowest) quartile:	0.2	
Dispersion tests		
Adjusted deviance:	0.3	p: n.s.
Adjusted Pearson Chi-Square:	0.2	p: n.s.

Predictor	DF	Coefficient	Stand Error	Tolerance	t-value	p
INTERCEPT	1	-0.3591	0.151	-	-2.38	0.05
HOUSEHOLDS	1	0.0073	0.001	0.994	10.31	0.001
MEDIAN HOUSEHOLD INCOME	1	0.000006	0.000003	0.994	1.88	n.s.

On the other hand, the probit model has applicability in *random utility* theory which will be discussed in Chapter 21. Train (2009) argues that the probit model can allow for variations in the 'tastes' of decision makers whereas the logit model imposes greater restrictions on the interpretation of coefficients. It can be used to estimate non-constant error variance (heteroscedastic probit models; see Train, 2009) while the logit cannot. But, in general, there really is not much of a difference in their conclusions when applied to the same data.

The final point is that a binary variable, whether measured by the logit or the probit model, is the simplest form of modeling a choice made by a decision-maker. Hence, the logit form (and to a lesser extent, the probit) has widespread applicability in decision theory and is the basis of discrete choice modeling (Train, 2009; McFadden, 1973). Chapter 21 will discuss this.

References

Chen, G. & Tsurumi, H. (2011). Probit and logit model selection. *Communications in Statistics – Theory and Methods*, 40, 159-175.

Greene, W. H. (2008). *Econometric Analysis* (sixth edition). Pearson Prentice Hall: Upper Saddle River, NJ.

Hahn, E. D. & Soyer, R. (2005). Probit and logit models: Differences in the multivariate realm. Unpublished paper. http://home.gwu.edu/~soyer/mv1h.pdf.

Hosmer, D. W. & Lemeshow, S. (2001). *Applied Logistic Regression: Textbook and Solutions Manual*. Wiley-Interscience, J. Wiley & Sons: New York.

Lambert, D. & Roeder, K. (1995). Overdispersion diagnostics for generalized linear models. *J. Amer. Stat. Assoc.*, 90, 1225-36.

Liao, T. F. (1994). *Interpreting Probability Models: Logit, Probit, and Other Generalized Linear Models*. Sage University Paper 101, Sage Publications, Inc: Thousand Oaks, CA.

Lord, D., Washington, S. P., & Ivan, J. N. (2005). Poisson, Poisson-Gamma and zero-inflated regression models of motor vehicle crashes: Balancing statistical fit and theory. *Accident Analysis & Prevention*, Vol. 37 (1), 35-46

McCullagh, P. & Nelder, J. A. (1989). *Generalized Linear Models* (2nd ed). Chapman & Hall: London.

McFadden, D. (1973). Conditional Logit Analysis of Qualitative Choice Behavior, in Zarembka, P. (ed.), *Frontiers in Econometrics*, Academic: New York.

Pampel, F. C. (2000). *Logistic Regression: A Primer*. Sage University Paper 132, Sage Publications, Inc.: Thousand Oaks, CA.

Spiegelhalter, D. (2006). Some DIC slides. http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/dicpage.shtml

Train, K. (2009). *Discrete Choice Methods with Simulation* (2nd edition). Cambridge University Press: Cambridge.

References (continued)

Wikipedia (2011a). Binomial probability. *Wikipedia*, http://en.wikipedia.org/wiki/Binomial_probability.

Wikipedia (2011b). Jacob Bernoulli. *Wikipedia*. http://en.wikipedia.org/wiki/Jacob_Bernoulli.

Wikipedia (2011c). Gumbel distribution. *Wikipedia*, http://en.wikipedia.org/wiki/Gumbel_distribution.

Wikipedia (2011d). Generalized extreme value distribution. *Wikipedia*, http://en.wikipedia.org/wiki/Generalized_extreme_value_distribution.

Wikipedia (2011e). Student's t-distribution. *Wikipedia*. http://en.wikipedia.org/wiki/Student%27s_t-distribution.

Wikipedia (2011f). Overdispersion. *Wikipedia*. http://en.wikipedia.org/wiki/Overdispersion

Chapter 19:

Spatial Regression Modeling[1]

Ned Levine
Ned Levine & Associates
Houston, TX

Dominique Lord
Zachry Dept. of Civil Engineering
Texas A & M University
College Station, TX

Byung-Jung Park
Korea Transport Institute
Goyang, South Korea

Srinivas Geedipally
Texas Transportation Institute
Arlington, TX

Haiyan Teng
Houston, TX

Li Sheng
Houston, TX

[1] This chapter was the result of the efforts of several people. Dr. Shaw-pin Miaou of College Station, TX designed the MCMC algorithm for the Poisson-Gamma-CAR model. Dr. Byung-Jung Park modified the algorithm to incorporate Poisson-Gamma-SAR, Poisson-Lognormal-CAR/SAR, and the MCMC binomial-CAR/SAR models. Dr. Srinivas Geeidpally added the MCMC Normal-CAR/SAR models. Dr. Dominique Lord of Texas A & M University provided technical consulting on the dispersion parameters in these models. Dr. Ned Levine developed the block sampling scheme and provided overall project management. Ms. Haiyan Teng and Dr. Li Sheng programmed the routines and added numerous technical improvements to the algorithms. The authors thank Dr. Richard Block for testing the routines.

Table of Contents

Chapter 19:
Spatial Regression Modeling

In this chapter, we examine spatial regression modeling using the the Markov Chain Monte Carlo (MCMC) method. Users should be thoroughly familiar with the materials in Chapters 15, 16, 17 and 18 before attempting to read this chapter. A good background in statistics is necessary to understand the material.

Spatial Regression Modeling

Spatial regression involves adding a spatial component into a regression model. There are two major ways to express this component, either as an explicit spatial variable or as an internally-estimated spatial parameter. There are advantages and disadvantages to each approach and frequently they are included together.

Explicit Spatial Variable

With an explicit spatial variable, a specific spatial relationship is added as an independent variable. Examples of this are the area of the zone, the distance to the central city, the distance to a particular facility, or an average value of the dependent variable for nearby zones.

The justification for including an explicit variable depends on what is being modeled. For instance, spatial statisticians frequently distinguish between *global* and *local* effects. Global effects are those that cover the entire study region whereas local effects affect only a small geographic area. Without distinguishing those two types of effects, ambiguity can be produced in a model.

Area of the zone

One of the most well known spatial variables that should be included in any statistical model is the area of the zone. Typically, zones based on a census will have a size that is proportional to their residential population. Thus, zones in the center of a metropolitan area will typically be very small, perhaps single blocks, while zones in the suburbs will be very large, covering several square miles. Without adjusting for the size of the zone, distortions in estimates can be produced. For example, all other things being equal, more events can occur within a larger zone than for a smaller zone. Modelers will frequently include the area of the zone as a statistical control variable.

Shape of the zone

Related to this is the shape of the zone. If two zones have very different shapes (e.g., one is square while the other is pointed and long and narrow), allocation error (and, hence, modeling error) is liable to be greater in the one that is more irregular, all other things being equal, than in the one that is square. This is the so-called *Modifiable Area Unit Problem* (or MAUP) problem (see Wikipedia, 2012; Hipp, 2007; Wooldridge, 2002; Openshaw, 1984).

There is not a simple statistical variable that can be included to adjust for irregularity, short of some fractal measure (Lam & De Cola, 1993). Ideally, if the zones could be uniform grid cells, then distortions due to shape can be minimized. Otherwise, the user needs to be cognizant of the potential for shape to influence the coefficients of a model and be prepared to modify the data to incorporate irregular boundaries (e.g., smoothing the distribution of events in a hot spot that are assigned to large zones to reduce shape effects; see Chapter 11 on Head-Bang Interpolation).

Distance from downtown

Another well known spatial variable that should be included in a spatial regression model is the distance from the zone to the central area in the study region. For example, with data from a city or metropolitan area, this variable would be the distance (in miles or kilometers) to the downtown area. Typically, the density of events is a function of distance from the central city primarily due to land costs. This effect has been studied as far back as the early 19th century with the work of von Thünen (1826). Alonso (1964) modernized the framework by demonstrating that each activity has its own land price (i.e., the cost of the land underlying the activity) and that a spatial equilibrium will be established in terms of the relative price of different activities.

All other factors being equal, there will be more events occurring in the center of a metropolitan area than in the periphery primarily due to the increased concentration of activities (which is a function of the underlying land costs). Frequently, there will be a relationship between distance from the downtown area and a variable of interest. For example, Levine (2011) showed that the risk of motor vehicle crashes was double in downtown Houston than in the suburbs. This was a function of the concentrated traffic in the downtown area, the greater number of intersections that created potential conflicts between drivers, and the greater number of driveways. Further, male drivers were more likely to be involved in a crash in the downtown area than female drivers so that part of the increased risk was due to a predominance of male drivers.

In another study, Levine and Lee (2013) showed that the distance traveled for crime trips (journey-to-crime) was associated with the distance an offender lived from central Manchester, England with a negative binomial model. Further, different types of crime were associated with specific travel distances, with property crimes being much longer, on average, than violent crime. Further, these distances were mediated by the distance the offender lived from the city center. Around one-fifth of the crimes committed by females were shoplifting and these were much more likely to occur in the city center or in one of the suburban town centers.

To see this, Figure 19.1 shows an estimate of the number of crimes committed in the City of Houston from 2007 to 2009 by distance from downtown Houston in quarter mile intervals. The data were 807,788 reported crime incidents and the estimate was produced by the *CrimeStat* journey-to-crime interpolation routine using a normal kernel and an adaptive bandwidth with a minimum of 200 crimes (see Chapter 13). As seen, the number of crimes per quarter mile increases from about 3,000 in downtown Houston to more than 20,000 at about 11 miles from downtown Houston. The number of crimes then drops rapidly, primarily because the search radius extends beyond the boundaries of the City of Houston.

However, each quarter mile 'band' covers a larger area. Consequently, one would expect, all other things being equal, for there to be more events with distance from downtown. It is essential to normalize this measure to allow equal comparisons. Consequently, we divided the number of crimes per quarter mile band by the area (in square miles) covered by each band.

Figure 19.2 shows the results. As seen, the number of crimes per square mile is greater than 30,000 in downtown Houston but drops very dramatically with distance. The curve is almost a perfect negative exponential function and is frequently modeled by that function (see Chapters 13 and 28). This could be converted into a probability estimate by dividing each density by the total density of all bins. In other words, the probability of a crime being committed in downtown Houston decreases rapidly with distance from downtown.

The point is that one should include explicit spatial variables to account for these potential effects, if only for statistical control. Including a global spatial variable such as the distance from downtown has advantages and disadvantages. It makes explicit how the dependent variable changes by distance, such as in the crash risk study mentioned above (Levine, 2011). As was discussed in Chapter 6, frequently, local spatial autocorrelation is a function of global spatial effects. Typically, the closer to the center of the city a zone is, the more likely that there will be correlations between that zone's value (of any dependent variable) and the values of nearby zones. This is merely a product of increasing concentration in the central city. Without making the global effect explicit can make it appear that the local effect is stronger than it is.

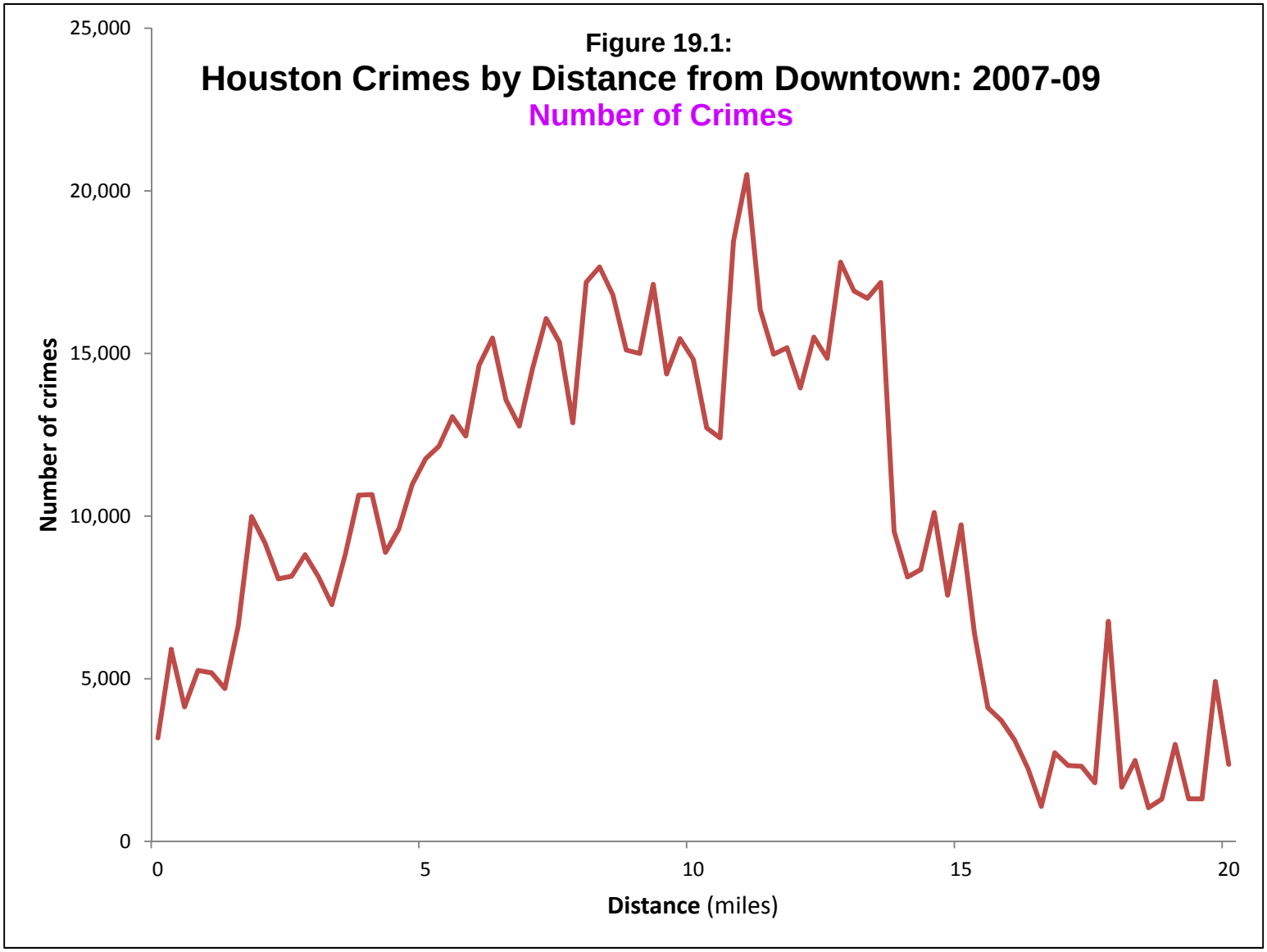
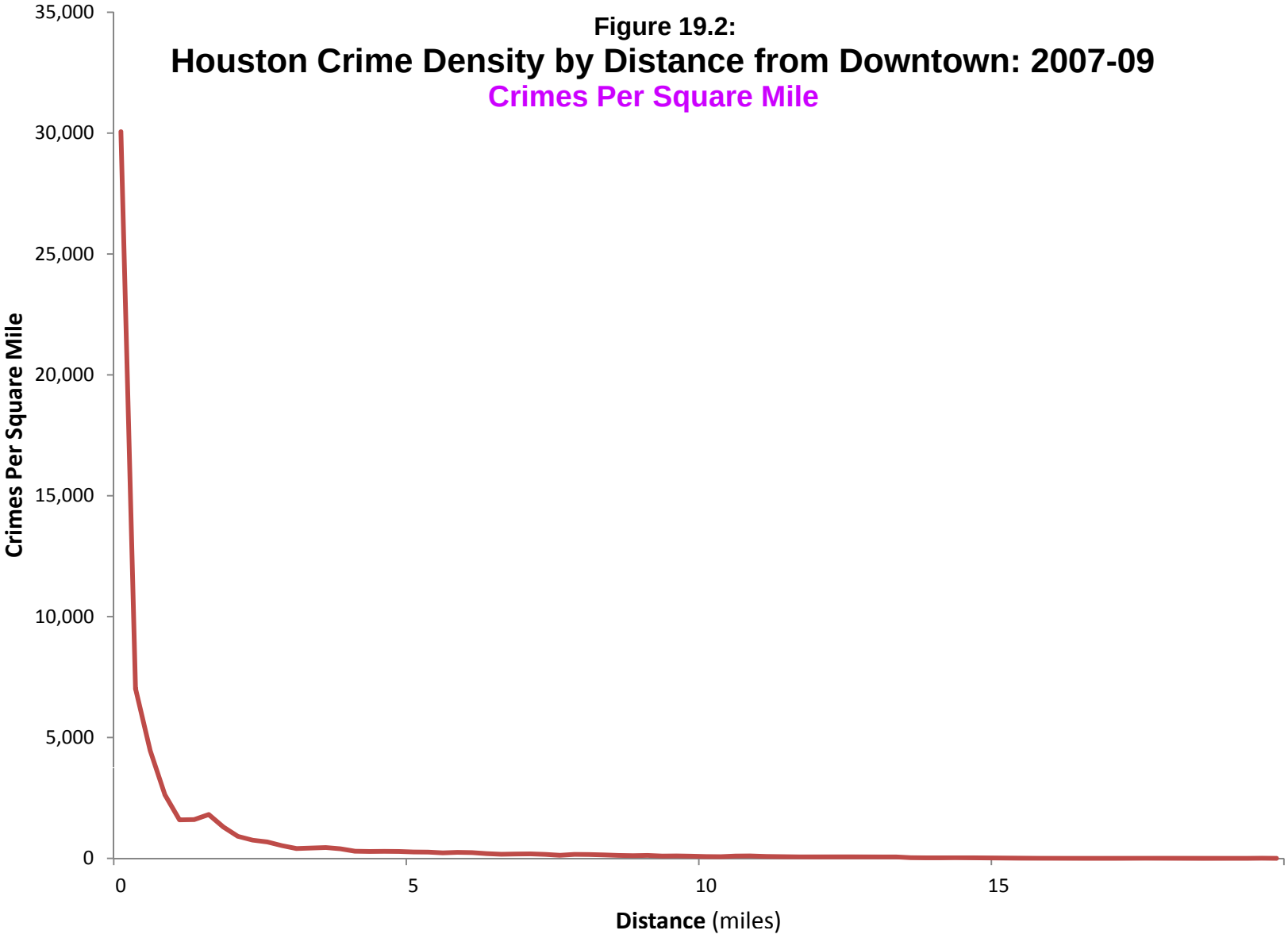


Figure 19.1:
Houston Crimes by Distance from Downtown: 2007-09
Number of Crimes
25,000
20,000
15,000
10,000
5,000
0
Number of crimes
0
5
10
15
20
Distance (miles)

Figure 19.2:

Houston Crime Density by Distance from Downtown: 2007-09

Crimes Per Square Mile



The disadvantage in building in an explicit global 'distance from center' variable assumes that the relationship will hold in future. The zones may be redefined which might distort the relationship slightly (e.g., what the U.S. Census Bureau does from census to census). Also, urban change may distort the distance relationship over time, for example with decreasing crime rates in central cities (Kneebone & Raphael, 2011). Of course, this applies to any spatial variable and not just to distance from the central area.

Also, the inclusion of a global distance variable may cover up a true local effect. For example, Heskin, Levine and Garrett (2000) examined housing and population change using an OLS model at the edges of four California cities that had rent control with vacancy control provisions.[2] By comparing block groups on both sides of the borders, they were able to show the effect of vacancy control over rents was to reduce the number of rental housing units from 1980 to 1990 in the block groups associated with vacancy control compared to the block groups in cities without vacancy control. Had all the block groups in the compared cities been included (which included two very large cities – Los Angeles and Oakland), the relationship would have been obscured.

In short, there are both global spatial effects and local spatial effects, but they need to be distinguished. To not separate out these effects could easily lead to misinterpretation as the relative importance of local spatial clustering (i.e., local spatial autocorrelation). The ideal would be to include both a global distance variable as well as a test for local spatial autocorrelation. The spatial parameter tests discussed below can do this.

Values of nearby zones

More questionable is the inclusion of values for nearby zones. A number of studies have included the values of nearby zones to account for spatial autocorrelation. For example, Wachter and Cho (1991) showed with an OLS model that the restrictiveness of the zoning in adjacent areas independently increased the price of single family homes in Montgomery County, MD. That is, by including the price of single family home in adjacent communities, they showed that it had an effect on the price of single family homes in the community they were studying; they also included a distance to downtown Washington DC as a statistical control variable.

In a recent study of crime trajectories on street segments between 1993 and 2004 in Seattle, Weisburd, Groff and Yang (2012) used a multinomial logit model to predict eight different crime trajectories (see Chapters 21 and 22 on Discrete Choice Analysis). They used

[2] Vacancy control involves maintaining the regulated rent levels even if the unit becomes vacant as opposed to allowing the rent to rise to market levels when a rental unit becomes vacant. At the time, four cities in California had vacancy control provisions.

three year intervals to estimate the effects. Among the 28 variables in their model was a spatial lag variable which was the average number of crimes on neighboring street segments within one-quarter of a mile as well as a variable measuring change in spatial lag between the first three year period (1993-95) and the last three year period (2002-04).

The advantage of including the values of nearby zones, no matter how defined, is that it builds in an effect of the nearby zones. It is somewhat intuitive to treat the nearby value as an exogenous variable to a model in that it produces a coefficient that appears to represent the value of those nearby values.

Problems with using values of nearby zones

There are a number of disadvantages with this approach, however. First, treating the values of nearby zones as an independent variable assumes independence of those zones and ignores reciprocal effects. This can cause s*imultaneity bias* (Wikipedia, 2013). That is, the value of the dependent variable in nearby zones is treated as independent of the value of the same variable in the zone being modeled (the central zone). Yet, in reality, the effect is two ways; the central zone influences the values of the nearby zones, and vice versa (i.e., they are interrelated). The result will be that the coefficient will be biased because some of the estimated effect (the coefficient) of the nearby zones is due to the central zone itself (i.e., the value of the central zone is on both sides of the equation). Specifying the values of nearby zones as being independent does not incorporate the simultaneous effect and will almost certainly produce a biased coefficient of the effect.

Second, treating the values of nearby zones as being independent assumes uniformity of their effect throughout the study area. In reality, spatial autocorrelation varies throughout a study area. For example, clustering of events (hot spots) occurs at only some locations, as was discussed in Chapters 7, 8 and 9. By assuming a uniform effect throughout the study area, the variable adds error to the model and may obscure locations where real clustering effects actually occur. The example given above from Heskin, Levine and Garrett (2000) illustrated the very specific local effect of a policy on housing and population change. In practice, any local spatial autocorrelation that affects the value of a central zone will vary throughout a study region, being strong in some places and weak in others. Using a single variable for the values of nearby zones will not capture that specificity.

Third, the grouping of nearby zones into a single measure uses arbitrary weighting of the zones to be included. Either contiguous (adjacent) zones are used or else a relationship is assumed to operate over a certain distance using a distance decay function (see Chapter 13). The choice of the method for weighting the zones can affect the results substantially. If contiguous zones are used, non-standardized zone size can alter the relationship. For example, in the

downtown area of most cities, the zones will be very small, typically a block or two whereas in the suburbs, zones are much larger. Using contiguous zones may not properly cover the spatial autocorrelation effect. In the central city, the effect might extend well beyond one or two blocks whereas in the suburbs, the effect might be smaller than adjacent zones. If distance is used, researcher must make assumptions about the decay function, the type of function used (e.g., linear, negative exponential) as well as the rate of decay. The internal approach to be discussed below also requires the making of assumptions, a point that will be discussed later in the chapter.

Fourth, adding in a spatial autocorrelation variable does not explain the reason for the spatial autocorrelation but simply accounts for some of the additional variance in a model. That is, the spatial autocorrelation variable accounts for additional variance of the dependent variable after all the independent variables have been accounted for. In other words, that there is additional variability that is spatially organized beyond that accounted for by the included independent variables. Note that this criticism applies to an internal spatial parameter as well.

The important thing to realize is that spatial autocorrelation is merely a statistical index created by spatial effects between nearby zones, either clustering or dispersion. It is not a 'thing' or a 'process' but merely a statistical index. The researcher or analyst would do well to find other variables that could explain some of the variability.

Eliminating bias from values of nearby zones

Eliminating the bias from treating the values of nearby zones as exogenous is complicated. There are two main approaches. First, substitute a truly exogenous variable for the externally-defined spatial autocorrelation variable. This is sometimes called an *instrumental* variable (Wikipedia, 2013a). For example, if the number of crimes is the dependent variable and is correlated with alcohol licenses, substituting the number of alcohol licenses in nearby zones for the spatial autocorrelation variable could capture some of the variance associated with nearby zones without adding bias to the estimate.

Second, one could run simultaneous models (i.e., Y_i predicts Y_j as well as Y_j predicts Y_i where $i \neq j$) iteratively many times until the estimates stabilize. In the models discussed below, we use the Markov Chain Monte Carlo (MCMC) approach to produce stabile estimates.

Internally-estimated Spatial Parameter

An alternative, and more elegant, approach is to utilize a spatial parameter within the model which is estimated within the calculations themselves. The advantage is that the parameter is estimated simultaneously with the coefficients and will include the reciprocal

effects of nearby zones on the central zone, and vice versa. As with a distance-based external variable, the user must make assumptions about the decay of the spatial autocorrelation effect.

There are two common ways to express the internally-estimated spatial parameter, either as a Conditional Autoregressive (CAR) function or as a Simultaneous Autoregressive (SAR) function (De Smith, Goodchild, & Longley, 2007). The CAR function was developed by Besag (1974) while the SAR model was developed by Whittle (1954).

The CAR model is expressed as:

$$E(y_i \mid y_{j \neq i}) = g[\mu_i + \rho \sum_{j \neq i} w_{ij}(y_j - \mu_j)] \tag{19.1}$$

where g is a function relating the expected mean to a linear set of predictors (e.g., Poisson, linear/OLS, logit), μ_i is the expected value for observation i, w_{ij} is a spatial weight between the observation, i, and all other observations, j (and for which all weights sum to 1.0), and ρ is a spatial autocorrelation parameter that determines the size and nature of the spatial neighborhood effect. The summation of the spatial weights times the difference between the observed and predicted values is over all other observations ($i \neq j$).

The SAR model has a simpler form and is expressed as:

$$E(y_i \mid y_{j \neq i}) = g[\mu_i + \rho \sum_{j \neq i} w_{ij} y_j] \tag{19.2}$$

where the terms are as defined above. Note, in the CAR model the spatial weights are applied to the difference between the observed and expected values at all other locations whereas in the SAR model, the weights are applied directly to the observed value. In practice, the CAR and SAR models produce very similar results.

In both these cases, the spatial autocorrelation component is estimated simultaneously with the coefficients. That is, the model assumes that the effects of nearby zones on the central zone are reciprocal, each affecting the other. The use of an internal spatial parameter overcomes one of the main problems of incorporating the values of nearby zones. Instead, the spatial parameter is treated as a function of *hyperparameters*, independent parameters that determine its properties.

MCMC Normal-CAR Model

This is the normal (OLS) model but with a spatial autocorrelation term. For a spatial model, we add a spatial effects parameter, essentially breaking the error term into unexplained

variance that is associated with spatial autocorrelation and unexplained variance that has no known associations (i.e., noise).

$$y_i = \mathrm{x}_i^T \beta + \varepsilon_i + \phi_i \tag{19.3}$$

where , $\boldsymbol{\beta}$ is a vector of unknown coefficients for the *k* covariates plus an intercept, ε_i is the model error independent of all covariates, and ϕ_i is a *spatial random effect*, one for each observation. Together, the spatial effects are distributed as a complex *multivariate normal* (or Gausian) density function.

The Normal-CAR model has two mathematical properties. First, both the error term, ε_i, and the dependent variable are normally-distributed. Second, the model incorporates an estimate of local spatial autocorrelation in a CAR format (equation 19.1).

To model the spatial effect, ϕ_i, we assume the following:

$$p(\phi_i \mid \boldsymbol{\Phi}_{-i}) \propto \exp\left(-\frac{w_{i+}}{2\sigma_\phi^2}\left[\phi_i - \rho \sum_{j \neq i} \frac{w_{ij}}{w_{i+}} \phi_j \right]^2 \right) \tag{19.4}$$

where $p(\phi_i \mid \boldsymbol{\Phi}_{-i})$ is the probability of a spatial effect given a lagged spatial effect, $w_{i+} = \sum_{i \neq j} w_{ij}$ which sums all over j except i (i.e.,all other zones). This formulation gives a conditional normal density with mean $\rho \sum_{j \neq i} \frac{w_{ij}}{w_{i+}} \phi_j$ and variance $\frac{\sigma_i^2}{w_{i+}}$. The parameter ρ determines the direction and overall magnitude of the spatial effects. The term w_{ij} is a spatial weight function between zones i and j (see below). In the algorithm, the term $\sigma_\phi^2 = 1/\tau_\phi$ and the same variance is used for all observations.

The Phi (ϕ_i) variable is, in turn, a function of three hyperparameters. The first - Rho (ρ), might be considered a global component. The second - Tauphi (τ_ϕ), might be considered a local component while the third - Alpha (α), might be considered a neighborhood component since it measures the distance decay. Phi (ϕ_i) is normally distributed and is a function of Rho and Tauphi.

$$\phi_i \mid \mathbf{\Phi}_{-i} \sim N\left(\rho \sum_{j \neq i}^{n} (w_{ij} / w_{i+}) \phi_j , \sigma_\phi^2 / w_{i+} \right) \tag{19.5}$$

Tauphi, in turn, is assumed to follow a Gamma distribution

$$\tau_\phi = \sigma_\phi^{-2} \sim Gamma(a_\phi, b_\phi) \tag{19.6}$$

where a_ϕ and b_ϕ are hyper-parameters. For a non-informative prior $a_\phi = 0.01$ and $b_\phi = 0.01$ are used as a default. Since the error term was assumed to be distributed as a Gamma distribution, it is easy to show that λ_i follows $Gamma(\psi, \psi e^{-\mathbf{x}_i^T \boldsymbol{\beta} - \phi_i})$. The prior distribution for ψ is again assumed to follow a Gamma distribution

$$\psi \sim Gamma(a_\psi, b_\psi) \tag{19.7}$$

where a_ψ and b_ψ are hyper-parameters. For a non-informative prior $a_\phi = 0.01$ and $b_\phi = 0.01$ are used as a default.

MCMC Normal-SAR Model

The Normal-SAR model is very similar to the Normal-CAR. The only difference is in the specification of the spatial autocorrelation term. The SAR (or Simultaneous Autoregressive) term is defined as:

$$\phi_i = \rho \sum_{j}^{n} (c_{ij} / c_{i+}) \phi_j + e_i \tag{19.8}$$

where e_i are iid $N(0, \sigma_\phi^2 / c_{i+})$. All the other variables (c_{ij}, c_{i+}, ρ) are exactly the same as for the CAR model described above. The Phi (ϕ_i) variable is estimated using equation 19.5 above.

Potential Problem in Running MCMC Normal-CAR/SAR Models

Users should be cognizant of a potential problem in using the MCMC Normal model with or without the CAR/SAR spatial autocorrelation parameter. The model is appropriate when the dependent variable is normally distributed and the CrimeStat MCMC routine will work well under these conditions. However, if the dependent variable is highly skewed, the MCMC Normal often will not produce accurate estimates.

We are not completely sure of the conditions that cause the MCMC Normal to not properly produce a good representation of the data. Users should test whether the MCMC Normal (without the spatial autocorrelation parameter) can replicate the results of the MLE Normal. If it can produce a reasonably close approximation, then the MCMC Normal is converging properly and the results of an MCMC Normal-CAR or MCMC Normal-SAR can be trusted. However, if the MCMC Normal does not produce a reasonably close approximation to the MLE Normal, then the algorithm has not converged properly and the user is advised to use one of the Poisson-based models.

A better convergence can often be obtained by first running the MLE Normal and using the estimated intercept and coefficients as prior values in an MCMC Normal. Chapter 20 discusses the specifics of assigning prior values in an MCMC model.

Also, the MCMC Normal is affected by multicollinearity among independent variables. Because multicollinearity creates ambiguity in the coefficients of the collinear variables, it will affect the stability of the MCMC model. This is also true in MCMC Poisson-based models but has much greater effect for MCMC Normal models. Our advice is to eliminate collinear variables in order that those independent variables in the model are truly independent of each other. This will tend to improve MCMC Normal estimates.

> To repeat, the MCMC Normal is appropriate when the dependent variable is normally-distributed. It is not appropriate for highly skewed dependent variables.

MCMC Poisson-Gamma-CAR Model

This is the negative binomial model but with a spatial autocorrelation term. Formally, it is defined as:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \tag{19.9}$$

with the mean of Poisson-Gamma-CAR organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i + \phi_i) \tag{19.10}$$

where exp() is an exponential function, $\boldsymbol{\beta}$ is a vector of unknown coefficients for the *k* covariates plus an intercept, and ε_i is the model error independent of all covariates. The $\xi_i = \exp(\varepsilon_i)$ is assumed to follow the gamma distribution with a mean equal to 1 and a variance equal to $1/\psi$

where ψ is a parameter that is greater than 0, and ϕ_i is a *spatial random effect*, one for each observation.

The assumption on the uncorrelated error term ε_i is the same as in the Poisson-Gamma model. The third term in the expression, ϕ_i, is a *spatial random effect*, one for each observation. Together, the spatial effects are distributed as a complex *multivariate normal* (or Gausian) density function. In other words, the second model is a spatial regression model within a negative binomial model.

The Poisson-Gamma-CAR model has three mathematical properties. First, the count is Poisson distributed, as is true of all Poisson-based models. Second, the mean is distributed as a Gamma function, similar to the negative binomial model. Third, it incorporates an estimate of local spatial autocorrelation in a CAR format (equation 19.1). The same assumptions about the spatial effect apply for the Poisson-Gamma-CAR model as for the Normal-CAR model.

MCMC Poisson-Gamma-SAR Model

The Poisson-Gamma-SAR model is very similar to the Poisson-Gamma-CAR. The only difference is in the specification of the spatial autocorrelation term. The SAR (or Simultaneous Autoregressive) term is defined as:

$$\phi_i = \rho \sum_j^n (c_{ij} / c_{i+}) \phi_j + e_i \qquad (19.11)$$

where e_i are iid $N(0, \sigma_\phi^2 / c_{i+})$. All the other variables (c_{ij}, c_{i+}, ρ) are exactly the same as for the CAR model described above. The Phi (ϕ_i) variable is estimated using equation 19.5 above.

MCMC Poisson-Lognormal-CAR/SAR Model

As described in Chapter 17, the Poisson-Lognormal model has a distribution that is Poisson-distributed.

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \qquad \text{repeat (16.24)}$$

However, the Poisson mean λ_i is organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i + \phi_i) = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \phi_i) \cdot \xi_i \qquad (19.12)$$

where exp() is an exponential function, $\boldsymbol{\beta}$ is a vector of unknown coefficients for the *k* covariates plus an intercept, ϕ_i is the spatial random effect, and ε_i is the model error independent of all covariates. The error, $\xi_i = \exp(\varepsilon_i)$, is assumed to follow the lognormal distribution with a mean equal to 0 and a variance equal to $\sigma_\varepsilon^{-2} = \tau_\varepsilon \sim Gamma \ \ (a_\varepsilon, b_\varepsilon)$.

To model the spatial effect, ϕ_i, equation 19.1 is used for the CAR spatial model while equation 19.2 is used for the SAR spatial model. An application of Poisson-Lognormal-CAR and SAR models is found in Kim and Lim (2010).

MCMC Binomial Logit-CAR/SAR MODELS

In Chapter 18, we discussed binomial models, the logit and the probit. As with Poisson-based model, these can have a spatial autocorrelation component, too. In CrimeStat, we include a spatial logit model which is the logit model with a spatial autocorrelation term, ϕ_i.

$$\log\{p_i/(1-p_i)\} = \mathbf{x}_i^T\boldsymbol{\beta} + \phi_i + \varepsilon_i \tag{19.13}$$

The assumption on the uncorrelated error term ε_i is the same as in the Poisson-Gamma model. The third term in the expression, ϕ_i, is a *spatial random effect*, one for each observation.

Together, the spatial effects are distributed as a complex *multivariate normal* (or Gausian) density function. In other words, the second model is a spatial regression component within a logit model. The spatial effect, ϕ_i, is a *spatial random effect*, one for each observation. It can be modeled as either a CAR (equation 19.1) or a SAR (equation 19.2).

Spatial Weights Function

For all the CAR and SAR models, the spatial weights function, w_{ij}, is a function of the neighborhood parameter, α, which is a distance decay function. Three distance weight functions are available in *Crimestat*:

1. **Negative Exponential Distance Decay**

$$w_{ij} = e^{-\alpha d_{ij}} \tag{19.14}$$

where d_{ij} is the distance between two zones or points and α is the decay coefficient. The weight decreases with the distance between zones with α indicating the degree of decay.

2. Restricted Negative Exponential Distance Decay

$$w_{ij} = Ke^{-\alpha d_{ij}} \tag{19.15}$$

where *K* is 1 if the distance between points is less than equal to a search distance and 0 if it is not. This function stops the decay if the distance is greater than the user-defined search distance (i.e., the weight becomes 0).

3. Contiguity Function

$$c_{ij} = w_{ij} \tag{19.16}$$

where w_{ij} is 1 if observation j is within a specified search distance of observation *i (*a neighbor) and 0 if it is not.

Estimation Procedures for Spatial Models

For each of the spatial regression models used, we follow the same steps that were outlined in Chapter 17. Conceptually, these are:

1. Specifying a functional model and setting up the model parameters.

2. A likelihood function is set up and prior distributions for each parameter are assumed.

3. A joint posterior distribution for all unknown parameters is defined by multiplying the likelihood and the priors.

4. Repeated samples are drawn from this joint posterior distribution.

5. The estimates for all coefficients are based on the results of the *M-L* samples, for example the mean, the standard deviation, the median and various percentiles. Similarly, the overall model fit is based on the *M-L* samples.

Determining a Distance Decay Function for Alpha

Each of these steps is applied to the specified models discussed above. For a spatial regression model, a distance function has to be defined. Alpha (α) is the exponent for the distance decay function in the spatial model. Essentially, the distance decay function defines the weight to be applied to the values of nearby records. The weight can be defined by one of three mathematical functions. First, the weight can be defined by a *negative exponential* function,

$$\text{Weight} = e^{-\alpha * d(ij)} \qquad (19.17)$$

where *d(ij)* is the distance between observations in specified units (e.g., miles, meters) and α is the value for alpha, again consistent with the specified distance units. It is automatically assumed that alpha will be negative whether the user puts in a minus sign or not. The user inputs the alpha value in this box.

Second, the weight can be defined by a *restricted negative exponential* whereby the negative exponential operates up to the specified search distance, whereupon the weight becomes 0 for greater distances

$$\text{Up to Search distance:} \quad \text{Weight} = e^{-\alpha * d(ij)} \qquad \text{for } d(ij) \geq 0,\ d(ij) \leq d_p \qquad (19.18)$$

$$\text{Beyond search distance:} \quad 0 \qquad \text{for } d(ij) > d_p \qquad (19.19)$$

where d_p is the search distance. The coefficient for the linear component is assumed to be 1.0.

Third, the weight can be defined as a *uniform* value for all other observations within a specified search distance. This is a *contiguity* (or adjacency) measure. Essentially, all other observations have an equal weight within the search distance and 0 if they are greater than the search distance. The user inputs the search distance and units in this box.

Determining Reasonable Values for Alpha

The default function for the weight is a negative exponential with a default alpha value of -1 in miles. For many data sets, this will be a reasonable value. However, for other data sets, it will not. Reasonable values for alpha with the negative exponential function are obtained with the following procedure:

1. Decide on the measurement units to be used to calculate alpha (miles, kilometers, feet, etc). The default is miles. *CrimeStat* will convert from the units defined for the Primary File input dataset to those specified by the user.

2. Calculate the nearest neighbor distance from the Nna routine on the Distance Analysis I page. These may have to be converted into units that were selected in step 1 above. For example, if the Nearest Neighbor distance is listed as 2000 feet, but the desired units for alpha are miles, convert 2000 feet to miles by dividing the 2000 by 5280.

3. Input the dependent variable as the Z (intensity) variable on the Primary File page.

4. Run the Moran Correlogram routine on this variable on the Spatial Autocorrelation page (under Spatial Description). By looking at the values and the graph, decide whether the distance decay in this variable is very 'sharp' (drops off quickly) or very 'shallow' (drops off slowly).

5. Define the appropriate weight for the nearest neighbor distance:

    a. Assume that the weight for an observation with itself (i.e., distance = 0) is 1.0.

    b. If the distance decay drops off sharply, then a low weight for nearby values should be given. Assume that any observations at the nearest neighbor distance will only have a weight of 0.5 with observations further away being even lower.

    c. If the distance decay drops off more slowly, then a higher weight for nearby values should be given. Assume that any observations at the nearest neighbor distance will have a weight of 0.9 with observations further away being lower but only slightly so.

    d. An intermediate value for the weight is to assume it to be 0.75.

6. A range of alpha values can be solved using these scenarios:

    a. For the sharp decay, alpha is given by:

$$\alpha = \ln(0.5)/NN(distance) \qquad (19.20)$$

where NN(distance) is the nearest neighbor distance in specified distance units (e.g., feet, meters, kilometers)

b. For the shallow distance decay, alpha is given by:

$$\alpha = \ln(0.9)/NN(distance) \quad (19.21)$$

where NN(distance) is the nearest neighbor distance.

c. For the intermediate decay, alpha is given by:

$$\alpha = \ln(0.75)/NN(distance) \quad (19.22)$$

where NN(distance) is the nearest neighbor distance.

These calculations will provide a range of appropriate values for α. The diagnostics routine automatically estimates these values as part of its output.

Value for Zero Distance Between Records

The advanced options dialogue has a parameter for the minimum distance to be assumed between different records. If two records have the same X and Y coordinates (which could happen if the data are individual events, for example), then the distance between these records will be 0. This could cause unusual calculations in estimating spatial effects. Instead, it is more reliable to assume a slight difference in distance between all records. The default is 0.005 miles but the user can modify this (including substituting 0 for the minimal distance).

GUIDELINE:

Note that MCMC spatial regression models will take a very long time to calculate. For large datasets, we recommend using the block sampling method discussed in chapter 17. A rough rule-of-thumb is that if the dataset is larger than 2,000 cases, the block sampling method should be used for spatial MCMC models. Of course, this will depend on the amount of available RAM as well as the processing speed of the computer.

Examples of Spatial Regression Modeling

Example 1: MCMC Normal-CAR Analysis of Houston Burglaries

The first example is the MCMC Normal-CAR model using the Houston burglary data set. The data came from the Houston Police Department. There were 26,480 burglaries that occurred in 2006 which were allocated to 1,179 Traffic Analysis Zones (TAZ) within the City of Houston. The independent variables were the number of households in 2006 (estimated by the Houston-Galveston Area Council, the metropolitan planning organization) and the median household income for 2000 (from the 2000 U.S. Census).

With a spatial regression model, the user has to provide a value for the distance decay term, alpha (α). The diagnostics routine that was discussed in Chapter 15 provides plausible values of α given the decline in spatial autocorrelation as measured by the Moran Correlogram. The diagnostic calculates the nearest neighbor distance (the average distance of the nearest neighbors for all observations) and then estimates values based on weights assigned to this distance. Three weights are estimated: 0.9, 0.75 and 0.5. We utilized the 0.75 weight. In the example, based on the nearest neighbor distance of 0.45 miles and a weight of 0.75, the α value would be -0.637 for distance units in miles.

Table 19.1 presents the results. For comparison, we repeat the results for the non-spatial MCMC Normal model (from Table 17.2 in Chapter 17). The R-square of the spatial model is slightly worse than the non-spatial model. But, remember, these are estimates based on samples and will vary from run to run. The log likelihood is better for the non-spatial model than for the spatial model. The AIC and BIC/SC statistics are also better. However, the Mean Absolute Deviation (MAD) and the Mean Squared Predictive Error (MSPE) are similar for the two models. There are subtle differences in the MAD and MSPE between the models (e.g., the MCMC Normal-CAR has better MAD and MSPE scores for the third and fourth quartiles but not for the first two), but the differences are very small.

The biggest differences are in the coefficients. The intercept is smaller for the MCMC Normal-CAR while the average Phi coefficient (the average of all the Phi values for individual records) is highly significant. The coefficients for households and for median household income are about the same. In other words, the spatial autocorrelation component (estimated by Phi) absorbed a lot of variance in the dependent variable and 'pulled' this from the intercept. Keep in the mind that the intercept is a constant that is added to the predicted values for all records.

Two points should be noted. First, as mentioned above, the MCMC Normal-CAR (or MCMC Normal-SAR) model assumes that the dependent variable is approximately normally distributed. However, as discussed in Chapter 15, the Houston burglary data by TAZ is highly

Table 19.1:

Predicting Burglaries in the City of Houston: 2006
MCMC Normal-CAR Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1179
Df:	1174
Type of regression model:	Normal-CAR
Method of estimation:	MCMC
Number of iterations:	25000 Burn in: 5000
Distance decay function:	Negative exponential

Likelihood statistics

Log Likelihood:	-6,685.7
AIC:	13,038.5
BIC/SC:	13,381.4
R^2:	0.47

Model error estimates

Mean absolute deviation:	13.2
1st (highest) quartile:	27.0
2nd quartile:	11.4
3rd quartile:	7.6
4th (lowest) quartile:	6.8
Mean squared predicted error:	510.0
1st (highest) quartile:	1,510.0
2nd quartile:	312.5
3rd quartile:	128.3
4th (lowest) quartile:	93.5

Predictor	Mean	Std	t-value[p]	MC error	MC error/ std	G-R stat
INTERCEPT	9.2048	0.582	15.83***	0.009	0.016	1.001
HOUSEHOLDS	0.0274	0.0004	74.93***	0.000003	0.009	1.000
MEDIAN HOUSEHOLD INCOME	-0.0001	0.00001	-10.77***	0.00000002	0.016	1.001
PHI(Average)	0.1538	0.210	0.73	0.095	0.453	4.266

n.s. Not significant
*** $p \leq .001$

Table 17.2 (REPEAT):

Predicting Burglaries in the City of Houston: 2006

MCMC Normal Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1,179
Df:	1,175
Type of regression model:	Poisson with Lognormal dispersion
Method of estimation:	MCMC
Number of iterations:	25,000 Burn in: 5,000

Likelihood statistics

Log Likelihood:	-5342.6
AIC:	10,693.2
BIC/SC:	10,713.6
R^2:	0.48

Model error estimates

Mean absolute deviation:	13.5
1st (highest) quartile:	26.5
2nd quartile:	10.6
3rd quartile:	8.2
4th (lowest) quartile:	8.6
Mean squared predicted error:	505.1
1st (highest) quartile:	1,501.7
2nd quartile:	272.3
3rd quartile:	130.5
4th (lowest) quartile:	120.0

Predictor	Mean	Std	t-value[p]	MC error	MC error/ std	G-R stat
INTERCEPT	12.7804	1.235	10.35***	0.020	0.016	1.001
HOUSEHOLDS	0.0255	0.001	32.62***	0.000009	0.011	1.0005
MEDIAN HOUSEHOLD INCOME	-0.0002	0.00003	-7.00***	0.0000004	0.015	1.0004

** p≤.01

*** p≤.001

skewed, and over-dispersed. Thus, the Normal model (whether tested with MLE or MCMC) is not appropriate for a highly skewed dependent variable. The MCMC Normal seems to have done a good job of replicating the MLE Normal for this dataset, but neither model is appropriate for a skewed dependent variable. Instead, one of the Poisson-family models should be used.

Second, the two diagnostic statistics for Phi that indicate whether the distribution has converged on an 'equilibrium' state, namely the MC Error/STD and the G-R statistics, are much higher than would normally be acceptable (i.e., below 0.05 and 1.20 respectively). We are not completely sure why this occurs, but these particular statistics do not get smaller for Phi in the MCMC Normal-CAR (or SAR) models even with a large number of iterations. In this model only, these diagnostic statistics are much higher than with the Poisson models. Users should be aware of this. Most importantly, though, is that users should ensure that the two diagnostic indicators are low for the independent variables, which they are in Table 19.1. This will indicate that the model has converged to an equilibrium and can be trusted.

Example 2: MCMC Poisson-Gamma-CAR Analysis of Houston Burglaries

In the second example, we ran the Houston burglary data set using a Poisson-Gamma-CAR model since this model is appropriate when there is an over-dispersed dependent variable. The procedure we follow is similar to that outlined in Oh, Lyon, Washington, Persaud, and Bared (2003). First, we ran the Poisson-Gamma model that was illustrated in Chapter 16, Table 16.2 and saved the residual errors.

Second, we tested the residual errors for spatial autocorrelation using the Moran's "I" routine in *CrimeStat*. As expected, the "I" for the residuals was highly significant ("I" = 0.0089; $p \leq .001$) indicating that there is substantial spatial autocorrelation in the error term.

Third, we utilized an α value of -0.637 for distance units in miles as in the MCMC Normal-CAR model and ran the Poisson-Gamma-CAR model. Table 19.2 present the results. The likelihood statistics indicate that the overall model fit was similar to that of the Poisson-Gamma model. However, the log likelihood was slightly lower and the DIC, AIC and BIC/SC were slightly higher. Similarly the deviance the Pearson Chi-square tests were slightly higher. In other words, the Poisson-Gamma-CAR model does not have a higher likelihood than the Poisson-Gamma model. The reason is that the inclusion of the spatial component, ϕ_i, has not improved the predictability of the model. The DIC, AIC, BIC, deviance, and Pearson Chi-square statistics penalize the inclusion of additional variables.

Regarding individual coefficients, the intercept and the two independent variables have values similar to that of MCMC Poisson-Gamma presented in Table 16.2. Note, though, that the coefficient value for the intercept is now smaller. The reason is that the spatial effects, the ϕ_i values, have absorbed some of the variance that was previously associated with the intercept. The table presents an average Phi value over all observations. The overall average was not statistically significant. However, Phi values for individual coefficients were output as an individual file and the predicted values of the individual cases include the individual Phi values.

Table 19.2:

Predicting Burglaries in the City of Houston: 2006
MCMC Poisson-Gamma-CAR Model

(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1179
Df:	1174
Type of regression model:	Poisson-Gamma-CAR
Method of estimation:	MCMC
Number of iterations:	25000 Burn in: 5000
Distance decay function:	Negative exponential
Likelihood statistics	
Log Likelihood:	-4433.3
DIC:	10,853.8
AIC:	8,876.5
BIC/SC:	8,901.9
Deviance:	1,469.5
p-value of deviance:	0.0001
Pearson Chi-square:	1,335.0
Model error estimates	
Mean absolute deviation:	45.1
Mean squared predicted error:	94,236.4
Over-dispersion tests	
Adjusted deviance:	1.3
Adjusted Pearson Chi-Square:	1.1
Dispersion multiplier:	1.4
Inverse dispersion multiplier:	0.7

Predictor	Mean	Std	t-value[p]	MC error	MC error/ std	G-R stat
INTERCEPT	2.2164	0.094	23.53***	0.0034	0.039	1.015
HOUSEHOLDS	0.0012	0.00007	17.90***	0.000001	0.021	1.003
MEDIAN HOUSEHOLD INCOME	-0.000008	0.000002	-5.18***	0.00000003	0.020	1.003
PHI(Average)	0.024	0.026	0.95n.s.	0.001	0.056	1.023

n.s. Not significant

*** $p \leq .001$

Figure 19.3 shows the residual errors from the Poisson-Gamma-CAR model. As seen, the model overestimated on the west, southwest and southeast parts of Houston. This is in contrast with the normal model (Figure 15.4 in Chapter 15), which underestimated in the southwest part of Houston with similar overestimation in the west and southeast. The Poisson-Gamma-CAR model has shifted the estimation errors in the southwest. As we have seen, this may not be the best model for this data set, though it is not particularly bad.

Spatial Autocorrelation of the Residuals from the Poisson-Gamma-CAR model

When we look at spatial autocorrelation among the residual errors, we now find much less spatial autocorrelation. The Moran's "I" test for the residual errors was 0.0091. It is significant, but much less than before. To understand this better, Table 19.3 presents the "I" values and the Getis-Ord "G" values for a search area of 1 mile for the raw dependent variable (2006 burglaries) and four separate models – the normal (OLS), the Poisson-NB1, the MCMC Poisson-Gamma (non-spatial), and the MCMC Poisson-Gamma-CAR, along with the Φ coefficient from the Poisson-Gamma-CAR model.

Table 19.3:

Spatial Autocorrelation in Residual Errors of the Houston Burglary Model Comparing Different Poisson Models

		Residual Errors				
	Raw Dependent Variable	**Normal Model**	**Poisson NB1 Model**	**MCMC Poisson-Gamma Model**	**MCMC Poisson-Gamma-CAR Model**	**Poisson Gamma-CAR Φ Coefficient**
Moran's "I"	0.252****	0.057****	0.119****	0.009***	0.009***	0.042****
Getis-Ord "G" (1 mile search radius)	0.007****	-6.785**	-16.118**	0.019n.s.	0.017 n.s.	0.027n.s.

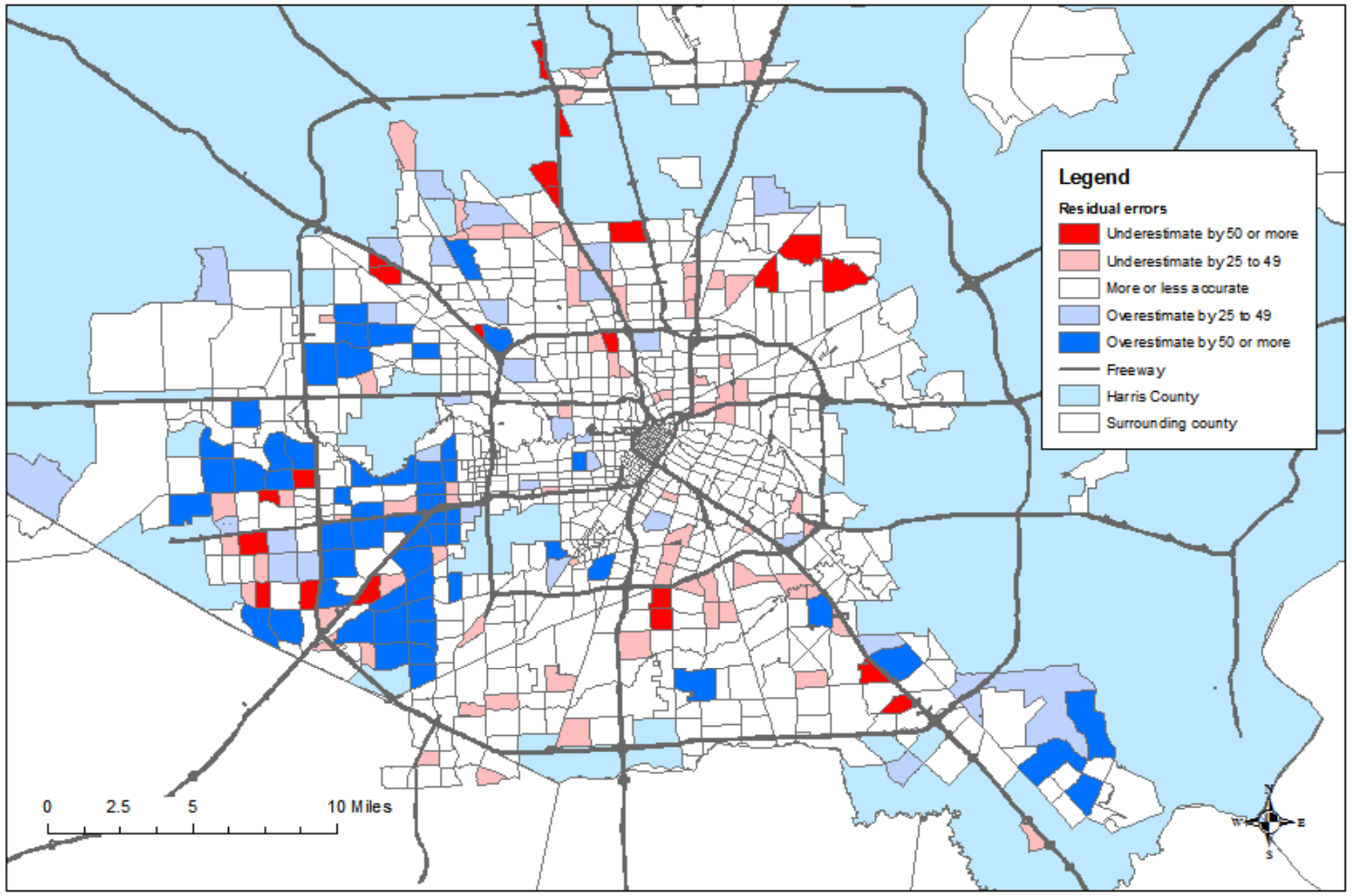
n.s. Not significant
** p≤.01
*** p≤.001
**** p≤.0001

Moran's "I" tests for positive and negative spatial autocorrelation. A positive value indicates that adjacent zones are similar in value while a negative value indicates that adjacent zones are very different in value (i.e., one being high and one being low). As can be seen, there

Figure 19.3:

Predicting Burglaries in the City of Houston: 2006

Residual Errors from Poisson-Gamma-CAR Model



is positive spatial autocorrelation for the dependent variable and for each of the four comparison models. However, the amount of positive spatial autocorrelation decreases substantially. With the raw variable – the number of 2006 burglaries per zone, there is sizeable positive spatial autocorrelation. However, the models reduce this substantially by accounting for some of the variance of this variable through the two independent variables. The two negative binomial (Poisson-Gamma) models have the least amount with little difference between the Poisson-Gamma and the Poisson-Gamma-CAR.

The Getis-Ord "G" statistic, however, distinguishes two types of positive spatial autocorrelation, positive spatial autocorrelation where the zones with high values are adjacent to zones also with high values (high positive) and positive spatial autocorrelation where the zones with low values are adjacent zones also with low values (low positive). This is a property that Moran's "I" test cannot do.

The "G" has to be compared to an expected "G", which is essentially the sum of the weights. However, when used with negative numbers, such as residual errors, the "G" has to be compared with a simulation envelope. The statistical test for "G" in Table 19.3 tests whether the observed "G" was higher than the 97.5^{th} or 99.5^{th} percentiles (high positive) or lower than the 2.5^{th} or 0.5^{th} percentiles (low positive) of the simulation envelope.

The results show that the "G" for the raw burglary values are 'high positive', meaning that zones with many burglaries tend to be near other zones also with many burglaries. For the analysis of the residual errors, however, the normal and Poisson-NB1 models are negative and significant, meaning that they show positive spatial autocorrelation but the 'low positive' type. That is, the clustering occurs because zones with low residual errors are predominately near other zones with low residual errors. The models have better predicted the zones with fewer numbers of burglaries than those with higher numbers. On the other hand, the residuals errors for the MCMC Poisson-Gamma and for the MCMC Poisson-Gamma-CAR models are not significant. In other words, these models have accounted for much of the effect measured by the "G" statistic.

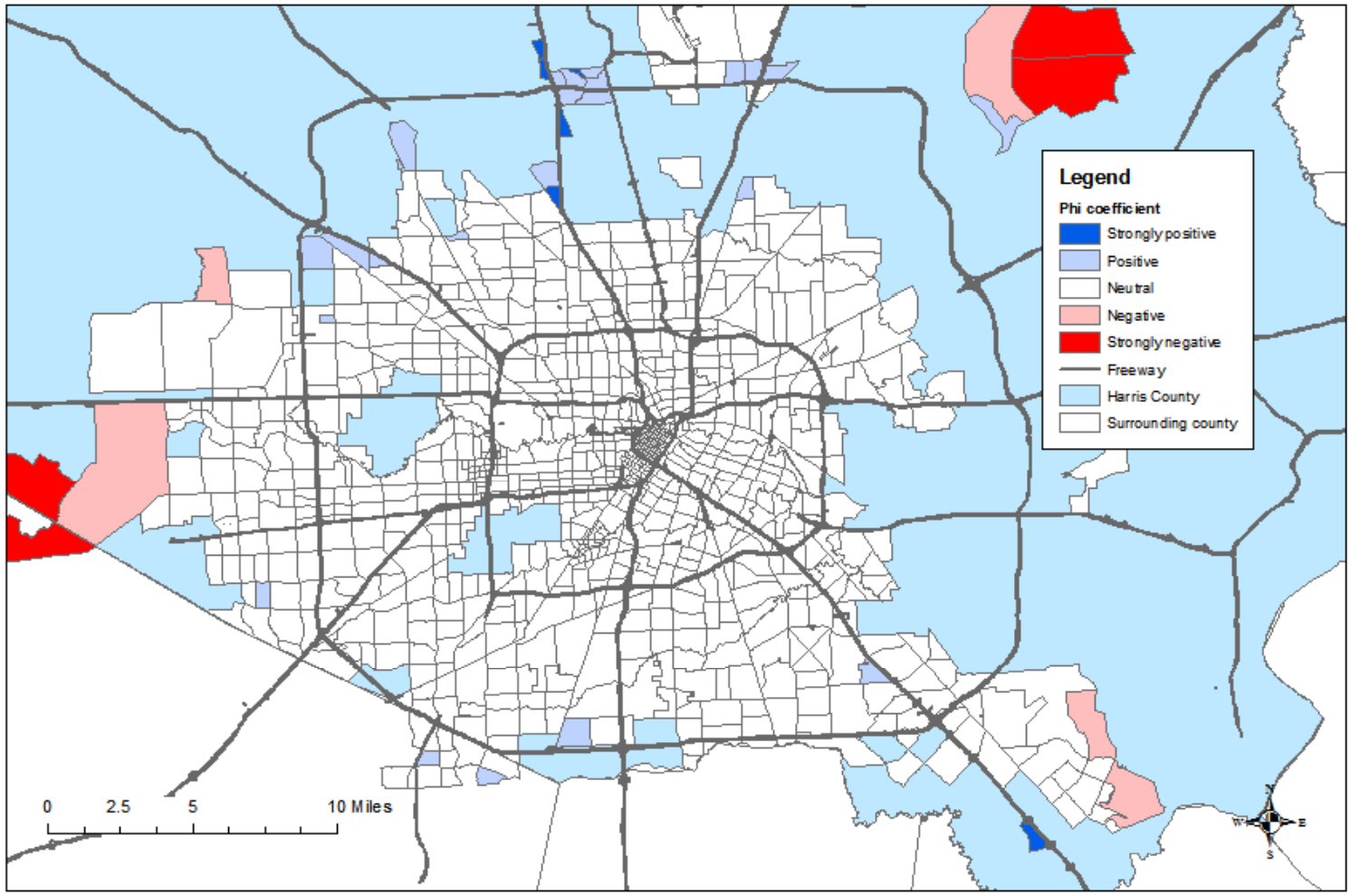
The last column analyzes the spatial autocorrelation tests on the individual Phi coefficients. There is spatial autocorrelation for the Phi values, as seen by a very significant Moran "I" value, but it is neither a 'high positive' or a 'low positive' based on the "G" test. In other words, the Phi values appear to be neutral with respect to the clustering of residual errors.

Figure 19.4 shows the distribution of the Phi values. By and large, the spatial adjustment is very minor in most parts of Houston with its greatest impact at the edges, where one might expect some spatial autocorrelation due to very low numbers of burglaries and 'edge effects'.

Figure 19.4:

Predicting Burglaries in the City of Houston: 2006

Phi Coefficients from Poisson-Gamma-CAR Model



Putting this in perspective, the spatial effects in the Poisson-Gamma-CAR model are small adjustments to the predicted values of the dependent variable. They slightly improve the predictability of the model but do not fundamentally alter it. Keep in mind that spatial autocorrelation is a statistical effect of some other variable operating that is not being measured in the model. Spatial autocorrelation is not a 'thing' or a process but the result of not adequately accounting for the dependent variable.

In theory, with a correctly specified model, the variance of the dependent variable should be completely explained by the independent variables with the error term truly representing random error. Thus, there should be no spatial autocorrelation in the residual errors under this ideal situation. The example that we have been using is an overly simple one. There are clearly other variables that explain the number of burglaries in a zone other than the number of households and the median household income – the types of buildings in the zone, the street layout, lack of visibility, the types of opportunities for burglars, the amount of surveillance, and so forth. The existence of a spatial effect is an indicator that the model could still be improved by adding more variables.

Example 3: Modeling Burglary Risk in Houston

In Chapter 17, we examined risk analysis and used the MCMC algorithm with the Poisson-Gamma model to estimate risk. This can be extended to spatial analysis. To illustrate this type of model, we ran an MCMC Poisson-Gamma-CAR model on the Houston burglary data using the number of households as the exposure variable. There was, therefore, only one independent variable, median household income. Table 19.4 shows the results along with the expanded output that is obtained by clicking on the 'Expanded output' button.

The summary statistics indicate that the overall model fit is good. The log likelihood is high while the AIC and BIC are moderately low. Compared to the non-exposure burglary model (Table 19.2), the model does not fit the data as well. The log likelihood is lower (i.e., more negative) while the AIC and BIC are higher. Further, the DIC is very high

For the model error estimates, the MAD and the MSPE are smaller, suggesting that the burglary risk model is more precise, though not more accurate. However, the dispersion statistics indicate that there is ambiguity over-dispersion. The dispersion multiplier is very low which, by itself, would suggest that a "pure" Poisson model could be used. However, both the adjusted Pearson Chi-square and the adjusted deviance are higher. In other words, the exposure variable has not eliminated the dispersion as much as in the random effects (non-exposure) model.

Looking at the coefficients, the offset variable (number of households) has a coefficient of 1.0 because it is defined as such. The coefficient for median household income is still negative, but is stronger than in Table 19.2. The effect of standardizing households as the

Table 19.4:
Predicting Burglary Risk in the City of Houston: 2006
MCMC Poisson-Gamma-CAR Model with Exposure Variable
Extended Output
(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**		
N:	1,179		
Df:	1,174		
Type of regression model:	Poisson-Gamma-CAR		
Method of estimation:	MCMC		
Number of iterations:	25000		
Burn in:	5000		
Distance decay function:	Negative exponential		
Likelihood statistics			
Log Likelihood:	-4,736.6		
DIC:	146,129.2		
AIC:	9,481.2		
BIC/SC:	9,501.5		
Deviance:	2,931.1	p:	0.0001
Pearson Chi-square:	34,702.9	p:	0.0001
Model error estimates			
Mean absolute deviation:	18.6		
Mean squared predicted error:	1,138.9		
Over-dispersion tests			
Adjusted deviance:	2.5		
Adjusted Pearson Chi-Square:	29.5		
Dispersion multiplier:	0.6	p:	n.s.
Inverse dispersion multiplier:	1.7	p:	n.s.

Predictor	Mean	Std	t-value[p]	MC error	MC error/ std	G-R stat
Exposure/offset variable:						
HOUSEHOLDS	1.0					
Linear predictors:						
INTERCEPT	-2.2794	0.0786	-29.00[***]	0.003	0.036	1.007
MEDIAN HOUSEHOLD INCOME	-0.00002	0.000002	-10.38[***]	0.00000005	0.032	1.006
AVERAGE PHI	0.0442	0.0320	-1.38[n.s.]	0.0098	0.021	1.002

n.s. Not significant *** p≤.001

Table 19.4: (continued)

Percentiles	0.5th	2.5th	97.5th	99.5th\
INTERCEPT	-2.4879	-2.4365	-2.1292	-2.0810
MEDIAN HOUSEHOLD INCOME	-0.00002	-0.00002	-0.00001	-0.00001
AVERAGE PHI	-0.1442	-0.1128	0.0145	0.0348

baseline exposure variable has increased the importance of household income in predicting the number of burglaries, controlling for the number of households. Finally, the average Φ value is positive but not significant, similar to what it was in Table 19.2.

Expanded output

The use of t-tests to evaluate whether coefficients are significantly different than zero depends on whether the underlying distribution for the coefficients is normal or not. In the case of skewed count data and complex models that are the products of multiple individual functions, it is not always clear whether that assumption is valid or not. Consequently, the *CrimeStat* MCMC module allows the output of statistics that show the distribution of the coefficients in terms of several percentiles: 0.5%, 2.5%, 5%, 10%, 25%, 50%, 75%, 90%, 95%, 97.5% and 99.5%. To obtain these percentiles, the user simply checks the 'Expanded output' box on the MCMC interfact.

In table 19.4 above, we have shown only four of them, the 0.5^{th}, 2.5^{th}, 97.5^{th}, and 99.5^{th} percentiles. The 2.5^{th} and 97.5^{th} represent 95% credible intervals for a two-tailed test while the 0.5^{th} and 99.5^{th} represent 99% credible intervals also for a two-tailed test.

One way to interpret the percentiles is to check whether a coefficient of 0 (the 'null hypothesis') or any other particular value falls outside the 95% or 99% credible intervals. For example, with the intercept term, the 95% credible interval is defined by -2.4365 to -2.1292. Since both are negative, clearly a coefficient of 0 is outside this range; in fact, it is outside the 99% credible interval as well (-2.4879 to -2.0810). In other words, the intercept is *significantly* different than 0, though the use of the term 'significant' is different than with the usual asymptotic normality assumptions since it is based on the distribution of the parameter values from the MCMC simulation.

Of the other parameters that were estimated, median household income is also significant beyond the 99% credible interval but the Φ coefficient is not significantly different than a 0 coefficient (i.e., a Φ of 0 falls between the 2.5th and the 97.5th percentiles).

In other words, percentiles can be used as a non-parametric alternative to the t- or Z-test. Without making assumptions about the theoretical distribution of the parameter value (which the t- and Z-test do – they are assumed to be normal or near normal for "t"), significance can be assessed empirically. Usually, the t-test and the percentile distribution will lead to the same inference, which they do in table 19.5. But, it is possible that they could differ.

In summary, in risk analysis, an exposure variable is defined and held constant in the model. Thus, the model is really a risk or rate model that relates the dependent variable to the baseline exposure. The independent variables are now predicting the rate, rather than the count by itself.

Example 4: MCMC Binomial Logit-CAR Analysis of Houston Robberies

A final example of spatial regression modeling applies the spatial autocorrelation component to the binomial logit model. The test is whether weapons were used in 3,709 Houston robberies that occurred in 2007-09 in which the offender had been arrested. This was the example presented in Chapter 18. The dependent variable was whether a physical weapon had been used, either a firearm, a knife, a stick or another physical object, compared to physical force or the threat of force. The independent variables were the age and gender of the offender, the number of suspects involved, whether the robbery occurred at night (6 PM – 6 AM), the median household income of the zone in which the robbery occurred, and the distance between the robbery location and downtown Houston.

However, now we will examine the distribution using a spatial autocorrelation function, the conditional autoregressive function. The diagnostic routine was run in CrimeStat to determine an appropriate distance decay value (α); this turned out to be -0.4237 in miles. Table 19.5 presents the results.

For this model, the block sampling method discussed in Chapter 17 was used. Comparing these results with the non-spatial binomial logit model for Houston robbery weapon use (Table 18.2), the log-likelihood, AIC and BIC values are slightly stronger for the spatial model than the non-spatial model, suggesting that the spatial adjustment to individual records has improved the overall probability. The goodness of fit statistics (the mean absolute deviation and mean squared predicted error) are slightly lower for the spatial model than for the non-spatial model. In particular, the second and third quartiles show slightly lower errors for the Mean Absolute Deviation in the spatial model than the non-spatial model.

Table 19.5:

Weapon Use by 2007-09 Houston Robbers: MCMC Binomial Logit-CAR Model

(N=3,709 Robberies with Known Origin & Destination Coordinates)

DepVar:	**WEAPON USE IN ROBBERY**		
N:	3,709		
Df:	3,700		
Type of regression model:	Logit		
Number of block samples:	25	Average block sample size: 395.2	
Method of estimation:	MCMC		
Number of iterations:	25,000	Burn in: 5,000	
Likelihood statistics			
Log Likelihood:	-2,501.3		
AIC:	5,020.5		
BIC/SC:	5,076.5		
Deviance:	-1,204.4	p: 0.0001	
Pearson Chi-square:	1,359.9	p: 0.0001	
Model error estimates			
Mean absolute deviation:	0.4		
1st (highest) quartile:	0.3		
2nd quartile:	0.3		
3rd quartile:	0.4		
4th (lowest) quartile:	0.6		
Mean squared predicted error:	0.2		
1st (highest) quartile:	0.1		
2nd quartile:	0.1		
3rd quartile:	0.3		
4th (lowest) quartile:	0.4		
Dispersion tests			
Adjusted deviance:	-0.3	p: n.s.	
Adjusted Pearson Chi-Square:	0.4	p: n.s.	

Predictor	Mean	Adj. Std	Adj. t-value[p]	.	MC error	MC error/ std	G-R stat	Odds ratio
Intercept:	0.6784	0.495	4.20***		0.019	0.038	1.011	1.971
AGE	-0.0242	0.012	-6.27***		0.0004	0.032	1.011	0.976
GENDER	-0.6122	0.372	-5.05***		0.005	0.013	1.003	0.542
# SUSPECTS	0.3486	0.147	7.25***		0.004	0.025	1.006	1.417
NIGHT	0.6753	0.323	6.40***		0.005	0.015	1.005	1.965
MED HH INC	-0.000006	0.0000	-2.21*		0.000	0.026	1.004	1.000
DISTANCE TO DOWNTOWN	0.0384	0.023	5.04***		0.000	0.016	1.004	1.039
AVERAGE PHI	-0.0006	0.005	-0.33n.s.		0.000	0.016	1.006	0.999

*** p≤.001 ** p≤.01 * p≤.05 n.s. Not significant

The coefficients and adjusted standard errors are very similar between the two models. They differ only in the second or third decimal place. The biggest difference is for nighttime weapon use, where the spatial coefficient is 0.6753 compared to the non-spatial coefficient of 0.5249. This suggest that when spatial location is considered, the nighttime effect in serious weapon use is actually stronger; that is, because the bulk of robberies occur during the daytime but are more clustered spatially, the use of weapons during robberies actually increases at nighttime when controlling for spatial location.

As with the Poisson-Gamma-CAR model, the overall spatial autocorrelation coefficient (Average Phi) is not significant. This is not surprising since the CAR spatial adjustment is done for individual records. In short, the binomial logit-CAR model has produced a slightly better fit to the data than the non-spatial binomial logit model. For prediction, one would use the spatial version because of its better fit.

Caveat

As mentioned earlier, any spatial regression model is attempting to identify a spatial effect due to clustering, dispersion or some combination whereby the values of nearby zones are similar or different than the central zone (the zone being modeled). In effect, the error term of the model is broken into two parts, one associated with a spatial effect (most likely clustering of nearby zones but sometimes dispersion – negative spatial autocorrelation) and the other with unexplained variance.

What this usually signifies is that there are missing variables that should be included in the model, but which are not. For example, Levine (1999) examined the effects of local growth control measures on housing production in California counties and cities using an OLS spatial lag model (Anselin, 1992). The initial model showed a significant negative spatial effect. However, it was discovered that this was mostly the result of low population density. When density was added to the model, the negative spatial lag effect disappeared.

The important thing to realize with these models is that they identify some variability associated with the dependent variable that needs to be explained. The spatial effect is not real, but merely a statistical artifact of examining similarities or differences between nearby zones in the dependent variable. The spatial indices are useful in that they will indicate whether there is a general spatial effect covering all observations (e.g., distance from downtown; area of the zone) or whether clustering or dispersion is specific to only a limited number of observations (e.g., the ϕ_i coefficient). However, ultimately, the researcher needs to find other variables that account for these effects in order to produce a more stable and realistic model.

Summary

To summarize, in this chapter we have gone through a number of spatial regression models that apply to normal, Poisson-distributed and binomial logit models. The choice of any of these models is going to depend on the actual distribution of the dependent variable and the underlying assumptions for that model. For example, an MCMC Normal-CAR or MCMC Normal-SAR model only applies if the dependent variable is normally distributed; the use of such a model with non-normal data will usually lead to biased coefficient estimates. Similarly, the two Poisson spatial regression models presented, the MCMC Poisson-Gamma-CAR/.SAR and the MCMC Poisson-Lognormal-CAR/SAR, are applicable if the dependent variable is a count variable or is highly skewed with an absolute zero minimum. The difference is that the MCMC Poisson-Lognormal-CAR/SAR model is used when there is a small sample and a low sample mean (i.e., most zones have 0 events). Finally, the MCMC Binomial Logit-CAR/SAR model is applicable when the dependent variable is binomial and takes the value 0 or 1.

For each of these, the user must define an appropriate distance decay function (α) on the Advanced Options page of the Regression I module. On the interface of the Regression I module, the user can check the diagnostics box to provide plausible values of α based on a Moran Correlogram (see Chapter 5).

In each of these cases, though, the user is advised to first fit a non-spatial model to see if it produces meaningful results. There are two reasons for this. First, unless the independent variables are properly chosen, ambiguity can be introduced by adding a spatial parameter since it is capturing unobserved variability. By 'properly', we mean that all the independent variables are relatively independent (i.e., little multicollinearity) and statistically significant. If a *clean* non-spatial model can be developed first, then adding a spatial autocorrelation component will allow the user to see whether there is clustering among the observations that could account for some of the effects assigned to the independent variables. But, if the model is not clean, then the results are liable to be confusing.

The second reason is practical. The spatial regression models can take a long time to run, as much as several hours. It is more practical to develop a non-spatial model before trying to fit a spatial to it. The spatial model should be the last step in the modeling, not the first one.

Finally, Chapter 20 will present an overview of the *CrimeStat* regression module. It should be seen as a guide to running the routines.

References

Alonso, W. (1964). *Location and Land Use: Towards a General Theory of Land Rent*. Harvard University Press: Cambride, MA.

Anselin, L.. (1992). *SpaceStat: A Program for the Statistical Analysis of Spatial Data*. Santa Barbara, CA: National Center for Geographic Information and Analysis, University of California.

Besag, J., 1974. Spatial interaction and the statistical analysis of lattice systems (with discussion). *J. Roy.Statist. Soc. Ser. B 36*, 192–236.

De Smith, M., Goodchild, M. F., & Longley, P. A. (2007). *Geospatial Analysis* (second edition). Matador: Leicester, U.K.

Heskin, A., Levine, N. & Garrett, M. (2000). "Rent control and vacancy control: a spatial analysis of four California cities". *Journal of the American Planning Association*. 66 (2), 162-176.

Hipp, J. R. (2007). Block, Tract, and Levels of Aggregation: Neighborhood Structure and Crime and Disorder as a Case in Point. *American Sociological Review* 72:659-680.

Kim, H. & Lim, H. J. (2010). Comparison of Bayesian spatio-temporal models for chronic diseases. *Journal of Data Sciences*, 8, 189-211.

Kneebone, E. & Raphael, S. (2011). *City and Suburban Crime Trends in Metropolitan America*. Metropolitan Opportunity Series, Metropolitan Policy Program, Brookings Institution: Washington, DC. http://www.brookings.edu/papers/2011/0526_metropolitan_crime_kneebone_raphael.aspx. Accessed April 28, 2012.

Lam, N. S. & De Cola, L. (1993). *Fractals in Geography*. The Blackburn Press: Caldwell, NJ.

Levine, N. (2011). "Spatial variation in motor vehicle crashes by gender in the Houston Metropolitan Area". *Proceedings of the 4th International Conference on Women's Issues in Transportation. Volume II: Technical Papers*, Transportation Research Board: Washington, DC. 12-25. http://onlinepubs.trb.org/onlinepubs/conf/cp46v2.pdf.

Levine, N. (1999). The effects of local growth management on regional housing production and population redistribution in California, *Urban Studies*. 1999. 36 12, 2047-2068.

References (continued)

Levine, N. & Lee, P. (2013). Crime travel of offenders by gender and age in Manchester, England. Leitner, M. (ed), *Crime Modeling and Mapping Using Geospatial Technologies,* Springer. 145-178.

Miaou, S. P. (2006). "Coding instructions for the spatial regression models in CrimeStat". Unpublished manuscript. College Station, TX.

Oh, J., Lyon, C., Washington, S., Persaud, B., & Bared , J. (2003). "Validation of FHWA crash models for rural intersections: lessons learned". *Transportation Research Record 1840*, 41-49.

Openshaw, S. (1984). *The Modifiable Areal Unit Problem.* Norwich: Geo Books. ISBN 0-86094-134-5.

von Thünen, J. (1826). *The Isolated State in Relation to Agriculture and Political Economy*. English edition, van Suntum, Ulrich. Palgrave Macmillan:Houndsmills, Basingstoke, Hampshire, England, 2009.

Wachter, S. M. & Cho, M. (1991). "Interjurisdictional price effects of land use controls". *Washington University Journal of Urban and Contemporary Law*, 40, 49-63.

Weisburd, D., Groff, E. R., & Yang, S-M (2012). *The Criminology of Place*. Oxford University Press: New York.

Whittle, P., 1954. On stationary process in the plane. *Biometrika*, 41, 434–449.

Wikipedia (2013a). Instrumental variable. Wikipedia. http://en.wikipedia.org/wiki/Instrumental_variable. Accessed January 31, 2013.

Wikipedia (2013b). Specification (regression). Wikipedia. http://en.wikipedia.org/wiki/Specification_(regression). Accessed January 31, 2013.

Wikipedia (2012). Modifiable Area Unit Problem. Wikipedia. http://en.wikipedia.org/wiki/Modifiable_areal_unit_problem. Accessed May 7, 2012.

Wooldridge, J. (2002). Examining the (Ir)Relevance of Aggregation Bias for Multilevel Studies of Neighborhoods and Crime with an Example Comparing Census Tracts to Official Neighborhoods in Cincinnati. *Criminology* 40:681-710.

Chapter 20:

The CrimeStat Regression Module[1]

Ned Levine
Ned Levine & Associates
Houston, TX

Dominique Lord
Zachry Dept. of Civil Engineering
Texas A & M University
College Station, TX

Byung-Jung Park
Korea Transport Institute
Goyang, South Korea

Srinivas Geedipally
Texas Transportation Institute
Arlington, TX

Haiyan Teng
Houston, TX

Li Sheng
Houston, TX

Ian Cahill
Cahill Software
Edmonton, AB

[1] The regression chapters were the result of the effort of many persons. The maximum likelihood routines were produced by Ian Cahill of Cahill Software in Edmonton, Alberta as part of his MLE++ software package. We are grateful to him for providing these routines and for conducting quality control tests. Dr. Shaw-pin Miaou of College Station, TX designed the MCMC algorithm for the Poisson-Gamma-CAR model. Dr. Byung-Jung Park modified the algorithm to incorporate Poisson-Gamma-SAR, Poisson-Lognormal-CAR/SAR, and the MCMC binomial-CAR/SAR models. Dr. Srinivas Geeidpally added the MCMC Normal-CAR/SAR models. Dr. Dominique Lord of Texas A & M University provided technical consulting on the dispersion parameters in these models. Dr. Ned Levine developed the block sampling scheme and provided overall project management. Ms. Haiyan Teng and Dr. Li Sheng programmed the routines and added numerous technical improvements to the algorithms. We are also grateful to Dr. Richard Block of Loyola University in Chicago (IL) for testing the MCMC and MLE routines.

Table of Contents

Table of Contents (continued)

Chapter 20:
The CrimeStat Regression Module

We now describe the *CrimeStat* regression module. There are two pages in the module. Regression I allows the testing of a model while Regression II allows a prediction to be made based on an already-estimated model. Figure 20.1 displays the Regression I page.

Regression I Module

Types of Regression Models

In the current version, 18 possible regression models are available with several options for each of these:

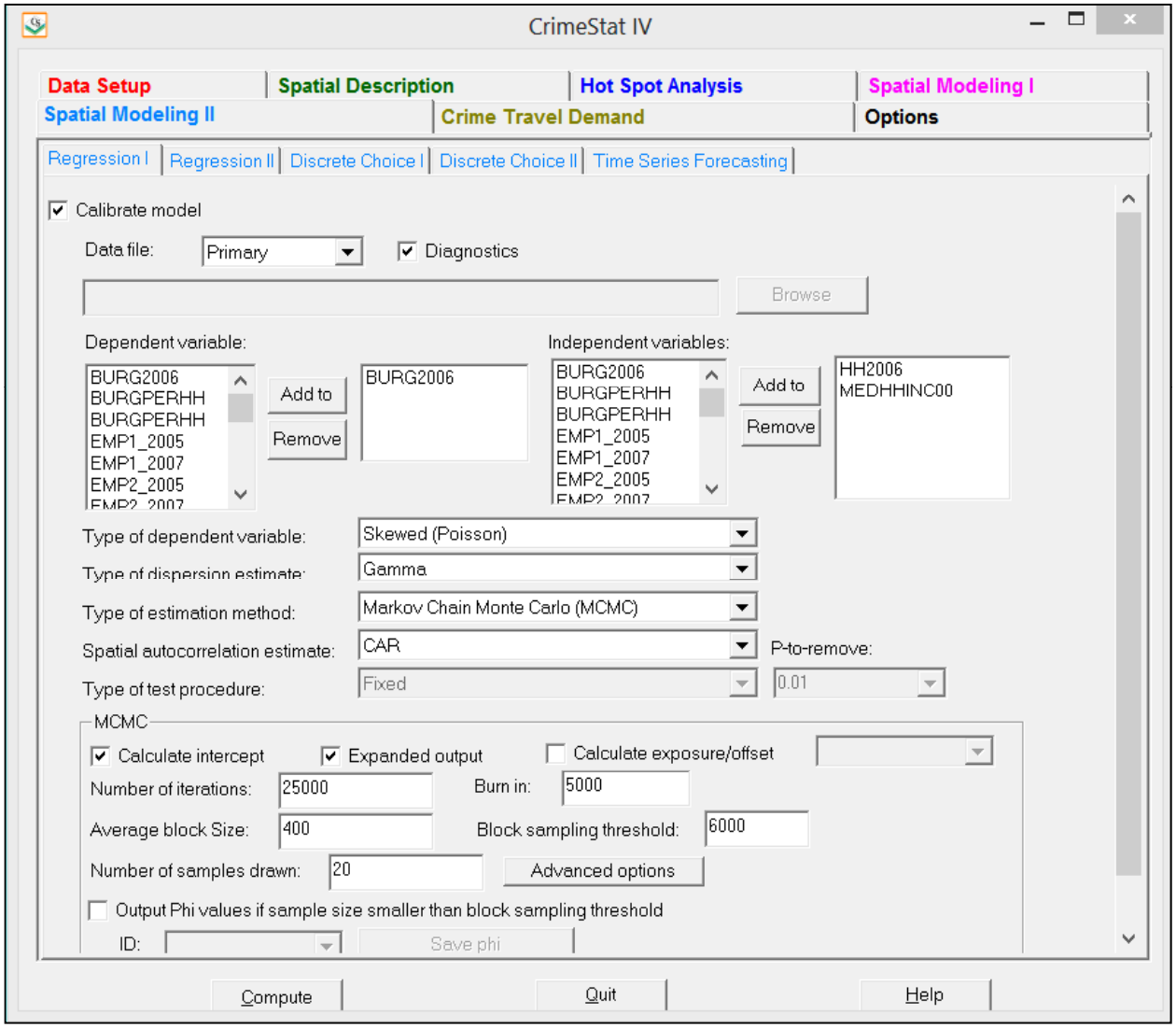
MLE Normal (OLS)
MCMC Normal
MCMC Normal-CAR
MCMC Normal-SAR
MLE Poisson
MLE Poisson with linear dispersion correction (NB1)
MLE Poisson-Gamma (NB2)
MCMC Poisson-Gamma (NB2)
MCMC Poisson-Gamma-CAR
MCMC Poisson-Gamma-SAR
MCMC Poisson-Lognormal
MCMC Poisson-Lognormal-CAR
MCMC Poisson-Lognormal-SAR
MLE Binomial Logit
MLE Binomial Probit
MCMC Binomial Logit
MCMC Binomial Logit-CAR
MCMC Binomial Logit-SAR

In addition, each of the 12 MCMC models can be run with an exposure (offset) variable used to define the population 'at risk' allowing a total of 30 possible regression models to be run.

There are two pages in the module. The Regression I page allows the testing of a model while the Regression II page allows a prediction to be made based on an already-estimated

Figure 20.1:

Regression Modeling I Setup Screen



model. Also, since the Regression I module and Trip Generation module in the Crime Travel Demand Model duplicate regression functions, only one of these can be run at a time.

Input Data Set

The data set for the regression module is the Primary File data set. The coordinate system and distance units are also the same. The routine will not work unless the Primary File has X/Y coordinates.

Dependent Variable

To start loading the module, click on the 'Calibrate model' tab. A list of variables from the Primary File is displayed. There is a box for defining the dependent variable. The user must choose one dependent variable. A keystroke trick is to click on the first letter of the variable that will be the dependent variable and the routine will go to the first variable with that letter.

Independent Variables

There is another box for defining the independent variables. The user must choose one or more independent variables. In the routine, there is no limit to the number. Keep in mind that the variables are output in the same order as specified in the dialogue so a user might want to think how these should be displayed.

Type of Dependent Variable

There are five options that must be defined. The first is the type of dependent variable: Skewed (Poisson), Normal (OLS), Binomial probit, or Binomial logit (logistic). The default is a Poisson.

Type of Dispersion Estimate

The second model decision is the type of dispersion estimate to be used. The choices are Gamma, Poisson, Lognormal, and Poisson with linear correction. For the MLE models, only Gamma, Poisson and Poisson with linear correction are available while for the MCMC models, only Gamma and Lognormal are available. The default is Gamma. For the MLE Normal (OLS) and MCMC Normal-CAR/SAR models, the dispersion is automatically normal. For the binomial logit or binomial probit, the dispersion is automatically binomial.

Type of Estimation Method

The third option is the type of estimation method to be used: Maximum Likelihood (MLE) or Markov Chain Monte Carlo (MCMC). The default is MLE. These methods were discussed in Chapters 15 and 17 and in appendices B and C.

Spatial Autocorrelation Estimate

Fourth, if the user accepts an MCMC algorithm, then a fourth decision is whether to run a spatial autocorrelation estimate along with it (a Conditional Autoregressive function - CAR, or a Simultaneous Autoregressive function - SAR). The MCMC Poisson-Gamma, MCMC Poisson-Lognormal, and MCMC Logit functions can be run with a spatial autocorrelation parameter.

> Note that the CAR model runs quite quickly whereas the SAR model runs very slowly. Unless the data set is small or a SAR model is absolutely essential, we recommend using a CAR function for the spatial regression models.

Type of Test Procedure

The fifth, and last model decision, is whether to run a fixed model or a backward elimination *stepwise* procedure (only with the normal or MLE models). A fixed model includes all selected independent variables in the regression whereas a backward elimination model starts with all selected variables in the model but proceeds to drop variables that fail the P-to-remove test, one at a time. Any variable that has a significance level in excess of the P-to-remove value is dropped from the equation.

If the fixed model is chosen, then all independent variables will be regressed simultaneously. However, if the stepwise backward elimination procedure is selected, the user must define a *p-to-remove* value. The choices are: 0.1, 0.05, 0.01, and 0.001. The default is 0.01. Traditionally, 0.05 is used as a minimal threshold for significance. We put in 0.01 as the default to make the model stricter; with the large datasets that typically occur in police departments, the less strict 0.05 criterion would not exclude many independent variables. But, the user can certainly use 0.05 instead.

MCMC Choices

If the user chooses the MCMC algorithm, then nine *additional* decisions have to be made.

Number of Iterations

The first MCMC decision is the number of iterations to be run. The default is 25,000. The number should be sufficient to produce reliable estimates of the parameters. Check the MC Error/Standard deviation ratio and the G-R statistic to be sure these are below 1.05 and 1.20 respectively.

'Burn in' Iterations

The second MCMC decision is the number of initial iterations that will be dropped from the final distribution (the 'burn in' period). The default is 5,000. The number of 'burn in' iterations should be sufficient for the algorithm to reach an equilibrium state and produce reliable estimates of the parameters. Check the MC Error/Standard deviation ratio and the G-R statistic to be sure these are below 1.05 and 1.20 respectively.

Block Sampling Threshold

The third MCMC decision is whether to run all the records through the MCMC algorithm or whether to draw block samples. This is called the *Block Sampling Threshold*. The algorithm will be run on all cases unless the number of records exceeds the number specified in the block sampling threshold. The default threshold is 6,000 cases. If the number of cases exceeds the threshold, then the block sampling method is used (see below).

Note that if you raise the run the block sampling threshold for more cases, calculating time will increase substantially. For the non-spatial Poisson-Gamma model, the increase is linear. However, for the spatial Poisson-Gamma model, the increase is exponential. Further, we have found that we cannot calculate the spatial model for more than about 6,000 cases. In short, the block sampling method must be used for spatial models with a large number of cases.

Average Block Size

The fourth MCMC decision is the number of cases to be drawn in each block sample if the total number of records is greater than the block sampling threshold. The default is 400 cases. Note that this is an average. Actual samples will vary in size. The output will display the expected sample size and the average sample size that was drawn.

Number of Samples Drawn

The fifth MCMC decision is the number of samples to be drawn if the total number of records is greater than the block sampling threshold. The default is 25. We have found that

reliable estimates can be obtained from 20 to 30 samples especially if the sequence converges quickly and even 10 samples can produce meaningful results. Obviously, the more samples that are drawn, the more reliable will be the final results. But, having more samples will not necessarily increase the precision beyond 30.

Calculate Intercept

The sixth MCMC decision is whether to run the model with or without an intercept (constant). The default is with an intercept estimated. To run the model without the intercept, uncheck the 'Calculate intercept' box.

Spatial Autocorrelation Estimate

The seventh MCMC decision is whether to run a spatial autocorrelation model. There are two alternative spatial autocorrelation functions that can be used, a *Conditional Autoregressive* (or CAR) or a *Simultaneous Autoregressive* (or SAR). These were defined in Chapter 19. The default is no spatial autocorrelation. Note that estimating the SAR function takes a long time, much longer than for the CAR model. Unless there is a reason for using the SAR, we recommend using the CAR for any spatial autocorrelation component.

Calculate Exposure/Offset

The eighth MCMC decision is whether to run a risk model. If the model is a risk or rate model, then an exposure (offset) variable needs to be defined. Check the 'Calculate exposure/offset' box and identify the variable that will be used as the exposure variable. The coefficient for this variable will automatically be 1.0.

Advanced Options

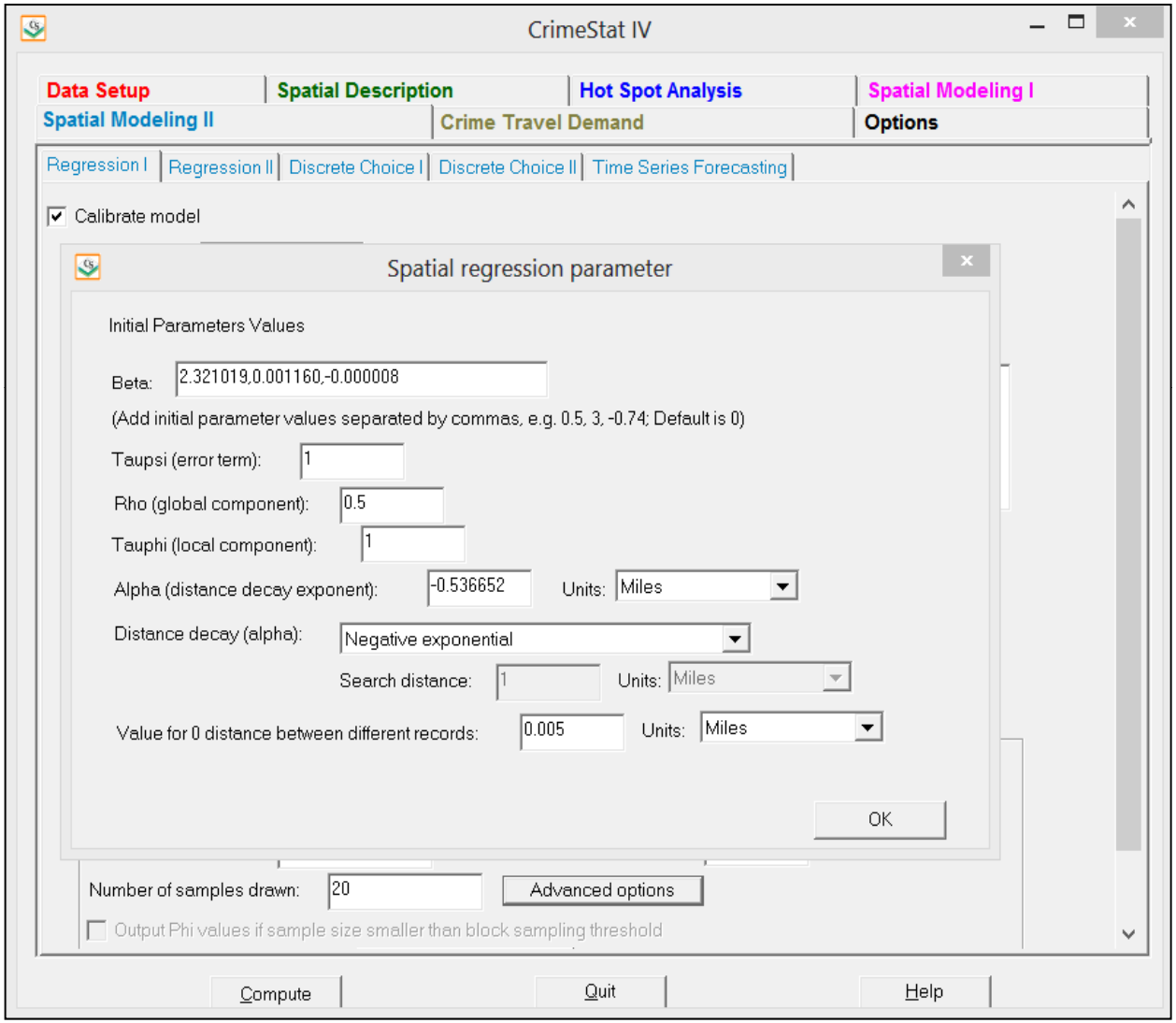
There is also a set of advanced options for the MCMC algorithm. Figure 20.2 displays the advanced options dialogue. We would suggest keeping the default values initially until you become very familiar with the routine.

Initial parameters values for Phi (φ)

The ninth, and last, MCMC decision is the prior values used for the different parameters being estimated. The MCMC algorithm requires an initial estimate for each parameter. There are default values that are used. For the beta coefficients (including the intercept), the default values are 0. This assumes that the coefficient is 'not significant' and has a large variance. It is frequently called a 'non-informative' prior. These are displayed as a blank screen for the Beta

Figure 20.2:

Advanced Options for MCMC Poisson-Gamma-CAR Model



box. However, estimates of the beta coefficients can be substituted for the assumed 0 coefficients. To do this, all independent variable coefficients plus the intercept (if used) must be listed in the order in which they appear in the model and must be separated by commas. Do not include the beta coefficients for the spatial autocorrelation, Φ_i, term (if used).

For example, suppose there are three independent variables. Thus, the model will have four coefficients (the intercept and the coefficients for each of three independent variables). Suppose a prior study had been done in which a Poisson-Gamma model was estimated as:

$$Y_i = e^{4.5+0.3X_{1i}-2.1X_{2i}+3.4X_{3i}} \quad (20.1)$$

The researcher wants to repeat this model but with a different data set and assumes that the model using the new data set will have coefficients similar to the earlier research. Thus, the following would be specified in the box for the betas under the advanced options:

4.5, 0.3, -2.1, 3.4 (20.2)

The routine will use these values for the initial estimates of the parameters before starting the MCMC process (with or without the block sampling method). The advantage is that the distribution will converge more quickly (assuming the model is appropriate for the new data set).

Rho (ρ) *and Tauphi* (τ_ϕ)

The spatial autocorrelation component, Φ, is made up of three separate sub-components, called Rho (ρ), Tauphi (τ_ϕ), and Alpha (α, see formula 19.5 in chapter 19). These are additive.

Rho is roughly a global component that applies to the entire data set. Tauphi is roughly a neighborhood component that applies to a sub-set of the data. Alpha is essentially a localized effect. The routine works by estimating values for Rho and Tauphi but uses a pre-defined value for Alpha. The default initial values for Rho and Tauphi are 0.5 and 1 respectively. The user can substitute alternative values for these parameters.

Alpha (α)

Alpha (α) is the exponent for the distance decay function in the spatial model. Essentially, the distance decay function defines the weight to be applied to the values of nearby records. The weight can be defined by one of three mathematical functions. First, the weight can be defined by a negative exponential function where:

$$Weight = e^{-\alpha d_{ij}} \quad (20.3)$$

where d_{ij} is the distance between observations and α is the value for alpha. It is automatically assumed that alpha will be negative whether the user puts in a minus sign or not. The user inputs the alpha value in this box.

Second, the weight can be defined by a restricted negative exponential whereby the negative exponential operates up to the specified search distance, whereupon the weight becomes 0 for greater distances:

Up to Search distance: $Weight = e^{-\alpha d_{ij}}$ for $d_{ij} \geq 0, d_{ij} \leq d_p$ (20.4)

Beyond search distance: 0 for $d_{ij} > d_p$ (20.5)

where d_p is the search distance. The coefficient for the linear component is assumed to be 1.0.

Third, the weight can be defined as a uniform value for all other observations within a specified search distance. This is a *contiguity* (or adjacency) measure. Essentially, all other observations have an equal weight within the search distance and 0 if they are greater than the search distance. The user inputs the search distance and units in this box.

For the negative exponential and restricted negative exponential functions, substitute the selected value for α in the alpha box.

Diagnostic test for reasonable alpha (α) value

The default function for the weight is a negative exponential with a default alpha value of -1 in miles. For many data sets, this will be a reasonable value. However, for other data sets, it will not.

Reasonable values for alpha with the negative exponential function are obtained with the following procedure:

1. Decide on the measurement units to be used to calculate alpha (miles, kilometers, feet, etc). The default is miles. *CrimeStat* will convert from the units defined for the Primary File input dataset to those specified by the user.

2. Calculate the nearest neighbor distance from the Nna routine on the Distance Analysis I page. These may have to be converted into units that were selected in step 1 above. For example, if the Nearest Neighbor distance is listed as 2000 feet, but the desired units for alpha are miles, convert 2000 feet to miles by dividing the 2000 by 5280.

3. Input the dependent variable as the Z (intensity) variable on the Primary File page.

4. Run the Moran Correlogram routine on this variable on the Spatial Autocorrelation page (under Spatial Description). By looking at the values and the graph, decide whether the distance decay in this variable is very 'sharp' (drops off quickly) or very 'shallow' (drops off slowly).

5. Define the appropriate weight for the nearest neighbor distance:

    a. Assume that the weight for an observation with itself (i.e., distance = 0) is 1.0.

    b. If the distance decay drops off sharply, then a low weight for nearby values should be given. Assume that any observations at the nearest neighbor distance will only have a weight of 0.5 with observations further away being even lower.

    c. If the distance decay drops off more slowly, then a higher weight for nearby values should be given. Assume that any observations at the nearest neighbor distance will have a weight of 0.9 with observations further away being lower but only slightly so.

    d. An intermediate value for the weight is to assume it to be 0.75.

6. A range of alpha values can be solved using these scenarios:

    a. For a sharp decay, alpha is given by:

$$\alpha = \frac{Ln(0.5)}{NN_{distance}} \tag{20.6}$$

    b. For a shallow distance decay, alpha is given by:

$$\alpha = \frac{Ln(0.9)}{NN_{distance}} \tag{20.7}$$

    c. For an intermediate decay, alpha is given by:

$$\alpha = \frac{Ln(0.75)}{NN_{distance}} \tag{20.8}$$

In all three equations, $NN_{distance}$ is the nearest neighbor distance.

These calculations will provide a range of appropriate values for α. The diagnostics routine automatically estimates these values as part of its output.

Value for 0 distance between records

The advanced options dialogue has a parameter for the minimum distance to be assumed between different records. If two records have the same X and Y coordinates (which could happen if the data are individual events, for example), then the distance between these records will be 0. This could cause unusual calculations in estimating spatial effects. Instead, it is more reliable to assume a slight difference in distance between all records. The default is 0.005 miles but the user can modify this (including substituting 0 for the minimal distance).

Output

The output depends on whether an MLE or an MCMC model has been run.

Maximum Likelihood (MLE) Model Output

The MLE routines (Normal, Poisson, Poisson with linear correction, MLE Poisson-Gamma, Binomial Probit, and MLE Binomial Logit/Logistic) produce a standard output which includes summary statistics and estimates for the individual coefficients.

MLE Summary Statistics

The summary statistics include:

Information About the Model

1. The data file
2. The dependent variable
3. The number of cases
4. The degrees of freedom (N – number of parameters estimated)
5. The type of regression model (Normal/OLS, Poisson, Poisson with linear correction, Poisson-Gamma, Binomial Logit)
6. The method of estimation (MLE)

Likelihood Statistics

7. Log-likelihood estimate, which is a negative number. For a set number of independent variables, the more negative the log-likelihood the better.

8. Log-likelihood per case. This divides the log-likelihood by the sample size (N). This indicates the average contribution to the log-likelihood of each observation. The more negative, the better.
9. Akaike Information Criterion (AIC) adjusts the log-likelihood for the degrees of freedom. The smaller the AIC, the better.
10. AIC per case. This divides the AIC statistic by the sample size (N). This indicates the average contribution to the AIC of each observation. The smaller, the better.
11. Bayesian Information Criterion (BIC), sometimes known as the Schwartz Criterion (SC), adjusts the log-likelihood for the degrees of freedom. The smaller the BIC, the better.
12. BIC per case. This divides the BIC/SC statistic by the sample size (N). This indicates the average contribution to the BIC/SC of each observation. The smaller, the better.
13. Deviance compares the log-likelihood of the model to the log-likelihood of a model that fits the data perfectly. A smaller deviance is better.
14. The probability value of the deviance based on a Chi-square test with *N-K-1* degrees of freedom where *K* is the number of independent variables.
15. Pearson Chi-square is a test of how closely the predicted model fits the data. A smaller Chi-square is better since it indicates the model fits the data well.
16. The probability value of the Pearson Chi-square based on a Chi-square test with *N-K-1* degrees of freedom where *K* is the number of independent variables.

Model Error Estimates

17. Mean Absolute Deviation (MAD). For a set number of independent variables, a smaller MAD is better.
18. Quartiles for the Mean Absolute Deviation. For any one quartile, smaller is better.
19. Mean Squared Predictive Error (MSPE). For a set number of independent variables, a smaller MSPE is better.
20. Quartiles for the Mean Squared Predictive Error. For any one quartile, smaller is better.
21. Squared multiple R (for linear model only). This is the percentage of the dependent variable accounted for by the independent variables.
22. Adjusted squared multiple R (for linear model only). This is the squared multiple R adjusted for degrees of freedom.

Dispersion Tests

23. Adjusted deviance. This is a measure of the difference between the observed and predicted values (the residual error) adjusted for degrees of freedom. The smaller the adjusted deviance, the better. A value greater than 1 indicates over-dispersion.
24. Probability of adjusted deviance. This is the Pearson Chi-square test with 1 degree of freedom.
25. Adjusted Pearson Chi-square. This is the Pearson Chi-square adjusted for degrees of freedom. The smaller the Pearson Chi-square, the better. A value greater than 1 indicates over-dispersion.
26. Probability of adjusted Pearson Chi-square. This is the Pearson Chi-square test with 1 degree of freedom.
27. Dispersion multiplier. This is the ratio of the expected variance to the expected mean. For a set number of independent variables, the smaller the dispersion multiplier, the better. For example, in a pure Poisson distribution, the dispersion should be 1.0. In practice, a ratio greater than 10 indicates that there is too much variation that is unaccounted for in the model. Either add more variables or change the functional form of the model.
28. Z-test for dispersion multiplier (Poisson models only). This is a test for whether the dispersion parameter is significantly greater than that assumed by the Poisson model. It is a test of over-dispersion.
29. P-value for Z-test of dispersion parameter (Poisson models only). This is the one-tail probability level associated with the Z-test.
30. Inverse dispersion multiplier. For a set number of independent variables, a larger inverse dispersion multiplier is better. A ratio close to 1.0 is considered good.

MLE Individual Coefficient Statistics

For the individual coefficients, the following are output:

31. The coefficient. This is the estimated value of the coefficient from the maximum likelihood estimate.
32. Standard Error. This is the estimated standard error from the maximum likelihood estimate.
33. Pseudo-tolerance. This is the tolerance value based on a normal prediction of the variable by the other independent variables.
34. Z-value. This is asymptotic Z-test that is defined based on the coefficient and standard error. It is defined as Coefficient/Standard Error.
35. p-value. This is the two-tail probability level associated with the Z-test.

Table 20.1 show the output for an MLE Poisson-Gamma model that relates the number of Houston 2007-09 burglaries to the number of 2008 households and the 2000 median household income of Traffic Analysis Zones.

Table 20.1:
Maximum Likelihood Output for Poisson-Gamma Model

```
 Model result:
 Data file:                          Burglaries_within_City_of_Houston.dbf
 DepVar:                             BURG2006
 N:                                  1179
 Df:                                 1175
 Type of regression model:           Poisson-Gamma-no spatial autocorrelation
 Method of estimation:               MLE

      Likelihood statistics
 Log-likelihood:                     -4430.800180
 AIC:                                8869.600361
 BIC/SC:                             8889.890048
 Deviance:                           1390.149554 P-value of Deviance:   0.0001
 Pearson Chi-Square:                 1112.717355 P-value of Chi-Square: 0.0001

      Model error estimates
 Mean absolute deviation:            39.580568
      1st (highest) quartile:        124.121350
      2nd quartile:                  19.377810
      3rd quartile:                  6.195620
      4th (lowest) quartile:         8.940150
 Mean squared predicted error:       62031.156586
      1st (highest) quartile:        242037.095867
      2nd quartile:                  6445.778853
      3rd quartile:                  118.261739
      4th (lowest) quartile:         154.880457

      Dispersion tests
 Adjusted deviance:                  1.183106  P-value of Deviance:   n.s.
 Adjusted Pearson Chi-Square:        0.946993  P-value of Chi-Square: n.s.
 Dispersion multiplier:              1.534057  Z= 910.799548 P-value: 0.0001
 Inverse dispersion multiplier:      0.651866

------------------------------------------------------------------------------
                                          Pseudo-
 Predictor   DF  Coefficient  Stand Error   Tolerance   z-value      p-value
 INTERCEPT   1     2.321019     0.083077           .    27.938042    0.001
    HH2006   1     0.001160     0.000066     0.993563   17.661356    0.001
MEDHHINC00   1    -0.000008     0.000002     0.993563   -5.129752    0.001
------------------------------------------------------------------------------
```

Markov Chain Monte Carlo (MCMC) Model Output

The MCMC routines (Normal-CAR/SAR, Poisson-Gamma, Poisson-Gamma-CAR/SAR, Poisson-Lognormal, Poisson-Lognormal-CAR/SAR, Binomial Logit, Binomial Logit-CAR/SAR) produce a standard output and an optional expanded output**.** The standard output includes summary statistics and estimates for the individual coefficients. Background information on these models is found in chapters 16, 17, 18, and 19.

MCMC Summary Statistics

The summary statistics include:

Information About the Model

1. The dependent variable
2. The number of records
   The sample number. This is only output when the block sampling method is used.
3. The number of cases for the sample. This is only output when the block sampling method is used.
4. Date and time for sample. This is only output when the block sampling method is used
5. The degrees of freedom (N – number of parameters estimated)
6. The type of regression model (Normal/OLS, Poisson, Poisson with linear correction, Poisson-Gamma, Poisson-Gamma-CAR/SAR, Poisson-Lognormal, Poisson-Lognormal-CAR/SAR, Binomial Logit, Binomial Logit-CAR/SAR)
7. The method of estimation
8. The number of iterations
9. The 'burn in' period
10. The block size is the expected number of records selected for each block sample. The actual number may vary.
11. The number of samples drawn. This is output when the block sampling method used.
12. The average block size. This is output when the block sampling method used.
13. The type of distance decay function used. This is output for models that use CAR or SAR spatial autocorrelation functions.
14. Condition number for the distance matrix. If the condition number is large, then the model may not have properly converged. This is output for the Poisson-Gamma-CAR model only.

15. Condition number for the inverse distance matrix. If the condition number is large, then the model may not have properly converged. This is output for the Poisson-Gamma-CAR/SAR, or Poisson-Lognormal-CAR/SAR models only.

Likelihood Statistics

16. Log-likelihood estimate, which is a negative number. For a set number of independent variables, the smaller the log-likelihood (i.e., the most negative) the better.
17. Log-likelihood per case. This divides the log-likelihood by the sample size (N). This indicates the average contribution to the log-likelihood of each observation. The more negative, the better.
18. Deviance Information Criterion (DIC) for models only. This adjusts the log-likelihood for the effective degrees of freedom. The smaller the DIC, the better.
19. Akaike Information Criterion (AIC) adjusts the log-likelihood for the degrees of freedom. The smaller the AIC, the better.
20. AIC per case. This divides the AIC statistic by the sample size (N). This indicates the average contribution to the AIC of each observation. The smaller, the better.
21. Bayesian Information Criterion (BIC), sometimes known as the Schwartz Criterion (SC), adjusts the log-likelihood for the degrees of freedom. The smaller the BIC, the better.
22. BIC per case. This divides the BIC/SC statistic by the sample size (N). This indicates the average contribution to the BIC/SC of each observation. The smaller, the better.
23. Deviance compares the log-likelihood of the model to the log-likelihood of a model that fits the data perfectly. A smaller deviance is better.
24. The probability value of the deviance based on a Chi-square test with *N-K-1* degrees of freedom where *K* is the number of independent variables.
25. Pearson Chi-square is a test of how closely the predicted model fits the data. A smaller Chi-square is better since it indicates the model fits the data well.
26. The probability value of the Pearson Chi-square based on a Chi-square test with *N-K-1* degrees of freedom where *K* is the number of independent variables.

Model Error Estimates

27. Mean Absolute Deviation (MAD). For a set number of independent variables, a smaller MAD is better.
28. Quartiles for the Mean Absolute Deviation. For any one quartile, smaller is better.

29. Mean Squared Predictive Error (MSPE). For a set number of independent variables, a smaller MSPE is better.
30. Quartiles for the Mean Squared Predictive Error. For any one quartile, smaller is better.

Dispersion Tests

31. Adjusted deviance. This is a measure of the difference between the observed and predicted values (the residual error) adjusted for degrees of freedom. The smaller the adjusted deviance, the better. A value greater than 1 indicates over-dispersion.
32. The probability value of the adjusted deviance based on a Chi-square test with 1 degree of freedom.
33. Adjusted Pearson Chi-square. This is the Pearson Chi-square adjusted for degrees of freedom. The smaller the Pearson Chi-square, the better. A value greater than 1 indicates over-dispersion.
34. The probability value of the adjusted Pearson Chi-square based on a Chi-square test with 1 degree of freedom.
35. Dispersion multiplier. This is the ratio of the expected variance to the expected mean. For a set number of independent variables, the smaller the dispersion multiplier, the better. In a pure Poisson distribution, the dispersion should be 1.0. In practice, a ratio greater than 10 indicates that there is too much variation that is unaccounted for in the model. Either add more variables or change the functional form of the model.
36. Inverse dispersion multiplier. For a set number of independent variables, a larger inverse dispersion multiplier is better. A ratio close to 1.0 is considered good.

MCMC Individual Coefficient Statistics

For the individual coefficients, the following are output:

37. The mean coefficient. This is the mean parameter value for the *N-K* iterations where *k* is the 'burn in' samples that are discarded. With the MCMC block sampling method, this is the mean of the mean coefficients for all block samples.
38. The standard deviation of the coefficient. This is an estimate of the standard error of the parameter for the *N-K* iterations where *k* is the 'burn in' samples that are discarded. With the MCMC block sampling method, this is the mean of the standard deviations for all block samples.
39. t-value. This is the t-value based on the mean coefficient and the standard deviation. It is defined by Mean/Std.

40. p-value. This is the two-tail probability level associated with the t-test.
41. Adjusted standard error (Adj. Std). The block sampling method will produce substantial variation in the mean standard deviation, which is used to estimate the standard error. Consequently, the standard error will be too large. An approximation is made by multiplying the estimated standard deviation by $\sqrt{\frac{\bar{n}}{N}}$ where $\bar{n}$ is the average sample size of the block samples and *N* is the number of records. If no block samples are taken, then this statistic is not calculated.
42. Adjusted t-value. This is the t-value based on the mean coefficient and the adjusted standard deviation. It is defined by Mean/Adj_Std. If no block samples are taken, then this statistic is not calculated.
43. Adjusted p-value. This is the two-tail probability level associated with the adjusted t-value. If no block samples are taken, then this statistic is not calculated.
44. MC error is a Monte Carlo simulation error. It is a comparison of the means of *m* individual chains relative to the mean of the entire chain. By itself, it has little meaning.
45. MC error/Std is the MC error divided by the standard deviation. If this ratio is less than .05, then it is a good indicator that the posterior distribution has converged.
46. G-R stat is the Gelman-Rubin statistic which compares the variance of *m* individual chains relative to the variance of the entire chain. If the G-R statistic is under 1.2, then the posterior distribution is commonly considered to have converged.
47. Spatial autocorrelation term (Phi, ϕ) for CAR/SAR models only. This is the estimate of the fixed effect spatial autocorrelation effect. It is made up of three components: a global component (Rho, ρ); a local component (Tauphi, τ_φ); and a local neighborhood component (Alpha, α, which is defined by the user).
48. The log of the error in the model (Taupsi). This is an estimate of the unexplained variance remaining. Taupsi is the exponent of the dispersion multiplier, $e^{\tau\psi}$. For any fixed number of independent variables, the smaller the Taupsi, the better.

Expanded Output (MCMC Only)

If the expanded output box is checked, additional information on the percentiles from the MCMC sample are displayed. If the block sampling method is used, the percentiles are the means of all block samples. The percentiles are:

49. 2.5^{th} percentile
50. 5^{th} percentile
51. 10^{th} percentile

52. 25th percentile
53. 50th percentile (median)
54. 75th percentile
55. 90th percentile
56. 95th percentile
57. 97.5th percentile

The percentiles can be used to construct confidence intervals around the mean estimates or to provide a non-parametric estimate of significance as an alternative to the estimated t-value in the standard output. For example, the 2.5th and 97.5th percentiles provide approximate 95 percent confidence intervals around the mean coefficient while the 0.5th and 99.5th percentiles provide approximate 99 percent confidence intervals.

The percentiles will be output for all estimated parameters including the intercept, each individual predictor variable, the spatial effects variable (Phi), the estimated components of the spatial effects (Rho and Tauphi), and the overall error term (Taupsi).

Table 20.2 show selective output from an MCMC Poisson-Lognormal-CAR spatial model that relates the number of Houston 2007-09 burglaries to the number of 2008 households and the 2000 median household income of Traffic Analysis Zones. The percentiles have been reduced to 0.5th, 2.5th, 97.5th, and 99.5th to fit the table.

Output Phi Values (CAR/SAR Models Only)

For the CAR and SAR models only, the individual Phi values can be output. This will occur if the sample size is smaller than the block sampling threshold. Check the 'Output Phi value if sample size smaller than block sampling threshold' box. An ID variable must be identified and a DBF output file defined.

Save Output

The predicted values and the residual errors can be output to a 'dbf' file with a REGOUT<*root name*> file name where *rootname* is the name specified by the user. The output is saved under a different file name. The output includes all the variables in the input data set plus two new ones: 1) the predicted values of the dependent variable for each observation (with the field name PREDICTED); and 2) the residual error values, representing the difference between the actual /observed values for each observation and the predicted values (with the field name RESIDUAL). The file can be imported into a spreadsheet or graphics program and the errors plotted against the predicted dependent variable (similar to Figure 15.3 in chapter 15).

Table 20.2:
MCMC Output for Poisson-Lognormal-CAR Model

```
DepVar:                          BURG2006
 N:                               1179
 Df:                              1174
 Number of iterations:            25000
 Type of regression model:        Poisson-Lognormal-CAR
 Method of estimation:            MCMC
 Distance decay function:         Negative exponential

      Likelihood statistics
 Log-likelihood:                  -6087.822981
      Per case:                   -5.163548
 DIC:                             30510.458212
 AIC:                             12185.645963
      Per case:                   6.246823
 BIC/SC:                          7390.366951
      Per case:                   6.268335
 Deviance:                        414.787381     P-value of Deviance:   0.0001
 Pearson Chi-Square:              422.236291     P-value of Chi-Square: 0.0001

      Model error estimates
 Mean absolute deviation:         5.387914
      1st (highest) quartile:     14.262519
      2nd quartile:               5.504652
      3rd quartile:               1.340483
      4th (lowest) quartile:      0.493941
 Mean squared predicted error:    149.000118
      1st (highest) quartile:     542.211088
      2nd quartile:               51.172821
      3rd quartile:               3.835512
      4th (lowest) quartile:      0.298416

      Dispersion tests
 Adjusted deviance:               4.456926     P-value of Deviance:      0.0001
 Adjusted Pearson Chi-Square:     20.611149    P-value of Chi-Square:    0.0001
 Dispersion multiplier:           0.904852     Z = 133.050700            P-value of Z: 0.0001
 Inverse dispersion multiplier:   1.105154
```

	Mean	Std	t-value	p-value	MC error	MC error/ std	G-R stat
Intercept:	0.057768	0.086334	0.669124	n.s.	0.001960	0.022698	1.004705
HH2006:	0.000156	0.000064	2.448304	0.02	2.7333e-006	0.042825	1.018906
MEDHHINC00:	-5.7411e-008	1.5194e-006	-0.037785	n.s.	2.7607e-008	0.018169	1.001817
Spatial autocorrelation (Phi):	1.660699	0.063369	26.206660	0.001	0.003377	0.053283	1.026494
Global component (Rho)	0.178264	0.142500	1.250969	n.s.	0.001356	0.009515	1.000174
Local component (Tauphi):	0.003762	0.000404	9.317862	0.001	0.000018	0.043832	1.019646
Neighborhood component (Alpha: defined)	-0.636652 Miles						

Table 20.2 (continued)

Percentiles	0.5th	2.5th	97.5th	99.5th
Intercept:	-0.153012	-0.103269	0.236579	0.300385
HH2006:	0.000008	0.000041	0.000289	0.000339
MEDHHINC00:	-0.000004	-0.000003	0.000003	0.000004
Spatial component				
(Phi):	1.486406	1.530011	1.776679	1.804173
Global component				
(Rho):	0.001125	0.005925	0.525452	0.657165
Local component				
(Tauphi):	0.002892	0.003060	0.004649	0.005008

Save Estimated Coefficients

The individual coefficients can be output to a DBF file with a REGCOEFF<*root name*> file name where *rootname* is the name specified by the user. This file can be used in the 'Make Prediction' routine under Regression II.

Diagnostics Relevant for Spatial Regression

In chapter 15, the diagnostic tests for the regression module were described. Among the statistics produced by the routine are two relevant for spatial regression.

Testing for Spatial Autocorrelation in the Dependent Variable

First, there is the Moran's "I" test for spatial autocorrelation. The statistic was discussed extensively in Chapter 5. If the "I" is significant, *CrimeStat* outputs a message indicating that there is definite spatial autocorrelation in the dependent variable and that it needs to be accounted for, either by a proxy variable or by estimating a CAR or SAR model.

A *proxy* variable would be one that can capture a substantial amount of the primary reason for the spatial autocorrelation. One such variable that we have found to be very useful is the distance of the location from the metropolitan center (e.g., downtown). Almost always, population densities are much higher in the central city than in the suburbs, and this differential in density applies to most phenomena including crime (e.g., population density, employment density, traffic density, events of all types). It represents a *first-order* spatial effect, which was discussed in Chapters 4 and 5, and is the result of other processes. Another proxy variable that can be used is income (e.g., median household income, median individual income) which tends to account for much clustering in an urban area. The problem with income as a proxy variable is that it is both causative (income determines spatial location) as well as a by-product of

population densities. The combination of both income and distance from the metropolitan center can capture most of the effect of spatial autocorrelation.

An alternative is to use the Poisson-Gamma-CAR model to filter out some of the spatial autocorrelation. As we discussed above, this is useful only when all obvious spatial effects have already been incorporated into the model. A significant spatial effect only means that the model cannot explain the additional clustering of the dependent variable.

Estimating the Value of Alpha (α) for CAR/SAR Models

Second, there is an estimate of a plausible value for the distance decay function alpha, α, in the CAR or SAR models. The way the estimate is produced was discussed above and is based on assigning a proportional weight for the distance associated with the nearest neighbor distance, the average distance from each observation to its nearest 'neighbor' (see chapter 6).

Three values of α are given in different distance units, one associated with a weight of 0.9 (a very steep distance decay, one associated with a weight of 0.75 (a moderate distance decay), and one associated with a weight of 0.5 (a shallow distance decay). Users should run the Moran Correlogram and examine the graph of the drop off in spatial autocorrelation to assess what type of decay function most likely exists. The user should choose an α value that best represents the distance decay and should define the distance units for it.

Regression II Module

The Regression II module allows the user to apply a model to another dataset and make a prediction. Figure 20.3 show the Regression II setup page. The 'Make prediction' routine allows the application of coefficients to a dataset.

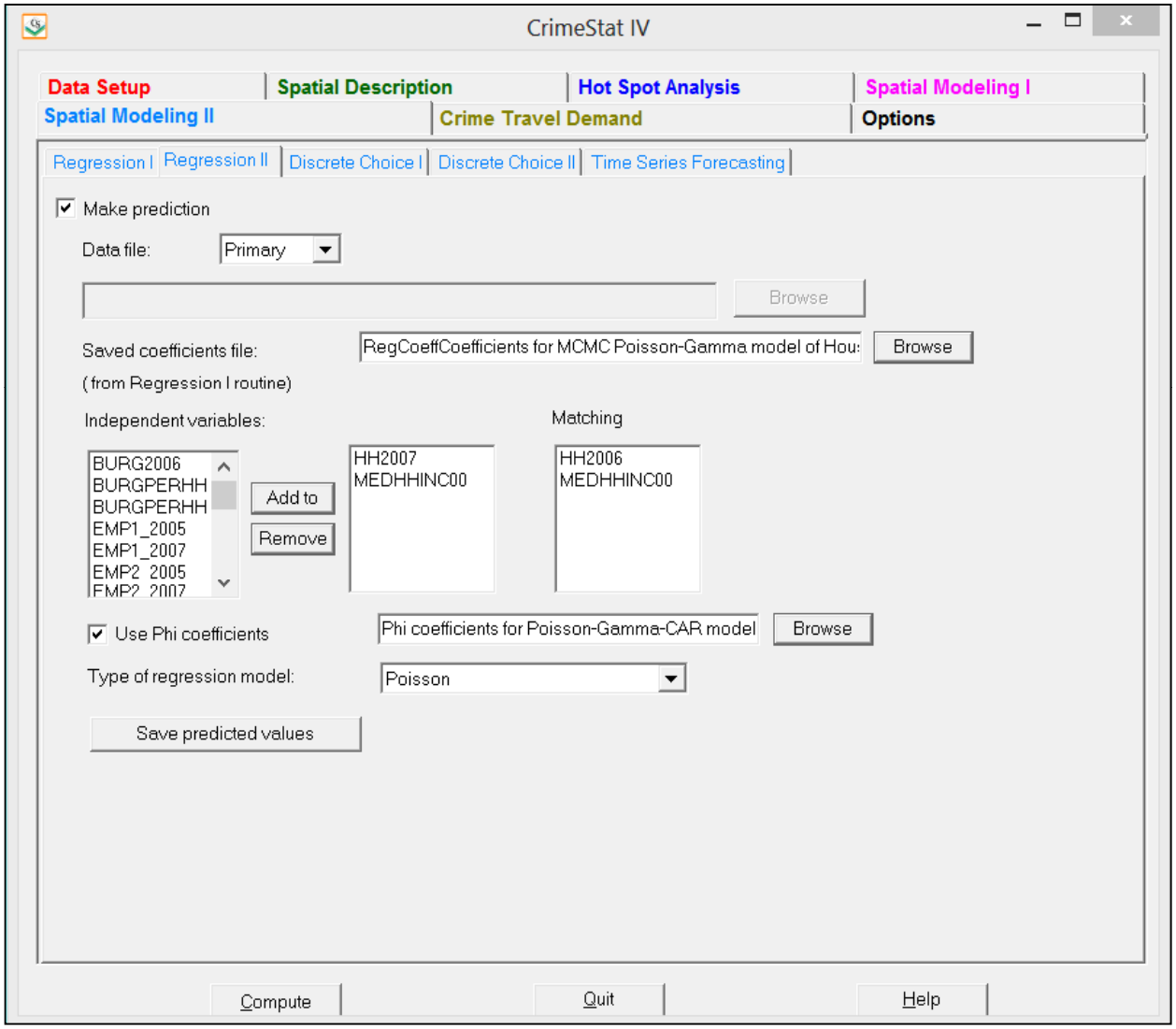
Note that, in this case, the coefficients are being applied to a different Primary File than that from which they were calculated. For example, a model might be calculated that predicts robberies for 2006. The saved coefficient file then is applied to another dataset, for example robberies for 2007.

There are four types of models that are fitted – normal, Poisson, binomial logit, and binomial probit. For the normal model, the routine fits the equation:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \cdots + \beta_k X_{ki} \qquad (20.9)$$

Figure 20.3:

Regression Modeling II Setup Screen



For the Poisson model, the routine fits the equation:

$$Y_i = e^{\beta_0+\beta_1 X_{1i}+\cdots+\beta_K X_{ki}+[\Phi_i]} \tag{20.10}$$

with β_0 being the intercept (if calculated), $\beta_1 \ldots \beta_k$ being the saved coefficients and Φ_i is the saved Phi values (if a CAR or SAR model was estimated). Notice that there is no error in each equation. Error was part of the estimation model. What were saved were only the coefficients.

For the binomial logit model, the routine fits the equation:

$$P(Y=1) = \frac{e^{\beta_0+\sum_1^K \beta_K X_K+[\Phi_i]}}{1+e^{\beta_0+\sum_1^K \beta_K X_K+[\Phi_i]}} = \frac{1}{1+e^{-(\beta_0+\sum_1^K \beta_I X_K+[\Phi_i])}} \tag{20.11}$$

with β_0 being the intercept (if calculated), $\beta_1 \ldots \beta_k$ being the saved coefficients and Φ_i is the saved Phi values (if a CAR or SAR model was estimated).

For the binomial probit model, the routine fits the equation:

$$p(Y=1) = \Phi^{-1}(p_i) = \beta_0 + \sum_1^K \beta_K X_K \tag{20.12}$$

with β_0 being the intercept (if calculated), $\beta_1 \ldots \beta_k$ being the saved coefficients and Φ_i is the saved Phi values (if a CAR or SAR model was estimated), and Φ is the cumulative standard normal distribution,

For all four types of model, the coefficients file must include information on the intercept and each of the coefficients. The user reads in the saved coefficient file and matches the variables to those in the new dataset based on the order of the coefficients file.

If the model had estimated a general spatial effect from a CAR or SAR model, then the general Φ_i will have been saved with the coefficient files. If the model had estimated specific spatial effects from a CAR or SAR model, then the specific Φ_i values will have been saved in a separate Phi coefficients file. In the latter case, the user must read in the Phi (Φ_i) coefficients file along with the general coefficient file.

Table 20.3 shows the output for the first 20 cases from a prediction of the number of burglaries per zone based on the estimation model shown in Table 20.2 (Poisson-Lognormal-CAR). The output will include all variables in the input data set plus the Phi coefficient and the predicted values. The user can then calculate residuals by subtracting the predicted from the actual (observed) values of the dependent variable.

Table 20.3:
File Output from Poisson-Lognormal-CAR Prediction of Houston Burglaries

TAZ03	BURG2006	PHI	PREDICTED
532	19	0.633593	7.922792
534	2	-0.163279	7.030844
536	2	-0.223977	11.555803
530	107	1.323602	21.462356
537	19	0.259453	15.658255
522	55	1.537228	5.987060
538	11	0.335330	7.432503
516	10	0.364732	9.598958
481	0	-0.350902	8.693915
474	1	-0.161788	8.348133
482	7	0.009940	12.178501
496	2	-0.245535	17.342402
548	0	-1.199179	13.717904
475	4	-0.037407	8.166218
435	3	-0.425498	8.307698
476	1	-0.056756	8.897897
484	8	0.014615	16.133065
483	2	-0.066611	7.888521
477	1	-0.076599	8.166218
478	0	-0.050627	9.293352

Conclusion

This chapter has summarized the structure of the Regression I and Regression II modules and most of the options that are available. The help menu on the program will provide context-specific help on individual items. Note that if you are using Windows Vista, Windows 7 or Windows 8, you must download a utility from *Microsoft* that allows the help menu to be viewed from the program. See Chapter 1 (p. 1.17) for details.

Chapter 21:

Discrete Choice Modeling

Wim Bernasco
Netherlands Institute
For The Study of Crime
and Law Enforcement
Amsterdam
&
Department of Spatial Economics
VU University Amsterdam,
Netherlands

Richard Block
Loyola University
Chicago, IL

Table of Contents

Chapter 21:
Discrete Choice Modeling

Introduction

This chapter describes the discrete choice framework and the two most well-known models that are part of it: the Multinomial Logit (MNL) and the Conditional Logit (CL). These techniques require a solid background in statistics and especially regression modeling. A background in economics will also be beneficial, though not necessary. Analysts wishing to use these techniques, in particular the conditional logit model, would be advised to find an expert to work with in developing applications.

The MNL and CL are two closely related statistical regression models that can be used to analyze a discrete outcome variable as a function of a set of independent variables. Discrete variables are also known as nominal or categorical variables. They can take on a finite number of unordered, mutually exclusive values. Both the MNL and the CL are generalizations of the logit model, which is used to analyze binomial (two category) outcome variables and which was discussed in Chapter 18.

Gender is an example of a binomial variable (it is either male or female). The weapon used in a robbery (gun, knife, strong arm, or other weapon) is a multinomial variable. Other examples are the mode of transport used by a rapist (car, scooter, train, bus, bike, walking) or the neighborhood in which a burglary was committed (any one of the city's neighborhoods).

Although the MNL and CL models can be used for all analytical problems where the outcome variable is discrete (nominal, categorical), in a number of disciplines the models are used to study the way that people or organizations make *choices*. Many research questions in the social and behavioral sciences, including criminology, deal with understanding and predicting discrete choices (Bernasco & Block, 2009). Political scientists aim to understand why people vote and what makes them choose a particular party (Palfrey & Poole, 1987). The party vote is a discrete variable. Sociologists want to understand what makes people decide in favor of a particular education, occupation, or marriage partner (Jepsen & Jepsen, 2002). Schools, occupations and partners are discrete choices. In marketing research, understanding and predicting consumer choice is a central concern (McFadden, 1980). Most consumer choices are discrete, such as which brand and model of car to purchase, or in which restaurant to have lunch. Transportation modelers predict why commuters choose to travel by bus, train, car or bicycle (Train, 1980). Behavioral ecological models try to find out what influences an animal's choice of where to forage, rest, or reproduce (Krebs & Davies, 1993).

Choice is also a central concern in crime analysis. What criteria does a police officer use to arrest or not arrest a juvenile? How does a robber choose a specific victim or a particular location to commit a robbery (Bernasco & Block, 2009)? This question addresses criminal location choice, which formed the major impetus to include these models in CrimeStat.

Although the MNL and CL models are both discrete choice models and share the same underlying likelihood function, they are quite different in practice. The main difference between the MNL and the CL model lies in the assumed sources of variation in choice outcomes. The MNL model assumes that variation in the characteristics of decision makers (e.g., age) determines variation in choice outcomes, whereas the CL model assumes that variation in the characteristics of the alternatives themselves (e.g., presence of a bar) determines variation in the choice outcomes.

Aggregated spatial interaction or 'gravity' models had been applied to criminal location choice and crime trips by Smith (1976) and Rengert (1981). These models bear a strong similarity in form and function to the discrete spatial choice models discussed in this chapter, but they are aggregated models of the volume of crime trips between areas. The discrete spatial choice approach was introduced in the criminological literature by Bernasco and Nieuwbeerta (2005) and has subsequently been applied in other studies (Bernasco, 2006, 2010a, 2010b; Bernasco & Block, 2009; Bernasco & Kooistra, 2010; Clare, Fernandez, & Morgan, 2009). Bernasco (2007) demonstrates how the discrete choice model can be reversed to form a tool in geographic offender profiling.

Neither the MNL nor the CL models require that the outcome variable be interpreted as a choice. In fact, the models can be used to model the outcomes of any process that results a finite number of unordered possible outcomes. For example, one study proposed a five-category typology of homicides in terms of the geographical relation between victim residence, offender residence and homicide location (Tita & Griffiths, 2005). It then used the MNL model to study the effects of various interactional, motivational and situational characteristics of the homicides on the type of the homicide. In this study it would be difficult to interpret the outcome as a decision, but the multinomial model is nevertheless useful to describe the effects of the variables on the different outcomes. Besides spatial choice, the conditional logit model has not been used very often in research on crime. An exception is a study that investigated the causes of criminal vengeance in conflicts (Phillips, 2003).

In the remainder of this chapter, the MNL and CL models are discussed in detail. First we demonstrate how the discrete choice model (encompassing both MNL and CL) is derived from random utility theory, and show the differences between the MNL and the CL models. Next we illustrate the structure of the data necessary to estimate MNL and CL models and give examples of both models.

Discrete Choice Framework

The discrete choice framework was developed in the 1970's by McFadden (1973) and others working in the field of travel demand, and the first applications of discrete choice were in the study of travel mode choice (i.e., the choice between train, bus, car, or airplane). Later the model was also applied to the choice of a travel routes and travel destinations (Ben-Akiva & Lerman 1985). This book is probably the most accessible and complete reference work on discrete choice that focuses on the conditional logit and multinomial logit model. A more advanced and more technical reference work is Train (2009), which is freely available (http://elsa.berkeley.edu/~train/).

The discrete choice framework consists of a set of assumptions regarding four elements of a choice situation (Ben-Akiva & Bierlaire 1999):

1. Decision makers. The decision maker is the person or agent that makes a choice.

2. Alternatives. The decision maker must choose one alternative from the choice set, i.e. the set of available alternatives that are mutually exclusive and collectively include all possible choices.

3. Attributes. Alternatives have attributes that make them attractive to the decision maker. The decision maker evaluates the attractiveness of all alternatives. The decision makers themselves can also have attributes.

4. Decision rule. According to economic theory, the decision maker chooses the alternative that maximizes his/her (expected) utility (net gain, profits, satisfaction).

The discussion that follows is mathematically advanced. Readers who prefer to skip the mathematical description of the models may want to continue reading at the "Data structures" section on page 21.7. We follow the notation of Train (2009).

A decision maker, labeled n, must make a choice among J alternatives. Note that the word 'alternatives' is used for the items, actions or locations that can be chosen, and the word 'choice' is used for the decision of the decision maker in selecting one of these alternatives. By convention the complete set of available alternatives is referred to as the 'choice set', although 'set of alternatives' might better describe it.

Decision maker n obtains a level of utility (profits, satisfaction), U_{ni}, from alternative i if that alterative is chosen. The principle of utility maximization asserts that the decision maker

decides in favor of the alternative *i* if and only if the individual expects to derive more utility from alternative *i* than from any other available alternative. Thus, if the decision maker decides in favor of alternative *i*, then that person must expect to derive less utility from each of the other alternatives (the expression $\forall j \neq i$ means 'for all values of *j* such that *j* not equals *i*).

$$U_{ni} > U_{nj} \forall j \neq i \,. \tag{21.1}$$

The utilities are known by the decision maker, but not by the analyst. The analyst only observes the *J* alternatives, some attributes a_{ni} of the alternatives, some attributes d_n of the decision maker, and can specify a function *V*, often called *representative utility* or *systematic utility*, that links these observed attributes to the decision maker's utility:

$$V_{ni} = V(a_{ni}, d_n) \forall i \tag{21.2}$$

The analyst incompletely observes utility, so that generally $U_{ni} \neq V_{ni}$. The utility can be written as the sum of representative utility V_{ni} and a term μ_{ni} that captures the factors that determine utility but are not observed by the analyst, and that is treated as random.

$$U_{ni} = V_{ni} + \varepsilon_{ni} \tag{21.3}$$

The probability that decision maker *n* chooses alternative *i* is the probability that the utility associated with choosing *i* is greater than the utility associated with any other alternative in the choice set:

$$P_{ni} = \Pr(U_{ni} > U_{nj} \forall j \neq i) \tag{21.4}$$

$$P_{ni} = \Pr(V_{ni} + \varepsilon_{ni} > V_{nj} + \varepsilon_{nj} \forall j \neq i) \tag{21.5}$$

$$P_{ni} = \Pr(\varepsilon_{nj} - \varepsilon_{ni} < V_{ni} - V_{nj} \forall j \neq i) \tag{21.6}$$

This is the most general formulation of the discrete choice model, and any specific choice model that is consistent with random utility maximization can be derived from specific assumptions on the joint distribution of the unobserved utility term μ_{ni}. CrimeStat can estimate the two most basic models of this family, the multinomial logit model and the conditional logit model. There are many others, including for example nested logit, mixed logit, and multinomial probit. These are described in Train (2009).

Multinomial and Conditional Logit

If the unobserved random utility components μ_{ni} are independent and identically distributed according to an *extreme value distribution* (also referred to as a Gumbel distribution), the MNL model and the CL can be derived. Originally, the general form of both was labeled the *conditional logit model* (McFadden 1973). Today both models are usually simply referred to as 'multinomial logit model' or even 'logit model' in the discrete choice literature. CrimeStat distinguishes between the MNL and the CL models because despite their mathematical equivalence, they require a different organization of the data. In the general model that encompasses both the CL and the MNL, the choice probability, P_{ni}, the probability that decision maker *n* chooses alternative *i*, is given by:

$$P_{ni} = \frac{e^{V_{ni}}}{\sum_{j=1}^{J} e^{V_{nj}}} \tag{21.7}$$

For computational convenience, and because any function can be closely approximated by a linear function, representative utility V_{ni} is usually assumed to be linear in the parameters. The specification of observed utility V_{ni} is different in the MNL and the CL models. In the MNL model, V_{ni} depends on the characteristics of the decision maker while in the CL model, it depends on the characteristics of the alternatives.

Multinomial Logit Model

In the MNL model,

$$V_{ni} = \beta_i' \mathbf{X_n} = \sum_{k=1}^{K} \beta_{ki} X_{kn} \tag{21.8}$$

In equation 21.8, *K* is the number of predictor variables in the model, X_{kn} is the value of the k^{th} predictor variable for observational unit (e.g. decision maker) *n*, and β_{ki} is a parameter associated with the k^{th} predictor variable and alternative *i*. Thus, as can be seen from the *k*, *i* and *n* indexes, in the MNL model, there is a separate parameter β_i for every alternative *i* in the choice set per predictor variable (including a constant). Note that the variables X_{kn} vary only across the decision makers *n*, but not across the alternatives (they have no *i* subscript). Characteristics of the alternatives do not explicitly play a role in this model (implicitly they do, as we would expect the β_{ki} to depend on characteristics of the alternatives).

Conditional Logit Model

In the CL model,

$$V_{ni} = \beta' \mathbf{X_{ni}} = \sum_{k=1}^{K} \beta_k X_{kni} \tag{21.9}$$

where K is again the number of predictor variables in the model, X_{kni} is the value of the k^{th} predictor variable for observational unit (e.g. decision maker) *n* and alternative *i*, and a β_k is a parameter associated with the k^{th} predictor variable. Thus, as can be seen from the *k*, *i* and *n* indexes, in the CL model, there is only a single parameter for all alternatives in the choice set per predictor variable.

Note that the variables X_{kni} vary across the decision makers *n* and alternatives *i*. Essentially, going from equation 21.8 to equation 21.9, the *i* index (that references alternatives) moves from the parameter β to the predictor variable *X*, a manifestation of the fact that in the MNL model characteristics of alternatives are implicitly included in the estimated alternative-specific parameters, while in the CL model they are explicitly measured and their effects estimated in generic parameters.

Probabilities in the Multinomial and Conditional Logit Models

Substituting equation 21.8 into equation 21.7, the multinomial logit probability that decision maker *n* chooses alternative *i* is:

$$P_{ni} = \frac{e^{\sum_{k=1}^{K} \beta_{ki} X_{kn}}}{\sum_{j=1}^{J} e^{\sum_{k=1}^{K} \beta_{kj} X_{kn}}} \tag{21.10}$$

Note that in equation 21.10, the predictor variables vary across decision makers *n* but not across alternatives *i*. Analogously, substituting equation 21.9 into equation 21.7, the conditional logit model asserts that the probability that decision maker *n* chooses alternative *i* is:

$$P_{ni} = \frac{e^{\sum_{k=1}^{K} \beta_{k} X_{kni}}}{\sum_{j=1}^{J} e^{\sum_{k=1}^{K} \beta_{k} X_{knj}}} \tag{21.11}$$

Note that in equation 21.11, it is impossible to estimate effects of attributes of the decision maker that do not vary across alternatives (such as age or gender), because such variables (and their parameters) automatically cancel out of the equation; the characteristic of the decision maker cannot affect which alternative is chosen because those characteristics do not vary across the alternatives. This feature differentiates the conditional logit model from the multinomial logit model.

It is possible, though, to estimate the interaction of characteristics of decision makers and characteristics of alternatives by creating or measuring variables that vary across both alternatives and decision makers. Such variables must have the *n* and the *i* subscript so that they do not cancel out in the equation.

An example of a measured interaction is the experience that the decision maker has with each of the alternatives. A given decision maker has more experience with one alternative than others, and, therefore, is more or less likely to choose the alternative. Repeat and near repeat victimization may be examples of this. Another example of a measured interaction is the distance between decision makers and alternatives.

An example of a created interaction is the multiplication of a characteristic of decision makers (e.g. gender S_n) with a characteristic of alternatives (e.g. location L_i) resulting in SL_{ni} . The resulting variable varies across decision makers (as for a given alternative *i* its value is different for males and females) and across alternatives (because for a given decision maker *n* it varies across locations).

Data Structures

Although the same mathematical model underlies the MNL model and the CL model, the estimation of the CL model requires the data to be organized differently than the estimation of the MNL model. This section considers the data structures that hold the information that is required to estimate either model.

The MNL model applies to a $n \times k$ matrix (where *n* refers to cases and *k* refers to variables that vary across cases), while the CL model applies to a $n \times i \times k$ matrix, where *n* refers to cases, *i* refers to alternatives, and *k* refers to variables that vary across cases and across alternatives. The distinctions between these two data structures are explained below.

The Multinomial Logit Model

The MNL model is estimated on a data set that is similar to the data structure of most other regression models and many incident spreadsheets. Each row (record) represents an

observational unit *n* (a case, sometimes a decision maker) and each column represents a variable (a characteristic of the unit). The dependent variable is nominal and indicates which alternative from a set of alternatives was chosen. The variation in outcomes is explained by variation in the characteristics of the observational units (decision makers). Table 21.1 shows a simple example that describes the first 5 incidents of a larger data file. For each case we know the area where the offender lived (*Origin*), the area where the crime was committed (*Destination*), the offender's age (*Age*), the type of crime committed (*CrimeType*), and the time of day it was committed (*Time*). There is also a variable that uniquely identifies cases (*ID*).

The first record indicates that at 3AM (*Time*) a burglary (*CrimeType*) was committed in zone P (*Destination*) by an 18 year old (*Age*) offender who lived in zone P (*Origin*). Case 2 is a robbery committed at 7 PM in zone P by an offender aged 23 living in zone Q. The third record shows that someone aged 42 living in zone R purchased an illicit drug in zone S at 2pm.

Table 21.1:
Case file Describing 5 Incidents

ID	*Origin*	*Destination*	*Age*	*CrimeType*	*Time*
1	P	P	18	Burglary	3am
2	Q	P	23	Robbery	7pm
3	R	S	42	Illicit drug	2pm
4	R	Q	32	Robbery	1pm
5	S	R	19	Burglary	6am

In principle, any variable (except ID) in the case file can be analyzed as representing a choice outcome (an alternative being chosen) although for some variables a choice interpretation is more natural than for others. *Destination* represents the choice outcome of the decision of where to commit the crime, *CrimeType* would be the choice outcome of the decision which type of crime to commit, and *Time* would be the choice outcome of the decision of when to commit the crime. *Origin* could also be a choice outcome, the outcome of the decision of where to live. *Age* can be a seen as the outcome of the choice at what age to commit the offence.

If the decision to be analyzed is where to commit the offence, the first record in Table 21.1 indicates that the offender offended in zone P rather than in zones Q, R or S. If the decision to be analyzed is which type of crime to commit, the first record in Table 21.1 shows that the offender decided to commit a burglary rather than commit a robbery or purchase an illicit drug. If the decision to be analyzed is when to commit the offence, the first record in Table 21.1 indicates that the offender offended at 3am rather than at 1pm, 3pm or any other time of the day.

Let us assume that in Table 21.1, the outcome variable is *Destination* (i.e. the area in which the offender committed the crime). Note that we assume that each offender was able to choose any of the alternatives (zones P=1, Q=2, R=3, and S=4 for four alternatives), and also note that the data do not contain attributes of the alternatives (e.g. whether the areas are affluent, have mixed land use, etc.). A MNL model could be used to assess the relation between linear combinations of *Time* (T) and *Age* (A) with the choice of a *Destination* (D) zone. In this case, equation 21.12 becomes:

$$P_n(D=i) = \frac{e^{\beta_i + \beta_{Ti} T_n + \beta_{Ai} A_n}}{\sum_{j=1}^{4} e^{\beta_j + \beta_{Tj} T_n + \beta_{Aj} A_n}} \tag{21.12}$$

where $P_n(D=i)$ is the probability that in the in n^{th} case the *Destination* chosen is i, T_n is the *Time* of the n^{th} case, A_n is the *Age* of the n^{th} case, and $^2_{Ti}$ is the parameter that represents the effect of *Time* on the probability that *Destination i* is chosen, $^2_{Ai}$ is the effect of *Age* on the probability that *Destination i* is chosen, and ^2_i is an alternative-specific constant, representing the average attractiveness of alternative *i* in the sample. Note that if *Destination* has four categories, the multinomial logit model involves the following four categorical equations.

$$P_n(D=1) = \frac{e^{\beta_1 + \beta_{T1} T_n + \beta_{A1} A_n}}{\sum_{j=1}^{4} e^{\beta_j + \beta_{Tj} T_n + \beta_{Aj} A_n}} \tag{21.13}$$

$$P_n(D=2) = \frac{e^{\beta_2 + \beta_{T2} T_n + \beta_{A2} A_n}}{\sum_{j=1}^{4} e^{\beta_j + \beta_{Tj} T_n + \beta_{Aj} A_n}} \tag{21.14}$$

$$P_n(D=3) = \frac{e^{\beta_3 + \beta_{T3} T_n + \beta_{A3} A_n}}{\sum_{j=1}^{4} e^{\beta_j + \beta_{Tj} T_n + \beta_{Aj} A_n}} \tag{21.15}$$

$$P_n(D=4) = \frac{e^{\beta_4 + \beta_{T4} T_n + \beta_{A4} A_n}}{\sum_{j=1}^{4} e^{\beta_j + \beta_{Tj} T_n + \beta_{Aj} A_n}} \tag{21.16}$$

All four equations are linked by having the same denominator and by

$$P_n(D=1) + P_n(D=2) + P_n(D=3) + P_n(D=4) = 1 \tag{21.17}$$

Altogether, 12 parameters are estimated, 4 alternative-specific constants ($^2_1$, $^2_2$, $^2_3$, $^2_4$), 4 for the Time predictor variable ($^2_{T1}$, $^2_{T2}$, $^2_{T3}$, $^2_{T4}$), and 4 for the Age predictor variable ($^2_{A1}$, $^2_{A2}$,

$^{2}_{A3}$, $^{2}_{A4}$). However, because the effects apply to the differences between the alternatives, the parameters for one of the J alternatives must be fixed, and the remaining effects are expressed in relation to this fixed 'reference' alternative. Like other programs, CrimeStat fixes these parameters of the reference alternative to 0 (the user can choose which alternative to use as the reference alternative. By default the most frequent alternative is the reference alternative.

Example 1: Modeling Choice of Premises in Chicago Non-street Robberies with the Multinomial Logit Model

In 1997, there were 1,587 robbery incidents in Chicago that did not occur on the street, in which a specific type of premises was robbed, and for which at least one offender was arrested. In 1998, there were 1,441 such incidents. In this example, characteristics of offenders and incidents will be used to describe differences in the type of premises victimized. The statistics used to differentiate models will be explained and the premise pattern of robberies in 1998 will be predicted using the robbery patterns of 1997.

Figure 21.1 maps the premises type of non street robberies in 1997. In 1997, 48.6% of these robberies were residential, 11.3% were in parking lots and garages, 23.8% were commercial, 2.1% were at banks or currency exchanges, 5.5% were in schools and school yards, 5.1% were in parks, and 3.5% were in public transit or stations. Although parks are amenities, not premises, they will be subsumed under 'premises ' here.

Some areas of the city are nearly free of non-street robberies. Unsurprisingly, commercial, and bank robberies are concentrated on main streets. Residential robberies are widespread over large sections of the west and south sides. The remainder of this brief will look at crime and offender characteristics that differentiate residential robberies from each of the other premises types using the multinomial logit model in CrimeStat IV.

In Table 21.2, 1,587 non-street robberies in 1997 are analyzed using the multinomial logit model. Residential robberies are compared to 6 other robbery premises. In the summary section of the table, the log likelihood ratio (LLR), Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC/SC) are measures of the differences between a model that includes offender and crime characteristics and the naïve (null) hypothesis that only includes the frequency of the various premises types.

The best use for these statistics is in comparing models. The most negative log likelihood ratio, and the smallest positive AIC or BIC are best. Unlike the LLR, the AIC and BIC correct for the number of explanatory (independent) variables. This is important because a model with

Figure 21.1:

Distribution of Chicago Non Street Robberies in 1997

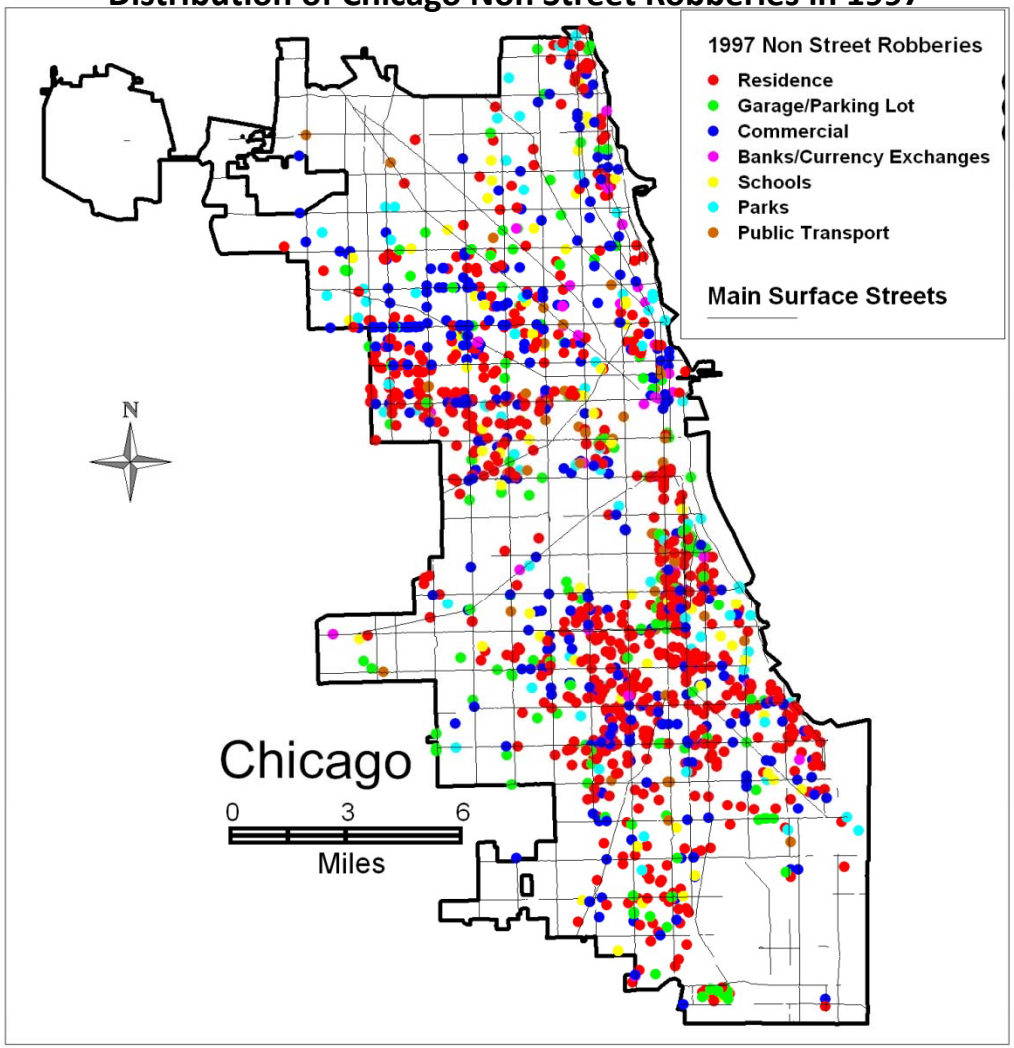


Table 21.2:
Multinomial Logit Model of Crime Premises:
Non-Street Robbery 1997

Model result:	
Data file:	1997 CHICAGO NON-STREET ROBBERIES
DepVar:	TYPE OF PREMISES
N:	1,587
Df:	1,580
Type of choice model:	Multinomial logit model
Number of Alternatives:	7
Method of estimation:	MLE
Likelihood statistics	
Log Likelihood:	-1,963.1
Per case:	-1.2
AIC:	3,996.3
Per case:	2.5
BIC/SC:	4,184.2
Per case:	2.6
Model error estimates	
Mean absolute deviation:	0.2
Mean squared predicted error:	0.1

REFERENCE CHOICE: 2 RESIDENTIAL

Predictor	Coefficient	Stand Error	t-value	p-value	Odds Ratio
3 GARAGES AND PARKING LOTS					
Alternative N=180					
Constant	-1.3156	0.250	-5.27	0.001	0.27
GUNCRIME	0.2768	0.187	1.48	n.s.	1.32
EVENING	0.4431	0.193	2.30	0.05	1.56
LATENIGHT	-0.3754	0.235	-1.60	n.s.	0.69
TRAVEL DIST	0.0059	0.001	4.43	0.001	1.01
OFFAGE	-0.0016	0.001	-1.88	n.s.	1.00
OFFBLACK	-0.4792	0.237	-2.02	0.05	0.62
4 COMMERCIAL					
Alternative N=378					
Constant	-0.6512	0.197	-3.31	0.001	0.52
GUNCRIME	1.3900	0.137	10.12	0.001	4.01
EVENING	-0.0614	0.163	-0.38	n.s.	0.94
LATENIGHT	-0.5360	0.180	-2.97	0.01	0.59
TRAVEL DIST	0.0049	0.001	4.23	0.001	1.00
OFFAGE	-0.0018	0.001	-2.66	0.01	1.00
OFFBLACK	-0.6909	0.186	-3.71	0.001	0.50

Table 21.2: (continued)

Predictor	Coefficient	Stand Error	t-value	p-value	Odds Ratio
5 BANKS AND CURRENCY EXCHANGES					
Alternative N=34					
Constant	-2.2444	0.411	-5.46	0.001	0.11
GUNCRIME	1.3043	0.362	3.60	0.001	3.69
EVENING	-1.5034	0.620	-2.43	0.05	0.22
LATENIGHT	-1.9124	0.742	-2.58	0.01	0.15
TRAVEL DIST	0.0083	0.002	3.51	0.001	1.01
OFFAGE	-0.0022	0.002	-1.09	n.s.	1.00
OFFBLACK	-1.4572	0.395	-3.69	0.001	0.24
6 SCHOOLS					
Alternative N=88					
Constant	6.6012	0.783	8.43	0.001	735.99
GUNCRIME	-2.0686	0.612	-3.38	0.001	0.13
EVENING	-2.1280	0.480	-4.44	0.001	0.12
LATENIGHT	-2.2797	0.750	-3.09	0.01	0.10
TRAVEL DIST	0.0065	0.003	2.30	0.05	1.01
OFFAGE	-0.3692	0.040	-9.17	0.001	0.69
OFFBLACK	-1.1614	0.382	-3.04	0.01	0.31
7 PARKS					
Alternative N=81					
Constant	3.1511	0.573	5.50	0.001	23.36
GUNCRIME	-0.6429	0.328	-1.96	0.05	0.53
EVENING	0.0558	0.282	0.20	n.s.	1.06
LATENIGHT	-1.1378	0.459	-2.48	0.05	0.32
TRAVEL DIST	0.0070	0.002	3.14	0.01	1.01
OFFAGE	-0.1966	0.025	-7.90	0.001	0.82
OFFBLACK	-1.2570	0.311	-4.05	0.001	0.28
8 PUBLIC TRANSPORT					
Alternative N=55					
Constant	-3.4423	0.617	-5.58	0.001	0.03
GUNCRIME	-0.6316	0.395	-1.60	n.s.	0.53
EVENING	-0.4296	0.409	-1.05	n.s.	0.65
LATENIGHT	-0.0134	0.337	-0.04	n.s.	0.99
TRAVEL DIST	0.0091	0.002	5.21	0.001	1.01
OFFAGE	-0.0010	0.001	-0.88	n.s.	1.00
OFFBLACK	0.6914	0.608	1.14	n.s.	2.00

Reference Alternative: 2 Residential

Multicollinearity statistics

Predictor	Pseudo-Tolerance
GUNCRIME	0.98
EVENING	0.93
LATENIGHT	0.93
TRAVEL DIST	0.99
OFFAGE	1.00
OFFBLACK	1.00

many explanatory variables is likely to have the most negative log likelihood ratio (LLR), but part of the size of the LLR results from the large number of variables used in the explanation.

In the second section of Table 21.2, each of the other types of premises is compared to the reference type (residential units), which is the type that is chosen most frequently. Note that the coefficients will differ for each of the alternatives. This is because the variables predicting each alternative are unique and will differ in their weights. For some alternatives, an independent variable may have a significant positive effect while for other alternatives it may have a significant negative effect.

For even other alternatives, the variable may not have a significant effect. For example, the use of a gun in a robbery (GUNCRIME) is positively associated with bank robberies but negatively associated with school robberies. For robberies in parks, the use of a gun is not related to the type of robbery. Note, also, that these are relative to the reference alternative, which in this case are residential robberies.

The numbers in the far right column, the Odds Ratios, are useful for substantive interpretation of the model. They indicate the odds increase or decrease associated with the variable that the robbery took place on the specific premise compared to the reference alternative (residential premises). They are measured as the relative change when the corresponding predictor variable increases by one unit. Odds ratios above 1 indicate that the odds increase as the predictor variable increases, odds ratios between 0 and 1 indicate they decrease as the predictor variable increases.

The p value indicates whether the odds ratio were likely to have occurred by chance if there was no relationship in the unit of the explanatory variable. For example, in the top panel on 'Garages and parking lots', the value of 1.56 indicates that-- if a robbery occurs in the evening (1) it is 1.56 times more likely (or, in other words, 56% more likely) to be in a garage or parking lot than at a residence. The p value indicates whether the odds ratio could have occurred by chance if there was no relationship. Thus, the p-value of 0.05 in the output demonstrates that the above odds ratio of 1.56 could have occurred by chance 5% of the time if there was no time-of-day difference in the probability that robberies take place in residences or in parking lots.

Commercial robberies are 4 times as likely to be committed with a gun than residential robberies, and a difference this large could occur only .1% of the time. Robberies occurring in and around schools are significantly different from residential robberies on all six explanatory variables. They are much less likely to involve guns or be committed by black offenders and are slightly further away from the offender's home. Unsurprisingly, they are all less likely to occur in the evening or late night and the offenders are younger than in residential robberies.

The coefficients column in section two are similar to those in any regression equation. Coefficients are created for each choice (here premises type) including the reference category. They are particularly useful for prediction of choice with a new data set(see below).

The third section indicates to what extent the explanatory variables vary together (multicollinearity). A pseudo tolerance below .90 indicates that this may be a problem in the model. If this is so, delete the variable with the lowest pseudo tolerance and run the model again. In this model all pseudo tolerances are above .9. Multicollinearity is not a problem.

Adding another variable to the 1997 model

Using the Log Likelihood, AIC, and BIC/SC statistics, it is possible to compare one decision making model to another. The decision making model in table 21.2 included six explanatory variables. Table 21.3 below adds the variable, number of offenders, to the model. Perhaps residential robbers are more likely to solo offenders than school yard robbers? However, adding the number of offenders to the model has little effect.

The more explanatory variables, the fewer degrees of freedom (df) and the more complex the model. The log likelihood decreases from -1963 to -1957 (more negative is better). The AIC declines slightly from 3996 to 3994, but the more comprehensive BIC/SC increases from 4184 to 4209 (closer to 0 is better for both). In other words, adding number of offenders to the model does not improve the differentiation of residential premises from other premises. For every premises type, the number of offenders is not significantly differentiated from residential robberies.

Predicting non-street Robberies in 1998 based on the 1997 model

Once a multinomial logit model is estimated, the parameter estimates can be used to predict a dependent variable in other data. The model developed in predicting the premises of robberies in 1997 can be used to predict the premises of robberies in 1998. This is done by saving the coefficients and applying them to the 1998 robbery data. The results can show how well the 1997 model estimated predicted 1998 robberies.

Table 21.3:
Multinomial Logit Model of Crime Premises:
Non-Street Robbery 1997
Number of Offenders Added

Model result:
 Data file: 1997 CHICAGO NON-STREET ROBBERIES
 DepVar: TYPE OF PREMISES
 N: 1,587
 Df: 1,579
 Type of choice model: Multinomial logit model
 Number of Alternatives: 7
 Method of estimation: MLE

 Likelihood statistics
 Log Likelihood: -1957.2
 AIC: 3,994.3
 BIC/SC: 4,209.1

 Model error estimates
 Mean absolute deviation: 0.2
 Mean squared predicted error: 0.1

REFERENCE CHOICE: 2 RESIDENTIAL

Predictor	Coefficient	Stand Error	t-value	p-value	Odds Ratio
3 GARAGES AND PARKING LOTS					
Alternative N=180					
Constant	-1.2044	0.302	-3.98	0.001	0.30
GUNCRIME	0.3039	0.191	1.59	n.s.	1.36
EVENING	0.4481	0.193	2.32	0.05	1.57
LATENIGHT	-0.3608	0.236	-1.53	n.s.	0.70
TRAVEL DIST	0.0059	0.001	4.46	0.001	1.01
OFFAGE	-0.0016	0.001	-1.90	n.s.	1.00
OFFBLACK	-0.4807	0.237	-2.02	0.050	0.62
NUM OFF	-0.1011	0.155	-0.65	n.s.	0.90
4 COMMERCIAL					
Alternative N=378					
Constant	-0.7871	0.221	-3.55	0.001	0.46
GUNCRIME	1.3501	0.140	9.61	0.001	3.86
EVENING	-0.0709	0.163	-0.43	n.s.	0.93
LATENIGHT	-0.5702	0.183	-3.12	0.01	0.57
TRAVEL DIST	0.0048	0.001	4.13	0.001	1.00
OFFAGE	-0.0017	0.001	-2.62	0.01	1.00
OFFBLACK	-0.6908	0.186	-3.71	0.001	0.50
NUM OFF	0.1261	0.093	1.36	n.s.	1.13

Table 21.3: (continued)

Predictor	Coefficient	Stand Error	t-value	p-value	Odds Ratio
5 BANKS AND CURRENCY EXCHANGES					
Alternative N=34					
Constant	-1.2864	0.715	-1.80	n.s.	0.28
GUNCRIME	1.4098	0.365	3.86	0.001	4.09
EVENING	-1.4180	0.621	-2.28	0.05	0.24
LATENIGHT	-1.7920	0.743	-2.41	0.05	0.17
TRAVEL DIST	0.0085	0.002	3.56	0.001	1.01
OFFAGE	-0.0023	0.002	-1.12	n.s.	1.00
OFFBLACK	-1.4442	0.396	-3.64	0.001	0.24
NUM OFF	-0.9035	0.560	-1.61	n.s.	0.41
6 SCHOOLS					
Alternative N=88					
Constant	6.8782	0.873	7.87	0.001	970.84
GUNCRIME	-2.0414	0.614	-3.32	0.001	0.13
EVENING	-2.1276	0.479	-4.44	0.001	0.12
LATENIGHT	-2.2976	0.752	-3.06	0.01	0.10
TRAVEL DIST	0.0066	0.003	2.35	0.05	1.01
OFFAGE	-0.3716	0.040	-9.19	0.001	0.70
OFFBLACK	-1.1790	0.381	-3.09	0.01	0.31
NUM OFF	-0.1896	0.279	-0.68	n.s.	0.83
7 PARKS					
Alternative N=81					
Constant	2.7911	0.618	4.52	0.001	16.30
GUNCRIME	-0.7293	0.337	-2.17	0.05	0.48
EVENING	0.0657	0.283	0.23	n.s.	1.07
LATENIGHT	-1.1658	0.461	-2.53	0.05	0.31
TRAVEL DIST	0.0068	0.002	3.05	0.01	1.01
OFFAGE	-0.1946	0.025	-7.83	0.001	0.82
OFFBLACK	-1.2409	0.312	-3.98	0.001	0.30
NUM OFF	0.2586	0.177	1.46	n.s.	1.30
8 PUBLIC TRANSPORT					
Alternative N=55					
Constant	-3.0490	0.721	-4.23	0.001	0.05
GUNCRIME	-0.5504	0.400	-1.38	n.s.	0.58
EVENING	-0.4107	0.409	-1.00	n.s.	0.66
LATENIGHT	0.0074	0.338	0.02	n.s.	1.01
TRAVEL DIST	0.0092	0.002	5.28	0.001	1.01
OFFAGE	-0.0010	0.001	-0.90	n.s.	1.00
OFFBLACK	0.6919	0.608	1.14	n.s.	2.00
NUM OFF	-0.3629	0.348	-1.04	n.s.	0.70

Reference Alternative: **2 RESIDENTIAL**

Table 21.3: (continued)

```
-------------------------------------------------------------------
Multicollinearity statistics
Predictor           Pseudo-Tolerance
GUNCRIME            0.94
EVENING             0.92
LATENIGHT           0.92
TRAVEL DIST         0.98
OFFAGE              0.99
OFFBLACK            1.00
NUM OFF             0.94
```

In Table 21.4, the percentage distribution of the 7 premises types is compared for 1997 and 1998 with the 1998 percentage correctly predicted for each type of premises using the MNL equation developed for 1997 robberies. Overall, not much has changed between the two years.

Table 21.4:
Non-Street Robberies in 1997 & 1998
1998 Predicted by the 1997 Multinomial Logit Model

Type of Premises	**1997**	**1998**	**Percent Correctly Predicted for 1998**
Residential	48.6%	47.8%	53.0%
Garages/Parking	11.3%	10.5%	12.6%
Commercial	23.8%	25.4%	33.2%
Banks/CurrEx	2.1%	3.7%	4.8%
Schools	5.5%	5.3%	39.2%
Parks	5.1%	4.0%	11.7%
Public Transit	3.5%	3.3%	5.0%
Number of Robberies	1587	1441	

In order to be an improvement on the naïve assumption that the percentage of incidents at each premises type is no better than the overall distribution of premises in 1998, the multinomial logit model based on 1997 (Column 3) should predict the premises of incidents better than the marginal percentage distribution of incidents in 1998 (Column 2). It does for all premise types. A few examples:

1. 47.8% of incidents were residential with the model correctly predicting 53.0% percent of them.
2. 25.4% of incidents were commercial with the model correctly predicting 33.2% percent.

3. 5.3% of incidents were in schools or school yards and the model correctly predicted 39.2% percent.
4. Garages and parking lots were only slightly better predicted by the model than by the 1998 percentage distribution. 10.5% of incidents were in garages or parking lots, and the model correctly predicted 12.6% of these.

Example 1 Conclusion

When an offender chooses a type of premise to commit a robbery, the choice is not random. Personal characteristics such as age and racial group make a difference, but so do decisions that the offenders make when coming into the incident such as gun availability, distance from home, and time of day. This example demonstrates how Multinomial Logit models can be used to clarify the offender's choice by the type of premise. The example also demonstrates that a model based on robbery choices made in one year can be useful in prediction of robberies in another year.

Another example of the Multinomial Logit model is presented in the attachment where Levine, Robertson and Fosberg analyze the type of weapon used in Houston robberies.

The Conditional Logit Model

The CL model is estimated on a different data structure. It is a matrix where each row (record) represents a combination of an observational unit *n* (a case, often a decision maker) with an alternative in the choice set *i*, and where each column represents a variable (a characteristic of the observational unit and/or the alternative). In this case, each record represents a possible alternative that the case (or decision maker) is presented with. The dependent variable is a binomial variable and indicates which alternative *i* from a set of alternatives was chosen by observational unit *n*.

For example, a community is divided into twenty neighborhoods (alternatives). Each of these is classified according to number of businesses, wealth, racial makeup and population size (5 variables). For each case, an offender 'selects' a neighborhood where the crime is committed (choice).[1] For 100 cases and twenty alternatives, a matrix of 2,000 records and a minimum of six variables would be necessary. The sixth variable identifies the chosen alternative. The number of records can grow quickly. The following is a simplified example. The variation in outcomes is explained by variation in the characteristics of the alternatives. CrimeStat is able to construct such a file by combining a 'case file' and an 'alternatives file'. Below we present a simplified description of the process.

[1] The offender may not do this rationally, of course, and may simply be at that location (a routine activity). Nevertheless, the model assumes that the offender has made a utility calculation to commit the crime at the location. To that extent, it is a decision among many alternatives.

Destination Choice

We start with the case file shown in Table 21.1, that is the file that is used for estimating a multinomial logit model, and by a model of the *Destination* (i.e.. in which area, P, Q, R or S, did the offender commit the crime?). Whereas a MNL model is used to assess whether linear combinations of characteristics associated with the cases (e.g.,*Origin*, *Age*, *Time* and *CrimeType*) predict which zones the offender selects to commit the crime, a CL model is used to assess whether characteristics of the alternatives predict the zone chosen. The alternatives are the zones themselves and, obviously, additional information is needed on the alternatives.

Table 21.5 shows part of an example file, containing the four alternative destination areas P, Q, R and S for the decision on where to offend. The variables include an identifier (Zone), the percentage of the household below a poverty threshold (*Poverty*) and the percentage of the non-residential land use (*Non-Residential*) in the zone.

Table 21.5:
Zone File Describing 4 Alternative Zones

Zone	*Poverty*	*Non-Residential*
P	2	40
Q	2	16
R	4	23
S	5	12

The data structure required for estimation of the CL model represents all possible combinations of the rows in the case file and the rows in the zone file, including the variables in both files. It also includes for each decision maker a binomial variable indicating the alternative that was chosen by the decision maker. For example, if there are 200 cases (decision makers) and 7 alternatives that are available, there will be 1,400 records (200 x 7) in the data set. Each decision maker will be represented 7 times, representing each of the 7 alternatives that the decision maker is confronted with. However, the decision maker will have selected only one of these alternatives. For that record, the value of the binomial choice variable will be 1; for the other six records, the value of the binomial choice variable will be 0.

To go back to the example, Table 21.6 displays the combination of Tables 21.1 and 21.5. Note that the columns 1-6 of Table 21.6 are a copy of Table 21.1 with each row repeated four times (the first original row in rows 1-4). Also verify that columns 7-9 are copies of Table 21.5, with each row repeated five times (the first original row in rows 1, 5, 9, 13 and 17, the second original row in rows 2, 6, 10, 14 and 18, etc.). Finally note that the indicator variable,

Table 21.6:

Case-alternative File Describing 20 Case-Alternative Combinations

ID	Org	Dest	Age	CrimeTyp	Time	Zone	Pov	NonRes	Chosen	Home
1	P	P	18	Burglary	3am	P	2	40	1	1
1	P	P	18	Burglary	3am	Q	2	16	0	0
1	P	P	18	Burglary	3am	R	4	23	0	0
1	P	P	18	Burglary	3am	S	5	62	0	0
2	Q	P	23	Robbery	7pm	P	2	40	1	0
2	Q	P	23	Robbery	7pm	Q	2	16	0	1
2	Q	P	23	Robbery	7pm	R	4	23	0	0
2	Q	P	23	Robbery	7pm	S	5	62	0	0
3	R	S	42	Illicit drug	2pm	P	2	40	0	0
3	R	S	42	Illicit drug	2pm	Q	2	16	0	0
3	R	S	42	Illicit drug	2pm	R	4	23	0	1
3	R	S	42	Illicit drug	2pm	S	5	62	1	0
4	R	Q	32	Robbery	1pm	P	2	40	0	0
4	R	Q	32	Robbery	1pm	Q	2	16	1	0
4	R	Q	32	Robbery	1pm	R	4	23	0	1
4	R	Q	32	Robbery	1pm	S	5	62	0	0
5	S	R	19	Burglary	6am	P	2	40	0	0
5	S	R	19	Burglary	6am	Q	2	16	0	0
5	S	R	19	Burglary	6am	R	4	23	1	0
5	S	R	19	Burglary	6am	S	5	62	0	1

Chosen, is set to 1 if the value in variable *Destination* matches the value in variable *Zone*. The variable *Home* will be discussed below.

Note that in Table 21.6, the zone characteristics *Pov* and *Nonres* only vary across alternatives but not across cases (decision makers): the values of these two variables are just repeated in every case. Quite often, however, the model includes variables that vary across alternatives and across cases as well. The last column in Table 21.6 contains a binomial variable, *Home*, that indicated whether an alternative zone is the zone of residence of the offender. Thus, it has value 1 if *Origin=Zone*, and 0 otherwise. This variable varies both across alternatives (e.g. for a given ID, one alternative equals 1 and the other equal 0) and across cases (for a given Zone, say A, it equals 1 for case 1, but 0 for cases 2-5). In a similar fashion (but more difficult to verify by just inspecting the table), we could define a new variable that represents the distance between the alternative zone and the zone of the offender's residence.

Note also that the indexing in the independent variables in the MNL and CL models reflects the data structures used for estimation. In the MLN model, $V_{in} = {}^{2}{}_{i}X_{n}$, where the index *n* in the term X_n indicates that the variables only vary between cases (decision makers), so that we only need one row per case. In the CL model, $V_{in} = {}^{2}X_{ni}$, where the indices *n* and *i* in the term X_{ni} reflect that the variables can vary between cases (decision makers) and between alternatives, so that we need multiple rows case (equation 21.11 demonstrated that in the CL model, the variables must vary across alternatives and may vary across decision makers, but cannot be estimated when they vary across decision makers only).

Crime Type Choice

Now let us consider another type of choice: the choice of a crime type. An offender urgently needing money may have to choose a criminal activity that generates the required amount as easily and with as little risk as possible. If we assume that burglary, robbery and illicit drug dealing are the available alternatives, an alternatives file could look like Table 21.7. The variables in this file represent attributes that may differentiate between the crime types: *Expected Profits*, *Detection Risk* and *Time needed to search and attack a target* and that may affect the attractiveness of these offences to the offenders.

Table 21.7:
Crime Type File
(Alternative Crime Types)

Crime type	*Expected Profits*	*Detection Risk*	*Sanction Severity*	*Time Needed*
Burglary	200	.07	3	60
Robbery	50	.15	5	20
Illicit drug	20	.02	2	40

Analogously to the case of destination zone choice, the data structure required for estimation of the CL model represents all possible combinations of the rows in the case file (Table 21.1) and the rows in the alternatives file (Table 21.6), including the variables in either file, and also including for each decision maker a binomial variable indicating the alternative (which crime type) that was chosen by the decision maker.

Table 21.8 displays the combination of Tables 21.1 and 21.7. The first six columns of Table 21.8 are a copy of Table 21.1 with each row repeated three times. Also verify that column 7-9 are copies of Table 21.3, with each row repeated four times (the first original row in rows 1,4,7, 10, and 13, the second original row in rows 2, 5, 8, 11 and 14, etc.). Finally note that the indicator variable *Chosen* is set to 1 if the value in variable *Type* matches the value in variable *CrimeType*.

Table 21.8:
Case-alternative File Describing 15 Case-alternative Combinations

ID	*Org*	*Dest*	*Age*	*CrimeType*	*Time*	*Type*	*Profit*	*Risk*	*Sanc*	*Time*	*Chosen*
1	P	P	18	burglary	3am	burglary	200	.07	3	60	1
1	P	P	18	burglary	3am	robbery	50	.15	5	20	0
1	P	P	18	burglary	3am	il. drug	20	.02	2	40	0
2	Q	P	23	robbery	7pm	burglary	200	.07	3	60	0
2	Q	P	23	robbery	7pm	robbery	50	.15	5	20	1
2	Q	P	23	robbery	7pm	il. drug	20	.02	2	40	0
3	R	S	42	il. drug	2pm	burglary	200	.07	3	60	0
3	R	S	42	il. drug	2pm	robbery	50	.15	5	20	0
3	R	S	42	il. drug	2pm	il. drug	20	.02	2	40	1
4	R	Q	32	robbery	1pm	burglary	200	.07	3	60	0
4	R	Q	32	robbery	1pm	robbery	50	.15	5	20	1
4	R	Q	32	robbery	1pm	il. drug	20	.02	2	40	0
5	S	R	19	burglary	6am	burglary	200	.07	3	60	1
5	S	R	19	burglary	6am	robbery	50	.15	5	20	0
5	S	R	19	burglary	6am	il. drug	20	.02	2	40	0

Example 2: Modeling Choice of Neighborhood for Residential Burglaries in The Hague with the Conditional Logit Model

The discrete spatial choice approach was first applied to criminal location choices by Bernasco & Nieuwbeerta (2005). This example uses CrimeStat to replicate their analysis of 548 cleared burglaries committed in the years 1996-2001 in the city of The Hague, the Netherlands, by solitary offenders (i.e., offenders who perpetrated the burglary without known accomplishes).

The discrete spatial choice model of burglary integrated journey-to-crime research (that focuses on distance traveled without considering other aspects of criminal location choice) and ecological research (that addresses variation in opportunities and target attractiveness, but ignores the distance offenders have to travel to reach the targets).

Bernasco & Nieuwbeerta distinguished 89 neighborhoods in The Hague, which served as the spatial units of analysis. They argued that neighborhoods would be attractive for burglary if they (1) were affluent, (2) had a large proportion of single-family dwellings, (3) had high population turnover (4) had high ethnic heterogeneity, (5) had large numbers of households, (6) were situated relatively close to the city center and (7) were located relatively close to the offender's residence. Note that the first six criteria are the attributes of the 89 alternative neighborhoods (independently of any attributes of the burglar), while the last criterion (proximity to offender's home) depends on the locations of both the offender and the potential target neighborhoods.

In Table 21.9, the 548 The Hague burglaries are analyzed with the conditional logit model, using the following 7 variables as predictors of the burglars' selection of a target neighborhood:

1. PROPVAL. Average value of residential properties, in 100,000 euro
2. SINGFAM. Percentage of units that are single-family dwellings, in 10% units
3. RESMOBIL. Percentage of residents that moved during past year, in 10% units
4. ETNHETERO. (Ethnic Heterogeneity). Blau / Herfindahl index (x 10)
5. PROXIMITY. Negative distance between offender neighborhood and potential target neighborhood, in kilometers. The authors used negative distance instead of distance because this yielded a model in which all expected parameters were positive.
6. PROXCITY.Negative distance between city center and potential target neighborhood, in kilometers
7. RESUNITS. Number of residential properties in the neighborhood, in 1,000 properties

The results in Table 21.9 replicate the findings reported by Bernasco & Nieuwbeerta (2005, p. 308).[2] The summary section of the output reports general information about the model and the estimation procedure, including the names of the data file and the dependent variable. The output shows that the number of records is 48,772, which is 548 × 89 (i.e. the number of offenders multiplied by the number of The Hague neighborhoods). The number of degrees of freedom is 541 (the number of offenders -548, minus the number of estimated parameters -7). As discussed in the multinomial logit example, the likelihood statistics (Log Likelihood, Akaike Information Criterion, AIC ; and Bayesian Information Criterion, BIC) indicate how well the model fits the data (lower values indicate better fit). These statistics are only used to compare different models, and have no useful interpretation for a single model.

The coefficient section reports the results for each predictor variable and include the estimated coefficients, their standard errors, t-values, and p-values. The odds ratios column is the most useful statistic for substantive interpretation of the outcome. The odds ratio (which equals e^{β}) represents the factor by which the odds that a neighborhood is chosen for a burglary increases or decreases when the value of the predictor increases by one unit. An odds ratio greater than 1 indicates that the odds increase while an odds ratio between 0 and 1 indicate that the odds decrease.

For example, the odds ratio of 1.05 for variable PROPVAL indicates that as the average value of properties in the neighborhood increases by 100,000 euro, the odds that it is selected by a burglar increase by a factor 1.05 (i.e. by approximately 4.5 percent). Another example: the

[2] The standard errors reported here are slightly smaller than those reported by Bernasco & Nieuwbeerta (2005), a difference due to their correction of the standard errors for the possible interdependence among the burglaries (the 548 burglaries were committed by 290 unique persons; thus, some of them committed multiple burglaries).

estimated value of 1.67 for proximity means that if a neighborhood is located one kilometer closer to the offender's home, the odds that it will be selected by this burglar increase by a factor 1.67 (i.e. by approximately 67 percent).

Table 21.9:
Conditional Logit Model of Burglary Neighborhood Choice

```
 Model result:
 Data file:                          TheHagueBurglary.dbf
 DepVar:                             CHOSEN
 N:                                  48,772
 Df:                                 541
 Type of choice model:               Conditional logit model
 Number of Alternatives:             89
 Method of estimation:               MLE

      Likelihood statistics
 Log Likelihood:                     -2,203.3
 AIC:                                4,420.6
 BIC/SC:                             4,450.7

      Model error estimates
 Mean absolute deviation:            0.02
 Mean squared predicted error:       0.01
```

Predictor	Coefficient	Stand Error	Pseudo-Tolerance	t-value	p-value	Odds Ratio
PROPVAL	0.0445	0.112	0.33	0.40	n.s.	1.05
SINGFAM	0.1239	0.042	0.43	2.96	0.01	1.13
RESMOBIL	-0.0285	0.046	0.48	-0.62	n.s.	0.97
ETNHETERO	0.1380	0.032	0.37	4.37	0.001	1.15
PROXIMITY	0.5140	0.034	0.74	15.22	0.001	1.67
PROXCITY	-0.0812	0.049	0.37	-1.66	n.s.	0.92
RESUNITS	0.3039	0.029	0.80	10.61	0.001	1.36

```
Average Predicted Probability:
            Mean
   CHOSEN   Probability      N          StdDev
     0      0.011           48224      0.012
     1      0.028           548        0.024
 Total      0.011           48772      0.013
```

The section also includes the pseudo-tolerances of the indicator variable (see Chapters 15 and 17 for discussion of this statistic). If the tolerance of a variable is low, this indicates that the variable is strongly correlated with linear combinations of the other predictor variables in the equation, and that it therefore does not add much unique variability to the prediction of the

dependent variable. This situation is called multicollinearity and is usually solved by removing the variable with the lowest tolerance from the equation. Note that three of the variables are not significant and several have low tolerances and that a simplified model can be produced by dropping them without much loss of generality (not shown).

The last section also lists average predicted probabilities for neighborhoods that were chosen (.028) and those that were not chosen (.011). Note that the average predicted probability multiplied by the total number of records yields the total number of burglary cases in the file.[3]

Conclusion

In discrete choice modeling, the dependent variable is made up of mutually exclusive and exhaustive categories. The category that is chosen is based upon characteristics of the decision maker (in the multinomial logit model), the characteristics of the alternatives (in the conditional logit model), or the interaction of the two (also in the conditional logit model). Interpretations of discrete choice models can be closely linked to the economic theory of utility maximization. Of all possible alternatives, the alternative is selected that maximizes gain and minimizes cost.

The CrimeStat discrete choice module is designed for regression when the dependent variable consists of unordered categories such as type of weapon or neighborhood where a crime is committed. This is in contrast to more traditional regression that is mainly concerned with dependent variables that are continuous or quasi-continuous, such as rates or counts. The Discrete Choice module is a multinomial extension of binomial logistic regression, discussed in Chapter 18, which allows for only two categories of the dependent variable.

The Discrete Choice module provides for two different models, the multinomial and the conditional logit model. Which one is used must be based upon the availability and relevance of data that reflect attributes of the categories and attributes of the cases (usually offenders, or crimes). To some extent it also depends upon the number of categories of the dependent variable since the tractability of the multinomial model decreases as the number of categories grows.

The conditional logit model is most appropriate if the outcome is assumed to be based on characteristics of the alternatives or their interaction with characteristics of the situation or the decision-maker. The CL data structure duplicates every possible alternative for each case and designates one as chosen. The results summarize the difference between the chosen selection and all others. For example, Chicago has seventy-seven neighborhoods that vary in terms of wealth, number of businesses, level of drug crime, and population. They also vary in distance from an offender's home. Each offender's decision about in which neighborhood to commit the

[3] To do this accurately, one needs more than 3 decimal places. The CrimeStat output includes six decimal places. We have reduced the number of decimal places in the table to make it clearer.

crime is based upon a comparison of the characteristics of the 77 neighborhoods. The data file has one record for each alternative that the decision maker faces. If 1,000 offenders are analyzed, the resultant file would have 77,000 records.

The multinomial model may be appropriate if the choice has fewer categories and is dependent mainly on characteristics of the offender and the particular incident. A separate equation is constructed that compares a reference category with every other category of the dependent variable. For example, if weapon choice is dependent upon the victim's age and gender, type of target, and time of day, then a separate equation is constructed comparing gun incidents, the most frequent category, to knives, other weapons and strong armed. The data file contains one record for each offender.

In Chapter 22, we discuss the use of the CrimeStat discrete choice module routines to estimate these two models. Two additional routines are included in the discrete choice module. First, as discussed above the Conditional logit requires data organization that combines characteristics of the incident and all possible choices. CrimeStat will build this file for you. Second, both discrete and conditional models allow for prediction of dependent variables in one data set from the relationships found in another. Thus, in the example of Chicago robberies (above) 1998 robbery locations are predicted based on MNL coefficients of 1997 robberies.

If an analyst wants to consider the 'who', 'where', or 'why' of choice among multiple mutually exclusive possibilities and has a model in which criminals maximize the utility of their choices, then the either Conditional logit or Multinomial logit in the discrete choice module are appropriate techniques.

References

Ben-Akiva, M. E., & Bierlaire, M. (1999). Discrete Choice Methods and their Applications to Short Term Travel Decisions. In R. W. Hall (Ed.), *Handbook of Transportation Science* (pp. 5-34). Norwell, MA: Kluwer.

Ben-Akiva, M. E., & Lerman, S. R. (1985). *Discrete Choice Analysis: Theory and Application to Travel Demand*. Cambridge, MA: MIT Press.

Bernasco, W. (2010a). Modeling Micro-Level Crime Location Choice: Application of the Discrete Choice Framework to Crime at Places. *Journal of Quantitative Criminology, 26*(1), 113-138.

Bernasco, W. (2010b). A Sentimental Journey to Crime; Effects of Residential History on Crime Location Choice. *Criminology, 48*, 389-416.

Bernasco, W. (2007). The usefulness of measuring spatial opportunity structures for tracking down offenders: A theoretical analysis of geographic offender profiling using simulation studies. *Psychology, Crime & Law, 13*, 155-171.

Bernasco, W. (2006). Co-Offending and the Choice of Target Areas in Burglary. *Journal of Investigative Psychology and Offender Profiling, 3*, 139-155.

Bernasco, W., & Block, R. (2009). Where Offenders Choose to Attack: A Discrete Choice Model of Robberies in Chicago. *Criminology, 47*(1), 93-130.

Bernasco, W., & Kooistra, T. (2010). Effects of Residential History on Commercial Robbers' Crime Location Choices. *European Journal of Criminology, 7*(4), 251-265.

Bernasco, W., & Nieuwbeerta, P. (2005). How Do Residential Burglars Select Target Areas? A New Approach to the Analysis of Criminal Location Choice. *British Journal of Criminology, 45*, 296-315.

Clare, J., Fernandez, J., & Morgan, F. (2009). Formal Evaluation of the Impact of Barriers and Connectors on Residential Burglars' Macro-Level Offending Location Choices. *Australian and New Zealand Journal of Criminology, 42*, 139-158.

Jepsen, L., & Jepsen, C. (2002). An empirical analysis of the matching patterns of same-sex and opposite-sex couples. *Demography, 39*(3), 435-453.

References (continued)

Krebs, J. R., & Davies, N. B. (1993). *An Introduction to Behavioural Ecology* (3th ed.). Oxford: Blackwell.

McFadden, D. (1980). Econometric Models for Probabilistic Choice Among Products. *The Journal of Business, 53*(3), S13-S29.

McFadden, D. (1973). Conditional Logit Analysis of Qualitative Choice Behavior. In P. Zarembka (Ed.), *Frontiers in Econometrics* (pp. 105-142). New York: Academic Press.

Palfrey, T. R., & Poole, K. T. (1987). The Relationship between Information, Ideology, and Voting Behavior. *American Journal of Political Science, 31*(3), 511-530.

Phillips, S. (2003). The Social Structure of Vengeance: A Test of Black's Model. *Criminology, 41*(3), 673-708.

Rengert, G. F. (1981). Burglary in Philadelphia: A Critique of an Opportunity Structure Model. In P. J. Brantingham & P. L. Brantingham (Eds.), *Environmental Criminology* (pp. 189-202). Beverly Hills, CA: Sage.

Smith, T. S. (1976). Inverse Distance Variations for the Flow of Crime in Urban Areas. *Social Forces, 54*(4), 802-815.

Tita, G., & Griffiths, E. (2005). Traveling to Violence: The Case for a Mobility-Based Spatial Typology of Homicide. *Journal of Research in Crime and Delinquency, 42*, 275-308.

Train, K. E. (2009). *Discrete Choice Methods with Simulation* (2nd ed.). New York: Cambridge University Press.

Train, K. E. (1980). A Structured Logit Model of Auto Ownership and Mode Choice. *The Review of Economic Studies, 47*(2), 357-370.

Attachment A

Modeling Correlates of Weapon Use in Houston Robberies With the Multinomial Logit Model

Ned Levine	**Alan Robertson**	**Barry Fosberg**
Ned Levine & Associates	Houston Police Department	Houston Police Department
Houston, TX	Houston, TX	Houston, TX

Introduction

We made an analysis of weapon use in Houston robberies. Because the type of weapon used rarely changes, the Multinomial Logit model was an appropriate modeling tool. Between 2007 and 2009, there were 33,419 robberies that occurred within the City of Houston. Of these, suspect information was obtained for 3,709 of these offenses. Using the suspect information for these 3,709 offenses, we modeled predictors of weapon use.

Figure 21A.1 shows the distribution of weapons within the area covered by the Houston Police Department. Of the weapons used, 1,744 (or 47%) involved firearms, 272 (or 7%) involved knives, 1,184 (or 32%) involved bodily force, 192 (or 5%) involved threat, and 317 (or 9%) involved another weapon. Using the 'other weapon' as the reference category, we related weapon choice to 11 variables grouped into five categories: 1) Offender characteristics (age, gender, being of Hispanic ethnicity, being of African-American ethnicity); 2) Presence of co-offenders (the number of suspects); 3) Whether the robbery occurred on a commercial premise or not; 4) Time period (night, afternoon, morning), and 5) Crime location characteristics (median household income of the block group at the crime location, distance from the offenders residence to the crime location).

Method

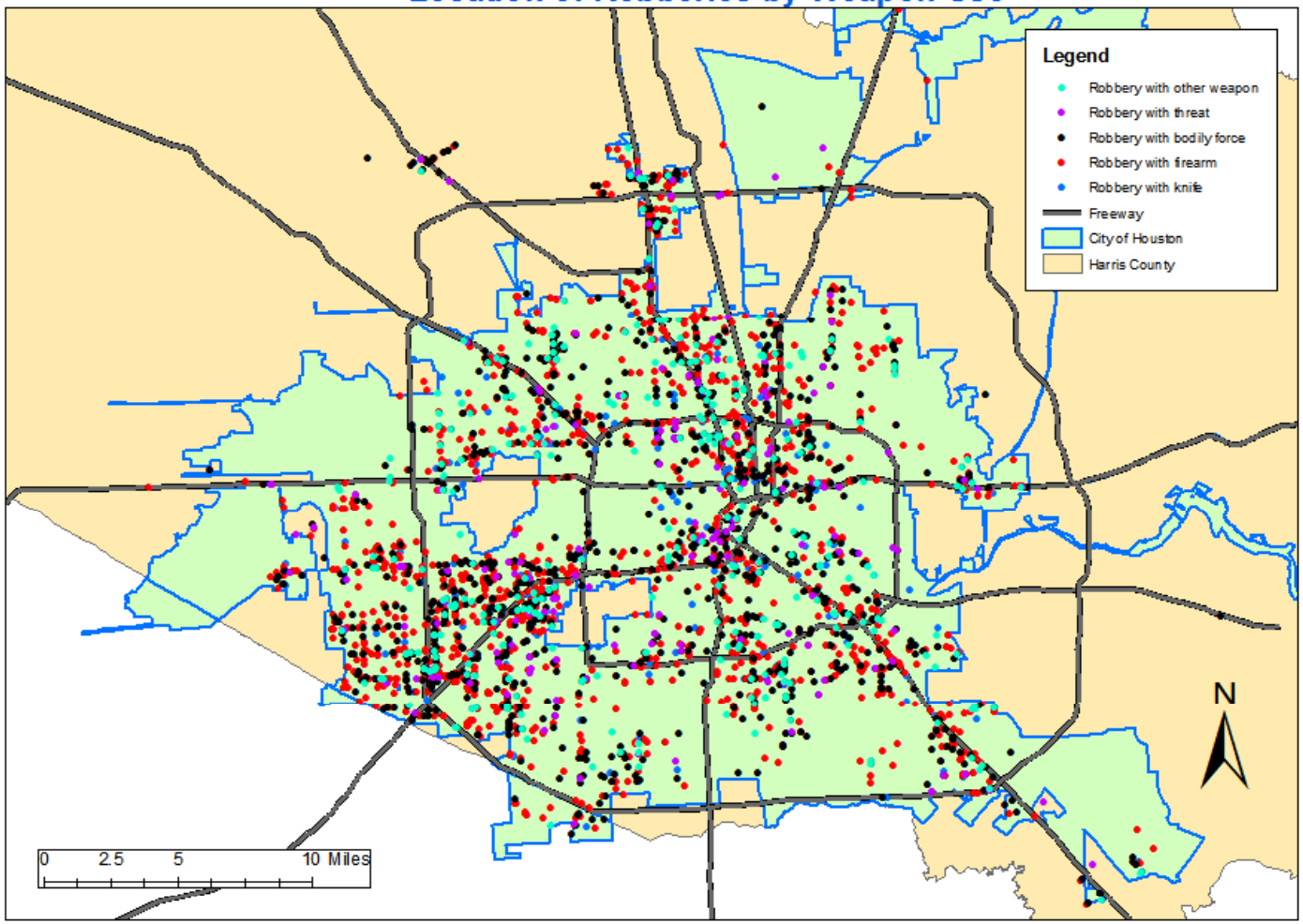
The multinomial logit model was used to estimate the effect of the coefficients on weapon choice. The choice probability, P_{ni}, that the offender, *i*, chooses a particular weapon, *j*, is estimated by an exponentiated linear combination of independent predictors associated with the offender, *k*, proportional to the choice probabilities for all weapons:

$$P_{ij} = \frac{Observed\ utility\ of\ weapon\ j}{Observed\ utility\ of\ all\ weapons} = \frac{e^{\beta_{0j}+\sum_1^K(\beta_{jk}X_{ijk})}}{\sum_1^J e^{\beta_{0j}+\sum_1^K(\beta_{jk}X_{ijk})}}$$

Figure 21A.1:

Houston Robberies: 2007 to 2009

Location of Robberies by Weapon Use



That is, the probability of the offender choosing any one weapon is estimated by an exponentiated linear combination of observed variables associated with the offenders divided by the sum of the exponentiated linear combination for all weapon choices. The coefficients are estimated across offenders but the probabilities are calculated for each offender separately.

Table 21A.1 presents the results of the model and Table 21A.2 summarizes the initial frequencies and the average predicted probabilities. Compared to the use of another weapon (the reference group), firearm use was associated with younger Hispanic or African-American males, with more accomplices, and was more likely to be committed on commercial premises in higher income locations at night or in the early morning. Crime travel distance was farther.

Table 21A.1:

Multinomial Logit Predictors of Weapon Use in Houston Robberies: 2007-09

Model result:	
DepVar:	**WEAPON**
N:	**3709**
Df:	**3696**
Likelihood statistics	
Log Likelihood:	**-4432.1**
AIC:	**8936.3**
BIC/SC:	**9160.2**
Model error statistics	
Mean absolute deviation:	**0.27**
Mean squared predicted error:	**0.14**

Weapon: Firearm

Predictor	Coefficient	Stand Error	t-value	p-value	Odds ratio
Constant	**0.4959**	**0.005**	**91.43**	**0.001**	**1.64**
	Offender characteristics				
AGE	**-0.0279**	**0.003**	**-9.79**	**0.001**	**0.97**
FEMALE	**-0.9308**	**0.005**	**-171.60**	**0.001**	**0.39**
HISPANIC	**1.0317**	**0.005**	**190.54**	**0.001**	**2.81**
AFRICAN-AMERICAN	**1.3980**	**0.005**	**258.29**	**0.001**	**4.05**
	Co-offenders				
NUMSUSPCTS	**0.1774**	**0.005**	**33.08**	**0.001**	**1.19**
	Type of premise				
COMMERCIAL	**0.6431**	**0.005**	**118.73**	**0.001**	**1.90**
	Time period				
NIGHT	**0.3927**	**0.005**	**72.47**	**0.001**	**1.48**
AFTERNOON	**-0.2119**	**0.005**	**-39.11**	**0.001**	**0.81**
MORNING	**0.1672**	**0.005**	**30.84**	**0.001**	**1.18**
	Crime location				
MED HH INC	**0.00001**	**0.000003**	**1.97**	**0.05**	**1.00**
TRAVEL DISTANCE	**0.0289**	**0.004**	**6.65**	**0.001**	**1.03**

Table 21A.1: (continued)

Weapon: Knife

Predictor	Coefficient	Stand Error	t-value	p-value	Odds ratio
Constant	-0.6883	0.005	-126.82	0.001	0.50
Offender characteristics					
AGE	0.0205	0.004	5.82	0.001	1.02
FEMALE	-0.1706	0.005	-31.43	0.001	0.84
HISPANIC	0.4801	0.005	88.52	0.001	1.62
AFRICAN-AMERICAN	-0.1024	0.005	-18.88	0.001	0.90
Co-offenders					
NUMSUSPCTS	-0.1842	0.005	-34.03	0.001	0.83
Type of premise					
COMMERCIAL	0.0914	0.005	16.84	0.001	1.10
Time period					
NIGHT	0.2574	0.005	47.45	0.001	1.29
AFTERNOON	-0.0165	0.005	-3.05	0.01	0.98
MORNING	0.1056	0.005	19.46	0.001	1.11
Crime location					
MED HH INC	0.000004	0.000003	1.123	n.s.	1.00
TRAVEL DISTANCE	-0.0300	0.005	-5.91	0.001	0.97

Weapon: Bodily force

Predictor	Coefficient	Stand Error	t-value	p-value	Odds ratio
Constant	0.5384	0.005	99.27	0.001	1.71
Offender characteristics					
AGE	-0.0012	0.003	-0.42	n.s.	1.00
FEMALE	0.0542	0.005	9.99	0.001	1.06
HISPANIC	0.2362	0.005	43.61	0.001	1.27
AFRICAN-AMERICAN	0.5802	0.005	107.17	0.001	1.79
Co-offenders					
NUMSUSPCTS	-0.1342	0.005	-24.96	0.001	0.87
Type of premise					
COMMERCIAL	0.2963	0.005	54.70	0.001	1.34
Time period					
NIGHT	0.0861	0.005	15.88	0.001	1.09
AFTERNOON	0.4638	0.005	85.64	0.001	1.59
MORNING	0.3381	0.005	62.35	0.001	1.40
Crime location					
MED HH INC	0.00001	0.000003	4.14	0.001	1.00
TRAVEL DISTANCE	-0.0227	0.005	-5.02	0.001	0.98

Table 21A.1: (continued)

Weapon: Threat

Predictor	Coefficient	Stand Error	t-value	p-value	Odds ratio
Constant	-1.9169	0.005	-353.16	0.001	0.15
Offender characteristics					
AGE	0.0187	0.004	5.17	0.001	1.02
FEMALE	-1.2088	0.005	-222.67	0.001	0.30
HISPANIC	0.2279	0.005	42.00	0.001	1.26
AFRICAN-AMERICAN	0.6623	0.005	122.08	0.001	1.94
Co-offenders					
NUMSUSPCTS	-0.3061	0.005	-56.47	0.001	0.74
Type of premise					
COMMERCIAL	0.7707	0.005	142.04	0.001	2.16
Time period					
NIGHT	-0.1765	0.005	-32.52	0.001	0.84
AFTERNOON	-0.0113	0.005	-2.08	0.05	0.99
MORNING	0.4941	0.005	91.049	0.001	1.64
Crime location					
MED HH INC	0.00002	0.000003	4.25	0.001	1.00
TRAVEL DISTANCE	0.0193	0.005	3.86	0.001	1.02

Reference choice: Other weapon

On the other hand, knife use was associated with older, Hispanic males with few accomplices. The robberies were more likely to be committed on commercial premises at night or early morning. Crime travel distance was shorter.

Bodily force was associated with Hispanic or African-American females and with few accomplices. The robberies were more likely to be committed in higher income locations on commercial premises in the afternoon, morning or, to a lesser extent, late at night. The crime travel distance was shorter.

Finally, threats were associated with older Hispanic or African-American males with no or few accomplices. The robberies were more likely to be committed on commercial premises in the morning in higher income locations. The crime travel distance was farther.

Table 21A.2 shows that the average predicted probabilities for weapon use across all robbers exactly predicted the actual distribution of weapon use.

Conclusion

The most distinguishing variable is the number of suspects. More co-offenders lead to a greater use of firearms, suggesting the involvement of gangs. Other consistent predictors are

ethnicity - Hispanic or African-Americans are more likely to use weapons than non-Hispanic White or Asian suspects, and gender - males are more likely to use firearms, knives or threats than females, who in turn are more likely to use bodily force. Commercial properties tend to be

Table 21A.2:
Summary of Predictions

Weapon	Frequency of Weapon Use	(%)	Average Predicted Probability
Firearm	1,744	(47%)	0.47
Knife	272	(7%)	0.07
Bodily force	1,184	(32%)	0.32
Threat	192	(5%)	0.05
Other weapon	317	(9%)	0.09
TOTAL	**3,709**	**(100%)**	**1.00**

disproportionately associated with weapons of all sorts primarily because they are the most common location for robberies in general. There are subtle differences in the time period and in the travel distance in predicting the type of weapon used.

Chapter 22:

The CrimeStat Discrete Choice Module[1]

Wim Bernasco
NSCR, Amsterdam
&
VU University Amsterdam
Netherlands

Richard Block
Loyola University
Chicago, IL

Ned Levine
Ned Levine & Associates
Houston, TX

Ian Cahill
Cahill Software
Edmonton, AB

[1] The code for the Multinomial Logit and Conditional Logit models was produced by Mr. Ian Cahill of Cahill Software, Edmonton, Alberta, as part of his *MLE++* software package. We have added summary statistics, significance tests, the routine for creating a conditional logit dataset, and the predictive module. We would like to thank Ms. Haiyan Teng for her programming.

Table of Contents

Table of Contents (continued)

Chapter 22:

The CrimeStat Discrete Choice Module

We now describe the *CrimeStat* discrete choice module. There are two pages in the module. The Discrete Choice I page allows the creation of a data set appropriate for the conditional logit model and it estimates either multinomial logit or conditional logit models. The model coefficients can be saved. Using the saved model coefficients, the Discrete Choice II page calculates predicted probabilities in either the same or another data set.

Discrete Choice Modeling I

The aim of the discrete choice I modeling module is to estimate a functional relationship between a discrete (nominal) dependent variable and one or more independent variables. It is a statistical method that is derived from utility theory, i.e. random utility maximization (RUM) theory. A 'decision maker' (e.g., an offender committing a crime) is faced with a set of alternatives, labeled 1 through *J*, from which s/he has to select exactly one. The probability that an alternative will be chosen is a function of its observed and unobserved utility to the decision maker. The observed utility is a function of known variables and can be expressed as a linear combination of the independent variables. The unobserved utility is the random error component of the model. The estimated probability is the exponentiated observed utility of a specific alternative, *J*, divided by the sum of the exponentiated observed utilities of all available alternatives (see Chapter 21).

There are two general forms of the discrete choice model, multinomial logit and conditional logit. The *multinomial logit* model estimates the probability that a specific alternative, 1 to *J*, as a function of characteristics of the decision makers, either personal characteristics (e.g., age, gender, ethnicity) or environmental characteristics (e.g., the median household income of the block in which the decision maker lives). The probability that any one alternative is chosen is estimated as a function of these characteristics. Per variable (characteristic), there is one parameter estimated for every alternative, one of which is the reference alternative in which the coefficients are automatically set to 0. The multinomial logit model is most appropriate when the outcome of the choice is expected to depend mostly on characteristics of the decision maker (and not on observed characteristics of the alternatives) and when there are only a limited number of alternatives available (e.g., 5 weapon choices). The *conditional logit* model is a more general model and estimates the probability of a set of alternatives, 1 to *J*, as a function of characteristics of the alternatives themselves, possibly in interaction with characteristics of the decision maker. The conditional logit model is most appropriate when the outcome of the choice is expected to depend mostly on the characteristics of the alternatives, and can handle a large number of alternatives. However, the analysis file

becomes very large. There is a single parameter estimated for every characteristic of the alternative.

Although the multinomial and the conditional logit are based on a single underlying statistical model, their estimation requires different data structures. In the multinomial logit model, the data contain a single record for every decision maker, and a single dependent (nominal) variable that indicates which alternative (*1..J*) was chosen. Thus, if there are *N* decision makers, there are *N* records and at least one variable indicates which alternative was chosen. The file structure is thus similar to that used in the regression module.

In the conditional logit model, for each decision maker there is a record for every choice that this decision maker is faced. Thus, if there are *N* decision makers and *J* alternatives available to every decision maker, then the data set has *N*J* records, one for every alternative faced by the decision maker. In this case, the alternative that was selected has to be indicated by a dichotomous (dummy) variable (1 for chosen and 0 for not chosen).

Figure 22.1 show the interface for the Discrete Choice I page. The discrete choice I section includes two routines:

1. A utility for creating a data set appropriate for the conditional logit model. It matches a data set of *Ncases* (individuals/offenders/records) with a data set of *J* alternatives. The result is a data set with *N*J* records.

2. A routine for estimating either the multinomial logit model or the conditional logit model.

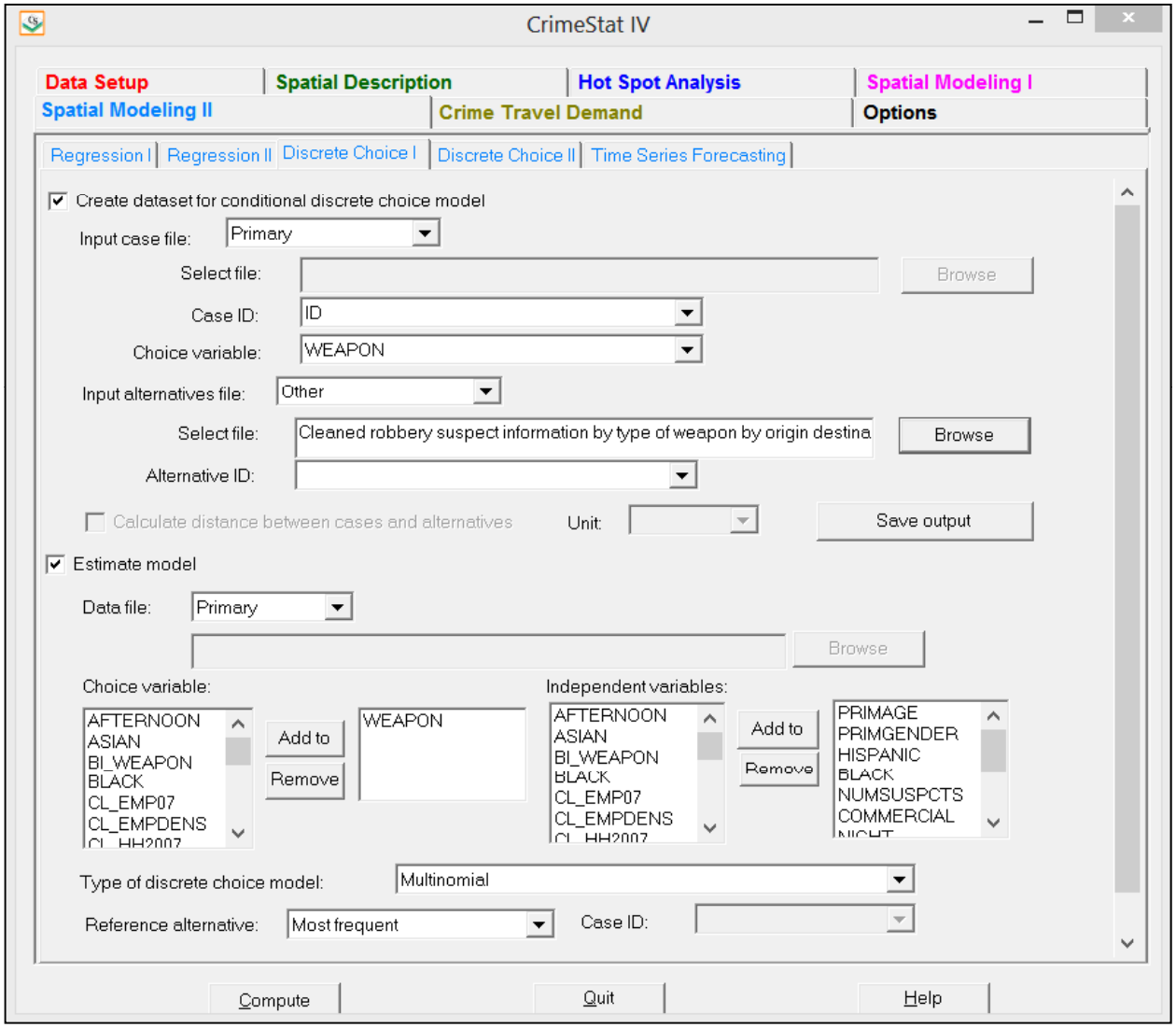
Create Data set for Conditional Logit Model

This routine is optional. It simplifies the task of creating a database for use in the conditional logit model. It matches a *case* database with a alternatives data base, producing the cross join of both databases. The ***case*** database is the database for the multinomial logit model. It will thus have the individual records of the decision makers – offenders, individuals, organizations. It will include at least one variable indicating the alternative that the decision maker selected (e.g., type of crime committed, the type of weapon used, the location where the crime was committed) as well as characteristics of the individuals or characteristics associated with the individuals (e.g., age, gender, ethnicity, median household income of the zone where the decision maker lives time of event, day of week of event).

The ***alternatives*** database, on the other hand, lists the individual alternatives that were available (e.g., all the locations where a crime could be committed, all the different types of

Figure 22.1:

Discrete Choice Modeling I



weapons that were used by different offenders) as well as attributes associated with the alternatives themselves (e.g., median household income or number of employees working at the locations, or characteristics associated with each type of weapon).

The joined file has one record per alternative for each case. Thus, if there are *N* individuals faced with *J* choices, then the matching routine will create *N*J* records. It should be noted that the matching assigns every characteristic associated with a choice to every case associated with a decision maker. A field, called CHOSEN, is automatically added to every record. This field has the value 1 for alternatives that were chosen and 0 for alternatives that were not chosen. The Chosen field should thus sum to N (i.e., only one record per decision maker should have a selected alternative). Also, as an option, and only if both the individuals and the alternatives have geographic coordinates, a second field called DISTANCE will be added that calculates the distance from each case record to each alternative record. The user must specify which distance units are to be used (miles, kilometers, meters, feet, or nautical miles).

For example, if both the case database and the alternatives database contain X and Y coordinates, then it is possible to calculate the distance between every decision maker and every choice.

The routine cannot calculate other interactions associated with a specific alternative and particular decision maker, and such interactions must be added to the data outside CrimeStat. Interactions between variables in the data can be calculated. For example, to test whether increasing distance makes alternatives less attractive for juvenile offenders but not for adult offenders, an interaction DISTANCE x AGE can be calculated. Other interactions require additional information, for example if location choice is what is modeled, one may want to add a variable indicating, for each alternative location, how many prior offences the offender has committed in that alternative location. In these cases the external file is constructed by the user, and the step “Create data set for conditional discrete choice model” is skipped.

Input Case File

The case data set for the Discrete Choice I module can be the Primary file, the Secondary file, or another file. If the Primary file or Secondary files are used, the coordinate system and distance units were defined on the Primary file page. If another file is used, then any coordinates in that file are not defined and the file is treated as a non-spatial file. The user must browse and identify the file. To avoid confusion, the user must verify that no variable/field in the input case file has the same name as any variable in the Input Alternatives File (see below). Note: If a primary file is used, coordinates must be defined for that file. If the file is not spatial, then input it as ‘Other file’.

Case ID

Select the Case ID. The Input Case File must have a Case ID, a variable that uniquely identifies cases in the Input Case File.

Choice Variable

Select the Choice Variable. The Input Case File must contain a variable (field) that identifies alternative chosen by the decision maker. For example, if the choice is about the type of weapon used, then the Choice Variable indicates whether it was a gun, a knife , strong-arm, and so forth. Or, if the choice is the census tract in which a crime was perpetrated, then the Choice Variable identifies the census tract where the incident occurred.

Input Alternatives File

The alternatives data set for the Discrete Choice I module can be the Primary file, the Secondary file, or another file. If the Primary or Secondary files are used, the coordinate system and distance units were defined on the Primary file page. If another file is used, then any coordinates in that file are not defined and the file is treated as a non-spatial file. The user must browse and identify the Input Alternatives File. To avoid confusion, the user must verify that no variable in the input alternative file has the same name as any variable in the Input Case File.

Alternatives ID

Select the Alternatives ID. The Alternatives File must have an Alternative ID, a variable that uniquely identifies records in file. The Alternatives File must contain a record for every possible alternative even those that were never chosen. For example, most census tracts in a city have no homicides during a year, but the alternatives file must include every tract. The coding must match the coding of the Choice Variable in the Input Case File. Be careful about duplicate ID names in the two files as the name will appear twice in the output file with the first use representing the cases and the second use representing the alternatives. The names reflect the link between each case ID and each alternatives ID and it is better to use different names for the ID fields to avoid confusion.

Calculate Distance between Cases and Alternatives

There is an optional box that allows the routine to calculate the distance from each case record to each alternative record. If checked, the routine will calculate the distance. This only applies if both the case file and the alternatives file are either the Primary file or Secondary. The user must specify the distance units to be used in the calculation (in miles, kilometers, feet,

meters, or nautical miles). The box is checked by default. The saved filed will have a new field called DIST. That is, if the X/Y coordinates for an offender's home address are coded in the Input Case File while the coordinates for census tract are recorded in the Input Alternatives File, then the distances from the offender's home to each alternative census tract will be calculated.

Save Output

The matched Input Case and Input Alternatives file is saved as a new file in 'dbf' format, that can subsequently be used to estimate a conditional (but not multinomial) logit model, as described below under 'Estimating a conditional logit model'. The user should define the name of the file and point to the directory where it is saved. The output includes all fields from the case file and all fields from the choice file, and optionally a field DIST containing calculated distances. There will be *J* records for each of the *N* cases. An automatically added field called CHOSEN takes the value '1' for the choice that was selected and '0' for choices that were not selected.

Note that the joined data base can be very large. Before creating a data set for a conditional discrete choice model, include in the alternatives and choice files only variables that are likely to be used in the analysis, and to format them to be as small as possible.

Estimate Model

The Estimate Model routine will estimate a discrete choice model, either the multinomial logit or the conditional logit.

Estimating a Multinomial Logit Model

The *multinomial logit* model is used when there is one record per decision maker with a choice having been made by the decision maker. The model estimates the effect of each independent variable on the probability of each distinct alternative. The data are structured so that there is one record per decision maker with the choice variable indicating which alternative was chosen. The data set is similar to that of the regression model in that there is one record per decision maker. The model then estimates the effects of the independent variables on the probability of each alternative. By definition, one of the alternatives (by default the most frequently chosen alternative, otherwise to be chosen by the user) is the reference alternative to which the other alternatives are compared.

The multinomial logit model is always estimated with a constant. This type of model is appropriate when values of the predictor variables only vary across cases (decision-makers), not across alternatives.

Estimating a Conditional Logit Model

The *conditional logit* model, on the other hand, is used when the values of the predictor variables vary across alternatives. In that case, there is one record per alternative per decision maker. That is, the decision maker is faced with *J* alternatives but chooses only one. The database must indicate which of the *J* alternatives was selected and the model estimates the effect of each independent variable on choosing an alternative. There is a record for every alternative faced by the decision maker. The parameter estimates indicate the effects of the independent variables on the likelihood that the alternative is selected.

Data File

The data set for the model can be either the Primary file or another file (the Secondary file is not available). If the Primary file is used, the coordinate system and distance units are the same as were defined on the Primary file page.

Select file for Other Discrete Choice File

If the discrete choice file is another file than the Primary file, the user must browse and identify the file.

Choice Variable

A list of variables from the discrete choice file is displayed. There is a box for defining the choice variable. The user must select one choice variable. . For the conditional logit model, on the other hand, the variable contains a set of 1's (for selected alternatives) or 0's (for alternatives that were not selected). If the data set was constructed with the *CrimeStat* 'Create data set for conditional discrete choice model' routine, then the field CHOSEN should be used.

Note that the field that is added for the choice variable (whether CHOSEN or another variable) is inspected for unique values. If the data set is large, it may take a while to filter through those values.

Independent Variables

There is a box for defining the independent variables. The user must choose one or more independent variables. There is no limit to the number. The variables are output in the same order as specified in the dialogue so a user should consider how these are to be displayed. The order in which the variables are entered does not affect the estimated parameters.

Type of Discrete Choice Model

The type of discrete choice model to be estimated must be specified. The choices are *Multinomial* (logit) or *Conditional* (logit). The default model is the Conditional logit. NOTE: the file used for a Multinomial Logit model is different than the file used for a Conditional Logit model. With the file used in the Multinomial Logit model, there is one record per case with the choice specified on the record. With the file used in the Conditional Logit model, there is one record per alternative with *J* records per case (where *J* is the number of alternatives). Be sure to use the correct file type. The routine ***assumes*** that the data are ***consistent*** with the type of model chosen. For a multinomial logit model, the routine will treat each record as a separate decision maker and will estimate a model for each choice less the reference choice. For a conditional logit model, the routine will treat each record as one of *J* choices (where *J* is defined by the user – see below) and will estimate a single model for the decision.

The user needs to be very careful that the correct data set is used with the appropriate model because the routine can estimate its equations with either of these data sets. That is, if the data set is appropriate for the multinomial logit model but the user specifies a conditional logit model, the routine will estimate a single equation treating multiples of *J* records as a single decision maker. Similarly, if the data set is appropriate for a conditional logit model but the user specifies a multinomial logit model, the routine will treat each record as if it were a separate decision maker and will estimate one equation for each choice that it finds in the choice variable. The results in both these cases will be meaningless since the there is a mismatch between the data set and the type of model selected. In short, the user should be aware of this.

Reference Alternative *(multinomial logit model only)*

For the multinomial logit model, the user should specify which choice is to be used as the reference. The constant and the coefficients for the reference choice will automatically be 0. The user should specify a particular choice from the list of available alternatives or select the most frequently used alternative as the reference choice. Keep in mind that the coefficients will change depending on which alternative is selected as the reference choice since a comparison is always relative. This will affect the interpretation of the coefficients though not the estimated probabilities.

For the conditional logit model, however, there is no reference choice. Therefore, this field will be blanked out when the type of discrete choice model is conditional.

Case ID (conditional logit model only)

When a conditional logit model is estimated, each case contributes multiple records to the data file (as many as there are alternatives). In order for *CrimeStat* to know which records belong to the same case (decision maker), the user must specify a Case ID variable, i.e. a variable that uniquely identifies cases (decision makers). If the data set was created with the *CrimeStat* 'Create Data set for Conditional Logit Model' routine, the variable is the Case ID variable specified in that routine. CrimeStat will check the number of alternatives per case, and only estimate the conditional logit model if all cases have an equal number of alternatives. If the number of alternatives per case is not equal, CrimeStat will issue an error message upon the start of the estimation.

Output for the Discrete Choice Model

The output includes both summary statistics and individual variable coefficients estimates. The output will vary between the multinomial logit and conditional logit models.

Discrete Choice Model Summary Statistics

The summary statistics include:

Information about the model

1. Date and time
2. The data file
3. The dependent (choice) variable
4. The number of records
5. The degrees of freedom
6. The type of choice model (multinomial discrete or conditional discrete)
7. Number of alternatives. For both the multinomial logit model and the conditional logit model, the routine will internally determine the number of alternatives.
8. The method of estimation (MLE – maximum likelihood estimation)

Discrete choice model likelihood statistics

9. Log likelihood estimate, which is a negative number. For a set number of independent variables, the smaller the log likelihood (i.e., the most negative) the better.
10. Log likelihood per case. Smaller (more negative) values are better. This is useful when comparing a similar model but with different numbers of records.

11. Akaike Information Criterion (AIC) adjusts the log likelihood for the degrees of freedom. The smaller the AIC, the better.
12. AIC per case. Smaller values are better.
13. Bayesian Information Criterion (BIC), sometimes known as the Schwartz Criterion (SC), adjusts the log likelihood for the degrees of freedom. The smaller the BIC, the better.
14. BIC per case. Smaller values are better.
15. Mean Absolute Deviation (MAD). For a set number of independent variables, a smaller MAD is better.
16. Mean Squared Predictive Error (MSPE). For a set number of independent variables, a smaller MSPE is better.

Discrete Choice Individual Coefficients Statistics

There is different coefficient output for the multinomial logit model than for the conditional logit model. The multinomial logit model will output constants ***and*** individual coefficients for each of *J*-1 alternatives (where *J* is the total number of alternatives). The constant and coefficients for the reference alternative are automatically defined as zero (0). For example, if there are four alternatives, then three sets of equations will be output, one for each of the *J-1* (4-1=3) alternatives. The coefficients are always relative to the reference alternative. Therefore, a positive coefficient indicates that the independent variable contributes more for that alternative than for the reference alternative while a negative coefficient indicates that the independent variable contributes less for that choice than for the reference choice. The significance test of the coefficient indicates whether the difference is statistically significant or not compared to the reference alternative. Note that the multinomial logit model *always* has a constant.

On the other hand, the conditional logit model will output a single set of individual coefficients with ***no*** constant. There is no reference choice and the coefficients are relative to not choosing a particular alternative (i.e., having a value of 0 for CHOSEN).

For the individual coefficients, the following are output for each independent variable:

1. The coefficient.
2. The standard error of the coefficient.
3. t-value.
4. p-value. This is the two-tail probability level associated with the t-test.
5. Odds ratio. This is the exponentiation of the coefficient (i.e., e^{β}). It indicates the change in the odds of that alternative (relative to the reference alternative in the multinomial model, and relative to 0 in the conditional logit model) caused by a one-unit increase in the independent variable.

Average predicted probability

For the conditional logit model only, an additional table is output that indicates the average predicted probability of the model for those cases that were selected (i.e., in which CHOSEN=1), for those cases that were not selected (i.e., CHOSEN=0), and for all cases. The number of records associated with each category is indicated as well as the standard deviation.

Table 22.1 show the output for two of the weapon alternatives for a multinomial logit model predicting weapon use during 2006 Houston robberies. Only the first two weapon alternatives (bodily force and firearms) are shown.

Multicollinearity Among Independent Variables in the Discrete Choice Model

A major consideration in any regression model (including discrete choice) is that the independent variables are statistically independent. Non-independence is called *multicollinearity* and means that there is overlap in prediction among two or more independent variables. This can lead to uncertainty in interpreting coefficients as well as to an unstable model that may not hold in the future. Generally, it is a good idea to reduce multicollinearity as much as possible.

A tolerance test is given for each coefficient. This is defined as 1 – the R-square of the independent variable predicted by the remaining independent variables in the equation using an Ordinary Least Squares model. It is an indicator of how much the remaining variables in a model account for the variance of any particular independent variable. Since the method uses the Ordinary Least Squares (OLS) methods, it is an approximate (pseudo) test for the discrete choice routines. OLS assumes normality and constant residual errors. However, many independent variables are not normally distributed (e.g., income, distance traveled, number of persons living in poverty). Consequently, the use of OLS to test for multicollinearity is exact only when the independent variable being examined for tolerance is normally distributed; otherwise, it is an approximate test. Nevertheless, it is useful indicator of multicollinearity. If the tolerance is low, that definitely indicates that there is multicollinearity. On the other hand, a high tolerance level does not necessarily indicate that there is little multicollinearity.

From the test, a guidance message is displayed that indicates probable or possible multicollinearity. If there is substantial multicollinearity (indicated by low tolerance values), it is a good idea is to drop one of the colinear independent variables and re-run the model.

Save Output

The output from the discrete choice model can be saved.

Table 22.1
Multinomial Logit Model Screen Output

```
Model result:
 Data file:                        Houston robberies 2007-2009.dbf
 DepVar:                           WEAPON
 N:                                3709
 Df:                               3697
 Type of choice model:             Multinomial logit model
 Number of Alternatives:           5
 Method of estimation:             MLE
      Likelihood statistics
 Log Likelihood:                   -4432.143485
      Per case:                    -1.194970
 AIC:                              8936.286971
      Per case:                    2.409352
 BIC/SC:                           9160.153603
      Per case:                    2.469710
      Model error estimates
 Mean absolute deviation:          0.319935
 Mean squared predicted error:     0.184770
-------------------------------------------------------------------------------
 Predictor     Coefficient  Stand Error   t-value      p-value   Odds Ratio
-------------------------------------------------------------------------------
      Bodily force
      Alternative N=1184
  Constant     0.538440     0.005424     99.266258      0.001     1.713332
   PRIMAGE    -0.001171     0.002809     -0.416833       n.s.     0.998830
PRIMGENDER     0.054200     0.005423      9.994519      0.001     1.055695
  HISPANIC     0.236188     0.005416     43.606294      0.001     1.266412
     BLACK     0.580160     0.005414    107.168407      0.001     1.786324
NUMSUSPCTS    -0.134192     0.005376    -24.959032      0.001     0.874422
COMMERCIAL     0.296323     0.005417     54.700174      0.001     1.344904
     NIGHT     0.086105     0.005421     15.884819      0.001     1.089921
 AFTERNOON     0.463824     0.005416     85.641166      0.001     1.590144
   MORNING     0.338060     0.005422     62.350883      0.001     1.402224
CL_MDHHINC     0.000011     0.000003      4.135364      0.001     1.000011
  DISTANCE    -0.022691     0.004519     -5.021083      0.001     0.977565
-------------------------------------------------------------------------------
      Firearm
      Alternative N=1744
  Constant     0.495888     0.005424     91.425683      0.001     1.641956
   PRIMAGE    -0.027863     0.002846     -9.791397      0.001     0.972521
PRIMGENDER    -0.930784     0.005424   -171.601872      0.001     0.394245
  HISPANIC     1.031718     0.005415    190.544383      0.001     2.805882
     BLACK     1.397967     0.005412    258.286949      0.001     4.046965
NUMSUSPCTS     0.177425     0.005363     33.080392      0.001     1.194139
COMMERCIAL     0.643070     0.005416    118.730846      0.001     1.902312
     NIGHT     0.392673     0.005418     72.470517      0.001     1.480934
 AFTERNOON    -0.211853     0.005416    -39.114246      0.001     0.809084
   MORNING     0.167169     0.005421     30.835798      0.001     1.181955
CL_MDHHINC     0.000005     0.000003      1.970048      0.050     1.000005
  DISTANCE     0.028941     0.004351      6.652022      0.001     1.029364
-------------------------------------------------------------------------------
```

Saved Multinomial Logit Output

For the multinomial logit model, the output is a 'dbf' file that includes all the input variables along with the estimated probability for each choice and the residual error for each choice (the observed choice, 1 or 0, minus the predicted probability). The probability and residual error is presented for each of the *J* alternatives. These are labeled with a 'P_ ' for probability and 'R_' for residual error. The different alternatives are indicated by a subscript from 0 (for the reference choice) through *J-1* (for the other alternatives) in the same order in which they are listed in Reference Choice dialogue (excluding the reference choice itself). For example, P_Choice0 is the estimated probability for choice 0 (the reference choice) while R_Choice3 is the estimated residual error for choice 3 (the third one listed in the list under Reference Choice excluding the reference choice itself). Table 22.2 shows the first 25 records of the file output from the Multinomial Logit model.

Saved Conditional Logit Output

For the conditional logit model, the output is a 'dbf' file and includes all the input variables along with the estimated probability and the residual error for the case. For each case ID, there will be only one record that was chosen. Further, since the conditional logit model produces only one equation, there is only one probability and one residual error. The probability is labeled PREDPROB and the residual error is labeled RESID. The residual error can be used to compare different models. The MAD and MSPE statistics (discussed above) summarize the residual errors. But, a user might want to plot the residuals against one of the independent variables to see if it the errors are continuous and increasing (well behaved). A bizarre error pattern can usually indicate that an independent variable is not appropriate.

Specify a directory where the output file is to be saved and provide a root name. The saved file for the multinomial logit model will have a DCOutMNL prefix while the saved file for the conditional logit model will have a DCOutCNL prefix before the user defined root name.

Table 22.3 shows the first 32 records from the file output for the conditional logit model that was set up in Figure 22.9 the output of which is display in Figure 22.11. This file copies the input records and adds the predicted probability (PREDPROB) for each case-alternative combination. For example, for Case 1 the probability of choosing TAZ 403 equals 0.000370. Note that within these first 32 zones, the probability of Case 1 choosing TAZ 429 is highest (0.046303),which happens to be the TAZ actually chosen by the offender (CHOSEN=1).

Table 22.2:
File Output from Multinomial Logit Model
First 25 Records

Make prediction:

Data file: Houston robberies 2010.dbf
Type of discrete choice model: Multinomial discrete model
N: 3709

Predicted Probabilities

Case ID	Choice0	Choice1	Choice2	Choice3	Choice4
1	0.056060	0.370066	0.331705	0.073570	0.168599
2	0.096763	0.431871	0.365920	0.060182	0.045264
3	0.082294	0.316838	0.508919	0.049205	0.042744
4	0.183496	0.380540	0.208544	0.152571	0.074849
5	0.092852	0.248848	0.570410	0.045357	0.042533
6	0.054154	0.410175	0.446969	0.036294	0.052408
7	0.043498	0.405445	0.451540	0.029337	0.070181
8	0.083722	0.252532	0.522326	0.118092	0.023329
9	0.082219	0.156078	0.665132	0.077454	0.019117
10	0.080632	0.448033	0.371738	0.048678	0.050919
11	0.086503	0.273349	0.552494	0.045244	0.042410
12	0.144867	0.576979	0.041781	0.187909	0.048464
13	0.048195	0.159970	0.734329	0.020854	0.036652
14	0.107029	0.195817	0.633713	0.044797	0.018644
15	0.115121	0.322193	0.338518	0.168298	0.055870
16	0.090629	0.491720	0.283552	0.071254	0.062845
17	0.078795	0.591412	0.262103	0.042796	0.024894
18	0.122961	0.270860	0.446626	0.127957	0.031596
19	0.074225	0.261177	0.516627	0.094802	0.053169
20	0.156918	0.364621	0.132714	0.280764	0.064982
21	0.052718	0.322463	0.475312	0.032347	0.117159
22	0.081029	0.416482	0.297664	0.133562	0.071264
23	0.114424	0.425378	0.377130	0.070873	0.012195
24	0.081482	0.400866	0.316524	0.126742	0.074385
25	0.185771	0.322145	0.298299	0.111579	0.082205

Table 22.3:
File Output from Conditional Logit Model
First 32 records

CASE	TAZ	AREA	ARTERIAL	COMMACRES	DIST_CBD	DISTANCE	CHOSEN	PREDPROB
1	401	35.97	0.00	14.01	28.01	29.95	0	0.000000
1	402	37.64	13.65	54.58	26.96	34.87	0	0.000000
1	403	8.23	6.66	66.95	21.63	23.04	0	0.000370
1	404	11.10	2.96	0.00	22.42	24.90	0	0.000042
1	405	25.22	12.91	11.08	24.43	26.95	0	0.000001
1	406	21.48	10.70	7.26	20.73	21.92	0	0.000003
1	407	9.40	9.95	54.11	20.18	19.40	0	0.000410
1	408	10.26	0.65	0.00	19.31	18.52	0	0.000091
1	409	4.87	2.48	0.00	16.97	15.38	0	0.000795
1	410	5.49	0.38	0.00	18.28	17.80	0	0.000441
1	411	3.23	0.00	0.00	17.03	16.16	0	0.001030
1	412	4.43	2.38	2.57	19.28	20.98	0	0.000511
1	413	2.56	2.78	2.90	16.80	18.97	0	0.001039
1	414	3.03	1.52	1.66	16.09	18.66	0	0.000781
1	415	7.62	0.00	0.00	18.23	18.97	0	0.000175
1	416	4.13	1.98	0.00	17.05	15.44	0	0.000983
1	417	5.01	0.82	0.00	16.47	15.23	0	0.000655
1	418	8.85	4.72	1.36	22.32	27.84	0	0.000065
1	419	11.00	3.07	8.28	19.66	24.66	0	0.000038
1	420	11.93	2.51	0.36	17.48	18.81	0	0.000047
1	421	4.68	5.87	20.41	14.96	17.75	0	0.000773
1	422	4.41	2.87	15.36	17.13	23.06	0	0.000360
1	423	3.27	0.22	0.00	15.49	21.08	0	0.000420
1	424	5.27	0.36	28.30	14.03	16.51	0	0.000512
1	425	0.88	0.00	62.12	14.35	10.67	0	0.007878
1	426	0.52	0.00	10.82	13.45	10.26	0	0.004882
1	427	0.37	0.00	0.00	12.84	9.46	0	0.004833
1	428	0.80	0.00	0.00	13.56	10.26	0	0.003989
1	***429***	***0.40***	***0.00***	***201.95***	***12.76***	***9.23***	***1***	***0.046303***
1	430	3.83	0.00	21.03	15.02	12.46	0	0.001520
1	431	0.23	0.67	19.12	14.70	12.15	0	0.005282
1	432	0.70	0.00	0.00	14.79	11.92	0	0.003635

Save Estimated Coefficients

The coefficients from either the multinomial logit or the conditional logit models can be saved for use with other data sets. Specify a directory where the coefficients file is to be saved and provide a root name. The saved coefficients file for the multinomial logit model will have a DCCoeffMNL prefix while the saved coefficients file for the conditional logit model will have a DCCoeffCNL prefix before the user defined root name.

Example of Running a Multinomial Logit Model

To illustrate the process of running a multinomial logit model, we model the premises chosen for Chicago residential robberies for 1997. Figure 22.2 shows setting up the multinomial logit model including reading in the data file as the 'Other' file, defining the choice variable (PREMISES) and the selection of the predictors from the list of available independent variables (GUNCRIME, EVENING, LATENIGHT, TRAVELDIST, OFFAGE, and OFFBLACK). Finally, Figure 22.3 shows the screen output of the multinomial logit model.

Example of Creating and Running a Conditional Logit Model

To illustrate the process of creating a file for the conditional logit model and then running a file to estimate the predictors of the alternatives, we use an example of predicting which Traffic Analysis Zone (TAZ) offenders use to commit crimes. In Figure 22.4, the case file, which contains the origin TAZ and destination TAZ of each of 500 offences, is input as the Primary File and the coordinates of each crime location are input as the X and Y coordinates.

In Figure 22.5, the alternatives file is the information on the 325 TAZs themselves. This is input on the Secondary File page and the coordinate for each TAZ are defined. In figure 22.6, both the case file and the alternatives file are defined for the 'Create data set for conditional logit model' routine. The case file is defined by the Primary File with the case ID being CASE. The alternatives file (the TAZs) are defined by the secondary File with the alternative ID being TAZ. The 'Calculate distance between cases and alternatives' box is checked and the distance units will be calculated in miles.

Figure 22.7, the file for the created file is defined (CasesXAlternatives.dbf). Once the user calculates 'Compute', the routine runs. When it has finished, it gives a 'File saved' message (Figure 22.8).

The user should be sure to uncheck the 'Create data set for conditional logit model' routine box. Then, either the created file or another file prepared by the user is input as the Primary File. On the Discrete Choice I page, the 'Estimate Model' box is checked and the conditional logit model is set up. The dependent variable is CHOSEN if the file was created by the cross-joined file, and can have any name if it was prepared by the user. The dependent variable must be a binary (0/1) variable. Subsequently, several appropriate predictor variables are selected from the independent variables list (Figure 22.9). In the Conditional Logit example they are AREA, ARTERIAL, COMMACRES, DIST_CBD and DISTANCE An output file is then defined to save the results of the conditional logit model (Figure 22.10). Once the conditional logit model is run, the screen output can be viewed (Figure 22.11).

Figure 22.2:

Example of Running a Multinomial Logit Model

Step 1: Setting Up the Multinomial Logit Model

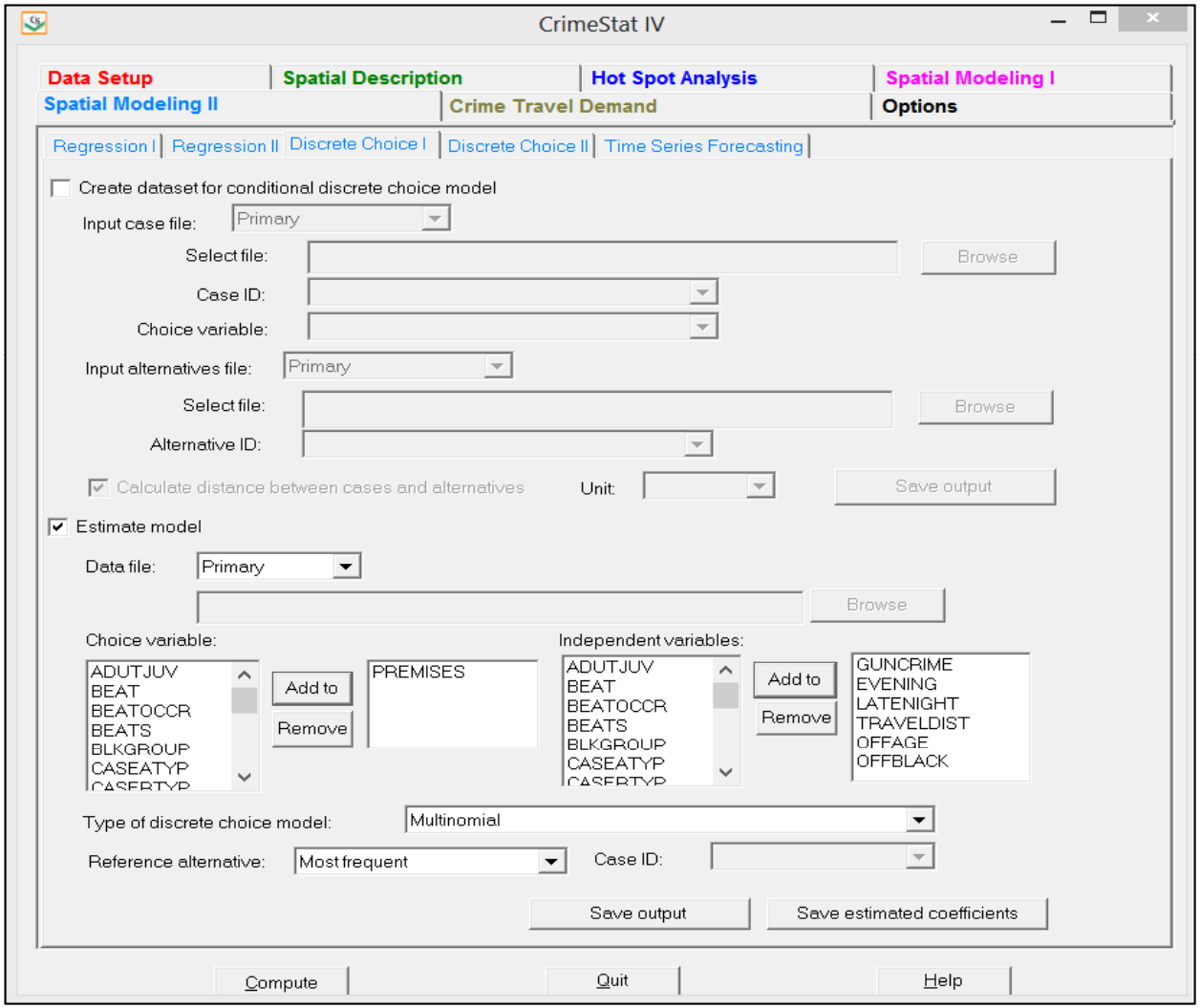
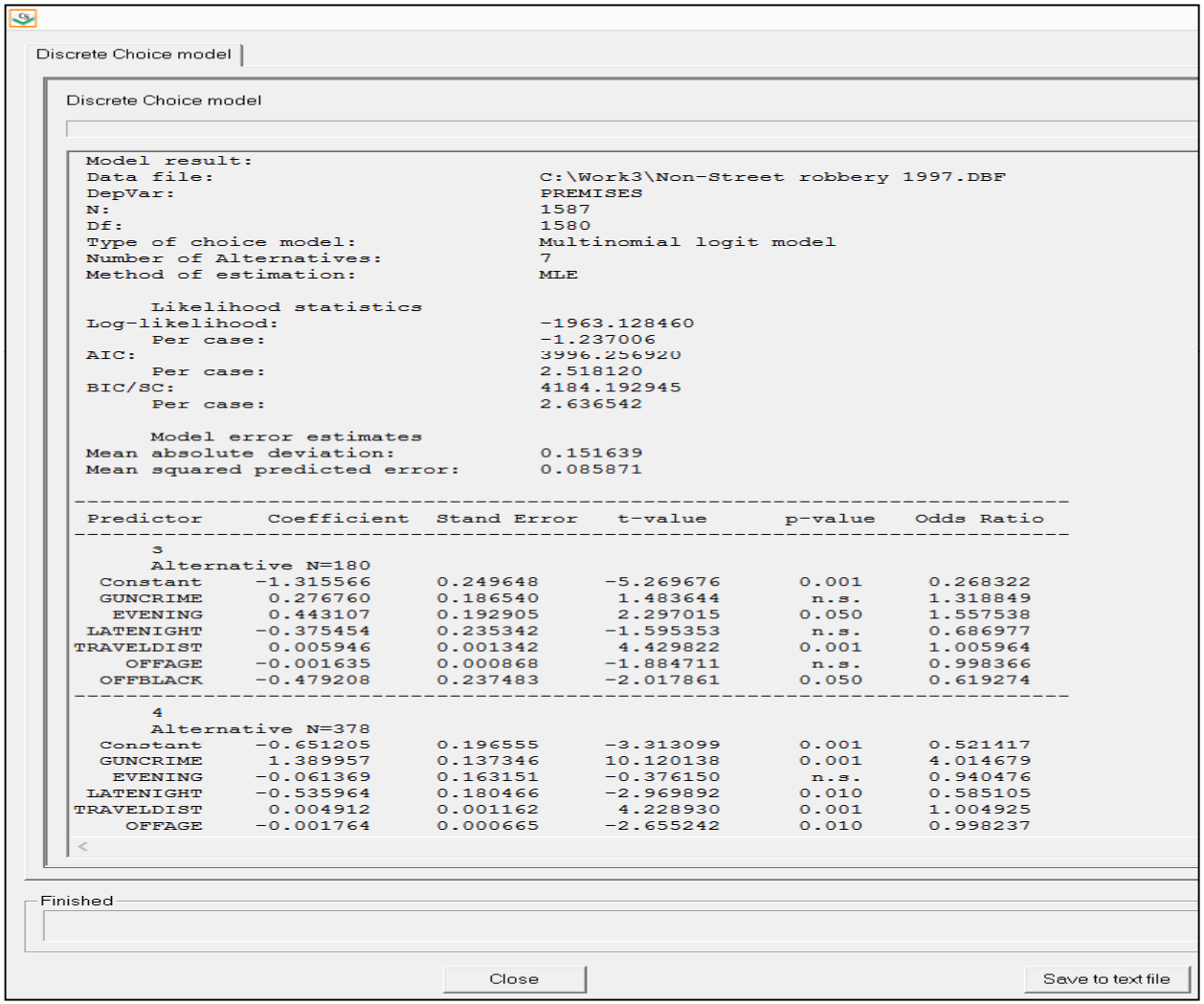


Figure 22.3:

Example of Running a Multinomial Logit Model

Step 2: Results of the Multinomial Logit Model Estimate



Discrete Choice model

Discrete Choice model

```
Model result:
Data file:                          C:\Work3\Non-Street robbery 1997.DBF
DepVar:                             PREMISES
N:                                  1587
Df:                                 1580
Type of choice model:               Multinomial logit model
Number of Alternatives:             7
Method of estimation:               MLE

      Likelihood statistics
Log-likelihood:                     -1963.128460
      Per case:                     -1.237006
AIC:                                3996.256920
      Per case:                     2.518120
BIC/SC:                             4184.192945
      Per case:                     2.636542

      Model error estimates
Mean absolute deviation:            0.151639
Mean squared predicted error:       0.085871

-------------------------------------------------------------------------------
 Predictor      Coefficient  Stand Error    t-value       p-value    Odds Ratio
-------------------------------------------------------------------------------
      3
      Alternative N=180
  Constant      -1.315566     0.249648     -5.269676       0.001      0.268322
 GUNCRIME        0.276760     0.186540      1.483644        n.s.      1.318849
  EVENING        0.443107     0.192905      2.297015       0.050      1.557538
LATENIGHT       -0.375454     0.235342     -1.595353        n.s.      0.686977
TRAVELDIST       0.005946     0.001342      4.429822       0.001      1.005964
   OFFAGE       -0.001635     0.000868     -1.884711        n.s.      0.998366
OFFBLACK        -0.479208     0.237483     -2.017861       0.050      0.619274
-------------------------------------------------------------------------------
      4
      Alternative N=378
  Constant      -0.651205     0.196555     -3.313099       0.001      0.521417
 GUNCRIME        1.389957     0.137346     10.120138       0.001      4.014679
  EVENING       -0.061369     0.163151     -0.376150        n.s.      0.940476
LATENIGHT       -0.535964     0.180466     -2.969892       0.010      0.585105
TRAVELDIST       0.004912     0.001162      4.228930       0.001      1.004925
   OFFAGE       -0.001764     0.000665     -2.655242       0.010      0.998237
```

Finished

Close

Save to text file

Figure 22.4:

Example of Setting Up and Running a Conditional Logit Model

Step 1: Inputting a Case File as the Primary File

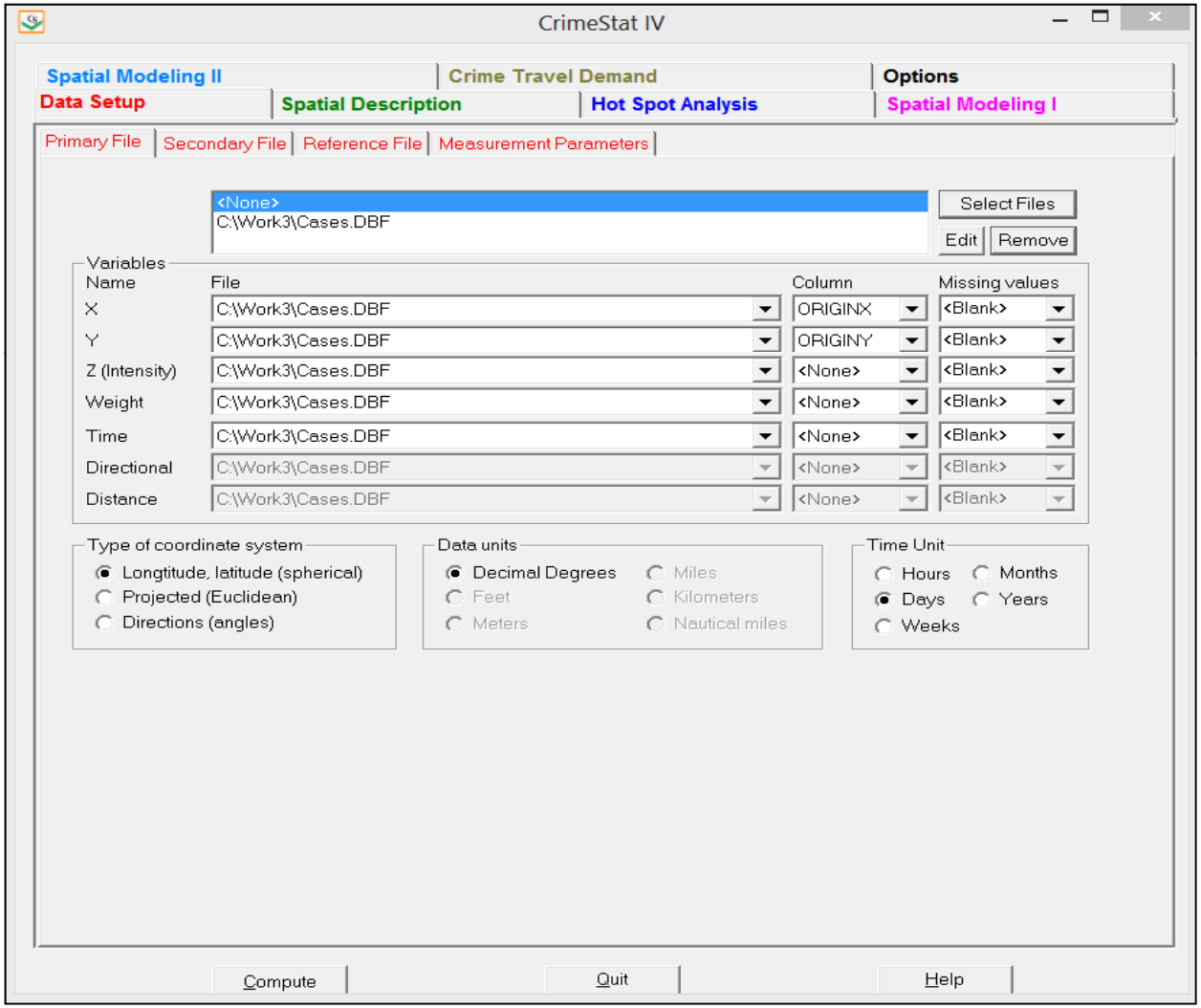


Figure 22.5:

Example of Setting Up and Running a Conditional Logit Model

Step 2: Inputting an Alternatives File as the Secondary File

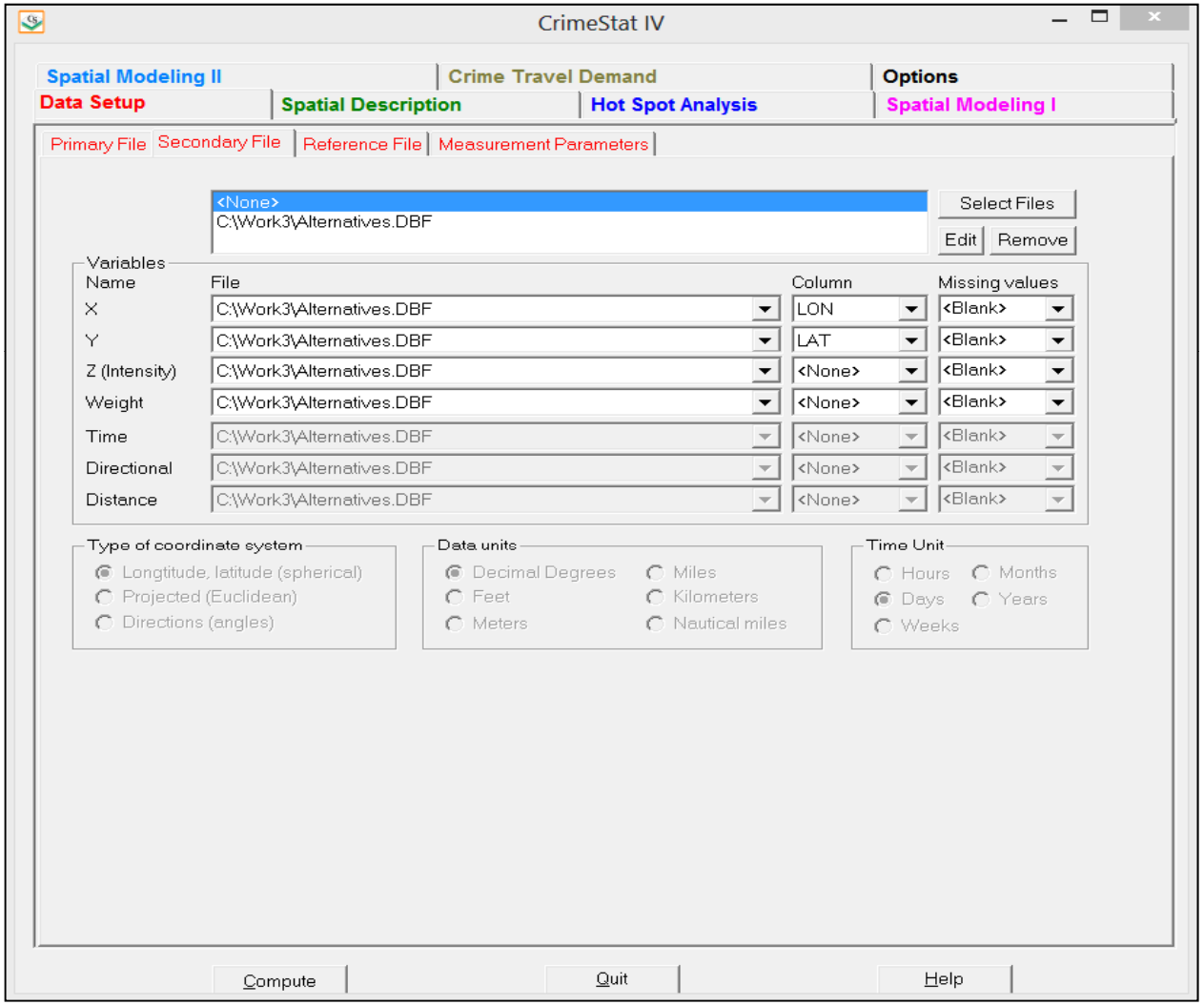


Figure 22.6:

Example of Setting Up and Running a Conditional Logit Model

Step 3: Setting Up the Routine for Creating a Conditional Logit Database

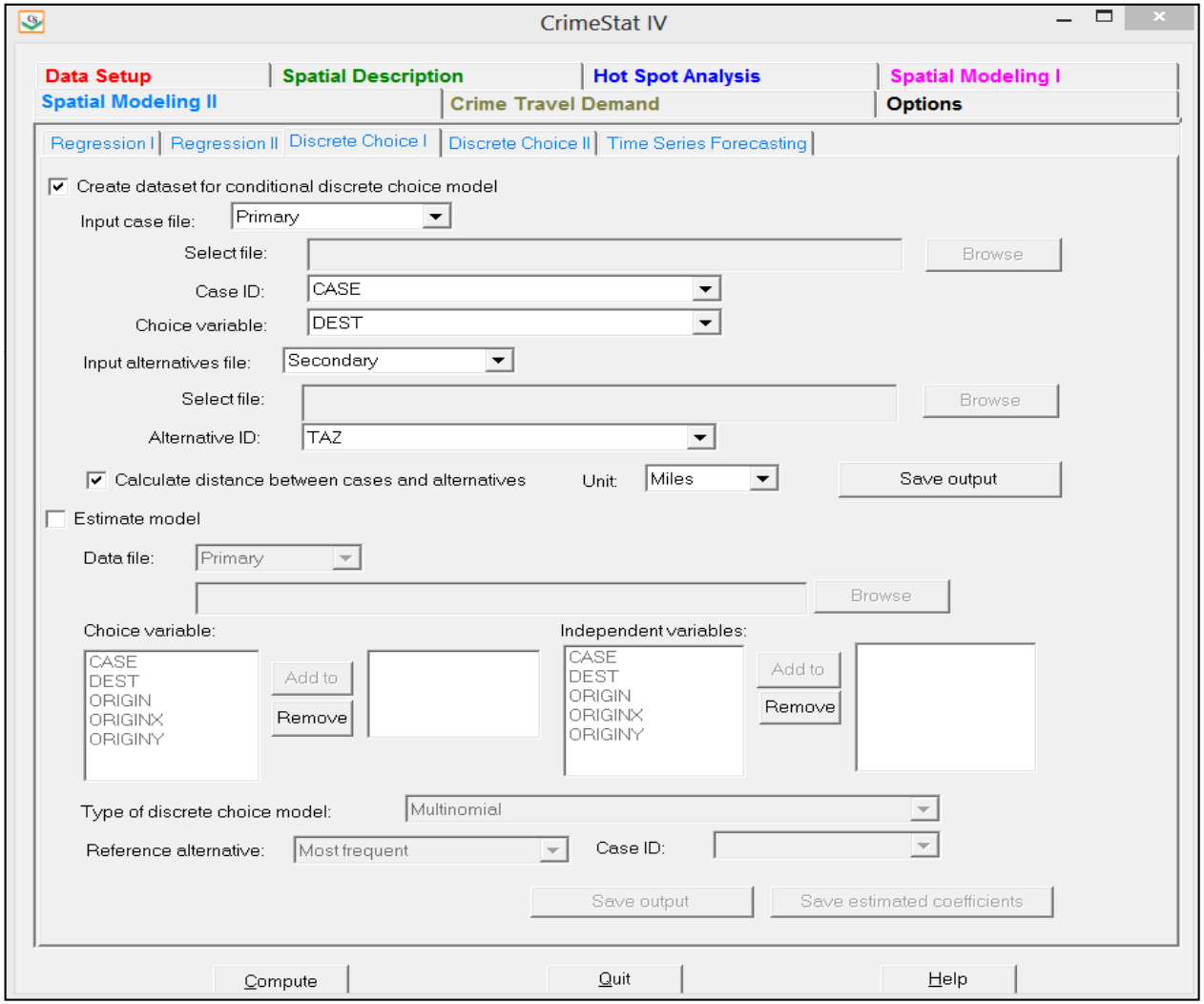


Figure 22.7:

Example of Setting Up and Running a Conditional Logit Model

Step 4: Choosing a File Name to Save the Created File

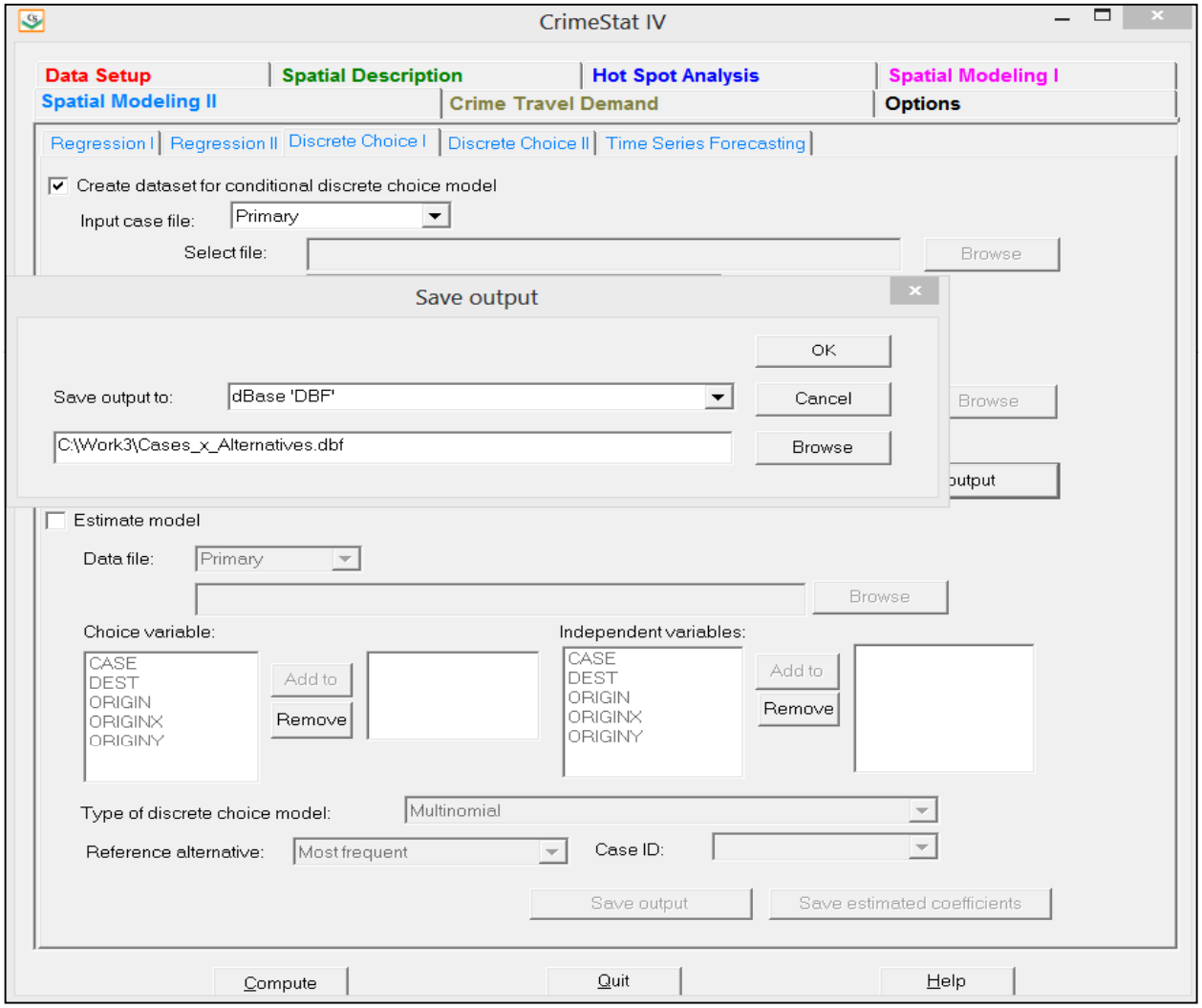


Figure 22.8:

Example of Setting Up and Running a Conditional Logit Model

Step 5: Running the Routine to Create the File

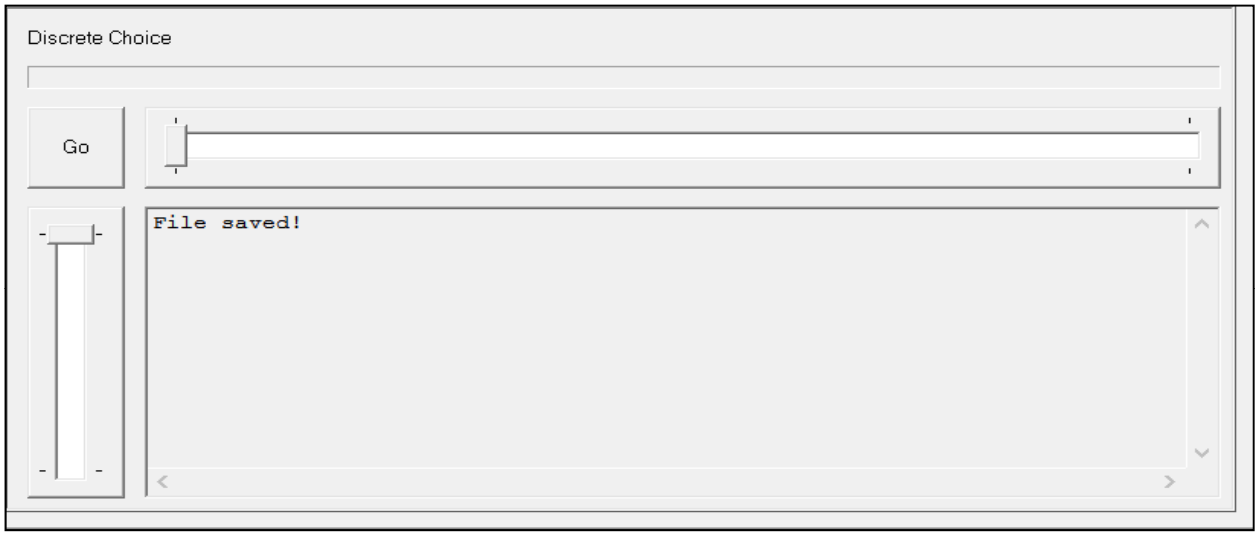


Figure 22.9:

Example of Setting Up and Running a Conditional Logit Model

Step 6: Setting Up Conditional Logit Model

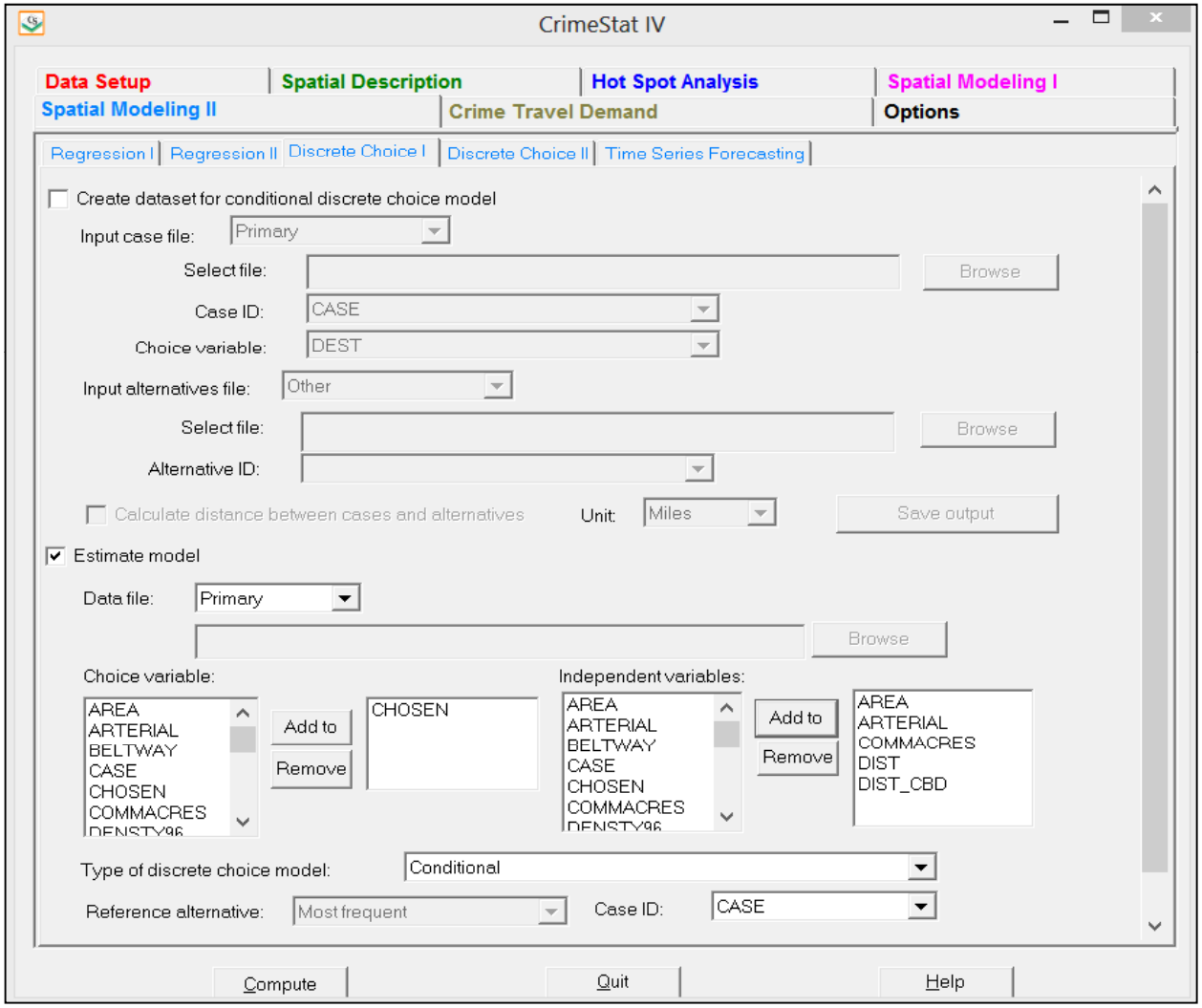


Figure 22.10:

Example of Setting Up and Running a Conditional Logit Model

Step 7: Defining Output File for the Conditional Logit Model

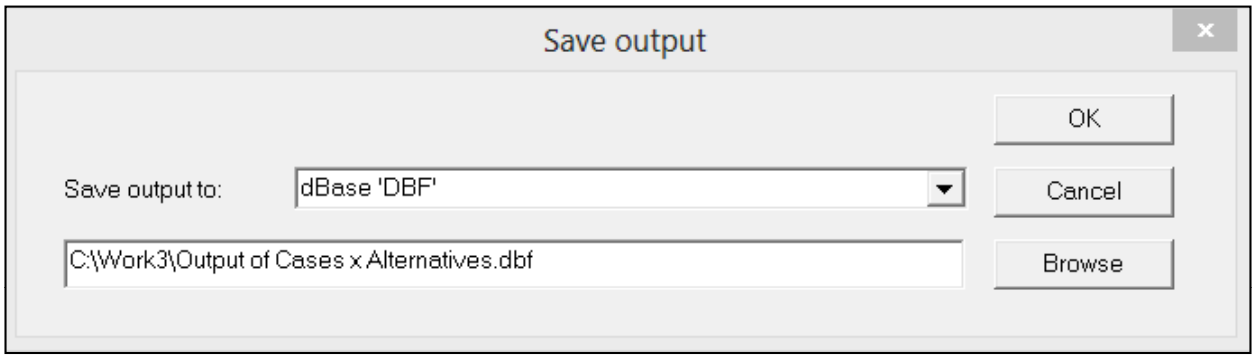
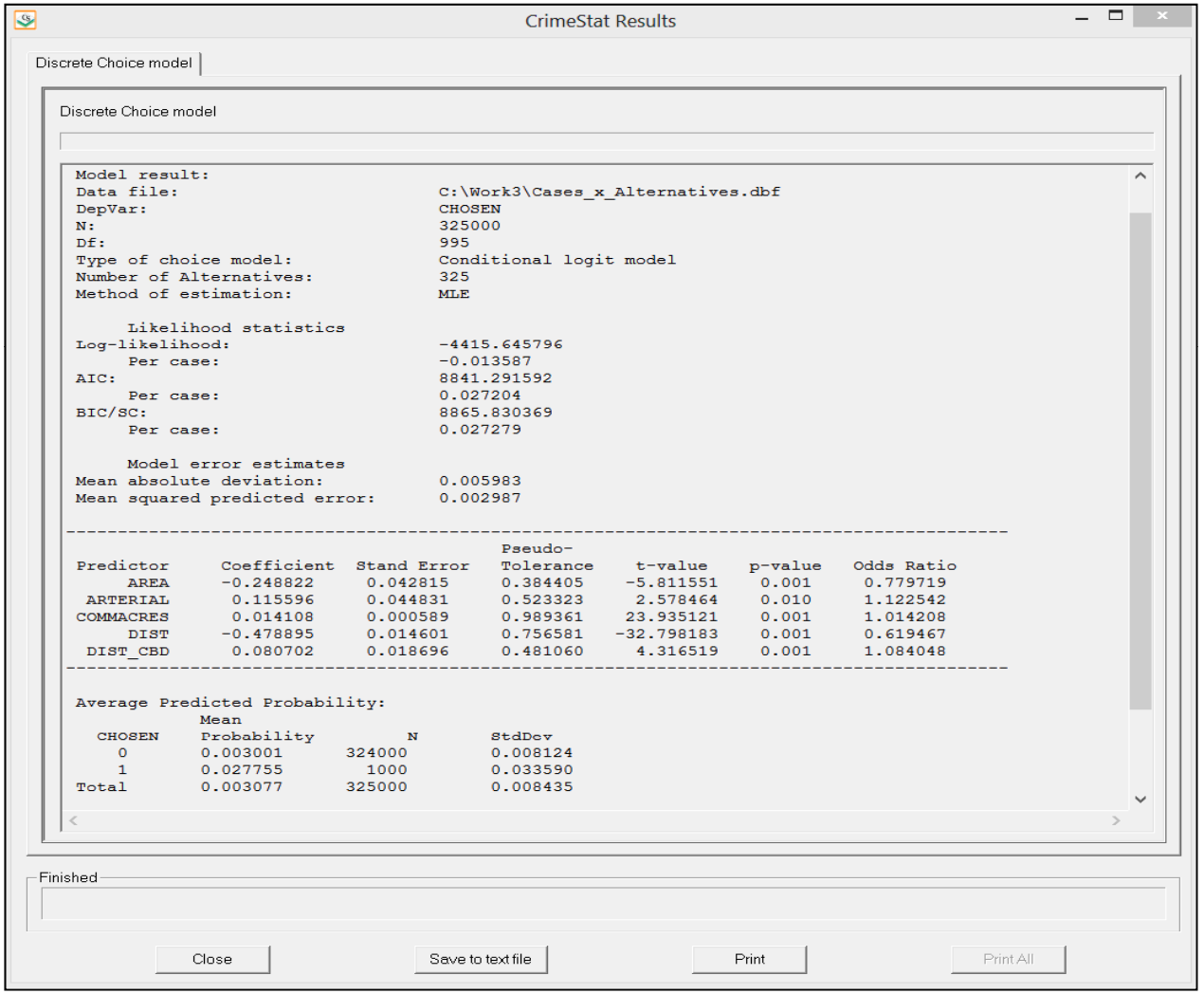


Figure 22.11:

Example of Setting Up and Running a Conditional Logit Model

Step 8: Screen Output for the Conditional Logit Model



Discrete Choice Modeling II

The Discrete Choice modeling II module allows the user to apply the estimated coefficients from a discrete choice model to another data set (or a subset of the same data set) and calculate predicted probabilities, for either the multinomial logit or the conditional logit model. The 'Make prediction' routine allows the application of coefficients to a data set. The saved coefficients are applied to similar independent variables and to corresponding values of the choice variable to produce an estimated probability of an alternative.

Make Prediction

Figure 22.12 show the interface for the Discrete Choice II page. There are two types of models that can be fitted – multinomial logit or conditional logit. For both types of model, the coefficients file must include information on each of the coefficients. In addition, the coefficients model for the multinomial must include the value of the constant. If the coefficients file was generated by CrimeStat on the Discrete Choice I page, then all the necessary information will be included. The user reads in the saved coefficient file and matches the variables to those in the new data set based on the order of the coefficients file.

Discrete Choice Saved Coefficients File

In order to make a prediction, a model must have already been calibrated and the coefficients saved in a coefficients file. Point to the directory where the coefficients file has been saved and identify it.

Available Variables

The box labeled 'Available variables' will list all the fields on the input data set.

Independent Predictors

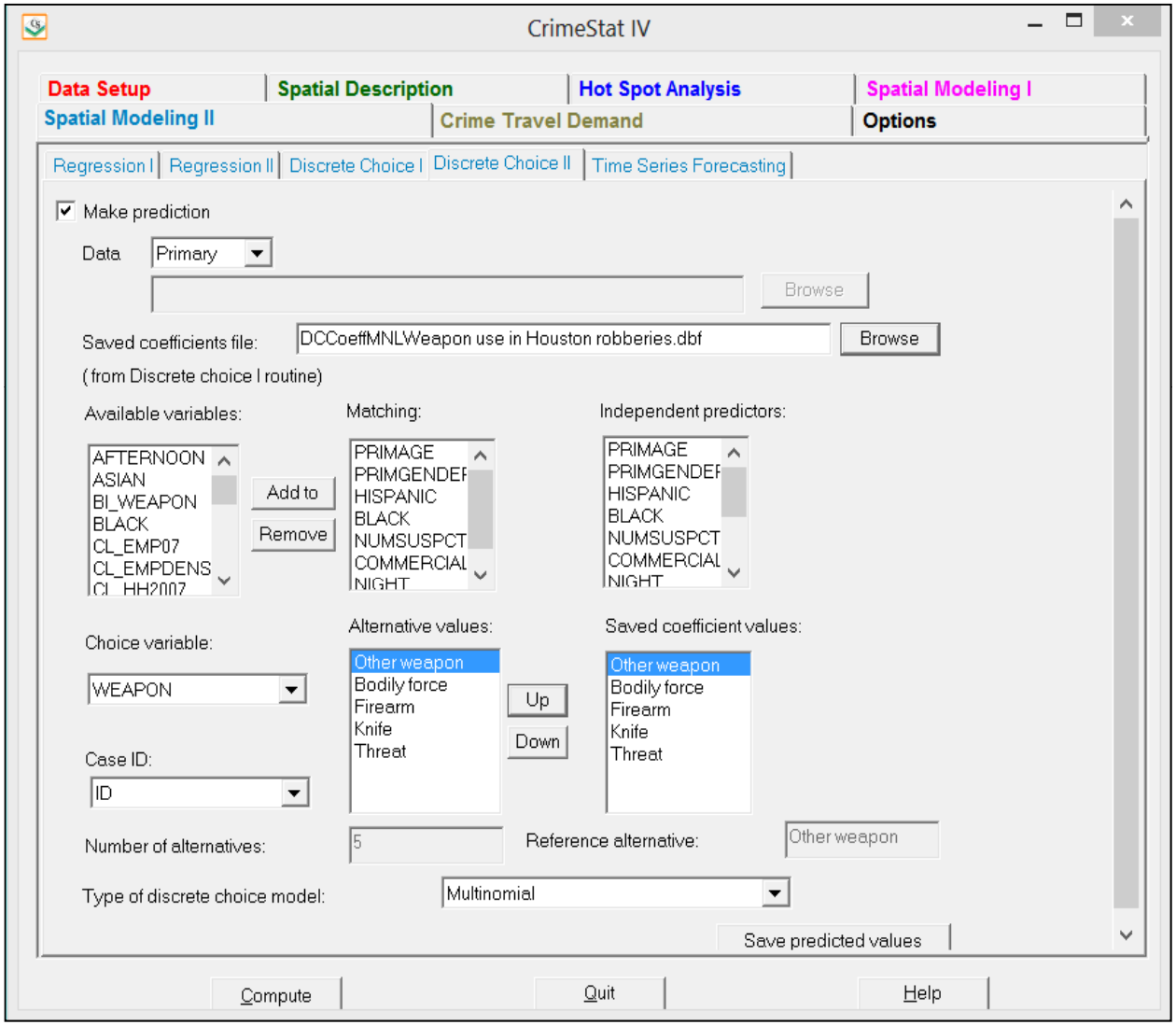
The independent variables that were used in the calibrated coefficients file will be listed in the right column. They will be in the same order as was estimated in the calibration file.

Matching variables

Select corresponding variables from the input data file for the middle column. The items should be listed in the same order as in the 'independent predictors' column. They should be

Figure 22.12:

Discrete Choice Modeling II



similar variables in content but need not have the names as in the original data file. Figure 22.13 shows an example of setting up a multinomial logit prediction model using an already estimated multinomial logit model from another data set. The user reads in the data file and then already-saved coefficients from the earlier calibration and then matches the variable names in the new data set with the saved names from the already calibrated model. In the example, the variable names for travel distance were different in the two files.

Figure 22.14 shows an analogous example of setting up a conditional logit prediction. Again, the variable names in the input file on which the prediction is to be calculated (AREA, ARTERIAL, etc.), are the same as those in the file on which the coefficients were estimated. This also holds for the ID variable name, which must be specified in case of a conditional logit prediction.

Alternative values (multinomial logit model only)

The values of the choice variables from the input file will be displayed in the middle column. The order should match the values in the adjacent saved coefficients file column. The 'Up' and 'Down' buttons can be used to re-order the values to be sure they are matched exactly.

Discrete Choice Data File

The new data set can be either the Primary file or another file. If another file is being used, point to the directory where it is stored and identify it. The structure of the file for which a prediction is made must be the same as that from which the model was initially calibrated. That is, for a multinomial logit prediction, there must be a file with one record per decision maker and which includes and ID and each of the independent variables used in the prediction. For a conditional logit prediction, there must be a joined file with a record for every combination of case and alternative.

Saved coefficient values *(multinomial logit model only)*

The values of the saved coefficients file will be displayed in the right column. Additional values can be added with the "Add to" button and existing values can be removed with the "Remove" button. It is essential that the values in the middle column match ***exactly*** their corresponding values in the right column.

Figure 22.13:

Example of Running a Multinomial Logit Prediction

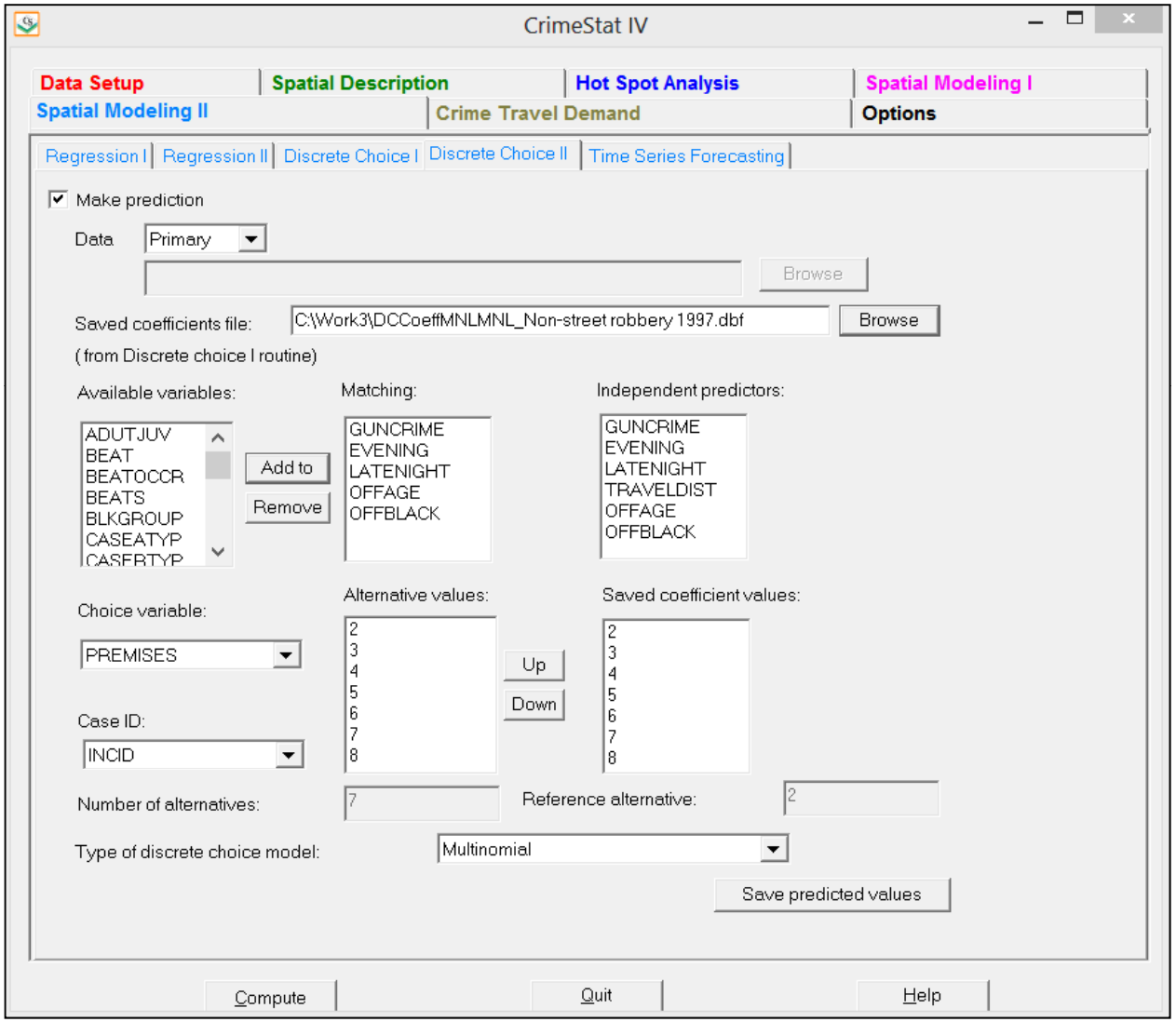
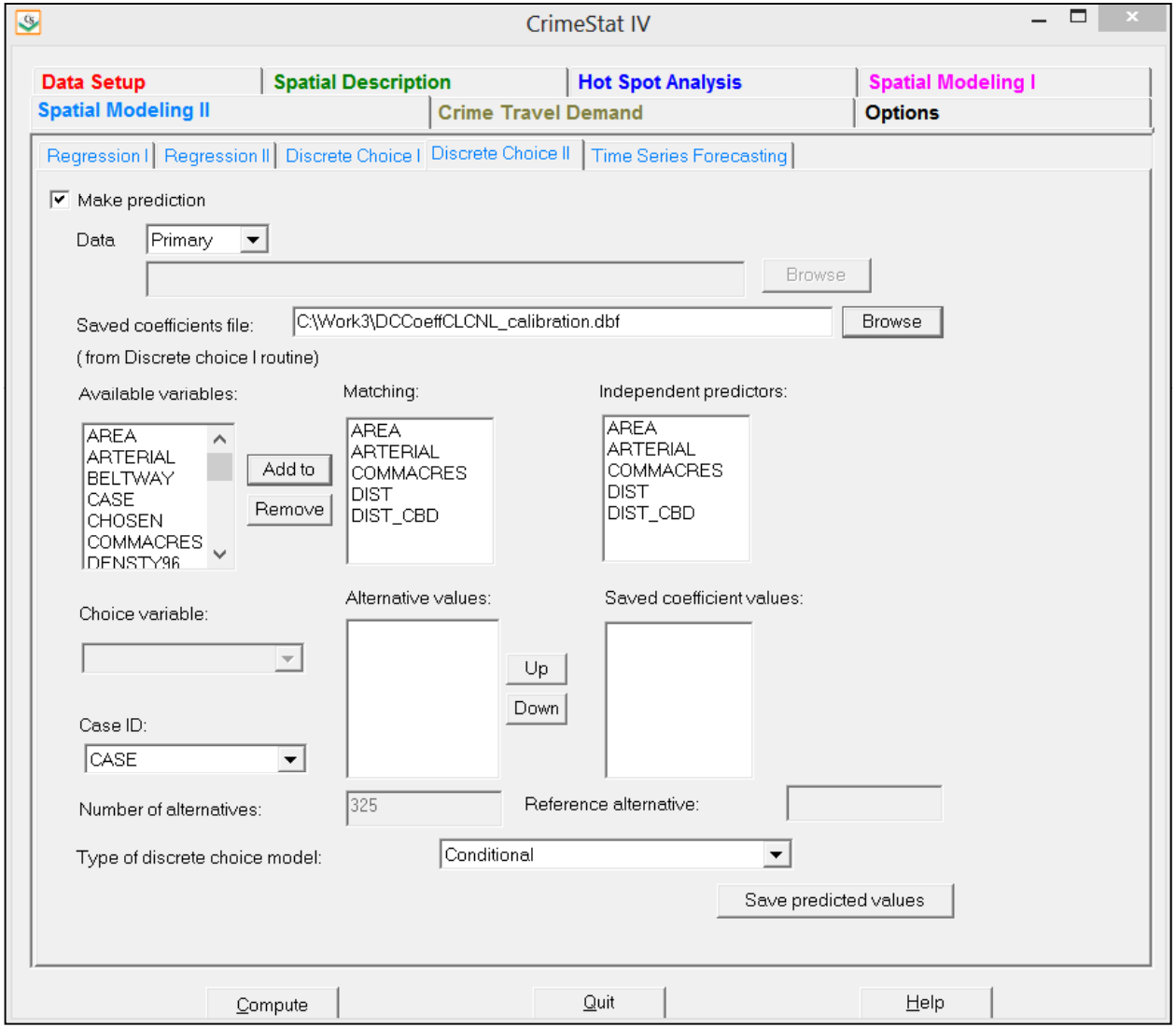


Figure 22.14:

Example of Running a Conditional Logit Prediction



Reference alternative *(multinomial logit model only)*

The reference alternative value is displayed. If it is not correct, type in the correct value to be used or, better yet, re-calibrate the original model. This field will be blanked out for the conditional logit model since it is not appropriate.

Discrete Choice Prediction Output

The screen output provides predictions of the value of the dependent variable in the same order as in the input data set. For the multinomial logit model, the predictions are labeled as CHOICE0 (for the reference choice), CHOICE1, CHOICE2, and so forth, in the same order as in the input data set. For each alternative, these predictions represent the probability that this alternative is chosen, given the values of the predictor variables.

For the conditional logit model, the prediction is applied to each available alternative. The screen output presents the predictions in matrix format with the case ID listed on the vertical axis and the choices listed on the horizontal axis (labeled CHOICE0, CHOICE1, CHOICE2, and so forth, in the same order as in the input data set).

Save Predicted Values for Discrete Choice Prediction

The predicted values and the residual errors can be output to a 'dbf' file with a DCMakePredMNL<*root name*> for the multinomial logit and DCMakePredCNL<*root name*> for the conditional logit with the root name being provided by the user. The output files differ between the multinomial and conditional logit models.

Multinomial Logit Prediction Output

For the multinomial logit prediction, there is the probability produced for each of the *J* alternatives. The probabilities are labeled P_CHOICE0 (for the reference choice), P_CHOICE1, P_CHOICE2, and so forth in the same order as in the Choice Values dialogue (with the exception of the reference alternative which is always defined as P_CHOICE0). The probabilities will sum to 1.0 for all J alternatives (within rounding-off error).

Table 22.4 shows the first 25 cases for the file output of a multinomial logit prediction of weapon use for 2010 Houston robberies. The specific alternatives are labeled Choice0, Choice1, Choice2, Choice3, and Choice4 and are the weapon categories in the same order as laid out on the interface (namely Other weapon, Bodily force, Firearm, Knife, and Threat).

Table 22.4:
File Output from Multinomial Logit Prediction Routine
First 25 records

ID	P_CHOICE0	P_CHOICE1	P_CHOICE2	P_CHOICE3	P_CHOICE4
1	0.056060	0.370066	0.331705	0.073570	0.168599
2	0.096763	0.431871	0.365920	0.060182	0.045264
3	0.082294	0.316838	0.508919	0.049205	0.042744
4	0.183496	0.380540	0.208544	0.152571	0.074849
5	0.092852	0.248848	0.570410	0.045357	0.042533
6	0.054154	0.410175	0.446969	0.036294	0.052408
7	0.043498	0.405445	0.451540	0.029337	0.070181
8	0.083722	0.252532	0.522326	0.118092	0.023329
9	0.082219	0.156078	0.665132	0.077454	0.019117
10	0.080632	0.448033	0.371738	0.048678	0.050919
11	0.086503	0.273349	0.552494	0.045244	0.042410
12	0.144867	0.576979	0.041781	0.187909	0.048464
13	0.048195	0.159970	0.734329	0.020854	0.036652
14	0.107029	0.195817	0.633713	0.044797	0.018644
15	0.115121	0.322193	0.338518	0.168298	0.055870
16	0.090629	0.491720	0.283552	0.071254	0.062845
17	0.078795	0.591412	0.262103	0.042796	0.024894
18	0.122961	0.270860	0.446626	0.127957	0.031596
19	0.074225	0.261177	0.516627	0.094802	0.053169
20	0.156918	0.364621	0.132714	0.280764	0.064982
21	0.052718	0.322463	0.475312	0.032347	0.117159
22	0.081029	0.416482	0.297664	0.133562	0.071264
23	0.114424	0.425378	0.377130	0.070873	0.012195
24	0.081482	0.400866	0.316524	0.126742	0.074385
25	0.185771	0.322145	0.298299	0.111579	0.082205

Conditional Logit Prediction Output

For the conditional logit prediction, there is a single probability output which is applied to the particular record. Since the data set for the conditional logit model has a single record for each alternative available to the decision maker, the probability applies to that alternative. The probabilities within a case will sum to 1.0 for all J alternatives (within rounding-off error). The column is labeled PREDPROB. Table 22.5 shows the first 32 cases for a CL prediction output.

Table 22.5:

File Output for Conditional Logit Prediction Routine

First 32 records

CASE	TAZ	AREA	ARTERIAL	COMMACRES	DIST_CBD	DISTANCE	PREDPROB
501	401	35.97	0.00	14.01	28.01	31.59	0.000000
501	402	37.64	13.65	54.58	26.96	34.76	0.000000
501	403	8.23	6.66	66.95	21.63	23.78	0.000331
501	404	11.10	2.96	0.00	22.42	26.45	0.000033
501	405	25.22	12.91	11.08	24.43	30.25	0.000000
501	406	21.48	10.70	7.26	20.73	25.48	0.000002
501	407	9.40	9.95	54.11	20.18	25.72	0.000158
501	408	10.26	0.65	0.00	19.31	24.38	0.000037
501	409	4.87	2.48	0.00	16.97	20.48	0.000368
501	410	5.49	0.38	0.00	18.28	25.22	0.000144
501	411	3.23	0.00	0.00	17.03	23.86	0.000322
501	412	4.43	2.38	2.57	19.28	21.17	0.000496
501	413	2.56	2.78	2.90	16.80	19.37	0.000979
501	414	3.03	1.52	1.66	16.09	18.08	0.000852
501	415	7.62	0.00	0.00	18.23	20.75	0.000134
501	416	4.13	1.98	0.00	17.05	18.92	0.000580
501	417	5.01	0.82	0.00	16.47	17.45	0.000469
501	418	8.85	4.72	1.36	22.32	26.56	0.000080
501	419	11.00	3.07	8.28	19.66	21.68	0.000059
501	420	11.93	2.51	0.36	17.48	16.77	0.000064
501	421	4.68	5.87	20.41	14.96	14.64	0.001236
501	422	4.41	2.87	15.36	17.13	19.13	0.000653
501	423	3.27	0.22	0.00	15.49	16.58	0.000830
501	424	5.27	0.36	28.30	14.03	12.03	0.001008
501	425	0.88	0.00	62.12	14.35	17.89	0.002650
501	426	0.52	0.00	10.82	13.45	17.67	0.001596
501	427	0.37	0.00	0.00	12.84	16.84	0.001586
501	428	0.80	0.00	0.00	13.56	18.02	0.001236
501	429	0.40	0.00	201.95	12.76	16.85	0.014658
501	430	3.83	0.00	21.03	15.02	20.21	0.000472
501	431	0.23	0.67	19.12	14.70	19.10	0.001849
501	432	0.70	0.00	0.00	14.79	18.71	0.001303

Chapter 23:

Time Series Forecasting

Wilpen L. Gorr
H. John Heinz III College
Carnegie Mellon University
Pittsburgh, Pennsylvania

Andreas M. Olligschlaeger
TruNorth Data Systems
Baden, Pennsylvania

Table of Contents

Chapter 23:
Time Series Forecasting

Introduction

This chapter presents time series methods useful for tactical deployment of police resources. The methods answer the following questions:

- What crime levels are expected in police zones, patrol districts, census tracts, or other areas of a jurisdiction given past time trends?

- Are there any new crime patterns, large increases or decreases, starting up in the jurisdiction?

The first question is answered using extrapolative time series models to forecast expected crime levels by geographic area. The second uses observed departures from the expected crime levels as the basis for detecting new crime patterns on an early warning basis. Automation is important for such work. For example, if a police organization has 100 census tracts and 10 crime types that it wishes to track, then it would have 1,000 time series to analyze—too much work for manual visual inspection. This chapter presents standard models and methods long-used in industry for time series forecasting and detection, making them available in highly optimized and automated computer code tailored for crime analysis.

Time series forecasting can be complex and require sophisticated software and highly-trained analysts. The good news here, however, is that the forecasting literature on operations management applications has shown that simple methods are as accurate as or more so than complex methods. Most influential have been the so called "M-Competitions" in which forecasters forecasted time series data without seeing the future data and independent judges analyzed forecast accuracy with the future data (see Makridakis et al.,1982 and Hibon & Makridakis, 2000).

This chapter presents simple methods that are easy to understand and use. These methods are self-adaptive to changing conditions, taking care of themselves in the dynamic setting of cities and actions of criminals and police. The implementation in *CrimeStat* optimizes forecast model parameters in extensive but fast algorithms, thereby making the module easy to use. The detection component has parameters that cannot be readily optimized, but the chapter provides default values from a research study on crime detection (Cohen, et al. 2009). Finally, all areas of interest such as all patrol districts in a jurisdiction are processed in a single run, again making it efficient for analysts. Outputs are easily displayed in GIS as choropleth maps. On these maps it is

desirable for analysts to also display individual crime points when zooming into areas of interest for detailed diagnosis (see Gorr and Kurland, Chapter 2, 2012). Out of the entire jurisdiction, the automated detection methods bring to attention the areas needing further analysis. Then the crime analyst zooms into those areas, one-by-one, and studies the detailed crime patterns.

The chapter starts with overviews of time series data and extrapolative forecast methods. It then presents details on exponential smoothing models, which are among the simplest but most accurate forecasting methods, along with classical decomposition for estimation of seasonal adjustments. Early detection of time series pattern changes is the final topic, covered first at the conceptual level and then as implemented in *CrimeStat*. Exponential smoothing forecasts are the basis for detection, so that all the methods in this chapter work together.

Time Series Data

This chapter uses univariate time series data, meaning that for a given observation unit, there is only a single variable:

$$y_{it} \; \mathrm{i} = 1, \ldots, \mathrm{m} \text{ and } \mathrm{t} = 1, \ldots, \mathrm{T} \qquad (23.1),$$

where

i = area identifier (e.g., patrol district, census tract)
m = total number of areas
t = time period such as month serial number
T = most recent time period, called the "forecast origin."

Service Demand

For example, in the private sector, y_{it} might be defined as the product demand by sales territory represented by the number of units of product sold per time period. For police the corresponding variable could be the number of crime incidents of a particular type per time period, usually counts of offense report incidents or computer aided dispatch (CAD) calls for service. Arrest and other incident-related data are less useful for tactical deployment of police resources. Offense reports are official records of crimes having been committed and can be either Uniform Crime Reporting (UCR) hierarchy-based in the U.S., with only the most serious crimes included, or incident-based with individual records for each crime type committed in an incident being included. Which one to use depends on the particular need required. For example, for dispatching purposes UCR data may be sufficient to determine priority of a call for service.

The question arises, should one use raw crime counts per area and time interval or crime rates which are crime counts per time interval and divided by the population at risk? The answer

for tactical deployment of police is to use raw crime counts which directly determine the size of effort needed for police deployment. Crime rate is somewhat academic in the sense that it tells something about behavior of criminals and the effects of congestion, valuable for insight and understanding, but not the needed measure for resource allocation.

Fixed time and observation units

The time interval of observations in the forecasting area can be any unit of time from hours to days, weeks, months, quarters, or years. For example, electric load forecasting needs hourly forecasts. Time series analysis (but not forecasting) of crime patterns can benefit from estimating hourly seasonality factors of crime for week days versus weekend days. While average hourly variations in crime over the 24 hours of a day can be informative, there is not a large enough sample size in crime data by zone or patrol district to yield accurate hourly forecasts.

Generally the smallest time intervals possible for crime forecasting are weeks or months, but then the time series data is still very noisy. The number of days per month of course varies, and some organizations adjust monthly time series data to the average number of days per month —multiplying by (365/12)/(days in a month), or doing a similar calculation with the number of work days (trading days) per month. This is not necessary if including seasonality in a forecast model, because the seasonal factor automatically includes an appropriate adjustment. Most crime time series data is seasonal.

The data used in this chapter might better be called “space and time series data” because for crime in a police jurisdiction there are administrative and other areas of interest, each having crime time series data and needing forecasts. The administrative areas include “zones” each with a police station, commander, field officers, and so forth operating semi-autonomously. A zone is partitioned into “patrol districts” each with a single patrol unit assigned three shifts per day. Other areas of interest include census reporting areas, such as census tracts in the U.S. Census areas which generally have a target population (e.g., 4,000 for census tracts) and are neighborhoods with common socio-economic patterns. Often in the U.S. patrol districts are made up of one or a few census tracts.

Whatever areas are of interest, the needed aggregate space and time series data can easily be generated using GIS:

1. Geocode crime incidents using street addresses.

2. Spatially join the geocoded crime incidents to polygons of the area of interest.

3. Using the date of incident, create variables for year and week or month.

4. Count crime incidents by area, year, and week or month to form the crime space and time series data.

The extrapolative forecast models of this chapter make minimum use of the spatial arrangement of the areas being forecasted. Multivariate models, not covered in this chapter, can use crime counts nearby a particular area as part of the model. Here, however, the only use of data outside of a particular area being modeled is in estimation of seasonal factors. As explained below, estimating seasonality using data from the entire jurisdiction is generally better than doing so for each individual district (whether zones or patrol districts).

Limitations

A major tradeoff associated with crime space and time series data is that police need forecasts for areas as small as possible so as to target resources precisely. However, the smaller the area, the fewer the crimes and the less reliable are the model estimates and forecasts. Past work suggests that for areas as small as census tracts, it is only possible to forecast high volume crimes or crime aggregates (e.g., all serious property crimes) with useful accuracy at the monthly level (Gorr, Olligschlaeger, & Thompson, 2003). Low volume crimes, such as homicides, cannot be forecasted accurately at all, even for an entire jurisdiction.

The inability to forecast small areas accurately is not a big a limitation as it seems, because of research on crime hot spots. Typically 50% of crime occurs in only 5% or less of the area of a city, in micro-area hot spots (e.g., Weisburd & Green, 1995, Weisburd et al, 2004; see Chapters 7, 8 and 9 of the *CrimeStat* manual). So if one has an early detection of a large crime increase, it likely will be in small areas. Then patrols and other police resources can be directed to the emerging hot spot.

Extrapolative Time Series Forecasting

Industry very likely uses more computer machine cycles for extrapolative forecasting than any other statistical method. Every week and month, firms forecast demand by product (or service type) and sales territory using mostly exponential smoothing models. The crime forecasting problem is analogous, with the count of crime incidents representing service demand for police. This section starts at the beginning of time series forecasting and detection, with general terms and notation used in the area before moving on to specific models.

Terms

While there are many specialized terms for specific forecast approaches and models, the general set, however, is not that large. Below is a collection of general terms and notation. A good free reference and textbook is online (Hyndman & Athanasopoulos, 2012).

- ***Univariate forecast model*** is one that uses time series data for the variable of interest (dependent variable only with no independent variables other than the time index itself and seasonality; see below in this list).

- ***Causal or multivariate forecast model*** is one that has true independent variables in addition to the dependent variable. Often multiple regression models are used for this category. This chapter does not include any such models but see Chapters 15-22 for a discussion of regression and discrete choice modeling. Much of the forecasting needs for operations management are met with univariate forecast models. Forecasts are often needed for one week or month ahead and it is difficult to beat the accuracy of univariate forecast models in the short run.

- ***Forecast origin*** is *T* in notation (23.1). It is the most recent data point.

- ***Steps ahead*** is how many time intervals into the future corresponds to a forecast. Most tactical needs for police are met with a one-step-ahead forecast.

- ***Forecast horizon*** is the maximum number of steps ahead, *K*, made from a forecast origin.

- ***Trace forecast*** is the full set of forecasts for a particular origin for each step ahead out to the horizon. For example, if one were making a trace forecast with monthly data and a 12 month horizon, forecasts would be made for each step ahead, 1, 2, and so forth up to 12. Generally, forecast errors increase as step ahead increases.

- ***Level*** is the current estimated mean of a time series, denoted in this chapter by a_{it} for area i at time t. Notice that level varies with time because this chapter uses exponential smoothing methods in which model parameters, such as a_{it}, self-adapt to changing time series patterns. Simple exponential smoothing, to be discussed in depth in the next section, has only a_{it} as its parameter and thus just estimates the current mean of a time series:

$$\bar{y}_{it} = a_{it} \text{ for t } = 1, \ldots, \text{T}. \qquad (23.2)$$

- ***Trend*** is the estimated change per time interval in the mean of a time series, moving ahead from the level. Here trend, denoted as b_{it} for area i, is also a varying parameter. This term is added to the simple exponential smoothing model to yield the Holt exponential smoothing forecast model.

$$F_{iT+k} = a_{iT} + b_{iT}k \text{ for k } = 1, \ldots, \text{K.} \quad (23.3)$$

The fitted model at time t is still $\bar{y}_{it} = a_{it}$ because a_{it} is the current level of the time series at time t. The slope, b_{it}, only comes into play when forecasting by adding the expected change.

- ***Seasonality*** is the adjustment made for each time observation for seasonal patterns such as when, for example, crime is low in February and high in July relative to the time series trend line. For weekly data, there are 52 seasonal adjustments, S_j with j= 1, …, 52. Likewise for monthly seasonal adjustments there are 12 seasonal adjustments. Seasonal adjustments can be additive or multiplicative. Additive seasonal adjustments are affected by the scale (volume) of data at an observation unit. So a low crime rate patrol sector would have a small seasonal adjustment for any time period but a high crime rate patrol sector would have a large adjustment. In contrast, multiplicative seasonality is unitless, having values such as 1.20 for a 20 percent increase for a summer month and 0.80 for a 20 percent decrease for a winter month relative to the time trend. For space and time series, it is desirable and necessary to use multiplicative seasonality so that seasonality estimated using all crime data of a jurisdiction can be used for any sub-area of the jurisdiction. Therefore, this chapter uses multiplicative seasonality. If time period t is in season j, seasonality is denoted as $S_{ij(t)}$ and the Holt model and forecast with seasonal adjustments are

$$\bar{y}_{it} = S_{ij(t)} a_{iT} \text{ for t } = 1, \ldots, \text{T} \quad (23.4)$$

$$F_{iT+k} = S_{ij(T+k)}(a_{iT} + b_{iT}k) \text{ for t } = 1, \ldots, \text{T} \quad (23.5)$$

The method by which this model is estimated, including seasonal factors, is covered in later parts of this section.

- ***Fit error*** is $e_{it} = y_{it} - \bar{y}_{it}$ where y_{it} is data from t = 1, …, T. Typically parameters such as a_{it} and b_{it} are estimated by finding values that minimize the sum of squared fit errors, $\Sigma\, e_{it}^2$ over all historical data.

- ***Hold-out sample*** is data used to estimate the forecast accuracy of a forecast model. The steps are to use data from t = 1, …, T to estimate model parameters (such as a_{it} and b_{it}) and to make forecasts F_{T+k} for k = 1, …, K. The hold out sample in this case is y_{iT+k} for k = 1, …, K and it cannot be used in parameter estimation. Researchers set aside (or hold out) the end of a time series and forecast that data as if it were not available. Then with forecasts made, they compare the forecast and hold out sample data to calculate forecast errors.

- ***Forecast error*** is $e_{iT+k} = y_{iT+k} - F_{iT+k}$ where y_{iT+K} is data from a hold out sample or, for contemporaneous forecasts (made in real time), is simply actual values realized after the forecast is made. Given a sample of forecast errors, researchers create summaries using measures such as the mean squared forecast error or mean absolute forecast error.

Extrapolative methods

There are two main approaches to extrapolative models. The first, time trends as used in exponential smoothing, estimates a time trend model and seasonal adjustments. To forecast, one merely continues (extrapolates) the most recently-estimated trend line into the future and makes corresponding seasonal adjustments.

The second approach estimates correlations of the dependent variable with its past values as well as other correlations. For example, if there was a recent run of large crime counts in a time series, an autocorrelation model tends to keep the run going. The Box-Jenkins model uses this approach (Box et al., 2008). Box-Jenkins has some limitations for practice. The first is that the method tends to be complex with many parameters being estimated and with several steps to the procedure. Also, the preponderance of comparative studies in the literature, including the M-Competitions, have provided evidence that the simpler time trend models are just as accurate if not more so than Box-Jenkins models. Box-Jenkins tends to overfit noisy data (such as crime data at the patrol sector level), thereby leading to less accurate forecasts. Therefore, this chapter uses trend models.

With exponential smoothing as the approach, the question arises as to how to estimate seasonality. An extension of simple and Holt smoothing is the Holt-Winters forecast model that simultaneously estimates level, trend, and seasonal factors. An alternative, and the one taken in this chapter, is to:

A. Estimate seasonal factors from the raw time series data, as a preliminary step, using Classical Decomposition (Hyndman & Athanasopoulos, section 6-3, 2012);

B. Deseasonalize the raw data (in the case of multiplicative seasonal factors) by dividing each time series data point by its appropriate seasonal factor;

C. Estimate the simple or Holt exponential smoothing model using the deseasonalized data;

D. Make extrapolative forecasts; and

E. Reseasonalize the forecasts.

This approach provides alternatives for estimating seasonal factors. The obvious one to try is to estimate seasonal factors for each area or district. A problem with this approach, however, is that a season only occurs one a year. For example, if there are five years of weekly data and one wants to estimate the seasonal factor for week 21 (or any other week), then there are only five observations. This makes seasonality the least precisely estimated parameters in extrapolative models. Often, more accurate seasonality estimates can be obtained and therefore more accurate forecasts can be had by pooling data across areas. For example, one could estimate separate seasonal factors for residential versus commercial areas and then use the resulting estimates for each residential and commercial district's time series.

A better alternative is to estimate seasonality using all of the data of a jurisdiction. There is evidence that this is the best alternative for crime forecasting (Gorr, Olligschlaeger, & Thompson, 2003). Jurisdiction-level seasonal adjustments are smaller than those for the district-level, closer to overall mean levels, which, in turn, provides greater forecast accuracy. District-level seasonal factors are overly-influenced by individual data points of the small sample sizes of districts.

Simple Exponential Smoothing

This model estimates the time-varying mean of a univariate time series. Perhaps confusing is that there are two types of parameters in the model. First is the mean of the series, estimated by a_{it}. The second is the smoothing parameter, α, that determines how quickly or slowly the method adapts to changes in the mean of the series over time. Each historical data point in the time series is weighted in a sum to estimate a_{it}, with the weights declining exponentially with the age of the data from the forecast origin. The weights should approximately sum to 1.0. For small α, the weights drop off slowly with data age, retaining more of the history in the estimate and making the mean estimate change slowly. For larger α, the weights drop off quickly, retaining relatively little of the historical data and making the mean estimate change quickly. The closed form version of the estimated mean is as follows:

$$a_{iT} = \alpha y_{iT} + \alpha(1- \alpha)y_{iT-1} + \alpha(1 - \alpha)^2 y_{iT-2} + \alpha(1 - \alpha)^3 y_{iT-3} + \cdots \quad (23.5)$$

where $0< \alpha<1$.

For example, for $\alpha=0.05$, this expression is

$$a_{iT} = 0.05y_{iT} + 0.0475y_{iT-1} + 0.045125y_{T-2} + 0.042869y_{iT-3} + \cdots \quad (23.5a)$$

but for $\alpha=0.25$ it is

$$a_{iT} = 0.25y_{iT} + 0.1875y_{iT-1} + 0.140625y_{T-2} + 0.105469y_{iT-3} + \cdots \quad (23.5b)$$

If T=60 (for example, five years of monthly data) the sum of weights for $\alpha=0.05$ is 0.954 (and as T gets larger, the sum approaches 1) and for $\alpha=0.25$, it is already 1.000. So a_{iT} is a weighted average of all the historical data.

Equation 23.5 is rarely used to calculate the current mean, a_{iT}. Instead an equivalent, recursive form is used:

$$a_{it} = \alpha y_{it} + (1 - \alpha)a_{it-1} \text{ for } t = 1, \ldots., T. \quad (23.6)$$

The only issue with this form is that it needs initial values for a_{i0}. Generally y_{i1} or the average of the first few observations is used for this purpose. After a brief "burn in" period, equation (23.6) forgets the initial value and tracks the mean of the series, so the choice of initial values is not critical. They just have to be reasonable.

Without a seasonal model component, the forecast for simple exponential smoothing is:

$$F_{iT+k} = a_{iT} \text{ for } k = 1, \ldots K. \quad (23.7)$$

In other words, the forecast is a straight line, a constant, for any future time period.

For noisy data, such as crime counts in patrol districts, and short-term forecasts of one or a few steps ahead, it is hard to beat forecasts from (23.7). If, however, the forecasts go out to six months or a year ahead, then a time trend term can improve forecast accuracy, if there is a time trend in the data. Therefore a section below introduces the Holt exponential smoothing model which includes a time trend.

Selecting a value for the smoothing constant, α

Simple smoothing has two things going for it. First is that a_{iT} is a measure of central tendency, a mean, and it is almost always more accurate over the long-haul to forecast using means. Second is that a_{iT} is self-adaptive to changes in the mean, albeit with a time lag. The larger α is, the smaller the lag to estimating a time varying mean, but also larger is the chance that a_{iT} will overly respond to noise in the time series, depart from the underlying mean, and thus forecast poorly. So there is tradeoff, a balancing act in getting α just right.

The traditional criterion for computing parameters is to minimize the sum of squared fit errors, and that is what is done in *CrimeStat*. Unfortunately the corresponding functional form is highly nonlinear, so there is not a closed form solution or equation for computing the optimal α. Instead, it is easy enough to try a grid of α values (for example, 0.01, 0.02, …, 0.99 in *CrimeStat*), compute the sum of squared fit errors for each grid value, and choose α that has the minimum.

Figures 23.1 through 23.3 show the effects of different smoothing parameter values for estimating the time series mean from the sample data provided in *CrimeStat* for this chapter. The data are from census tract 20100, Pittsburgh, Pennsylvania (a high crime area) and are monthly time series for serious violent crimes (homicide, rape, robbery, and aggravated assault) between 1990 and 1999. Figure 23.1 shows smoothed values with near-optimal α = 0.15. The smoothed values track the mean of the series well. Jurisdiction-level seasonality, estimated from all of Pittsburgh's serious violent crimes (see the Classical Decomposition section), is included in the smoothed values, accounting for much of the month-to month variation.

Figures 23.2 and 23.3 are extreme cases for α values to make a point. Figure 23.2 has smoothing for α = 0.01, resulting in very slow adaptation and a very long memory for old data values. Values that are too low for *α* cannot keep up with the changes in this time series so that forecasts based on them would do poorly. In contrast, Figure 23.3 has smoothing for α = 0.99 so that there is practically no memory of historical values. The smoothed values are very close to being the raw data with no smoothing. These smoothed values are too noisy and would forecast poorly. So an optimal α value between these two extremes is needed for forecasting, 0.15 as shown in Figure 23.1.

Straw man forecasts

Clearly simple exponential smoothing is simple, but there are even simpler methods. However, to justify its use, simple exponential smoothing has to forecast more accurately than these simpler methods. Comparative research on forecast methods, as published in journal articles such as in the *International Journal of Forecasting* or the *Journal of Forecasting*, thus

Figure 23.1:

Simple Smoothing with α=0.15

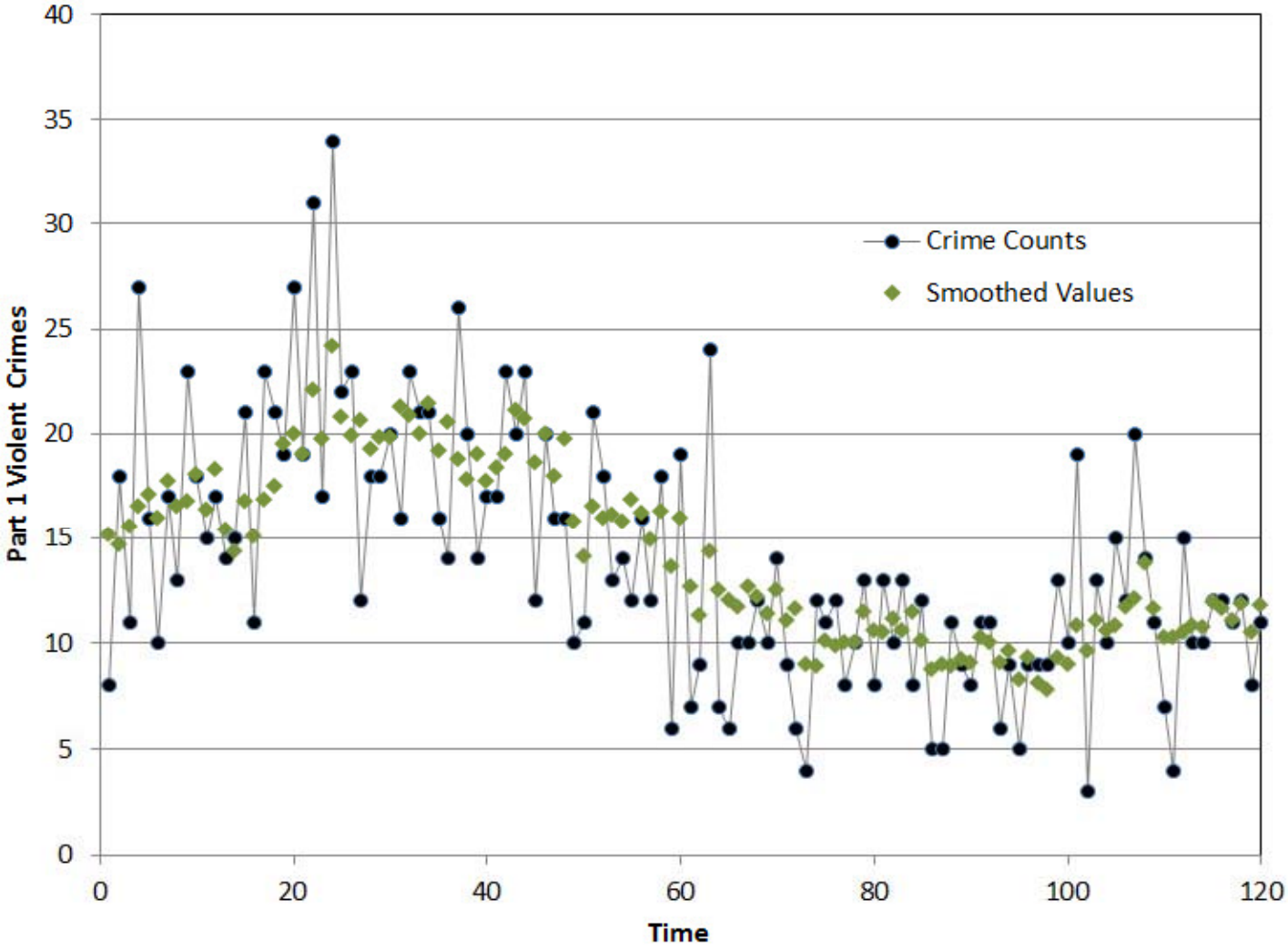


Figure 23.2:

Simple Smoothing with α=0.01

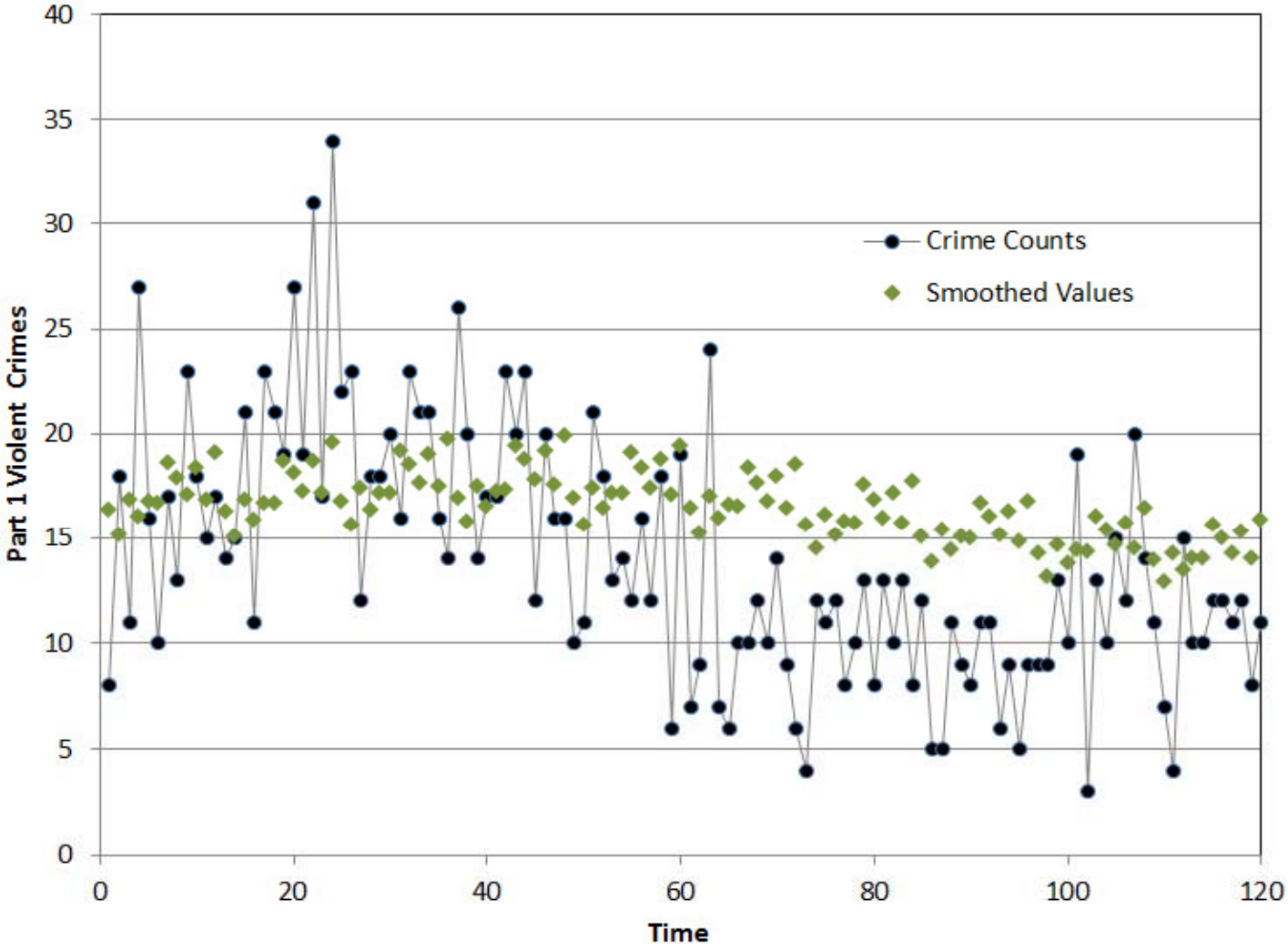
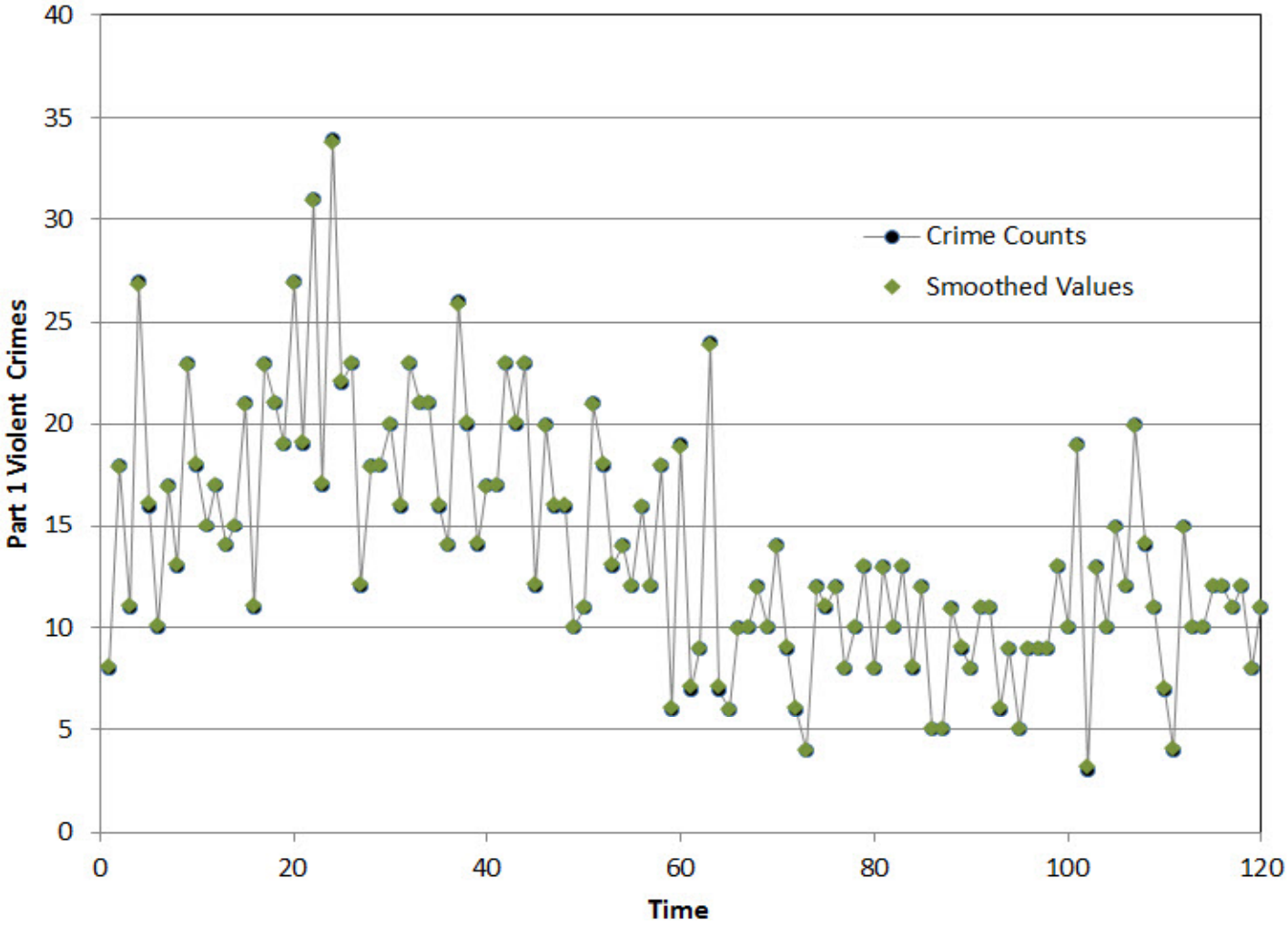


Figure 23.3:

Simple Smoothing with α=0.9



always include *straw man* methods. Hold-out sample forecast accuracy is compared on the same data between multiple forecast models/methods, including straw man methods.

The simplest method, for the case of no seasonality, is the naïve method forecast, using the last historical data point (the forecast origin) as the forecast:

$$F_{iT+k} = y_{iT} \text{ for all k.} \tag{23.8}$$

This method, also called the random walk forecast, is sometimes hard to beat, for example, in attempting to forecast stock market prices. At other times, when there is a time trend, smooth mean over time, or seasonality, the naïve method is easy to beat. Gorr, Olligschlaeger, and Thompson (2003) provided evidence that naïve forecasts and other data value-based forecasts (such as using the same month's data from the previous year) forecast very poorly for crime data.

A second straw man is the sample mean of the time series as the forecast:

$$F_{iT+k} = \frac{1}{T}\sum_{t=1}^{T} y_{it}. \tag{23.9}$$

Closest to simple exponential smoothing is the mean of a moving window of recent data, for example with the sum in (23.9) over t=T-w+1, …, T where *w* is the window length. This mean is also self-adaptive to changes in the mean of the time series, but has a larger time lag than exponential smoothing. The choice of a value for *w* is similar to that for α: too large a value makes the mean unresponsive and too small a value makes the mean unreliable.

Holt Exponential Smoothing

This model retains a_{it} and introduces a second model parameter, b_{it}, that is the coefficient of the time index of the time series as in (23.3). The Holt recursive equations for estimation are as follows:

$$a_{it} = \alpha y_{it} + (1 - \alpha)\left(a_{it-1} + b_{it-1)}\right) \tag{23.10}$$

$$b_{it} = \beta(a_{it} - a_{it-1}) + (1 - \beta)b_{it-1} \tag{23.11}$$

where a_{it} is the current level of the time series at time t, b_{it} is the time trend slope used in making forecasts, α is the smoothing parameter for the level with $0 < \alpha < 1$, β is the smoothing parameter for the trend with $0 < \beta < 1$, and the estimated model at the forecast origin, T, is

$$\bar{y}_{iT} = a_{iT}. \tag{23.12}$$

The forecast equation is:

$$F_{iT+k} = a_{iT} + b_{iT}k \text{ for all k.} \quad (23.13)$$

It is worth reiterating that equation (23.10) estimates the current mean or level of the time series while equation (23.11) estimates the trend, or change in the series for each step ahead forecast. This is in contrast to a linear regression model with the time index as the independent variable, $\bar{y}_{it} = b_0 + b_1 t$ where b_0 is the intercept term, the value of $\bar{y}_{it}$ at t is b_0. Parameter a_{it} in (28.10) is not an intercept term, but is the mean of the time series at time t.

The Holt model needs initial values, a_{i0} and b_{i0}. For the former, one can use the first observation or average of the first few observations as in simple exponential smoothing. For the latter, one can use the difference y_{i2}-y_{i1} or the average of the first few such differences. Again, as long as reasonable values are used, Holt will soon forget the initial values and be on track with the mean parameter values.

Holt parameter estimation also uses a grid search, but over the two-dimensional α and β space. For example, in *CrimeStat*, if there are 100 values to try for each smoothing parameter, then all possible pairs need to be tried with 10,000 pairs in total for the optimization. This is easily and quickly done by *CrimeStat* every time it forecasts using the Holt model.

Classical Decomposition: Seasonality

This section covers one of the earliest and most robust methods for estimating seasonality in a time series, Classical Decomposition. It is a separate procedure for estimating seasonality from raw time series data, so it can be applied just for the sake of understanding seasonal patterns that are part of a time series whether there is a time trend or not in the series. There are no smoothing or other parameters as part of the method, just straightforward calculations.

As stated earlier, CrimeStat uses the multiplicative form of seasonality, a dimensionless factor for each season that is valuable for cross-sectional data, such as in the case of crime space and time series data with its several or many zones, patrol districts, or census tracts of interest. The basic idea is to create an observation of seasonality for each data point in the series. With monthly data and the month of July for example, all of the July observations of seasonality are collected over the series and averaged to yield the July seasonality estimate.

The approach to creating a seasonality estimate for, say, July 2012 is to center an average of crimes on July that is one full year long with July in the middle. The average is an estimate of the mean of the series with seasonality removed, because the entire year is included in the average. Then the observation for multiplicative seasonality is the crime count actual value for

July 2012 divided by its centered average. The only problem with this procedure is that the number of seasons per year is usually an even number (4 for quarters, 12 for months, and 52 for weeks) so there is no natural center of a data window. Therefore, a simple adjustment is made, including an extra data point on each end of the window.

The Detection Problem

Exponential smoothing provides a relevant model and estimation method for time series that are predictable. As long as the data being smoothed do not change abruptly, exponential smoothing provides good forecasts. It is difficult to forecast abrupt changes, so often the best that can be done is to detect them as soon as possible after they have occurred or started to occur. That is the purpose of this section, to provide a method that works in partnership with exponential smoothing and extrapolative forecasts for detection of abrupt or large changes.

This section uses a world view that has two states: (a) *business-as-usual* which has time trends that can be accurately extrapolated and (b) *exceptions* which are departures from business-as-usual including outliers, step jumps, and turning points when a trend reverses direction. In crime space and time series data, a large crime increase is caused by some change in the underlying criminal element, for example, formation of a gang rivalry, sales by a new illegal gun dealer, parole of serial criminals who continue crime careers, and so forth. In some cases it can be due to the withdrawal of police resources, such as when a special enforcement program ends. Large decreases are also of interest and may be due to special police enforcement programs.

Time series detection methods merely draw attention to areas where there is evidence that an exception is occurring. It is up to the crime analyst to diagnose a detected area, to determine the nature of the problem if one is thought to exist and to recommend interventions.

Counterfactuals

To detect a change in a crime space and time series, we need a basis for comparison, that which would have happened had it been business-as-usual instead of an exception. This is called the "counterfactual forecast" and is provided by extrapolative forecasts. Suppose that one has data up to y_{iT}. Then one makes an extrapolative forecast, F_{iT} using data from t = 1, …, T-1 and computes the counterfactual forecast error, $e_{iT} = y_{iT} - F_{iT}$. Similar to hypothesis testing, if e_{iT} (and other recent counterfactual forecast errors) is large enough, then there is evidence that there is an exception. If the change is more moderate, the tracking signal to be described next accumulates consistent counterfactual forecast errors (e.g. all positive) over several successive time periods to also provide evidence of an exception.

Tracking Signals

Detection methods calculate tracking signals, or test statistics. When the signal gets large there is evidence that there is exceptional behavior in a time series. This requires one to choose a threshold value for a "signal trip" a topic covered later in this section.

A simple tracking signal (and one of the oldest to be used) is the cumulative sum of errors:

$$CUSUM_{wiT} = \sum_{t=T-w+1}^{T} e_{it}. \tag{23.14}$$

where w is the window length of the summation. If there is a large error in one direction (say positive) or there is a series of medium-sized errors in one direction, then CUSUM may be large enough to signal an exception.

Standardized data is created by subtracting the mean from a sample of data and dividing by a measure of spread, such as the sample standard deviation. The advantage of working with standardized data is that if there are many samples to be examined, such as all patrol districts in a police jurisdiction, then one can use a single threshold value for a "signal trip" indicating exceptional behavior. However, with raw data for each area (or zone), one would have to choose a separate threshold for each area depending on the scale of the crime problem in each area.

The counterfactual forecast error, e_{it}, has an expected value of zero, so it already behaves like the numerator of standardized data. The k-period Brown tracking signal (Brown, 1959) divides CUSUM by an alternative measure of spread to the standard deviation, the simple smoothed mean of absolute counterfactual forecast errors (called the *mean absolute deviation*):

$$w - period\ Brown = \left|\frac{CUSUM_{wiT}}{MAD_{iT}}\right| \tag{23.15}$$

where

$$MAD_{iT} = \beta|e_{iT}| + (1-\beta)MAD_{iT-1} \tag{23.16}$$

and $0<\beta<1$ is a smoothing parameter

While β is a symbol used in 23.16 and also for Holt smoothing, they are two different parameters. One problem with w-period Brown is that after an exception has occurred, the signal often has to be manually reset to a low, *business-as-usual* value. Otherwise, the signal

continues to be large indicating exceptional behavior that may already have passed. Trigg (1964) thus proposed a modification to smooth the numerator as well as the denominator so that it is self-adaptive, resetting itself. Trigg used a common smoothing parameter for both the numerator and denominator while others, including McClain (1988), found evidence of better performance with separate smoothing parameters for the numerator and denominator. Now Trigg is calculated as:

$$Trigg_{iT} = \left| \frac{E_{iT}}{MAD_{iT}} \right| \quad (23.17)$$

where

$$E_{iT} = \alpha e_{iT} + (1 - \alpha) E_{iT-1} \quad (23.18)$$

and $0 < \alpha < 1$ is a smoothing parameter.

Note that while the smoothing parameter is denoted by α here, it is different than the parameters also called α for simple and Holt smoothing. While Trigg and Brown methods are similar in performance, Trigg is more convenient to use and so *CrimeStat* uses it.

Decision Rules

Tracking signals are implemented with decision rules such as the following:

If $Trigg_{iT} \geq L$ then issue a signal trip report (23.19)

Else do nothing (23.20)

where L is a threshold level to be chosen by the decision maker. While similar to decision rules used in statistics for hypothesis testing, there are important differences.

In the academic world of theory building and model testing, L is chosen to yield traditional type I error levels of 1 or 5 percent. A type I error occurs when the signal trips but in fact there is no exceptional behavior, in other words, a *false positive* occurs. These error levels are conservative so as not to claim to have evidence that a theory is true unless the evidence is strong—scientists are skeptical. A type I error rate (or *false positive rate*) of 5 percent means that 5 percent of the *negatives* (periods without exceptions) are falsely signaled as positives, which is a waste of resources if the decision maker takes action.

In management of events such as crimes, however, the false positive rate needs to be chosen to fit the situation. Larger false positive rates are often desired; for example, they are approximately in the range of 10 to 15 percent for cancer screening in the U.S. because society

values early detection of cancer (and therefore more successful treatments on the average) much more than the consequences of false positives (pain and wasted cost of biopsies that show no cancer present; see Banez et al. 2003, Elmore et al. 2002). In other words, a *false negative*, a positive that is missed by the decision rule with no signal trip, is much worse than a false positive in the cost to society. For example, with crime, an area that is experiencing a large crime increase but goes undetected is costly. It is better to accept a larger number of false positives in this case than to fail to detect an area which shows a real increase in crime (false negative). Crime analysts, when interviewed by Cohen et al. (2009), stated that in their judgment it is 10 times more important to avoid a false negative than a false positive when monitoring crime time series for exceptions.

A *true positive* is the case where there is an exception (disease or flare up in crimes) and the decision rule (23.19) signals it. The *true positive rate* is the percentage of all positives (cases where disease or crime flare up exists, for example) that are signaled by the decision rule—values in the range of 60 to 90 percent should be attainable for crime data. However, there is a trade-off. To increase the true positive rate, one must also increase the false positive rate. One increases both rates by making the threshold level, L, smaller. The optimal level, L, depends on three things. One makes L smaller if prevalence is relatively high (the fraction of all cases that are positives), the benefits of finding a positive are high, and the costs of a false positive is relatively low. Benefits and costs for police are likely similar to those of physicians screening for cancer because the benefits of preventing crimes is high and costs are incurred anyways but with efforts redirected to areas with potentially better results.

There is a formal decision framework for choosing L called *receiver operating characteristics* (ROC), a name that comes out of the signal processing field where signals are received with equipment. Cohen et al. (2009) provide an overview of this framework applied to crime data. While the concepts are good, it is impractical for police departments to carry out a formal optimization of choosing values for smoothing parameters for equations (23.16), (23.19) and L, except perhaps for the largest organizations. It is necessary for crime analysts to label points in a sample of time series that they believe to be positives (true large change points) and then independently run monitoring through the sample in simulation mode. Then all possible threshold level values and smoothing parameter values are assessed in a grid search and optimization model to provide optimal values.

Instead, this chapter recommends default values L=1.5, α=0.9, and β=0.15. We also suggest trying L=1.75 and L=2.0 for more conservative values, providing fewer signal trips. The values for α and β are optimal ones from Cohen et al. (2009) for the Pittsburgh monthly crime data, with α=0.9 being very reactive to the signal and β=0.15 more slowly updating the spread measure. Most of the value of the Trigg signal comes from keeping the measure of spread up to date so that L functions correctly (see Cohen et al., 2009).

Figures 23.4 through 23.7 illustrate Trigg time series monitoring for two Pittsburgh serious violent crime, monthly time series: for census tract 20100 as seen earlier in Figures 23.1 through 23.3, a high crime area, and for census tract 50900 a more moderate crime area. All figures use default values of α=0.9, and β=0.15 for the Trigg signal and 0.015 optimal smoothing parameter for simple smoothing with jurisdictional-level seasonality. Figure 23.4 uses the default value L=1.5 for tract 20100. One can see that Trigg with these values does a good job of signaling exceptional values, both high and low values. Figure 23.5 shows the effect of raising L to the more conservative 2.0. Only three months are signaled as exceptional. Perhaps this value of L is too conservative. Figure 23.6 uses L=1.5 for tract 50900. Again there are many signal trips, but they appear to provide good information. Finally Figure 23.7 substitutes the conservative L=2.0 which again appears too conservative in this case.

These cases suggest that once a signal is tripped police should provide extra resources to the area for three or more additional months, for example, because of the four instances in Figure 23.6 of repeated exceptions with zero to two months between exceptions.

Conclusions

This chapter has presented models and methods for time series forecasting of crime applied to all subareas of a police jurisdiction. The subareas can be as small as patrol districts or census areas such as census tracts. It is best to forecast aggregates of crime types, for example all serious violent crimes, all serious property crimes, all disorder crimes, etc. so as to increase the sample sizes for each time period and allow for reasonably accurate estimation. Time periods used in *CrimeStat* for time series include weeks and months. Day periods are too short to provide adequate sample sizes per period for accurate estimation of model parameters.

This chapter defined relevant terms, provided two univariate time series models (one for a time-varying mean and a second for a time-varying time trend that includes a slope/growth term for forecasted values), exponential smoothing methods for estimating model parameters, and extrapolative forecasts from the models. Exponential smoothing automatically learns and adjusts models for smooth changes in time trends and is perhaps the most-widely used time series method in industry for operations management.

Seasonality for week of the year (1–52) or month of the year (1–12) is estimated separately, using Classical Decomposition, a method provided by the U.S. Census Bureau. *CrimeStat* uses a multiplicative form, seasonal factors, that is valuable for cross-sectional data such sub areas of a police jurisdiction where the scale of crime varies considerably. The factors are dimensionless and represent percentage changes to time series trends due to season, such as a 25 percent increase in crime level in July.

Figure 23.4:

Trigg Signal Trips with Simple Smoothing

Jurisdiction Seasonality & L=1.5

(Pittsburgh Tract 20100)

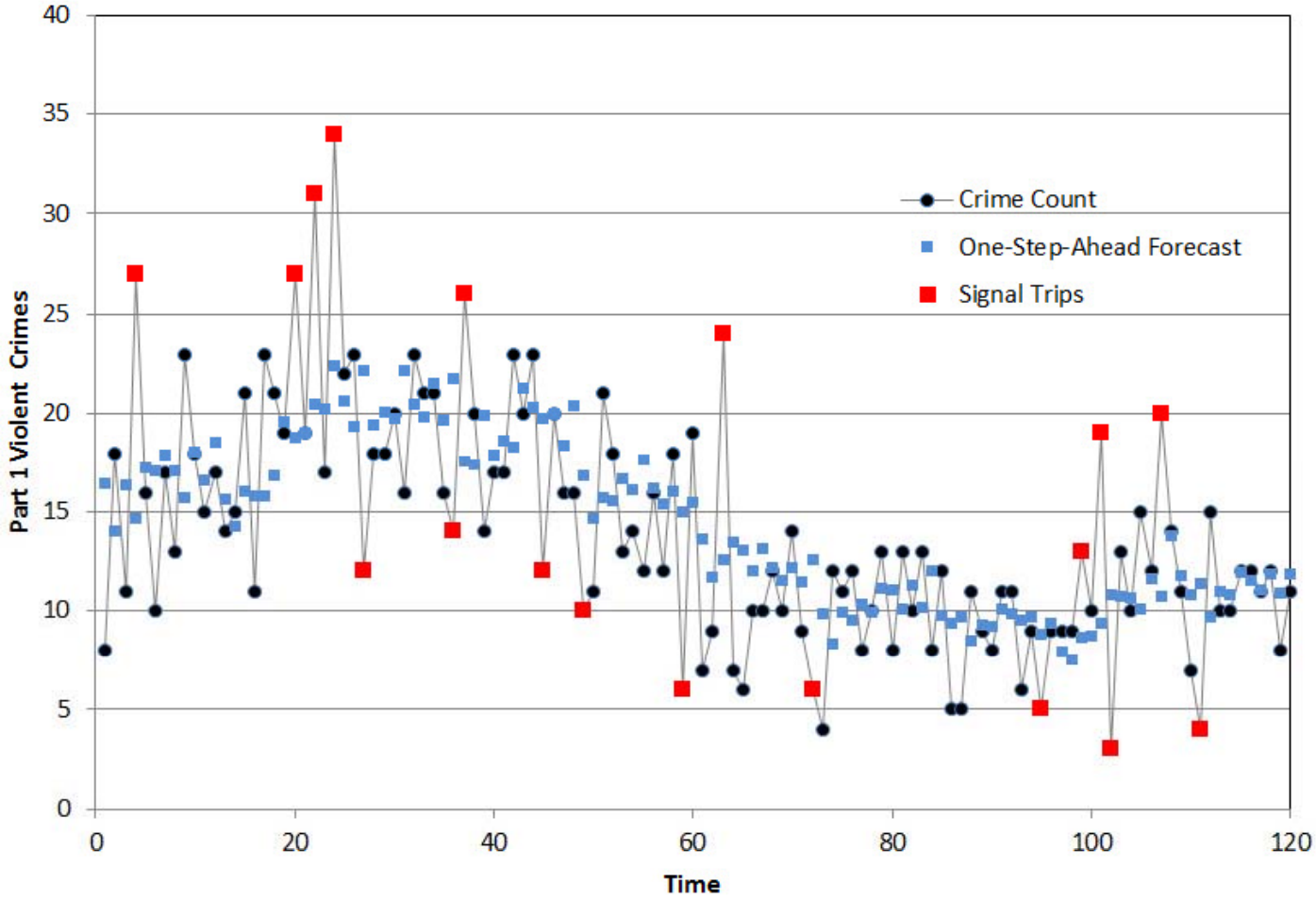


Figure 23.5:

Trigg Signal Trips with Simple Smoothing

Jurisdiction Seasonality & L=2.0

(Pittsburgh Tract 20100)

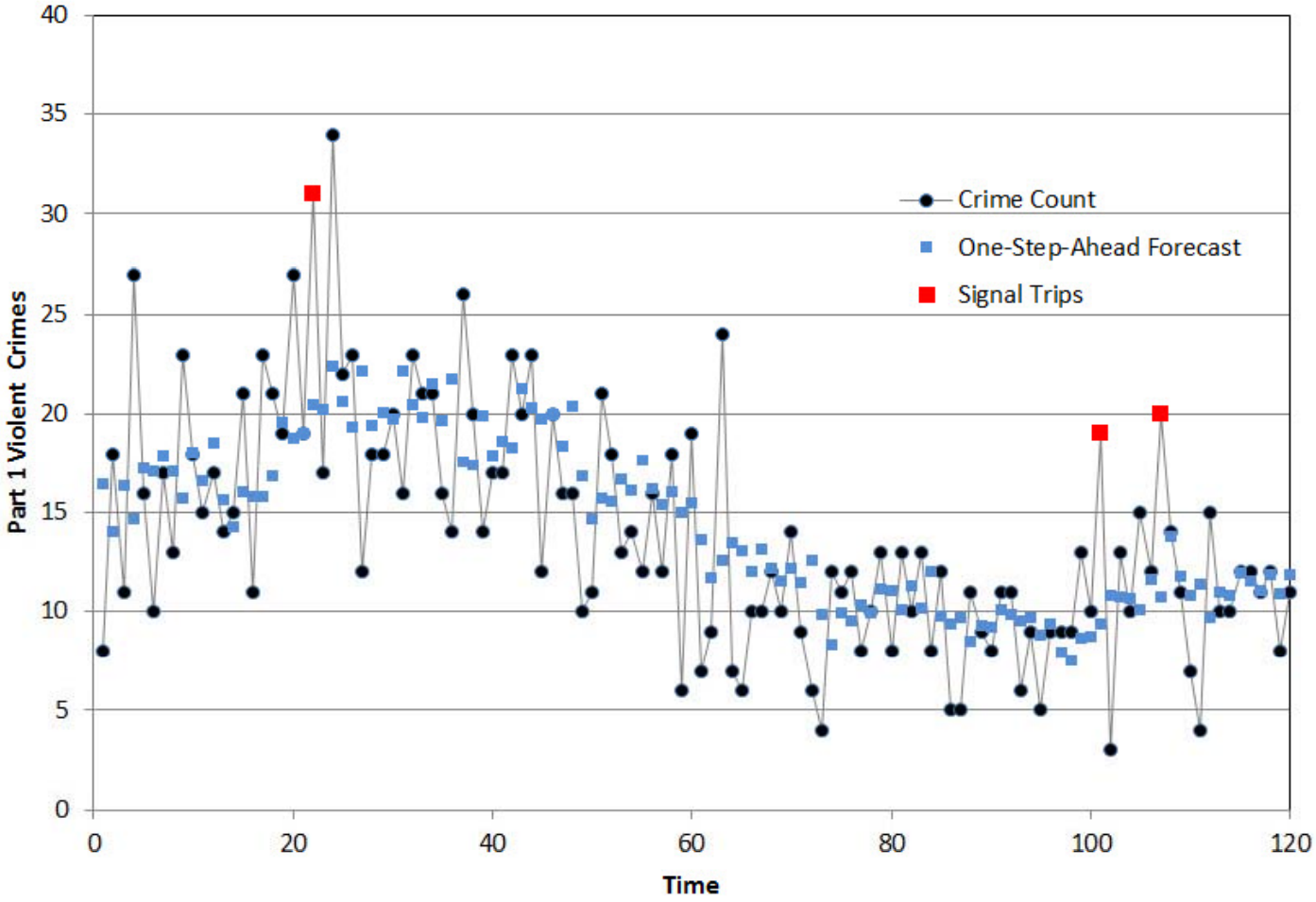


Figure 23.6:

Trigg Signal Trips with Simple Smoothing

Jurisdiction Seasonality & L=1.5

(Pittsburgh Tract 50900)

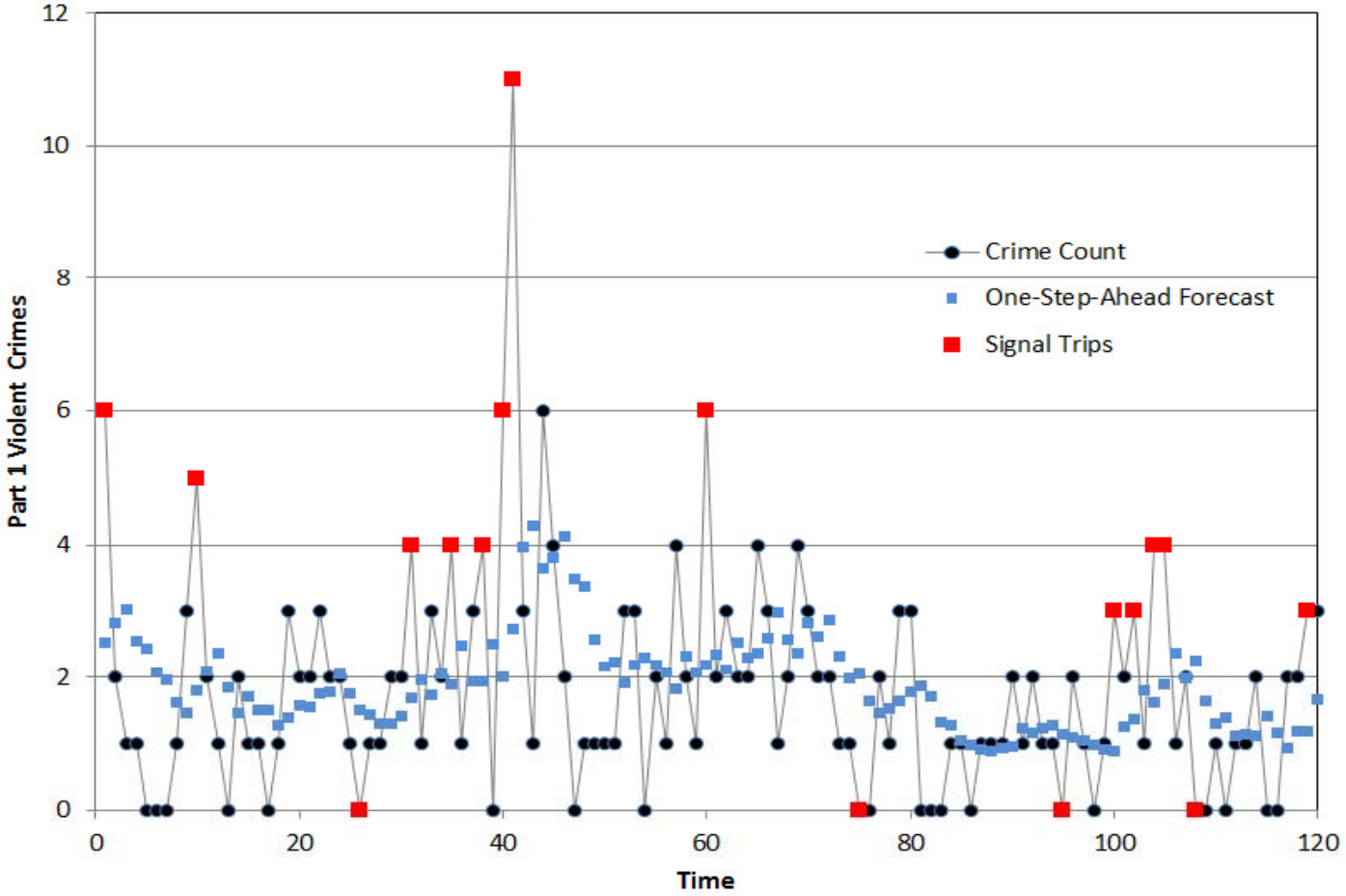
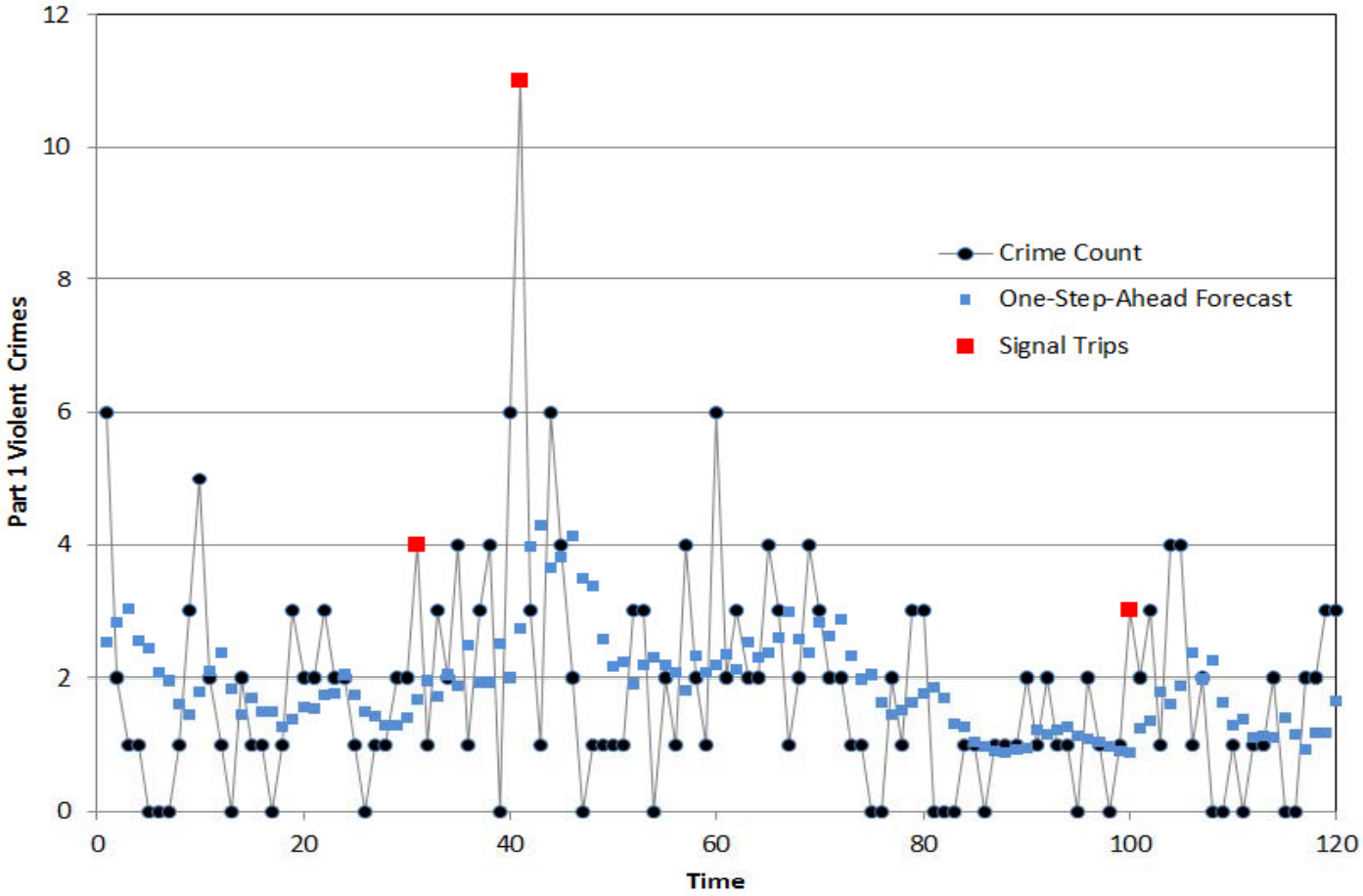


Figure 23.7:

Trigg Signal Trips with Simple Smoothing

Jurisdiction Seasonality & L=2.0

(Pittsburgh Tract 50900)



Forecasts are extrapolations of estimated time series. They are values from the trend line extended into the future with adjustments made for seasonal effects. *CrimeStat* automatically optimizes the fit of time trend parameters and batch forecasts time series for all subareas of interest, making it convenient and efficient for the crime analyst.

Perhaps more valuable than the forecasts themselves is the ability to use forecasting as a means for detecting large changes in crime time series. *CrimeStat* uses a signal tracking mechanism, the Trigg time series signal, to automatically detect large changes in crime time series such as crime flare ups. The objective is to detect large changes as soon as possible. The basis for comparison is the counterfactual forecast, an extrapolative forecast made for the last data point of a time series, which represents the crime level expected given "business-as-usual" or no change in time series pattern. In *CrimeStat*, every time series in a police jurisdiction gets an assessment for every historical data point including the very last, as to whether it appears to be ordinary or exceptional.

References

Banez, L., Prasanna, P., Sun, L., Ali, A., Zhiqiang, Z., & Adam, B. (2003). Diagnostic potential of serum proteomic patterns in prostate cancer, *The Journal of Urology*, 170, 442–446.

Box, E.P., Jenkins, G.M., & Reinsel, G.C. (2008). *Time series analysis: forecasting and control*, Wiley: Hoboken.

Brown, R. G. (1959). *Statistical forecasting for inventory control*, New York: McGraw-Hill.

Cohen, J., Garman, S. & Gorr, W. L. (2009). Empirical calibration of time series monitoring methods using receiver operating characteristic curves, *International Journal of Forecasting*, 2009, 25(3), 484–497.

Elmore, J. G., Miglioretti, D. M., Reisch, L. M., Barton, M. B., Kreuter, W., & Christiansen, C. L., (2002). Screening mammograms by community radiologists: Variability in false positive rates. Journal of the National Cancer Institute, 94, 1373–1380.

Gorr, W. L. & Kurland, K. S. (2012). *GIS Tutorial for Crime Analysis*, Esri Press, Redlands.

Gorr, W. L., Olligschlaeger, O. M. & Thompson, Y. (2003). Short-term forecasting of crime, *International Journal of Forecasting, Special Section on Crime Forecasting*, 19(4), 579–594.

Hibon M. & Makridakis S. (2000). The M3 Competition: results, conclusions and implications, *International Journal of Forecasting*, 16, 451-476.

Hyndman, R. J. & Athanasopoulos, G. (2012). *Forecasting: principles and practice: An online textbook,* http://otexts.com/fpp/.

Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, E. & Winkler, R. (1982), The accuracy of extrapolation (time series) methods: Results of a forecasting competition, *Journal of Forecasting*, 1, 111-153.

McClain, J. O. (1988). Dominant time series monitoring methods, *International Journal of Forecasting*, 4, 563–572.

Trigg, D. W. (1964). Monitoring a forecasting system, *Operational Research Quarterly*, 15, 271–274.

References (continued)

Weisburd, D. L., Bushway, S., Lum, C., & Yang, S. (2004). Trajectories of crime at places: a longitudinal study of street segments in the city of Seattle. Criminology 42:283−321.

Weisburd, D.L. & Green, L. (1995). Policing drug hot spots: The Jersey City drug market analysis experiment, *Justice Quarterly* 12-711−736.

Chapter 24:

The *CrimeStat* Time Series Forecasting Module

Wilpen L. Gorr
H. John Heinz III College
Carnegie Mellon University
Pittsburgh, Pennsylvania

Andreas M. Olligschlaeger
TruNorth Data Systems
Baden, Pennsylvania

Table of Contents

Chapter 24:
The *CrimeStat* Time Series Forecasting Module

Introduction

The *CrimeStat* Time Series Forecasting module is designed for the forecasting of crime counts (or counts of any type of event) and the early detection of unusual levels of activity in current data. A single run detects and forecasts all districts making up a jurisdiction. The module has a single interface page. It requires the user to specify an input file—either the Primary file or another file, identify variables in the file used for forecasting, select a seasonality adjustment, specify an exponential smoothing model, turn on Trigg tracking signal, use default values or choose Trigg parameter values, and save the output. Included in this chapter is an overview of the module. The theory behind the methods is discussed in Chapter 23.

Rationale of the Module

While time series forecasting is useful to police for estimating future crime levels by extrapolating the current time series trends and seasonal adjustments, the impetus for the Time Series Forecasting Module was to provide a detection mechanism for early warning of large changes in crime patterns, either large increases or decreases. Through experience, police generally know seasonal patterns, such as increases in summer months and decreases in winter months, and know if crime is gradually on the increase or decrease. The time series methods in CrimeStat make objective estimates and forecasts for such trends which can be an aid for decision making.

Likely more valuable is early detection of a crime flare up or evidence of an abrupt crime decrease during a police intervention. Forecasting *large crime changes* requires advanced models not included in this module (e.g., see Gorr, 2009) but early detection (also called "early warning") of large crime changes which have already started is quite feasible (Cohen, Garman, & Gorr, 2009). This module uses the Trigg tracking signal, the best of the simple time series monitoring methods, and requires counterfactual forecasts as inputs to make jurisdiction-wide scans of all subareas (districts) for detection. More sophisticated tracking signals exist, in particular the spatial scan statistic (e.g., Neill, 2009), but the Trigg signal is widely used in the private sector for monitoring product demand and has simplicity as a virtue. Used as inputs to the Trigg tracking signal are one-week-ahead or one-month-ahead forecasts for all districts in a jurisdiction as the basis for judging each district's status as being a departure from the existing time trends or not. These forecasts are extrapolations of past patterns and thus represent what would have happened given no pattern change (i.e., are counterfactual). The most recent crime count of each district is compared to a forecast made from data up to but not including the most

recent time period. For districts flagged as having a new pattern or large change, the crime analyst can then drill down to details to diagnose problems and determine a course of recommended action.

The advantage of using CrimeStat for detection is that it is automated and objective. It saves the analyst from visually reviewing time series plots for all district time series and making judgments as to what series have unusual changes in crime levels.

Overview of the Module

This module implements the univariate time series models and methods described in Chapter 23. It takes as input space and time series data for weekly or monthly crime counts of each district of a single police jurisdiction. The crime counts are for a single crime type or crime aggregate (such as vehicle thefts, all serious violent crimes, or all property crimes). The districts can be any partition of areas dividing up a jurisdiction, for example police zones, patrol districts, census tracts, or grid cells. The data are stacked by district with a district name or identification number. See data descriptions later in this chapter and the sample data sets provided: "*Pittsburgh monthly crimes by tract 1990-99.dbf*" and "*Pittsburgh weekly crimes by tract 1990-99.dbf*".

CrimeStat provides two exponential smoothing estimation methods for time series models: 1) Simple Exponential Smoothing which estimates a smoothly-varying time series mean for each district and 2) Holt Exponential Smoothing which estimates a smoothly changing time trend line used in forecasting expected change (growth or decline) for each time series. The module computes the smoothing parameters of exponential smoothing to optimize one-step-ahead forecast accuracy. It does so by individual district (district optimization) to achieve the most tailored and widely-ranging forecasts, but also does so with pooled data by jurisdiction to provide robust, stable and often most accurate forecasts and counterfactual forecasts for detection.

The methods use multiplicative Classical Decomposition for seasonal adjustments in two forms, the first using data from the entire jurisdiction and the second using data for each district. Multiplicative seasonal adjustments (factors) are dimensionless quantities, for example, 1.25 which increases a forecasted crime count by 25 percent relative to the underlying mean estimated by exponential smoothing. Jurisdictional seasonal adjustments, while estimated from all district time series summed to the jurisdiction level, are used for each district's seasonal adjustments. District-specific seasonal adjustments apply to each district separately. All model estimates in the module use seasonal adjustments. There are no model degrees of freedom consumed by estimating seasonal adjustments with Classical Decomposition (the method uses averages and ratios in a simple computational procedure to make estimates). If a time series is non-seasonal, then the adjustments will all be near one in value and have no effect on performance.

The most valuable forecasts for tactical deployment of police resources are ***one step ahead***, either one week ahead or one month ahead. The routine chooses smoothing parameters that minimize the sum of squared one step-ahead forecast errors and provides one-step ahead forecasts for each district in its output. If the user wishes forecasts for additional steps ahead, the necessary parameters are available in the output and can be carried out in Excel.

Finally, the module uses the Trigg tracking signal to provide an assessment of each time period in each district as to whether the number of events is expected ("business-as-usual") or is exceptional (e.g., large change). The basis of the tracking signal for a given time period is a ***counterfactual*** forecast for each district made using exponential smoothing on time series data up to but not including the time period in question. The counterfactual forecast is the expected count given business-as-usual conditions in a district. Then if the actual count of the time period is very different than its counterfactual (expected) value, there is a ***signal trip*** that provides evidence of a change in the structure of the time series.

Data Preparation for Time Series Forecasting

The type of data that is needed is recorded events by individual areas over time. Each record must have the number of events that occurred during a single time period for a single area. The events can be crime events (e.g., burglaries, robberies, total part 1 crimes) or they can be other types of events (e.g., motor vehicle crashes, flu incidents). For example, if there are 100 districts that are being monitored and the data are measured by month over a three year period, then there will be 3,600 records, one for each month for each district.

There must be a minimum of three years worth of data for the module to work. The reason is that seasonality (and other parameters) must be estimated with sufficient precision. For example, because a season is defined by the time period (month or week), with three years worth of data each season (month or week) only occurs three times. Clearly, the variability of an estimate based on only a sample size of 3 is very high. That is why having more years worth of data provides more reliable estimates. Three years is the minimum: if there is less than three years, the module will stop and output an error message.

Data for input to the Time Series Forecasting Module need to be prepared using a GIS package or a database package. For example, a discussion and methods are found in chapter 9 of *Preparing Incident Data for Mapping* by Gorr and Kurland (2012). In the module, time series data can be input in two different ways. First, time series data can be input as spatial data as the Primary file. The requirement for either of these is that the X and Y coordinates (centroids) of each study district be listed on each of the records. For example, if there are 100 districts and 52 weeks per area with three years worth of data, then there will be 15,600 records each with an X/Y coordinate listed (100 x 52 x 3).

Second, many time series data will not have spatial coordinates assigned. Consequently, the data can be input as a non-spatial file. In the input file dialogue on the Time Series Forecasting page, the user will choose 'Other' for the input file.

Required Fields

Whether the data is spatial or not and whether there are other data elements listed, each record must incorporate the following four data elements (with names that can be different than those listed):[1]

- **Areal unit** —the name or identifier for the district of the incident
- ***Year***—the year (e.g., 2012)
- ***Season number***—either the month number (1–12) or the week number (1–52)
- ***Event Count***—the count of crimes or other types of events

All districts need data starting at the same period (Year and Week or Year and Month) through to the end period. Note that there can be **no** missing records or data values in records. If a district has zero crimes in a given period, the period for the district must be included with the value zero for the event count.

Note that data aggregation algorithms generally leave out records with zero frequency. One way to add records that have zero frequency is to create a table with all possible districts and time periods[2] and then to left join that table with the aggregated data, forcing all rows of the new table to be in the join. The district and month rows with zero frequency will have the null value and the analyst must replace null values with zeros.

The finished data must be sorted by District, Year, and Season number (Week or Month) in that order.

1 Individual incidents can be aggregated into time periods by district in one of several different types of programs – GIS, database packages such as Microsoft Access, or spreadsheet programs like Excel. The result is that each record includes a count of the number of events that occurred in a particular district for a particular time period. See chapter 9 in Gorr and Kurland (2012) for examples on how to do this.

2 An easy way to create the table with all possible districts and time periods in Microsoft Access is to (1) make a table with all district names or identifiers, (2) make a table with all years; (3) make a table with all time periods (e.g., 1, 2, …, 12 for months; 1,2, ….52 for weeks), and (4) use a make table query that includes data elements from all three tables but has no joins. This is called a "Cartesian product" and has all possible combinations of districts, years, and time periods.

See the sample data sets, "*Pittsburgh monthly crimes by tract 1990-99.dbf* "and "*Pittsburgh weekly crimes by tract 1990-99.dbf*", that are available on the download site. These data meet these requirements. The examples below use the Pittsburgh monthly data.

> The week function in Excel creates week 53 of one or two day's length. The analyst must delete points for week 53 because the module only accepts weeks 1–52.

Fields to be Defined

Figure 24.1 shows the interface for the Time Series Forecasting module for a run with monthly data, jurisdiction-wide seasonality, Simple Exponential Smoothing, and with Trigg tracking. There are 10 fields that must be selected for the module to work. These include:

Input file.

This is the file with the data for the Time Series Forecasting module. It can be the Primary file or another file. If it is the Primary file, then it must be a spatial data file with and X and Y coordinates defined on each record. If it is another file, select Other and then identify the file.

Areal unit

This is the variable name of each district being forecasted. For example, in Figure 24.1 the Pittsburgh monthly data set is shown and the areal unit is TRACT, the census tract ID.

Year

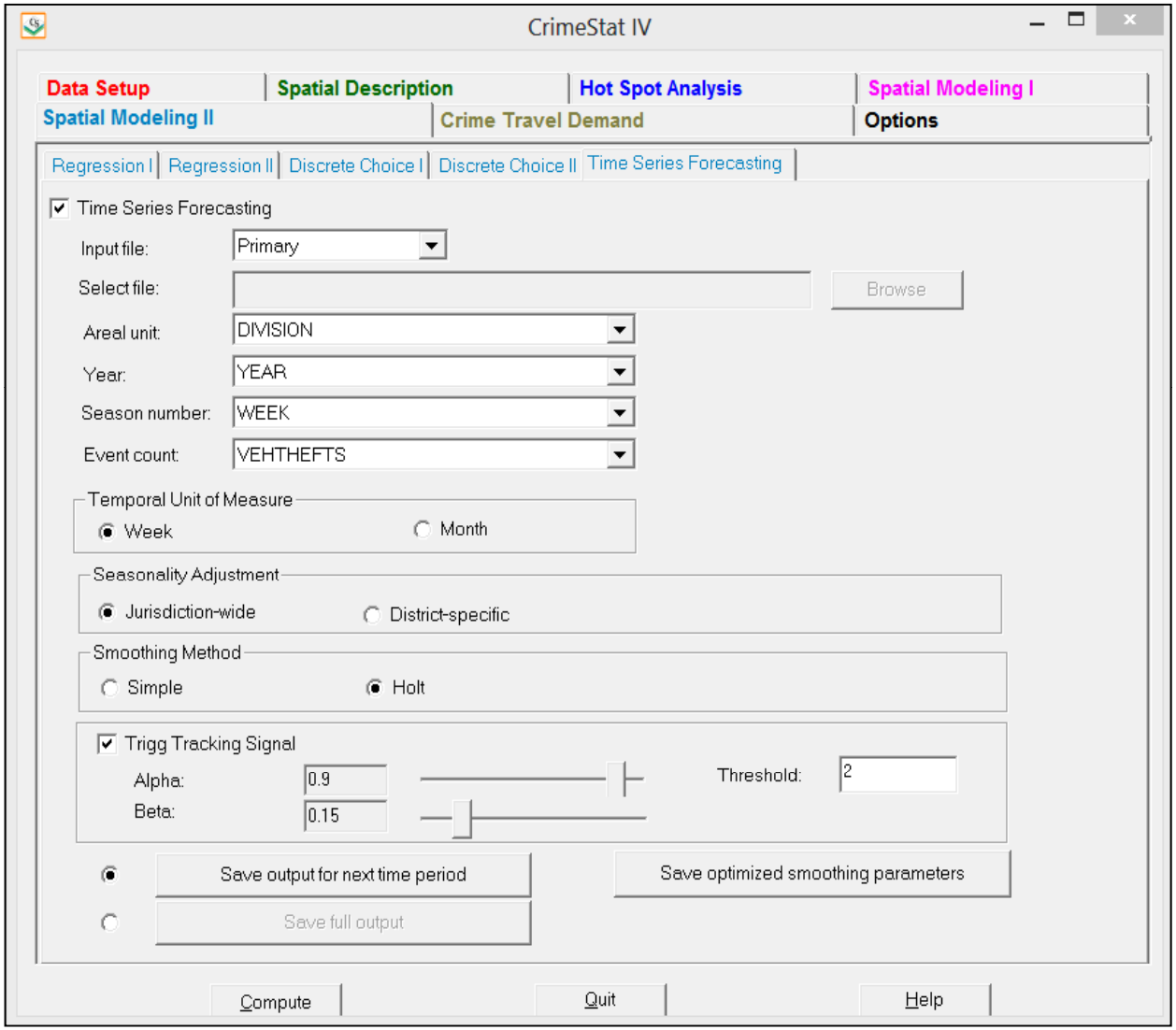
The year is the calendar year such as 2012 of each data record. This must be recorded. As mentioned above, there must be at least three years of data.

Season number

The season number is a unique temporal identifier. With this module, only months or weeks are allowed. Thus, the season number is 1 (January) through 12 (December) for months

Figure 24.1:

Time Series Forecasting



and 1 (first week of the year) through 52 (last week of the year) for weeks. Note that there cannot be partial weeks.

Event count

This is the count of the number of events for a given areal unit, year, and time period.

Temporal Unit of Measure

This defines the type of season used, either week or month.

Seasonality Adjustment

The seasonality adjustment is the adjustment made for each time observation for seasonal patterns such as when, for example, crime is low in February and high in July relative to the time series trend line. The routine uses:

1. **Jurisdiction-wide**. Either data from the entire jurisdiction (e.g., the entire city) and applies this to each district or

2. **District-specific**. Individual data from each district so that each gets its own unique seasonal pattern.

In Figure 24.1, the choice is jurisdiction-wide, which generally provides more accurate forecasts overall because district-specific seasonal factors are overly influenced by individual data points.

Smoothing Method

There are two alternative smoothing models, simple smoothing or Holt exponential smoothing:

1. **Simple smoothing** assumes that there is no trend and that future values will follow the mean of recent past values. Estimated means for data are weighted with weights summing to one for all data points but falling of exponentially with the age of data points.

2. **Holt smoothing** adds a smoothed time trend line into the expected number of future events. The models have smoothing parameters which determine how quickly

exponential smoothing "forgets" the past. The larger a smoothing parameter, the more quickly weights fall off with data point age.

The routine automatically chooses smoothing parameters by minimizing one-step-ahead forecast errors. It uses jurisdiction-level optimization of simple exponential smoothing's smoothing parameter for a single time series for the entire jurisdiction. The result is a smoothing parameter that is relatively small and does a good job of ignoring large changes and, therefore, yields good counterfactual forecasts. If there is a strong time trend (increasing or decreasing) in time series, then Holt is the better choice. Also, given that the option of optimizing smoothing parameters by individual district (instead of by the entire jurisdiction) is only available for Holt in CrimeStat, Holt is the better choice for estimating widely-differing time series patterns across districts and highly dynamic time series patterns within districts. For detection, Holt is likely more conservative because it captures more change in model estimates and, therefore, issues fewer signal trips than Simple Exponential Smoothing.

Trigg Tracking Signal

The Trigg Tracking Signal provides a test statistic for unusual activity in the number of events. If the absolute value of the signal exceeds a pre-specified threshold value, then there is a "signal trip" meaning that it is likely that there is an unusual change in events. The signal has three parameters with default values provided, alpha, beta and the threshold value.

1. Alpha is a smoothing parameter that varies between 0 and 1. An alpha of 0.9 (the default value) makes the tracking signal very reactive to current data on the anticipation of changes in a time series pattern. Note that "Alpha" is the same name for a parameter as used in Simple Exponential Smoothing for forecasting, but here is used to smooth the Trigg tracking signal instead of crime counts.

2. Beta is a smoothing parameter that varies between 0 and 1. A value of beta of 0.15 (the default value) smooths the measure of spread used to standardize the Trigg signal and retains a good amount of history while allowing estimates to drift and follow changing spread in the data.

3. The threshold is the value of the Trigg Tracking Signal that indicates whether the expected number of events will be greater than what is normally expected ("business-as-usual"). The default threshold of 1.5 is somewhat liberal in the sense that it will signal more periods of unusual activity. However, most police organizations would rather respond to more expected events even if the increased activity does not materialize (i.e., are false positives) than not respond and have events blow up. To use more conservative values, try 1.75 or 2.0 to get fewer signal trips.

Cohen, Garman, and Gorr (2009) found that the default values in the CrimeStat module are the best-performing parameter values. However, the user can experiment. Making alpha smaller than 0.9 will reduce the importance of recent events and give more sensitivity to increases building over several time periods. Similarly, increasing beta above 0.15 will smooth the data less and make the Trigg more reactive to changing variability in time series.

Running the Time Series Forecasting Module

With variations in seasonal adjustment and smoothing method, there are 8 possible models that can be run: four for weekly data and four for monthly data (Table 24.1).

Table 24.1:
Time Series Forecast Combinations

Season Number	Seasonality	Smoothing Method	Optimization
Weekly data	Jurisdiction-wide	Simple smoothing	Jurisdiction
Weekly data	District-specific	Simple smoothing	Jurisdiction
Weekly data	Jurisdiction-wide	Holt smoothing	District
Weekly data	District-specific	Holt smoothing	District
Monthly data	Jurisdiction-wide	Simple smoothing	Jurisdiction
Monthly data	District-specific	Simple smoothing	Jurisdiction
Monthly data	Jurisdiction-wide	Holt smoothing	District
Monthly data	District-specific	Holt smoothing	District

Output

There are three types of output: full, one-step ahead, and the optimized smoothing parameters. The first two outputs produce the following calculated values:

1. DE_SEASON is the number of events per period (EVENTCOUNT) divided by the seasonal factor for the current observation's season (December) and, thus, is a de-seasonalized count of events. To calculate the seasonal factor for each record divide EVENTCOUNT by DE_SEASON.

2. SMTH_LEVEL is the smoothed estimate for the current observation (e.g., December 2012).

3. When using the Holt smoothing method, there is one additional estimated parameter. SMTH_SLOPE is the change in estimated crime for each step ahead. If, for example, you need the forecasts for February 2013 and your current time period is December 2012, you add two times SMTH_SLOPE to SMTH_LEVEL because February 2013 is two steps ahead of December 2012.

4. SQ_ERROR is the squared forecast error of the current observation from the forecast made for it from the previous period (e.g., November 2012 if the current period is December 2012).

5. TRIGG is the value of the Trigg Tracking Signal for the current observation.

6. SIGNALTRIP indicates whether the Trigg level was higher than the threshold. If it was, this field will have a **1** to indicate that the Trigg value was greater than or equal to the threshold selected and the detected change is an increase, a **-1** if the Trigg value is greater than or equal to the threshold but the detected change was a decrease, and a **0** otherwise.

7. FORECAST is the one-step-ahead forecast, for the next observation in time (e.g., January 2013 if the current period is December 2012). For a January 2013 forecast and simple exponential smoothing it is SMTH_LEVEL for December 2012 multiplied by the seasonal factor for January 2013. For January 2013 and Holt smoothing it is the sum of SMTH_LEVEL and SMTH_SLOPE times the seasonal factor for January 2013.

Full Output

First, the full output includes all the input fields plus the calculated values. If the user clicks the option button for Save full output button and then clicks the Save full output button, a save output window opens (Figure 24.2). Select dBase 'DBF' for the Save output to field, browse to the folder of your choice, and type a file name (Run99.dbf in Figure 24.2). Both the input data and the one-step ahead forecast are output to the screen and to a 'dbf' file. The file will be saved with a "TS_F" prefix before the defined file name, Run99.dbf, resulting in TS_FRun99.dbf.

Figure 24.3 is an example of the full output from the Pittsburgh monthly data given the selections made in Figure 24.1. This output is useful for not only seeing current conditions but also the history of a district. For example, census tract 20300 has had a lot of unusual activity according to the Trigg signal. It had an unexpected decline in August 1998, two exceptionally high values in November and December 1998 and another two in April and May 1999. So if

Figure 24.2:

Defining Full File Output

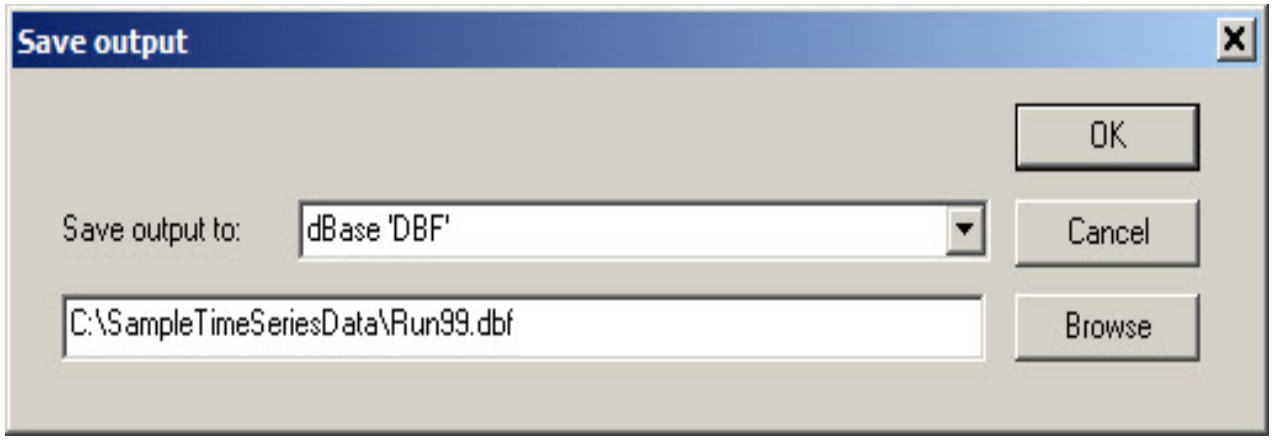


Figure 24.3:

Example Full Output from Pittsburgh Monthly Data

	A	B	C	D	E	F	G	H	I	J
1	AREA	YEAR	SEASON	EVENTCOUNT	DE_SEASON	SMTH_LEVEL	SQ_ERROR	TRIGG	SIGNALTRIP	FORECAST
341	20300	1998	4	2	2.11	1.85	0.09	0.37	0	1.86
342	20300	1998	5	1	1.00	1.75	0.74	0.76	0	1.62
343	20300	1998	6	4	4.34	2.06	6.68	1.76	1	2.23
344	20300	1998	7	2	1.85	2.04	0.05	0.00	0	2.13
345	20300	1998	8	0	0.00	1.79	4.14	1.60	-1	1.71
346	20300	1998	9	3	3.15	1.95	1.84	0.80	0	2.08
347	20300	1998	10	1	0.94	1.83	1.03	0.73	0	1.81
348	20300	1998	11	5	5.07	2.22	10.50	1.86	1	2.47
349	20300	1998	12	6	5.39	2.60	10.03	1.92	1	2.39
350	20300	1999	1	2	2.18	2.55	0.18	0.00	0	2.45
351	20300	1999	2	3	3.12	2.62	0.32	0.34	0	2.61
352	20300	1999	3	2	2.00	2.54	0.38	0.38	0	2.41
353	20300	1999	4	6	6.34	3.00	14.41	1.93	1	3.01
354	20300	1999	5	6	5.98	3.36	8.85	1.62	1	3.10
355	20300	1999	6	2	2.17	3.21	1.42	0.40	0	3.48
356	20300	1999	7	2	1.85	3.05	1.87	0.82	0	3.20

census tract 20300 had a positive signal trip in the last time period in the series, most likely it would have been a true positive needing police intervention, given its history of past flare-ups.

Next Time Period Output

Second, the next time period output includes only the calculated fields for both the screen and saved file. The word "next" refers to the forecast made for the next time period, while the Trigg tracking signal evaluates the current period. Again, in the dialog for saving the output file, type the .dbf extension in the chosen file name. The file is saved as a 'dbf' file with a "TS_C" (for 'current') prefix, resulting in TS_CRun2.dbf if Run2.dbf were entered by the user. The same field definitions as used in full output apply.

Figure 24.4 shows a sample output, which provides a scan of the entire jurisdiction for the current time period. The last time period in the corresponding input data set was December 1999, so this was taken as the current time period. Here you can see that the first 11 areas shown appear to have ordinary crime levels for December 1999 but the last four areas have unusual activity, three large increases and one large decrease. The forecasts of this output, just as for the full output, are "business-as-usual" forecasts for the next time period, January 2000.

> Note that the user must **choose** between full output and next time period output. Only one of these can be output for a single run.

Optimized Smoothing Parameters Output

The third type of output shows the results of the optimization process for exponential smoothing. This provides information on the parameters used to optimize the smoothing for each district. Define the file name and it will be saved as an ASCII text file with a 'txt' extension. Figure 24.5 illustrates the major output of the optimized smoothing parameters file, in this case for all selections in Figure 24.1 except the smoothing method which is now Holt.

1. Optimum Alpha is the smoothing parameter value for *level* of a time series that minimizes the one-step-ahead forecast sum of squared errors.

2. Optimum Gamma is the smoothing parameter value for *time trend slope* of a time series that minimizes the one-step-ahead forecast sum of squared errors.

3. SSE is the resulting optimal sum of squared errors for the time series.

Figure 24.4:

Example Output for Next Period Forecast

	A	B	C	D	E	F	G	H	I	J
1	AREA	YEAR	SEASON	EVENTCOUNT	DE_SEASON	SMTH_LEVEL	SQ_ERROR	TRIGG	SIGNALTRIP	FORECAST
17	51100	1999	12	1	0.90	1.57	0.58	0.81	0	1.44
18	60300	1999	12	2	1.80	2.05	0.08	0.05	0	1.88
19	60500	1999	12	1	0.90	0.54	0.16	0.77	0	0.50
20	70300	1999	12	0	0.00	0.73	0.69	1.37	0	0.67
21	70500	1999	12	0	0.00	0.80	0.83	1.35	0	0.73
22	70600	1999	12	0	0.00	0.21	0.06	0.64	0	0.19
23	70800	1999	12	0	0.00	0.65	0.54	1.21	0	0.59
24	70900	1999	12	1	0.90	1.13	0.07	0.24	0	1.03
25	80200	1999	12	0	0.00	0.98	1.23	0.80	0	0.90
26	80400	1999	12	3	2.69	1.52	1.79	1.39	0	1.39
27	80600	1999	12	1	0.90	0.51	0.20	0.71	0	0.47
28	80700	1999	12	2	1.80	0.71	1.53	1.72	1	0.65
29	80900	1999	12	0	0.00	2.38	7.32	1.64	-1	2.18
30	90100	1999	12	2	1.80	0.83	1.20	1.66	1	0.76
31	90200	1999	12	4	3.59	1.74	4.43	2.02	1	1.59

Figure 24.5:

Example Optimized Smoothing Parameters Output

Pittsburgh Monthly Data

Holt Exponential Smoothing Results

Data File:
Output File:
Deseasonalization Level: District
Optimum Sum of Squared Errors: 38140.61
Optimum Mean Square Error: 2.27
Trigg Alpha : 0.90
Trigg Beta: 0.15
Trigg Threshold: 1.50
Results by District:
District: 10300 Optimum Alpha: 0.03 Optimum Gamma: 0.01 SSE: 771.71
District: 20100 Optimum Alpha: 0.27 Optimum Gamma: 0.04 SSE: 2800.01
District: 20300 Optimum Alpha: 0.03 Optimum Gamma: 0.28 SSE: 230.15
District: 30500 Optimum Alpha: 0.07 Optimum Gamma: 0.14 SSE: 713.50
District: 40200 Optimum Alpha: 0.18 Optimum Gamma: 0.04 SSE: 485.49
District: 40300 Optimum Alpha: 0.11 Optimum Gamma: 0.01 SSE: 63.12
District: 40400 Optimum Alpha: 0.04 Optimum Gamma: 0.22 SSE: 328.64
District: 40500 Optimum Alpha: 0.26 Optimum Gamma: 0.04 SSE: 621.25
District: 40600 Optimum Alpha: 0.03 Optimum Gamma: 0.29 SSE: 126.85
District: 40900 Optimum Alpha: 0.01 Optimum Gamma: 0.03 SSE: 201.86
District: 50100 Optimum Alpha: 0.17 Optimum Gamma: 0.10 SSE: 986.29
District: 50600 Optimum Alpha: 0.16 Optimum Gamma: 0.13 SSE: 176.69
District: 50700 Optimum Alpha: 0.10 Optimum Gamma: 0.01 SSE: 288.04
District: 50900 Optimum Alpha: 0.01 Optimum Gamma: 0.39 SSE: 368.07

It is valuable to review the optimal parameters to see which areas have stable versus dynamic time series. Note that for the Trigg calculation, we want a large alpha to detect large changes in the number of recent events. That is why the default value of alpha is 0.9. However, for forecasting, we want a low alpha in order to smooth the data to produce a stable forecast.

For example in Figure 24.5 district 40900 has a very stable time series with low alpha and gamma while district 20100 has a dynamic level with large alpha and district 40600 has a dynamic time trend with high gamma. Therefore, the forecast for district 40900 is liable to be more accurate than for district 20100 but district 20100 is liable to have more accurate detections of large crime increases.

Guidelines for Running Forecast Models

Each model that is run for a fixed set of data will produce slightly different output. Based on our experience, we have found that there is not a single model type that will cover all data sets. Indeed, much of the forecasting literature for the last 35 years has been attempting to design and verify rules on which forecast model to use in different situations and for different data. Each jurisdiction will have its own unique characteristics in its time series data and districts within the same jurisdiction will often differ in their characteristics. Consequently, an analyst must experiment with this framework to identify which parameter choices produce the best overall fit for that jurisdiction and for the individual districts.

However, some guidelines can be provided based on our studies with this methodology.

1. Detection does not need very accurate forecasts, but just reasonable values because large changes in a time series are fairly easily detected. Detection does not attempt numerical accuracy but just a binary categorization, either a crime count is exceptional or not. That reduces the need for forecast accuracy of the counterfactuals. We recommend using Simple Exponential Smoothing with jurisdiction seasonality and optimization for detection.

2. Gorr, Olligschlaeger, and Thompson (2003) provide the guideline from empirical testing, that to get good forecast accuracy (20% mean absolute forecast error or less) crime time series should average at least 25 crimes or so per time period (month or week). It is hard to get that volume of crime unless using crime aggregates (such as all serious property crimes) and fairly large districts (tracts or police divisions). Very large cities such as New York, Los Angeles, Chicago, Houston, Philadelphia or Phoenix may have sufficient crime levels in some areas to provide good accuracy by week and for some crime types (e.g., vehicle theft, burglaries). It is the high-crime areas that are the most important and need the best forecasting accuracy as well as good detection accuracy for tactical

deployment of police. The low-crime areas do not need as good forecast accuracy for resource allocation as much as they need good detection for large crime increases, which is easily obtainable.

3. Seasonal factors are the least accurately estimated parameters in univariate time series models because a season, such as month (e.g., July, 7) or a week (e.g., 23) is only observed once a year. If an analyst has five years of data, then there are only five observations per seasonal adjustment. Gorr, Olligschlaeger, and Thomson (2003) found that jurisdiction-wide seasonality is about 10% more accurate for forecasting than district-level seasonality (estimated separately for each district using district data).

4. Different districts, however, have different seasonal patterns, some widely different from each other. For example, motor vehicle thefts peak in Pittsburgh in summer months in most areas but the major university area has a trough in summer because students (and their cars) are mostly gone then. For some applications, then, it may be valuable to give up some forecast accuracy to gain information (albeit noisy) on seasonal patterns by district. To obtain the most detailed forecasts tailored to each district, we recommend district seasonality with the district optimization of smoothing parameters available with the Holt method.

5. There is an objective way to choose forecast models in Table 24.1. The CrimeStat output includes a sum of squared errors term for each district from minimizing one-step-ahead forecasts in selecting smoothing parameters. Average these for a random sample of districts a across alternative models and select the method with the minimum sum of squared errors.

Overall, for detection/early warning in small-to medium-sized cities, the authors recommend using monthly data, simple exponential smoothing, jurisdiction-wide seasonality and smoothing parameter optimization. For the most informative and tailored one-step-ahead forecasts, we recommend using monthly data, Holt Smoothing, district seasonality and optimization of smoothing parameters. For large metropolises, however, it may be possible to run weekly forecasts as long as the average number of events per week is 25 or more in high crime areas.

Current research under way by one of the authors suggests that once detected, crime flare ups in areas tend to continue for some stretch of time with some periods on and an others off. Figure 24.3 showed an example of this behavior. This research suggests that police can adopt decision rules to maximize their effectiveness such as “Each time a crime flare up is detected, maintain extra police resources in the area for a fixed number additional time periods.” For example, the research has experimented with one through three period stretches of time for

prevention efforts, each time a signal is tripped. Such rules can expose a relatively large number of crimes to prevention while maintaining reasonable workloads for police officers on patrol (Gorr & Lee, 2013).

Counterfactual Detection v. Forecasting

The time series forecasting module has two main purposes: 1) as a signal for early **detection of large changes** in time series patterns; and 2) as a **forecast** for the next time period into the future to aid resource allocation decisions.

Signal detection needs a forecast of the level of a time series under "business-as-usual" conditions for the most recent data point to answer the question "Does this data point seem unusual?" The counterfactual forecast applies exponential smoothing to all data before the most recent time period in the time series to forecast its data point. Exponential smoothing is ideal for making counterfactual forecasts because by definition it largely ignores recent large changes by smoothing them with a relatively low weight, even for the last data point used for estimation. Then if the difference between the forecast of what was expected versus the actual value is large, we have a signal trip.

Time series forecasting for the next or future time periods, on the other hand, is different in that when we make the forecast we do not already have the future actual values. As time passes after a forecast is made, eventually the future is realized and we get the actual values. Then we can calculate forecast errors "after the fact". The module also uses the same exponential smoothing routines for forecasting as it does for making counterfactual but for different purposes. Smoothing for detection tunes the signal for how reactive the signal is to current large changes as opposed to a series of smaller but accumulating changes while smoothing for forecasting estimates modules that "drift" with the data, albeit with a lag, to self-adapt to changing conditions.

However, we believe that major value of exponential smoothing for police is in providing counterfactual forecasts for detection while forecasts of the future necessarily are limited to extrapolations of known patterns under the existing conditions (i.e., with no surprises). The user should be very cautious with forecasting and not assume that the forecast will necessarily come to pass. The forecasts are useful for extrapolating what is to be expected *if* current conditions persist.

Example with Pittsburgh Monthly Crime Data

Using the sample data for Pittsburgh monthly part 1 crime counts, the monthly data was run with jurisdiction-wide seasonality and simple smoothing. Figure 24.6 shows a choropleth

map of Pittsburgh of serious violent crimes for December 1999 along with tracts that have signal trips. Out of the 140 tracts, there are 19 (or 13.5%) that have signal trips for increases and 8 (or 5.7%) that have signal trips for decreases. Clearly, police are more interested in the areas where crime increased than in areas where crime decreased. Nineteen seems manageable for investigation by crime analysts and targeted patrol or other police interventions.

Figure 24.7 shows the map zoomed into tract 261400 which has a signal trip for an increase and a current crime count of 6. There are three serious violent crimes recorded at the same street intersection (at different dates) which display as one point in the map, two crimes adjacent to each other, plus a crime near the border of the tract. Research on crime hot spots (e.g., Weisburd, Bushway, Lum, & Yang, 2004; see Chapters 7, 8 and 9 of the *CrimeStat* manual) and current research on crime flare ups show that they tend to occur in very small areas ("micro areas" on the order of blocks) so when detected at the tract level, the crime analyst can zoom in and specify small subareas for targeted patrol.

Conclusion

CrimeStat's Time Series Forecasting Module provides a crime early warning system for police that is comprehensive, uses simple but proven methods, and is easy to use. A crime analyst can scan all the districts or areas of an entire jurisdiction in a single run of the module and see which ones are starting large crime increases (or decreases). Once detected, police have the option and ability to intervene with directed patrol and other means to prevent additional crimes in affected districts. The Time Series Forecasting Module also provides extrapolative forecasts of expected crimes for the next week or month by district, which can aid resource allocations by police.

A major limitation of any approach to working with crime time series data for tactical deployment of police resources (e.g., "Where should we target patrols this week?") is that the size of area units needs to be small, certainly patrol districts and smaller, but then the associated time series data has relatively low crime counts and any estimated models have sizable estimation and prediction errors. In such a situation it is better to use simple models, if for no other reason than there is very little else to get out of the limited data than what simple models can find. In our academic research we have used many complex models on this kind of data, from spatial multiple regression, to Bayesian versions of spatial regression models, to neural networks, and to spatial scan statistics without much more benefit than is available now in this chapter. Nevertheless, time series analysis of crime data does add value to crime analysis. We recommend using the automated detection methods of this chapter for early warning of crime increases in conjunction with crime mapping and other sources of information to diagnose and respond to emerging crime-area problems.

Figure 24.6:

Pittsburgh Violent Crimes in December 1999 with Signal Trips

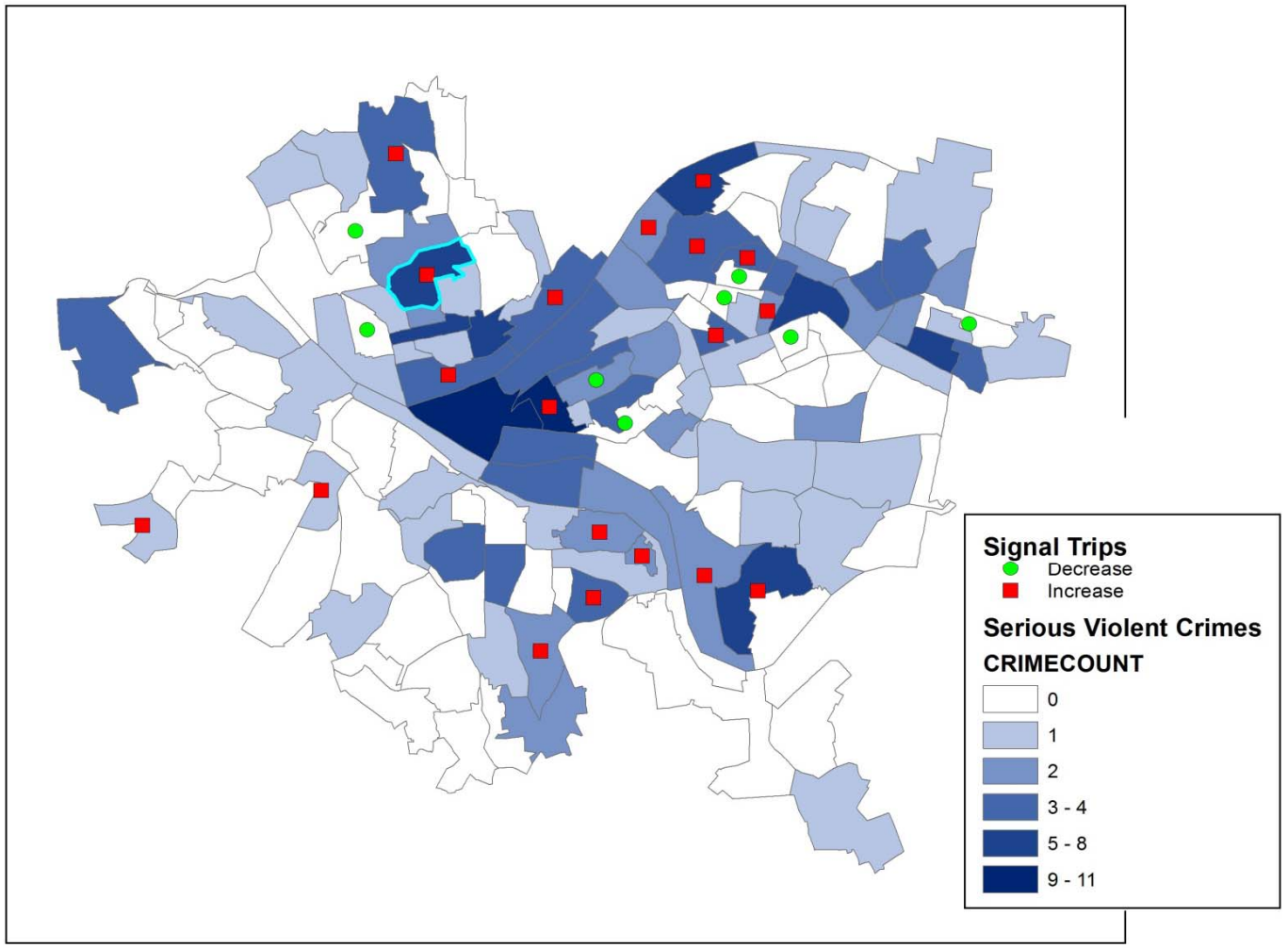
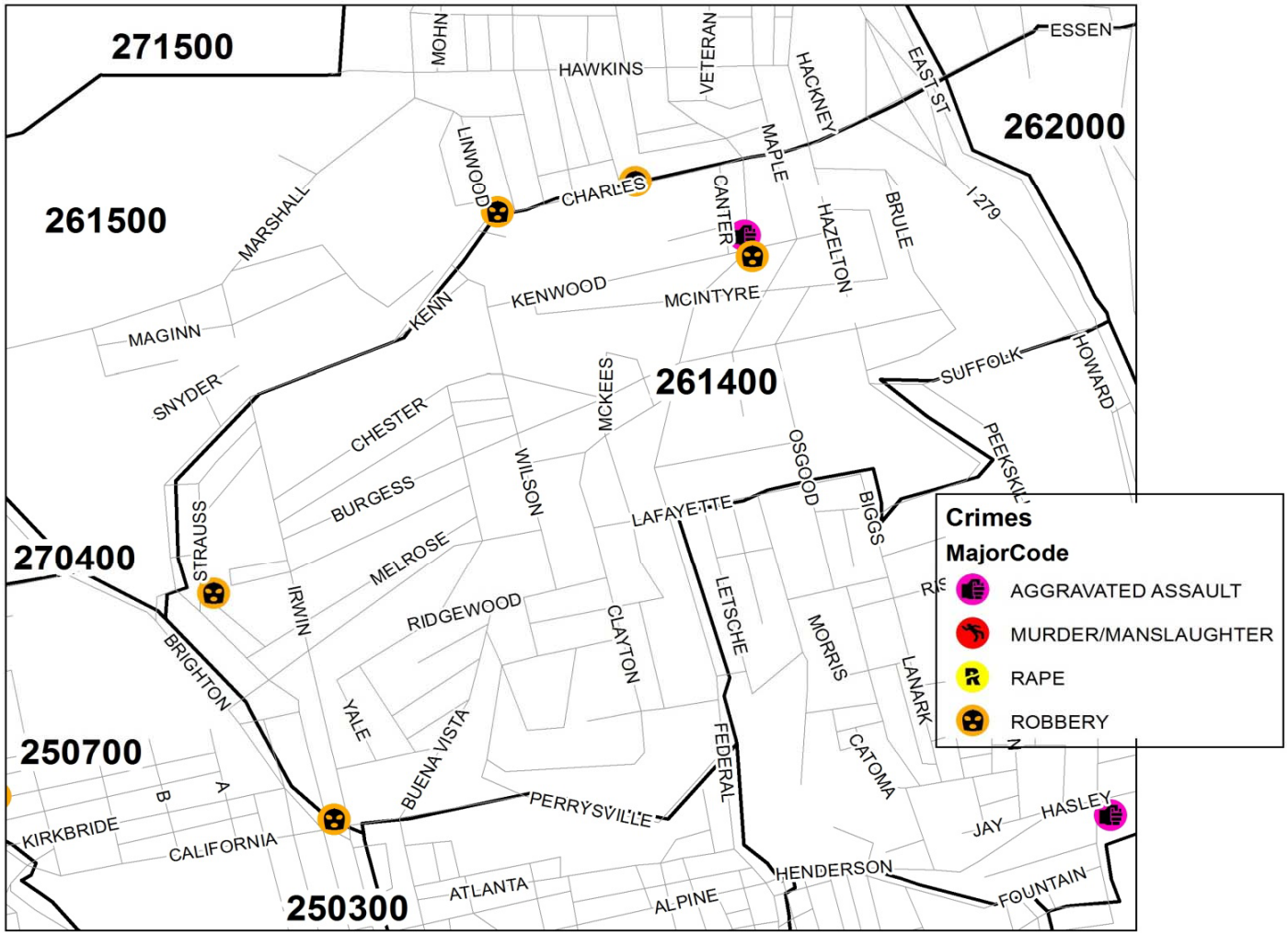


Figure 24.7:

Tract 261400 Showing Crimes Causing Signal Trip



References

Cohen, J., Garman, S. & Gorr, W. L. (2009). Empirical calibration of time series monitoring methods using receiver operating characteristic curves, *International Journal of Forecasting*, 2009, 25(3), 484–497.

Gorr, W. L. (2009). Forecast accuracy measures for exception reporting using receiver operating characteristic curves, *International Journal of Forecasting*, 2009, Vol. 25(1), 48–61.

Gorr, W. L. & Kurland, K. S. (2012). *GIS Tutorial for Crime Analysis*, Esri Press, Redlands. See chapter 9, Preparing incident data for mapping, especially Tutorials 10-1 and 10-2 of Chapter 10 for automating those steps).

Gorr, W.L. & Lee, Y. J. (2013). Early warning system for crime hot spots, Heinz College, Carnegie Mellon University Working Paper Series (http://www.heinz.cmu.edu/faculty-and-research/research/research-details/index.aspx?rid=482).

Gorr, W. L., Olligschlaeger, O. M. & Thompson, Y. (2003). Short-term forecasting of crime, *International Journal of Forecasting, Special Section on Crime Forecasting*, 19(4), 579–594.

Neill, D. B. (2009). Expectation-based scan statistics for monitoring spatial time series data. *International Journal of Forecasting* 25: 498–517.

Weisburd, D. L., Bushway, S., Lum, C., Yang, S. (2004). Trajectories of crime at places: a longitudinal study of street segments in the city of Seattle. *Criminology* 42:283−321

CrimeStat IV

Part VI: Crime Travel Demand Forecasting

Chapter 25:

Overview of Crime Travel Demand Modeling

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Chapter 25:

Overview of Crime Travel Demand Modeling

The next seven chapters present a module on crime travel demand modeling. Crime travel demand modeling is a framework for examining crime trips over an entire metropolitan area. In this chapter, an overview is presented. In this and the next five chapters, each of the separate components of crime travel demand modeling are presented. Finally, in Chapters 31 and 32, Richard Block and Dan Helms present case studies of the method applied to Chicago and Las Vegas crime data.

Much of the theoretical background was discussed in Chapter 13 (Journey-to-crime). Readers would be advised to review that material before proceeding with the crime travel demand model.

Travel Demand Forecasting

Crime travel demand modeling is an application of travel demand forecasting (or travel demand modeling). It is used by transportation planners for examining travel patterns over an entire metropolitan area and for forecasting future trends. It is a model of transportation patterns in a metropolitan area and is used for both forecasting and the analysis of the likely effects of building new roadways or installing new transit facilities. In the United States, it is required by Federal law to be used in every metropolitan area greater than 50,000 population as a basis for making decisions on highway and transit expenditures (USDOT, 2003: 23CFR450). It is also used for transportation planning in the metropolitan areas of many other countries of the world (Field & MacGregor, 1987).

The aim is to model travel over an urban area as a means for coordinating the approximately $36 billion dollars in transportation highway funds and $8.6 billion in transit funds that are spent *every* year in the U.S. (BTS, 2007). Rather than waste funds by building new roadways and transit facilities that will be little used, it is a lot more effective to first model the likely benefit of a new facility as a basis for making a decision to build it. In essence, Congress requires a transportation model be developed for every metropolitan area as part of an evaluation of the benefits to be obtained from particular transportation investments.

The framework has emerged slowly since the 1950s and is now starting its ‘third generation’. For the ‘first generation’ - what is used by most Metropolitan Planning Organizations (MPO) today, modeling is conducted entirely at a zonal level. The ‘second generation’ involves modeling individual level choices in travel mode taken within a zonal

framework (Horowitz, Koppelman, & Lerman, 1986; McFadden, 2002), while a 'third generation' involves modeling individual-level trips in a framework known as 'activity-based' modeling (FHWA, 2009; Culp & Lee, 2005). In *CrimeStat IV*, we implement a modified 'first generation' model, primarily due to the lack of data on individual-level crime trips. In later versions, we may add individual-level choice components.

Need for More Complex Travel Model of Crime

Crime travel demand modeling is an application of travel demand theory targeted specifically to crime analysis. There are many reasons why such an approach is appropriate. First, current models of criminal travel behavior are too simple with respect to travel. As Chapter 13 discussed, journey-to-crime models assume that many offenders commit crimes in their neighborhoods. While this assumption is frequently empirically found, it is not a realistic model of modern day crime travel. Prior to World War II, Americans tended to live and shop almost exclusively in their residential community. Many people would grow up and live in a single community for most of their lives. Since World War II, however, American society has become very mobile. People move frequently, not just within metropolitan areas, but between metropolitan areas. For example, between 2009 and 2010, at the height of the Great Recession, 28.5 million persons moved homes (U.S. Census, 2010a). This was down from 1999-2000 where 43.4 million moved (Schacter, 2001) but still represented a substantial amount of movement. More than two-thirds of the households who moved (68%) stayed in the same county but 12 percent moved to a different State.

Second, the almost universal use of personal automobiles has increased daily mobility. For example, in the 2010 census, 91% of households owned at least one motor vehicle, an increase over 2000 where it was 86% (NHMC, 2012; U.S. Census, 2003; Aizcorbe & Starr-McCluer, 1996). For certain metropolitan areas, particularly in the west and in the south, motor vehicle ownership was greater than 92%. Even in cities with lower vehicle ownership, more than half the population do own vehicles (e.g., New York City had 56% of households with one or more vehicles in 2000; Wikipedia, 2012a).

Further, per capita vehicle travel has consistently increased over time. Since at least 1960, and probably before, the growth in vehicle miles traveled (VMT) has increased at a much faster rate than population, a trend that does not seem to be abating (NAP, 2009; BTS, 2003; FHWA, 1996). Essentially, automobile use has become almost ubiquitous. There is no reason to think that offenders would not be affected by these trends. Since there is no data available that could test whether offenders are less likely to own an automobile than non-offenders, it has to be assumed that more and more offenders have access to an automobile for the use of committing a crime. Clearly, the existence of an automobile makes crime travel much more fluid and difficult to model. While offenders will probably commit crimes in locales for which they are familiar,

there is no reason to think that those locales will necessarily be the communities in which they live.

Third, the widespread availability of motor vehicles has allowed major shifts in intra-urban travel patterns. In the last census (2010), approximately half the U.S population lived in areas that would normally be called 'suburbs', even though the U.S. Census Bureau does not use this nomenclature (They use Outlying Counties within a Metropolitan Statistical Area type of Core Based Statistical Areas; Wikipedia, 2012b; OMB, 2010). Within metropolitan areas, approximately two-thirds of the population lives in suburban areas.

Much of the community-oriented crime patterns that were described by the so-called 'Chicago School of Criminology' in the 1920s and 1930s are no longer true (Burgess, 1925; Thrasher, 1927). Crimes have decreased in both the central cities and in the suburbs throughout the U.S. between 1990 and 2008 (Kneebone & Raphael, 2011). However, the differentials between the central city and the suburbs have decreased over time. In 1990, for example, the differential in violent crime between the central cities and the suburbs was 2.8 times whereas it was 2 times in 2008. Similarly, the differential in property crimes between the central cities and the suburbs decreased from 2 times to 1.7 times over that period. The researchers noted that this gap decreased in nearly two-thirds of metropolitan areas in the U.S. These decreases are associated with a dramatic drop in crime within the central cities but a more gradual drop in the suburbs.

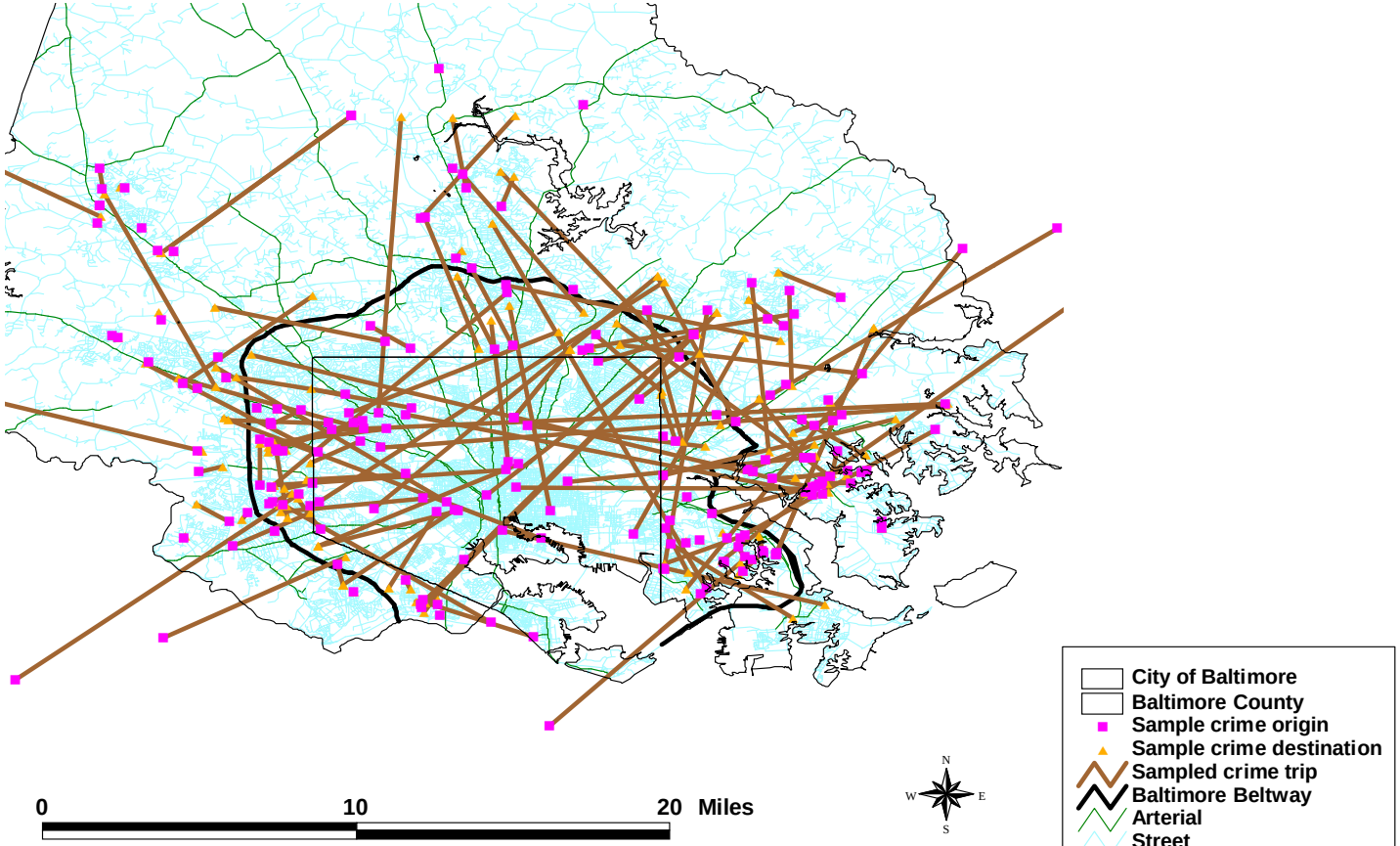
While crime has been decreasing in most metropolitan areas within the U.S., the travel patterns of offenders has become quite complex. Figure 25.1 below shows a sample of 200 crime trips in Baltimore County that occurred between 1993 and 1997. As seen, there is a complex pattern. Some of the trips are short; for some, the origin and destination are the same location. But, for other trips, the travel distances are substantial. In other words, there is a complex pattern of crime trips in Baltimore County which is not easily modeled by a simple distance decay-type function.

Fourth, an empirical examination of travel patterns shows considerable temporal variation. There are hourly variations, daily variations, and seasonal variation in crimes. Some of this can be understood as reflecting existing travel patterns in congested metropolitan areas. For example, in Baltimore County, crime travel distances were generally shorter during the peak afternoon 'rush hours' (4-7 PM) than at other times. Such a pattern suggests an adaptation to traffic by offenders, a not unreasonable assumption given the difficulties of traversing a metropolitan area during peak travel times.

Fifth, crime travel behavior represents a complex pattern in itself. Especially for personal crimes, there is an interaction in the travel patterns of offenders and victims that is very difficult

Figure 25.1:

Baltimore County Crime Trips: 1993-1997
Origins and Destinations
Sample of 200 Crime Trips



to even describe, least of all model. Many crimes are committed by multiple offenders and the existence of intermediate locations (e.g., 'fences' for the distribution of stolen goods, auto theft drop locations) makes crime travel even more of a complex pattern to be understood.

In short, American society has become very mobile, leading to larger travel distances, more frequent trips, and more complex trips. Again, offenders are going to be affected by these trends. Because of this, there is a need to understand crime patterns in terms of the complexity of travel rather than continue to rely on overly simple models of travel 'distance decay'.

Crime Travel Demand Framework

Crime travel demand theory is a framework for understanding this complexity. There are two phases:

1. An inventory (or data gathering) phase; and
2. A modeling phase.

The data gathering involves putting together the necessary data to estimate the model. This involves selecting an appropriate zone system (since the model is estimated at the zonal level), obtaining data on crime 'trips' and allocating it to the zones, obtaining zonal variables that will predict trips (both on the production side and on the attraction side), creating possible policy or policing interventions, and obtaining one or more modeling networks.[1]

The modeling phase involves four distinct modeling steps (or stages) that represent a logical 'causative' pattern:

1. **Trip generation** - separate models are produced of crime trip productions (i.e., the number of crime trips that originate in each zone) and crime trip attractions (i.e., the number of crime trips that occur in each zone). The model may include policy or intervention variables as predictors as well as socio-economic variables. One of the major uses of the model is to explore how different interventions might alter the number of trips taken.

2. **Trip distribution** - a model that predicts the number of crime trips that will begin in every production zone and will end in every attraction zone.

[1] In the usual travel demand modeling framework, data gathering is called a *land use inventory* and involves estimating population and employment by different land uses, particularly retail trade and several other types of industry.

3. **Mode split** - a model that predicts, for each production-attraction zone pair, which travel modes will be taken (e.g., walking, bicycle, driving, bus).

4. **Network assignment** - a model that predicts, for each production-attraction zone pair by travel model, which route is liable to be taken.

The modeling is typically sequential following these steps. The output from each stage is then used as an input for the subsequent stage. Figure 25.2 below shows the sequence.

One can think of the model as a *plausible* behavioral representation. First, someone decides to make a trip (e.g., an offender decides to commit a robbery to get some money to purchase drugs). That would be the first stage. Second, that individual decides where to go to commit the robbery. This is the second stage. Third, the individual decides how to travel to that location (walk, drive, or take the bus). This is the third stage. Finally, the individual chooses a route; in the case of walking, biking, or driving, that is a deliberate choice whereas in the case of transit trips, it is dependent on the actual bus or rail network. This is the fourth stage.

However, the analogy to behavioral decisions quickly breaks down as alternative behavioral sequences can be generated (e.g., the offender first makes a trip and then decides to commit a crime; the offender first decides to commit a crime and chooses a destination, but then commits a crime at an intermediate location in the trip). As a behavioral model, this type of framework is actually not very accurate for predicting individual behavior as a number of studies have suggested (Ortuzar &Willumsen, 2001; Domencich & McFadden, 1975).

Consequently, it is important to understand this framework as a *zonal* model, rather than a behavioral explanation. The data are aggregated at the zonal level and the model is applicable to that level. The model is good at predicting total trips in a metropolitan area and for predicting the major trip links, and should be used only at that level.

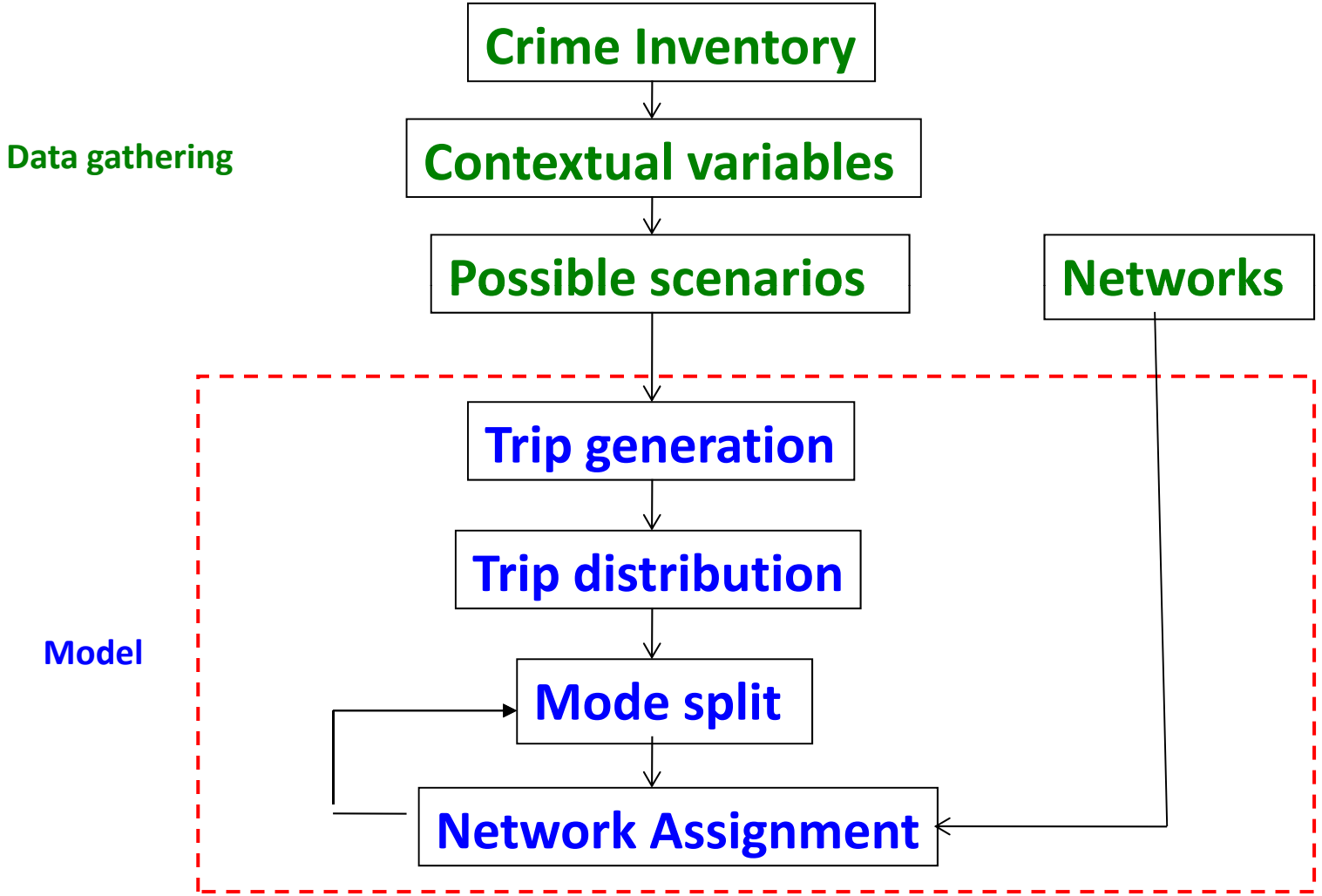
Note in Figure 25.2 that there is feedback from the network assignment stage to the mode split stage. This is a function of transit use since the choice of travel mode is dependent on the availability of an appropriate network (e.g., one cannot have train trips if there are no trains nearby).[2]

---

[2] In classic travel demand modeling, there are several feedback loops. One is from the network to the mode choice, as in the crime travel demand version. A second is from the network to both mode choice and trip distribution stage. If a particular route becomes very congested (having a traffic volume-to-capacity ratio greater than 1.0), it has been noted alternative destinations become more attractive. For example, people will often travel farther and more out of the way to avoid congested corridors. In short, there are a variety of feedbacks from later stages to earlier stages, and the model is quite flexible in being able to accommodate the different sequences.

Figure 25.2:

Crime Travel Demand Model



Also, crime travel demand modeling is a framework, rather than a specific theory. There is more than one way to implement the framework. In transportation modeling, there are many variations of the model and transportation planning organizations implement it in slightly different ways. For this reason, it is best thought of as a framework.

In this version of *CrimeStat*, we implement one particular version of the framework. It is a framework that is consistent and appears to produce reasonable predictions of crime travel behavior. But, clearly, it is not the only way that this could have been implemented.

The 'second-' and 'third-generation' travel demand models represent alternative ways of modeling travel in a metropolitan area. In the following chapters, these alternatives will be mentioned where appropriate. Nevertheless, the type of framework implemented in this version should be seen as a first step in developing a more realistic model of crime travel behavior.

Crime Travel Definitions

Let us start with two definitions.

Crime Trip

In the *CrimeStat* implementation, a **crime trip** is a round-trip journey from an offender's residence that includes a committed crime at a specified location. From a modeling perspective, the offender's residence will be considered the **origin** of the trip and the crime location will be considered the **destination**. Note that there may be intermediate trips between the origin and the destination, as Figure 25.3 below illustrates. But, at some point, the offender will probably return home to the initial origin. Defining a crime trip in this way avoids the issue of identifying the actual origin of the trip. As mentioned in Chapter 13, routine activity theory suggests that many crimes are committed while offenders are involved in other activities. The possibilities can become quite complex (e.g., an offender stays overnight at some other location than his/her residence and commits a crime as a part of that stay rather than while en route to home).

Nevertheless, by referencing all trips with respect to the offender's residence, a consistent set of estimates can be obtained. Since intermediate trips are almost never known, it is a hypothetical question whether modeling origins from offender residences will produce better estimates than modeling origins from other locations.[3]

---

3 *If* it were possible to obtain data on intermediate locations during crime trips by offenders, then it would be possible to test whether modeling the origin with respect to these intermediate locations produces more stable and clearer predictions than with respect to the residences of the offenders. But, until that data is obtained, the question is speculative.

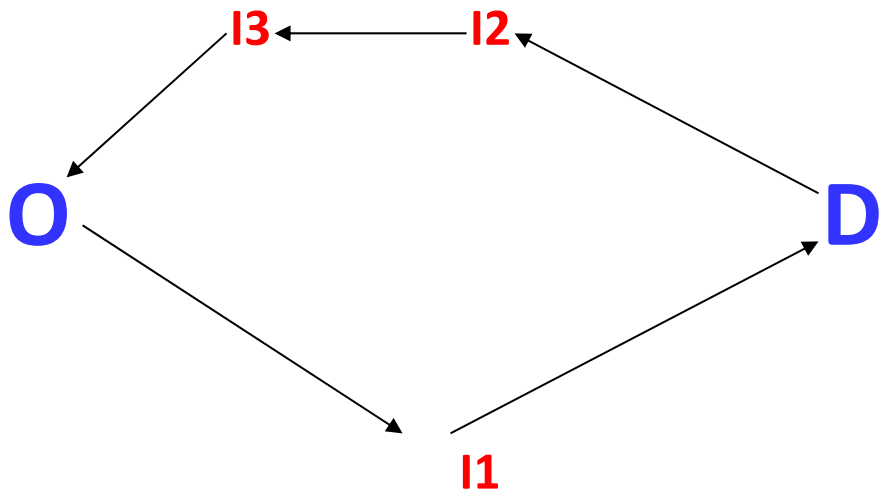
Figure 25.3:

Origin-Destination Links

There is an *origin* (residence)

There is a *destination* (crime location)

There may be *intermediate* links



In the usual travel demand forecasting framework, transportation modelers usually distinguish **productions** and **attractions** from origins and destinations. The reason is that origins are asymmetrical in time. For example, for a home-to-work (commuting) trip, the origin location in the morning is the residence while the destination is the work location. On the return trip, however, the origins and destinations are reversed (i.e., the work location is the origin while the home location is the destination). The models are referenced in the same way that is done here, namely from the residence location and the trips are assumed to be reciprocal. Thus, the production end of a trip is always the residence location and the attraction end of a trip is always the work location. The round-trip journey can be broken into different time sequences (e.g., morning home-to-work trips; afternoon work-to-home trips), but the production and attraction ends are always the same.

In crime travel demand modeling, there is usually data on intermediate trips. Consequently, some of the finer analysis cannot be done. Therefore, we adopt a similar logic, but with a slightly different terminology. As with the usual travel demand modeling, the production end is *always* the home location and the attraction end is *always* the crime location. However, we use origin and destination interchangeably with production and attraction since we cannot document the return part of a crime trip.

Crime Travel Demand

Crime travel demand is the number of offenders per unit time that are expected to travel on a given segment of the transportation system under a set of socioeconomic, land-use, and environmental conditions. That is, the final model output is an estimate of the number of trips (or offenders) that travel on any given segment of the transportation system at a given time under a given set of conditions:

Number of trips = number of offenders traveling to crime (25.1)

First, as mentioned above, the model is estimated sequentially. In the first stage, trip generation, there is a prediction of the number of crime trips that originate from each origin zone and the number of crime trips that occur (end) in each destination zone. In this case, a crime **trip** is equated with an offender because of the nature of arrest records from which these estimates come. With most arrest records, there is a single record for each crime that an individual commits. Thus, the origin is the residence location of the offender while the destination is the crime location. If the individual committed more than one crime, there will be a separate record for each crime (or, at least, those that are known). If two individuals commit a single crime and both are arrested, then there will be two records in the data base. In other words, the nature of the data equates a crime trip with a single offender. Thus, the total number of crime trips

estimated (whether from the production or attraction end) is equivalent to the number of offenders.

Aggregate Volume/count Model

Second, by a 'set of socioeconomic, land-use, and environmental conditions' is meant correlates of crime trips. At the aggregate level of a zone, predictors of crime trips (whether productions or attractions) are correlates of those trips. Since the number of trips are being predicted, the model estimates **volumes (**or counts), not rates.[4] That is, the number of crime trips originating in a zone or ending in a zone is a count of events. Aggregate counts, in turn, tend to be related to other aggregates, particularly population. Thus, in developing a predictive model, population is almost always one of the dominant variables. Sometimes it can be a sub-set of population, such as number of households, number of vehicles, or number of males aged 16-25. But, since the number of incidents is usually a function of the size, there can be difficulties in inferring individual characteristics from ecological models.[5] It is important to keep this distinction in mind and not make inferences about individuals.

In addition to population, variables that predict crime trips are also ecological variables - employment, retail space, number of bars, number of pawn shops, existence of a freeway, number of arterial lane miles, and so forth.

O-D Zone Pairs

In the second stage, trip distribution, a model is estimated of the number of crime trips that occur from any particular origin zone to any particular destination zone. Since the input to the second stage is the number of predicted crimes originating in each origin zone and the number of predicted crimes ending in each destination zone, the second stage estimates how many trips will be distributed from each origin zone to each destination zone. The result is an

---

4 Some agencies have actually used it to predict rates. Since a rate is an event relative to a baseline, population is factored into the dependent variable. It is possible to apply the model as a rate, though the user needs to ensure that all the predictor variables are also rates.

5 The question of whether an ecological inference is valid or not has been studied extensively. Sometimes it holds and sometime it does not. An ecological inference occurs when data are aggregated with a *grouping* variable (e.g., state, county, city, census tract; see Freedman, 1999; Langbein & Lichtman, 1978). The relationship is often called an *ecological fallacy*, but that is an oversimplification. Typically, if the between-group variance (i.e., differences) is greater than those within groups, then the ecological relationship will be a lot stronger than at an individual level. Conversely, if the within-group variance is greater than the between-group variance, a relationship that holds at the individual level will not be seen at the aggregate level. There are other ecological characteristics that account for typically higher R^2 at the aggregate level - spatial autocorrelation, skewness in the dependent variable, and hetereoscedasticity (unequal estimation errors around a statistical estimate).

estimate of crime trips between zone pairs (an origin zone and a destination zone). There are different names that are used for this combination - zone-to-zone trips; zone pairs; zone-to-zone links, O-D links (for origin-destination links), O-D pairs, but in all cases the term refers to the number of trips that start in any one origin zone that go to any one destination zone.

Travel Mode

In the third stage, mode split, the number of trips by any O-D combination are then split into different travel modes - walking, biking, driving, bus (if available) or train (if available). In the usual travel demand modeling done by transportation modelers, some of these modes are broken down very finely (e.g., drive alone trips, car pooling trips, park-and-ride trips). There is no logical reason why mode split cannot be defined in multiple ways. For our purposes in modeling crime trips, simple choices are probably adequate because of a lack of data that would allow finer distinctions to be made.

Estimating Travel Routes by Mode

Finally, in the fourth stage, the number of trips from any origin to any destination by separate travel modes are assigned to a route on the transportation network. Thus, if the trip is by walking, biking, or driving, the model may predict a different route than if the trip is by transit since a transit system is limited to particular bus or rail routes. Hence, the final stage is an estimate of the total number of crimes that occur on any segment of a transportation network by separate travel mode.

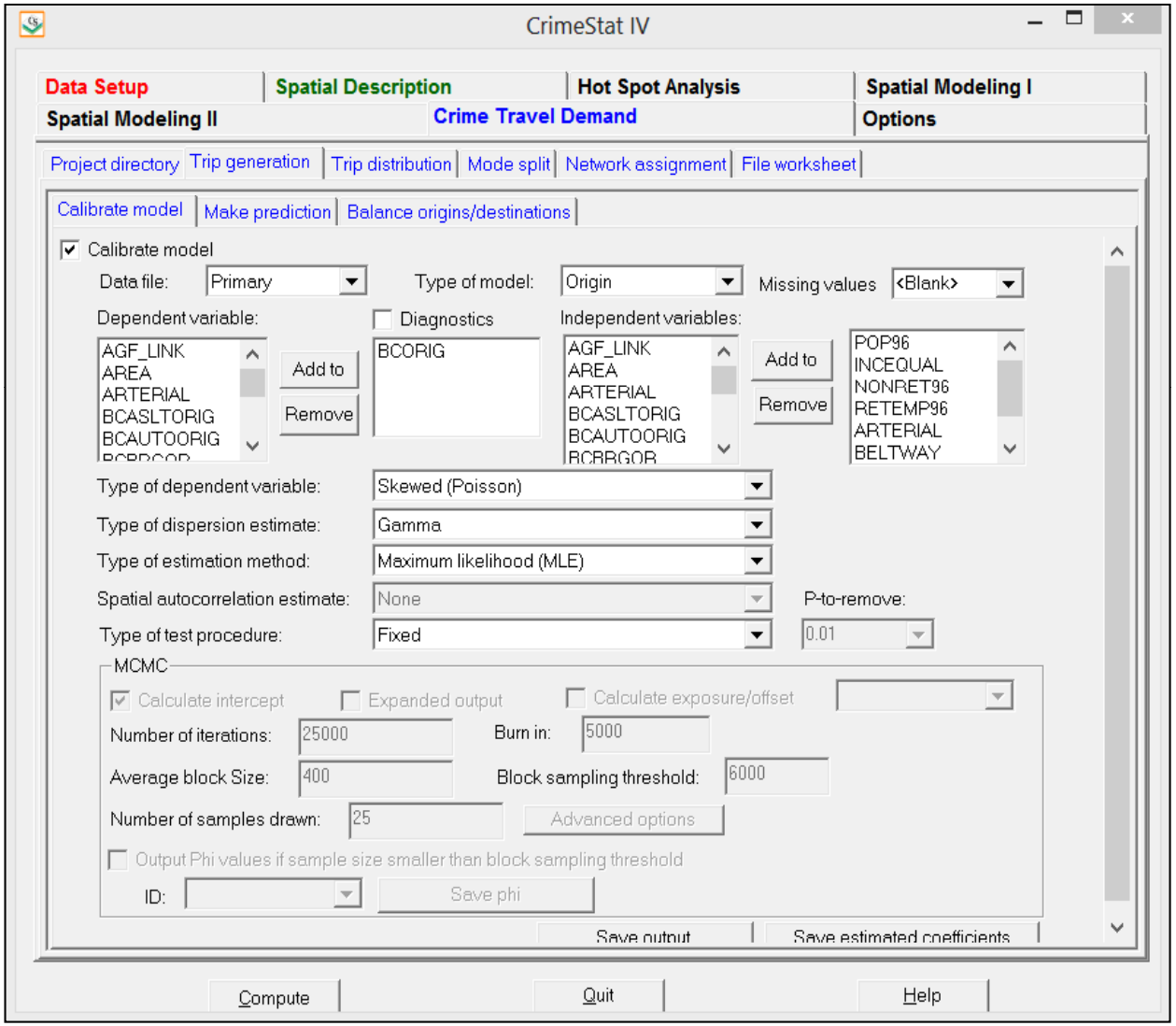
The *CrimeStat* Crime Travel Demand Module

The *CrimeStat* crime travel demand module follows this logic fairly closely, but adapts it to the nature of crime data. Figure 25.4 below shows a screen image of the module. There are five main sections (tabs). Four of them correspond to the four stages. Each of the four sections has several routines associated with them. These will be explained in the subsequent chapters.

In addition, there is a 'File worksheet' section. This allows the user to save the file names in order to keep track of them. The module is complicated and there are a lot of files used - 38 of them, many used multiple times. In addition, there are a variety of parameters that are used for the different files. The result is complex because not only is the model tested sequentially but there are multiple options available for each stage. The subsequent chapters, the file worksheet tab, and the online help menu will try to make the routines easy to understand. But, the user has to realize that it will take time to gather the data and to construct the model.

Figure 25.4:

Crime Travel Demand Module



Crime Travel Demand v. Journey-to-crime

A distinction should be made between crime travel demand and journey-to-crime. Crime travel demand modeling is not journey-to-crime modeling. Journey-to-crime modeling (and its use in geographical profiling) is a much simpler system. Research on journey-to-crime has been conducted since the 1930s (see Chapter 13). For the most part, journey-to-crime modeling is a descriptive framework. Estimates are made of the distance that offenders travel during particular crime trips. A distance decay-type function is estimated from these trips and comparisons are made between different types of crime or the same type of crime for different time periods.

There is very little in the way of theory for this type of model. Crime trips are a function of distance plus some other characteristics, such as the crime type or whether there is or is not a 'buffer zone; around the offender's residence (see Chapter 13). Most of the journey-to-crime studies have compared different types of crime by distance traveled, whether measured as average distance or by type of function as was used in Chapter 13. Almost exclusively, the key variable is travel distance. There are very few studies that have looked at travel time (see Kent, Leitner & Curtis, 2004 for an exception).

In other words, journey-to-crime modeling is a single-stage model, essentially a description, with the primary variable being distance. It is also 'non-adjustable' in the sense that the conditions cannot be varied since there is no model that predicts distance other than crime type (or buffer zone, for which we did not find evidence; see Chapter 13). There is very little in the way of predictions that the model can make other than to estimate the likely origin location of an offender (for events committed by a single offender).

Crime travel demand modeling, on the other hand, is a predictive framework. Crime trips are a function of productions, attractions, and impedance. Productions are a function of some socio-economic and policy variables. Attractions are a function of some other socio-economic and policy variables. Impedance is a function of cost and availability variables. Each of these components is predicted by different variables. Hence, the model can be adjusted (e.g., by adding or subtracting a policy intervention variable). One of its benefits is the ability to adjust conditions. For example, if it can be shown that the amount of policing in a zone impacts the number of crimes that either originate or end in that zone, then a subsequent run can 're-assign' police personnel to impact crimes in other zones.

The model is multi-stage since it is estimated sequentially and, therefore, can be used for prediction. Thus, once the model is estimated on one data set, it can be used on another set. Thus, it represents a calibration against a known data set. For example, one could calibrate the model on one year's worth of data and then use the estimated coefficients and parameters on a second year's worth of data. This, in fact, is how it is used in transportation modeling. The

model is calibrated on a current year and then applied to a future year to make a forecast of future travel demand.

In short, crime travel demand is a theory of travel behavior whereas journey-to-crime modeling is but a simple description. In many ways, crime travel demand modeling is a quantum leap in complexity and analysis, requiring gathering a lot more data and calibrating many individual steps. Nevertheless, that complexity allows a far greater use of the model than the traditional journey-to-crime.

Models v. Description

A key distinction in the crime travel demand framework is that of an empirical description versus a model. The framework can be applied both as an empirical description and as a model, assuming that data can be obtained. An empirical description describes the data that have been collected. For example, for trip generation, it is a count of the number of crimes that originate in each zone and the number of crimes that occur in each zone. For trip distribution, it is the actual number of trips that go from each origin zone to each destination zone. For mode split, assuming that data could be obtained on travel mode, it is a count of the number of trips for each origin-destination pair that are taken by each travel mode. Finally, for network assignment, again assuming that data could be obtained on the actual routes taken, it is a documentation of the actual routes that are taken and a count of the number of trips on each segment of the transportation network. In other words, an empirical description is a count of the number of offenders, whether by origin location, destination location, O-D pair, travel mode, or route.

A model, on the other hand, is a simplified set of relationships that approximate the most important features of the actual count. The model is not reality, but is a rough approximation to it. Because it is rough, a model inevitably makes errors. Consequently, there always will be a difference between a model and the actual events to which the model approximates.

The two differ on other dimensions as well. A model has only a few variables whereas the actual events have many (perhaps hundreds). A model has a simplified set of relationships among the variables whereas the actual events have very complex relationships among the variables, often too complex to describe properly. By simplifying the relationships, a model produces, what Herbert Simon and Allen Newell called, an *analogy* to the actual situation, whereas the actual events are literal (Simon & Newell, 1963; Newell, Shaw & Simon, 1957).

The *CrimeStat* crime travel demand routines can be applied both to empirical data as well as modeled relationships. In fact, two of the routines are directly concerned with the differences between the model and the actual data. Both sets of endeavors have value in their own right, but they differ. An empirical description is most relevant to the present. For a police department

trying to mitigate crime and catch offenders, the empirical description is probably of more use than an abstract model. As will be seen, the empirical description of crime trips will always be more complex than the estimated model. If the only purpose is to describe the actual patterns that are occurring, then a model is not needed.

On the other hand, a model has definite advantages that a description does not. First, it can be used for prediction. If a model is calibrated against a known data set, that model can then be applied to a new data set. For example, one could create an estimate of crime trip productions based on existing socioeconomic and land use data. Then, one could apply that model to a forecast data set of future socioeconomic and land use conditions. The result is a prediction of future crime levels. Of course, since the model was never completely correct in the first place, it will inevitably make errors.[6] Further, since there is no guarantee that past relationships will necessarily hold in the future, there is no certainty about whether the most important part of the predicted relationships will actually hold. Nevertheless, there has been enough success in demographic, economic, and transportation modeling that new fields of forecasting have emerged as legitimate research activities.

A second advantage of a model is that it can be manipulated. Variables can be modified to explore their effect. Distributions can be re-arranged to, again, understand their effect. For example, if relationships can be established between the number of crimes produced or attracted to zones, on the one hand, and the number of police personnel in a zone or to the existence of a large shopping mall, or to the existence of a drug treatment center, on the other hand, then scenarios could be run that explore the different arrangements. These 'What if?' types of scenarios can be very useful. For example, if a relationship exists between shopping malls and crime trips, what is liable to happen when a new shopping mall is built? One could take the model, add the new shopping mall (or the retail employment or acreage associated with the mall) and run the model to make a prediction about its likely impact. Or to take another example, if it can be shown that there is a negative relationship between the number of beat police officers and the number of crimes originating in zones, then it would be possible to evaluate the likely consequences of re-arranging police personnel across different beats.

In short, a model is a very powerful tool for evaluating policy or intervention type strategies. Rather than speculate or gather evidence from other metropolitan areas on their experience (which is valuable, of course), a model can be used to simulate the likely

---

6 Simon and Newell described two kinds of errors: 1) errors of commission (Type I errors); and 2) errors of omission (Type II errors). The first kind represents relationships and predictions that do not exist (to use our terminology) while the second kind represents the failure to detect relationships that do exist. Any model will have both sets of errors. The point to keep in mind is whether a model captures the most important relationships and does not make too many Type I errors (Simon & Newell, 1963; Newell, Shaw, & Simon, 1957).

consequences of an action on crime levels. In transportation planning, the travel demand model is used all the time to evaluate the likely consequences of implementing particular projects. This does not mean that it is the only factor considered in making a decision or even the most important factor; clearly, politics, financing, and community support are also major components of any decision. Nevertheless, the travel model is a very important input into any decisions about future investments.

Uses of a Crime Travel Demand Model

Table 25.1 illustrates some possible uses of the crime travel demand model, assuming that data could be obtained.

Table 25.1:
Possible Uses of Crime Travel Demand Model

	Trip Generation	**Trip Distribution**	**Mode Split**	**Network Assignment**
Description	Identify correlates of crimes	Identify crime trip links	Identify crime travel models	Identify routes taken by offenders
Calibration	Estimate coefficients of predictor variables for crime origins & destinations	Estimate origin-to-destination coefficients for crime trips	Estimate formula for travel modes used by offenders	Estimate model for routes taken by offenders
Prediction	Predict future crime levels	Predict future crime trip links	Predict future crime travel modes	Predict future routes used by offenders

The model could be used for description, calibration, or prediction. In description, the emphasis is on describing the travel behavior of offenders. For trip generation, it involves identifying the correlates of crimes, both by origin zone and by destination zone. For trip distribution, it involves describing the actual crime trips taken between specific origin zones and

specific destination zones. For mode split, it involves identifying the different modes that offenders are using, describing the proportion of each mode that are used, as well as describing the modes used for particular origin-destination links. Finally, network assignment involves describing the actual routes taken by offenders. In other words, the emphasis on description is identifying the specifics used in crime trips.

On the other hand, calibration involves selecting variables that can approximate the description and estimating coefficients for their use. The emphasis is on finding a limited number of general variables and coefficients that can produce a reasonable approximation to the actual travel behavior. Thus, in trip generation, the aim is to find a few variables that can predict reasonably accurately the number of crimes by origin zone and destination zone. In trip distribution, the aim is to estimate coefficients that can approximately describe the trips that are taken from particular origin zones to particular destination zones. In mode split, the aim is to develop coefficients that can approximate the travel modes used while in network assignment, the aim is to find an algorithm that approximates the actual travel routes used by offenders. The result of a calibration is a model that can be generalized whereas a description cannot be generalized.

Finally, in prediction, the calibration models are applied to other data, either forecast values of future levels of the predictive variables or data from other jurisdictions to see the similarities or differences. The existence of a model (ideally calibrated against a real data set) allows a forecast to be made whereas a description cannot be forecast.

Research Uses of a Crime Travel Demand Model

For research, a crime travel demand model has many different uses, only some of which are explored in the next five chapters. First, it organizes crime travel information in a systematic manner. The model is logical and proceeds in a systematic way. As opposed to a journey-to-crime-type model, which is just a description, the crime travel demand model systematically steps through the four stages in an understandable way. It is a very good way to organize information on crime travel, though, clearly, it is not the only way.

Second, compared to the journey-to-crime literature, it is a more realistic model of offender travel. For one thing, it incorporates information about origin locations. This helps answer the question of why certain areas produce more crimes than others (remember, it is not a behavioral explanation, but an ecological model). For another thing, it incorporates information about destination locations and helps answer the question of why certain areas attract more crimes than others. For a third thing, it models travel choice in a more complex manner. Instead of assuming that all offenders will travel to a crime in exactly the same way (e.g., by walking), the model allows the separation of different travel modes. For journey-to-crime models, distance

is the only impedance variable, whereas for crime travel demand modeling, travel time and travel cost are often better predictors of travel behavior, especially in relationship to an available network. In short, it is a much more complex, yet realistic, representation of crime travel behavior.

Third, it is a dynamic analysis of travel behavior. Crime trips are seen as a product of neighborhood production factors, attractions, and travel costs (impedance). Since these change by various hours of the day, so too do the travel patterns change. The ability to model travel at different times of the day is one of the strengths of the travel demand type of framework.

Fourth, and finally, a crime travel demand model can allow comparisons between different types of crimes in the productions, the attractions, and the costs. So, too, can journey-to-crime models be used to compare different type of crimes. But those comparisons are uni-dimensional, essentially comparing different distance decay functions. The crime travel demand model can explain the 'distance decay' function and hence allow a more structural interpretation than was previously possible. For example, in comparing data sets from Baltimore, Chicago, and Las Vegas, Richard Block, Dan Helms and myself are finding that there may be very little difference in the cost function used for different types of crime trips, but that differences in these trips are more a function of the distribution of opportunities (attractions). To link this up to the early theme of this chapter, American society has become so mobile and the automobile so ubiquitous that distances are not as much a barrier to offenders as they used to be. In other words, the distribution of opportunities appears to be the more dominant factor predicting types of crimes than the limitations of neighborhoods and small communities. If this turns out to be true, then we are in for a major shift in the type of crimes that our society will experience over the next few decades. Mobility may replace neighborhood as a determining factor in crime behavior. In other words, the local 'community-based' offender is morphing into a metropolitan-wide and, perhaps, regional offender, a not very desirable prospect. The Kneebone and Raphael (2011) study may indicate that this has already started to occur.

Crime travel demand modeling allows for a more complex, more interventionist and, perhaps, deeper understanding of crime travel than previous types of model, particularly the journey-to-crime and serial walk type of model (see Chapter 13).

Utility for Policing and Law Enforcement

For police department and other law enforcement agencies, crime travel demand modeling has some advantages as well. First, it can be used to model different policing strategies, as suggested above. For example, it could be used to evaluate the likely effect of shifting patrol deployment. The 'What if?' nature of crime travel demand modeling makes it useful to explore alternative arrangements before they are actually implemented.

Second, it could be used for forecasting. As mentioned, if a model has been calibrated on one set of data, then it could be applied to another set to predict, for example, the distribution of crimes five or ten years later. Typically, police departments have not done forecasting, but they are often expected to be able to anticipate changes. This type of model can be useful for that purpose since Councils of Governments (COG) and Metropolitan Planning Organizations (MPO) systematically make forecasts of future population and employment levels.

Third, it can be used for modeling interventions. Aside from modeling different policing strategies, a range of land use and communities changes could be explored. For example, what would be the effect of introducing more drug treatment centers or more 'weed and seed' adolescent facilities? The logic is similar to forecasting. A model is calibrated against one data set. But, in addition to socioeconomic and land use variables, variables on facilities are added to the equation as predictors. If it can be shown that they have any effect (which we hope they do), then these can be used as variables in a modeling scenario.

Fourth, these types of models can be used for anticipating changes in the community. Again, this is similar to the forecasting purpose mentioned above. But, it is slightly different in that it anticipates structural changes. An example was given of anticipating changes from new shopping malls. In Baltimore County, for example, shopping malls were shown to be the strongest attractors of crime trips. In that context, what would happen if a new mall was built? This type of model can be used to model this scenario. Conversely, a lack of employment opportunities appears to be correlated with crime productions, at least in Baltimore County. What would happen to crime if local employment was increased in certain zones? Again, this type of model is useful for exploring that type of question.

Again, going back to an earlier point, there is, of course, a difference between a model and reality (an actual situation). Reality is complex; models are not, or are a lot simpler. Still, models as analogies can provide insight into mechanisms and allow police, law enforcement, and the policy community as a whole to try to simulate changes without having to commit to expensive, and perhaps disastrous, changes with little information. In other words, modeling in general, and crime travel demand modeling in particular, is a tool that may have wide utility for the law enforcement community.

References on Travel Demand Modeling

In this final section, some sources on travel demand forecasting are listed. There are a large number of sources, though there are few actual textbooks. A very good textbook on the subject is by Ortuzar and Willumsen (2001), while an older, out of print book is by Stopher and Meyburg (1975). There are several major handbooks on the topic (Hensher & Button, 2003;

ITE, 2003). Some good chapters on the subject are found in Beimborn (1995), Field and MacGregor (1987, Ch. 6) and by Engelen (1986, Ch 17). Discussions of 'second' generation models can be found in Domencich and McFadden (1975) and Ben-Akiva and Lerman (1985).

However, probably the best source for articles on the subject are found on the Federal Highway Administration (http://www.fhwa.dot.gov) and other web sites. Among the articles/presentations that can be found on that site are FHWA (2012), FHWA (2009), Jeannotte, Sallman, Margiotta, and Howard (2009), and McKeever and Griesenbeck (2009). Of particular interest is a study of bicycle and pedestrian travel modeling (Turner, Shunk, and Hottenstein, 1998), which may be relevant for crime analysis, and 'third' generation models.

Older sources, which are still good are by Oppenheim (1975, ch. 4) and Krueckeberg and Silvers (1974, Ch. 10), aside from the Stopher and Meyburg text mentioned above.

References

Aizcorbe, A. & Starr-McCluer, M. (1996). Vehicle Ownership, Vehicle Acquisitions, and the Growth of Auto Leasing: Evidence from Consumer Surveys. Finance and Economic Discussion Series, Federal Reserve Board of Governors: Washington, DC. http://www.federalreserve.gov/pubs/feds/1996/199635/199635pap.pdf. Accessed April 28, 2012.

Ben-Akiva, M. & Lerman, S. (1985). *Discrete Choice Analysis: Theory and Application to Travel Demand*. MIT Press: Cambridge.

BTS (2007). Table 2: A matrix of transportation expenditure by source of finance and type of expenditure, *Government Transportation Financial Statistics*, Bureau of Transportation Statistics, U.S. Department of Transportation: Washington, DC., http://www.bts.gov/publications/government_transportation_financial_statistics/2007/html/table_02.html. Accessed April 28, 2012.

BTS (2003). U.S. Vehicle-miles (millions), Table 1-32. *National Transportation Statistics 2004*, Bureau of Transportation Statistics, U.S. Department of Transportation: Washington, DC. http://www.bts.gov/publications/national_transportation_statistics/2004/html/table_01_32.html. Accessed April 28, 2012.

Burgess, E. W. (1925). The growth of the city: an introduction to a research project. In Park, R.E., Burgess, E. W. & Mackensie, R. D. (ed), *The City*. University of Chicago Press: Chicago, 47-62.

Culp, M. & Lee, E. J. (2005). Improving travel models through peer review. *Public Roads*, 68 (6), FHWA-HRT-05-005. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/publications/publicroads/05may/07.cfm. Accessed April 28, 2012.

Domencich, T. & McFadden, D. (1975). *Urban Travel Demand: A Behavioral Analysis*. North Holland Publishing Company: Amsterdam & Oxford (republished in 1996). Also found at http://emlab.berkeley.edu/users/mcfadden/travel.html. Accessed April 28, 2012.

Engelen, R. E. (1986). Transportation planning. In So, F. S. *The Practice of State and Regional Planning*. American Planning Association: Chicago, Ch. 17, 431-453.

FHWA (2012). FHWA Resource Center Planning Team. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/resourcecenter/teams/planning/travel.cfm. Accessed April 29, 2012.

References (continued)

FHWA (2009). Integrated Urban Systems Modeling, *The Exploratory Advanced Research Program Fact Sheet*, FHWA-HRT-09-042. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/advancedresearch/pubs/interurbsys.pdf. Accessed April 28, 2012.

FHWA (1996). Latest VMT growth estimates, *Highway Information Update*, 1(1), Federal Highway Administration, U.S. Department of Transportation: Washington, DC., http://www.fhwa.dot.gov//ohim/vol1no1.html. Accessed April 28, 2012.

Field, B. & MacGregor, B. (1987). *Forecasting Techniques for Urban and Regional Planning*. UCL Press, Ltd: London.

Freedman, David A. (1999). Ecological inference and ecological fallacy. *International Encyclopedia of the Social and Behavioral Sciences*, Technical Report No. 549, October. http://www.stanford.edu/class/ed260/freedman549.pdf. Accessed March 26, 2012.

Hensher, D. A. & Button, K. J. (2002). *Handbook of Transport Modeling*. Elsevier Science: Cambridge, UK.

Horowitz, J. L., Koppelman, F. S. & Lerman, S. R. (1986). *A Self-instructing Course in Disaggregate Mode Choice Modeling*. Federal Transit Administration, U.S. Department of Transportation: Washington, DC. http://ntl.bts.gov/DOCS/381SIC.html. Accessed April 28, 2012.

ITE (2003). *Trip Generation* (7th edition). Institute of Transportation Engineers: Washington, DC.

Jeannotte, K., Sallman, D., Margiotta, R. & Howard, M. (2009). *Applying Analysis Tools in Planning for Operations*. FHWA-HOP-10-001, Federal Highway Administration: Washington, DC. http://ops.fhwa.dot.gov/publications/fhwahop10001/fhwahop10001.pdf. Accessed April 29, 2012.

Kent, J., Leitner, M., & Curtis, A. (2006). Evaluating the usefulness of functional distance measures when calibrating journey-to-crime distance decay algorithms. *Computers, Environment and Urban Systems, 30* (2), 181-200.

Kneebone, E. & Raphael, S. (2011). *City and Suburban Crime Trends in Metropolitan America*. Metropolitan Opportunity Series, Metropolitan Policy Program, Brookings Institution: Washington, DC. http://www.brookings.edu/papers/2011/0526_metropolitan_crime_kneebone_raphael.aspx. April 28, 2012.

References (continued)

Krueckeberg, D. A. & Silvers, A. L. (1974). *Urban Planning Analysis: Methods and Models*. John Wiley & Sons: New York.

Langbein, L. I. & Lichtman, A. J. (1978). *Ecological Inference*. Sage University Paper series on Quantitative Applications in the Social Sciences, series no. 07-010. Beverly Hills and London: Sage Publications.

McFadden, D. L. (2002). The path to discrete-choice models. *Access*, No. 20, Spring. 20-25. http://www.uctc.net/access/access20.shtml. Accessed April 28, 2012.

McKeever, M. & Griesenbeck, B. (2009). *Linking Transportation and Land Use*. Federal Highway Administration: Washington, DC. http://www.fhwa.dot.gov/policy/otps/innovation/issue1/linking.htm. Accessed April 29, 2012.

NAP (2009). Driving and the Built Environment: The Effects of Compact Development on Motorized Travel, Energy Use, and CO2 Emissions, Special Report 298. The National Academies Press: Washington, DC. http://www.nap.edu/catalog.php?record_id=12747. Accessed April 28, 2012.

Newell, A., Shaw, J. C. & Simon, H. A. (1957). Empirical Explorations of the Logic Theory Machine, Proceedings of the Western Joint Computer Conference, pp. 218-239.

NHMC (2012). Quick Facts: Resident Demographics. National Multi Housing Council: Washington, DC. http://www.nmhc.org/Content.cfm?ItemNumber=55508. Accessed April 28, 2012.

OMB (2012). 2010 Standards for Delineating Metropolitan and Micropolitan Statistical Areas. U.S. Office of Management and Budget, *Federal Register*, June 28, 2010, 37246. http://www.whitehouse.gov/sites/default/files/omb/assets/fedreg_2010/06282010_metro_standards-Complete.pdf. Accessed April 28, 2012.

Oppenheim, N. (1980). *Applied Models in Urban and Regional Analysis*. Prentice-Hall, Inc.: Englewood Cliffs, NJ.

Ortuzar, J. D. & Willumsen, L. G. (2001). *Modeling Transport* (3rd edition). J. Wiley & Sons: New York.

Schachter, J. (2001). Geographical mobility: March 1999 to March 2000. *Current Population Reports*, P20-538, March. U.S. Census Bureau: Hyattsville, MD.

Simon, H. A. & Newell, A. (1963). The uses and limitations of models. In Marx, M. (ed), *Theories of Contemporary Psychology*, Macmillan: New York, 89-104.

References (continued)

Stopher, P. R. & Meyburg, A. H. (1975). *Urban Transportation Modeling and Planning.* Lexington, MA: Lexington Books.

Thrasher, F. M. (1927). *The Gang,* University of Chicago Press: Chicago.

Turner, S., Shunk, G. & Hottenstein, A. M. (1998). *Development of a Methodology to Estimate Bicycle and Pedestrian Travel Demand*. Report 1723-S, Texas Transportation Institute: College Station. http://tti.tamu.edu/publications/catalog/record/?id=146. Accessed April 28, 2012.

U.S. Census (2010a). Geographic Mobility/Migration. U.S. Census Bureau, U.S. Department of Commerce: Washington, DC. http://www.census.gov/hhes/migration/data/cps/cps2010.html. Accessed April 28, 2012.

U.S. Census (2003). Net Worth and Asset Ownership of Households: 1998 and 2003 (Table A). *Current Population Reports*, P70-88. U. S. Census Bureau, U. S. Department of Commerce: Washington, DC. http://www.census.gov/prod/2003pubs/p70-88.pdf. Accessed April 28, 2012.

USDOT (2003). *Title XXIII, Part 450*. Code of Federal Regulations. Code of Federal Regulations, Title 23, Part 450, Volume 1. 23CFR450. Washington, DC.

Wikipedia (2012a). List of Cities with Most Households without a Car. Wikipedia. http://en.wikipedia.org/wiki/List_of_U.S._cities_with_most_households_without_a_car. Accessed April 28, 2012.

Wikipedia (2012b). Table of United States Metropolitan Statistical Areas. Wikipedia. http://en.wikipedia.org/wiki/Table_of_United_States_Metropolitan_Statistical_Areas#cite_note-MSA-0. Accessed April 28, 2012.

Chapter 26:

Data Preparation for Crime Travel Demand Modeling

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 26:

Data Preparation for Crime Travel Demand Modeling

In this chapter, the data requirements for the crime travel demand model are discussed. At the minimum, there are four types of data that are needed for the crime travel demand module:

1. A zonal system;
2. Matched crime data listing both crime location and likely origin location. This can be, further, broken down by crime types, time of day, day of week, and other sub-sets of the total number of crimes;
3. Socioeconomic and land use data for the zones which are used as predictor variables; and
4. Network data on the road system and the transit system.

In addition, there can be supplementary data that help expand the predictive models. These include:

5. Policy-related data (e.g., strategic or planned interventions)
6. Crime data on the actual distribution of crimes by zones, which is used to correct the implied distribution from 2 above.

The following is a discussion of each of these requirements.

Choice of a Zonal System

The crime travel demand model is a zonal model. That is, it analyzes crime trips by zones. For all four stages, the estimates are for zones, not for individuals. Thus, at the trip generation stage, there are two zonal models - one predicting the number of crimes originating in each origin zone and one predicting the number of crime ending in each destination zone. At the trip distribution stage, there is a prediction of the number of crimes which originate in each origin zone that end up in each destination zone (the implicit number of *trips*). At the mode split stage, the trips for each origin-destination zone pair are, further, sub-divided into different travel modes. Finally, each origin-destination zone pair by travel mode is assigned a route. But, at all stages, the estimates are for zones.

Typical Zone Systems

This makes the choice of a zonal system very critical. In practice, three types of zone system have been used:

1. Census geography
2. Traffic analysis zones
3. Grid cells

Census geography follows the geography used by the U.S. Census Bureau (in the United States) or by other national census agencies. Traffic analysis zones are used by most transportation planning agencies for modeling transportation in a metropolitan area. They are typically super-sets of census geography (e.g., two census tracts combined). Finally, grid cells are uniform zones imposed on a metropolitan area. While they have desirable statistical properties, they are rarely used in practice.

Problems with Large Zones

In deciding on a choice of a zonal system, there are several important issues that must be balanced. The first problem one faces is that of zone size. Large zones can distort relationships. It can be shown that the size of a zone has an impact on the statistical relationships between the predictor variables and the dependent variables, which are the number of crime trips by either origin or destination zone. Typically, the larger the zone size, the stronger the relationship. The reason for this effect is complex and has to do with a number of factors, for example minimizing within-zone differences in travel behavior and, therefore, maximizing the between-zone variance relative to the within-zone variance (Langbein & Lichtman, 1978) or aggregating spatial autocorrelation to minimize adjacency effects (Anselin, 1995). But, the effect is well known. The cost of having this stronger statistical relationship is to produce a less precise estimate for the region since within-zone differences are minimized.

One can think of this in terms of an arbitrary point within a zone (e.g., the centroid of the zone though it could be any location within the zone that is taken as the focal point for estimation). All the data in the zone are assigned to that point. Thus, the number of crimes that originate within the zone or end within the zone are assigned to a single point. This means that whether a crime occurred at the edge of the zone or directly in the middle, it is assigned geographically to a single point. Similarly, any of the predictive socioeconomic or land use variables are also assigned to that point (e.g., median household income). Hence, any spatial differences within the zone are eliminated as all events and households are assumed to ‘live’ at that point. If there are two adjacent zones, for example, that differ in income levels, most likely there is a gradient of income from one to the other; however, putting the measurement of income

at a single point in each zone exacerbates the differences between the zones while ignoring the similarities (e.g., at the edges of the zones where the population on both sides are liable to be more similar). It should be clear that the larger the zone size, the greater the exaggeration between the zones. In other words, larger zones exacerbate differences between zones while minimizing similarities. The result is an oversimplification of the distribution of characteristics of those neighborhoods.

In addition, larger zones have too many trips that both originate and end in the same zone (intra-zonal or 'local' trips). Clearly, the larger the average size of a zone, the more likely that a trip will be entirely within the zone. Thus, there is a strong relationship between average zone size and the number of intra-zonal trips. This will be less useful since it minimizes the complexity of travel. The extreme would be to divide a metropolitan area into only a few zones (e.g., 4 or 5). The result would detect large scale travel patterns, but would lead to a majority of trips occurring within each zone. One would not be able to say very much about crime travel other than a few general patterns (e.g., crime trips from the central city to the suburbs).

On the other hand, if the zones are too small, there is a danger that there would be more cells in the trip distribution stage (see Chapter 28) than there are actual events. The result would be inadequate degrees of freedom in a model and unreliable coefficients. A zone model has to balance the need for increased precision with the ability to produce stable estimates.

Problems in Obtaining Data for Small Zones

In theory, the ideal zone size would be small, say on the order of a block or two. This would allow precision in estimates and the ability to examine the complexity of travel in a metropolitan area. The reason that this is not done very often, however, is the lack of data at the block or block group level. While crime data can be allocated to blocks or block groups, it is often difficult to obtain socioeconomic data at that level. In the United States, for example, while the U.S. Census Bureau will release data down to the block level, confidentiality requirements require that no data be able to identify individuals. Hence, there is very limited data at the block level, typically gender and race distribution. Block group data, on the other hand, is often easily available, including critical income factors.

The biggest problem with a block group zonal system is in obtaining employment data. The U.S. Census Bureau only collects a sample of employment data from the decennial census which they release in their Journey-to-work data set (U.S. Census Bureau, 2012). They release this data for fairly small geographical units (e.g., block groups) and also produce yearly estimates for larger geographical units. These data can be used to construct employment estimates for small geographical areas; however, it is current only in those years close to the

census year and becomes quickly outdated. The Bureau of Labor Statistics also collects employment information, but will not release it at such a small geography.

Thus, obtaining these data depends on local organizations, such as a Council of Government (COG) or a Metropolitan Planning Organization (MPO). Till now, these data have not typically been released at small geographies such as block groups, but, instead, at a larger geographical unit called a *traffic analysis zone* (TAZ). However, because of the widespread use of GIS and the increasing incorporation of high resolution aerial photography into GIS-based land information systems, this situation is changing. For example, at the Houston-Galveston Area Council, the MPO for the greater Houston area, employment estimates are made for as small a geography as a 1000 foot by 1000 foot grid cell, essentially a couple of city blocks. Thus, it is starting to become possible to obtain employment data at very small geographical levels. In the next few years, more and more data will be available for small geographical units and the size limitation mentioned above will slowly disappear.

There is a converse problem with size, however, that also occurs. If the zones are too small (e.g., if data could be obtained at a block face level), there will be too many cells with no crime events. The smaller the geographical unit, the more likely that there will be no events recorded. For example, to illustrate the crime travel demand model, I have used data from Baltimore County. The crime data were 41,974 incidents that occurred between 1993 and 1997 for which both a crime location and a crime origin were known. To model these incidents, traffic analysis zones (TAZ) were used. For Baltimore County, there were 325 destination TAZ's while for both Baltimore County and Baltimore City, there were 532 origin TAZ's. Taking the origin TAZ's, with 41,974 incidents the average number per TAZ was 78.9. However, in practice, 27 zones had no crimes originate from them (or approximately 5%). If a smaller geography was used (e.g., block groups), the number of zones with no crime originating in them would increase substantially, as would the percentage. At some point, if the geography becomes very small, a high proportion of the zones will have no crimes originating from them. This makes modeling very difficult as the average number of events will tend towards zero. While there are techniques for modeling a skewed distribution (which will be discussed in Chapter 27), the more skewed the distribution, the less accurate typically is the estimate. Extremely skewed distributions are more problematic for modeling than mildly skewed distributions as the variance terms become very complex to estimate (see Chapter 16).

Still, on average, a small zone system is preferable to a large one. There is so little data for very small geographies that the problem of zones being too small is an unlikely one, at least for the foreseeable future. Where possible, users should try to obtain data at the smallest geographical level for which data can be obtained.

Problems with Irregular Size and Shape

Another problem facing the choice of a zonal system is the irregular sizes and shapes of most zonal data. For example, the U.S. Census Bureau uses a unit called the *census tract* for the collection of census information. The census tract is supposed to be an area of approximately equal population (though it is rarely entirely equal). These units generally are wholly within jurisdictions (though there are exceptions) and they are made up of blocks and block groups (collections of blocks), but in turn are aggregated upward to form enumeration areas within each jurisdiction. This logic makes sense in terms of the mission of the U.S. Census Bureau, which is to take the census. The geography respects political jurisdictions (counties and cities), but is fine enough to help manage the data that is collected during the decennial census.

But, from a modeling viewpoint, this geography has problems. First, the area of census tracts typically increase from the central city outward to the far suburban edges of a metropolitan area. Because the logic of the census tract is to approximate an area of equal population, by necessity the tract area will increase with the lower densities in most suburban communities. Thus, any data assigned to a tract (or to a block or block group within a tract) will be less precise in the suburbs than in the central city. In a travel demand model, one can end up with absurdities whereby trips appear to originate at locations where there are no people simply because the centroid of the zone falls at a location where there are no households (e.g., in a reservoir). The uneven size of zones usually means that a travel model will be more precise in the center of a metropolitan area than in a suburb.

Second, because census tracts are often defined with respect to principal arterial roads (which form their edge), they often will have irregular shapes. This could add a potential source of error in that all events and household characteristics within a boundary are assigned to a single point in the zone. On the other hand, if the zones have been selected to represent a neighborhood which is relatively uniform, such irregularity may not be a problem. Nevertheless, if two zones have very different shapes (e.g., one is square while the other is pointed), allocation error (and, hence, modeling error) is liable to be greater in the one that is more irregular, all other things being equal, than in the one that is square. This is the so-called Modifiable Area Unit Problem (MAUP)(Wikipedia, 2012; Hipp, 2007; Wooldridge, 2002; Openshaw, 1984).

Again, ideally, a zone system should be a grid whereby each zone is a square of equal size; shape and area effects are constant for all zones. While geographers recognize the value of a grid cell for zonal allocation, in practice, it is rarely used. Among the transportation planning agencies in the country, very few use a grid system. Of the ones with which I am familiar, only

the Chicago Area Tranportation Survey (CATS) uses a grid system.[1] In Chapters 31 and 32, Richard Block and Dan Helms discuss applying the crime travel demand model to Chicago.

Therefore, to sum up, in practice, one has to balance four different criteria in selecting a zone system for a crime travel demand model:

1. Zone size (generally, smaller is better within limits)
2. Consistency of zone size (less variability is better)
3. Distortion due to shape (more regular is better)
4. Availability of data

Unfortunately, it is the fourth criterion - the availability of data that is usually the determining factor in the choice of a modeling zonal system. Hopefully, this will change in the future as more data at the smaller geographical level become available.

Trips from Outside the Study Area

One other problem confronts the choice of a zone system. Irrespective of which zone system is used (census geography, TAZ, grid cells), a decision has to be made about the extent of the area to be used in modeling. The choice of destination zones is made by the availability of crime data. Typically, data are collected by police departments for their jurisdiction. Unless data sets from several adjacent jurisdictions can be obtained and combined, the analyst typically will be restricted to modeling the jurisdiction for which the crime data has been collected. This is called the *Modeled Jurisdiction*.

Modeling the origin zones is a decision about which zones contribute to the crimes occurring in the modeled jurisdiction. That is, some of the origins of the crime trips occurring within the modeled jurisdictions may come from outside that jurisdiction. For example, in the case of Baltimore County, approximately 42% of the crimes occurring within that jurisdiction were committed by offenders who lived outside that jurisdiction, of which 38% originated from the City of Baltimore.

In such a case, it is very important to include zones beyond the modeled jurisdiction in the crime origin model. That is, to use Baltimore County as an example, if the predictive model for crime origins only included the 325 TAZ's within that jurisdiction, the model would not adequately assess the factors predicting crime origins.

---

[1] CATS was used as the prototype by the Federal Highway Administration for developing the original travel demand model. The grid was used because it minimized errors due to irregular size and shape. Nevertheless, that model has not been followed by planning agencies in the U.S.

But where does one draw the line? Eventually, because of limitations due to data or due to the need to restrict the analysis, a boundary has to be drawn around the study region. Some crimes will inevitably originate from outside that line. These are called *External Trips* and refer to the trips that originate from outside the study area. While there is no 'hard and fast' principle, generally transportation planners recommend that the study area include at least 95% of the trips that end in the modeled jurisdiction (Ortuzar & Willumsen, 2001). With such coverage, the 5% (or less) that are external trips will have little effect on the model parameters, and the amount of bias will be small (but will always exist unless 100% of the trips can be measured).

I will come back to this point in the next chapter. But, the critical point is that the zone system must incorporate a sizeable area in which at least 95% of the crimes originate from within. Going back to the Baltimore County example, adding in the City of Baltimore increased the percentage of trips originating within the study area from 58% (for just Baltimore County) to 96%, an acceptable level to 'draw a boundary' around the study region.

Small Area Limitations

A travel demand model is aimed at modeling travel patterns in a metropolitan-wide area. The model is particularly good at estimating travel for the region as a whole and for large sub-areas of the region. The model is not particularly good at estimating travel within small geographical areas. The problem of intra-zonal trips - trips in which the origin and the destination occur in the same zone, represent trips for which the model cannot describe the travel pattern. These are trips that the model detects are within a small area, but cannot estimate where these occur. Similarly, trips between adjacent zones are often imprecise in a travel demand model; the model can indicate the level of short trips, but the level of precision is low.

In other words, the crime travel demand model is good at capturing major travel patterns over a large area and not very good at localized travel. There are other modeling tools for small area travel analysis that provide much more detail about the neighborhoods and road system in which this travel occurs, such as microsimulation software of travel behavior in a neighborhood (Kitamura, Yoshii, & Yamamoto, 2009; Miller & Salvini, 1999).

Therefore, in order to apply a travel demand model to crime analysis, it is important to model a substantial part of a metropolitan area. The model will not be as accurate if a small city or area within a metropolitan area is chosen. In these chapters, crime travel in Baltimore County is used as an example case in order to illustrate the different components of the model. Baltimore County is a large jurisdiction covering approximately 640 square miles; it represents a sizeable part of the Baltimore metropolitan area. Combining the origin zones of Baltimore City with those of Baltimore County provides a very large proportion of the metropolitan area. In

other words, Baltimore County is large enough to model the crime destinations while the origin zones represent much of the metropolitan area.

On the other hand, if we attempted to apply the model to a small part of the region, for example the town of Towson, the model would be less precise and less accurate since that town represents a very small proportion of the overall region. In short, a crime travel demand model is useful for modeling either an entire metropolitan region or a sizeable part of a metropolitan region, but should not be considered for a small geographical area. It is a regional travel model, not a local model.

Calculation Limits for the Number of Zones

A final consideration has to do with the number of zones that can be modeled with the *CrimeStat* crime travel demand model. Depending on whether a computer is 64 bits or 32 bits and depending on the operating system, limits may be reached on the number of zones. For example, with a Windows 32 bit operating system, the routine can only access 4 Gb of RAM. If M is the number of origin zones and N is the number of destination zones, then a trip distribution matrix, which is subsequently used in the mode split and network assignment stages, involves N*M cells. Each digit requires 64 bits of RAM with 16 digits assigned per cell. There are also seven fields output. Thus, a trip distribution output file requires approximately M*N*64*16 bits of RAM.

To use an example, if the user has 1 Gb of RAM available, then approximately 8,388,608 grid cells could be handled (or a square matrix of 2,896 x 2,896). However, Windows requires some overhead as does *CrimeStat*. Thus, the actual number of grid cells that could be processed will be a little less.

One could, of course, add more RAM. In this case, the file size of the trip distribution matrix could be increased. However, there are limits to this. First, the calculations will slow down, at a rate that is exponential to the file size. At some point, the calculations will take so long as to be impractical.

Second, as mentioned a 32 bit operating system a 4 Gb limit. Thus, the maximum file size would be a square matrix of about 5,793 x 5,793. A 64 bit operating system, on the other hand, can access 32 Gb of RAM thus allowing about 268 million cells (or a square matrix of 16,384). Clearly, if the study area has many zones, then a 64 bit computer and operating system will be essential. But, even here there are limits. For example, in Chicago there are 21,068 blocks. Using these blocks as a zone model in the crime travel demand would be impossible even for a 64 bit computer since the matrix routines could not handle such a large matrix, even assuming that it is desirable to do so. Therefore, any zonal model that is selected must be

compatible with the calculation limits of the available RAM and the Windows operating system. In the case of Chicago, using block groups was an acceptable choice since there are only 2400.

Obtaining Crime Data

There are four types of data that need to be obtained.

Crime Data by Origins and Destinations

First, there is crime data. But, in order to estimate a crime trip, it is essential that these data have information on both crime origins as well as crime destinations. The most likely source of these data will be arrest records whereby both the crime location and the charged offender's residence are given. Only the police are liable to have these data. Thus, it will be necessary to obtain cooperation from the local police department for access to arrest records.

In the data, the residence location is taken as the origin while the crime location is taken as the destination of the trip. As mentioned in Chapter 13 on journey-to-crime, the "true" origin of the crime may not be known. First, the offender may not even have been living at the same residence as when arrested. Many offenders are highly transitory persons and a residence at the time of the arrest may not be the actual one from which the crime occurred. Second, the offender may not have traveled directly from home to the crime location, but may have committed the crime as part of his/her daily activities (intermediate trips). However, without any alternative data on the actual origins, there is little that can be done except assume that the residence when arrested is the origin. As long as this definition is kept, a consistent estimate can be obtained.[2]

In effect, one is asking the question, "What is the likelihood that an offender who lives in zone i will commit a crime in zone j at some point during a day?" It really does not matter whether the offender traveled from the home location to the crime location as opposed to going to the crime location from an intermediate location. The model is simply constructed with respect to residence location.

The data has to be organized so that the X and Y coordinates of both the residence location (the origin) and the crime location (the destination) are given. Figure 26.1 illustrates a

---

[2] If the actual origin was an intermediate location between the home and the crime location, then with a large sample of crimes and offenders the idiosyncrasies of one offender's crime travel pattern is not going to affect the coefficients of the prediction model to any great extent. If *all* offenders from a particular zone committed crimes from an intermediate location which was always the same, then that condition might affect the coefficients (assuming one could obtain the data). But, it is highly unlikely that all offenders will commit crimes in the same destination zone using the same intermediate zone as an origin.

typical data set. It will be necessary to geocode both locations in order to establish a 'crime trip', an assumed trip from a particular origin location to a particular destination location.

Figure 26.2 shows the location of 41,974 crimes committed in Baltimore County between 1993 and 1997 while Figure 26.3 shows the assumed origin location of the offenders who committed these 41,974 crimes. As seen, the origins are all over the region, but most (96%) are in either Baltimore County or Baltimore City. In other words, a 'crime trip' links the origin location of each crime with the actual destination where it occurred. If arrows were to be drawn from the origin to the destination, the entire map would be swamped with a series of lines.

Choosing a Zonal Model

The zonal framework used for the Baltimore County analysis was traffic analysis zones (TAZ). The reason for selecting theis was the availability of both population and employment data. The Baltimore Metropolitan Council is the Council of Governments and the Metropolitan Planning Organization for the greater Baltimore region. They use TAZ's for their transportation model. Since data were available by the TAZ's, it seemed like a plausible decision. But, as mentioned above, there are advantages and disadvantages to this decision. Approximately, 20% of all crime trips occur within the same zone (intra-zonal trips). Such a high proportion makes the overall model estimates prone to some error. Figure 26.4 shows the TAZ's for both Baltimore City and Baltimore County.

Note that there is a difference between the zones used for the origins and the zones used for the destinations. In the case of Baltimore County, there are 325 TAZ's that cover the County. However, as mentioned above, since many of the crimes occurring in Baltimore County originate in the City of Baltimore, the origin zones include those of the City as well as the County. Thus, there are 532 origin zones.

Assigning Crime Events to Zones

The next step involves assigning the crime origins and the crime destinations separately to the zonal model. That is, each crime event is assigned to zones twice, once for the origins and once for the destinations. Since an arrest record is an implicit crime trip, the residence location is assigned to a zone and the destination location is assigned to a zone. Then, the number of crimes originating from each zone are calculated by summing over all records to produce a distribution of crimes by origin zone. Similarly, the number of crimes ending in each zone are calculated by summing over all records to produce a distribution of crimes by destination zone. The result is two distributions of crimes by zone, one for origins and one for destinations.

Figure 26.1:

Crime Data Requirements

Minimum data requires origin and destination location

UCR	DATE	INCIDX	INCIDY	HOMEX	HOMEY
430	1/5/97	-76.8131	39.3822	-76.8131	39.3822
440	5/17/95	-76.4490	39.3355	-76.4489	39.3355
210		-76.4068	39.3388	-76.5281	39.3085
210		-76.4142	39.2801	-76.4142	39.2801
430		-76.5527	39.3908	-76.4410	39.3080
440		-76.7581	39.3131	-76.7709	39.3105
440	3/29/94	-76.5095	39.2735	-76.5095	39.2735
440	1/22/96	-76.7344	39.3212	-76.6899	39.3364
690	7/13/93	-76.4525	39.3012	-76.6050	39.3020
690	10/8/94	-76.5278	39.2584	-76.5051	39.3970
690	8/10/97	-76.7384	39.3275	-76.7384	39.3275
690	3/10/96	-76.7325	39.3018	-76.7325	39.3018

Figure 26.2:

Baltimore County Crime Locations: 1993-1997
Location of Crimes Committed by Offenders (N=41,974)

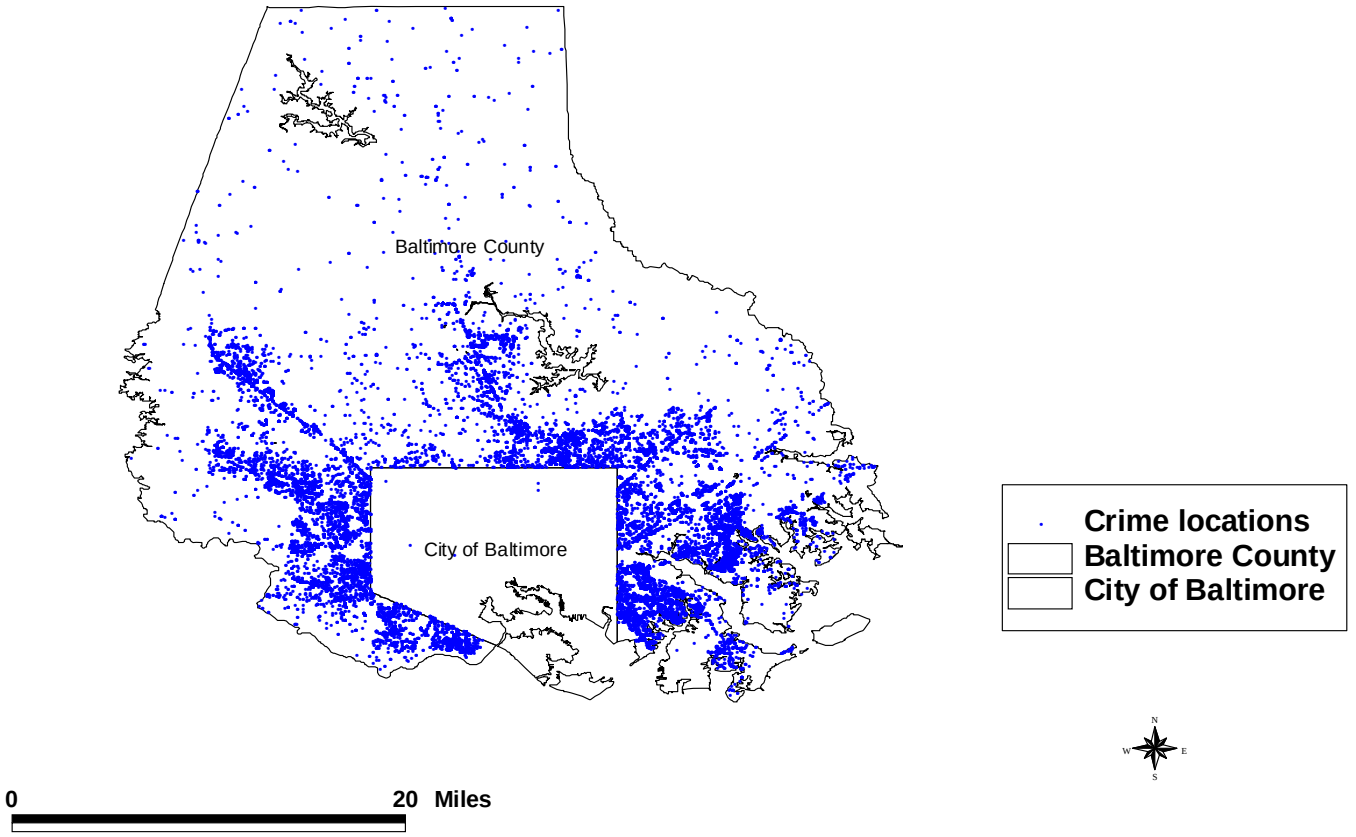


Figure 26.3:

Baltimore County Offender Residences: 1993-1997
Location of Offenders When Arrested (N=41,974)

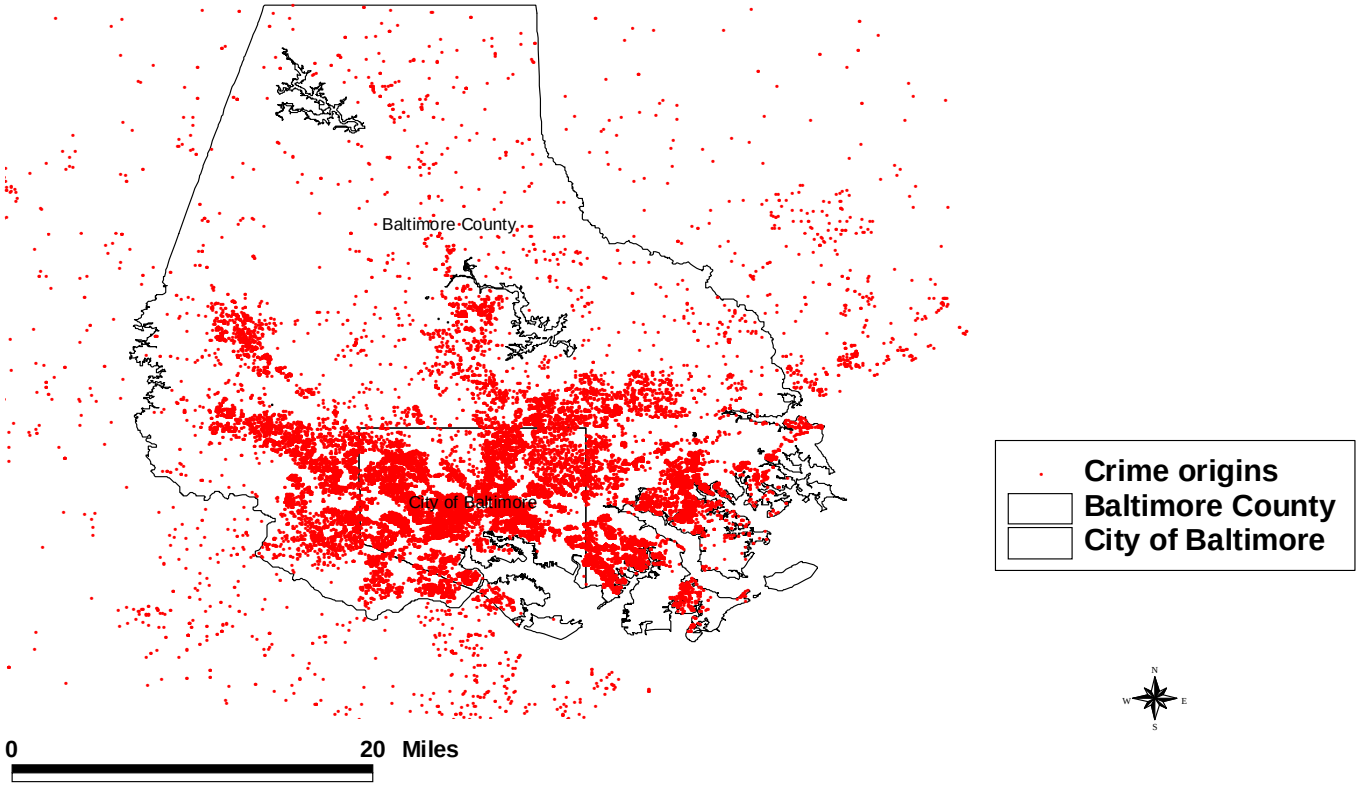
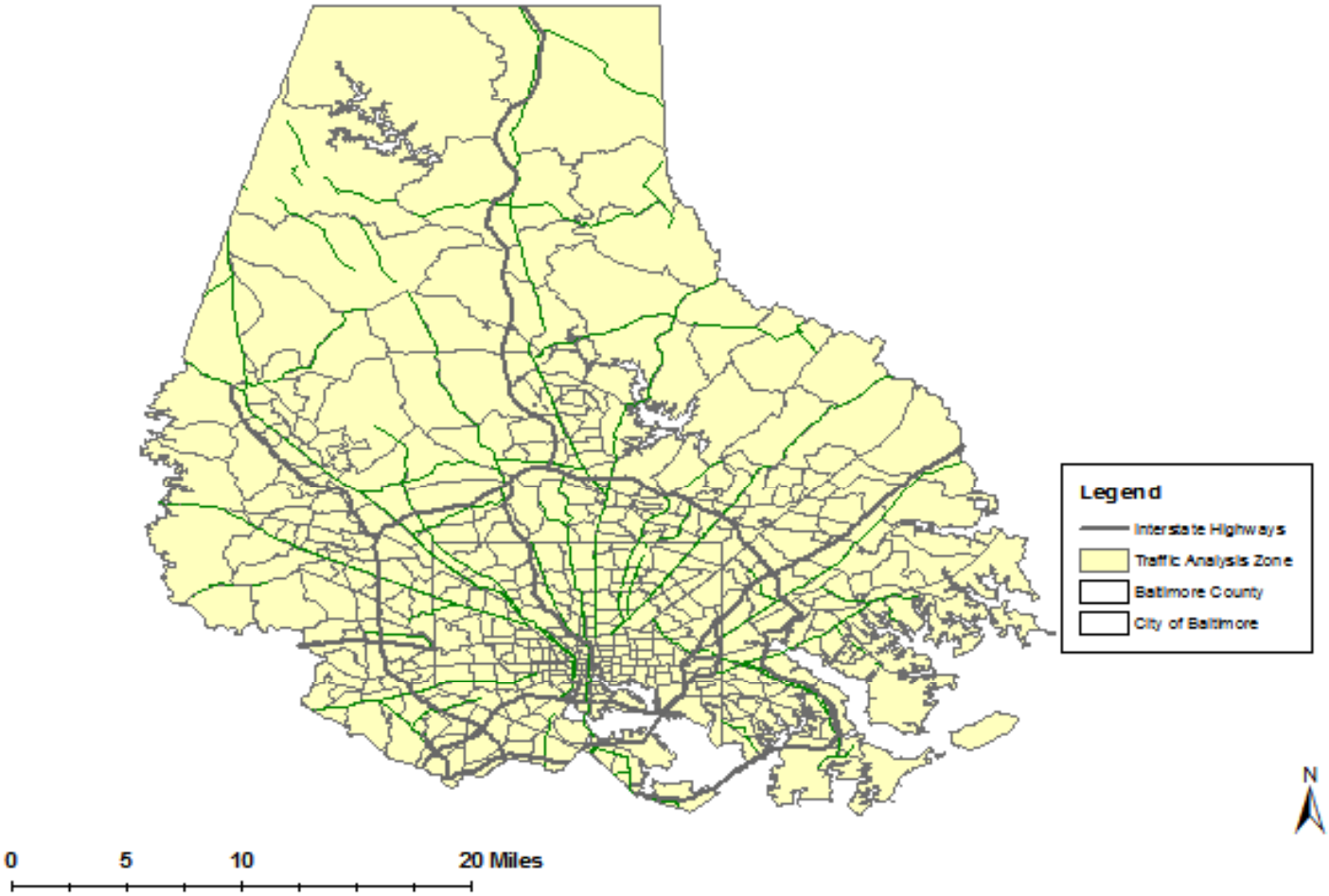


Figure 26.4:

Metropolitan Baltimore Traffic Analysis Zones: 1998



How does one assign crime events to a zone? There are two general ways to do this:

1. Nearest zone centroid - events are assigned to the zone centroid that is closest.

2. Point-in-polygon - events are assigned to the polygon within which it falls.

With the nearest zone centroid method, an incident is assigned to a zone to which it is closest whereas with the point-in-polygon method, an incident is assigned to a zone in which it falls within the boundary of that zone. Most GIS packages have a point-in-polygon routine and can implement that method.

In *CrimeStat*, on the Distance Analysis I page, there is an Assign Primary Points to Secondary Point routine that will make this assignment based on either method (see Chapter 6). In both cases, the incident file must be the primary file and the zonal file must be the secondary file. In the nearest zone centroid method, the routine will assign each event to the centroid to which it is closest. It will then sum the number of incidents assigned by zone and will add this as a new field to the secondary file (called *Freq*).

In the point-in-polygon method, the user must also provide the boundary file for the zones as an *ArcGIS* shape file. The routine will read the boundary file and will determine in which polygon an incident falls, and will then assign the incident to that zone. As with the nearest zone centroid method, it will then sum the number of incidents assigned by zone and will add this as a new field (*Freq*) to the secondary file. Chapter 6 presents details of these two routines, and is not repeated here.

There are advantages and disadvantages to each method. The nearest zone centroid has attributes that are probably closest to the location where the incident occurs. This is important in relating socioeconomic and land use characteristics to the events during the trip generation stage (see Chapter 27). Typically, social characteristics change gradually over an urban landscape so that an incident is probably closer to its nearest zone centroid than to any other zone centroid. In the case of the point-in-polygon method, incidents are not necessarily assigned to the nearest centroid since zonal polygons are frequently irregular in shape. Thus, to represent the underlying characteristics of the location in which the incident occurs by a point-in-polygon may end up assigning an incident to a zone that is quite different from where it should be located.

On the other hand, the main advantage of a point-in-polygon assignment is if the zone has a meaning in terms of containment or membership. For example, if a police reporting district (which could be a sub-set of a larger police precinct) is used as the zonal model, assigning incidents to the reporting district within which they fall will ensure that the incidents are assigned to the correct police precinct.

In other words, if it is important that events be assigned to the area to which they belong, then the point-in-polygon method is usually the best. On the other hand, if it is important that the incidents be assigned to the zone to which they are most similar, then the nearest centroid method is usually the best.

For Baltimore County, figure 26.5 shows the number of crimes by origin zone while Figure 26.6 shows the number of crimes by destination zone. In both cases, events were assigned by the nearest zone centroid method.

Adjusting Crime Events Estimated from Arrest Records for Accuracy

There is another subtlety that affects the assignment to a zone. The method that has been described assigns records in which there is both an origin and a destination location, such as an arrest record. The reason for doing this is that there is an implied trip between the origin and the destination, as was discussed above and in Chapter 25. However, there may be a difference between the distribution of crimes by destination from the arrest records and the actual distribution of crimes from all incidents. The reason is that arrest records represent only a sub-set of all the crime records and, often, a small sub-set. If there are any spatial differences in the arrest likelihood across a metropolitan area, it is possible that some areas will have a higher proportion of offenders being arrested than other areas. The result would be a discrepancy between the distribution of crimes by arrested individuals and the actual distribution of crimes. In other words, the distribution of crimes as identified by the arrest records could be a biased estimate of the actual distribution of crimes. The result could be that the origins of those offenders who were caught will be exaggerated relative to the origins of those offenders who were not caught, and the entire model could end up being biased.[3]

If there is a sizeable discrepancy between the distribution of crimes from the arrest records and the actual distribution of crimes, it is important to correct this. In the Assign Primary Points to Secondary Points routine on the Distance Analysis II page, it is possible to weight the assignment by another variable. This variable can reside on either the secondary (zone) file or on another file. A typical correction weight variable would be a proportion that adjusts the empirical distribution of crime destinations by the true distribution. Thus, a weight greater than 1.0 would increase the proportion whereas a weight smaller than 1.0 would decrease the proportion. A weight of 1.0 would maintain the same proportion.

---

[3] In the usual travel demand modeling conducted by transportation planners, the origins are assumed to be more accurate than the destinations. The origins are identified typically from census and other population enumerations whereas the destinations are estimated from surveys and employment databases. In the case of crime travel, however, the destinations are known with much greater accuracy since those locations are documented in police reports.

Figure 26.5:

Crimes Origins by TAZ
Number of Crimes Originating in TAZ
Baltimore County: 1993-1997

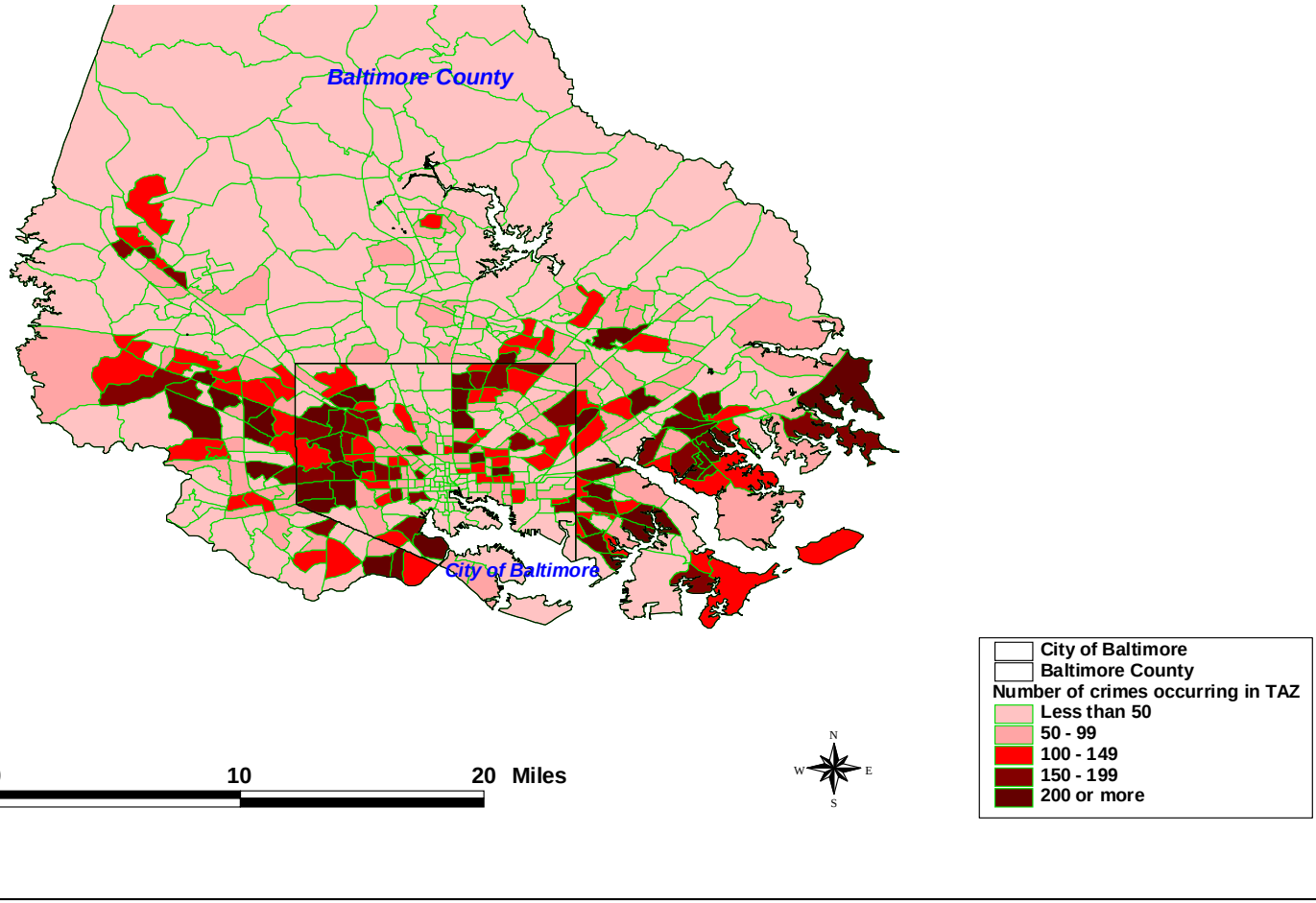
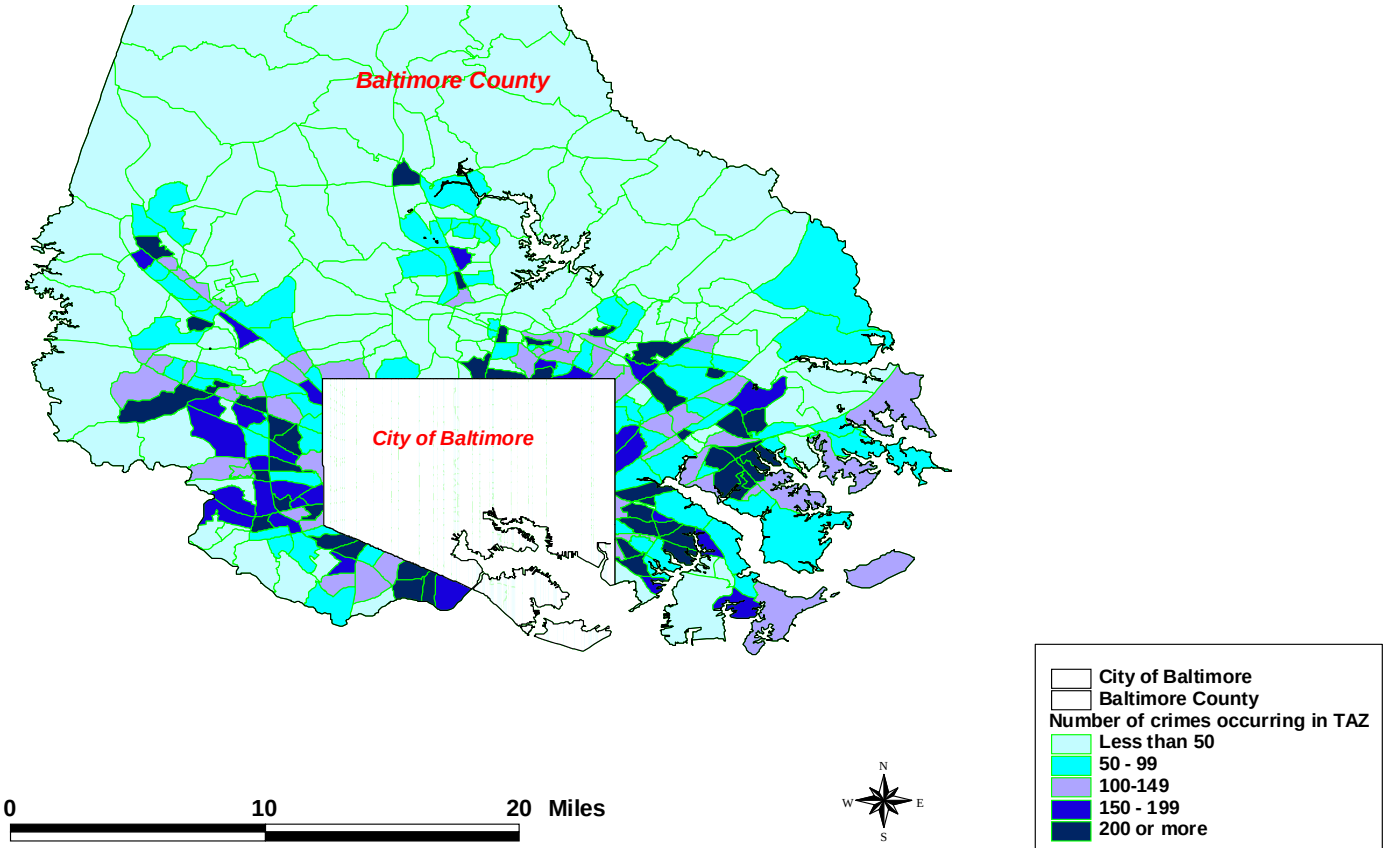


Figure 26.6:

Crimes Destinations by TAZ
Number of Crimes Occurring in TAZ
Baltimore County: 1993-1997



In order to do this, however, one has to convert the number of crime destinations into proportions. Let's take an example. Suppose the empirical and true distribution of crime destinations was as follows (Table 26.1):

Table 26.1:
Proportional Weighting Empirical Assignment of Crime Destinations

Zone	Empirical Distribution	True Distribution	Proportional Weight
101	.04	.05	1.25
102	.03	.025	0.83
103	.015	.015	1.00
etc.			

In the example, the actual (true) distribution of crimes for zone 101 is greater than what was measured in the incident-to-zone assignment by a factor of 1.25 to 1 (i.e., .05/.04). Thus, the weight assigned to zone 101 is 1.25. In zone 102, on the other hand, the actual distribution of crime destinations was smaller than what was estimated from the incident-to-zone assignment by a factor of 0.83. Thus, the weight assigned to zone 102 is 0.83. Finally, the proportion of crimes in the empirical and actual distributions for zone 103 is exactly the same. Thus, the weight assigned to zone 103 is 1.00.

The weight variable will be typically a column in the secondary file that corrects the empirical distribution. Naturally, the first time this is done, an analyst would probably not know the empirical distribution. Thus, it will be necessary to repeat the incident-to-zone assignment, the first time in order to count the empirical distribution while the second time to weight that count by the correction factor (which will have been added as a variable to the secondary - zonal, file). See Chapter 6 for a more complete discussion of weighting a primary points (incidents) to secondary points (zones) assignment.

Note, the adjustment of the empirical count (assignment) is done usually for the destination variable, not the origin variable. In the case of crime events, police will know the destination of the crime a lot more accurately than they will the origin since there is a crime record on file for the incident. Hence, any discrepancy between the empirical distribution of crimes and the actual distribution will only be known for crime locations (destinations). Therefore, in correcting the empirical distribution, we are assuming that we are also correcting the true distribution of origins, too. It should be obvious, though, that we really don't know. Unless one can obtain a "true" distribution of crime origins and, thereby, correct the origin distribution as well as the destination distribution, one has to assume that the adjustment in the destinations will

also correct the distribution of the origins during the balancing stage (see Chapter 27 on trip generation).

Obtaining Crime Data by Sub-Types

Till now, the discussion has focused on the total number of crimes that occur within a zone. Clearly, it is possible (and preferable) to break this down into distinct sub-groups. Thus, a separate distribution for robberies, burglaries, vehicle thefts, homicides, and other crime types can be complied. In each case, the separate distribution is being assembled in order to produce distinct models of crime travel by that type. The journey-to-crime literature has long illustrated the differences in travel distance by crime type and it would be expected that there are substantial differences in travel patterns as well. Most crime analysts and researchers will want to break down crimes into these distinct categories. Similarly, an analysis by time of day or day of week also would require breaking down crimes by these different temporal categories. In general, an analysis of all crimes is not very meaningful for most police departments. Instead, the focus has to be on crime types and, perhaps, times of day with other sub-sets also being important (e.g., method of operation, use of weapons).

The method used to assign these individual crimes to zones would be, however, exactly the same as for the total number of crimes that was illustrated above. As with the total number of crimes, there would be differential weighting of zones in order to correct any bias in the distribution of crimes calculated from the arrest records compared to the actual distribution of incidents as identified by total crime reports.

Adequate Sample Size

A problem with this approach arises, however. By breaking down crimes into distinct sub-groups (by crime type, time of day, day of week, method of operation, etc), smaller samples are produced. As the sample size decreases, the likelihood of modeling error increases. If the sample is too small, then any of the zonal estimates that are produced in the trip generation stage will be subject to considerable sampling error. Similarly, in subsequent stages (trip distribution and mode split), these small sample sizes are further broken down into cells with very small sample sizes, with most having zero incidents. In other words, sampling error becomes a problem if the total number of crimes is broken down into very small sub-sets, and the model becomes unreliable.

How would one know whether a model is unreliable or not? Probably the simplest way is to repeat the model on two different years worth of data. That is, the analyst constructs the travel demand model on one year's worth of data and then repeats it on another year. If the variables selected during the trip generation stage are the same and if their coefficients are approximately

equal, then the model would appear to be reasonably stable. On the other hand, if there are substantial differences in the selected variables and in their coefficients, most likely the data set was too small for the construction of a stable model. One could do formal tests on differences between the coefficients to see whether they are similar or different. But, a general review of the coefficients should indicate whether there is stability or variability. There is not a 'hard and fast' rule since any differences could be due to real changes in the environment creating crime. But, unless there is some obvious explanation for the differences, most likely they indicate that a model is too unreliable to be used from one year to the next (i.e., the sample size is probably too small).

Thus, there is a balance that has to be maintained between having a large enough sample to produce reasonably reliable trip generation and trip distribution coefficients, and breaking down the data into more meaningful categories for analysts and researchers. In general, I believe it is a good idea to model all crimes first before modeling specific sub-types. The reason is to establish baseline characteristics - variables and coefficients. It will become easier to understand how different crime sub-types vary once the overall distribution is known.

Developing a Predictive Model

The above discussion dealt with summarizing crime incidents by zones, both the location where the crimes occurred (the destinations) as well as the locations where the offender was living (the assumed origins). In order to develop a predictive model of crime origins and destinations, it is also necessary to put together a data set of predictive variables. Typically, these will be socioeconomic and land use variables, though other types of variables can be included.

Obtaining Socioeconomic Data

Population

The most common type of predictive data will be socioeconomic variables. Among these are population, employment, income levels, poverty data, and household characteristics. At the minimum, population will be an important variable. As mentioned in Chapter 25, the crime travel demand model is an aggregate (volume) model. That is, it counts the total number of crime trips (by origin and by destination). Since, the number of trips is generally a function of the total number of persons living in a zone, all other factors being equal, population inevitably will enter as either the most important or among the most important variables, as both an origin and a destination variable.

Population could be measured by sub-sets (or proxy) variables, too. For example, the number of households, the number of teenagers, and the number of married couples are also sub-

sets of the total population; the correlation among these variables is usually very high. Which variable is chosen will depend on what type of crime is being predicted. For the total number of crimes, probably the total population (or total number of households) should be used because it is a larger and more stable estimate of the total "at risk" population. For specific crimes, however, it may be desirable to choose a sub-set of population. For example, for car thefts, the distribution of males, ages 16-30, might be a more intuitive baseline variable as those age groups contribute disproportionately to vehicle thefts (as they do to most crime types). The disadvantage in using this variable may be the smaller sample sizes that are obtained for some zones. A good way to test this is to model it twice, once with total population and once with the sub-set variable. If the overall predictability of the model is about the same (or, better, if the sub-set variable predicts better than the total population), then the use of the sub-set population will be preferable to the total population. On the other hand, if there is not much difference, stick with total population as it is a larger, and more stable, variable.

Employment

A second variable that usually comes up is total employment. This is particularly valuable as a predictor of crime destinations since many crimes are attracted to employment areas (e.g., robberies, burglaries, vehicle thefts). Usually a distinction is made between *retail* and *non-retail* employment, though other distinctions can also be made (e.g., office employment, government employment, military employment). The reason is that retail employment is usually found in commercial areas (e.g., shopping malls, strip malls, retail centers). In the case of Baltimore County, for example, retail employment is the strongest predictor of crime destinations.

As an origin variable, too, employment could be important. In the three models that were compared for this version of *CrimeStat* (Baltimore County, Chicago, Las Vegas), employment was seen as a predictor variable for crime origins in several cases, too, usually as a negative predictor (i.e., less employment is associated with more crime). The reason may be less clear, but may have to do with the lack of opportunities in certain districts and neighborhoods.

Income levels

Another obvious variable is income measured in some way. The relationship between crime and low income has long been noted. There are several possible income-type variables that could be used in a model. The most obvious is the total income level of a zone. The U. S. Census Bureau has a total income variable that is part of their SF 3 release (U.S. Census Bureau, 2011a). This measures the total of all household incomes in the census. While this variable captures the total available income in the zone, it is not a very intuitive measure. Consequently, other measures are usually used, such as income per capita or median household income. Median

household income is usually a more typical measure since the average income per person can be affected by extreme values.

An important issue about income levels, no matter how measured, is that they inflate over time. That is, since income reflects monetary value at any one point, it does not have a fixed reference point. What this could mean in a model is that, over time, income levels will increase (in absolute terms) due simply to inflation. A model that established, for example, a negative relationship between income and crime (i.e., the higher the income of the zone, the less crime) for one year would end up predicting lower crime levels for another year simply due to inflation.

It is important to standardize income in order to prevent the impact of inflation affecting the model. There are two ways that this is usually done. First, one can standardize income by subtracting the mean and dividing by the standard deviation. That is,\

$$Z_i = \frac{I_i - \bar{I}}{SD_I} \quad (26.1)$$

where I_i is the income of each zone, i, $\bar{I}$ is the mean income of all zones, and Sd_I is the standard deviation of all zones. This is a classic standardized measure.

A second way to standardize income is to define *relative* income. That is, the income level of each zone is compared to the income level of the zone with the highest income. That is,

$$I_i = \frac{I_{max} - I_i}{I_{max}} \quad (26.2)$$

where I_{max} is the income level of the zone with the highest income. This index measures the income of a zone relative to the income of the highest income zone. The closer the income level of the zone is to the highest income zone, the smaller the index. Thus, this is an *income inequality index*, similar to the Gini index though more simply calculated. The zone with the highest income will have a value of 0 whereas the zone with the lowest income will have a positive value roughly reflecting the relative differences in income levels between the lowest and the highest.

Each of these measures will prevent a shift in the predicted values due to inflation, though they each measure slightly different attributes; the first measures just absolute income levels (standardized) while the second measures the degree of inequality.

Another type of income variable is the number of persons living under poverty. Again, the relationship between poverty and crime has long been noted (Bursik & Grasmick, 1993).. Thus, a variable that measures poverty directly could add sensitivity to a model that simple

income might not detect. The issue of measuring poverty, however, is a complex one. Different government agencies use different measures. For a discussion, see Citro and Michael (1995).

In general, typically the variables 'median household income' and the' number of persons (or households) living under the poverty line' do correlate quite well. Therefore, it is unlikely that both variables would be significant in a regression equation without, essentially, measuring the same thing. The same is true for education and income, which tend to correlate quite highly. Again, both variables in a regression equation would, essentially, be measuring the same thing. Thus, in a regression model, it is important to select only the strongest and most stable income variable in order to avoid duplicate measures (multicolinearity). I will return to this point in the next chapter.

Other socioeconomic variables

Other socioeconomic variables might be useful in a predictive model. Among these are race or ethnicity, vehicle ownership, number of single parent households, number of unemployed workers, number of persons living in large rental buildings, and others. Again, these variables might produce greater differentiation in a model. But, at the same time, they tend to overlap with income variables and may be measuring the same thing.

Obtaining Land Use Data

Aside from socioeconomic variables, there are land use variables that could be important in predicting both crime origins and destinations. Among these are parks, bars, pawn shops, check cashing businesses, the location of shopping malls, retail space, stadiums, train stations, intra-urban metro stations, bus stations, parking lots, hospitals, and adjacency to major freeways or arterial roads. There are a wide variety of land use variables that appear to be important in attracting crime as well as in providing an environment that may encourage people to commit crimes. A thorough elaboration of potential land use variables would help to identify particular attributes associates with crime and, thereby, increase the predictive ability of a model.

There are two ways to document these land uses. One is as a simple categorical ('dummy') variable whereby the field is given a '1' if that land use exists in the zone and a '0' otherwise (e.g., there is a park in the zone; a freeway runs through the zone; there is a stadium in the zone). The second is a count of the level of that land use variable (e.g., the number of bars; retail square footage; park acreage; number of parking stalls in a parking lot). The second variable is, clearly, more precise than the first, but is much harder to document. The availability of data will be a constraining factor in building up a set of land use variables that might predict crime origins or destinations.

Still, before an extensive data inventory is initiated, some cautionary words are in order. In the three studies illustrated in this version of *CrimeStat*, however, few land use variables survived once population, employment and income levels were included. The reason is that many land use variables correlate with these basic variables (e.g., the amount of retail space correlates with retail employment; bars correlate with low income). Thus, in spite of intuitively being related, it was found that most of the land use variables did not improve the models beyond the basic variables.

Special Generators

There are exceptions, however. Particularly, there are *special generators* that attract crimes out of proportion to the amount of employment at those locations. Among these are stadiums, major train stations, airports, and large parks. Because these are major regional facilities and, in the case of stadiums and parks, used only periodically, they may attract more crimes that would be expected on the basis of the level of employment at those locations. Traditional travel demand models have incorporated these as special variables because they can account for variability that is not general throughout the study area. In the next chapter, I will discuss this in more depth.

Spatial Location Variables

Centrality

In addition to socioeconomic and land use variables, spatial location variables *might* be relevant. There are two types of spatial location variables that might be relevant. The first is the *centrality* of the metropolitan area. In most American cities, the central downtown area has a uniqueness that is greater than that which is explained by any one variable. For example, not only is there a large amount of employment in most Central Business Districts (CBD), but there are amenities that are associated with a central location. Usually, there is a greater concentration of restaurants and stores in CBD's and other employment centers. Entertainment activities are often more concentrated in the CBD; this is not true in many large metropolitan areas (e.g., Los Angeles), but it is true in enough of them to make the CBD an entertainment center as well as an employment center. Similarly, transit lines tend to concentrate in the CBD.

In other words, the CBD is a unique place that affects crime trips. Some CBD's have a large number of crime incidents whereas others do not. Nevertheless, measuring it in a predictive equation *might* increase the predictability of a production or attraction model. A simple variable is the distance from some point within the CBD, for example distance from the City Hall. Zones that are close are liable to have a greater number of crime productions and crime attractions,

especially, than zones farther away. This type of spatial effect is very similar to the *first-order* effect described in Chapter 6.

The use of a distance from the CBD variable can usually strengthen a regression model. A study which illustrates how centrality predicts male-female differentials in motor vehicle crashes in Houston is by Levine (2011); male drivers are much more likely to get involved in crashes in the central city, particularly the CBD, than females whereas the differentials are much less in the suburbs. Another example is by Levine and Canter (2011) who showed that distance from the CBD predicted positively the number of DWI trips that ended in crashes that originated in each zone in Baltimore County (i.e., zones farther from the CBD produced more). Similarly, Levine (2007) found that distance from the CBD negatively predicted the number of bank robbery trips that originated in each zone (i.e., zones closer to the CBD produced more).

Local spatial autocorrelation

The second type of spatial effect is a localized similarity between adjacent zones. In other words, there frequently is spatial autocorrelation in crime productions or attractions between adjacent zones. These are the *second-order* spatial effects described in Chapter 6. Zones that have a lot of crimes occurring within them are frequently located next to zones that also have a lot of crimes occurring, and the converse.

If the user wants to incorporate local spatial autocorrelation explicitly in the trip generation stage, then the use of a Anselin's Local Moran, the Getis-Ord Local 'G' (see Chapter 9) or a simple adjacency measure (e.g., '1' if the average of adjacent zones is greater than the mean for all zones and '0' if it is not) may be sufficient in account for the localized spatial autocorrelation.

However, it should be noted that apparent second-order spatial effects may be simply by-products of first-order spatial effects. Because of the concentration of events in the central city, there are usually more local hot spots in the central city, too. Before arriving at a conclusion that there is definite local spatial autocorrelation, a user would be wise to incorporate a first-order global spatial autocorrelation variable, such as distance from the CBD. If there is additional variability after that in incorporated, then the local effect would most likely be rea.

Estimating spatial effects

The spatial regression models discussed in Chapter 19 explicitly incorporate spatial effects as a predictor variable. If a trip generation model includes both a first-order spatial effect (e.g., distance from the CBD) and a local spatial autocorrelation adjustment for each case (the Phi

coefficient to use the terminology of Chapter 19), then the model will handle both types of spatial autocorrelation.

An alternative is to ignore the spatial effects in the first stage – trip generation, since the second stage of the model - trip distribution, incorporates an explicit spatial component by weighting distance in estimating the interaction between zones. Thus, any spatial error produced during the trip generation stage is frequently compensated for during the trip distribution stage.

There are advantages and disadvantages to including first- or second-order spatial effects in a travel model. Since the trip distribution stage has an explicit spatial interaction term, any errors from the first stage (trip generation) are usually accounted for during the second stage. Thus, there is little advantage to be gained from including a second-order (spatial autocorrelation) variable. However, including a first-order variable can usually improve the predictability of the trip generation model.

Defining Policy or Intervention Variables

Aside from socioeconomic and land use variables, a model might include some policy or intervention variables. One of the best uses of a travel demand model is to model the likely effect of a change in one of the predictive variables. A simple one would be the likely effect of building a new facility, for example a shopping mall. In the estimation stage, if the analyst can show that shopping malls are associated with higher (or lower) numbers of crime occurring, then a theoretical mall could be placed in a zone and the model run with that as a new input for the zone (with every other variable being the same for all zones). Since the travel demand model is sequential, the impact of new crime trips being attracted to the zone can be followed through the different stages of the model.

There may be other policy or intervention *experiments* that can be conducted with a crime travel demand model. In each case, it is necessary to include the variable in the estimation model to establish a coefficient for it. Then, in the simulated experiment, the variable is re-arranged or allocated differentially and the model is recalculated. Again, the result can be used to estimate what the likely effects of the intervention could be on crime travel patterns.

Among the possible policy or interventions are the construction of a particular type of facility (as mentioned above with a new shopping mall), changing the level of policing in a zone, the creation of a drug treatment center, the establishment of a job retraining center, or the reduction in the number of adult book shops. There are a large number of possible interventions that might affect the level of crime - either produced (origins) or attracted (destinations). Further, not all of the interventions might reduce crime levels, but some could even increase it (e.g., add new shopping malls). Nevertheless, the ability to add interventions in the model makes it a useful

device to estimate the likely effects on crime levels without having to actually implement the changes.

In the three studies presented in this version of *CrimeStat*, there were no interventions that were estimated. Examples of simulated interventions can be seen in Levine and Canter (2011) who modeled selective police interventions to reduce DWI trips in Baltimore County that end in crashes from zones where a higher proportion of offenders resided and from zones where a higher proportion of crashes occurred and Levine (2007) who modeled both bank robbery trips in Baltimore County from residence to the bank and the escape route trip back to the residence. Still, this type of experiment or 'variable' is an important one and which could make the crime travel demand model a very powerful analysis tool.

Where to Obtain these Data?

Many of these data are easily found while other data are more difficult to locate. A lot of socioeconomic data is available in the decennial census and distributed by the U.S. Census Bureau. Data on population, households, and income levels can be obtained from the Census Bureau for geographies as small as blocks or block groups. One of the deficiencies of the census data, however, is the lack of information on employment.

An alternative is to obtain data from a Council of Governments or Metropolitan Planning Organization. A Council of Governments (COG) is a regional association of cities and counties that is involved in planning; sometimes it is called an Association of Governments. Virtually every metropolitan area in the United States has a COG that can be a source of information on both population, employment, and, occasionally, land use. Many COG's have a forecasting group that estimates both population and employment, sometimes for very small geographical units. The Houston-Galveston Area Council, for example, has an extensive database of all firms with 10 or more employees and updates this continually utilizing information on business permits, purchased lists from other organizations, and aerial photographs for identifying new commercial developments. They produce estimates of employment for small grid cells that are approximately 1000 feet on a side; however, these data are released only at the Traffic Analysis Zone (TAZ) level. For more information and a detailed list of local regional councils, see NARC (2012).

A Metropolitan Planning Organization (MPO) is a regional transportation planning agency. In many metropolitan areas (e.g., Los Angeles, Houston, Washington, DC), the MPO is part of the COG while in other metropolitan areas (e.g., San Francisco, Chicago), it is not. They will obtain both population and employment data for the TAZ's as part of their travel modeling functions. For more information and a detailed list of local MPOs, see AMPO (2012). In short, it is generally possible to obtain data on population and employment from either COGs or MPOs.

Land use data is more difficult to obtain. Simple information can often be obtained from Yellow Pages or online business directories, for example the location of bars and nightclubs. More detailed data may have to be obtained from particular cities and counties. Generally, larger cities have a planning department or a public works department who maintains some land use data. The quality of this information will vary, however, and may not be consistent across jurisdictions. In a large metropolitan area, it may be possible to obtain regional land use information from the COG, the MPO, regional utility companies, a database of business permits, tax assessors' offices, or even the Army Corps of Engineers.

The point that has to be realized is that a lot of effort is needed to put together a data base for modeling crime travel. Once developed, however, it can be used repeatedly as predictors for different types of crime and can be updated more easily. Like a GIS system, there is a substantial amount of effort 'up front' in order to build a model. But, once collected, the information can be very useful for a multitude of purposes.

Creating an Integrated Data Set

The information that has been collected - both data on crime origins and destinations as well as socioeconomic, land use and policy interventions, needs to be integrated into a single zonal model. That is, the data need to be allocated to zones, both origin zones and destination zones. The result will be ***two*** different data sets, one for crime origins and one for crime destinations. The origin data set will cover the origin zones while the destination data set will cover the destination zones. The same predictor variables, however, can be in both data sets as these variables could predict either origins or destinations, or both.

Allocating Data to Zones

There are two steps in assembling the data into two data sets. First, the data have to be allocated to the zonal system used. In some cases, these data may be easily available (e.g., obtaining population and employment data by TAZ's when the TAZ is the zonal unit used). In other cases, it may be necessary to allocate the data from one geographical zonal unit to another (e.g., from census block groups to TAZ's). GIS is a very powerful tool for allocating data from one "layer" to another. However, it has to be realized that errors will result from an allocation. For example, breaking up a larger zone into small sub-zones (e.g., breaking up a large census tract into four small grid cells) will lead to some error in the allocation. The GIS splitting routines usually assume that the data are split proportionately between the four 'pieces'. Thus, if employment from a census tract is allocated to two grid cells, one assumes that the workers are uniformly distributed within the census tract and the two grid cells will each capture a share equal to their area relative to the larger tract. This may or may not be true. Where it is not true, adjustments need to be made to ensure that zones represent relatively uniform populations.

The point is, there is error in allocating data from one type of unit to another, and the analyst has to be aware of these potential sources. It is generally better to obtain data at the smallest possible geographical unit in order to minimize the splitting problem described above. Aggregation usually causes less error than splitting. On the other hand, as mentioned at the beginning of this chapter, the larger the zonal unit that is used the greater the likelihood that there will be within-zone (intra-zonal) trips.

Combining Data into Origin and Destination Data Sets

The second step is the combining of all the data into two separate data sets, one for origins and one for destinations. All the data that are used for the origin model should be together while all the data that are used in the destination model should be together. Many variables will be in both data sets (e.g., population, employment, income) whereas some variables only make sense as an origin or a destination variable (e.g., residential areas as an origin variable for bank robberies; a rail station as a destination variable for larceny or robbery). Since the origin zones will usually be more numerous than the destination zones (because they include the destinations and those from surrounding jurisdictions), the data have to be consistent across all zones that are used.

For use in the *CrimeStat* crime travel demand module, these data sets should be in one of the acceptable formats (Excel, dbf, dat, or ODBC-compliant). I have found that building the data first in a spreadsheet (e.g., Excel) is easier to do because variables can be more easily added. Once constructed, the spreadsheet is converted into a dbf file for use by *CrimeStat*.

Obtaining Network Data

The final type of data that needs to be obtained is a network. This is important for the third and fourth stages in the crime travel demand model - mode split and network assignment. In the mode split routine, trips from each origin zone to each destination zone are divided into different travel modes. For driving travel modes, travel has to go along a road network. For walking or biking, there may be additional segments that are not in the road network (e.g., bike paths, short cuts for pedestrians); these can usually be added to the road network to make a more realistic representation. However, for transit modes, the trips have to go along a transit route. In the network assignment routine, all zone-to-zone trips by each travel mode are assigned to particular routes. For this, a network is needed, one for each mode.

In both these cases, travel occurs along a network. That is, the distance (or travel time or travel cost) from one location to another is calculated using the network, rather than as direct or indirect distance. A network is a collection of segments that are interconnected. Travel can only occur on the segments. Each segment has two or more nodes and one or more connecting lines.

Travel is from segment to segment. Hence, the *end nodes* have a special status as the connectors which allow travel from one segment to another.

In Chapter 30, a more extensive discussion of the shortest cost/path algorithm used for network travel is explained. But, essentially, a 'trip' goes from the origin location to the closest location on the network. It then proceeds along the network, taking the shortest path, until it reaches a node closest to the destination. It then travels from that node to the final destination. Thus, the *representation* of the network is very critical. It has to be accurate and reasonably comprehensive.

There are three types of basic networks that need to be considered:

1. Road network (with additional walking or biking segments)
2. Bus network
3. Train network (if appropriate).

In addition, there can be specialized bicycle networks that are distinct from the road network. However, most transportation agencies model bike trips using the road network. I will discuss each of these.

Road Network

In a GIS system, there are typically two types of road networks that are used:

1. A bi-directional (or linear) network
2. A single-directional network.

Bi-directional road network

In a bi-directional network, travel can occur in both directions along a segment. A typical example is the TIGER system created by the U.S. Census Bureau (2011b). In this system, each segment typically represent the travel along a road from one intersection to another (i.e., a block in length), though there are exceptions. Travel can occur in both directions in the network unless there are special codes added to indicate a one-way street. The TIGER system, in particular, has a number of attributes associated with it - sides (left side, right side), address ranges (on both sides), census and political designators (again, by sides), and other attributes. This type of network is very common in GIS systems and is widely used in police departments. Because of the address ranges and because it is easily available from the U.S. Census Bureau or companies who improve the TIGER system, this type of network forms the basis of most geo-coding systems.

There are problems with a bi-directional network, however. Among these are the inabilities to distinguish direction and one-way streets. From a network modeling perspective, travel can occur in either direction. It is possible to put a field in the data base that identifies whether the street is one-way or not and to indicate the direction of travel. But, this has to be added by the user since the TIGER system does not specify that information.

A second problem is the lack of information about travel time or cost on the network. The only metric in the TIGER system are address ranges and, implicitly, distance. However, since travel varies substantially by type of road (larger functional classes have higher speeds) and by time of day due to differing levels of congestion, such a system lacks very important information for modeling travel. The TIGER (or similar) system does have functional class codes that distinguish different levels of road capacity (e.g., Interstate highways, state highways, principal arterial roads, collector roads, etc). It is possible to assign arbitrary average speeds to each of these classes (e.g., 45 miles per hour to an interstate highway; 30 miles per hour to a principal arterial; 20 miles an hour to a collector road; and so forth). By doing so, a reasonable approximation to actual travel can be obtained. However, there is still not a sensitivity to travel time by time of day. For example, in an urban area, travel at the peak afternoon 'rush hour' (e.g., 3:30 PM - 7 PM) will be, on average, a lot slower than at off-peak hours.

This brings up a third problem, namely that there is no interaction between the direction of travel and the travel time. On most principal arterial roads, travel is unequal in speed at any one time. For example, in many metropolitan areas, travel towards the downtown area is much slower in the morning than in the opposite direction, whereas the reverse is true in the afternoon. A bi-directional network cannot distinguish this and the analysts have to add multiple fields to the attribute file in order to make these distinctions (e.g., PM peak from node A to node B direction; PM peak from node B to node A direction; etc).

A fourth problem may or may not exist with a bi-directional network. These networks were designed to allow the U.S. Census Bureau to carry out the decennial census. Thus, a lot of attention has given to accuracy of streets and address ranges. Much less attention has been paid to the connectivity of the streets. A lot of the digitizing that goes into the network has been done by local governments, and the quality of this digitizing varies considerably. Some jurisdictions have very precise networks that are updated frequently while other jurisdictions have poorly defined networks that are often out of date. Drivers may know that they can travel from point A to point B via road C, but the network may not have been sufficiently updated to allow that trip to occur in a representation. In some cases, gaps between segments have been noted; the gaps may be very small, but they would prevent a model from 'traveling' from one segment to the next.

Single-directional road network

A single-directional network, on the other hand, separates travel in each direction. For example, if there are two nodes that connect a segment (node A and node B), then there are typically two segments for travel in each direction (from node A to node B, and from node B to node A). In this representation, a one-way street is simply a segment that does not have a reciprocal pair (i.e., there is only a node A to node B segment, and not the reverse).

Most transportation agencies use single-directional road networks for their travel demand modeling. The reason is that multiple attributes can be assigned to each direction separately, a feature that simplifies the building of a realistic network. Thus, speeds for different time periods can be assigned as separate fields on each segment (or, what is usually, done there are separate networks for the different travel periods that are modeled). Travel volumes can be assigned to each segment which, in turn, allows the creation of a *vehicle miles traveled* (VMT) field (length x volume). VMT, in turn, can be combined with travel speed to produce an estimate of travel time (e.g., VMT divided by speed - in miles per hour, times 60 to produce minutes traveled). Further, one-way streets are automatically handled since each direction is a separate segment (i.e., there just won't be a reciprocal pair in the opposite direction).

In short, a single-directional network allows more flexibility in the creation of a network and the ability to distinguish travel in different directions as well as travel time by direction and time of day. It is not surprising, therefore, that most travel demand models use a single-directional representation. Note, one can add these attributes to a single-directional network, but this requires many additional fields.

A further strength of a single-directional network is that it is usually quite up-to-date and connectivity has been ensured. Most transportation agencies spend a lot of time cleaning and updating the network. While there are always errors in a network representation, the accuracy of most modeling networks is very good.

There is a downside to single-directional networks, however. Typically, most single-directional networks model only the larger roadways, those that contribute to regional travel. Thus, all freeways, principal arterial roads, minor arterial roads, and some collector roads are included. However, most neighborhood streets are not included. The reason this is done is because the travel demand model is aimed at estimating regional and sub-regional travel patterns. Very localized travel is not of importance (and, in fact, is typically intra-zonal in nature). The result is a very efficient network because it is a lot smaller. But, there may be some error by using a 'skeleton' network. In particular, local travel might be distorted with such a simplified network. For example, if a neighborhood is bounded by four arterial roads, but with no internal streets, according to the model a crime event that originates from within the neighborhood (i.e.,

the offender lives inside the neighborhood) could take any of the four arterial roads to leave the network. In reality, the offender will probably take a particular route rather than necessarily the arterial that is closest to the offender's address. This can be handled, but it requires additional coding.[4]

As an example, for Baltimore County and the City of Baltimore, Figure 26.7 shows the 49,015 segments in the TIGER representation of these two jurisdictions while Figure 26.8 shows the 11,045 segments that are used by the Baltimore Metropolitan Council in their travel demand modeling.[5] Further, since most of the streets in the modeling representation are two-way streets, in effect, there are only about 5,000-6,000 actual streets. In other words, the TIGER network is 4.4 times larger than the modeling network. This makes calculation a lot slower than with a simplified network.[6] As we shall see in Chapter 30, the accuracy of a network is essential for a more realistic modeling of actual travel routes by offenders.

Bus Network

A bus network, on the other hand, is a specialized road network that follows the actual routes used by buses. The general road network is useful for modeling driving, walking and bicycle trips. But, it cannot be used for bus trips. The reason is simply that buses don't use every street but only the larger arterial roads. Further, travel along many bus routes is variable. That is, a full route might be used during the peak rush hours, but a shortened route might be used during the off-peak hours. Similarly, the frequency of buses (what is called *headway* by transit agencies) varies by time of day; again, in the rush hours, buses are more frequent (though slower) than during the off-peak hours.

A bus network, therefore, is essential for modeling bus trips during both the mode split stage (when trips between zones are split into separate travel modes) and during the actual network assignment.

---

4 For example, transportation modelers often put in *centroid connectors*. These are pseudo-segments that connect a zone centroid with an arterial. It is possible to add pseudo-roads to the modeling network to force travel to follow a particular route. But, it does take a lot of editing to do this.

5 The modeling network was obtained from the Baltimore Metropolitan Council and, with their permission, is illustrated here.

6 As an example of the efficiency of a modeling network compared to a TIGER network, the network assignment routine took six times longer to run with the TIGER network for Baltimore City and Baltimore County than with the modeling network. See Chapter 30 on network assignment for more information about the rules for network travel.

Figure 26.7:

TIGER Street Network
49,015 Road Segments

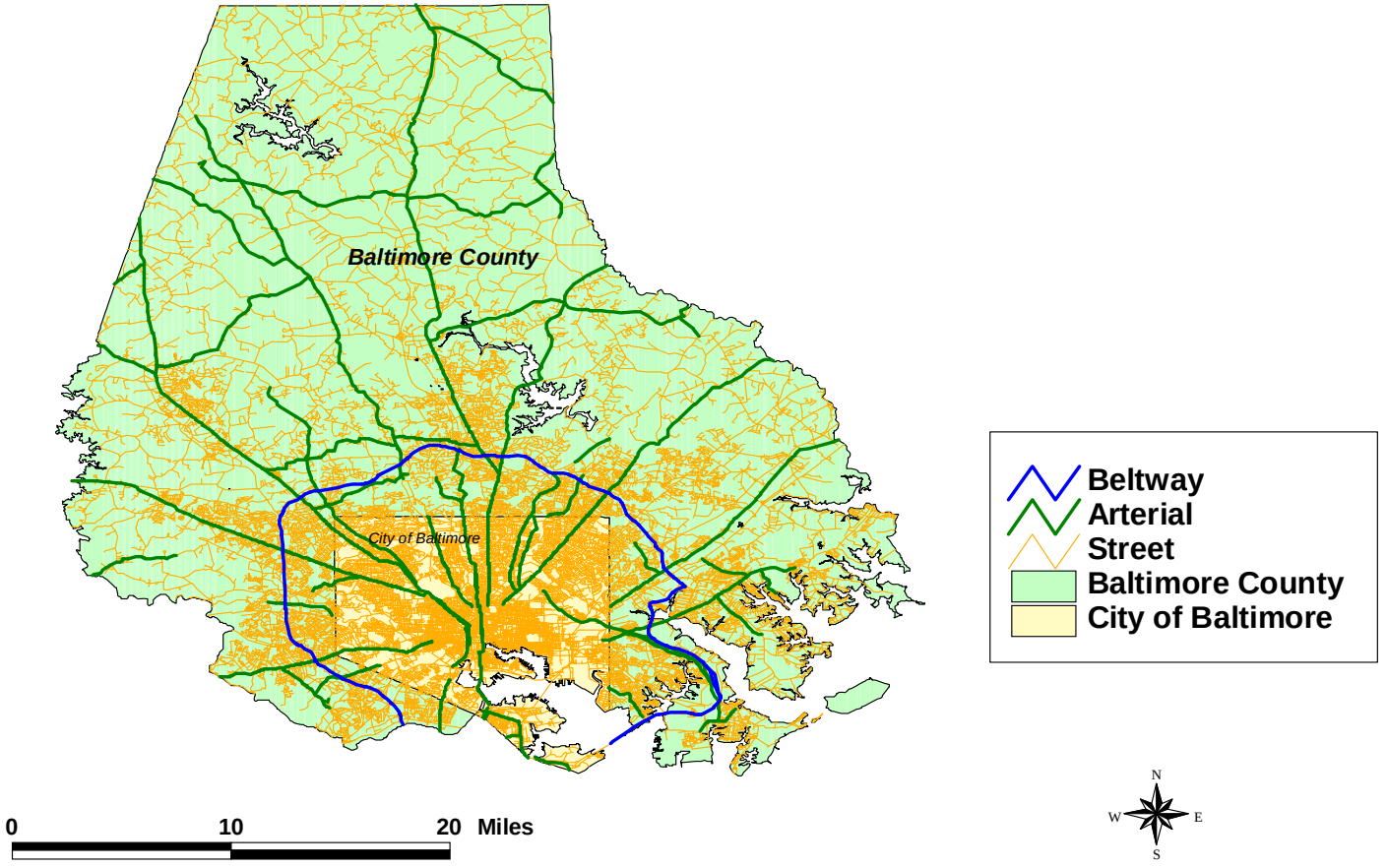
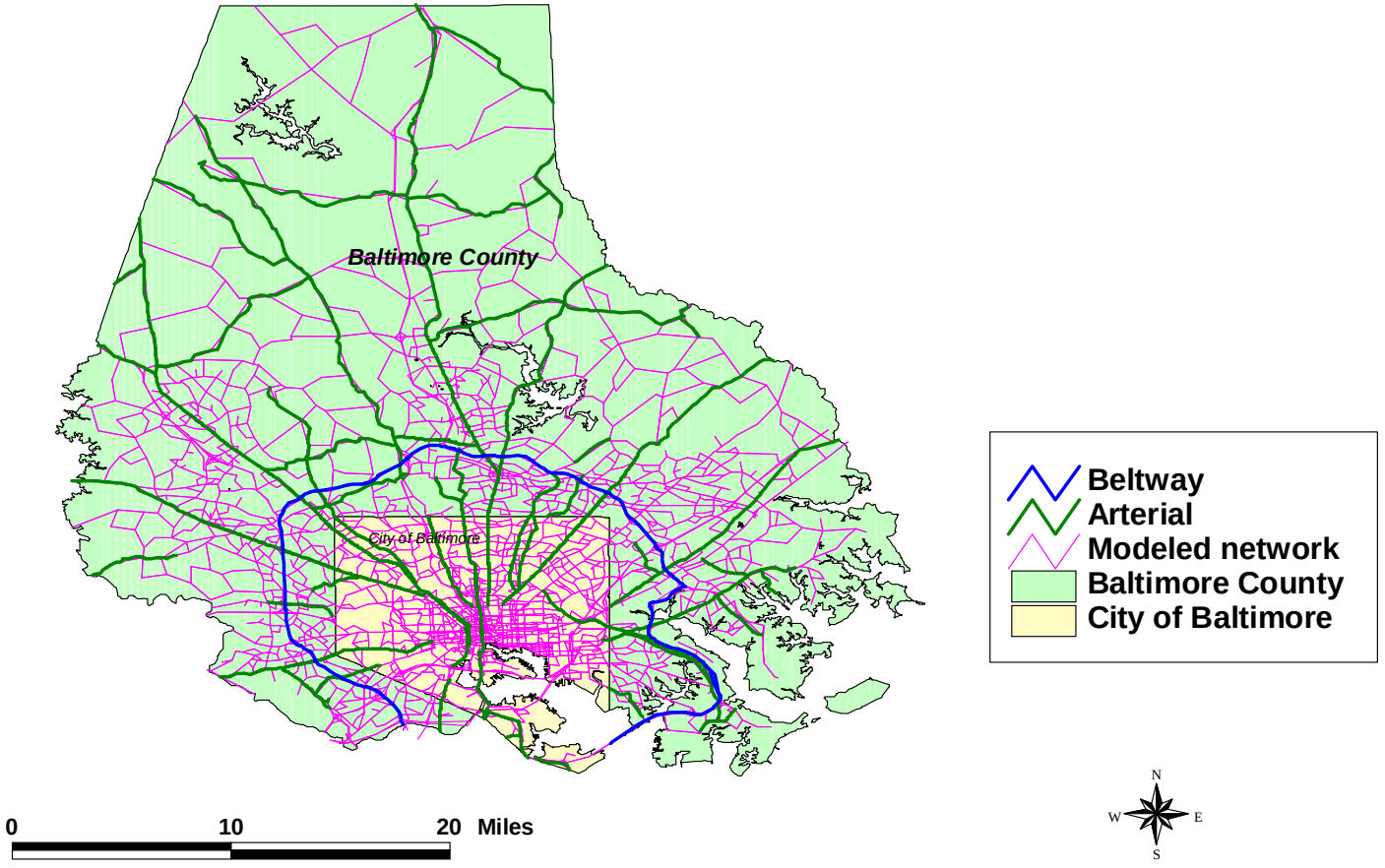


Figure 26.8:

Modeled Street Network
11,045 Road Segments



There are two components of a bus network that are required in the network, one of which is essential and the other is more optional. The first is a representation of the segments used in a bus network. Essentially, this is a network that shows where the buses travel. Bus travel can only occur along this network. As with road travel, the bus network can be represented either as a bi-directional or as a single-directional network though, again, most transportation modelers and transit agencies represent bus routes as single directions.

The second component is the location where access to the buses is allowed (i.e., the bus stops). Without explicitly indicating where there are loading and unloading points, a network routine would simply find the shortest distance from the origin to the bus route and 'add' the trip at that location. In practice, for most transit agencies, the degree of error in allowing direct access anywhere on the route is small since most bus routes stop very frequently (every couple of blocks). Thus, it may not be that important to actually code the bus stops since the amount of modeling error will be insignificant. However, for express buses and for those routes where there is a sizeable distance between bus stops, it is important to code the actual bus stops. In Chapter 30, there is a more extensive discussion of coding bus routes. Figure 26.9 illustrates the bus network for Baltimore County and Baltimore City.

Train network

In those metropolitan areas that have intra-urban train travel, it is important to also obtain a rail network. An offender cannot travel on a train except by using the existing rail system. Further, unlike the bus network, it is impossible to 'enter' the train except at explicit station locations. Thus, it is critical to obtain both the network and the station locations. Figure 26.10 illustrates the intra-urban rail system in Baltimore County and Baltimore City.

Where to Obtain Network Data?

There are many more choices in obtaining network data than with socioeconomic or land use data. Road networks can be obtained from the U.S. Census Bureau (for the TIGER system) or from vendors who improve on the TIGER system. For a modeling network, however, about the only choice is the Metropolitan Planning Organization (MPO). Since MPO's model regional travel on a continuous basis, most agencies in a metropolitan area will defer to them for that activity. Transit networks can also be obtained from MPOs though the transit agencies will have their own networks that are usually more comprehensive than those of the MPO. As with all data, the MPO might charge for the data set, though policies vary widely.

Figure 26.9:

Baltimore Bus Network
Bus Routes

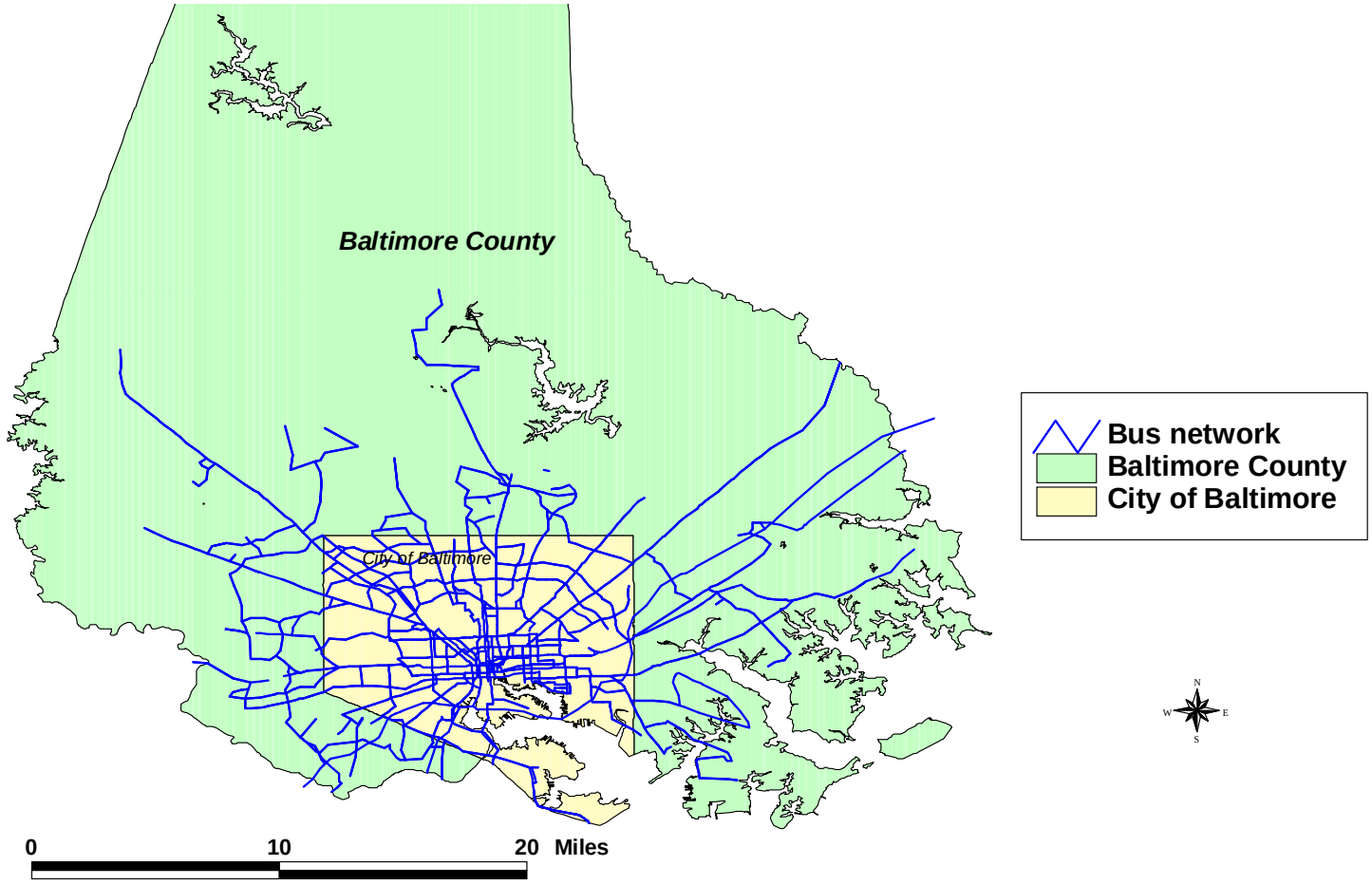
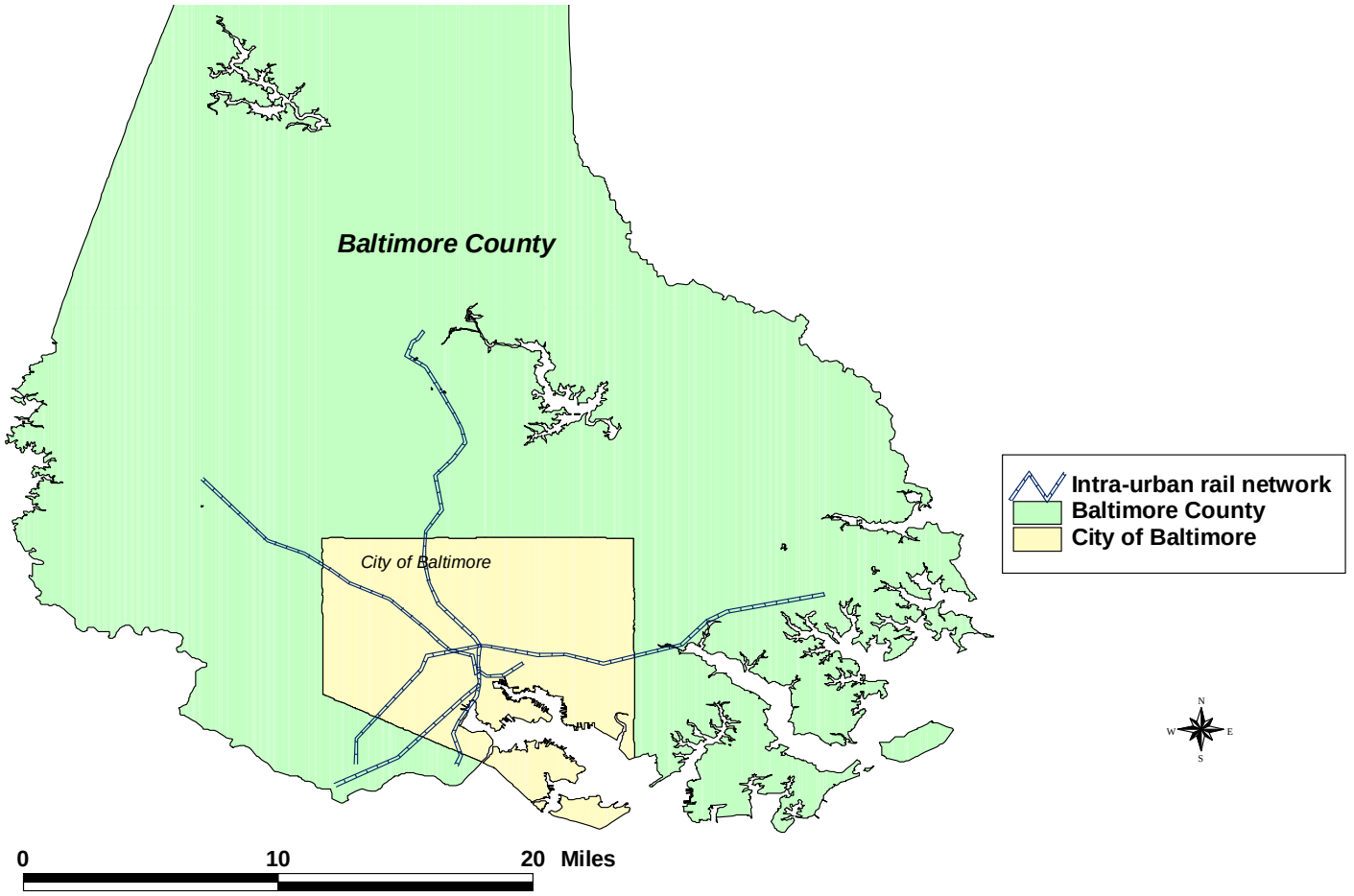


Figure 26.10:
Baltimore Intra-Urban Rail Network



Conclusion

In summary, a quite extensive collection of data is needed to run the crime travel demand model. Crime data, socioeconomic data, land use data, policy intervention scenarios, and network data must be obtained and prepared prior to running the models. Further, in practice, a lot of editing and 'cleaning' of data will be required during the modeling phase in order to improve the predictions.

Nevertheless, once the data are obtained, the model can be developed quite quickly. In the next chapter, we will examine the first stage of the crime travel demand model - trip generation.

References

AMPO (2012). *AMPO: Highlights & What's New*. Association of Metropolitan Planning Organizations: Washington, DC. http://www.ampo.org/. Accessed May 7, 2012.

Anselin, Luc (1995). Local indicators of spatial association - LISA. *Geographical Analysis*. 27, No. 2 (April), 93-115.

Bursik, R. J., Jr. & Grasmick, H. G. (1993). Economic deprivation and neighborhood crime rates, 1960-1980. *Law and Society Review*, 27, 263-268.

Citro, C. F. & Michael, R. T. (eds) (1995). *Measuring Poverty : A New Approach*. Panel on Poverty and Family Assistance, Committee on National Statistics, Commission on Behavioral and Social Sciences and Education, National Research Council: Washington, DC.http://www.census.gov/hhes/www/img/povmeas/ack.pdf. Accessed May 7, 2012.

Hipp, J. R. (2007). Block, Tract, and Levels of Aggregation: Neighborhood Structure and Crime and Disorder as a Case in Point. *American Sociological Review* 72:659-680.

Kitamura, R., Yoshii, T., & Yamamoto, T. (2009). The Expanding Sphere of Travel Behaviour Research: Selected Papers from the 11th International Conference on Travel Behaviour Research. Emerald Group Publishing, Ltd: Bingley, U.K. http://books.google.com/books?id=fFqEnNOWKw8C&pg=PA375&lpg=PA375&dq=microsimulation+of+travel+behavior&source=bl&ots=ArxmN7EIZl&sig=rIUukRBjCApH22qDQ0UXp5dUOGs&hl=en&sa=X&ei=jRmkT_3aFIOi8ATImsS5CQ&ved=0CGQQ6AEwCA#v=onepage&q=microsimulation%20of%20travel%20behavior&f=false. Accessed May 4, 2012.

Langbein, L. I. & Lichtman, A. J. (1978). *Ecological Inference*. Sage University Paper series on Quantitative Applications in the Social Sciences, series no. 07-010. Beverly Hills and London: Sage Publications.

Levine, N. (2011). Spatial variation in motor vehicle crashes by gender in the Houston Metropolitan Area. *Proceedings of the 4th International Conference on Women's Issues in Transportation. Volume II: Technical Papers*, Transportation Research Board: Washington, DC. 12-25. http://onlinepubs.trb.org/onlinepubs/conf/cp46v2.pdf. Accessed May 7, 2012.

Levine, N. (2007), "Crime travel demand and bank robberies: Using CrimeStat III to model bank robbery trips". *Social Science Computer Review*, 25(2), 239-258.

Levine, N. & Canter, P. (2011). Linking origins with destinations for DWI motor vehicle crashes: An application of crime travel demand modeling. *Crime Mapping*, 3, 7-41.

References (continued)

Miller, E. J. & Salvini, P. A. (1999). Activity-based travel behavior modeling in a microsimulation framework. Paper presented at IATBR Conference, Austin, TX. December. http://www.civ.utoronto.ca/sect/traeng/ilute/downloads/conference_papers/miller-salvini_iatbr-97.pdf. Accessed May 4, 2012.

NARC (2012). *Welcome to NARC*. National Association of Regional Councils: Washington, DC. http://www.narc.org/. Accessed May 7, 2012.

Openshaw, S. (1984). *The Modifiable Areal Unit Problem.* Norwich: Geo Books. ISBN 0-86094-134-5.

Ortuzar, J. D. & Willumsen, L. G. (2001). *Modeling Transport* (3rd edition). J. Wiley & Sons: New York.

U.S. Census Bureau (2012). *Commuting (Journey to Work)*. U.S. Census Bureau: Washington, DC. http://www.census.gov/hhes/commuting. Accessed May 7, 2012.

U.S. Census Bureau (2011a). *Summary File 3 (SF3)*. U.S. Census Bureau: Washington, DC. http://www.census.gov/census2000/sumfile3.html. Accessed May 7, 2012.

U.S. Census Bureau (2011b). *Tiger Products*. U.S. Census Bureau: Washington, DC. http://www.census.gov/geo/www/tiger/. Accessed May 8, 2012.

Wikipedia (2012). Modifiable Area Unit Problem. Wikipedia. http://en.wikipedia.org/wiki/Modifiable_areal_unit_problem. Accessed May 7, 2012.

Wooldredge, J. (2002). Examining the (Ir)Relevance of Aggregation Bias for Multilevel Studies of Neighborhoods and Crime with an Example Comparing Census Tracts to Official Neighborhoods in Cincinnati. *Criminology* 40:681-710.

Chapter 27:

Crime Trip Generation

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Chapter 27:
Crime Trip Generation

Background

In this chapter, the theory and mechanics of the trip generation stage will be explained. *Trip generation* is a model of the number of trips that originate and end in each zone for a given jurisdiction. Given a set of *N* destination zones and *M* origin zones (which include all the destination zones and, possibly, zones from adjacent jurisdictions), separate models are produced of the number of crimes originating and ending in each of these zones. That is, a separate model is produced of the number of crimes originating in each of the *M* origin zones, and another model is produced of the number of crimes ending in each of the *N* destination zones. The first is a *crime production* model while the second is a *crime attraction* model.

Two points should be emphasized. First, the models are predictive. That is, the results of the models are a prediction of both the number of crime trips originating in each zone and the number of crime trips ending in each zone (i.e., crimes occurring in a zone). Because the models are predictions, there is always error between the actual number and that predicted. As long as the error is not too large, the models can be useful for both analyzing the correlates of crime as well as being useful for forecasting or for simulating policy interventions.

Second, because the number of crimes attracted to the study jurisdiction will usually be greater than the number of crimes predicted for the origin zones, due primarily to crime trips coming from outside the origin areas, it is necessary to balance the productions and attractions. This is done in two steps. One, an estimate of trips coming from outside the study area (external trips) is added to the predicted origins as an 'external zone'. Two, a statistical adjustment is done in order to ensure that the total number of origins equals the total number of destinations. This is called *balancing* and is essential as an input into the second stage of crime travel demand modeling - trip distribution.

In the following discussion, first, the logic behind trip generation modeling is presented, including the calibration of a model, the addition of external trips in making a model, and the balancing of predicted origins and predicted destinations. Second, the mechanics of conducting the trip generation model within *CrimeStat* is discussed and illustrated with data from Baltimore County.

Modeling Trip Generation

The process of modeling trip generation is fairly well developed, at least with respect to ordinary trips. It proceeds through a series of logical steps that make up the aggregate trip generation model.

Trip Purpose

Trip generation modeling starts with the reasons behind travel. At an individual level, people make trips for a reason - to go to work, to go shopping, to go to a medical appointment, to go for recreation, or, in the case of offenders, to commit a crime. These are called *trip purposes*. Since there are a very large number of trip purposes, usually these are categorized into a few major groupings. In the case of the usual travel demand forecasting, the distinctions are *home-to/from-work* (or home-based work trips), *home-to/from-non-work* (or home-based non-work trips, e.g., shopping), and a *non-home trip* where neither the origin nor the destination are at the traveler's residence location (non-home-based trips).

Since the model has aggregated trips to a zone, the trip purposes are collections of trips from each origin zone to each destination zone. Thus, each zone produces a certain number of home-work trips, home-non-work trips, and non-home trips and each zone attracts a certain number of home-work trips, home-non-work trips, and non-home trips. This is the usual distinction that most transportation modeling organizations make. The trip purposes are documented during a large travel survey that asks individuals to fill out travel diaries for one or two days of travel. In the travel diaries, detailed information about each trip is documented - time of day, destination of trip, purpose of trip, travel modes used in making the trips, accompanying passengers, route taken, and time to complete the trip.

Aggregated Crime Trips

For crime trips, however, these distinctions are not very meaningful. There is very little information on how offenders make trips. One cannot just take a sample of offenders and ask them to complete a travel diary about how, when, and where the trip took place. With arrested offenders, it might be possible to produce such a diary, but both memory problems as well as legal concerns quickly make this an unreliable source of information. Therefore, as indicated in Chapter 26, a decision has been made to reference all trips with respect to the residential home location. All crime trips are analyzed as *home-crime* trips.

However, other distinctions can be made. The most obvious is by type of crime. There are robbery trips, burglary trips, vehicle theft trips, and so forth. Similarly, distinctions can be made by travel time such as afternoon trips or evening trips. However, the sample size will

decrease with greater distinctions. Logically, one can divide a sample into a very large number of important distinctions (e.g., afternoon burglary trips involving two or more offenders). However, this reduces the sample size and increases the error in estimation, particularly at the trip distribution and subsequent stages.

An important point that distinguishes the aggregate demand types of travel demand models, as is being implemented here, and the newer generation of activity-based trips is that there are no *linked trips* with the aggregate approach (Pribyl & Goulias, 2005). If an offender first steals a car, then uses the car to rob a grocery store followed by a burglary, the aggregate approach models this as three separate trips, rather than as a series of three linked crime trips (which the activity-based models do). This is a deficiency with the aggregate travel demand model. In order to make the aggregate models work, each trip is considered independent of any other trip. While this is not realistic behaviorally, since we know that many crimes are committed in sequence as part of a single journey (or tour), the zonal approach does limit the underlying logic of crime trips. Nevertheless, the aggregate approach can be very useful as long as it implemented consistently. With the current state of activity-based modeling, there is not yet any evidence that they produce more accurate predictions than the cruder, aggregate approach (Culp & Lee, 2005).

Correlates of Crime

Any trip has contextual correlates associated with it. It is well documented that the likelihood of making a trip (crime or otherwise) is not equal across areas of a metropolitan region. There are age and gender correlates of travel, socioeconomic correlates of travel, and land use correlates of travel; the latter are usually associated with trip purposes (e.g., retail areas attract shopping trips).

The trip generation model being implemented in this version of *CrimeStat* is an aggregate model. Thus, the predictors are aggregate, rather than behavioral, in nature, as discussed in Chapter 25. They are correlates of trips, not necessarily the *reasons* for the trips. For example, typically population is the best predictor of trips. Zones with many persons will produce, on average, more crime trips than zones with fewer persons. The observation is not a reason, but is simply a by-product of the size of the zone. Similarly, low-income zones will tend to produce, on average, more crime trips than wealthier zones; again, this is not a reason, but a correlate of the characteristics that might contribute to individual likelihoods for committing crimes.

As mentioned in Chapter 25, there are a number of different variables that could be used for prediction, although population (or a proxy for population, such as households), income or poverty, and land use variables would be the most common (NCHRP, 1998).

Theoretical Relevance of the Variables

In general, the variables that are selected should be empirically stable and theoretically meaningful. That is, they should be stable variables that do not change dramatically from year to year. They should be reliably measured so that an analyst can depend on their values. Finally, they should be meaningful in some ways. That is, they should be plausible enough that both crime analysts and researchers and informed outsiders should agree that the relationship is plausible. The variables either should have been demonstrated to be predictors in earlier research or else to be so correlated with known factors as to be considered meaningful proxies.

Spurious correlates

On the other hand, if a variable is either a correlate of a known predictor or idiosyncratic, then it is liable not be believed. For example, the number of taxis usually correlates with the amount of employment since taxis tend to ply commercial areas for their trade. Adding the number of taxis in a predictive model is liable to produce significant statistical effects in predicting crime destinations. However, few persons are going to believe that this is a real factor since it is understood to be a correlate of a more structural variable.

Idiosyncratic variables are those that appear in unique situations. For example, in some cities, adjacency to a freeway is a correlate of crime origins (e.g., in Baltimore County where low income populations live) whereas in other cities, it is a correlate of crime destinations (e.g., in Houston where there are frontage roads with major commercial strips that attract crimes). The variables may be real predictors. However, the analyst or researcher will have difficulty persuading others to believe in the model, at least until the results can be replicated.

In other words, what is required for the model is a set of reasonable correlates of crime trips that would be plausible and stable over time. It is an ecological model, not a behavioral one.

Social Disorganization Variables

There is a very large literature on the predictors of crime, typically following from the social disorganization literature (for example, Park &Burgess, 1924; Thrasher, 1927; Shaw & McKay, 1942; Newman, 1972; Ehrlich, 1975; Cohen & Felson, 1979; Wilson & Kelling, 1982; Stack, 1984; Messner, 1986; Chiricos, 1987; Kohfeld & Sprague, 1988; Bursik & Grasmick, 1993; Hagan & Peterson, 1994; Fowles & Merva, 1996; Bowers & Hirschfield, 1999 among many other studies). Much of this literature identifies correlates that are associated with crime incidents. Among the factors that have been associated with crime and delinquency at an aggregate geographical level are poverty, low income households, overcrowding, substandard

housing, low education levels, single-parent households, high unemployment, minority and immigrant populations.[1]

Statistical Problems with Predictor Variables

Multicollinearity among the independent variables

There are two statistical problems associated with using these variables as predictors. The first is the high degree of overlap between the variables. Zones that have high poverty levels typically also have low household income levels, higher population densities, substandard housing, a high percentage of renters, and higher proportion of minority and immigrant populations. In a regression model, this overlap causes a condition known as *multicollinearity*. Essentially, the independent variables correlate so highly among themselves that they produce ambiguous, and sometimes strange, results in a regression model. For example, if two independent variables are highly correlated, frequently one will have a positive coefficient with the dependent variable while the other will have a negative coefficient; conversely, they sometimes can cancel each other out. Chapter 17 discussed multicollinearity and provided an example that showed correlated independent variables can cancel each other out. Thus, in spite of the correlates with crime levels, in a model it is usually best to eliminate *co-linear* variables. The result is that simple variables usually end up being the most straightforward to use (population, median household income) with many of the subtle, but theoretically relevant, variables typically dropping out of the equation.

Failure to distinguish origins from destinations

Second, in much of this literature, however, there is not a clear distinction between origin predictors and destination predictors. That is, in most cases, the correlates of crimes were identified but it is often unclear whether these correlates are associated with the neighborhoods of the offenders (origins) or the locations where the crimes occur (destinations). This can result in a set of vague correlates without clear direction about whether the variables are associated with producing or attracting conditions. In fact, in much of the early literature on social disorganization, it was implicitly assumed that crimes are produced in the neighborhoods where the offenders lived, a linkage that is increasingly becoming disconnected. For modeling crime trips, however, it is essential that the predictors of origins be kept separate from the predictors of destinations.

---

[1] Note that a correlation at an aggregate level does not necessarily imply a correlation at the individual level. As has been noted frequently, the vast majority of people do not commit serious crimes and that most crimes are committed by a small proportion of the population (Ratcliffe, 2008).

Accuracy and Reliability

A trip generation model should be accurate and reliable. *Accuracy* means that the model should replicate as closely as possible the actual number of trips originating or ending in zones and that there should be no bias (which is a systematic under- or over-estimating of trips). *Reliability* means that the amount of error is minimized.

These criteria have two implications which are somewhat at odds. First, we have to choose models that replicate as closely as possible the number of trips originating or ending in a zone. In general, this would be a model that had the highest overall predictability. But, second, we have to choose models that minimize total prediction errors. This allows a model to replicate the number of trips for as many zones as possible. The two criteria are somewhat contradictory because crime trips are highly skewed. That is, a handful of zones will have a lot of crimes originating or ending in them while most zones will have few or no crimes. The zones with the most crimes will have a disproportionate impact on the final model. Thus, a model that obtains as high a prediction as possible (i.e., highest log-likelihood or R^2) may actually only predict accurately for a few zones and may be very wrong for the majority.

The strategy, therefore, is to obtain a model that balances high predictability but by keeping the total prediction error low.

Count Model

Another element of the model is that the trip generation model is for *counts* (or volumes), not for rates. The model predicts the number of crimes originating in each origin zone and the number of crimes occurring in each destination zone. The model could be constructed to predict rates, but normally it is not done. For most travel demand modeling, as mentioned in Chapter 25, the model predicts the *number* of trips originating or ending in a zone. Thus, there is a *crime production* model that predicts the number of crimes originating in each zone and a *crime attraction* model that predicts the number crimes

Approaches Toward Trip Generation Modeling

Trip Tables

There are two classic approaches to trip generation modeling. The first uses a *trip table* (sometimes called a cross-classification table or a category analysis). A trip table is a cross-classification matrix. Several predictive variables are divided into categories (e.g., three level of household income; four levels of vehicle ownership; three levels of population density) and a mean number of trips is estimated for each cell, usually from a survey. For example, a survey of

household income might show the relationship between household income and the number of trips taken by individuals of the households. Based on a sample, estimates of the *average number of trips per person* can be obtained for each income level (e.g., 3.4 trips per day for persons from low income households; 4.5 trips per day for persons from median income households; 6.7 trips per day for persons from high income households). These variables are further subdivided into two-way or three-way cross-tabulation tables (e.g., low income and medium vehicle ownership; low income and high vehicle ownership). Table 27.1 illustrates a *possible* trip table model involving two variables. In practice, three or four variables are used.

The main reason that trip tables are used in a trip generation model is because of the non-linear nature of trips. Predictive variables are usually not linear in their effects on the number of trips. Thus, unless a sophisticated non-linear model is used, sizeable error can be introduced in a prediction. It is usually safer to use a trip table approach (Ortuzar & Willumsen, 2001). There are some major handbooks on the topic (Henscher & Button, 2002; ITE, 2003). In fact, the Institute of Transportation Engineers publishes a large handbook that gives extensive trip production and trip attraction tables by detailed land uses (ITE, 2003). These tables are often used in formal environmental review processes for site analysis and are frequently accepted by courts in litigation. They are not without their problems, however, and there have been numerous critiques of the tables (Shoup, 2002; NCHRP, 1998). They also cannot be used in a travel demand model and will produce erroneous results.

The problem for crime analysis, however, is that it is impossible to obtain these data. One cannot ask a sample of offenders how many crimes they undertake each day in order to estimate the mean expectations for a table. Thus, one has to adopt a more indirect approach in modeling crime productions and attractions.

A second problem with the trip table approach is its use of zonal data. While it could be applied to zonal data (e.g., using median household income and average vehicle ownership in Table 27.1 instead of individual household income and vehicle ownership), this type of approach is prone to ecological inference errors and could be very wrong (Freedman, 1999; Langbein & Lichtman, 1978). There is no guarantee that the splitting of two aggregate variables (essentially, the cross-product of their marginal probabilities) will produce an accurate trip estimate; often, such an approach leads to very wrong results.

Further, such an approach requires interpretation and some degree of arbitrariness. For example, how does one subdivide median household income? One person might interpret it slightly differently than another; unlike simple numerical counts (e.g., 0 vehicle ownership;

Table 27.1:
Illustration of Possible Trip Table Approach to Trip Generation
Average Trips per Adult, Age 16+

		Household income		
		Low	Medium	High
Vehicle Ownership	0-1	3.2	4.6	6.7
	2+	5.4	7.8	8.1

1 vehicle ownership; 2 vehicle ownership), there is too much variability in categorizing variables at the zonal level.[2]

Linear/OLS Regression Modeling

The second approach is to use a *regression* framework. In this approach, the number of crimes either originating or ending in each zone is estimated from zone characteristics using a regression model. This can be written in a generalized linear model ('link' function) form (see Chapter 16):

$$f(Y_i) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots . \beta_K X_K + \epsilon \qquad (27.1)$$

This equation says that some function of the mean number of crimes, $f(Y_i)$, either originating or ending in zone I, is a linear function of a number of independent variables, X_1, X_2, X_3,....X_K for these zones; there are *K* independent variables plus a possible constant. There is also an error term which represents the discrepancy between the actual observation and what the model predicts. This is sometimes called *residual error* since it is the difference between the observed and predicted values (O_i - Y_i). The function is unspecified and can be non-linear.[3]

The traditional approach to regression modeling assumed that the independent variables are linear in their effect on the dependent variable. Thus,

---

[2] There is also subjectivity in subdividing variables at an individual level. For example, household income levels can be subdivided in different ways. However, with aggregate data, all variables have to be subdivided arbitrarily whereas with individual level data, typically only income is done this way.

[3] Some statisticians often refer to the number of *parameters* that have to be estimated in an equation, not just the number of independent variables. In most regression models, for example, there are K+1 parameters that are estimated - coefficients for the K independent variables and a constant term. In this text, K refers to the number of independent variables, not estimated parameters.

$$Y_i = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots . \beta_K X_K + \epsilon \quad (27.2)$$

In this model, there are K independent variables and one constant term (β_0, sometimes called α) that needs to be estimated. For each zone, *i*, each of the independent variables has a weight associated with it (the coefficient, β). The product of the value of the independent variable times its weight represents its *effect*. The individual effects of each of the K independent variables are summed to produce an overall estimate of the dependent variable, Y.

The method for estimating this equation usually minimizes the sum of the squares of the residual errors. Hence, the procedure is called *Ordinary Least Squares* (or OLS). If the equation is correctly specified (i.e., the dependent variable is normally distributed and all relevant variables have been included), the error term, ε, will be normally distributed with a mean of 0 and a constant variance, σ^2.

Problems with OLS Regression Modeling

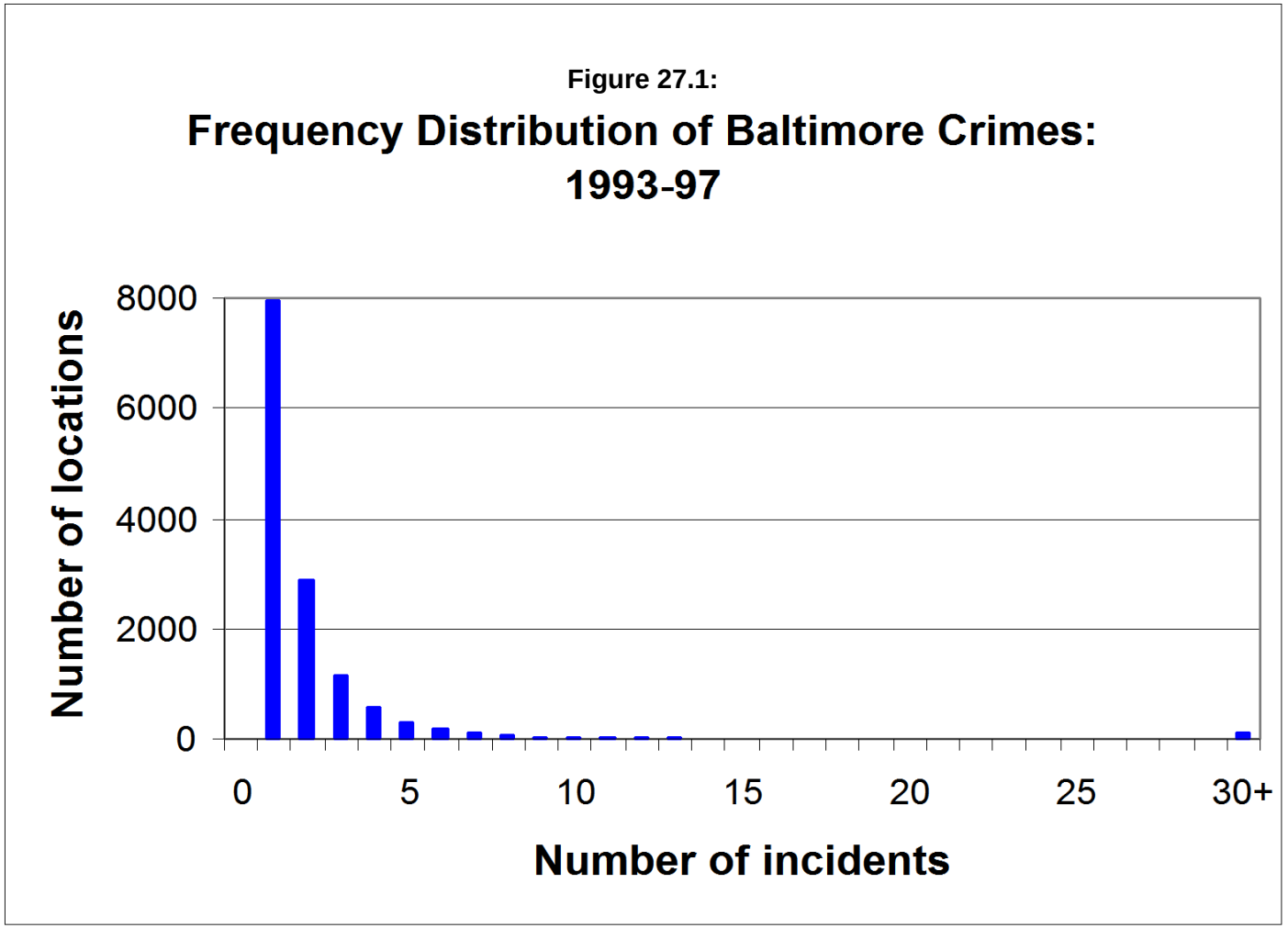
However, there are a number of major problems associated with OLS regression modeling. These were discussed in Chapter 15 (15.16-15.19). To repeat, there are five major problems with the OLS model.

Skewness of crime events

First, crime events are extremely statistically skewed. Some locations have a much higher likelihood of a crime event (either an origin or a destination) than others. Figure 27.1 below shows the number of crimes from 1993 to 1997 in Baltimore County that occurred at each location. That is, the graph shows the number of incidents that occurred at every location, plotted in decreasing order of frequency. Thus, there were 7,965 locations where only one crime occurred between 1993 and 1997. There were 2,878 locations where two crimes occurred in that period. There were 1,138 locations where three crimes occurred in that period. At the other end of the spectrum, there were 332 locations that had 10 or more crimes during the period and there were 97 locations that had 30 or more crimes occur. If we add to this the very large number of locations where no crimes occurred, the unequal likelihoods of crime by location is even more dramatic. In other words, the data are highly skewed with respect to the frequency of crimes. Most locations either had no crimes occur or very few, while a few locations had many crimes occur.

Aggregating crimes into zones tends to reduce *some* of the skewness. For example, grouping the crimes by origin traffic analysis zone (TAZ) reduced it a little bit. Nineteen of the 525 origin zones in Baltimore County and Baltimore City did not have any crimes occur in them while 15 zones had only one crime occur. Six zones had two crimes originate from them while 8

Figure 27.1:


Frequency Distribution of Baltimore Crimes: 1993-97
Number of locations
8000
6000
4000
2000
0
0
5
10
15
20
25
30+
Number of incidents

zones had three crimes originate from them. At the other end, 1 zone had 738 crimes originate from it and another zone had 533 originate from it. Of the 525 origin zones, 155 had 100 or more crime events. Similar results are found for the destination zones. Figure 27.2 graphs the distribution of origins and destinations by TAZ's in bins of 50 incidents each.

Skewness in the dependent variable usually makes the final model biased and unreliable. Particularly if the skewness is positive (i.e., a handful of cases have very large values), the resulting regression coefficients will reflect the cases with the highest values rather than represent all the cases with approximately equal weights. These so-called 'outliers' can overwhelm a regression equation. In an extreme case, a very large outlier may totally determine the model.[4]

Skewness makes prediction difficult. The OLS model assumes that each independent variable contributes to the dependent variable at an arithmetic rate; there is a constant slope such that a one unit change in the independent variable is associated with a constant change in the dependent variable. With skewness, on the other hand, such a relationship will not be found. Large changes in the independent variable will be necessary to produce small changes in the dependent variable, but the effect is not constant. In other words, the OLS model typically cannot explain the non-linear changes in the dependent variable.[5]

Negative predictions

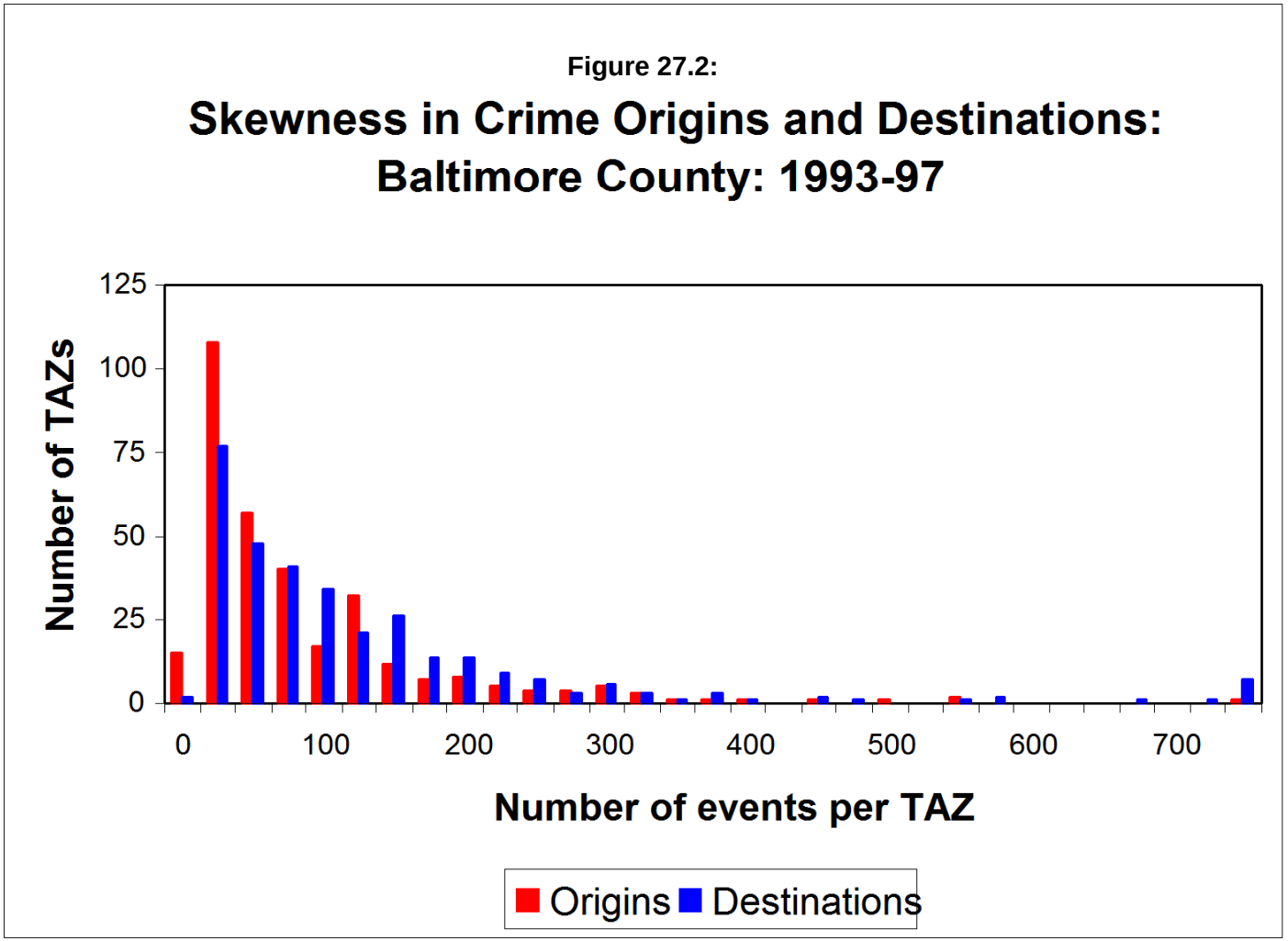
A second problem with OLS is that it can have negative predictions. With a count variable, such as the number of crimes originating or ending in a zone, the minimum number is zero. That is, the count variable is always *positive*, being bounded by 0 on the lower limit and some large number on the upper limit. The OLS model, on the other hand, can produce negative predicted values since it is additive in the independent variables. This clearly is illogical and is a major problem with data that are skewed. If the most common value is close to zero, it is very possible for an OLS model to predict a negative count.

---

4 For example, an experiment with 100 cases was created with a progressing dependent variable and a **random** independent variable (i.e., the independent variable had its value selected randomly). The dependent variable progressed from 1 to 100. For the first 99 cases, the independent variable took values from 0.12 to 9.9, randomly assigned. The correlation between these two variables for the first 49 cases was 0.04. However, for the 100th case, the independent variable was given a value of 100. The correlation between the two variables now shot up to 0.17. Even though the F-test for this was not significant, it represented a sizeable jump. Replacing one other independent value with a 50 caused the correlation to jump to 0.23, which was statistically significant. In other words, two outliers caused a random series to appear significant!

5 It is possible to transform the independent variable into a non-linear predictor, for example by taking the log of the independent variable or raising it to some power (e.g., X^2). However, this will not solve the other problems associated with OLS, namely negative and non-summative predictions.

Figure 27.2:


Skewness in Crime Origins and Destinations:
Baltimore County: 1993-97
Number of TAZs
125
100
75
50
25
0
0
100
200
300
400
500
600
700
Number of events per TAZ
Origins
Destinations

Non-consistent summation

A third problem with OLS models is that the sum of the input data values do not necessarily equal the sum of the predicted values. Since the estimate of the constant and coefficients is obtained by minimizing the sum of the squared residual errors, there is no balancing mechanism to require that they add up to the same as the input values. For a trip generation model in which the number of predicted origins must equal the number of predicted destinations (after adding in the number of predicted external trips), this can be a big problem. In calibrating the model, adjustments can be made to the constant term to force the sum of the predicted values to be equal to the sum of the input values. But in applying that constant and coefficients to another data set, there is no guarantee that the consistency of summation will hold. In other words, the OLS method cannot guarantee a consistent set of predicted values.

Non-linear effects

A fourth problem with the OLS model is that it assumes the independent variables are linear in their effect. If the dependent variable was normal or relatively balanced, then a linear model might be appropriate. But, when the dependent variable is highly skewed, as is seen with these data, typically the additive effects of each component cannot usually account for the non-linearity. Independent variables have to be transformed to account for the non-linearity and the result is often a complex equation with non-intuitive relationships.[6] It is far better to use a non-linear model for a highly skewed dependent variable.

Uneven residual errors

The final problem with an OLS model and a skewed dependent variable is that the model tends to over- or under-predict the correct values, but rarely comes up with the correct estimate. With skewed data, typically an OLS equation produces non-constant residual errors. That is, one of the major assumptions of the OLS model is that all relevant variables have been included. If that is the case, then the errors in prediction (the residual errors - the difference between the observed and predicted values) should be uncorrelated with the predicted value of the dependent variable. Violation of this condition is called *heteroscedasticity* because it indicates that the residual variance is not constant. The most common type is an increase in the residual errors with

---

[6] For example, to account for a skewed dependent variable, one or more of the independent variables have to be transformed with a non-linear operator (e.g., log or exponential term). When more than one independent variable is non-linear in an equation, the model is no longer easily understood. It may end up making reasonable predictions for the dependent variable, but it is not intuitive and not easily explained to non-specialists. It is possible to transform the independent variable into a non-linear predictor, for example by taking the log of the independent variable or raising it to some power (e.g., X^2). However, this will not solve the other problems associated with OLS, namely negative and non-summative predictions.

higher values of the predicted dependent variable. That is, the residual errors are greater at the higher values of the predicted dependent variable than at lower values (Draper & Smith, 1981, 147).

A highly skewed distribution tends to encourage this. Because the least squares procedure minimizes the sum of the squared residuals, the regression line balances the lower residuals with the higher residuals. The result is a regression line that neither fits the low values or the high values. For example, motor vehicle crashes tend to concentrate at a few locations (crash hot spots). In estimating the relationship between traffic volume and crashes, the hot spots tend to unduly influence the regression line. The result is a line that neither fits the number of expected crashes at most locations (which is low) nor the number of expected crashes at the hot spot locations (which are high). The line ends up over-estimating the number of crashes for most locations and under-estimating the number of crashes at the hot spot locations.

Poisson Regression Modeling

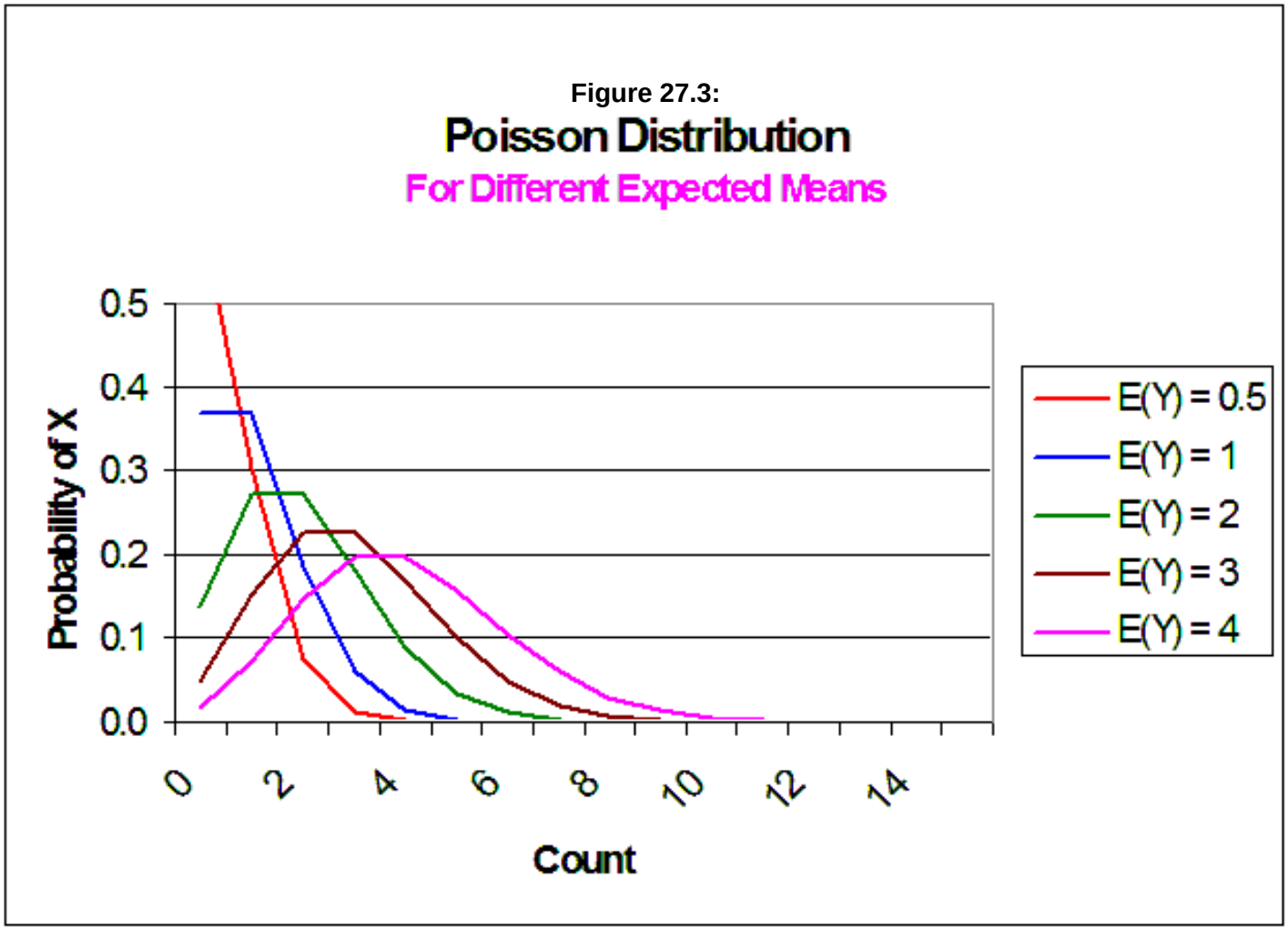
Poisson regression is a non-linear modeling method that overcomes some of the problems of OLS regression. It is particularly suited to count data (Cameron & Trivedi, 1998). In the model, the number of events is modeled as a Poisson random variable:

$$E(Y_i) = \frac{e^{-\lambda}\lambda^{Y_i}}{Y_i!} \qquad (27.3)$$

where Y_i is the count for one group or class, i, λ is the mean count over all groups, and **e** is the base of the natural logarithm. The distribution has a single parameter, λ, which is both the mean and the variance of the function.

The 'law of rare events' assumes that the total number of events will approximate a Poisson distribution *if* an event occurs in any of a large number of trials but the probability of occurrence in any given trial is small (Cameron & Trivedi, 1998). Thus, the Poisson distribution is very appropriate for the analysis of rare events such as crime incidents (or motor vehicle crashes or rare diseases or any other rare event). The Poisson model is not particularly good if the probability of an event is more balanced; for that, the normal distribution is a better model as the sampling distribution will approximate normality with increasing sample size. Figure 27.3 illustrates the Poisson distribution for different expected means.

Figure 27.3:


Poisson Distribution
For Different Expected Means
Probability of X
0.5
0.4
0.3
0.2
0.1
0.0
0
2
4
6
8
10
12
14
Count
E(Y) = 0.5
E(Y) = 1
E(Y) = 2
E(Y) = 3
E(Y) = 4

The mean can, in turn, be modeled as a function of some other variables (the independent variables). Given a set of observations on dependent variables, X_{ki} (X_1, X_2, X_3,...,X_K), the *conditional mean* of Y_i can be specified as an exponential function of the X's:

$$E(y_i \mid \mathbf{x}_i) = \lambda_i = e^{\mathbf{x}_i^T \boldsymbol{\beta}} \tag{27.4}$$

where X_{ki} is a set of independent variables, β is a set of coefficients, and **e** is the base of the natural logarithm.. Now, the conditional mean (the mean controlling for the effects of the independent variables) is non-linear. Equation 27.4 is sometimes written as:

$$Ln(\lambda_i) = X_{ki}\beta \tag{27.5}$$

and is known as the *loglinear* model. In more familiar notation, this is written as:

$$Ln\big((\lambda_i) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots . + \beta_K X_K = \beta_0 + \textstyle\sum_{K=1}^{K} (\beta_K X_K\big) \tag{27.6}$$

That is, the natural log of the mean is a function of K random variables and a constant.

Note, that in this formulation, there is not a random error term. The data are assumed to reflect the Poisson model. There can be residual errors, but these are assumed to reflect an incomplete specification (i.e., not including all the relevant variables. Also, since the variance equals the mean, it is expected that the residual errors should increase with the conditional mean. That is, there is inherent heteroscedasticity (Cameron & Trivedi, 1998). This is very different than an OLS where the residual errors are expected to be constant.

The model is estimated using a maximum likelihood procedure, typically the Newton-Raphson method. In Appendix B, Luc Anselin presents a more formal treatment of both the OLS and Poisson regression models

Advantages of the Poisson Regression Model

The Poisson model overcomes some of the problems of the OLS model. First, the Poisson model has a minimum value of 0. It will not predict negative values. This makes it ideal for a distribution in which the mean or the most typical value is close to 0. Second, the Poisson is a fundamentally skewed model; that is, it is non-linear with a long 'right tail'. Again, this model is appropriate for counts of rare events, such as crime incidents.

Third, because the Poisson model is estimated by either maximum likelihood or Markov Chain Monte Carlo (MCMC; see chapters 16 and 17), the estimates are adapted to the actual data.

In practice, this means that the sum of the predicted values is virtually identical to the sum of the input values, with the exception of very slight rounding off error. In the subsequent balancing of the predicted origins and the predicted destinations, this leads to a more stable estimate since the only difference between the predicted origins and predicted destinations is the number of trips that come from outside the study area (external trips). Since the external trips are added to the predicted origins, the balancing operation is less prone to adjustment error.

Fourth, compared to the OLS model, the Poisson model generally gives a better estimate of the number of crimes for each zone. The problem of over- or under-estimating the number of incidents for most zones with the OLS model is usually lessened with the Poisson. When the residual errors are calculated, generally the Poisson has a lower total error than the OLS.

In short, the Poisson model has some desirable statistical properties that make it very useful for predicting crime incidents (origins or destinations).

Problems with the Poisson Regression Model

On the other hand, the Poisson model is not perfect. The primary problem is that count data are usually *over-dispersed* but occasionally can be *under-dispersed*.

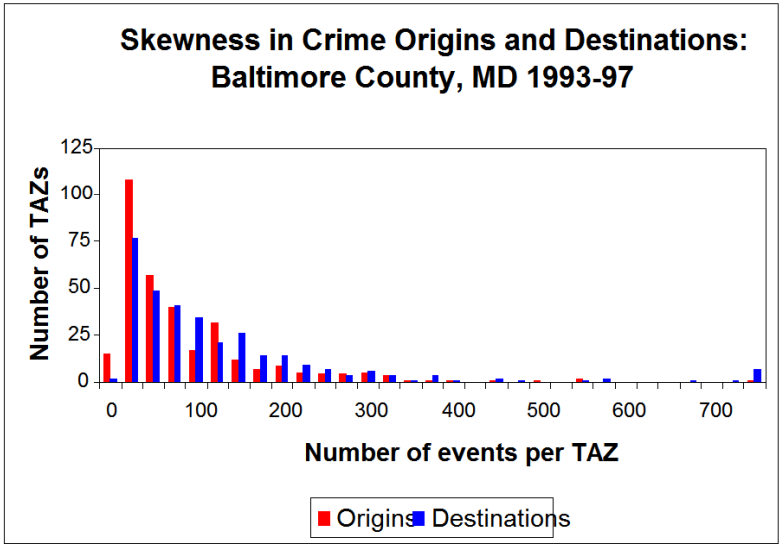
Over-dispersion in residual errors

In the Poisson distribution, the mean equals the variance. In a Poisson regression model, the mathematical function, therefore, equates the conditional mean (the mean controlling for all the predictor variables) with the conditional variance. However, most real data are over-dispersed; the variance is generally greater than the mean. Figure 27.4 shows the distribution of Baltimore County and Baltimore City crime origins and Baltimore County crime destinations by TAZ (repeat of Figure 27.2) and also indicates the variance-to-mean ratio of each variable. For the origin distribution, the ratio of the variance to the mean is 14.7; that is, the variance is 14.7 times that of the mean! For the destination distribution, the ratio is 401.5!

In other words, the variance is many times greater than the mean. Most real-world count data are similar to this; the variance will usually be a lot greater than the mean. What this means in practice is that the residual errors - the difference between the observed and predicted values for each zone, will be greater than what is expected. The Poisson model calculates a standard error as if the variance equals the mean. Thus, the standard error will be underestimated using a Poisson model and, therefore, the significance tests (the coefficient divided by the standard error) will be greater than it really should be. This would have the effect of identifying variables as being more statistically significant in a model than what they actually should be. In other words, in a Poisson

Figure 27.4:

Over-dispersion



Origins:

Mean = 75.8

Variance = 7848.8

Ratio of variance to mean = 14.7

Destinations:

Mean = 129.1

Variance = 51,849.1

Ratio of variance to mean = 401.5

regression model, we would end up selecting variables that really should not be selected because we think they are statistically significant when, in fact, they are not.

Another problem with the Poisson, which is true for most of the common regression methods, is the lack of a spatial predictor component. For these, the MCMC Poisson models, discussed in Chapter 17 can be used. Also, in the crime travel demand model, spatial interaction is explicitly incorporated during the second stage of the model - trip distribution. Thus, any errors introduced in the first stage - trip generation, are usually compensated for during the second. Nevertheless, the inclusion of a spatial component in a regression model will generally improve the prediction.

Dispersion correction parameter

There are a number of methods for correcting the over-dispersion in a count model. Most of them involve modifying the assumption of the conditional variance equal to the conditional mean. For example, the negative binomial model assumes a Poisson mean but a gamma-distributed variance term (Cameron & Trivedi, 1998, 62-63; Venables & Ripley, 1997, 242-245). That is, there is an unobserved variable that affects the distribution of the count. There are several interpretations of the negative binomial (see Boswell & Patil, 1970) but the most common is to assume that there are mixtures of distinct Poisson distributions that make up the real distribution.

The negative binomial model has a Poisson mean but with a 'longer tail' variance function and is usually preferred for over-dispersed data sets, such as typical with crime data. In Appendix C, Dominique Lord and Byung-Jung Park present a formal treatment of the negative binomial model. Other adjustments that can be made include the Poisson-Lognormal model (which can be estimated in *CrimeStat*; see Chapter 17) and the zero-inflated Poisson model assumes a Poisson function combined with a degenerate function with a probability of 1 for zero counts (Hall, 2000). These generally produce better estimates that the simple Poisson especially if a spatial component is added.

There is a simple linear correction for over-dispersion that frequently works, called the ***NB1*** model (Cameron & Trivedi, 1998, 63-65). The model proceeds in two steps. In the first, the Poisson model is fitted to the data and the degree of over- (or under-) dispersion is estimated. The dispersion parameter is defined as:

$$\Phi = \frac{1}{N-K-1}\sum_{i=1}^{N}\frac{(Y_i-P_i)^2}{P_i} \qquad (27.7)$$

where N is the sample size, K is the number of independent variables, Y_i is the observed number of events that occur in zone i, and P_i is the predicted number of events for zone i. The test is similar

to an average Chi-square in that it takes the square of the residuals (Y_i - P_i) and divides it by the predicted values, and then averages it by the degrees of freedom. The dispersion parameter is a standardized number. A value greater than 1.0 indicates over-dispersion while a value of less than 1 indicates under-dispersion (which is rare, though possible). A value of 0 indicates *equidispersion* (or the variance equals the mean).

In the second step, the Poisson standard error is multiplied by the square root of the dispersion parameter to produce an *adjusted standard error*:

$$SE_{adj} = SE * \sqrt{\Phi} \quad (27.8)$$

The new standard error is then used in the t-test to produce an adjusted t-value. Cameron and Trivedi (1998) have shown that this adjustment produces results that are almost identical to that of the negative binomial, but involving fewer assumptions. Chapter 16 discussed the NB1 model in more depth.

The point is that the Poisson model needs to be adjusted for over-dispersion. CrimeStat provides a number of regression tools for accounting over-dispersion and which can also include a spatial autocorrelation adjustment. Chapters 16 and 17 provide information on these models.

Under-dispersion in residual errors

Occasionally, a data set will be under-dispersed, meaning that the conditional variance is substantially lower than the mean. As a rough approximation, Cameron and Trivedi (1998) suggest that if the raw variance-to-mean ratio is less than 2.0, then most likely the model will show under-dispersion with the conditional mean. If the under-dispersion is slight, then the NB1 model can be used to adjust the standard errors. If it is substantial, however, then other models have to be considered. See Chapter 17 for more details.

Diagnostic Tests

There are a number of diagnostics tests that are used in a regression framework.

Skewness Tests

First, there are tests of skewness in the dependent variable. As mentioned above, the OLS model cannot be applied to data that are highly skewed. If they are skewed, a non-linear model, such as the Poisson, must be used. Therefore, it is essential to evaluate the degree of skewness.

A commonly used measure of skewness is the g statistic (Microsoft, 2012):

$$Skewness\ (g) = \frac{N}{(N-1)(N-2)} \sum_{i=1}^{N} [\frac{(X_i - \overline{X})}{s}]^3 \quad (27.9)$$

where N is the sample size, X_i is observation i, $\bar{X}$ is the mean of X, and s is the sample standard deviation (corrected for degrees of freedom):

$$s = \sqrt{\sum_{i=1}^{N} \frac{(X_i - \bar{X})^2}{N-1}} \quad (27.10)$$

The standard error of skewness (SES) can be approximated by (Tabachnick & Fidell, 1996):

$$SES = \sqrt{\frac{6}{N}} \quad (27.11)$$

An approximate Z-test can be obtained from:

$$Z(g) = \frac{g}{SES} \quad (27.12)$$

Thus, if Z is greater than +1.96 or smaller than -1.96, then the skewness is significant at the $p \leq .05$ level.

As an example, for the data on the origins of crimes by TAZ in Baltimore County, we have:

$$\bar{X} = 75.108 \quad (27.13)$$

$$s = 96.017 \quad (27.14)$$

$$N = 325 \quad (27.15)$$

$$\sum_{i=1}^{N} [\frac{(X_i - \overline{X})}{s}]^3 = 898.31 \quad (27.16)$$

Therefore,

$$g = \frac{325}{324*323} * 898.391 = 2.79 \quad (27.17)$$

$$SES = \sqrt{\frac{6}{325}} = 0.136 \quad (27.18)$$

$$Z(g)=20.51 \quad (27.19)$$

The Z of the g value shows the data are highly skewed as was, of course, already known.

Likelihood Ratio Test

Second, there are tests of the overall model. In a maximum likelihood framework, the first test is of the *log-likelihood* function. A *likelihood* function is the joint density of all the observations, given a value for the parameters, β, and the variance, σ^2. The log-likelihood is the natural log of this product, or the sum of the logs of the individual densities. For the OLS model, the log-likelihood is:

$$L = 1\left(\frac{N}{2}\right)\ln(2\pi) - \left(\frac{N}{2}\right)\ln(\sigma^2) - \left(\frac{\sigma}{2}\right) - 0.5\frac{(Y_i - X_{ki}\beta_k)^2}{\sigma^2} \quad (27.20)$$

where N is the sample size, σ^2 is the variance, Y_i is the observed number of events for zone i, and $X_{ki}\beta_k$ is a series of K independent predictors multiplied by their coefficients.

In the Poisson model, the log-likelihood is:

$$L = \sum_{i=1}^{N}[-\lambda_i + Y_i X_{ki}\beta_k - lnY_i!] \quad (27.21)$$

where λ_i is the conditional mean for zone i, Y_i is the observed number of events for zone i, and Y_i $X_{ki}\beta_k$ is a cross-product of the observed events times the K independent predictors multiplied by their coefficients. As mentioned above, Luc Anselin provides a more detailed discussion of these functions in Appendix B.

Since the maximum likelihood method achieves the model with the highest log-likelihood, the log-likelihood is a negative number. Even though the model with the highest log-likelihood is considered 'best', it is not an intuitive number. Consequently, the *Likelihood Ratio* compares the log-likelihood of the regression model with the log-likelihood that would be obtained if only the mean number of counts was taken. This latter log-likelihood is:

$$L_R = -N\bar{Y} + [\ln(\bar{Y})\sum_{i=1}^{N} Y_i] - \sum_{i=1}^{N} Y_i! \quad (27.22)$$

The Likelihood Ratio test is:

$$LR = 2(L - L_R) \tag{27.23}$$

where L is the model log-likelihood and L_R is the log-likelihood of the mean count. The Likelihood Ratio is twice the difference between log-likelihood values of the regression and mean models respectively. It follows a χ^2 distribution with K degrees of freedom (where K is the number of independent variables).[7]

Adjusted likelihood ratio

The Likelihood Ratio is a more intuitive index since it is a Chi-square test. However, it is prone to the problem of all regression methods of over-fitting - the more independent variables are added to the model, the higher is the Likelihood Ratio. Consequently, there are several methods that adjust for the number of parameters fit. One is the Akaike Information Criterion (AIC) which is defined as:

$$AIC = -2L + 2(K + 1) \tag{27.24}$$

where L is the log-likelihood and K is the number of independent variables. A second one is the Bayesian Information Criterion/Schwartz Criterion (BIC/SC), which is defined as:

$$\text{BIC/SC} = \text{-2L+[(K+1)ln(N)]} \tag{27.25}$$

These two measures penalize the number of parameters added in the model, and reverse the sign of the log-likelihood (L) so that the statistics are more intuitive. The model with the lowest BIC/SC value is 'best'.

R-square Test

The most familiar test of an overall model is the R-square (or R^2) test. This is the percent of the total variance of the dependent variable accounted for by the model. More formally, it is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(Y_i - P_i)^2}{\sum_{i=1}^{N}(Y_i - \bar{Y})^2} \tag{27.26}$$

[7] Note, in Appendix B Luc Anselin uses K for the number of parameters (coefficients + intercept) whereas we use it for the number of independent variables. Readers should be aware of this difference.

where Y_i is the observed number of events for a zone i, P_i is the predicted number of events given a set of K independent variables, and $\bar{Y}$ is the mean number of events across zones. The R^2 is a number from 0 to 1; 0 indicates no predictability while 1 indicates perfect predictability.

R-square for OLS model

For an OLS model, R^2 is a very consistent estimate. It increases in a linear manner with predictability and is, therefore, a good indicator of how effective one model is compared to another. As with all diagnostic tests, the value of the R^2 increases with more independent variables. Consequently, R^2 is usually adjusted for degrees of freedom:

$$R_a^2 = 1 - \frac{\sum_{i=1}^{N} \frac{(Y_i - P_i)^2}{N-K-1}}{\sum_{i=1}^{N} \frac{(Y_i - \bar{Y})^2}{N-1}} \tag{27.27}$$

where N is the sample size and K is the number of independent variables.

R-square for Poisson model

With the Poisson model, however, the R^2 value (whether adjusted or not) is not a good measure of overall fit. While the Poisson R^2 varies from 0 to 1, similar to the OLS, it is not monotonic. That is, the addition of a new variable to an equation often has unpredictable effects; sometimes it will increase substantially and sometimes it will increase only a little independent of how strong is a variable's association with the dependent variable (Miaou, 1996). This inconsistency comes from the decomposition of the total sum of squares:

$$\sum_{i=1}^{N}(Y_i - \bar{Y})^2 = \sum_{i=1}^{N}(Y_i - P_i)^2 + \sum_{i=1}^{N}(P_i - \bar{Y})^2 + 2\sum_{i=1}^{N}(Y_i - P_i)(P_i - \bar{Y}) \tag{27.28}$$

The first term in the equation is the residual sum of squares (or error term) while the second term is the explained sum of squares. In an OLS model, the third term is zero if an intercept is included (Cameron & Trivedi, 1998, 153). Hence, the total sum of squares is broken into two parts - that which is explained and that which is unexplained. However, for the Poisson and other non-linear regression methods, the last term is not zero. Consequently, a test that compares the explained sum of squares to the total sum of squares will not produce consistent results.

Other measures have been proposed, such as the deviance R-square which measures the reduction in the Likelihood Ratio due to the inclusion of predictor variables (Cameron & Windmeijer, 1996). It produces a slightly different R-square, one that is typically higher than the traditional R-square. Nevertheless, it has problems, too. Miaou (1996) argues that there is not a

single R-square index that is perfectly consistent. The AIC, BIC/SC and Deviance statistics (discussed in Chapter 16) are better indicators of goodness of fit.

Dispersion Parameter

Finally, in the Poisson model only, the dispersion parameter indicates the extent to which the variance is different from the mean. This was defined in equation 27.7 above.

Coefficients, Standard Errors, and Significance Tests

The second type of diagnostic test is those for the individual predictors in the model. In both the OLS and Poisson models, there are three tests:

1. The coefficient. This indicates the change in the dependent variable associated with the change in the independent variable. In the case of the OLS, it is a linear term (i.e., the value of the dependent variable is multiplied by the coefficient) while in the Poisson model, the change in the dependent variable is estimated by exponentiating the coefficient (i.e., $e^{\beta X}$).

2. The standard error. Each estimated coefficient in a model accounts for some of the variance in the dependent variable. This variance is the contribution of the particular independent variable to the variance of the dependent variable. The square root of that variance is the *standard error*.

3. The significance level. The ratio of the coefficient to the standard error produces a significance test of the coefficient. In the OLS model, it is a t-test with N-K-1 degrees of freedom whereas in the Poisson model it is an asymptotic t-test, which is effectively a Z-test. The appropriate tables (t-test or standard normal) produce approximate probability levels of a Type I error (the likelihood of falsely rejecting a true null hypothesis of no relationship).

Testing for Multicollinearity

One of the major problems with any regression model, whether OLS or Poisson, is multicollinearity among the independent variables. In theory, each independent variable should be statistically independent of the other independent variables. Thus, the amount of variance for the dependent variable that is accounted for by each independent variable should be a unique contribution. In practice, however, it is rare to obtain completely independent predictive variables. More likely, two or more of the independent variables will be correlated. The effect is that the estimated standard error of a predictor variable is no longer unique since it shares some of the

variance with other independent variables. The greater *communality* of shared variance, the more ambiguous will be the predicted effects. If two variables are highly correlated, it is not clear what contribution each makes towards predicting the dependent variable. In effect, multicollinearity means that variables are measuring the same effect.

Multicollinearity among the independent variables can produce very strange effects in a regression model. Among these effects are: 1) If two independent variables are highly correlated, but one is more correlated with the dependent variable than the other, the stronger one will usually have a correct sign while the weaker one will sometimes get flipped around (e.g., from positive to negative, or the reverse); 2) Two variables can cancel each other out; each coefficient is significant when it alone is included in a model but neither are significant when they are together; 3) One independent variable can inhibit the effect of another correlated independent variable so that the second variable is not significant when combined with the first one; and 4) If two independent variables are virtually perfectly correlated, many regression routines break down because the matrix cannot be inverted.

All these effects indicate that there is non-independence among the independent variables. Aside from producing confusing coefficients, multicollinearity can overstate the amount of predictability in a model. Since every independent variable accounts for some of the variance of the dependent variable, with multicollinearity the overall model will appear to improve when it probably has not.

Tolerance test

A user has to be aware of the problem of multicollinearity and seek to minimize it. The simplest solution is to drop variables that are co-linear with other independent variables already in the equation. A relatively simple test for assessing this is called *tolerance*. Tolerance is defined as *lack of predictability* of each independent variable by the other independent variables, or:

$$Tol_i = 1 - (R_{jk\ldots})^2 \qquad (27.29)$$

where $(R_{jk..})^2$ is the R^2 of an OLS equation where independent variable, i, is predicted by the other independent variables, j, k, l, and so forth. That is, *each* independent variable in turn is regressed against the *other* independent variables in the equation. The R^2 associated with that model is subtracted from 1. The higher the tolerance level, the less a particular independent variable shares its variance with the other independent variables.

Note that the tolerance test uses an OLS model; it assumes the dependent variable in the test (i.e., one of the independent variables) is normally distributed, which may or may not be true. Thus, in a Poisson or other non-linear model, one has to be careful about interpretation based on

the tolerance test. Nevertheless, the test can be a good indicator of whether two variables are collinear. As a rough guideline, a tolerance value of 0.7 or less usually indicates substantial multicollinearity. This level means that there is overlap of 50% or more in the variance of the tested variable with the other independent variables. A more strict and conservative approach uses a tolerance level of 0.8 or less as indicating multicollinearity. This level means that there is overlap of 36% or more in the variance of the tested variable with the other independent variables. An even stricter criterion is to use a tolerance level of 0.9 or less, essentially allowing 18% overlap in the variance of the tested variable with the other independent variables. In general, it is better to have a stricter model that has little multicollinearity. The interpretation of the coefficients will be cleaner and the model will generally be more reliable with other data sets.

Fixed model v. stepwise variable selection

There are several strategies designed to reduce multicollinearity in a model. One is to start with a defined model and eliminate those variables that have a low tolerance. The total model is estimated and the coefficients for each of the variables are estimated at the same time. This is sometimes called a *fixed model*. Then, variables that are co-linear are removed from the equation, and the model is re-run.

Another strategy is to estimate the coefficients a step at a time, a procedure known as *stepwise* regression (Der & Everitt, 2002, 88-89). There are several standard stepwise procedures. In the first procedure, variables are added one at a time (a *forward selection* model). The independent variable having the strongest linear correlation with the dependent variable is added first. Next, the independent variable from the remaining list of independent variables with the highest correlation with the dependent variable, *controlling for* the one variable already in the equation, is added next and the model is re-estimated. In each step, the independent variable with the highest correlation with the dependent variable controlling for the variables already in the equation is added to the model, and the model is re-estimated. This proceeds until either all the independent variables are added to the equation or else a stopping criterion is met. The usual criterion is only variables with a specified significance level are allowed to enter (called a *p-to-enter*).

A *backward elimination* procedure works in reverse. All independent variables are initially added to the equation. The variable with the weakest coefficient (as defined by the significance level) is removed, and the model is re-estimated. Next, the variable with the weakest coefficient in the second model is removed, and the model is re-estimated. This procedure is repeated until either there are no more independent variables left in the model or else a stopping criterion is met. The usual criterion is that all remaining variables pass a specified significance level (called a *p-to-remove*).

There are combinations of these procedures, for example adding variables in a forward selection but then removing any that are no longer significant or using a backward elimination procedure but allowing new variables to enter the model if they suddenly become significant.

There are advantages to each approach. A fixed model allows defined variables to be all included. If either theory or previous research has indicated that a particular combination of variables is important, then the fixed model allows that to be tested. A stepwise procedure might drop one of those variables. On the other hand, a stepwise procedure usually can obtain the same or higher predictability than a fixed procedure (whether predictability is measured by a log-likelihood or an R-square).

Within the stepwise procedures, there are also advantages and disadvantages to each method, though the differences are generally very small. A forward selection procedure adds variables one at a time. Thus, the contribution of each new variable can be seen. On the other hand, a variable that is significant at an early stage could become not significant at a later stage because of the unique combinations of variables. Similarly, a backward elimination procedure will ensure that all variables in the equation meet a specified significance level. But, the contribution of each variable is not easily seen other than through the coefficients. In practice, one usually obtains the same model with either procedure, so the differences are not that critical.

A stepwise procedure will not guarantee that multicollinearity will be removed entirely. However, it is a good procedure for narrowing down the variables to those that are significant. Then, any co-linear variables can be dropped manually and the model re-estimated. In the *CrimeStat* trip generation, both a fixed model and a backward elimination procedure are allowed.

Available Regression Models

CrimeStat has 10 different regression models that can be used for trip generation and which can be estimated with either maximum likelihood (MLE) or Markov Chain Monte Carlo (MCMC):

MLE Normal (OLS)
MCMC Normal
MCMC Normal-spatial autocorrelation component (CAR or SAR)
MLE Poisson
MLE Poisson with Linear Correction (NB1)
MLE Negative Binomial (Poisson-Gamma)
MCMC Poisson-Gamma
MCMC Poisson-Gamma-spatial autocorrelation component (CAR or SAR)
MCMC Poisson-Lognormal
MCMC Poisson-Lognormal-spatial autocorrelation component (CAR or SAR)

Users should consult Chapters 16 and 17 for details of these models. There are other methods for estimating the likely value of a count given a set of independent predictors. Among these are the zero-inflated Poisson (or ZIP; Hall, 2000), the Weibul function, the Cauchy function, and the lognormal function (see NIST 2004 for a list of common non-linear functions). There are also other spatial regression type models that correct for spatial autocorrelation in the dependent variable, such as geographically-weighted regression using a Poisson function (Fotheringham, Brunsdon, & Charlton, 2002). These are not included in this version of CrimeStat.

Adding Special Generators

In a travel demand model, there are *special generators*. These are unique land uses or environments that produce an extra large number of trips. For regular travel demand modeling, stadiums, airports, train stations, large parks, and mega-malls generate more than their share of trips, or at least than what would be predicted by the amount of permanent employment at those locations. They are usually attractors, not producers. In a normal transportation travel demand model, these zones are excluded from the cross-classification and independent estimates are made of them.

For crime trips, there are also special generators. Typically, these are zones that have more crimes being attracted to them than are expected on the basis of the population and employment at those locations. Since we are using a regression model to estimate the productions and attractions, a simple way to model a special generator is to create a simple *dummy* variable. This is a variable where zones with the special generator get a value ‘1' and zones without the special generator get a ‘0'. Essentially, the variable is a cross-classification of the special generator versus every other zone.

One has to be cautious is doing this, however. Typically, special generators are identified by having a greater number of crimes being attracted to a zone than is predicted by the model. In other words, they have a greater positive residual error (observed - predicted) and are ‘outliers’ in the residual error distribution. By adding a variable to explain those cases, the residual error decreases. But, in doing so, we are not really explaining why the zone has more crimes than expected, but simply accounting for it by putting in an empirical variable. In re-running the model, there will be, usually, new outliers that have a greater positive residual error. If this logic is to be repeated, then we would create new special generators for those zones and re-estimate the model. If continued without limits, eventually there would not be a model anymore but just a collection of dummy variables, one for each zone.

Therefore, a user should be cautious in introducing special generators. It is generally alright to introduce a few for the truly exceptional zones. These are zones where it is logical to treat them as special generators and where one would expect continuity over time. In other words, they

should be used if the special generator status is expected to last over time. For example, a stadium or an airport or a train station is liable to remain at its location for many years. A particular shopping mall, on the other hand, may attract crimes at one particular point in time but not necessarily in the future. Unless a mall is so much larger than other malls in the region (a mega-mall), it should not be assigned a special generator status.

Adding External Trips

External trips are, by definition, trips that come from outside the region. They are part of the origin/production model in that these are trips that are not accounted for by the model. There are also trips that originate within the study area, but end outside the area; however, those are usually not modeled since the focus will be on the study area itself. In the usual travel demand framework, external trips are those coming from major corridors into the region. Estimates of the travel on these corridors are obtained by *cordon counts*, counts of vehicles coming into the region and leaving the region (net inflow). Estimates of future growth of those external trips has to based on expectations of future population growth the metropolitan region and in nearby regions.

For crime trips, external trips are defined as trips that originate outside the study area. But they must be estimated by the difference between the total number of crimes occurring in the destination study area and the total originating in the origin zones. That is, of all the crimes occurring in the study area, the origin zones are modeled. Those trips that originate from outside the origin zones are external trips. They must be added to the predicted number of origin trips to produce an adjusted estimate of total origins, or:

$$O_j = O_{pi} + O_e \tag{27.30}$$

where O_j is the total number of crime origins for crimes committed in study area, j, O_{pi} is the total number of crimes originating in the origin zones, i, and O_e is the total number of crimes originating outside the region, e.

In other words, for the production (origin) model **only**, we add an external zone to account for crime trips that originated outside the modeled region. If that is not done, in the balancing step the number of crimes originating in each zone will be overestimated because the predicted origins will be multiplied by a factor to ensure that the total number of origins equals the total number of destinations.

Not including the external trips can lead to bias in the model. If the number of external trips is a sizeable percentage of all crime origins occurring in the study area, then the coefficients of the origin model could be misleading. In practice, most travel demand modelers assume that if

the percentage of external trips is not greater than 5%, there usually is little bias introduced (Ortuzar & Willumsen, 2001). If it is greater than 5%, then origin zones from adjacent jurisdictions need to be included in the origin model.

Balancing Predicted Origins and Predicted Destinations

The trip generation 'model' is actually two separate models: 1) a model of trips produced by every zone and 2) a model of trips attracted to every zone. Since a trip has an origin and a destination (by definition), then the total number of productions must equal the total number of attractions,

$$\sum_{i=1}^{M} O_i = \sum_{j=1}^{N} D_j \qquad (27.31)$$

where *O* is a trip origin, *D* is a trip destination, and i and j are zone numbers. Note that in equation 27.31, there are *M* origin zones and *N* destination zones. This implies that *M* and *N* do not have to be equal. In fact, including an external zone guarantees that *M* and *N* will not be equal (*M* will be at least 1 greater than *N*). If an entire metropolitan area is being modeled, the *M* and *N* will be almost identical (differing only in the external zone). However, if the study area being modeled is a sub-set of the metropolitan region, then *M* will be much greater than *N*. For example, in modeling crime trips in Baltimore County, there are 532 origin zones (including those from Baltimore County and from the City of Baltimore) and only 325 destination zones (only those in Baltimore County).

To ensure that this equality is true, a balancing operation is conducted. Essentially, this means multiplying either the number of predicted origins in each origin zone or the number of predicted destinations in each destination zone by a constant which is the ratio of either the total destinations to the total origins (to multiply the number of predicted origins) or the ratio of the total origins to the total destinations (to multiply the number of predicted destinations).

With crime analysis, the number of destinations would generally be considered a more reliable data set than the number of origins. Because crimes are enumerated where they occur, the number of crimes occurring at any one location is more accurate than the location of the offenders. Thus, we adjust the predicted origins so that they equal the predicted destinations.[8]

[8] In the usual travel demand modeling, on the other hand, modelers usually adjust the predicted destinations since the origin data is more reliable. These numbers are obtained from the census or from the sample of households who are interviewed to produce a sample from which data on destinations are obtained.

Summary of the Trip Generation Model

In summary, the trip generation model is estimated in four steps:

1. A model of the predictors of the number of crimes origins (a crime production model);
2. A model of predictors of the number of crime destinations (a crime attraction model);
3. External trips are estimated and added to the number of predicted origins as an external zone; and
4. The total number of predicted crime origins is balanced to be equal to the total number of predicted crime destinations.

The *CrimeStat* Trip Generation Model

In this section, we describe the trip generation model implemented in *CrimeStat*. As mentioned above, this step involves calibrating a regression model against the zonal data. Two separate models are developed, one for trip origins and one for trip destinations. The dependent variable is the number of crimes originating in a zone (for the trip origin model) or the number of crimes ending in a zone (for the trip destination model). The independent variables are zonal variables that may predict the number of origins or destinations.

There are three steps to the model, each corresponding to a separate tab in *CrimeStat*:

1. Calibrate the model
2. Make a prediction
3. Balance the predicted origins and the predicted destinations

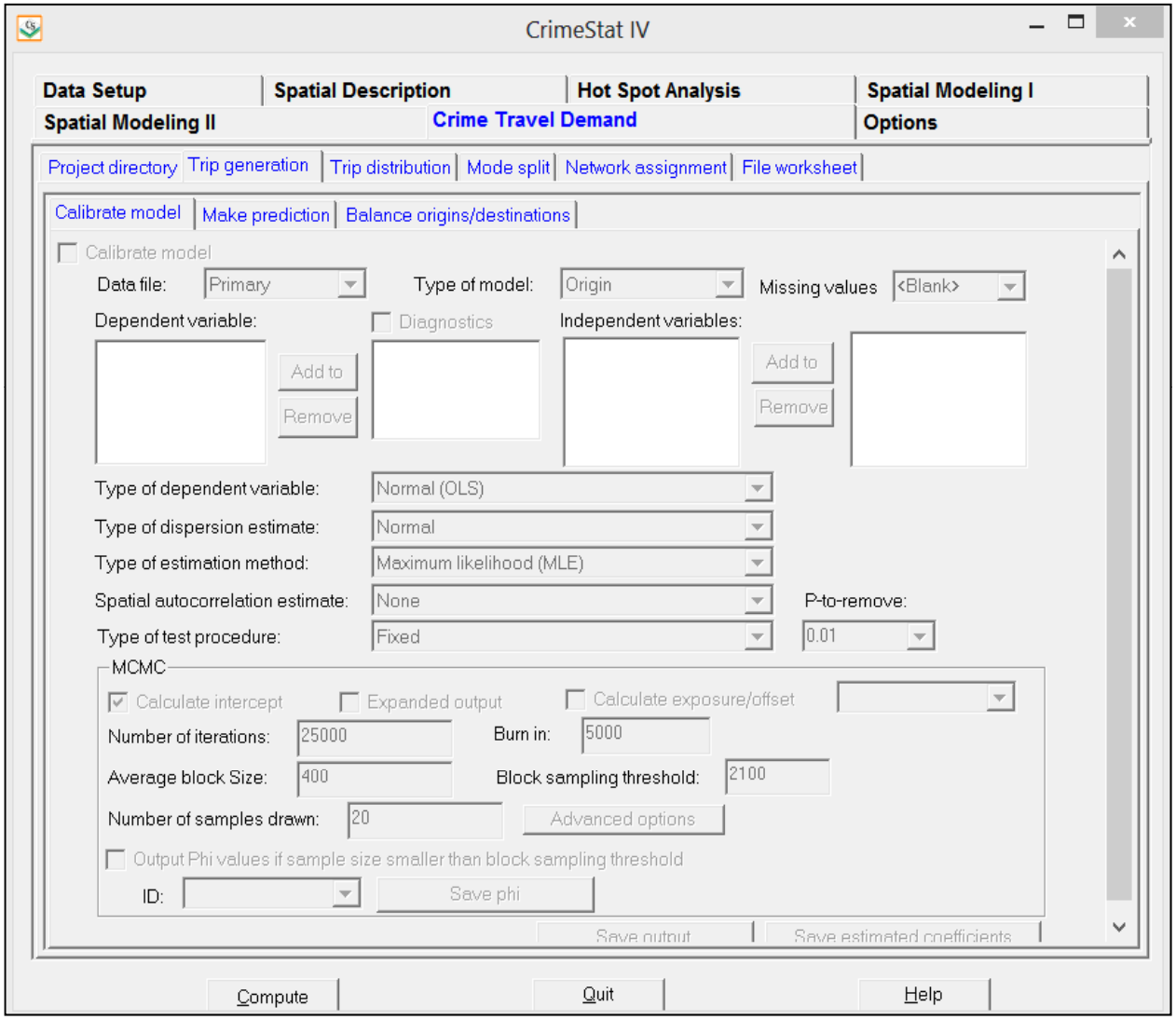
Figure 27.5 shows an image of the trip generation model page within *CrimeStat*. The trip generation model is made up of three separate pages (or tabs):

1. A *Calibrate model* page in which a regression model can be run to estimate either an origin (production) model or a destination (attraction) model;

2. A *Make prediction* page in which the estimated coefficients can be applied to the same or a different data set and in which the external trips can be added to the predicted origins; and

Figure 27.5:

Trip Generation Module



3. A *Balance predicted origins & destinations* page in which the total predicted origins can be adjusted to equal the total predicted destinations.

Calibrate Model

In the first step, models are calibrated using the input data. There is a model for the origin zones and another model for the destination zones. The user should indicate what type of model is being run in order to make the output more clear.

Data File

The data file is input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Type of Model

Specify whether the model is for origins or destinations. This will be printed out on the output header.

Dependent Variable

Select the dependent variable from the list of variables. There can be only one dependent variable per model.

Skewness Diagnostics

If checked, the routine will test for the skewness of the dependent variable. The output includes:

1. The 'g'statistic
2. The standard error of the 'g' statistic
3. The Z value for the 'g' statistic
4. The probability level of a Type I error for the 'g' statistic
5. The ratio of the sample variance to the sample mean

Error messages indicate whether there is probable skewness in the dependent variable. If there is skewness, use a Poisson regression model.

Independent variables

Select independent variables from the list of variables in the data file. Up to 15 variables can be selected.

Missing values

Specify any missing value codes for the variables. Blank records will automatically be considered as missing. If any of the selected dependent or independent variables have missing values, those records will be excluded from the analysis.

Type of Regression Model

Specify the type of regression model to be used. The default is a Poisson regression with over-dispersion correction (NB1). Other alternatives are:

1. Ordinary Least Squares regression;
2. Poisson regression;
3. MLE Poisson-Gamma (negative binomial; NB2);
4. MCMC Poisson-Gamma;
5. MCMC Poisson-Lognormal; and
6. MCMC Conway-Maxwell Poisson.

Each of the MCMC models can be run with a spatial autocorrelation component added, either a CAR or a SAR. See Chapters 16 and 17 for more details.

Type of Regression Procedure

If the model being run is an MLE routine (Poisson, Poisson with linear correction (NB1), or Poisson-Gamma (NB2), specify whether a fixed model (all selected independent variables are used in the regression) or a backward elimination stepwise model is used. The default is a fixed model. If a backward elimination stepwise model is selected, choose the P-to-remove value (default is .01). The backward elimination starts with all selected variables in the model (the fixed procedure). However, it proceeds to drop variables that fail the P-to-remove test, one at a time. Any variable that has a significance level in excess of the P-to-remove value is dropped from the equation.

With MCMC routines, however, only fixed models can be run.

Save Estimated Coefficients/Parameters

The estimated coefficients of the final model can be saved as a 'dbf' file. Specify a file name. This would be useful in order to repeat the regression while adding in external trips to the predicted origins (see Make trip generation prediction below) or to apply the coefficients to another dataset (e.g., future values of the independent variable).

Save Output

The output is saved as a 'dbf' file under a different file name. The output includes all the variables in the input data set plus two new ones: 1) the predicted values of the dependent variable for each observation (with the name PREDICTED); and 2) the residual error values, representing the difference between the actual /observed values for each observation and the predicted values (with the name RESIDUAL).

Poisson output

The output of the Poisson regression routines includes 13 fields for the entire model:

1. The dependent variable
2. The type of model
3. The sample size (N)
4. The degrees of freedom (N - # dependent variables – 1)
5. The type of regression model (Poisson, Poisson with over-dispersion correction)
6. The log-likelihood value
7. The Likelihood Ratio
8. The probability value of the Likelihood Ratio
9. The Akaike Information Criterion (AIC)
10. The Bayesian Information Criterion/Schwartz Criterion (BIC/SC)
11. The Dispersion Multiplier
12. The approximate R-square value
13. The deviance R-square value

and 5 fields for each estimated coefficient:

14. The estimated coefficient
15. The standard error of the coefficient
16. The pseudo-tolerance value of the coefficient (see below)

17. The Z-value of the coefficient
18. The p-value of the coefficient.

OLS output

The output of the Ordinary Least Square (OLS) routine includes 9 fields for the entire model:

1. The dependent variable
2. The type of model
3. The sample size (N)
4. The degrees of freedom (N - # dependent variables – 1)
5. The type of regression model (Norma/Ordinary Least Squares)
6. Squared multiple R
7. Adjusted squared multiple R
8. F test of the model
9. p-value of the model

and 5 fields for each estimated coefficient:

10. The estimated coefficient
11. The standard error of the coefficient
12. The tolerance value of the coefficient (see below)
13. The t-value of the coefficient
14. The p-value of the coefficient.

Multicollinearity Among Independent Variables

To test multicollinearity, a tolerance test is run (see equation 27.29 above). There is not a simple test of whether a particular tolerance is meaningful or not. In *CrimeStat*, several qualitative categories are used and error messages are output:

1. If the tolerance value is 0.80 or greater, then there is little multicollinearity (No apparent multicollinearity);

2. If the tolerance is between 0.60-0.79, there is some multicollinearity (possible multicollinearity);

3. If the tolerance is between 0.25-0.59, there is probable multicollinearity (probable multicollinearity. Eliminate variable with lowest tolerance and re-run); and

4. If tolerance is less than 0.25, there is definite multicollinearity (Definite multicollinearity. Results are not reliable. Eliminate variable with lowest tolerance and re-run).

Graph

While the output page is open, clicking on the graph button will display a graph of the residual errors (on the Y axis) against the predicted values (on the X axis).

Make Trip Generation Prediction

This routine applies an already-calibrated regression model to a data set. This would be useful for several reasons: 1) if external trips are to be added to the model (which is normally preferred); 2) if the model is applied to another data set; and 3) if variations on the coefficients are being tested with the same data set. The model will need to be calibrated first (see Calibrate Trip Generation Model) and the coefficients saved as a parameters file. The coefficient parameter file is then re-loaded and applied to the data.

Data File

The data file is input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

Type of Model

Specify whether the model is for origins or destinations. This will be printed out on the output header.

Trip Generation Coefficients/Parameters File

This is the saved coefficient parameter file. It is an ASCII file and can be edited if alternative coefficients are being tested (be careful about editing this without making a backup). Load the file by clicking on the Browse button and finding the file. Once loaded, the variable names of the saved coefficients are displayed in the 'Matching parameters' box.

Independent Variables

Select independent variables from the list of variables in the data file. Up to 15 variables can be selected.

Matching parameters

The selected independent variables need to be matched to the saved variables in the trip generation parameters file in the same order. Add the appropriate variables one by one in the order in which they are listed in the matching parameters box. It is essential that the order by the same otherwise the coefficients will be applied to the wrong variables.

> Hint: With your cursor placed in the list of independent variables, typing the first letter of the matching variable name will take you to the first variable that starts with that letter. Repeating the letter will move down the list to the second, third, and so forth until the desired variable is reached.

Specify any missing value codes for the variables. Blank records will automatically be considered as missing. If any of the selected dependent or independent variables have blank values, those records will be excluded from the analysis.

Add External Trips

External trips are those that start outside the modeled study area. Because they are crimes that originate outside the study area, they were not included in the zones used for the origin model. Therefore, they have to be independently estimated and added to the origin zone total to make the number of origins equal to the number of destinations. Click on the 'Add external trips' button to enable this feature.

Origin ID

Specify the origin ID variable in the data file. The external trips will be added as an extra origin zone, called the 'External' zone. Note: the ID's used for the destination file zones should be the same as in the origin file. This will be necessary in subsequent modeling stages.

Number of external trips

Add the number of external trips to the box. This number will be added as an extra origin zone (the External zone).

Type of Regression Model

Specify the type of regression model to be used. The default is a Poisson regression and the other alternative is an Ordinary Least Squares regression.

Save Predicted Values

The output is saved as a 'dbf' file under a different file name. The output includes all the variables in the input data set plus the predicted values of the dependent variable for each observation (with the name PREDICTED). In addition, if external trips are added, then there should be a new record with the name EXTERNAL listed in the Origin ID column. This record lists the added trips in the PREDICTED column and zeros (0) for all other numeric fields.

Output

The tabular output includes summary information about file and lists the predicted values for each input zone.

Balance Predicted Origins & Destinations

Since, by definition, a 'trip' has an origin and a destination, the number of predicted origins must equal the number of predicted destinations. Because of slight differences in the data sets of the origin model and the destination model, it is possible that the total number of predicted origins (including any external trips) may not equal the total number of predicted destinations. This step, therefore, is essential to guarantee that this condition will be true. The routine adjusts either the number of predicted origins or the number of predicted destinations so that the condition holds. The trip distribution routines will not work unless the number of predicted origins equals the number of predicted destinations (within a very small rounding-off error).

Predicted Origin File

Specify the name of the predicted origin file by clicking on the Browse button and locating the file.

Origin variable

Specify the name of the variable for the predicted origins (e.g., PREDICTED).

Predicted Destination File

Specify the name of the predicted destination file by clicking on the Browse button and locating the file.

Destination variable

Specify the name of the variable for the predicted origins (e.g., PREDICTED).

Balancing Method

Specify whether origins or destinations are to be held constant. The default is 'Hold destinations constant'.

Save Predicted Origin/Destination File

The output is saved as a 'dbf' file under a different file name. The output includes all the variables in the input data set plus the adjusted values of the predicted values of the dependent variable for each observation. If destinations are held constant, the adjusted variable name for the predicted trips is ADJORIGIN. If origins are held constant, the adjusted variable name for the predicted trips is ADJDEST.

Output

The tabular output includes file summary information plus information about the number of origins and destinations before and after balancing. In addition, the predicted values of the dependent variable are displayed.

Example of the Trip Generation Model

To illustrate this model, an example from Baltimore County. In the case of Baltimore County, MD, will be used. The zonal geography is traffic analysis zones (TAZ). Two data sets were produced, one for the crime origins and one for the crime destinations. For Baltimore County, the origin data set had 532 zones covering both Baltimore County and the City of Baltimore with the total number of crime origins for each zone (sub-divided into different crime types - robberies, burglaries, vehicle theft) and a number of possible predictor variables (population, retail and non-retail employment, median household income, poverty levels, and vehicle ownership). Similarly, the destination data set had 325 zones with the number of crime destinations for each zone (again, sub-divided into different crime types) and number of possible predictor variables (population, retail and non-retail employment, median household income, and

several land use categories - acreage allocated for retail, residential, office space, and conservation uses). Sample data sets are provided on the *CrimeStat* download page.

Setting Up the Origin Model

In the first step, an origin model is created. Figure 27.6 shows the selection of the dependent variable and some possible independent variables. The type of model is an ordinary Poisson regression. The dependent variable is the number of crimes occurring between 1993 and 1997 in each origin zone (BCORIG). Eight possible independent variables have been selected:

1. 1996 population of each zone (POPULATION)
2. 1990 median household income of the zone relative to the zone with the highest median household income (INCOME EQUALITY)
3. Number of 1996 non-retail employees in each zone (NON-RETAIL EMPLOYMENT)
4. Number of 1996 retail employees in each zone (RETAIL EMPLOYMENT)
5. Total linear miles of arterial roads in each zone (ARTERIAL ROADS)
6. A dummy variable for whether the Baltimore Beltway (I-695) passed through the zone or not (BELTWAY)
7. Linear distance of the zone from Baltimore harbor in the CBD (DISTANCE FROM CENTER); and
8. 1990 Number of households without automobiles (HOUSEHOLDS WITH NO AUTOMOBILES)

The model is set up to run an ordinary Poisson regression (without an adjustment to the dispersion). It is a fixed model in which all independent variables are included. The coefficients are saved under 'Save estimated coefficients' dialogue box and the output (the predicted values) are saved under the 'Save output' dialogue box. Both boxes ask for a file name.

Table 27.2 shows the results. The format is simplified from that shown in Chapter 16. Key statistics are highlighted. The overall model is highly significant. The log likelihood is shown as are the AIC and BIC/SC adjusted log likelihood. The deviance and Pearson are highly significant, indicating that the model predicts significantly better than chance. The coefficients for each of the variables are all significant.

However, there are two major problems. First, the dispersion multiplier (parameter) is very large (36.09) and significant, indicating that the conditional variance is more than 36 times greater than the conditional mean. Second, while all of the coefficients are significant, several show sizeable multicollinearity as evidenced by the pseudo-tolerance value (POPULATION, DISTANCE, HOUSEHOLDS WITH NO AUTOMOBILES as well as INCOME EQUALITY). This indicates that these variables are essentially measuring the same thing.

Table 27.2:
Full Origin Model: Poisson

Model result:		
Data file:	BaltOrigins.dbf	
Type of model:	Origin	
DepVar:	**BCORIG**	
N:	532	
Df:	522	
Type of regression model:	MLE Poisson	
Likelihood statistics		
Log Likelihood:	-10,678.05	
AIC:	21,376.10	
BIC/SC:	21,418.87	
Deviance:	18,547.38	p≤.0001
Pearson Chi-square:	19,396.48	p≤.0001
Model error estimates		
Mean absolute deviation:	38.70	
Mean squared predicted error:	3,920.66	
Dispersion tests		
Dispersion multiplier:	36.09	p≤.01

Predictor	**Coefficient**	**Stand Error**	**Tolerance**	**Z-value**	**p-value**
CONSTANT	**4.1890**	0.0202	.	207.03	0.001
POPULATION	**0.0003**	0.000003	**0.46**	121.13	0.001
INCOME EQUALITY	**-0.0330**	0.0007	**0.61**	-48.85	0.001
NON-RETAIL EMPLOYMENT	**-0.0002**	0.000005	0.84	-36.87	0.001
RETAIL EMPLOYMENT	**-0.0004**	0.00002	0.96	-18.91	0.010
ARTERIAL ROAD	**-0.1083**	0.0059	0.77	-18.49	0.001
BELTWAY	**0.1510**	0.0193	0.96	7.84	0.001
DISTANCE FROM CENTER	**0.0343**	0.0016	**0.49**	21.09	0.001
HOUSEHOLDS WITH NO AUTOMOBILES	**-0.0005**	0.00002	**0.36**	-18.95	0.010

Restructuring the Origin Model

Consequently, the model was restructured in three ways (Figure 27.7). First, to correct for over-dispersion, an MLE Poisson-Gamma (negative binomial) model was run. This is the most common approach to handling over-dispersion (see Chapter 16). Second, two co-linear variables - DISTANCE and ZEROAUTO, were dropped from the model. Third, a stepwise backward elimination procedure is used with the probability for keeping a variable in the equation (p-to-remove) being 0.01; that is, unless the probability that a coefficient could be obtained by chance is less than 1 in 100, the variable was dropped.

Table 27.3:
Reduced Origin Model: Poisson-Gamma

Model result:		
Data file:	BaltOrigins.dbf	
Type of model:	Origin	
DepVar:	**BCORIG**	
N:	532	
Df:	526	
Type of regression model:	MLE Poisson-Gamma	
Likelihood statistics		
Log Likelihood:	-2,627.65	
AIC:	5,267.30	
BIC/SC:	5,292.96	
Deviance:	623.49	p≤.0001
Pearson Chi-square:	500.59	p≤.0001
Model error estimates		
Mean absolute deviation:	57.28	
Mean squared predicted error:	18,143.78	
Dispersion tests		
Dispersion multiplier:	0.74	n.a.

Predictor	**Coefficient**	**Stand Error**	**Tolerance**	**Z-value**	**p-value**
CONSTANT	**3.4832**	0.131	.	26.61	0.001
POPULATION	**0.0004**	0.00003	0.95	17.10	0.001
INCOME EQUALITY	**-0.0178**	0.003	0.91	-5.46	0.001
NON-RETAIL EMPLOYMENT	**-0.0001**	0.00002	0.87	-6.71	0.001
RETAIL EMPLOYMENT	**-0.0002**	0.0001	0.96	-2.17	0.05

Figure 27.6:

Origin Poisson Model Setup

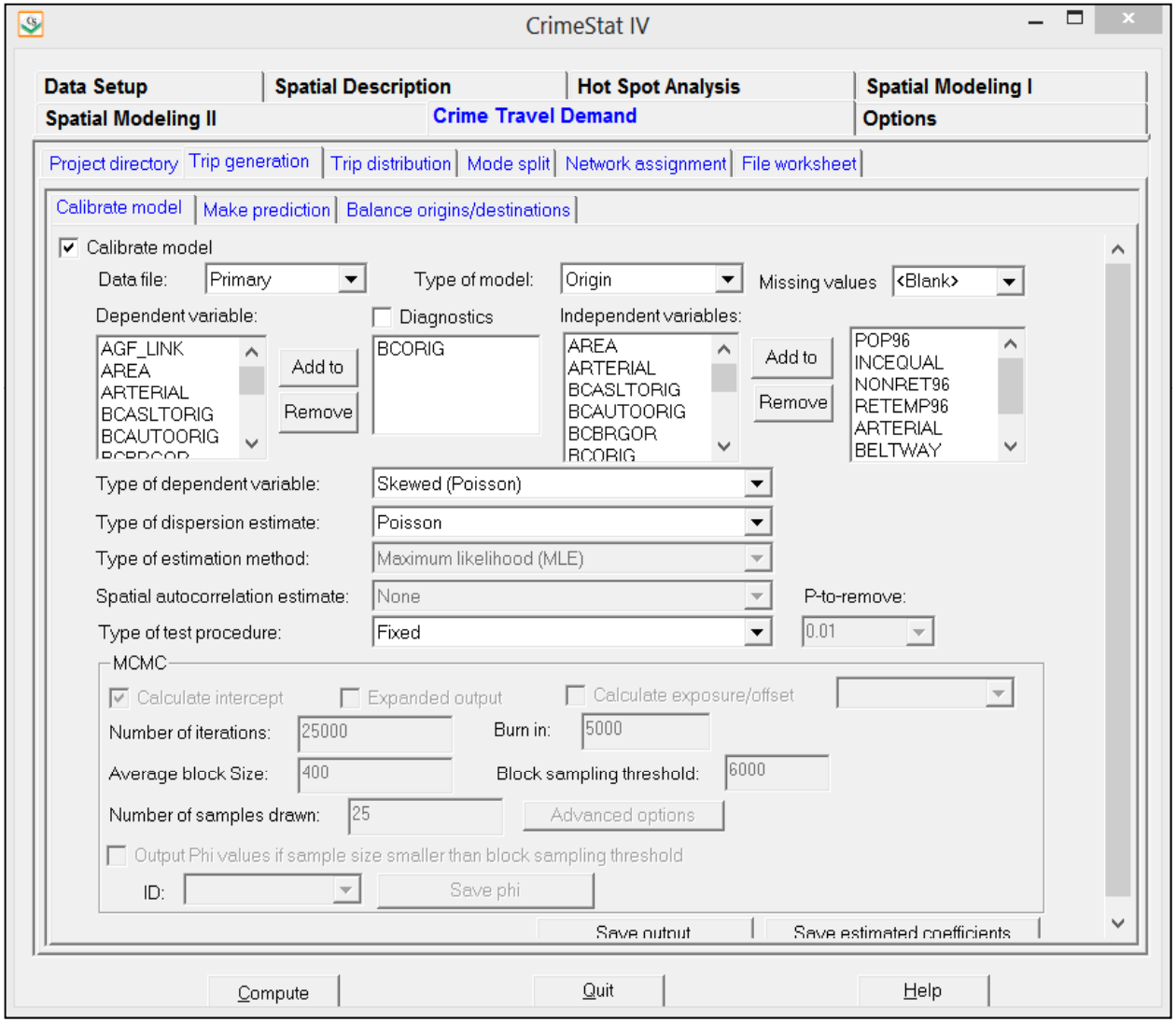
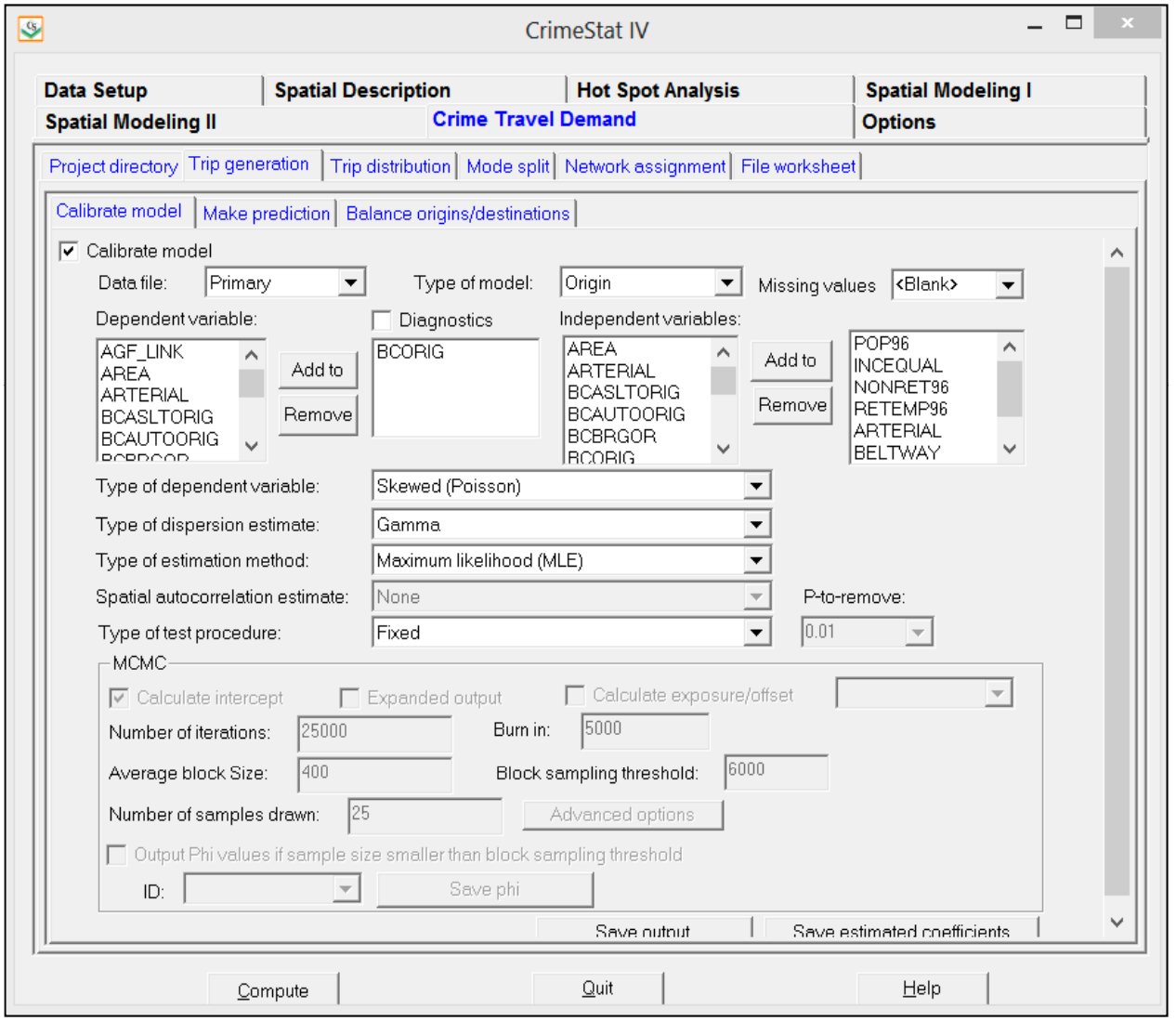


Figure 27.7:

Origin Poisson-Gamma Model Setup



The result is a model with four significant variables. Note that the full Poisson model (Table 27.3) has a greater negative log likelihood and a much greater AIC and BIC/SC value than the reduced model. This is because the reduced model was tested with a Poisson-Gamma mixed function and has a different probability structure. To properly compare it, the full model was run as a Poisson-Gamma (not shown). In the reduced model, the log likelihood was -2,627, which is even stronger than -2,618 for the full model, while the AIC was 5,267 and the BIC/SC was 5,293, compared to 5,526 and 5,299 for the full model respectively. In other words, the reduced model produced likelihood values very similar to the full model. More importantly, the overall fit of the model was almost as good as the full model. The mean absolute deviation was 57, compared to 55 for the full model, while the mean squared predicted error was 18,144, compared to 17,881 for the full model. Most importantly, the dispersion parameter is now less than 1.0.[9]

In other words, we have a simpler model that predicts almost as well as the full model but with coefficients that are less ambiguous. Such a model is liable to be more stable because the Poisson-Gamma has adjusted for over-dispersion in the Poisson while collinear and less significant variables have been removed.

Looking at the model, we see four variables that significantly predict the number of crime origins. Population is the strongest, as indicated by its Z-test. Non-retail employment is the next strongest with a negative coefficient (i.e., zones with less non-retail employment generate more crime trips). This is followed by relative income equality is the next strongest, also with a negative coefficient (i.e., zones with low relative income equality produce more crime origins). The fourth variable is retail employment and, like non-retail employment, the coefficient is also negative. In other words, zones with less overall employment produce more crime trips.

Residual Analysis of Origin Model

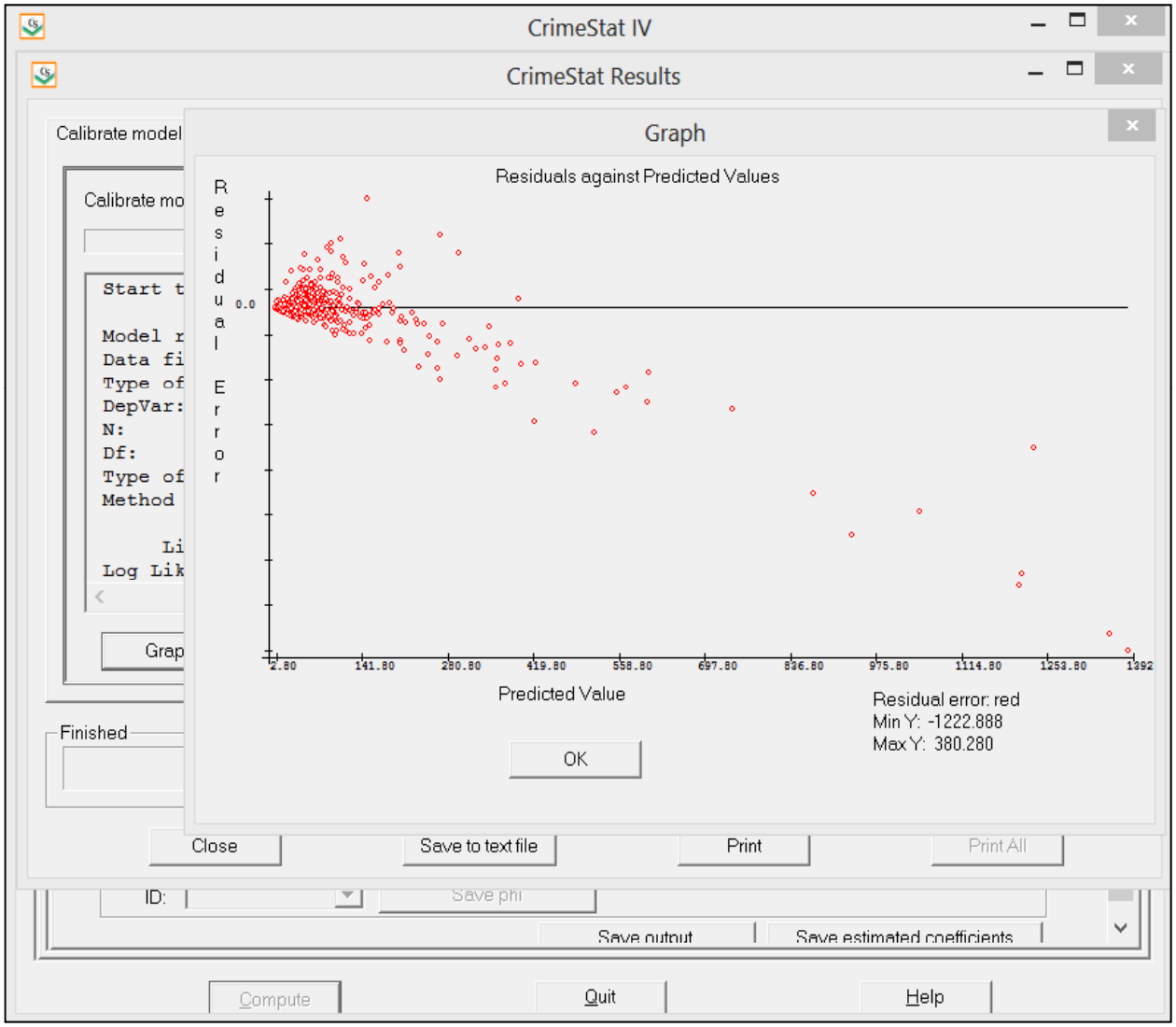
The *CrimeStat* output includes a graph of the residual errors (actual values minus the predicted values) on the Y-axis by the predicted values on the X-axis. It is important to examine the residual errors as these can indicate outliers, problems in the data, and violation of assumptions. Figure 27.8 shows an image of the residual graph screen. As seen, the errors increase with the value of the predicted dependent variable. With the Poisson model, this is expected and does not indicate the violation of the independent errors assumption, as it does with the OLS. The errors are reasonably symmetrical and do not indicate differences in over- and under-estimation across the band of the predicted values.

There are some outliers. There are two zones in which the predicted number of crimes originating from the zones substantially exceeded the number that actually originated from those

[9] A test of the dispersion parameter is not appropriate since it only tests for over-dispersion, not under-dispersion.

Figure 27.8:

Plot of Residual Errors and Predicted Values



zones and there is one zone that had more crimes originate from it than was predicted by the model. But, in general, the model appears to be reasonably balanced.

Setting Up the Destination Model

The same logic was applied for the destination model. In this case, the destination file has data on 325 zones within Baltimore County only. Similar possible predictor variables are included in the file. Aside from population, retail and non-retail employment, and the roadway variables, more detailed analysis on land uses were included (acreage of commercial, residential, office space, recreational, and conservation lands). The model that was run was a Poisson-Gamma (negative binomial) because the simple Poisson showed very high over-dispersion. Again, a backward elimination procedure was adopted. Once a final model was selected, it was re-run as a fixed model to ensure that the coefficients were consistently estimated. Table 27.4 presents the results.

Four variables ended up in the final model. Again, population was significantly related to the number of crimes attracted to a zone, but was not the strongest predictor as indicated by the Z-test. The strongest relationship was for the number of retail employees. This suggests that retail/commercial areas attract many crimes. Two other variables are in the equation. Relative income equality was, again, negatively related to crime destinations/attractions; zones with low income tend to attract more crimes. Also, there was a negative association with distance from the CBD. The farther away from the CBD was the zone, the lower the number of crimes. Overall, the model suggests that zones with commercial activities, which are closer to the city center, and which have households with relatively lower incomes are those that attract the most crimes.

The overall model was highly significant, as indicated by the Deviance and the Pearson Chi-square. The amount of multicollinearity is very low, which is ideal. Even though a model with more negative log likelihood (and more positive AIC and BIC/SC) could be produced by adding more variables, the amount of multicollinearity would be substantial. The philosophy expressed here is that a simpler model, but with little multicollinearity, is to be preferred over a more complex model but where the coefficients are less stable and more ambiguous. Generally, simpler models hold up better with new data sets (Radford, 2006; Nannen, 2003).

Residual Analysis of Destination Model

As with the origin model, an analysis was conducted of the residual errors. This time, the output 'dbf' file was brought into Excel and a nicer graph created (Figure 27.9). Unlike the best origin model, the dispersion of the residuals is not symmetrical. There are several major outliers, both on the negative end of the residuals (over-estimation of crime attractions) and on the positive end (under-estimation of crime attractions). In particular, there are two zones that seem to stand

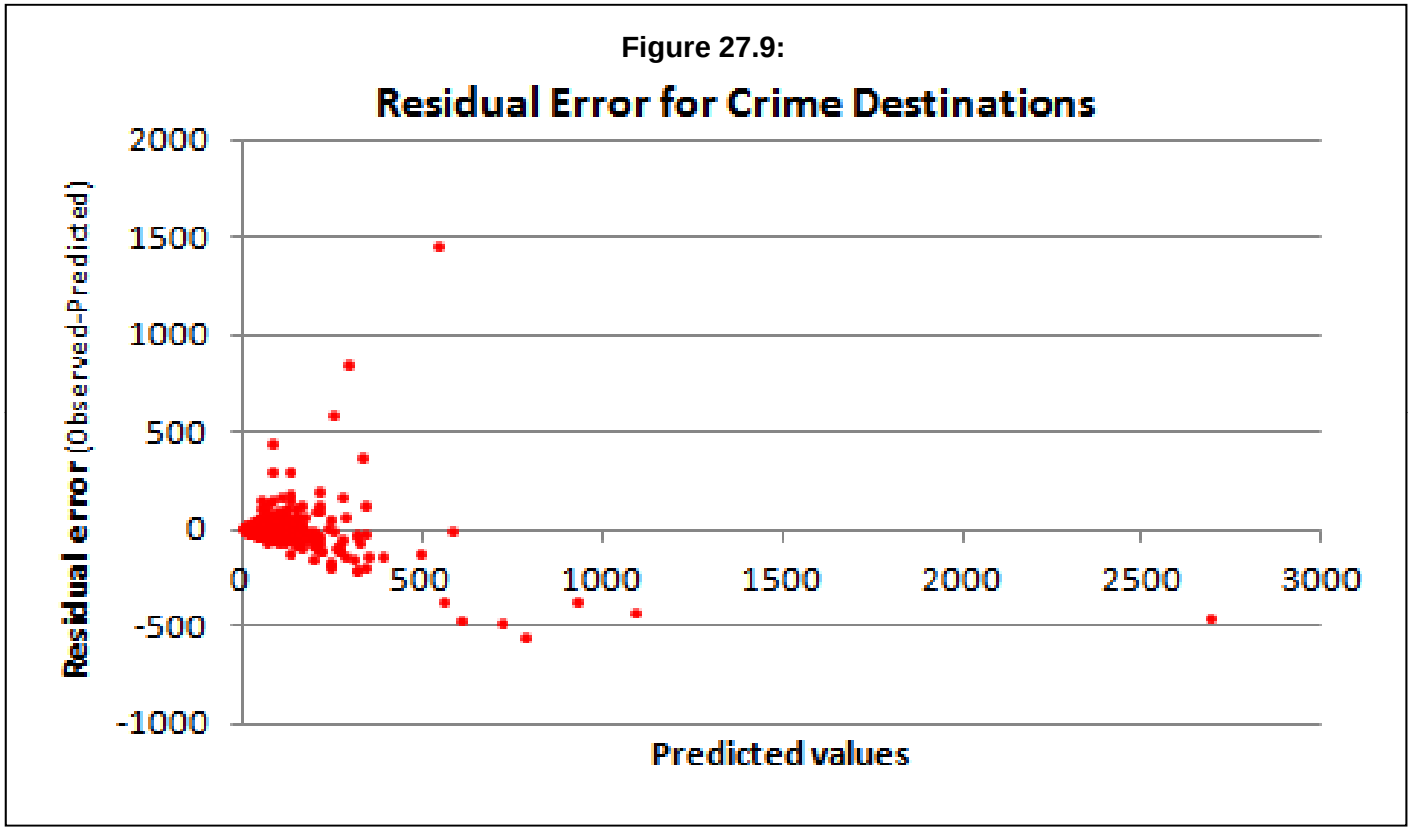
out. Both of them have shopping malls (Golden Ring Mall and Eastpoint Mall) but the amount of crime in those zones was much greater than the model predicted. This is seen as high positive residuals (i.e., there were more actual crimes than predicted). They both are older malls, but are located in relatively high crime areas. Golden Ring Mall was demolished some years ago, but after the data used in this example were collected.

Table 27.4:
Reduced Destination Model: Poisson-Gamma

Model result:		
Data file:	BaltOrigins.dbf	
Type of model:	Origin	
DepVar:	**BCDEST**	
N:	325	
Df:	319	
Type of regression model:	MLE Poisson-Gamma	
Likelihood statistics		
Log Likelihood:	-1,697.01	
AIC:	3,406.03	
BIC/SC:	3,428.73	
Deviance:	350.74	p≤.0001
Pearson Chi-square:	379.87	p≤.0001
Model error estimates		
Mean absolute deviation:	167.43	
Mean squared predicted error:	2,893,931.60	
Dispersion tests		
Dispersion multiplier:	0.43	n.a.

Predictor	**Coefficient**	**Stand Error**	**Tolerance**	**Z-value**	**p-value**
CONSTANT	**4.5208**	0.153	.	29.53	0.001
POPULATION	**0.0003**	0.00003	0.94	12.06	0.001
INCOME EQUALITY	**-0.0213**	0.003	0.90	-7.87	0.001
RETAIL EMPLOYMENT	**0.0020**	0.0001	0.94	17.49	0.001
DISTANCE FROM CENTER	**-0.0714**	0.010	0.88	-7.46	0.001

Figure 27.9:


Residual Error for Crime Destinations
Residual error (Observed-Predicted)
2000
1500
1000
500
0
-500
-1000
0
500
1000
1500
2000
2500
3000
Predicted values

Adding in Special Generators

Since the number of crime incidents (attractions) in those two zones was much higher than expected, they were treated as 'special generators'. Keeping in mind the caution that one does not want to over-use this category, a demonstration of how it works will be illustrated. Two new variables were created for the data set. One was for the Golden Ring Mall and one was for the Eastpoint Mall. For the Golden Ring Mall, the zone that included it received a '1' for this variable while all other zones received a '0'. Similarly, for the Eastpoint Mall variable, the zone in which it occurred received a '1' while all other zones received a '0'. These *dummy* variables were then included in the model (Table 27.5).

Adding the two special generators produces a model that, on the face of it, has not improved the predictability.. The log likelihood value is less negative than without the special generators and the AIC and BIC/SC statistics are also lower. The deviance values are about the same.

However, the Pearson Chi-square is quite a bit higher with the special generators. Also, the Mean Absolute Deviation (MAD) and the Mean Squared Predictive Error (MSPE) are substantially better with the special generators. This indicates that the new model which includes the special generators fit the data much better.

The coefficients for the two zones, treated as special generators, are both significant though not as strongly as the other variables. All other variables have the same relationships as in the first run. There is little multicollinearity. In other words, adding dummy variables for the two zones with higher than expected numbers of crime committed has produced a closer fitting model than not including the dummy variables.

This brings up an issue over the status of a special generator. In this example, the two zones were treated as special generators in the model. While the model fit increased substantially, one has to wonder whether this was a meaningful operation or not? That is, if this model were applied to data for a later time period (e.g., 2010-2012 crime data), would the relationships still hold? In the case of the Golden Ring Mall, it would not since that mall has since been demolished

The value of a special generator is that it identifies a land use that would be expected to be relatively permanent (e.g., a stadium or a train station or an airport). If it is a high visibility 'regional' mall, then treating it as a special generator is probably a good idea. If it is a smaller, older mall, on the other hand, the analysis is guessing that the mall will maintain its status as a high crime attraction location. Clearly, judgment and knowledge of the particular mall is essential.

Table 27.5:

Destination Model with Special Generators: Poisson-Gamma

Model result:		
Data file:	BaltOrigins.dbf	
Type of model:	Origin	
DepVar:	**BCDEST**	
N:	325	
Df:	319	
Type of regression model:	MLE Poisson-Gamma	
Likelihood statistics		
Log Likelihood:	-1,688.30	
AIC:	3,392.60	
BIC/SC:	3,422.87	
Deviance:	350.89	p≤.0001
Pearson Chi-square:	398.73	p≤.0001
Model error estimates		
Mean absolute deviation:	116.54	
Mean squared predicted error:	1,017,064.23	
Dispersion tests		
Dispersion multiplier:	0.40	n.a.

Predictor	**Coefficient**	**Stand Error**	**Tolerance**	**Z-value**	**p-value**
CONSTANT	**4.4625**	0.149	.	29.91	0.001
POPULATION	**0.0004**	0.00003	0.93	12.72	0.001
INCOME EQUALITY	**-0.0205**	0.003	0.90	-7.84	0.001
RETAIL EMPLOYMENT	**0.0018**	0.0001	0.90	16.44	0.001
DISTANCE FROM CENTER	**-0.0686**	0.009	0.87	-7.30	0.001
GOLDEN RING MALL	**1.5163**	0.645	0.98	2.35	0.05
EASTPOINT MALL	**1.6111**	0.648	0.97	2.49	0.05

Comparing Different Crimes Types

With or without special generators, a trip generation model is an ecological model that predicts crime origins and crime destinations. A point was made in Chapter 25 that these models are not behavioral, but are correlates of crimes. That is, the variables that end up predicting the number of crimes are not *reasons* (or explanations) for the crimes. Population almost always

enters the equation because, all other things being equal, zones with larger numbers of persons will have more crimes, both originating and ending in them. Similarly, low income status is frequently associated with high crime areas. It does not follow that low income persons will be more prone to commit crimes; it may be true but these models do not test that proposition (Ratcliffe, 2008). These are only correlates with crime in those environments. As was mentioned earlier, these variables are often correlated with many specific conditions that *may* be predictors of individual crime - poverty, drug use, substandard housing, and lack of job opportunities.

To see this, three separate models of specific crime types were run for robbery, burglary, and vehicle theft. For each crime type, the general model was tested for both the origin and the destination models. If a variable was not significant, it was dropped and the model was re-run.

Table 27.6:
Models for Specific Crime Types: Poisson-Gamma Origin Model

	All Crimes	Robbery	Burglary	Vehicle Theft
CONSTANT	3.483	1.1165	1.1165	-1.4994
POPULATION	0.0004	0.0004	0.0004	0.0005
INCOME EQUALITY	-0.0178	-	-	-0.0214
NON-RETAIL EMPLOYMENT	-0.0001	-0.0002	-0.0002	-0.0001
RETAIL EMPLOYMENT	-0.0002	-	-	-

Table 27.7:

Models for Specific Crime Types: Poisson-Gamma Destination Model

	All Crimes	Robbery	Burglary	Vehicle Theft
CONSTANT	4.5208	2.7489	0.7977	2.2903
POPULATION	0.0003	0.0003	0.0004	0.0004
INCOME EQUALITY	-0.0213	-0.0295	-0.0220	-0.0131
RETAIL EMPLOYMENT	0.0020	0.0019	-	0.0009
DISTANCE FROM CBD	-0.0714	-0.0965	-0.0365	-0.1013

The population variable appears in every single model. As mentioned, all other things being equal, the larger the number of persons in a zone, the more crime events will occur whether those events are crime productions (origins) or crime attractions (destinations). Similarly, relative income equality appears in four of the six crime-specific models with the coefficient always being negative. In general, zones with relatively lower incomes will have more robberies, burglaries, and vehicle thefts. The only model for which income equality did not appear was as an origin variable for burglaries; apparently, burglars come from zones with various income levels, at least in Baltimore.

The other general variables have more limited applicability. Retail employment predicts both total crime origins and total crime destinations, but only predicts specifically robbery destinations and vehicle theft destinations; the latter tend to occur more in commercial areas than not. On the other hand, non-retail employment appears to be important only as a crime origin variable; zones with less non-retail employment tend to produce more offender trips. Distance from the CBD only appears as a destination variable; the closer a zone is to the metropolitan center, the higher the number of crimes being attracted to that zone; this variable was not important in the origin model.

In other words, these models are measuring general conditions associated with crime, not causes *per se*. They capture the general contextual relationships associated with crime productions and attractions. But, they do not necessarily predict individual behavior. Nevertheless, the models can be used for prediction since the conditions appear to be quite general.

Adding External Trips to the Origin Model

After an origin and destination model has been developed, the next step is to add any crime trips that came from outside the modeling area (external trips). In this case, these would be trips that came from areas that were not in either Baltimore County or the City of Baltimore (the modeling area).

A simple estimate of external trips is obtained by taking the difference between the total number of crimes occurring in the study area (Baltimore County destinations) and the total number of crimes originating in the modeling area (Table 27.8).

The difference between the number of crime enumerated within Baltimore County and that originating from both Baltimore County and the City of Baltimore is 1,627. This is 3.9% of the total Baltimore County crimes. In general, it is important that the external trips be as small as possible. Ortuzar and Willumsen (2001) suggest that this percentage be no greater than 5% in order to minimize potential bias from not including those cases in the origin model. It is not an absolute percentage, but more like a rule of thumb; in theory, any external trips could bias the origin model. But, in practice, the error will be small if external crime trips are a small percentage of the total number enumerated in the destination county.

In this case, the condition holds. For the three types of crime modeled, the percentage of external trips was also less than 5%: robbery (4.0%), burglary (4.5%), and vehicle theft (1.4%). On the other hand, if the percentage of external trips is greater than approximately 5%, a user would be advised to widen the origin study area to include more zones in the model.

Predicting External Trips

If a model is being applied to another data set from which it was initially estimated, a problem emerges about how to estimate the number of external trips. It is one thing to apply simple arithmetic in order to determine how many trips originated outside the modeling area (as in Table 27.8). It is another to know how to calculate external trips when the model is being applied to other data. For the modeled zones, the coefficients are applied to the variables of the model (see 'Make Prediction' below). But, the external trips have to be estimated independently.

There is not a simple way to estimate external crime trips. Unlike regular trips that can be estimated through cordon counts, crime trips are not detectable while they are occurring (i.e., one cannot stand by a road and count offenders traveling by). Thus, they have to be estimated.

Table 27.8:
Estimating External Crime Trips in Baltimore County

Number of crimes ending in 325 Baltimore County zones:	41,969
Number of crimes originating in 532 Baltimore County/City zones:	40,342
Crimes from outside the modeling area:	1,627

Note: external trips are only added to the origin model since they are crime trips that originate outside the modeling area. They are not relevant for the destination model.

A simple method is to calculate the number of external trips for two time periods. For example, external trips could be calculated from a 2010 data set by subtracting the total number of crimes occurring in the modeling region from the total number of crimes occurring in the study area (e.g., as in Table 27.8 above). If a similar calculation was made for, say, 2012, then the difference (the 'trend') could be extrapolated. To take our example, between 1993 and 1996, there were 1,627 external trips. If the number of external trips turned out to be 1,850 for 1997-2000, then the difference (1,850 - 1,627 = 223) could be applied for future years. Essentially, a slope is being calculated and applied as a linear equation:

$$Y_i = 1850 + 223X_i \tag{27.32}$$

where Y_i is the number of crime origins during a four year period, I, and X_i is an integer for a four year period starting with the next period (i.e., the base year, 1997-2000, has integer value of 0). In other words, a linear trend is being extrapolated.

How realistic is this? For short time periods, linear extrapolation is probably as good a method as any. But for longer time periods, it can lead to spurious conclusions (e.g., crime trips from outside the region will always increase). Short of developing a sophisticated model that

relates crime trips to the growth of the metropolitan area and to other metropolitan areas within, say, 500 miles, a linear extrapolation is one of the few methods that one can apply.[10]

Make Prediction

In *CrimeStat*, external trips are added on the second page of the trip generation - Make prediction. This is a page where the modeled coefficients and any external trips are applied to a data set. There are two reasons why this is a separate page from the 'Calibrate model' page where the model was calibrated. First, the coefficients might be applied to another data than that from which it was calibrated. For example, one might calibrate the model with a data set from 2008-2010 and then apply to a data set covering 2011-2013. Similarly, one might take future year forecasts (e.g., 2025) and apply the model. In effect, the model would be predicting the number of future crimes *if* the same conditions hold over the time frame.

A second reason for separating the calibration and application pages is to add external trips to the origin zones. As mentioned above, external trips are, by definition, those that were not modeled in the calibration. They have to be calculated independently of the model and then added to the estimates.

Thus, the 'Make prediction' page allows these operations to occur. Figure 27.10 shows the page. There are several steps that have to be implemented for this page to be operative.

1. The data file has to be input as either the primary or secondary file (not shown in the image). In this example, the same data set is being used as was used for the calibration. But, if it is a different data set, that will need to be input in the Data Setup section. Whether the input data set is a primary file (the usual occurrence) or a secondary file needs to be specified. Also, indicate whether the applied model is to be an origin or destination model. In Figure 27.10, it is defined as an origin file.

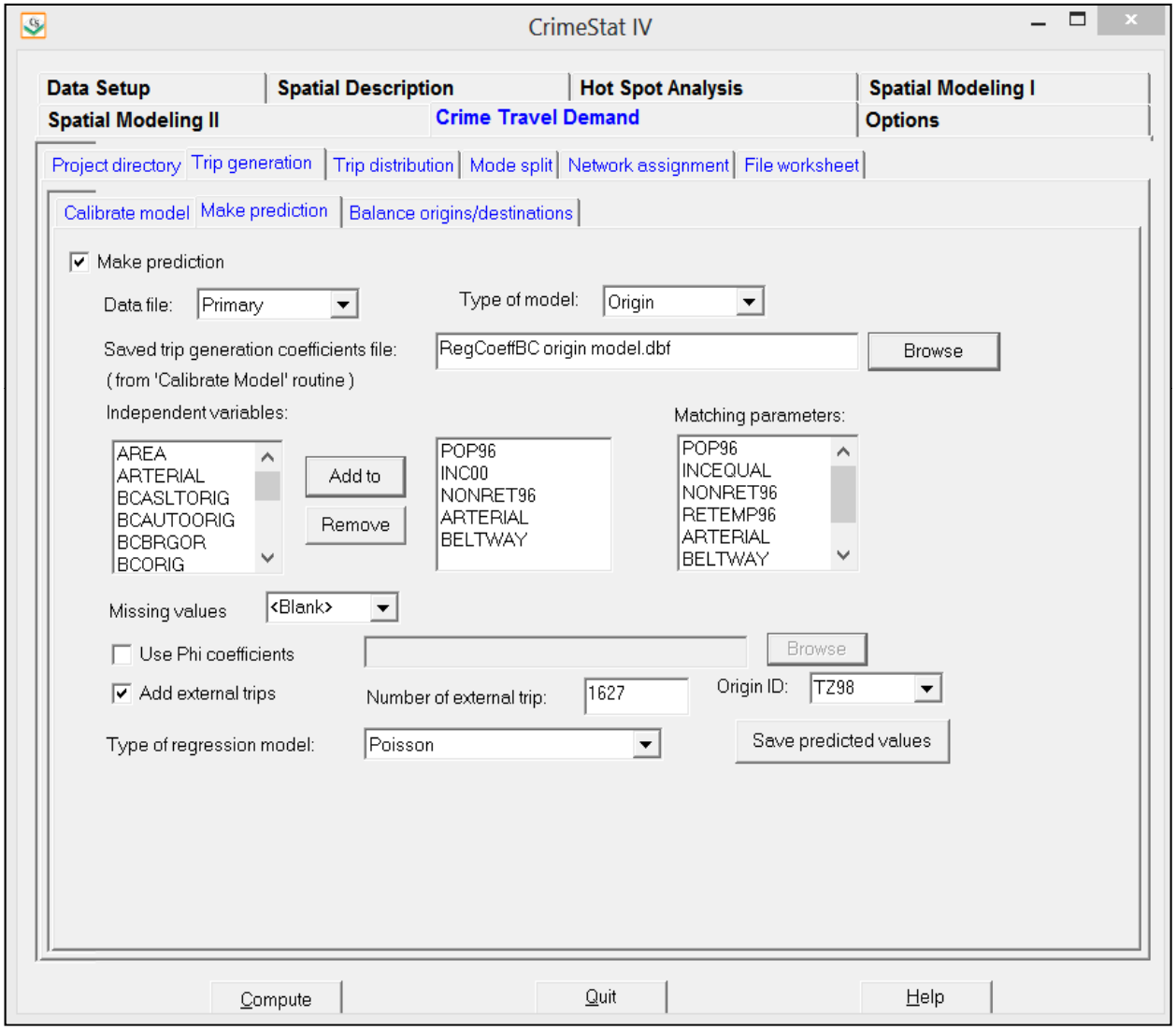
2. A trip generation coefficients file needs to be input. These were the estimated coefficients from the calibration stage. Inputting this file brings in the coefficients in the order in which they were saved. They are listed in the 'Matching parameters' dialogue box on the right side of the page.

---

[10] An alternative might be to use cordon counts from major highways coming into the region and assume that crime trips represent a constant proportion of those trips. Thus, if the total number of estimated external highway trips increases by 5%, one could assume that the external trips also increase by 5%. While this is plausible, it is not necessarily an accurate estimate. Talk to your Metropolitan Planning Organization or the State Department of Transportation if you are interested in developing this type of model as you will need their estimates of external trips.

Figure 27.10:

"Make Prediction" Setup Page



3. On the left side of the page are listed all the variables in the input data set (primary or secondary file). In the middle box, the variables are added in the **same order** as in the matching parameters box. That is, each independent variable needs to be matched to the variable from the coefficients file, one for one. **This is very important**. The names do not have to be the same (e.g., if the model was calibrated with data set and applied to another, the variable names may not be identical). But the content and order of the variables needs to be the same. In the example, the first variable in the coefficients file is INCEQUAL. The selected variable in the middle box has to be the income equality variable (whatever its name). In the example, the same data set is being used so the names are identical. This is repeated for each of the independent variables in the coefficients file.

4. Next, any missing value codes are specified in the missing values box. Any records with a missing value for *any* of the selected independent variables will be dropped from the calculation. In the example, there are no missing value codes applied other than the default blank field.

5. If external trips are to be added, the external trips box must be checked. External trips could be applied in an origin model, but not in a destination model. If they are to be added, the number of trips should be specified in the 'Number of external trips' box and the zone ID field for the file indicated; in the example, 1627 is added as external trips and the TAZ field is specified as the ID variable (TZ98).

6. The type of model to be applied is indicated in the "Type of regression model" box. There are only two choices: Poisson (the default) and Normal (OLS). Since the coefficients are being applied to the data, no over-dispersion correction is necessary (since it was probably used in calibrating the model).

7. Finally, the output file name is defined in the 'Save predicted values' box.

For each zone, the routine will then take the appropriate variable from the input data set and apply the matching coefficient from trip generation coefficients file to produce a predicted estimate of the number of trips. To calculate this value, for the OLS model, the routine will use equation 27.2 above while for the Poisson model, the routine will use equation 27.6 above. For the latter, it will then raise the predicted log value to the power, *e*, to produce a prediction for the expected number of crime trips:

$$\lambda_i = e^{Ln(\lambda_i)} \tag{27.33}$$

If external trips are added, a new zone is created called EXTERNAL in the ID field that was indicated on the page. Then, the specified number of external trips is simply placed in that field with zeros being placed for the values of all the remaining variables in the file. By default, the output name for the predicted number of crimes will be called PREDORIG for an origin model and PREDDEST for a destination model. An example data set is available on the *CrimeStat* download page.

> Note: for a destination model, this 'Make prediction' operation is not necessarily needed if the same data set is used for calibration and prediction. This step is primarily for the origin file

Balancing Predicted Origins and Destinations

After the origin model and destination model are calibrated and applied to a data set, the final step in trip generation is to ensure that the number of predicted origins equals the number of predicted destinations. This is necessary for the next stage of crime travel demand modeling - trip distribution. Since a trip has both an origin and a destination, the total number of origins *must* equal the total number of destinations. This is an *absolute* requirement for the trip distribution model to work. The routine will return an error message if the number of origins does not equal the number of destinations.

If the Poisson model is used for calibration, the routine ensures that the number of predicted trips equals the number of input trips. Further, if the calculation of external trips has been obtained by subtracting the total number of predicted origins from the total number of predicted destinations, and if the external trips are then added to the predicted origins, then most likely the total number of origins will equal the total number of destinations. However, because of rounding-off errors and inconsistent external trip estimates, it is possible that the sums are not equal.

Consequently, it is important to balance the predicted origins and destinations to ensure that no problems will occur in the trip distribution model. There are two ways to do this in *CrimeStat*. First, the number of predicted destinations is held constant and the number of predicted origins is adjusted to match this number. This is the default choice. Second, the number of predicted origins is held constant and the number of predicted destinations is adjusted to match this number.

The calculation is essentially a multiplier that is applied to each zone. If destinations are to be held constant, the multiplier is defined as the ratio of total destinations to total origins:

$$M_j = \frac{Sum\ of\ crimes\ by\ destinations}{Sum\ of\ crimes\ by\ origins} = \frac{\sum_{j=1}^{N} X_j}{\sum_{i=1}^{M} X_i} \quad (27.34)$$

The predicted number of origins is multiplied by M_j. If, on the other hand, the origins are to be held constant, the multiplier is defined as the ratio of total origins to total destinations:

$$M_j = \frac{Sum\ of\ crimes\ by\ origins}{Sum\ of\ crimes\ by\ destinations} = \frac{\sum_{i=1}^{M} X_i}{\sum_{j=1}^{N} X_j} \quad (27.35)$$

The predicted number of destinations is multiplied by M_i. The multiplication simply ensures that the sums of the predicted origins and predicted destinations are equal.

The third page in the trip generation model is the 'Balance predicted origins & destinations' page. Figure 27.11 shows the setup for this page. The steps are as follows:

1. The box is checked indicating that it is a balancing operation.

2. The predicted origin file is input and the predicted origin variable is identified. In the example, the predicted origin file is called 'PredictedOrigins.dbf' and the field with the predicted numbers was called PREDORIG.

3. The predicted destination file is input and the predicted destination variable is identified. In the example, the predicted destination file is called 'PredictedDestinations.dbf' and the field with the predicted numbers was called PREDDEST.

Note that these files are input on this page and not on the primary or secondary file pages.

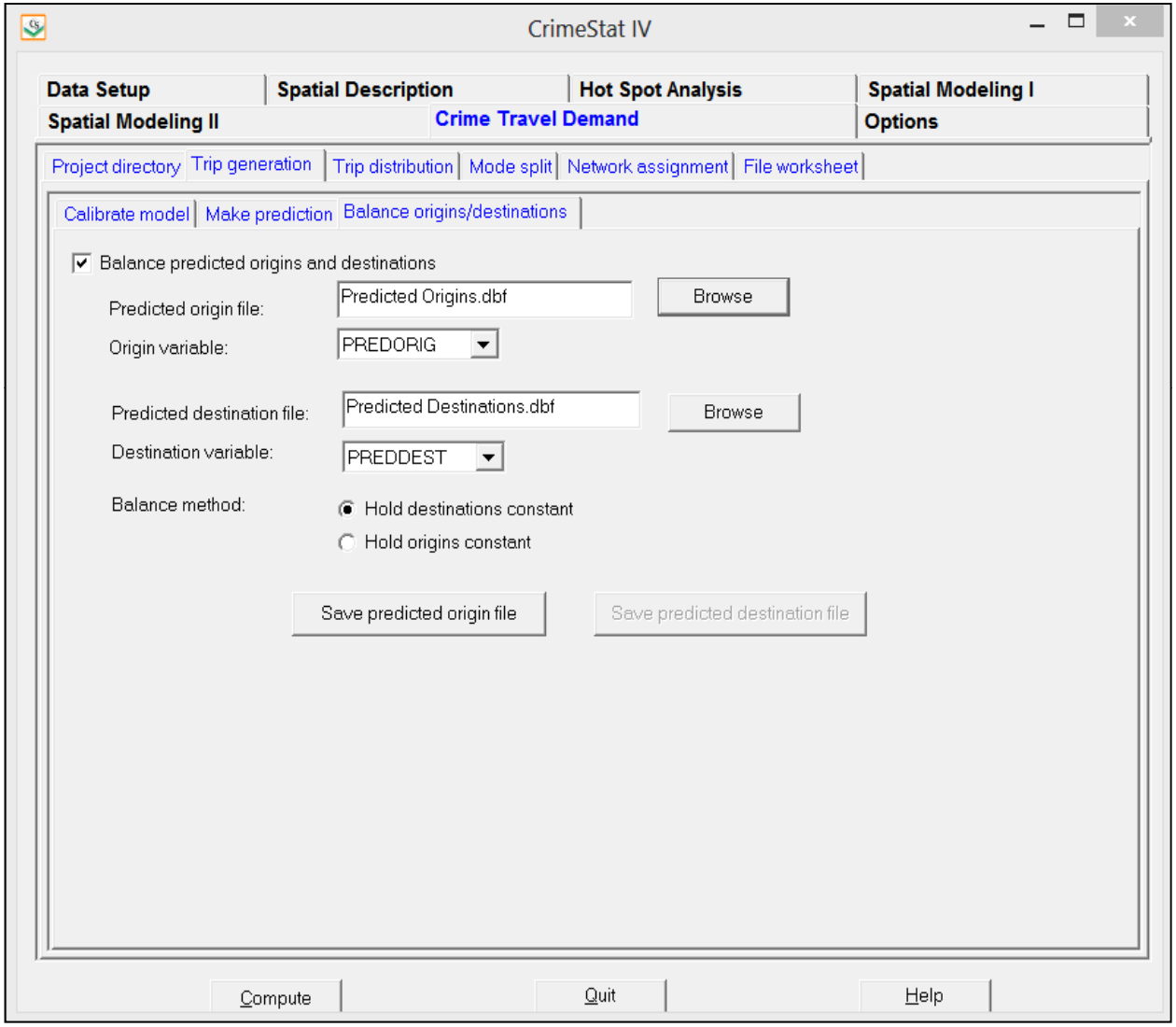
4. Next, the type of balancing is specified - Holding destinations constant (the default) or holding origins constant. In the example, the destinations are to be held constant.

5. Finally, the output file is specified. If the origins are to be adjusted, then only the origin file is saved. If the destinations are to be adjusted, then only the destination file is saved. In other words, the adjustment is applied to only one of the two predicted crime files. In the example, the file was named 'AdjustedPredictedOrigins.dbf' (not shown) since the origin file was adjusted.

The output produces a new column with the adjusted values. Table 27.9 shows the origin output for the Baltimore data of the first 11 records. Once the balancing has been completed, the trip generation model is finished and the user can go on to the trip distribution model. In other

Figure 27.11:

Balance Predicted Origins and Destinations Setup



words, the output file ensures that both the predicted origin file (crime productions) and predicted destination file (crime attractions) are balanced.

Table 27.9:
Adjusted Data Should Have These Fields

Zone	PREDICTED	ADJORIGIN
0001	225.818482	225.850955
0002	187.527819	187.554785
0003	320.877458	320.923600
0004	75.096631	75.107430
0005	44.981775	44.988243
0006	32.574758	32.579442
0007	107.334835	107.350270
0008	74.683931	74.694671
0009	76.425236	76.436226
0010	34.183846	34.188762
0011	66.975803	66.985434
etc	etc	etc

Strengths and Weaknesses of Regression Modeling of Trips

As mentioned earlier, the use of regression for producing the trip generation model has its strengths and weaknesses. The advantages are that, first, the approach is applicable to crime incidents. Unlike regular travel behavior, crime trips have to be inferred from police reports. Thus, starting with counts of the number of crimes occurring in each zone and the number of crimes that originate from each zone, a model can be constructed.

Second, the use of a non-linear model, such as the Poisson, allows more complex fitting of crime counts. In the early 1970s when trip generation models were starting to be implemented in Metropolitan Planning Organizations around the U.S., the major type of regression modeling available was OLS. At that time, researchers could not demonstrate that this method was reliable in terms of predicting travel. However, with the availability of software for conducting Poisson and other non-linear models, that criticism is no longer applicable. The Poisson model is very well behaved with respect to count data. It does not produce negative estimates. It requires high levels of an independent variable to produce a slight effect in the dependent variable, but that the level increases as the values of the independent variable increase. It maintains constancy between the sum of the input counts and the sum of the predicted counts. Non-linear models are much more realistic for modeling trips than OLS.

Third, the use of a multivariate regression model allows multiple variables to be included. In our example, there were four independent variables in the reduced origin and destination models. Trip tables, on the other hand, typically only have three or four independent predictors; it becomes too complicated to keep track of multiple conditions of predictor variables. Thus, a more complex and sophisticated model can be produced with a regression framework.

Fourth, and finally, a regression framework allows for complex interactions to be estimated. For example, the log of an independent variable can be defined. An interaction between two of the independent variables can be examined (e.g., median household income for those zones having a sizeable amount of retail employment). In the trip table approach, these interactions are implicit in the cell means. Thus, overall, the regression framework allows for a more complex model than is available with a trip table approach.

On the other hand, there are potential problems associated with a regression framework. First, the regression coefficients can be influenced by zone size. Since the model is estimating differences between zones (i.e., differences in the number of crimes as a function of differences in the values of the independent variables), zone size affects the level of those differences. With small zone sizes, there will be substantial differences between zones in both the independent and dependent variables. Conversely, large zone sizes will minimize within-zone differences, but will usually increase the estimate of the between-zone differences. The result could be an exaggeration of the effect of a variable that would not be seen with small zone geography. As was argued in Chapter 25, one should choose the smallest zone geography that is practical in order to minimize this problem.

Second, a point that has been repeated again and again, these models are not behavioral explanations. They represent ecological correlations with crime trips. It is important to not try to convert these models into explanations of offender behavior. Too often, researchers have jumped to conclusions about individuals based on the relationships with environments and neighborhoods. It is important to not do this. This criticism, incidentally, applies both to the trip table as well as the regression approach to trip generation modeling.

The new generation of travel demand models are specifically behavioral and involve modeling the behavior of specific individuals. Probabilities are calculated based on individual choice and a micro-simulation routine can apply these probabilities to a large metropolitan area (Shifton et al, 2003; Recker, 2000). While this approach offers some definite theoretical advantages and is the subject of much current research, to date there has not been a demonstration that this approach is more accurate at predicting trips than the tradition trip-based travel demand model. For crime, such an approach would have to be simulated.

Conclusion

In summary, the trip generation model is a valuable tool for predicting the number of crimes that originate in each zone and the number of crimes that end in each zone. Even if the model is not behavioral, the model can be stable and useful for many years in the future. It is best thought of as a *proxy model* in which the variables in the models are proxies for conditions that are generating crimes, either in terms of environments that produce offenders or in terms of locations that attract them.

In the next chapter, we will examine the second stage in the travel demand model - trip distribution. In that stage, the predicted crime origins and the predicted crime destinations are linked to produce crime trips.

References

Boswell, M. T. & Patil, G. P. (1970). "Chance mechanisms generating negative binomial distributions". In *Random Counts in Scientific Work*, Vol. 1, G. P. Patil, ed., Pennsylvania State University Press:University Park, PA, 3-22.

Bowers, K. & Hirschfield, A. (1999). Exploring links between crime and disadvantage in North-West England: An analysis using Geographic Information Systems. *International Journal of Geogrpahical Information Science*, 13,b 159-184.

Bursik, R. J., Jr. & Grasmick, H. G. (1993). Economic deprivation and neighborhood crime rates, 1960-1980. *Law and Society Review*, 27, 263-268.

Cameron, A. C. & Trivedi, P.K. (1998). *Regression Analysis of Count Data*. Cambridge University Press: Cambridge, U.K.

Cameron, A. C. & Windmeijer, F. A. G. (1996). R-squared measures for count data regression models with applications to health-care utilization. *Journal of Business & Economic Statistics*, 14(2), 209-20.

Chiricos, T. (1987). Rates of Crime and Unemployment *Social Problems*, 34, 187-211

Cohen, L.E. & Felson, M. (1979) Social change and crime rate trends: a routine activity approach, *American Sociological Review*, 44: 588-608.

Culp, M. & Lee, E. J. (2005). Improving travel models through peer review. *Public Roads*, 68 (6), FHWA-HRT-05-005. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/publications/publicroads/05may/07.cfm. Accessed April 28, 2012.

Der, G. & Everitt, B. S. (2002). *A Handbook of Statistical Analyses using SAS*. Chapman & Hall/CRC: London.

Draper, N. & Smith, H. (1981). *Applied Regression Analysis, Second Edition*. John Wiley & Sons: New York.

Ehrlich, I. (1975). On the relation between education and crime. In F. T. Juster (ed), *Education, Youth and Human Behavior*. McGraw-Hill: New York, 313-337.

Fotheringham, A. S., Brunsdon, C. & Charlton, M. (2002). *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. John Wiley & Sons: New York.

References (continued)

Fowles, R. & Merva, M.. (1996). Wage Inequality and Criminal Activity, *Criminology*, 34, 163-82.

Freedman, David A. (1999). Ecological inference and ecological fallacy. *International Encyclopedia of the Social and Behavioral Sciences*, Technical Report No. 549, October. http://www.stanford.edu/class/ed260/freedman549.pdf. Accessed March 26, 2012.

Hagan, J. & Peterson, R. (1994). *Inequality and Crime*. Stanford University Press: Palo Alto, CA.

Hall, D. B. (2000). Zero-inflated Poisson and binomial regression with random effects: a case study. *Biometrics*, 56, 1030-1039.

Hensher, D. A. & Button, K. J. (2002). *Handbook of Transport Modeling*. Elsevier Science: Cambridge, UK.

ITE (2003). *Trip Generation* (7th edition). Institute of Transportation Engineers: Washington, DC.

Kohfeld, C. W. & Sprague, J. (1988). Urban unemployment drives crime. *Urban Affairs Quarterly*, 24, 215-241.

Langbein, L. I. & Lichtman, A. J. (1978). *Ecological Inference*. Sage University Paper series on Quantitative Applications in the Social Sciences, series no. 07-010. Beverly Hills and London: Sage Publications.

Messner, S. (1986). Economic inequality and levels of urban homicide, *Criminology*, 23, 297-317.

Miaou, S.P (1996). *Measuring the Goodness-of-Fit of Accident Prediction Models*. FHWA-RD-96-040. Federal Highway Administration, U.S. Department of Transportation: Washington, DC.

Microsoft (2012). SKEW - skewness function, *Microsoft Office Excel 2010, Microsoft: Redmond, WA. http://office.microsoft.com/en-us/excel-help/skew-HP005209261.aspx. Accessed May 21, 2012.*

Nannen, V. (2003). *The Paradox of Overfitting*. Artificial Intelligence, Rijksuniversitat: Groningen, Netherlands. http://volker.nannen.com/pdf/the_paradox_of_overfitting.pdf. Accessed March 11, 2010.

References (continued)

NCHRP (1998). *Integration of Land Use Planning with Multimodal Transportation Planning*. Project 8-32(3). Prepared by Parsons Brinkerhoff Quade & Douglas, Inc. for the National Cooperative Highway Research Program, Transportation Research Board, National Research Council: Washington DC. October.

NIST (2004). Gallery of distributions. *Engineering Statistics Handbook*. National Institute of Standards and Technology: Washington, DC. http://www.itl.nist.gov/div898/handbook/eda/section3/eda366.htm. Accessed May 21, 2012.

Newman, O. (1972). *Defensible Space: Crime Prevention Through Urban Design*. Macmillan: New York.

Ortuzar, J. D. & Willumsen, L. G. (2001). *Modeling Transport* (3rd edition). J. Wiley & Sons: New York.

Park, R. & Burgess, E. (1924). *Introduction to the Science of Sociology*. Chicago University Press: Chicago.

Pribyl, O & Goulias, K. G. (2005). Simulation of **d**aily **a**ctivity **p**atterns **i**ncorporating **i**nteractions **w**ithin **h**ouseholds: Algorithm overview and performance. *Transportation Research Record*, 1926 (January), 135-141. http://trb.metapress.com/content/r7u36h005758h304/**.** Accessed May 9, 2012.

Radford, N. (2006). The problem of overfitting with maximum likliehood . CSC 411: Machine Learning and Data Mining, University of Toronto: Toronto, CA. http://www.cs.utoronto.ca/~radford/csc411.F06/10-nn-early-nup.pdf Accessed March 11, 2010.

Ratcliffe, J.H. (2008). The magnitude of the crime challenge (Chapter 3). *Intelligence-Led Policing*, Willan Publishing: Cullompton.

Recker, W. (2000). A bridge between travel demand modeling and activity-based travel analysis. *Center for Activity Systems Analysis*. Paper UCI-ITS-AS-WP-00-11. http://repositories.cdlib.org/itsirvine/casa/UCI-ITS-AS-WP-00-11/. Accessed May 23, 2012.

Shaw, C. R. & McKay, H. D. (1942). *Juvenile Delinquency in Urban Areas*. Chicago: University of Chicago Press.

References (continued)

Shifton, Y., Ben-Akiva, M., Proussaloglu, K., de Jong, G., Popuri, Y., Kasturirangan, K., & Bekhor, S. (2003). Activity-based modeling as a tool for better understanding travel behaviour. *Conference Proceedings*. 10th International Conference on Travel Behaviour Research, Lucerne, Switzerland. August. http://www.ivt.ethz.ch/news/archive/20030810_IATBR/shiftan.pdf. Accessed May 23, 2012.

Shoup, D. (2002). Roughly right vs. precisely wrong. *Access*, No. 20, Spring. 20-25.

Stack, S. (1984). Income inequality and property crime, *Criminology*, 22, 229-257.
Thrasher, F. M. (1927). *The Gang*, University of Chicago Press: Chicago.

Venables, W.N. & Ripley, B. D. (1997). *Modern Applied Statistics with S-Plus (second edition)*. Springer-Verlag: New York.

Wilson, J.Q. & Kelling, G. (1982) Broken Windows: The Police and Neighborhood Safety. *Atlantic Monthly*, March. 29-38.

Chapter 28:

Crime Trip Distribution

Ned Levine
Ned Levine & Associates
Houston, TX

Richard Block
Loyola University
Chicago, IL

Dan Helms
Scytale Consulting
Reston, VA

Phil Canter
Towson University
Towson, MD

Table of Contents

Table of Contents (continued)

Chapter 28:
Crime Trip Distribution

In this chapter, the mechanics of the second crime travel demand modeling stage -trip distribution, is explained. *Trip distribution* is a model of the number of trips that occur between each origin zone and each destination zone. It uses the predicted number of trips originating in each origin zone (trip production model) and the predicted number of trips ending in each destination zone (trip attraction model). Thus, trip distribution is a model of travel between zones - trips or links. The modeled trip distribution can then be compared to the actual distribution to see whether the model produced a reasonable approximation.

Theoretical Background

The theoretical background behind the trip distribution module is presented first. Next, the specific procedures and tests are discussed with the model being illustrated with data from Baltimore County.

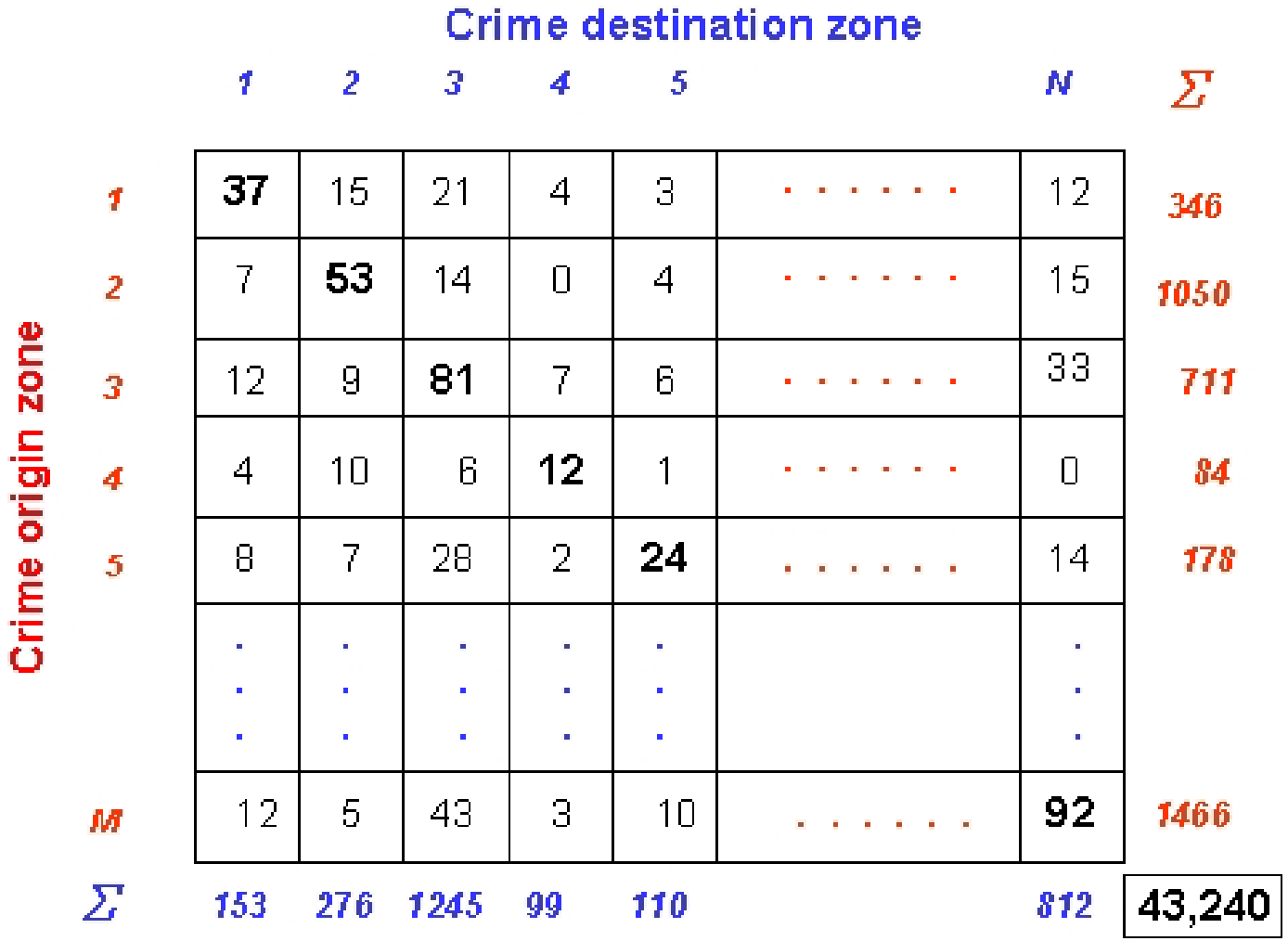
Logic of the Model

Trip distribution usually occurs through an allocation model that splits trips from each origin zone into distinct destinations. That is, there is a matrix which relates the number of trips originating in each zone to the number of trips ending in each zone. Figure 28.1 illustrates a typical arrangement. In this matrix, there are a number of origin zones, M, and a number of destination zones, N. The origin zones include ***all*** the destination zones but may also include additional ones. The reasons that there would be different numbers of zones for the origin and destination models are that crime data for other jurisdictions are not available but that many crimes that occurred in the study jurisdiction were committed by individuals who lived in other jurisdictions.

For example, with crimes that occurred in Baltimore County, approximately 35% were committed by offenders who lived in the City of Baltimore. Thus, it is important to include the City of Baltimore as an originating area for Baltimore County crimes. Hence, there are 325 destinations zones for Baltimore County while the origin zones include both the 325 in Baltimore County and 207 more from the adjacent City of Baltimore. If it were possible to obtain crime data for the City of Baltimore, then it would be possible to have the same number of zones for both the origin file and the destination file. As Chapter 26 pointed out, the study area should extend beyond the modeling area until the origins of at least 95% of all trips ending in the study area are counted.

Figure 28.1:

Example Crime Origin-Destination Matrix



Each cell in the matrix indicates the number of *trips* that go from each origin zone to each destination zone. To use the example in Figure 28.1, there were 15 trips from zone 1 to zone 2, 21 trips from zone 1 to zone 3, and so forth. Note that the trips are asymmetrical; that is, trips in one direction are different than trips in the opposite direction. To use the table, there were 15 trips from zone 1 to zone 2, but only 7 trips from zone 2 to zone 1.

The trips on the diagonal are *intra-zonal* trips, trips that originate and end in the same zone. Again, to use the example above, there were 37 trips that both originated and ended in zone 1, 53 trips that both originated and ended in zone 2.

In such a model, constancy is maintained in that the number of trips originating from all origins zones *must equal* the number of trips ending in all destination zones. This is the fundamental balancing equation for a trip distribution. In equation form, it is expressed as:

$$\sum_{i=1}^{M} O_i = \sum_{j=1}^{N} D_j \qquad (28.1)$$

where the origins, O_i, are summed over M origin zones while the destinations, D_j, are summed over N destination zones. To use the example in Figure 28.1, the total number of origins is equal to the total number of destinations, and is equal to 43,240.

The balancing equation is implemented in a series of steps that include modeling the number of crimes originating in each zone, adding in trips originating from outside the study area (external trips), and statistically balancing the origins and destinations so that equation 28.1 holds. This was done in the trip generation stage. But, it is essential that the step should have been completed for the trip distribution to be implemented.

Observed and Predicted Distributions

There are two trip distribution matrices that need to be distinguished. The first is the *observed* (or empirical) distribution. This is the actual number of trips that are observed traveling between each origin zone and each destination zone. In general, with crime data, such an empirical distribution would be obtained from an arrest record where the residence (or arrest) location of each offender is listed for each crime that the offender was charged with. In this case, the residence/arrest location would be considered the origin while the crime location would be considered the destination.

In Chapter 26, it was mentioned that there is always uncertainty as to the true origin location of a crime incident, whether the offender actually traveled from the residence location to the crime location or even whether the offender was actually living at the residence location.

But absent any alternative evidence, a meaningful distribution can still be obtained by simply treating the residence location as an approximate origin.

The observed distribution is calculated by simply enumerating the number of trips by each origin-destination combination. This is sometimes called a *trip link* (or trip pair). The second distribution, however, is a *model* of the trip distribution matrix. This is usually called the *predicted* distribution. In this case, a simple model is used to approximate the actual empirical distribution. The trips originating in each origin zone are allocated to destination zones usually on the basis of being directly proportional to attractions and inversely proportional to costs (or impedance).

Thus, a model of the trip distribution is produced that approximates the actual, empirical distribution. There are a number of reasons why this would be useful - to be able to apply the model to a different data set from which it was calibrated, to use the model for evaluating a policy intervention, or to use the model for forecasting future crime trip distribution. But, whatever the reason, it has to be realized that the model is not the observed distribution. There will always be a difference between the observed distribution from which a model is constructed and the resulting predicted distribution of the model. It is useful to compare the observed and predicted model because this allows a test of the validity of the impedance function. But, rarely, if ever, will the predicted distribution be identical to the empirical distribution.

Another way to think of this is that the actual distribution of crime trips is complex, representing a large number of different decisions on the part of offenders who do not necessarily use the same decision logic. The model, on the other hand, is a simple allocation on the basis of three or, sometimes, four variables. Almost by definition, it will be much simpler than the real distribution. Still, the simple model can often capture the most important characteristics of the actual distribution. Hence, modeling can be an extremely useful analytical exercise that allows other types of questions to be asked that are not possible with just the observed distribution.

The Gravity Model

A model that is usually used for trip distribution is that of the *gravity function*, an application of Newton's fundamental law of attraction (Oppenheim, 1980; Field & MacGregor, 1987; Ortuzar & Willumsen, 2001). Much of the discussion below is also repeated in Chapter 13 on journey-to-crime modeling since there is a common theoretical basis. In the original Newtonian formulation, the attraction, F, between two bodies of respective masses M_1 and M_2, separated by a distance D, will be equal to

$$F = g\frac{M_1 M_2}{d^2} \tag{28.2}$$

where *g* is a constant or scaling factor which ensures that the equation is balanced in terms of the measurement units (Oppenheim, 1980). As we all know, of course, *g* is the gravitational constant in the Newtonian formulation. The numerator of the function is the *attraction* term (or, alternatively, the attraction of M_2 for M_1) while the denominator of the equation, d^2, indicates that the attraction between the two bodies falls off as a function of their *squared* distance. It is an *impedance* (or resistance) term.

Social Applications of the Gravity Concept

The gravity model has been the basis of many applications to human societies and has been applied to social interactions since the 19th century. Ravenstein (1895) and Andersson (1897) applied the concept to the analysis of migration by arguing that the tendency to migrate between regions is inversely proportional to the squared distance between the regions. Reilly's 'law of retail gravitation' (1929) applied the Newtonian gravity model directly and suggested that retail travel between two centers would be proportional to the product of their populations and inversely proportional to the square of the distance separating them:

$$I_{ij} = \alpha\frac{P_1 P}{d_{ij}^2} \tag{28.3}$$

where I_{ij} is the interaction between centers *I* and *j*, P_i and P_j are the respective populations, d_{ij} is the distance between them raised to the second power and α is a balancing constant. In the model, the initial population, P_i, is called a *production* while the second population, P_j, is called an *attraction*.

Stewart (1950) and Zipf (1949) applied the concept to a variety of phenomena (migration, freight traffic, information) using a simplified form of the gravity equation:

$$I_{ij} = K\frac{P_1 P}{d_{ij}} \tag{28.4}$$

where the terms are as in equation 28.3 but the exponent of distance is only 1. Given a particular pattern of interaction for any type of goods, service or human activity, an optimal location of facilities should be solvable.

In the Stewart/Zipf framework, the two P's were both population sizes. However, in modern use, it is not necessary for the productions and attractions to be identical units (e.g., P_i could be population while P_j could be employment).

Trips as Interactions

It should be obvious that this interaction equation can be applied to trips from one area (zone) to another. Changing the symbols slightly, the total volume of trips from a particular origin zone, i, to a single location, j, is directly proportional to the product of the productions at *i* and the attractions at *j*, and inversely proportion to the impedance (or cost) of travel between the two zones:

$$T_{ij} = \frac{\alpha P_1 \beta P}{d_{ij}} \tag{28.5}$$

where P_i are the productions for zone I, A_j are the attractions zone j, α is a production constant, β is an attraction constant, and d_{ij} is the impedance (cost) of travel between zone ii and zone j.

Over time, the concept has been generalized and applied to many different types of travel behavior. For example, Huff (1963) applied the concept to retail trade between zones in an urban area using the general form of:

$$A_{ij} = \alpha \frac{S_j^{\lambda}}{d_{ij}^{\rho}} \tag{28.6}$$

where A_{ij} is the number of purchases in location *j* by residents of location$i$, S_j is the attractiveness of zone j (e.g., square footage of retail space), d_{ij} is the distance between zones i *and* j, α is a constant, λ is the exponent of S_j, and ρ is the exponent of distance (Bossard, 1993). $D_{ij}^{-\rho}$ is sometimes called an *inverse distance* function. This differs from the traditional gravity function by allowing the exponents of the production from location i, the attraction from location j, and the distance between zones, d, to vary.

Equation 28.6 is a *single constraint* model in that only the attractiveness of a commercial zone is constrained, that is the sum of all attractions for *j* must equal the total attraction in the region. Again, it can be generalized to all zones by, first, estimating the total trips generated from one zone, *i*, to another zone, *j*,

$$T_{ij} = \alpha \frac{P_i^{\lambda} A_j^{\tau}}{d_{ij}^{\rho}} \tag{28.7}$$

where T_{ij} is the interaction between two locations (or zones), P_i is productions of trips from zone i, A_j is the attractiveness of zone j, d_{ij} is the distance between zones i and$j$$i$, λ is the exponent of P_i, τ is the exponent of A_j, ρ is the exponent of distance, and α is a constant.

Second, the total number of trips generated by a single location, i, to all destinations is obtained by summing over all destination locations, j:

$$T_i = \alpha P_i^{\lambda} \sum_{j=1}^{N} \frac{A_j^{\tau}}{d_{ij}^{\rho}} \tag{28.8}$$

and generalizing this to all zones, we get:

$$T_{ij} = \frac{\alpha P_i^{\lambda} \beta A_j^{\tau}}{d_{ij}^{\rho}} \tag{28.9}$$

where α is a constant for the productions, P_i^{λ} and β is a constant for the attractions, A_j^{τ}. This type of function is called a *double constraint* model because the equation has to be constrained by the number of units in both the origin and destination locations; that is, the sum of P_i over all locations must be equal to the total number of productions while the sum of A_j over all locations must be equal to the total number of attractions. Adjustments are usually required to have the sum of individual productions and attractions equal the totals (usually estimated independently).

Negative Exponential Distance Function

One of the problems with the traditional gravity formulation is in the measurement of travel impedance (or cost). For locations separated by sizeable distances in space, the gravity formulation can work properly. However, as the distance between locations decreases, the denominator approaches infinity. Consequently, an alternative expression for the interaction uses the negative exponential function (Hägerstrand, 1957; Wilson, 1970).

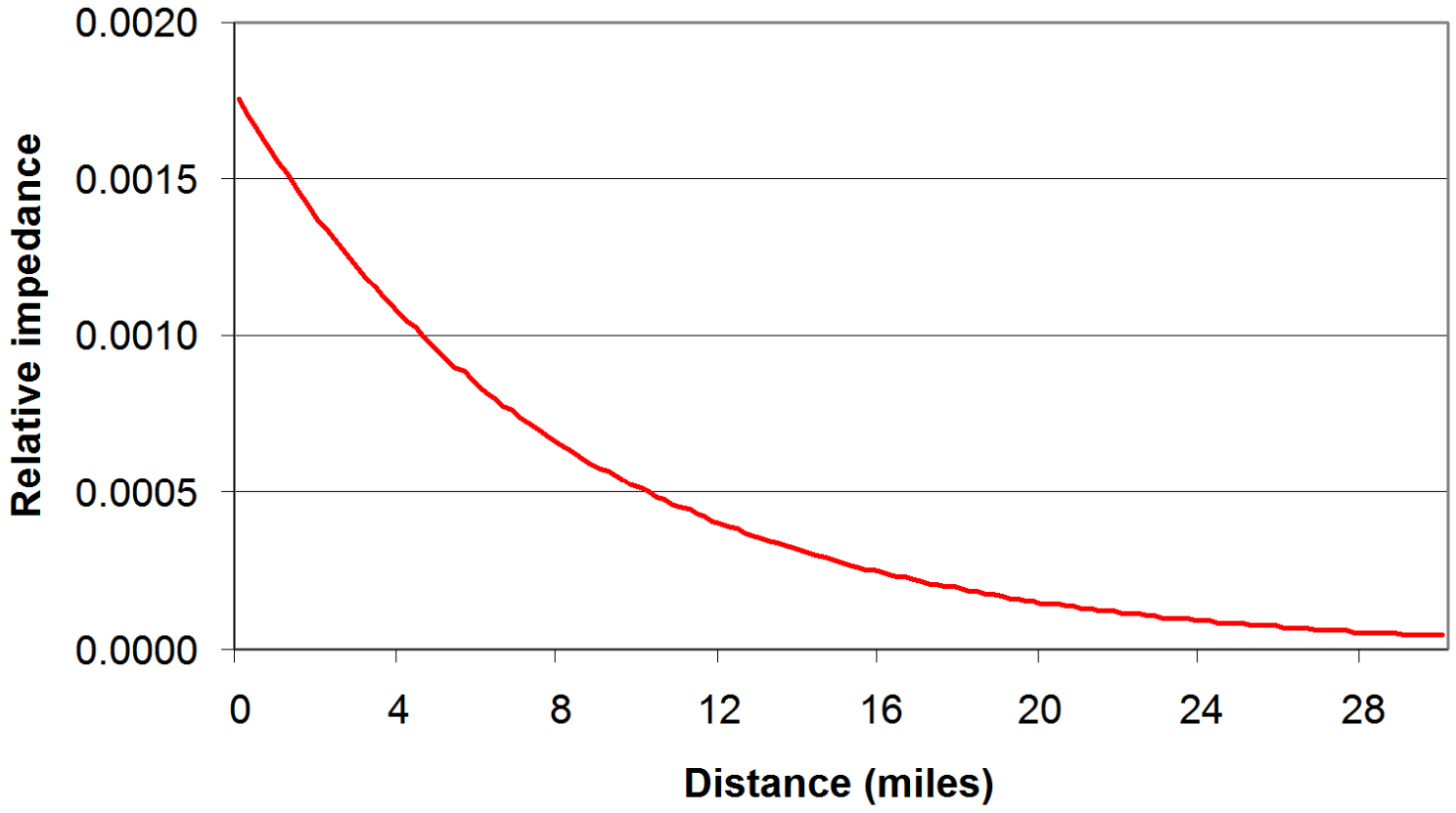
$$T_{ij} = \beta A_j^{\lambda} e^{-\alpha d_{ij}} \tag{28.10}$$

where T_{ji} is the attraction of location j for residents of location I, A_j is the attractiveness of location j, d_{ij} is the distance between locations i and *j*, *β* is the exponent of A_j, *α* is a coefficient of d_{ij} (and, also, an exponent) and *e* is the base of the natural logarithm (i.e., 2.7183...). Derived from principles of *entropy maximization*, the latter part of the equation is a negative exponential function that has a maximum value of 1 (i.e., $e^{-0} = 1$; Wilson, 1970). This has the advantage of making the equation more stable for interactions between locations that are close together. For example, Cliff and Haggett (1988) used a negative exponential gravity-type model to describe the diffusion of measles into the United States from Canada and Mexico. It has also been argued that the negative exponential function generally gives a better fit to urban travel patterns, particularly those by automobile (Bossard, 1993; Foot, 1981). Figure 28.2 shows a typical negative exponential function and one recommended for home-based work trips by the Transportation Research Board as a default value (NCHRP, 1995).

Figure 28.2:

Default Home-Based Work Trip Impedance

(Source: National Cooperative Highway Research Program 365, 1995)



Note that by moving the distance term to the numerator, strictly speaking it no longer is an impedance term since impedance increases with distance. Rather it is a *discount* factor (or *disincentive*); the interaction is discounted with distance. Nevertheless, the term 'impedance' is still used primarily for historical reasons.

There are other distance functions, as well. Chapter 13 explored some of these. For example, we are finding that, for crime trips, the lognormal function may produce better results than the negative exponential primarily because many crimes are committed at short-to-moderate distances. Chapter 17 discusses the MCMC Poisson-lognormal regression model which is useful with a low mean (e.g., very short distance traveled) and small sample sizes. It is possible that the lognormal function is more useful for very localized crime trips than the negative exponential.

Travel Impedance

One of the biggest advances in the negative exponential model of equation 28.10 has been to increase the flexibility of the denominator. In the traditional gravity model, the denominator is distance. This is a proxy for a *discount factor* (or cost); the farther two zones are from each other, the less likely there is to be interaction between them, all other things being equal. Conversely, the closer two zones are, the more likely there is to be interaction, all other things being equal.

Distance v. Travel Time

It has been realized, however, that distance is only an approximation for impedance. In real travel, travel time is a much better indicator of the *cost* of travel in that time varies by the time of day, day of week, direction of travel, type of road used, and other factors. For example, travel across town in any metropolitan area is generally a lot easier at 3 in the morning, say, than at the peak afternoon rush period. The difference in travel time can vary as much as two-to-three times between peak and off-peak hours. Using only distance, however, these variations are never picked up because the distance between locations is invariant.

This realization has led to the concept of *travel impedance* which, in turn, has led to the concept of *travel cost*. 'Impedance' is the resistance (or discounting) in travel between two zones. Using travel time as an impedance variable, the longer it takes to travel between two zones, the less likely there will be interaction between them, all other things being equal. Conversely, a shorter travel time leads to greater interaction between zones, again, all other things being equal. Similarly, a travel route that shortens travel time will generally be selected over one that takes longer even if the first one is longer in distance. For example, it has been

documented that people will change work locations that are farther from their home if traveling to the new work location takes less time (e.g., traveling in the 'opposite' direction to the bulk of traffic; Wachs, Taylor, Levine & Ong, 1993).

If travel time is a critical component of travel, why then don't offenders commit more crimes at, say, 3 in the morning than at the peak afternoon travel times? Since the impedance is less at 3 in the morning than at, say, 5 in the afternoon, would not the model predict more trips occurring in the early morning hours than actually occur in those hours? The answer has to do with the numerator of the gravity equation and not just the denominator. At 3 in the morning, yes, it is easier to travel between two locations, at least by personal automobile (not by bus or train when those services are less frequent). But the attraction side of the equation is also less strong at 3 in the morning. For a street robber, there are fewer potential 'victims' on the street at 3 in the morning than in the late afternoon. For a residential burglar, there is more likely to be someone at home at night than in the afternoon. The travel time component is only one dimension of the likelihood of travel between two locations. The distribution of opportunities and other costs can alter the likelihood considerably.

Nevertheless, shifting to an impedance function allows a travel model to better replicate actual travel conditions. Most travel demand models used by transportation planners use an impedance function, rather than a distance function.[1] Distance would only be meaningful if the standards were invariant with respect to time (e.g., a model calculated over an entire year, 24 hours a day). As will be demonstrated in Chapter 30 on network assignment, a travel time calculation leads to a very different network allocation than a distance calculation. For example, if distance is used as an impedance variable, then the shortest trips will rarely take the freeways because travel to and from a freeway usually makes a trip longer than a direct route between an origin and a destination. But as most people understand, taking a freeway to travel a sizeable distance is usually a lot quicker than traversing an urban arterial system with many traffic lights, stop signs, crossing pedestrians, cross traffic from parking lots and shopping malls, and other urban 'obstacles'. Today, the use of distance in travel demand modeling has virtually been dropped by most transportation planners.

---

[1] Distance can be used as a rough approximation for impedance, but is rarely a good predictor of actual travel behavior. For example, in the mode split mode that will be discussed in Chapter 29, the distance between a location and the nearest bus or rail route can be used to quickly select trip pairs that might travel by transit. However, the actual prediction must be based on a network calculation of travel time or travel cost in traversing the system.

Travel Cost

An even better concept of impedance is that of *travel cost* (sometimes called *generalized cost*) which incorporates real and perceived costs of travel between two locations. Travel time is one component of travel cost in that there is an implicit cost to the trip (e.g., an hourly wage or price assigned). In this case, two different individuals will value the time for a trip differently depending on their hourly 'wage'. For example, for an individual who prices his/her travel at $100 an hour, the per minute cost is $1.67. For another individual who prices his/her travel at $12 an hour, the per minute cost is 20¢. These relative prices assigned to travel will substantially affect individual choices in travel modes and routes. For instance, these two hypothetical individuals will probably use a different travel mode in getting from an airport to a hotel on a trip; the former will probably take a taxi whereas the latter will probably take a bus or train (if available).

But cost involves other dimensions that need to be considered. There are real operating costs in the use of a vehicle - fuel, oil, maintenance, and insurance. Many travel studies have suggested that drivers incorporate these costs as part of their implicit hourly travel price (Ortuzar & Willumsen, 2001; 323-327). But, there are also real, 'out-of-pocket' costs such as parking or toll costs. Parking is particularly a major expense for intra-urban driving behavior. In many built-up business districts, parking costs can be considerable, for example as much as $90 a day in major metropolitan centers. In most busy commercial areas, there are some parking costs, if only at on-street parking meters. Thus, a travel cost model needs to incorporate these real costs as the out-of-pocket costs may overwhelm the implicit value of the travel time. For example, an offender who lives 10 minutes from the downtown area by car would probably not drive into the downtown to commit a robbery since that individual will have to bear the price of parking. There are lots of well known stories that circulate about bank robbers who are caught because they incur parking tickets while committing their crime. How often this has occurred is not known from any study that we are aware, but the story line is cognizant of the actual costs of travel that must be incurred as part of travel.

In addition to real costs are perceived costs. For transit users, particularly, these perceived costs affect the ease and time of travel. One of the standard questions in travel surveys for transit users is the time it takes to walk from their home to the nearest bus stop or intra-urban rail system (if available) and from the last transit stop to their final destination; the longer it takes to access the transit system, the less likely an individual will use it. Similarly, transfers between buses or trains decrease the likelihood of travel by that mode, almost in proportion to the number of transfers. The reason is the difficulty in getting out of one bus or train and into another. But, the time between trains adds an implicit travel cost; the longer the wait between buses, the less likely that mode will be used by travelers. In short, ease of access and convenience are positive incentives in using a mode or a route while difficulty in accessing

it, lack of convenience, and even fear of being vulnerable to crime will decrease the likelihood of using that mode or route.[2]

If the concept is expanded to that of an offender, there are other perceived costs that might affect travel. One obvious one is the likelihood of being caught. It may be easy for one offender to travel to an upscale, high visibility shopping area, but if there are many police and security guards around, the individual is more likely to be caught. Hence, that likelihood (or, more accurately, an assumption that the offender makes about that likelihood since he/she does not really know what is the real likelihood) is liable to affect the choice of a destination and, possibly, even a route.

Another perceived cost is the likelihood of retaliation from other gangs. Bernasco and Block (2009) showed that robbers in Chicago will usually not commit robberies in the territories of rival gangs even if those areas are closer to where the robbers live.

Another perceptual component affecting a likely choice is the reliability of the transportation mode. Many offenders are poor and do not have expensive, well maintained vehicles. If the vehicle is not capable of higher speeds or is even likely to break down while an offence is being committed that vehicle is not liable to be used in making a trip or the choice of destination may be altered. It is well known that many offenders steal vehicles for use in a crime. Fears about not being identified are clearly a major factor in those decisions, but the reliability of their own vehicles may also be a factor.

Thus, in short, a more realistic model of the incentives or disincentives to make a trip between two locations requires a complex function that weights a number of factors affecting the cost of travel - the travel time, implicit operating costs, out-of-pocket costs, and perceived costs. Many travel demand models used by Metropolitan Planning Organizations use such a function, usually under the label of 'generalized cost'. The more complex the pricing structure for parking and travel within a metropolitan area, the more likely a generalized cost function will provide a realistic model of trip distribution.

Travel Utility

The final concept that is introduced in defining impedance is that of *travel utility*. 'Utility' is an individual concept, rather than a zonal one. Also, it is the flip side of cost (i.e., higher cost is associated with less utility). A generalized cost function calculates the objective

2 Most of the research on factors affecting use of transit were conducted in the 1960s and 1970s. These assumptions are more or less assumed by travel demand modelers, rather than documented *per se*. See Schnell, Smith, Dimsdale, & Thrasher, 1973; Roemer & Sinha, 1974; WASHCOG, 1974; Carnegie-Mellon University, 1975; Johnson, 1978; Levine & Wachs, 1986 for some examples.

and average perceived costs of travel between two zones. But the utility of travel for an individual is a function of both those real costs and a number of individual characteristics that affect the value placed on that travel. Thus, two individuals living in the same zone (perhaps even living next door to each other) who travel to the same destination location may 'price' their trip very differently. Aside from income differences which effect the average hourly 'wage', there may be differences due to convenience, attractiveness, or a host of other factors. Other factors are more idiosyncratic. For example, a trip by a gang member into another gang's 'turf' might be expected to increase the perceived costs to the individual of traveling to that location, above and beyond any objective cost factors. Alternatively, a trip to a location where a close friend or relative is located might decrease the perceived cost of travel to that zone. In other words, there are both objective costs as well as subjective costs in travel between two zones.

The concept of utility may be less useful for crime analysis than for general travel behavior. For one thing, since the concept is individual, it can only be identified by individual surveys (Domencich & McFadden, 1975). For crime analysis, this makes it virtually impossible to use since it is very difficult to interview offenders, at least in the United States. But, for completeness sake, we need to understand that the likelihood or disincentive to travel between two locations is a function of individual characteristics as well as objective travel cost components. A mixture of aggregate and individual variables can be used to produce a synthetic utility model for modeling locations where individuals commit crimes (Block & Bernasco, 2009).

The modeling of individual utility can be done with either a multinomial logit model for a limited number of discrete choices or a more general conditional logit model for many choices. Chapter 21 discusses these models while Chapter 22 presents the CrimeStat discrete choice module. At this point, it is impractical to utilize either model for predicting trip distribution links since the number of origin-destination pairs would require an enormous data set. So, we are left for the time being with the gravity function being the only practical approach to trip distribution.

Impedance Function

For a zonal type model, we can think of the gravity function as a generalized impedance function. For travel between any one zone and all other zones, we have:

$$T_i = \alpha P_i^{\lambda} \sum_{j=1}^{N} \frac{A_j^{\tau}}{I_{ij}} \qquad (28.11)$$

where the number of trips from zone i to all other zones is a function of the productions at zone i and the relative attraction of any one zone, j, to the impedance of that zone for i, I_{ij}. The

impedance function, I_{ij} , is some declining function of cost for travel between two zones. It does not have to be any particular form and can be (and usually is) a non-linear function. The costs can be in terms of distance, travel time, speed (which is converted into travel time) or general costs. The greater the separation between two zones (i.e., the higher the impedance), the less likely there will be a trip between them. Generalizing this to all zones, we get:

$$T_{ij} = \alpha P_i^{\lambda} \beta \frac{A_j^{\tau}}{I_{ij}} \tag{28.12}$$

where P_i is the production capacity of zone i, A_j is the attraction of zone j, I_{ij}, is a generalized function that discounts the interaction with increasing separation in distance, time, or cost, α and β are constants that are applied to the productions and attractions respectively, and λ and τ are 'fine tuning' exponents of the productions and attractions respectively. This is the gravity function that we will estimate in the *CrimeStat* trip distribution model.

Alternative Model: Intervening Opportunities

There are alternative allocations procedures to the gravity model. One well known one is that of *intervening opportunities*. Stouffer (1940) modified the simple gravity function by arguing that the attraction between two locations was a function not only of the characteristics of the relative attractions of two locations, but of intervening opportunities between the locations. His hypothesis "..assumes that there is no necessary relationship between mobility and distance... that the number of persons going a given distance is directly proportional to the number of opportunities at that distance and inversely proportional to the number of intervening opportunities" (Stouffer, 1940, p. 846). This model was used in the 1940s to explain interstate and inter-county migration (Isard, 1979; Isbell, 1944; Bright & Thomas, 1941). Using the gravity type formulation, this can be written as:

$$A_{ji} = \alpha \frac{S_j^{\beta}}{\sum_{k=1}^{O} s_k^{\xi} d_{ij}^{\lambda}} \tag{28.13}$$

where A_{ji} is the attraction of location j by residents of location i, S_j is the attractiveness of zone j, S_k is the attractiveness of all other locations, k, that are *intermediate* in distance between locations i and j (with there being O such locations), d_{ij} is the distance between zones i and j, β is the exponent of S_j, ξ is the exponent of S_k, and λ is the exponent of distance. While the intervening opportunities are implicit in equation 28.7 in the exponents, β and λ, and coefficient, α, equation 28.13 makes the intervening opportunities explicit. The importance of the concept is that travel between two locations becomes a complex function of the spatial environment of nearby areas and not just of the two locations.

In practice, in spite of its more intuitive theoretical model, the intervening opportunities model does not improve prediction much beyond that of the gravity model since it includes the attractions associated with the destination zones. Also, it is a more difficult model to estimate since the attractions of all other zones must be considered for each zone pair (origin-destination combination). Consequently, it is rarely used in actual practice (Ortuzar & Willumsen, 2001).

Another alternative method was conducted by Porojan (2000) in applying the gravity model to international trade flow. He added a spatial autocorrelation component in addition to impedance and obtained a slightly better fit than the pure gravity function. However, whether this approach would improve the fitting of intra-regional crime travel patterns is still unknown. Nevertheless, this and other approaches might improve the predictability of a gravity function for intra-urban crime travel.

Method of Estimation

The *CrimeStat* trip distribution model implements equation 28.12. The specific details are discussed below, but the model is iterative. The steps are as follows:

1. Depending on whether a singly constrained or doubly constrained model is to be estimated, it starts with aN initial guess of the values for α or β (or both for a doubly constrained model). Table 28.1 illustrates the three models.

Table 28.1:
Three Methods of Constraining the Gravity Model

Single constraint

Constrain origins:

$$T_{ij} = \alpha P_i^{\lambda} A_j^{\tau} I_{ij} \quad (28.14)$$

Constrain destinations:

$$T_{ij} = P_i^{\lambda} \beta A_j^{\tau} I_{ij} \quad (28.15)$$

Double constraint

Constrain both origins and destinations:

$$T_{ij} = \alpha P_i^{\lambda} \beta A_j^{\tau} I_{ij} \quad (28.16)$$

2. The routine proceeds to estimate the value for each cell in the origin-destination matrix (see Figure 28.1 above) using the existing estimates for α and β.

3. The routine then sums the rows and columns in the matrix. Then, depending on whether a single- or double-constraint model is to be estimated and, if a single-constraint, whether origins or destinations are to be held constant, it then calculates the ratio of the summed values (row totals or column totals or both) to the initial row or column sum. The inverse of that ratio is the subsequent estimate for α or β (or both for a double-constrained model).

4. The routine repeats steps 2 and 3 until the changes from one iteration to the next are very small.

5. The last estimate of α or β (or both for a double-constrained model) is taken as the final values of these parameters.

6. Once the parameters have been estimated, the model can be applied to the calibration data set or to another data set. Note that the parameters are row or column specific (or both). That is, in the 'constrain origins' model, there is a separate coefficient for each row. In the 'constrain destinations' model, there is a separate coefficient for each column. In the 'constrain both origins and destinations', there is a separate coefficient for each cell (row-column combination).

A comparison can be made between the observed distribution and the predicted (modeled) distribution. Because most origin-destination matrices are very large, the vast majority of cells will have zero in them. Thus, a chi-square test would be inappropriate. Instead, a comparison of the *trip length* distribution is made using two different statistics - a coincidence ratio and the Komologorov-Smirnov Two-sample statistic. Details are provided below.

CrimeStat IV Trip Distribution Module

Next, we examine the actual tools that are available in the *CrimeStat* trip distribution module. The tools are illustrated with examples from Baltimore County. The *CrimeStat* trip distribution module includes one setup screen and five routines that implement the model:

1. **Calculate observed origin-destination distribution**. If there is a file available with the coordinates for individual origins and destinations (e.g., an arrest record), this routine will calculate the empirical trip distribution matrix;

2. **Calibrate impedance function**. If there is a file available with the coordinates for individual origins and destinations, this routine will calibrate an empirical impedance function.

3. **Setup origin-destination model**. This screen allows the user to define the parameters of a trip distribution (origin-destination) model with either a mathematical or empirical impedance function.

4. **Calibrate origin-destination model**. This routine calibrates the parameters of the trip distribution model (equation 28.12) using the parameters defined on the setup page.

5. **Apply predicted origin-destination model**. This routine applies the estimated parameters to a data set. The data set can be either the calibration file or another file.

6. **Compare observed and predicted origin-destination trip lengths.** This routine compares the trip lengths from the observed (empirical) trip distribution with that predicted by the model. Comparisons are made graphically by a coincidence ratio, the Komologorov-Smirnov Two-Sample test, and a Chi square test on the most frequent trip links.

Each of these routines is described in detail below. Figure 28.3 shows a screen shot of the trip distribution module.

Describe Origin-Destination Trips

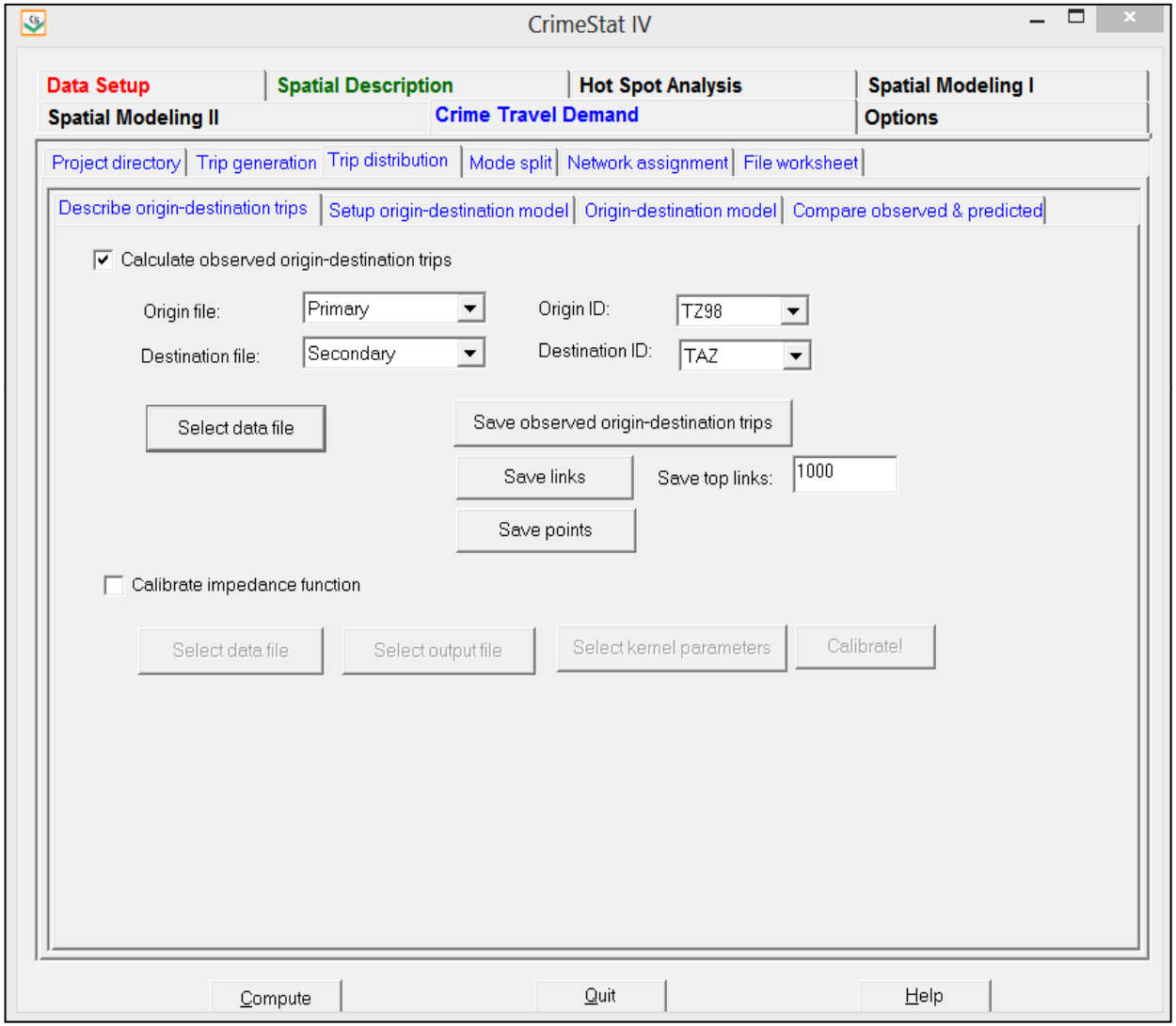
An empirical description of the actual trip distribution matrix can be made if there is a data set that includes individual origin and destination locations. The user defines the origin location and the destination location for each record and a set of zones from which to compare the individual origins and destinations. The routine matches up each origin location with the nearest zone, each destination location with the nearest zone, and calculates the number of trips from each origin zone to each destination zone. This is an *observed* distribution of trips by zone.

The steps in running the model are as follows:

1. **Calculate observed origin-destination trips**. Check if an empirical origin-destination trip distribution is to be calculated.

Figure 28.3:

Trip Distribution Module



2. **Origin file**. The origin file is a list of origin zones with a single point representing the zone (e.g., the centroid). It must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

   A. ***Origin ID***. Specify the origin ID variable in the data file (e.g., CensusTract, Block, TAZ).

3. **Destination file.** The destination file is a list of destination zones with a single point representing the zone (e.g., the centroid). It must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.Specify the destination ID variable in the data file (e.g., CensusTract, Block, TAZ).

   A. Note: all destination ID's should be in the origin zone file and must have the same names and both should be character (string) variables.

4. **Select data file**. The data set must have individual origin and destination locations. Each record must have the X/Y coordinates of an origin location and the X/Y coordinates of a destination location. For example, an arrest file might list individual incidents with each incident having a crime location (the destination) and a residence or arrest location (the origin).

   A. Select the file that has the X and Y coordinates for the origin and destination locations. *CrimeStat* can read ASCII, dbase '.dbf', ArcGIS '.shp' and MapInfo 'dat' files.

   B. Select the tab and specify the type of file to be selected. Use the browse button to search for the file. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns.

   C. ***Variables***. Define the file which contains the X and Y coordinates for both the origin (residence) and destination (crime) locations.

   D. ***Column***. Select the variables for the X and Y coordinates respectively for *both* the origin and destination locations (e.g., Lon, Lat, HomeX, HomeY, IncidentX, IncidentY.) Both locations must be defined for the routine to work.

   E. ***Missing values***. Identify whether there are any missing values for these four fields (X and Y coordinates for both origin and destination locations).

By default, *CrimeStat* will ignore records with blank values in any of the eligible fields or records with non-numeric values (e.g.,alphanumeric characters, #, *). Blanks will always be excluded unless the user selects <none>. There are 8 possible options:

a. <blank> fields are automatically excluded. This is the default
b. <none> indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0
c. 0 is excluded
d. −1 is excluded
e. 0 and −1 indicates that both 0 and -1 will be excluded
f. 0, -1 and 9999 indicates that all three values (0, -1, 9999) will be excluded.
g. Any other numerical value can be treated as a missing value by typing it (e.g., 99)Multiple numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99).

F. ***Type of coordinate system and data units.*** The coordinate system and data units are listed for information. If the coordinates are in longitudes and latitudes, then a spherical system is being used and data units will automatically be decimal degrees. If the coordinate system is projected (e.g., State Plane, Universal Transverse Mercator – UTM), then data units could be either in feet (e.g., State Plane) or meters (e.g., UTM.).

5. **Table output**. The full origin-destination matrix is output as a table to the screen including summary file information and:

a. The origin zone (ORIGIN)
b. The destination zone (DEST)'
c. The number of observed trips (FREQ)

6. **Save observed origin-destination trips.** If specified, the full origin-destination matrix output is saved as a 'dbf' file named by the user. The file output includes:

a. The origin zone (ORIGIN)
b. The destination zone (DEST)
c. The X coordinate for the origin zone (ORIGINX)
d. The Y coordinate for the origin zone (ORIGINY)
e. The X coordinate for the destination zone (DESTX)

f. The Y coordinate for the destination zone (DESTY)
g. The number of trips (FREQ)

Note: each record is a unique origin-destination combination. There are M x N records where *M* is the number of origin zones (including the external zone) and *N* is the number of destination zones.

7. **Save links.** The top observed origin-destination trip links can be saved as separate **line** objects for use in a GIS. Specify the output file format (*ArcGIS* '.shp', *MapInfo* '.mif' or *Atlas*GIS* '.bna') and the file name.

8. **Save top links**. Because the output file is very large (number of origin zones x number of destination zones), the user can select a sub-set of zone combinations with the most observed trips. Indicating the top K links will narrow the number down to the most important ones. The default is the top 100 origin-destination combinations. Each output object is a line from the origin zone to the destination zone with an ODT prefix. The prefix is placed before the output file name. The line graphical output for each object includes:

    a. An ID number from 1 to K, where K is the number of links output (ID)
    b. The feature prefix (ODT)
    c. The origin zone (ORIGIN)
    d. The destination zone (DEST)
    e. The X coordinate for the origin zone (ORIGINX)
    f. The Y coordinate for the origin zone (ORIGINY)
    g. The X coordinate for the destination zone (DESTX)
    h. The Y coordinate for the destination zone (DESTY)
    i. The number of observed trips for that combination (FREQ)
    j. The distance between the origin zone and the destination zone.

9. ***Save points***. Intra-zonal trips (trips in which the origin and destination are the same zone) can be output as separate **point** objects as an *ArcGIS* '.shp', *MapInfo* '.mif' or *Atlas*GIS* '.bna' file. Again, the top K points are output (default=100). Each output object is a point representing an intra-zonal trip with an ODTPOINTS prefix. The prefix is placed before the output file name. The point graphical output for each object includes:

a. An ID number from 1 to K, where K is the number of links output (ID)
b. The feature prefix (POINTSODT)
c. The origin zone (ORIGIN)
d. The destination zone (DEST)
e. The X coordinate for the origin zone (ORIGINX)
f. The Y coordinate for the origin zone (ORIGINY)
g. The X coordinate for the destination zone (DESTX)
h. The Y coordinate for the destination zone (DESTY)
i. The number of observed trips for that combination (FREQ)

Example of Observed Trip Distribution from Baltimore County

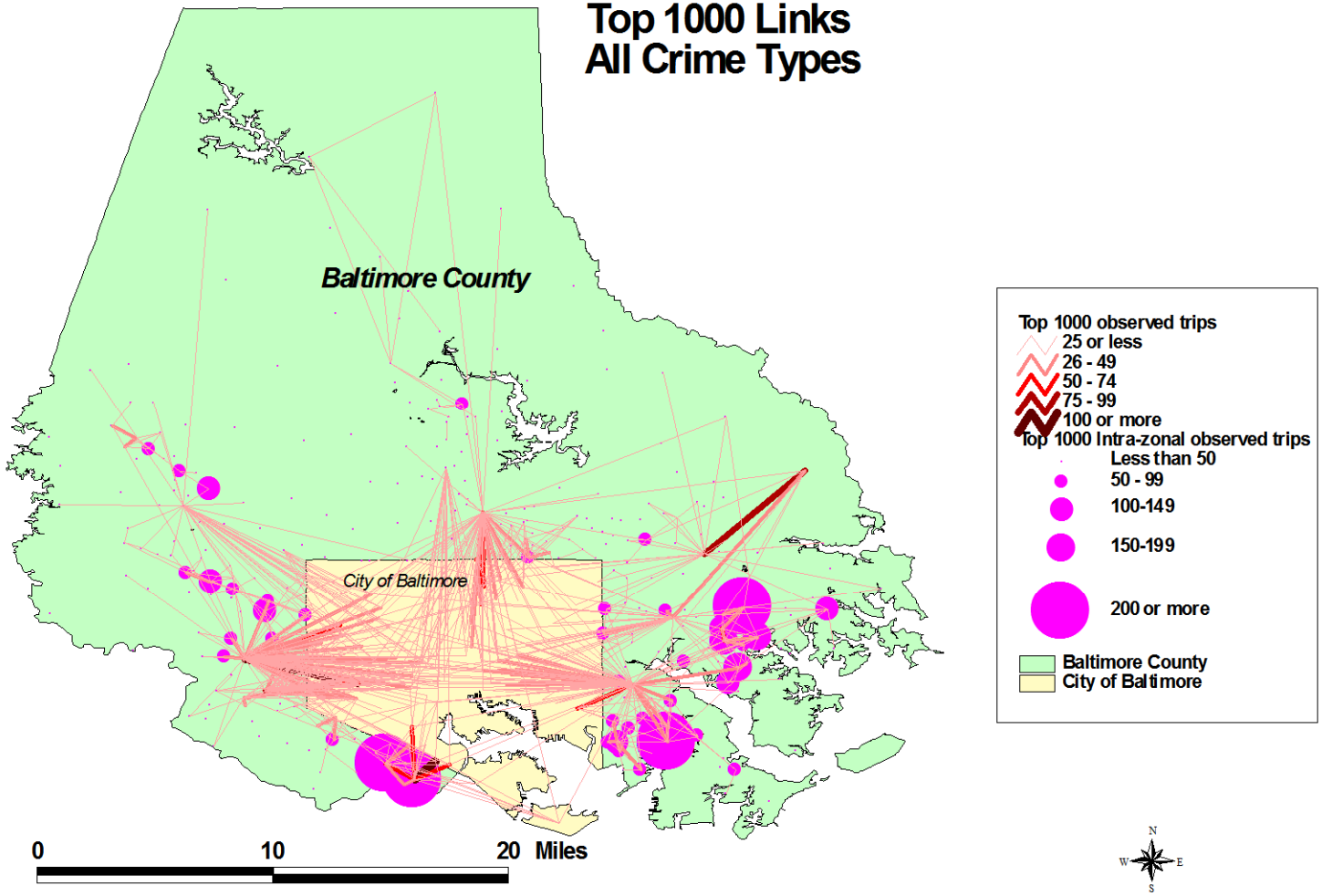
Figure 28.4 shows the output of the top 1000 links for the observed trip distribution from a sample of 41,974 records for incidents committed between 1993 and 1997. The zonal model used was that of traffic analysis zones (TAZ). These were discussed in Chapter 26. Because there are a large number of links (532 origin zones by 325 destination zones), the top 1000 were taken. These accounted for 19,615 crime trips (or 46.7% of all trips). A larger number of links could have been selected, but the map would have become more cluttered. Of the 19,615 trips that are displayed in the map, 7,913 or 40.3% are intra-zonal trips. These were output by the routine as points and have been displayed as circles with the size proportional to the number of trips. The remaining 11,702 trip links were output by the routine as lines and are displayed with the thickness and strength of color of the line being proportional to the number of trips.

There are several characteristics of the trip pattern that should be noted. First, the intra-zonal trips tend to concentrate on the eastern part of Baltimore County. This is an area that is relatively poor with a high number of public housing projects. This suggests that there are a lot of intra-community crimes being committed in these locations. Second, the zone-to-zone pattern, on the other hand, tends to concentrate at five different locations relatively close to border with the City of Baltimore. These five locations are all major shopping malls. Third, the origins for those trips to the shopping mall tend to come from within the City of Baltimore. Fourth, in general, the locations with high intra-zonal trips do not have a large number of zone-to-zone trips. However, there is one exception in the southwest corner of the county.

In other words, the observed distribution of crime trips is complex, but with several patterns being shown. A lot of crime trips occur over very short distances. But there is also a convergence of many crime trips on major shopping malls in the County.

Figure 28.4:

Observed Baltimore County Crime Trips: 1993-1997
Top 1000 Links
All Crime Types



Calibrate Impedance Function

This routine allows the calibration of an approximate travel impedance function based on actual trip distributions. It is used to describe the travel impedance in distance only of an actual sample (the calibration sample). Unlike the remaining routines in this section, the "Calibrate impedance function cannot use travel time, or cost. A file is input which has a set of incidents (records) that include both the X and Y coordinates for the location of the offender's residence (origin) and the X and Y coordinates for the location of the incident that the offender committed (destination.)

The routine estimates a travel distance function using a one-dimensional kernel density method. See the details in Chapter 13. Essentially, for each record, the separation between the origin location and the destination location is calculated and is represented on a distance scale. The maximum impedance is calculated and divided into a number of intervals; the default is 100 equal sized intervals, but the user can modify this. For each impedance point calculated, a one-dimensional kernel is overlaid. For each interval, the values of all kernels are summed to produce a smooth function of travel impedance. The results are saved to a file that can be used for the origin-destination model.

Note, however, that this is an empirical distribution and represents the combination of origins, destinations, and costs. It is not necessarily a good description of the impedance (cost) function by itself. Some of the mathematical functions produce a better fit than the empirical impedance function.

The steps in calculating an empirical impedance function are as follows:

1. **Select data file for calibration**. Select the file that has the X and Y coordinates for the origin and destination locations. *CrimeStat* can read ASCII, dbase '.dbf', ArcGIS '.shp' and MapInfo 'dat' files. Select the tab and select the type of file to be selected. Use the browse button to search for the file. If the file type is ASCII, select the type of data separator (comma, semicolon, space, tab) and the number of columns.

   A. ***Variables***. Define the file which contains the X and Y coordinates for both the origin (residence) and destination (crime) locations

   B. ***Columns***. Select the variables for the X and Y coordinates respectively for *both* the origin and destination locations (e.g., Lon, Lat, HomeX, HomeY, IncidentX, IncidentY.) Both locations must be defined for the routine to work.

C. ***Missing values***. Identify whether there are any missing values for these four fields (X and Y coordinates for both origin and destination locations). By default, *CrimeStat* will ignore records with blank values in any of the eligible fields or records with non-numeric values (e.g.,alphanumeric characters, #, *). Blanks will always be excluded unless the user selects <none>. There are 8 possible options:

a. <blank> fields are automatically excluded. This is the default
b. <none> indicates that no records will be excluded. If there is a blank field, *CrimeStat* will treat it as a 0
c. 0 is excluded
d. −1 is excluded
e. 0 and −1 indicates that both 0 and -1 will be excluded
f. 0, -1 and 9999 indicates that all three values (0, -1, 9999) will be excluded.
g. Any other numerical value can be treated as a missing value by typing it (e.g., 99)Multiple numerical values can be treated as missing values by typing them, separating each by commas (e.g., 0, -1, 99, 9999, -99).

D. ***Type of coordinate system and data units***. Select the type of coordinate system. If the coordinates are in longitudes and latitudes, then a spherical system is being used and data units will automatically be decimal degrees. If the coordinate system is projected (e.g., State Plane, Universal Transverse Mercator − UTM), then data units could be either in feet (e.g., State Plane) or meters (e.g., UTM.) Directional coordinates are not allowed for this routine.

2. **Select Kernel Parameters**. There are five parameters that must be defined.

A. ***Method of interpolation***. There are fives types of kernel distributions that can be used to estimate point density:

a. The **normal** kernel overlays a three-dimensional normal distribution over each point that then extends over the area defined by the reference file. This is the default kernel function.
b. The **uniform** kernel overlays a uniform function (disk) over each point that only extends for a limited distance.
c. The **quartic** kernel overlays a quartic function (inverse sphere) over each point that only extends for a limited distance.

d. The **triangular** kernel overlays a three-dimensional triangle (cone) over each point that only extends for a limited distance.

e. The **negative exponential** kernel overlays a three dimensional negative exponential function ('salt shaker') over each point that only extends for a limited distance

B. The methods produce similar results though the normal is generally smoother for any given bandwidth.

3. ***Choice of* bandwidth**. The kernels are applied to a limited search distance, called 'bandwidth'. For the normal kernel, bandwidth is the standard deviation of the normal distribution. For the uniform, quartic, triangular and negative exponential kernels, bandwidth is the radius of a circle defined by the surface. For all types, larger bandwidth will produce smoother density estimates and both adaptive and fixed bandwidth intervals can be selected.

A. ***Fixed bandwidth***. A fixed bandwidth distance is a fixed interval for each point. The user must define the interval, the interval size, and the distance units by which it is calculated (miles, nautical miles, feet, kilometers, meters.) The default bandwidth setting is fixed with intervals of 0.25 miles each. The interval size can be changed.

B. ***Adaptive bandwidth***. An adaptive bandwidth distance is identified by the minimum number of other points found within a symmetrical band drawn around a single point. A symmetrical band is placed over each distance point, in turn, and the width is increased until the minimum sample size is reached. Thus, each point has a different bandwidth size. The user can modify the minimum sample size. The default for the adaptive bandwidth is 100 points.

4. ***Specify Interpolation Bins***. The interpolation bins are defined in one of two ways:

A. By the number of bins. The maximum distance calculated is divided by the number of specified bins. This is the default with 100 bins. The user can change the number of bins.

B. By the distance between bins. The user can specify a bin width in miles, nautical miles, feet, kilometers, and meters.

5. **Output (Areal) Units**. Specify the density units as points per mile, nautical mile, foot, kilometer, or meter. The default is points per mile.

6. **Calculate Densities or Probabilities**. The density estimate for each cell can be calculated in one of three ways:

   A. **Absolute densities**. This is the number of points per grid cell and is scaled so that the sum of all grid cells equals the sample size.

   B. **Relative densities**. For each grid cell, this is the absolute density divided by the grid cell area and is expressed in the output units (e.g., points per square mile)

   C. **Probabilities**. This is the proportion of all incidents that occur in the grid cell. The sum of all grid cells equals a probability of 1. Unlike the Jtc calibration routine, this is the default. In most cases, a user would want a proportional (probability) distribution as the relative differences in impedance for different costs are what is of interest.

   Select whether absolute densities, relative densities, or probabilities are to be output for each cell. The default is probabilities.

7. **Select Output File**. The output *must* be saved to a file. *CrimeStat* can save the calibration output to either a dbase 'dbf' or ASCII text 'txt' file.

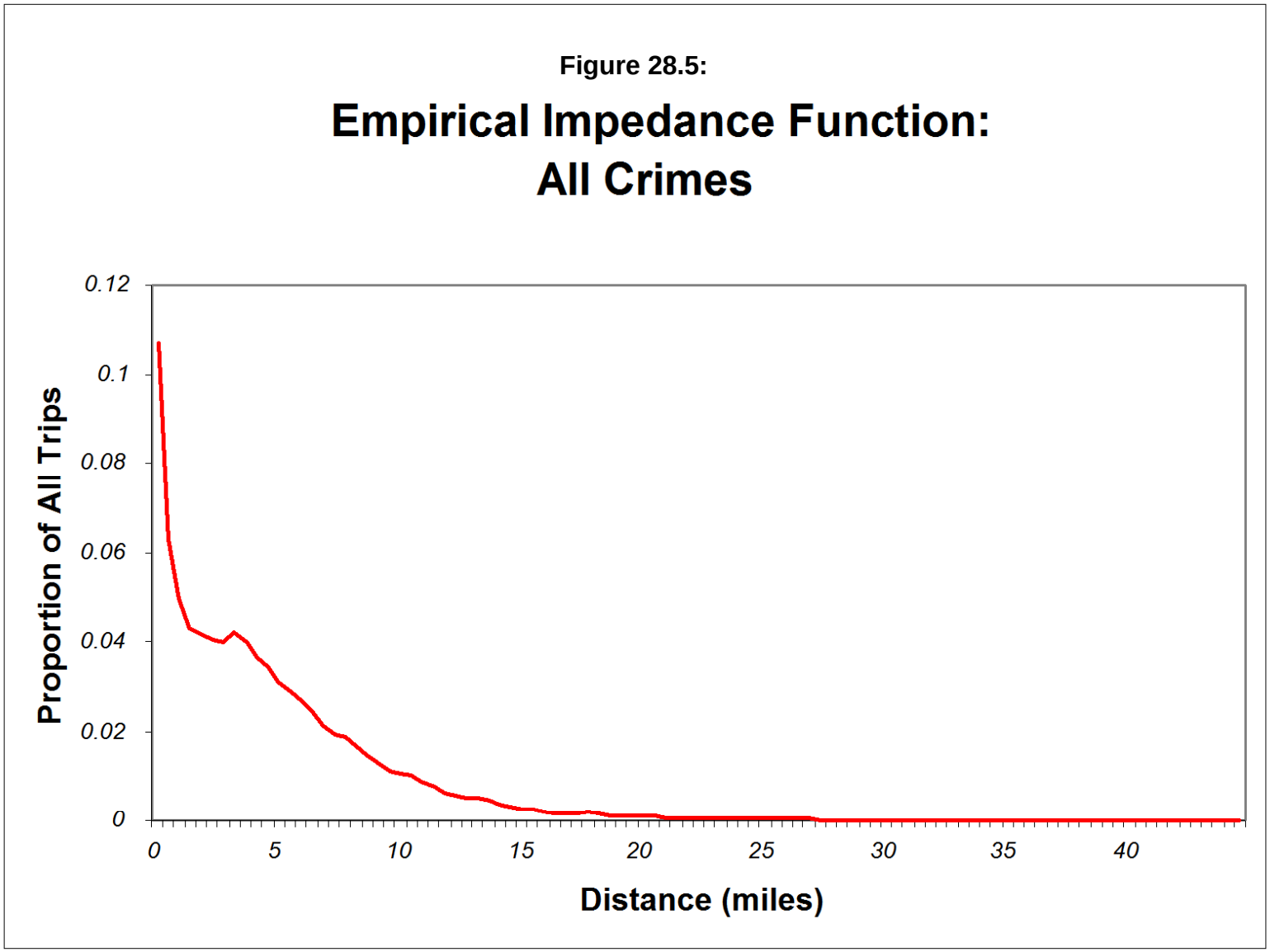
8. **Calibrate!** Click on 'Calibrate!' to run the routine. The output is saved to the specified file upon clicking on 'Close'.

9. **Graphing the travel impedance function**. Click on 'View graph' to see the travel impedance function. The screen view can be printed by clicking on 'Print'. For a better quality graph, however, the output should be imported into a graphics or spreadsheet program.

Example of Empirical Impedance from Baltimore County

An example of an empirical impedance function from Baltimore County is seen in Figure 28.5. This was derived from the 41,974 incidents in which both the crime location and the offender's origin location were known. As seen, the function looks similar to a negative exponential function. But there is a little 'hitch' around 3 miles where the travel likelihood

Figure 28.5:


Empirical Impedance Function:
All Crimes
Proportion of All Trips
0.12
0.1
0.08
0.06
0.04
0.02
0
0
5
10
15
20
25
30
35
40
Distance (miles)

increases, rather than decrease. This could possibly be due to the City of Baltimore border which abuts much of the southern part of the County.

Whatever the reason, the empirical impedance function can be used as a proxy for travel 'cost' by offenders. As we shall see, however, it may not produce as good a fit in the gravity model as some of the mathematical functions, particularly the lognormal. The reason is that it is a behavioral description. Consequently, the pattern reflects both the existence of crime opportunities (attractions) as well as costs. While an empirical description is useful for guessing where a serial offender might live, for a trip distribution model it apparently does not cleanly estimate the real costs to an offender. Nevertheless, it is a tool that can be used.

Setup of Origin-Destination Model

The page is for the setup of the origin-destination model. All the relevant files, models and exponents are input on the page and it allows the trip distribution model to be calibrated and allocated. Figure 28.6 shows the setup screen. There are a number of parameters that have to be defined:

1. **Predicted origin file.** The predicted origin file is a file that lists the origin zones with a single point representing the zone (e.g., the centroid) and also includes the predicted number of crimes by origin zone. The file must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

    A. ***Origin variable.*** Specify the name of the variable for the predicted origins (e.g., PREDICTED, ADJORIGINS).

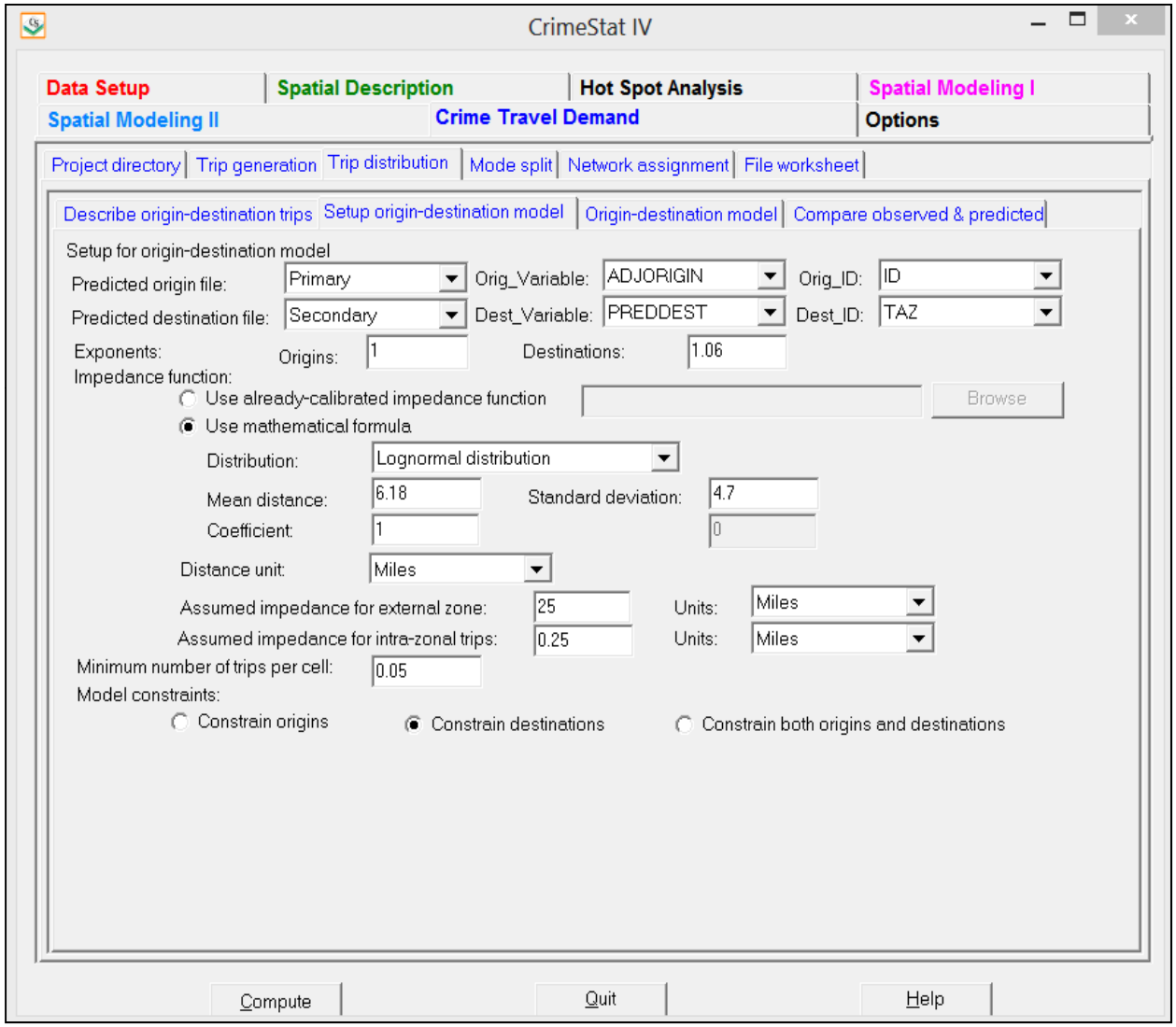
    B. ***Origin ID***. Specify the origin ID variable in the data file (e.g., CensusTract, Block, TAZ).

2. **Predicted destination file**. The predicted destination file is a list of destination zones with a single point representing the zone (e.g., the centroid) and also includes the predicted number of crimes by destination zone. It must be input as either the primary or secondary file. Specify whether the data file is the primary or secondary file.

    A. ***Destination variable***. Specify the name of the variable for the predicted destination (e.g., PREDICTED, ADJDEST).

Figure 28.6:

Trip Distribution Model Setup



B. **Destination ID**. Specify the destination ID variable in the data file (e.g., CensusTract, Block, TAZ).

Note: with a 32 bit operating system (e.g., Windows XP, 32 bit Windows 7), there is maximum allowable of 4 Gb. If M is the number of rows and N is the number of columns, then the total number of grid cells (M x N) cannot be greater than $\sqrt{\frac{(RAM-64)}{56}}$ where RAM is the available RAM. With a 64 bit operating system, on the other hand, 32 Gb are addressable.

3. **Exponents**. The exponents are power terms for the predicted origins and destinations. They indicate the relative strength of those variables. For example, compared to an exponent of 1.0 (the default), an exponent greater than 1.0 will strengthen that variable (origins or destinations) while an exponent less than 1.0 will weaken that variable. They can be considered 'fine tuning' adjustments.

   A. ***Origins.*** Specify the exponent for the predicted origins. The default is 1.0.

   B. ***Destinations.*** Specify the exponent for the predicted origins. The default is 1.0.

4. **Impedance function**. The trip distribution routine can use two different travel distance functions:

   A. ***Use an already-calibrated distance function.*** If a travel distance function has already been calibrated (see 'Calibrate impedance function' above), the file can be directly input into the routine. The user selects the name of the already-calibrated travel distance function. *CrimeStat* reads dbase 'dbf', ASCII text 'txt', and ASCII data 'dat' files.

   B. ***Use a mathematical formula.*** A mathematical formula can be used instead of a calibrated distance function. Similar to the Journey-to-crime module (see chapter 13), there are five mathematical functions. They measure a *separation* between two zones and estimate a likelihood value. 'Separation' can be in terms of distance, travel time, speed (which is converted into travel time), or travel costs.

5. **Mathematical functions.** Briefly, the five functions are:

A. **Linear**. The simplest type of distance model is a linear function. This model postulates that the likelihood of traveling to a zone from another by an offender declines by a constant amount with distance from the offender's home. It is highest near the offender's home but drops off by a constant amount for each unit of distance until it falls to zero. The form of the linear equation is;

$$f(d_{ij}) = \alpha + \beta S_{ij} \quad (28.17)$$

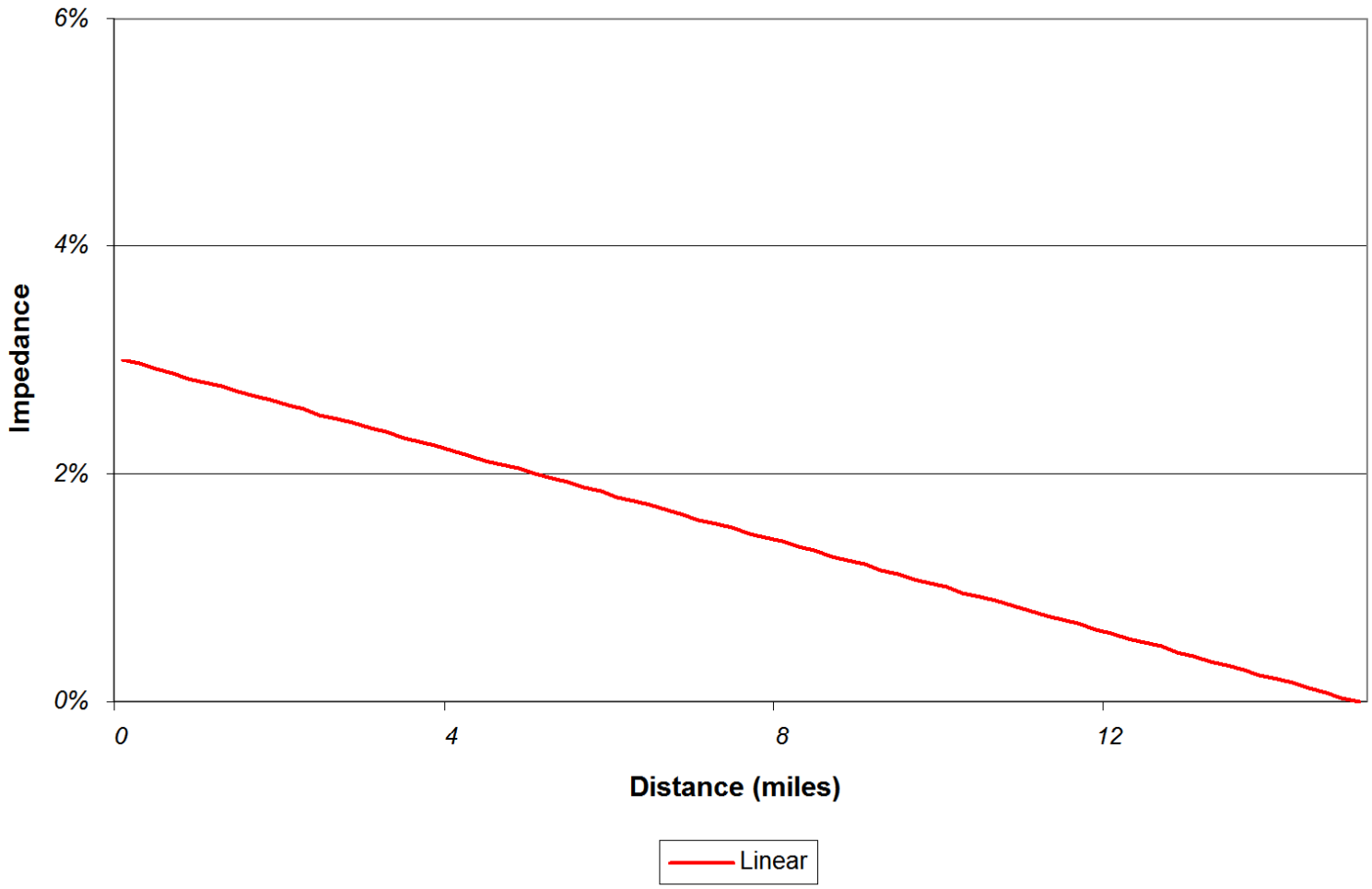
where $f(d_{ij})$ is the likelihood that the offender will travel from an origin zone, i, to a destination zone, j, S_{ij} is the *separation* in distance, time or cost between the offender's residence, i, and location j, α is a slope coefficient which defines the fall off in distance, and β is a constant. It would be expected that the coefficient β would have a negative sign since the likelihood should decline with separation. The user must provide values for *A* and β. The default for A is 10 and for β is -1. When the function reaches 0 (the X axis), the routine automatically substitutes a 0 for the function. Figure 28.7 illustrates this function.

B. ***Negative Exponential***. A slightly more complex function is the negative exponential. In this type of model, the likelihood of travel also drops off with distance. However, the decline is at a constant *rate* of decline, thus dropping quickly near the offender's home until is approaches zero likelihood. The mathematical form of the negative exponential is:

$$f(d_{ij}) = \alpha e^{-\beta S_{ij}} \quad (28.18)$$

where $f(d_{ij})$ is the likelihood that the offender will travel from an origin zone, i, to a destination zone, j, S_{ij} is the *separation* in distance, time or cost between the offender's residence, i, and location j, e is the base of the natural logarithm, α is the coefficient and β is an exponent of e. The user inputs values for α - the coefficient, and β - the exponent. The default for α is 10 and for β is 1.

Figure 28.7:
Linear Impedance Function



a. This function is the one most used by travel demand modelers. It has been recommended for use by the Federal Highway Administration (NCHRP, 1995). Figure 28.8 illustrates a typical negative exponential impedance function.

C. ***Normal***. A normal distribution assumes the peak likelihood is at some optimal distance from the offender's home base. Thus, the function rises to that distance and then declines. The rate of increase prior to the optimal distance and the rate of decrease from that distance is symmetrical in both directions. The mathematical form is:

$$Z_{ij} = \frac{(S_{ij} - \bar{d})}{\sigma_d} \tag{28.19}$$

$$f(d_{ij}) = \alpha \frac{1}{\sigma_d \sqrt{2\pi}} e^{-0.5 Z_{ij}^2} \tag{28.20}$$

where $f(d_{ij})$ is the likelihood that the offender will travel from an origin zone, i, to a destination zone, j, S_{ij} is the *separation* in distance, time or cost between the offender's residence, i, and location j, $\bar{d}$ is the mean distance input by the user, σ_d is the standard deviation of distances, e is the base of the natural logarithm, and α is a coefficient. The user inputs values for $\bar{d}$, σ_d, and α. The default values are 1 for each of these parameters.

a. By carefully scaling the parameters of the model, the normal distribution can be adapted to a distance decay function with an increasing likelihood for near distances and a decreasing likelihood for far distances. For example, by choosing a standard deviation greater than the mean (e.g., $\bar{d} = 1, S_d = 2$), the distribution will be skewed to the left because the left tail of the normal distribution is not evaluated. Figure 28.9 illustrates a possible normal impedance function.

Figure 28.8:
Negative Exponential Impedance Function

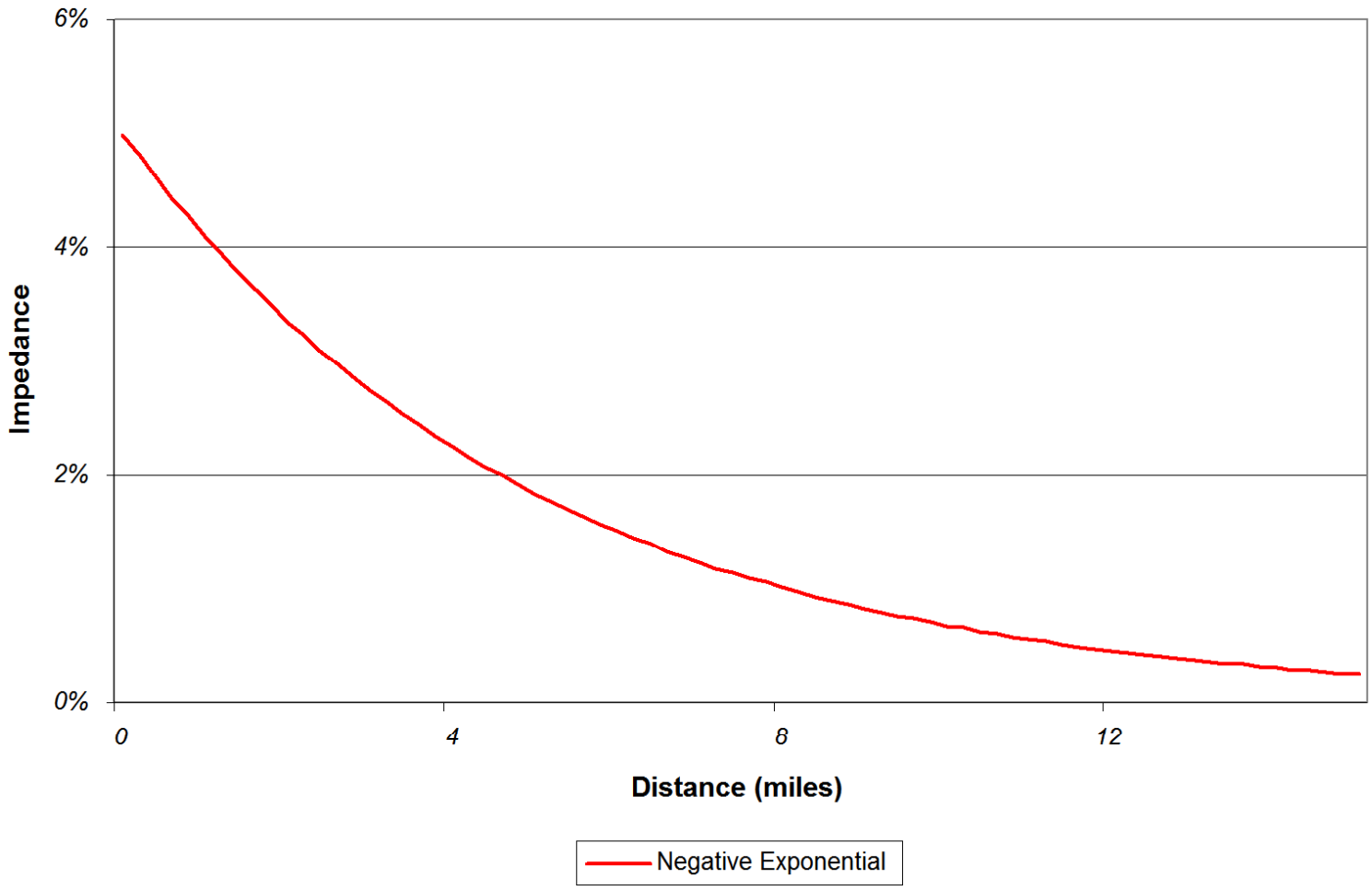
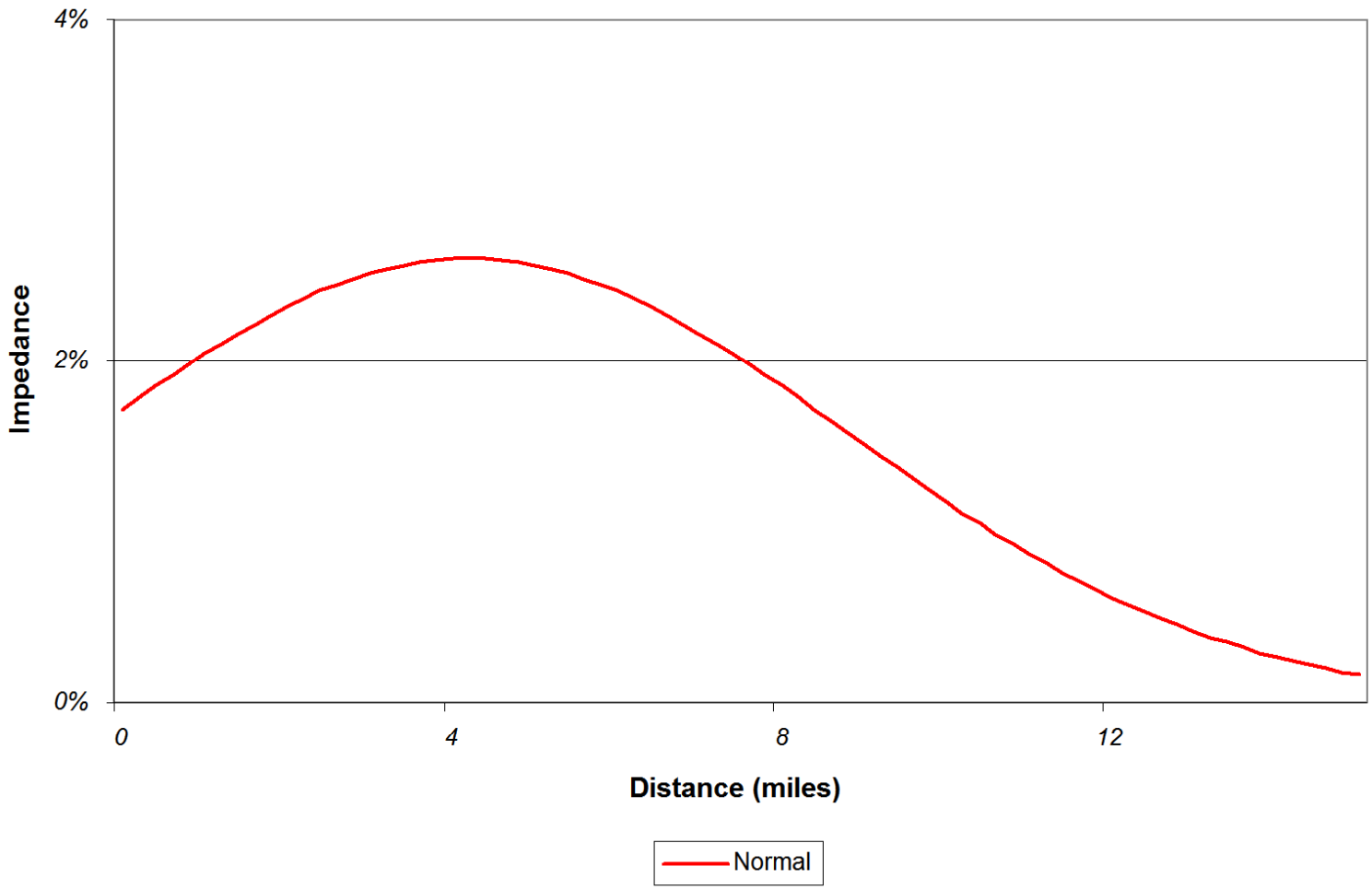


Figure 28.9:
Normal Impedance Function



D. ***Lognormal***. The lognormal function is similar to the normal except it is more skewed, either to the left or to the right. It has the potential of showing a very rapid increase near the origin with a more gradual decline from a location of peak likelihood. The mathematical form of the function is:

$$f(d_{ij}) = \alpha \frac{1}{S_{ij}^2 \sigma_d \sqrt{2\pi}} e^{-\frac{-(\ln(S_{ij}^2)-\bar{d})^2}{2\sigma_d^2}} \quad (28.21)$$

where $f(d_{ij})$ is the likelihood that the offender will travel from an origin zone, i, to a destination zone, j, S_{ij} is the *separation* in distance, time or cost between the offender's residence, i, and location j, $\bar{d}$ is the mean distance input by the user, σ_d is the standard deviation of distances, e is the base of the natural logarithm, and α is a coefficient. The user inputs values for $\bar{d}$, σ_d, and α. The default values are 1 for each of these parameters. Figure 28.10 illustrates a log-normal impedance function that had wide utility in several studies that are discussed below.

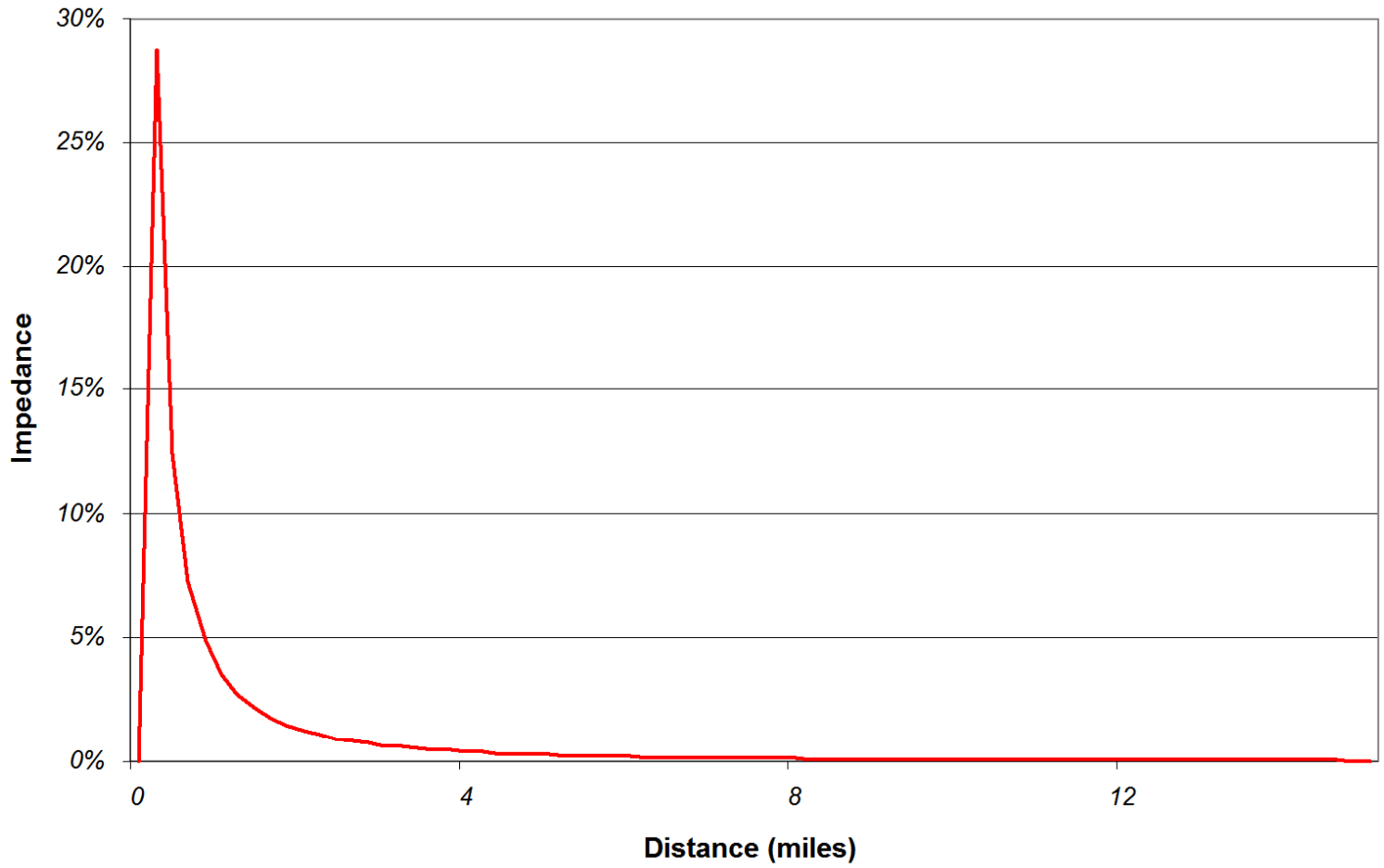
E. ***Truncated Negative Exponential***. Finally, the truncated negative exponential is a joined function made up of two distinct mathematical functions - the linear and the negative exponential. For the near distance, a positive linear function is defined, starting at zero likelihood for distance 0 and increasing to d_p, a location of peak likelihood. Thereupon, the function follows a negative exponential, declining quickly with distance. The two mathematical functions making up this spline function are:

Linear: $f(d_{ij}) = 0 + \beta S_{ij} = \beta d_{ij}$ for $S_{ij} \geq 0$, $S_{ij} \leq S_p$ (28.22)

Negative Exponential: $f(d_{ij}) = \alpha e^{-\xi S_{ij}}$ for $S_{ij} \geq S_p$ (28.23)

where $f(d_{ij})$ is the likelihood that the offender will travel from an origin zone, i, to a destination zone, j, S_{ij} is the *separation* in distance, time or cost between the offender's residence, i, and location j, β is the slope of the linear function (default=+1) and α is a coefficient and ξ is an exponent

Figure 28.10:
Lognormal Impedance Function



for the negative exponential function. Since the negative exponential only starts at a particular distance, S_p, α, is assumed to be the intercept if the Y-axis were transposed to that distance. Figure 28.11 illustrates a truncated negative exponential impedance function.

F. ***Model parameters***. For each mathematical model, two or three different parameters must be defined:

1. For the negative exponential, the coefficient and exponent
2. For the normal distribution, the mean distance, standard deviation and coefficient
3. For lognormal distribution, the mean distance, standard deviation and coefficient
4. For the linear distribution, an intercept and slope
5. For the truncated negative exponential, a peak distance, peak likelihood, intercept, and exponent.

The parameters will be obtained either from a previous analysis or from an iterative process of experimentation. See the example below under "Compare observed and predicted trips".

G. **'Fine Tuning' Exponents**. In addition, for each function, exponents for the attraction and production terms can adjusted. This allows a 'fine tuning' of the impedance function to better fit the empirical distribution.

5. **Distance Units.** The routine can calculate impedance in four ways, by:

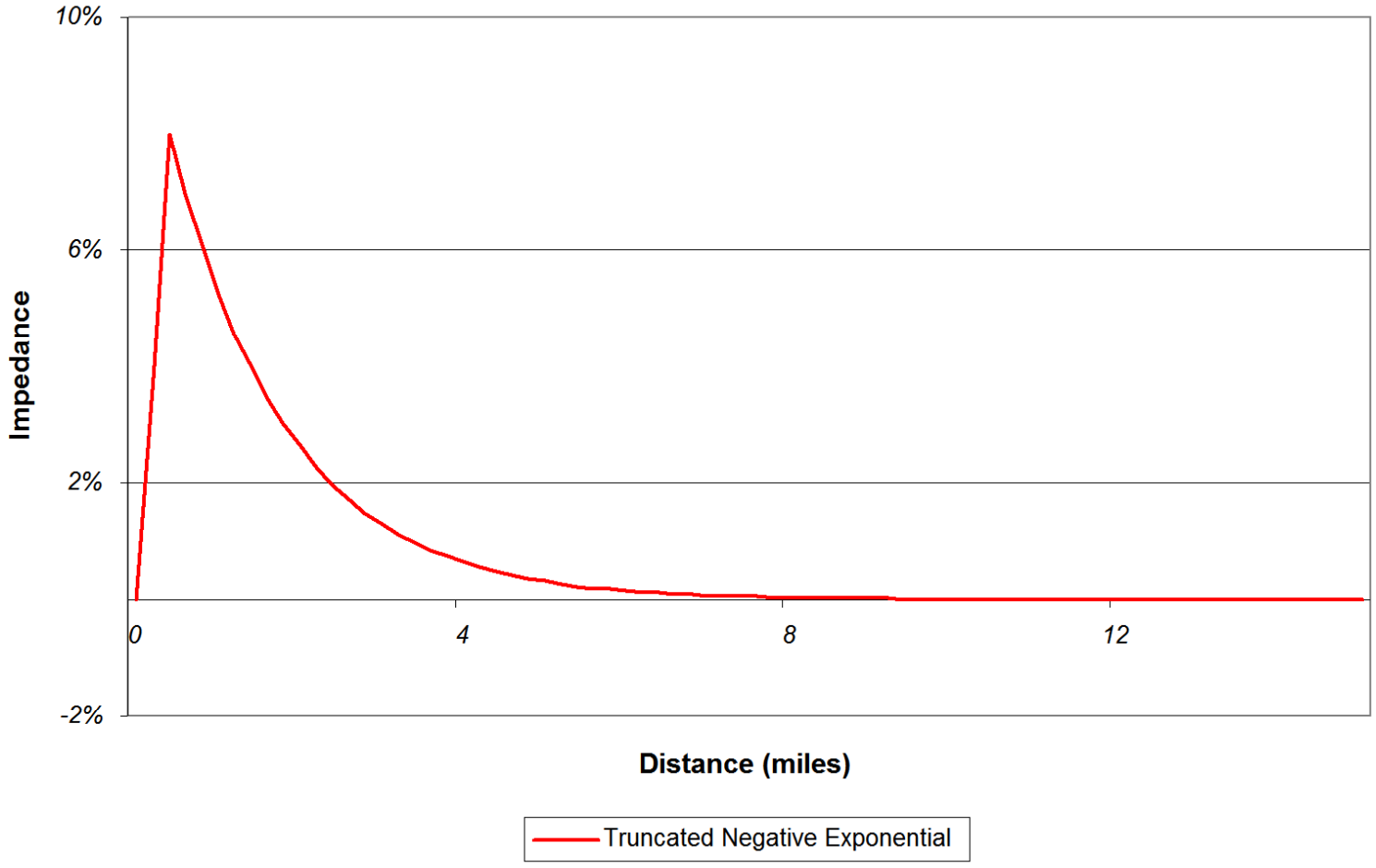
A. Distance (miles, nautical miles, feet, kilometers, and meters)
B. Travel time (minutes, hours)
C. Speed (miles per hour, kilometers per hour). Speed is then converted into travel time, in minutes.
D. General travel costs (unspecified units).

These must be setup under 'Network Distance' on the Measurement Parameters page. In the Network Parameters dialogue, specify the measurement units. The default is distance in miles.

6. **Assumed Impedance for External Zones.** For trips originating outside the study area (external trips), specify the amount and the units that will be assumed for these trips. The default is 25 miles.

Figure 28.11:

Truncated Negative Exponential Impedance Function



7. **Assumed Impedance for Intra-zonal Trips**. For trips originating and ending in the same zone (intra-zonal trips), specify the amount and the units that will be assumed for these trips. The default is 0.25 miles.

8. **Model Constraints.** In calibrating a model, the routine must constrain either the origins or the destinations (single constraint) or constrain both the origins and the destinations (double constraint). In the latter case, it is an iterative solution. The default is to constrain destinations as it is assumed that the destination totals (the number of crimes occurring in each zone) are probably more accurate than the number of crimes originating in each zone. Specify the type of constraint for the model.

   A. ***Constrain origins.*** If 'constrain origins' is selected, the total number of trips from each origin zone will be held constant.

   B. ***Constrain destinations.*** If 'constrain destinations' is selected, the total number of trips from each destination zone will be held constant.

   C. ***Constrain both origins and destinations.*** If 'constrain both origins and destinations' is selected, the routine works out a balance between the number of origins and the number of destinations.

Fitting the Impedance Function

The impedance function is fit in an iterative manner. First, either an empirical impedance or a mathematical impedance is chosen. Second, the particular mathematical function is selected. For example, with the lognormal function, which has been found to produce the best fit for three different data sets, there are three parameters: 1) the mean distance; 2) the standard deviation of distance; and 3) the coefficient.

Third, initial values of the parameters are chosen; one suggestion is to use the defaults available in the *CrimeStat* routines. The "Compare observed and predicted trips" routine can be used to evaluate the fit of the model. Fourth, the parameters are adjusted in small increments, one at a time, on both side of the initial guess in order to improve the fit. For example, with the lognormal function, the mean distance is fit first because it has the greatest impact on the overall fit. Then, after a "best" mean distance has been found, the standard deviation of distance is adjusted until it produces a "best" fit. Then, the coefficient is adjusted until it produces a "best" fit. Fifth, and finally, the 'fine tuning' exponents of the production and attraction functions are adjusted. Typically, these change the final fit only slightly. Hence, they represent a final adjustment.

This process is illustrated below in the discussion on the comparison of the observed and predicted trips. Essentially, the empirical (observed) distribution is being used as a calibration sample in order to find that impedance model and parameters that best approximate it.

Running the Origin-Destination Model

The trip distribution (origin-destination) model is implemented in two steps. First, the coefficients are calculated according to the exponents and impedance functions specified on the setup page. Second, the coefficients and exponents are applied to the predicted origins and destinations resulting in a predicted trip distribution. Because these two steps are sequential, they cannot be run simultaneously.

Calibrate Origin-Destination Model.

In this routine, the row or column parameters (or both if double constraint is used) are estimated using a calibration file. The steps are as follows:

1. **Check** the 'Calibrate origin-destination model' box to run the calibration model.

2. **Save Modeled Coefficients (parameters).** The modeled coefficients are saved as a 'dbf' file. Specify a file name.

Apply Predicted Origin-Destination Model

In this routine, the coefficients that were calibrated in the above routine can be applied to a data set. The data set can be the same as the calibration file or a different one. The reason for separating the calibration from application steps is that the coefficients can be used for many different data sets. The steps are as follows:

1. **Check** the 'Apply predicted origin-destination model' box to run the trip distribution prediction.

2. **Modeled Coefficients File**. Load the modeled coefficients file saved in the 'Calibrate origin-destination model' stage.

3. **Assumed Coordinates for External Zone**. In order to model trips from the 'external zone' (trips from outside the study area), specify coordinates for this zone. These coordinates will be used in drawing lines from the predicted origins to the predicted destinations. There are four choices:

A. Mean center (the mean X and mean Y of all origin file points are taken). This is the default.
B. Lower-left corner (the minimum X and minimum Y values of all origin file points are taken).
C. Upper-right corner (the maximum X and maximum Y values of all origin file points are taken).
D. User coordinates (user-defined coordinates). Indicate the X and Y coordinates that are to be used.

Because an arbitrary location is taken to represent the 'external zone', any lines that are shown from that zone will not necessarily represent any real travel behavior. However, if a very high proportion of all crime trips fall within the modeled origin zones (i.e., 95% or more), then it is very unlikely that any of the top trip links will come from the 'external zone'.

4. **Table Output**. The table output includes summary file information and:

   A. The origin zone (ORIGIN)
   B. The destination zone (DEST)
   C. The number of predicted trips (PREDTRIPS)

5. **Save Predicted Origin-destination Trips**. Define the output file. The output is saved as a 'dbf' file specified by the user.

6. **File Output**. The file output includes:

   A. The origin zone (ORIGIN)
   B. The destination zone (DEST)
   C. The X coordinate for the origin zone (ORIGINX)
   D. The Y coordinate for the origin zone (ORIGINY)
   E. The X coordinate for the destination zone (DESTX)
   F. The Y coordinate for the destination zone (DESTY)
   G. The number of predicted trips (PREDTRIPS)

   Note: each record is a unique origin-destination combination and there are M x N records where M is the number of origin zones (including the external zone) and N is the number of destination zones.

7. **Save Links.** The top predicted origin-destination trip links can be saved as separate **line** objects for use in a GIS. Specify the output file format (*ArcGIS* '.shp', *MapInfo* '.mif' or *Atlas*GIS* '.bna') and the file name.

Save Top Links

Because the output file is very large (number of origin zones x number of destination zones), the user can select a sub-set of zone combinations with the most predicted trips. Indicating the top K links will narrow the number down to the most important ones. The default is the top 100 origin-destination combinations. Each output object is a line from the origin zone to the destination zone with an ODT prefix. The prefix is placed before the output file name.

The graphical output includes:

A. An ID number from 1 to K, where K is the number of links output (ID)
B. The feature prefix (ODT)
C. The origin zone (ORIGIN)
D. The destination zone (DEST)
E. The X coordinate for the origin zone (ORIGINX)
F. The Y coordinate for the origin zone (ORIGINY)
G. The X coordinate for the destination zone (DESTX)
H. The Y coordinate for the destination zone (DESTY)
I. The number of predicted trips for that combination (PREDTRIPS)
J. The distance between the origin zone and the destination zone.

8. **Save Points**

Intra-zonal trips (trips in which the origin and destination are the same zone) can be output as separate **point** objects as an *ArcGIS* '.shp', *MapInfo* '.mif' or *Atlas***GIS* '.bna' file. Again, the top K points are output (default=100). Each output object is a point representing an intra-zonal trip with an ODTPOINTS prefix. The prefix is placed before the output file name.

The graphical output for each includes:

A. An ID number from 1 to K, where K is the number of links output (ID)
B. The feature prefix (POINTSODT)
C. The origin zone (ORIGIN)
D. The destination zone (DEST)
E. The X coordinate for the origin zone (ORIGINX)
F. The Y coordinate for the origin zone (ORIGINY)
G. The X coordinate for the destination zone (DESTX)
H. The Y coordinate for the destination zone (DESTY)
I. The number of predicted trips for that combination (PREDTRIPS)

Example of the Predicted Trip Distribution from Baltimore County

The predicted origins and predicted destinations from Baltimore County were input into a trip distribution model and a predicted trip distribution was output. The impedance function was a lognormal distribution, which produced a good fit to the observed (empirical) distribution (see discussion below).

Figure 28.12 outputs the top 1000 links from the model. The top 1000 links account for 14,271.9 trips, or 34.0% of the total number of trips. Compared to the observed distribution, the top 1000 links account for a smaller proportion of the total trips (14,272 v. 19,615). This suggests that the actual distribution is slightly more concentrated than the model suggests. Like the observed distribution, however, a sizeable number of the top links are intra-zonal trips (5,428 or 12.9%). The intra-zonal trips have been displayed as circles in the figure.

Comparing the predicted trip distribution to the observed trip distribution, some similarities and differences are seen. Figure 28.13 compares the top 1000 zone-to-zone links for the predicted and observed distributions. The model has captured many of the major links. For the five shopping malls that received a sizeable number of actual crime trips, the model has captured the majority of trips for three of them and some trips for a fourth. For the mall in the southeast corner of the county, on the other hand, the model has not allocated a large number of trips. Similarly, for a zone near the western edge of the county, the model has allocated more trips than actually occurred.

There are, of course, only 325 intra-zonal trip links (one for each destination zone). Looking at a comparison of the intra-zonal trips (Figure 28.14), some similarities and differences are seen. Generally, the model captured the location of many intra-zonal trips, but it did not capture the quantity very accurately. Zones that had many intra-zonal trips are shown as having only some by the model and, conversely, the model predicts many intra-zonal trips for two zones which had only some.

In other words, the fit between the actual distribution and the model is not perfect. Considering that only 1000 of the 172,900 trip links (532 origin zones x 325 destination zones) are shown, the model has still done a reasonable job of capturing the major links.

It is not surprising that the model is not perfect. The model is a simple analogue using only three variables (productions, attractions, impedance) whereas the actual distribution represents a very complex set of individual decisions made by offenders. What is perhaps remarkable is that the model has done a decent job of capturing some of these relationships at all.

Figure 28.12:

Predicted Baltimore County Crime Trips: 1993-1997
Top 1000 Links
All Crime Types

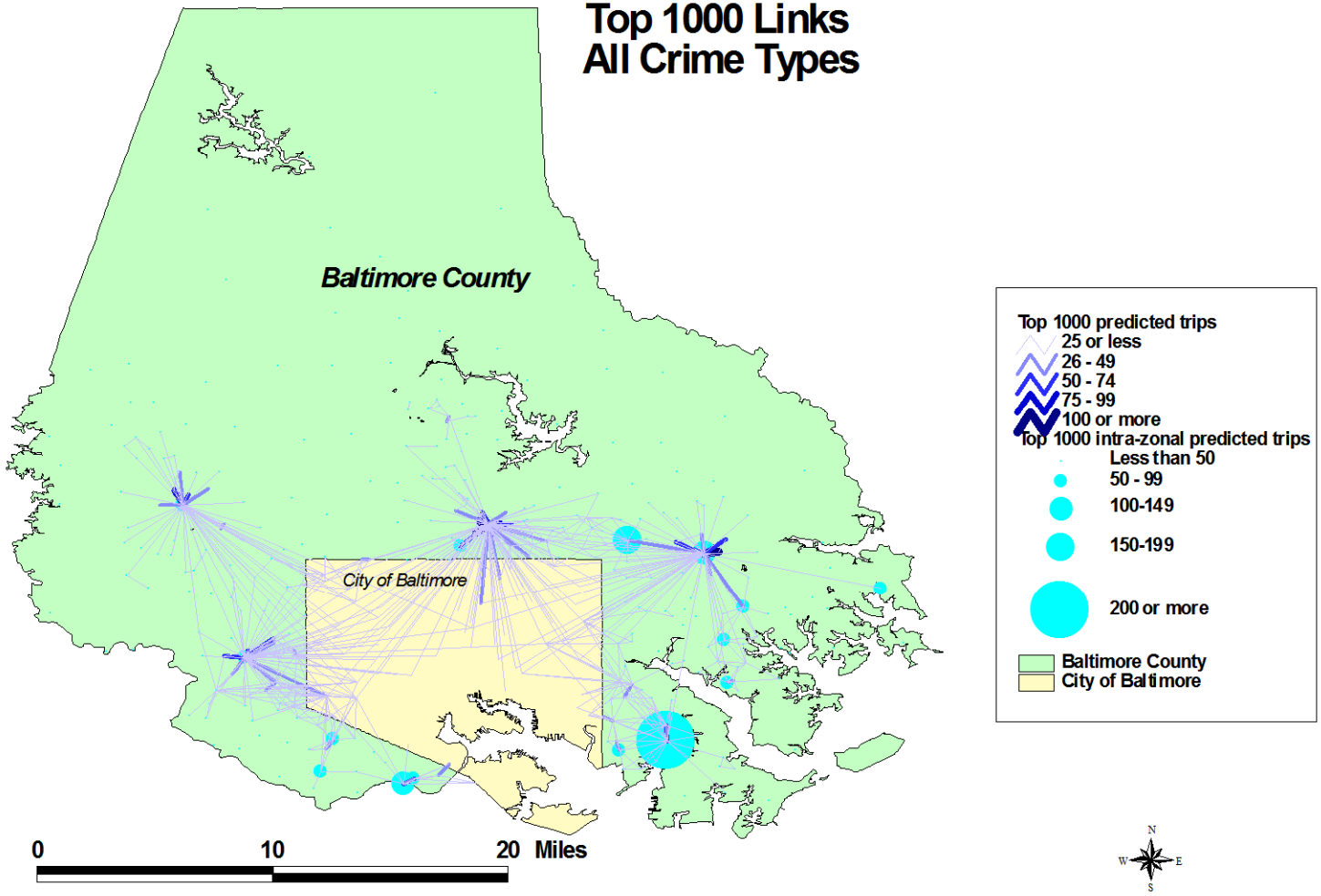


Figure 28.13:

Comparison of Predicted and Observed Crime Trips
1000 Top Zone-to-Zone Trips
All Crime Types

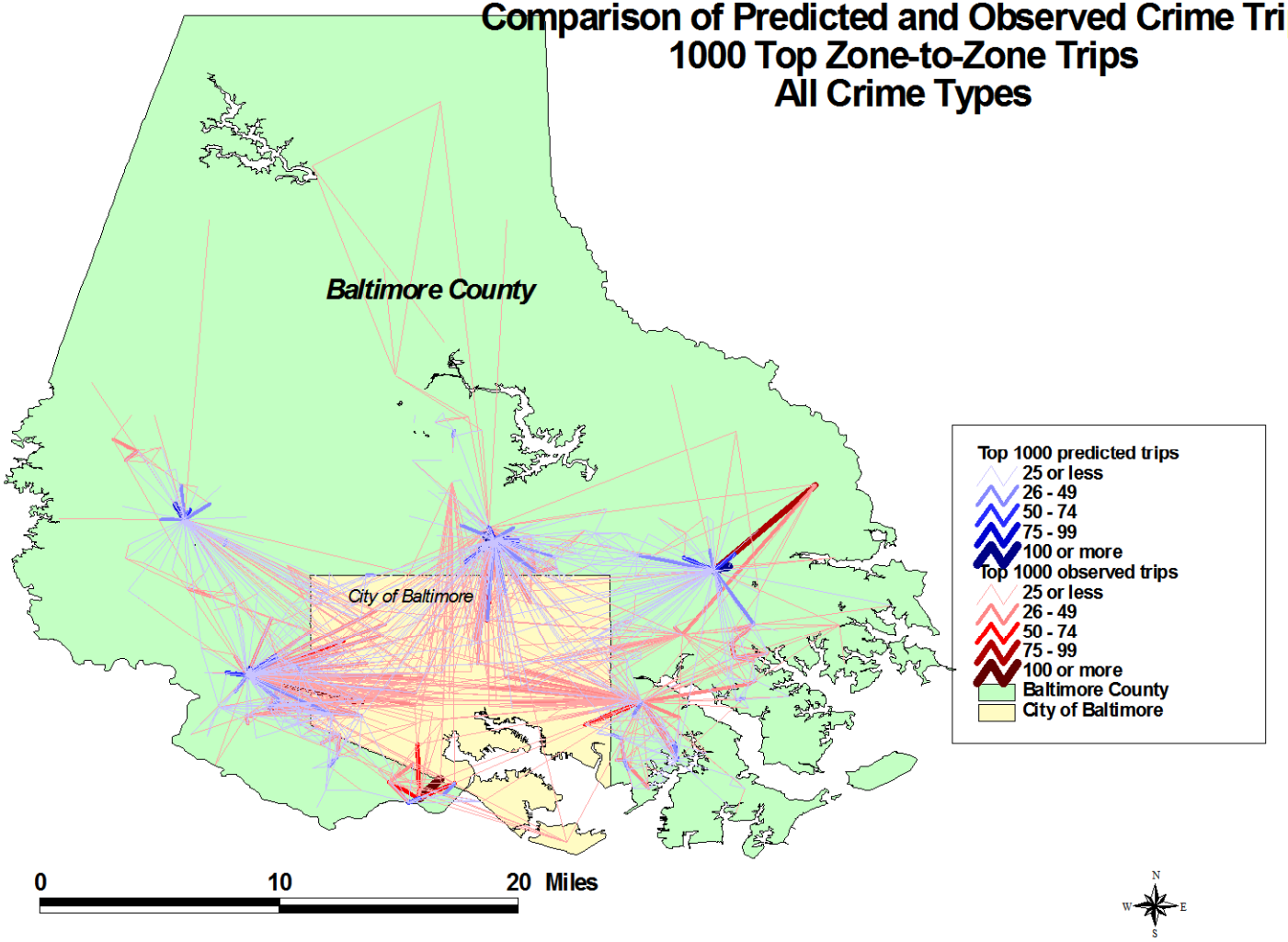
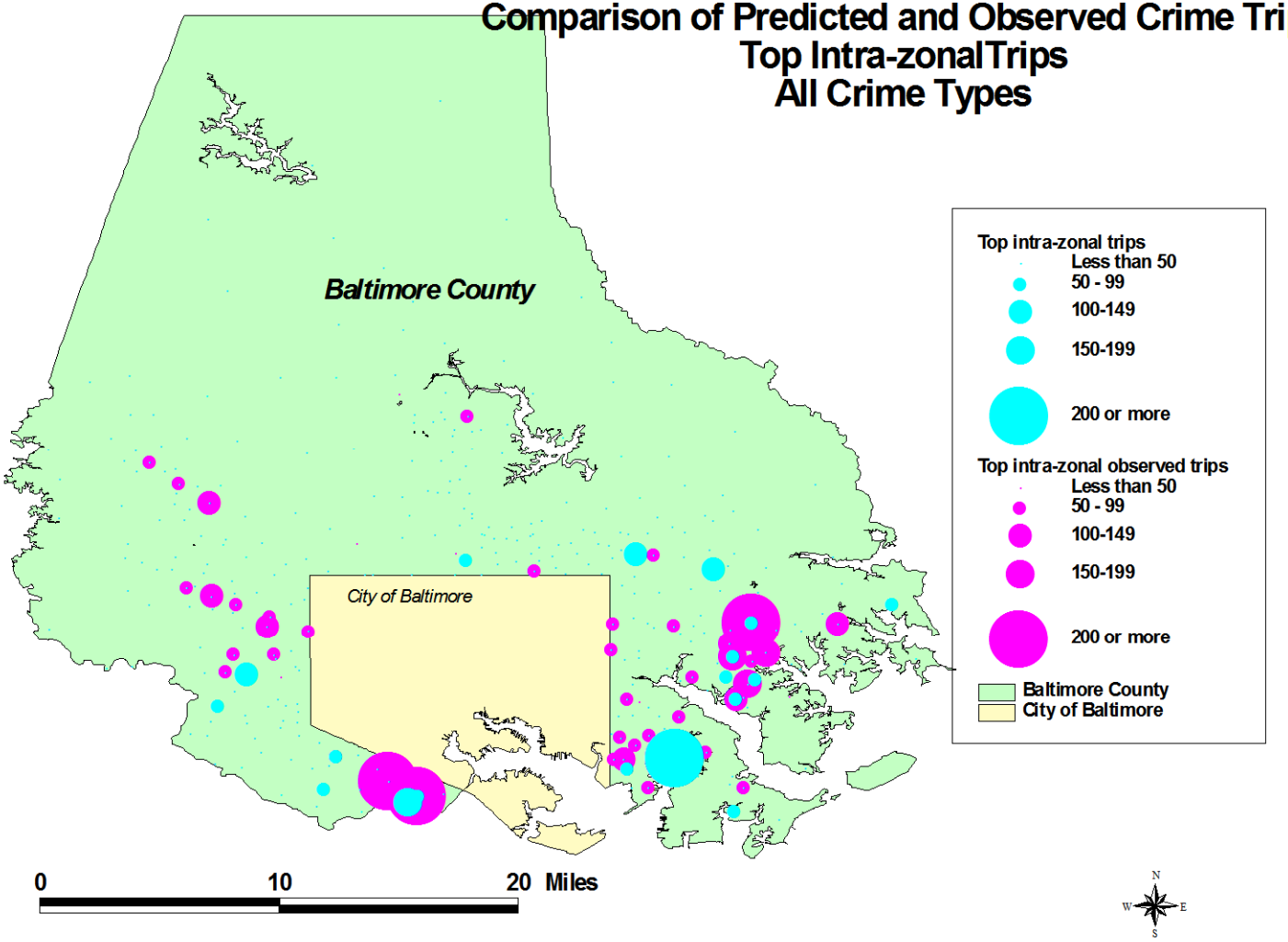


Figure 28.14:
Comparison of Predicted and Observed Crime Trips
Top Intra-zonalTrips
All Crime Types



This brings up an important point, namely that a model is not reality. It is only a simplified set of relationships that approximates reality (in this case, the observed distribution). It is important in developing any model to evaluate it relative to an observed set of facts, and this applies no less to the trip distribution model. One has to understand, however, that a good model will not capture all the relationships. Hopefully, it captures enough of them to make the model useful for prediction and evaluating policy options.

Comparing Observed & Predicted Trips

It is important to conduct a number of tests on the predicted model to ensure that it is capturing the most important elements of the observed distribution. These are conducted by comparing the predicted distribution with the observed (empirical) distribution. Figure 28.15 shows the setup page for comparing the observed with the prediction distribution

There are a number of tests that can be used to evaluate a model by comparing the predicted distribution with the observed one. *CrimeStat* includes three of these and the steps are as follows:

1. Estimate the parameters of the model and apply them to the calibration data set

2. Examine the intra-zonal trips to be sure that the predicted number corresponds to the observed number

3. Compare the trip lengths of the observed and predicted distributions using two tests:

   A. The Coincidence Ratio

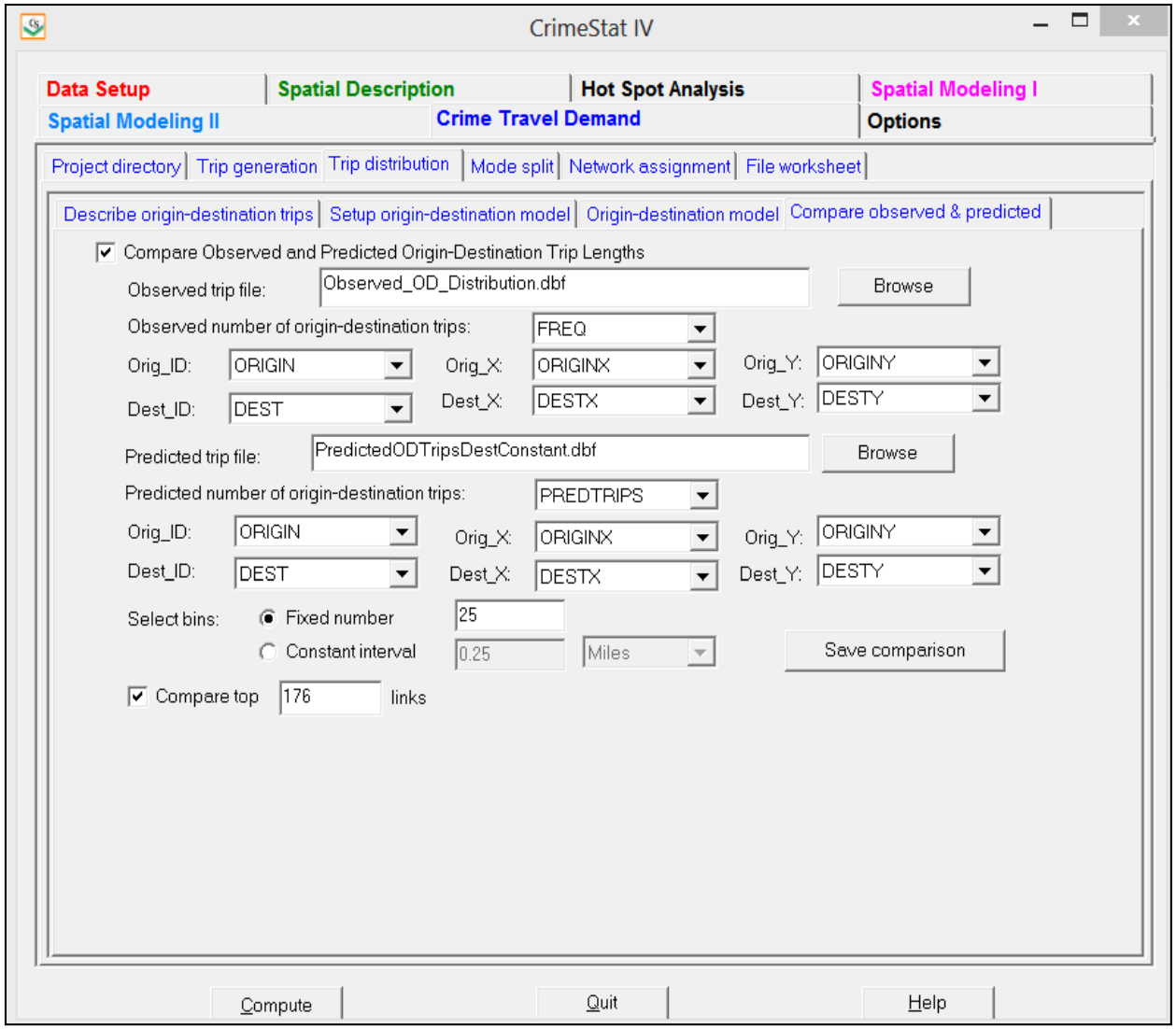
   B. The Komolgorov-Smirnov Two-sample Test

4. Compare the number of trips for the top links using a pseudo-Chi square test. That is, the number of trips for the most frequent links in the observed distribution is compared to the number predicted by the model for the same links.

Unfortunately, not one of these tests is sufficient to validate a model. Further, minimizing the discrepancy for only one of them may distort the others. It is very unlikely that there will be a model that minimizes the errors for all three tests. Consequently, the user will have to choose a model that balances these factors in a desirable way (an *optimum* model).

Figure 28.15:

Comparing Observed and Predicted Trip Lengths



Estimating Impedance Parameters and Exponents of the Gravity Model

While this is not strictly an evaluation test, this step is essential in estimating the particular impedance parameters that are used in the first place. Typically, an analyst will approximate an impedance function. Using a comparison between the observed and predicted models, the parameters can be adjusted to produce a better fit. The steps are as follows:

1. The model is estimated with a calibration data set. There is a file of predicted origins and another file of predicted destinations; typically, these are defined as the primary and secondary files respectively, though the order could be reversed or the same file used for both origins and destinations (if the number of origins zones was identical to the number of destination zones).

2. On the trip distribution setup page, select the type of impedance function that is to be used, already-calibrated (empirical) or mathematical. For the journey-to-crime routine, generally the empirical function led to better results than the mathematical. However, with a trip distribution function, a mathematical function may be as good, if not better. This was tested with three data sets for Baltimore County, Las Vegas, and Chicago and, in all cases, a mathematical function (the lognormal) gave a much better fit than an empirically-derived function (see Chapters 31 and 32).

3. *If* a mathematical function is to be used, select the type of distribution. The default value is a lognormal, but the user can choose a negative exponential, a normal, a linear, or a truncated negative exponential function.

4. For the particular mathematical function, select initial guesses for the parameters. For each mathematical model, two or three different parameters must be defined:

    A. For the negative exponential, the coefficient and exponent
    B. For the normal distribution, the mean distance, standard deviation and coefficient
    C. For lognormal distribution, the mean distance, standard deviation and coefficient
    D. For the linear distribution, an intercept and slope
    E. For the truncated negative exponential, a peak distance, peak likelihood, intercept, and exponent.

5. In addition, there are exponents of the production and attraction side that can be made to 'fine tune' the model. In general, these exponents will only affect the

results slightly, compared to the basic choices of the type of model and the selection of values for the main parameters.

6. Calibrate and apply the model to the calibration data set. Examine the three criteria discussed below to minimize the error between the actual distribution and that predicted by the model.

7. Modify the parameter values slightly.

8. Repeat steps 4 through 7 until a good fit is found between the actual and predicted distribution and in which the errors are minimized and optimized. The process by which this is done is discussed below.

Comparing Intra-zonal Trips

The first evaluation test is to compare the percentage of trips that occur within the same zone - intra-zonal trips. The Travel Model Improvement Program manual indicates that intra-zonal trips should represent typically no more than 5% of all trips for home-to-work trips; that is, commuting trips (FHWA, 1997, chapter 4). However, given that most crime trips are quite short, the proportion of trips that are intra-zonal is liable to be much higher. In Baltimore County, for example, 19.7% of all crime trips were intra-zonal. Ideally, the predicted model should also have 19.7% of all crime trips being intra-zonal.

The "Compare observed and predicted trip lengths" routine is discussed below. The routine outputs the number of trips that are intra-zonal in both the observed and predicted distributions. A good model should produce approximately the same number of intra-zonal trips in the predicted distribution as what actually occurred.

Illustration

For example, in the Baltimore County model displayed in Figure 28.12 above, there were 8,272 intra-zonal trips in the actual distribution (out of 41,979). On the other hand, there were only 5,428 predicted intra-zonal trips in the model. In other words, the predicted model assigned fewer intra-zonal trips than actually occurred.

It may be necessary to modify the model to produce a closer fit for the intra-zonal trips. A simple way to do this to increase or decrease the relative impedance parameter in the model. So, to use the example, if the predicted model is assigning too few intra-zonal trips, then the cost function can be strengthened (i.e., making travel more expensive). In this case, in the original model the lognormal function was used with a mean distance of 6.18 miles. If the mean

distance of the impedance function is reduced to 3.5, then the number of predicted intra-zonal trips increases to 8,275, almost the same number as occurred in the observed distribution.

In other words, by decreasing the mean distance for the lognormal function, the impedance function was strengthened (i.e., made more expensive) and a better fit was created between the observed and predicted distributions.

In and of itself, a mismatch for intra-zonal trips between the predicted model and what actually occurred does not necessarily require a modification of the gravity function. Other criteria must be considered, namely how well the predicted model fits the trip length distribution and how well the predicted models captures the most frequent inter-zonal (zone-to-zone) trip links. Later in the discussion, the issue of optimizing a model by balancing these different criteria will be described.

Comparing Trip Length Distributions

The second evaluation test in comparing the observed with the predicted distribution is a calculation of the trip length distribution (see steps below). Because the trip distribution matrix will typically be very large, most cell values will be zero. Rarely will there be enough data to cover all the cells and, even if there was, the skewness in a crime distribution will leave most cells with no data. For example, for the Baltimore County model, with 532 origin zones and 325 destination zones, there will be 172,900 cells (325 x 532). The calibration data set had only 41,974 cases. Thus, the number of cells is more than four times the sample size and it is not possible to fill all cells with a number.

Consequently, because of the large number of cells with zero counts, one cannot use the Chi square test to compare the observed and predicted distributions. The Chi square test assumes that, first, the distribution is relatively normal (which it is not since the data are highly skewed) and, second, that there are at least 5 cases per cell. The latter condition is impossible given the large number of cells.

Therefore, what is usually done is to compare the *trip length* distribution of the observed and predicted models. 'Trip length' is the length in distance, travel time, or cost of each trip. It is measured by the actual length (or separation) between two zones times the number of cases for that zone pair. For example, in Figure 28.1, there were 15 trips from zone 1 to zone 2 and 7 trips in the opposite direction (from zone 2 to zone 1). Let's assume that the distance between zone 1 and zone 2 is 1.5 miles. Thus, there are 22 trips that fall into a trip length of 1.5 miles (15 in the direction of zone 1 to zone 2 and 7 in the direction of zone 2 to zone 1).

If travel time is used, the calculation uses time rather than distance. For example, if a vehicle was traveling 30 miles per hour, then it would take 3 minutes to cover 1.5 miles (1.5 miles ÷ 30 miles per hour = 0.05 hours x 60 minutes per hour = 3 minutes). Thus, there are 22 trips that fall into a trip 'length' of 3 minutes. A similar logic would apply to travel cost categories.

This process is repeated for all cells and the distribution of trips is allocated to the distribution of trip lengths (in distance, travel time, or travel cost). In general, one uses many intervals (or bins) for trip length (25 or more). In *CrimeStat*, the default number of trip lengths is 25, but it is not unknown to use up to 100. The problem in using too many is that the distributions become unreliable and differences that appear may not be real.

Graphical fit

Once the trip length distribution is calculated for both the observed and predicted distributions, it is possible to compare them. *CrimeStat* outputs a graph showing the fit of the two distributions. In general, they should be very close. An examination of differences between the distributions can indicate at what trip lengths the model is failing. This might allow the parameters to be adjusted in order to improve the fit on the next iteration. Examples will be given below of the graphing of the two distributions. But, it is important to come up with a model in which the two distributions 'look' similar.

Coincidence ratio

The *coincidence ratio* compares the two trip length distributions by examining the ratio of the total area of those distributions that coincide, that are in common (FHWA, 1997, chapter 4). It is defined as:

$$Coincidence = \sum_{k=1}^{K} \min \left(\frac{f^O}{F^O}, \frac{f^P}{F^P}\right) \quad (28.24)$$

$$Total = \sum_{k=1}^{K} \max \left(\frac{f^O}{F^O}, \frac{f^P}{F^P}\right) \quad (28.25)$$

$$Coincidence\ ratio = \frac{Coincidence}{Total} \quad (28.26)$$

The steps are as follows:

1. Essentially, the two distributions are broken into K bins (or intervals). That is, the number of trips in each bin is enumerated (see example above).

2. Each of the two distributions is converted into a proportional distribution by dividing the bin count by the total number of trips in the distribution. This step is not absolutely essential as the test can be conducted of the raw counts. However, by converting into proportions, the two distributions are standardized.

3. A cumulative count is conducted of the *minimum* proportion in each interval. That is, starting at the lowest interval, the smaller of the two proportions is taken. At the next interval, the smaller of the two proportions is added to the count. This is repeated for all K bins. This is called the *coincidence* and measure the overlapping proportions over all intervals.

4. A similar cumulative count is conducted of the *maximum* proportion in each interval. That is, starting at the lowest interval, the larger of the two proportions is taken. At the next interval, the larger of the two proportions is added to the count. This is repeated for all K bins. This is called the *total* and measures the unique proportion over all intervals.

5. Finally, the coincidence ratio is defined as the ratio of the minimum count to the total count.

The coincidence ratio is a proportion from 0 to 1. It is analogous to the R^2 statistic in regression analysis in that it measures the 'explained' (or overlapping) variance. According to the Travel Model Improvement Program manual (FHWA, 1997, chapter 4), the higher the coincidence ratio, the better. A value of 0.9 would generally be considered good.

Komolgorov-Smirnov two-sample test

The Komolgorov-Smirnov Two-Sample Test is similar to the coincidence ratio, but it examines the maximum difference across all bins (Kanji, 1993). For each distribution, a cumulative sum is created. At each interval, the difference between the two cumulative sums is calculated. The ***maximum*** difference between the two distributions is taken as the test statistic:

$$D = |O_i - P_i| \tag{28.27}$$

where D is the maximum difference found, O_i is the cumulative sum of the actual (observed) trip lengths, and P_i is the cumulative sum of the predicted trip lengths. There are tables of critical values for the Komolgorov-Smirnov Two-Sample Test which are a function of the number of intervals, K (Smirnov, 1948; Massey, 1951; Siegel, 1956; Kanji, 1993).

Illustration

To illustrate the trip length comparison, figures 28.16 through 28.19 show the results for four different impedance models - an empirical impedance function, a negative exponential impedance function, a truncated negative exponential impedance function, and a lognormal impedance function. As seen, the fit of the empirical impedance function is not particularly good, but gets progressively better with the three different mathematical functions.

The best fit is clearly was with the lognormal function. With these parameters (mean center = 6.0 miles, standard deviation = 4.7 miles, coefficient = 1, origin exponent = 1, and destination exponent = 1.06), the Coincidence Ratio was 0.93.

But, again, this is just one criterion, though it fits most of the distribution matrix. As with the number of intra-zonal trips, minimizing the error for a trip length distribution will not necessarily minimize the error for the other two criteria (intra-zonal trips and the top links). But, it is important that the trip length comparison be reasonably close.

Comparing the Trips of the Top Links

The third evaluation test focuses on the top links. That is, it evaluates how well the predicted model captures the major trip links, both intra-zonal and inter-zonal. Since crime trip distributions are skewed (i.e., a handful of zones contribute to most crime origins and a handful of zones attract many crimes), capturing the most important links is essential for a good crime distribution model. This is particularly true since a model that produces the best fit for the overall trip length distribution may not capture the top links very well.

Therefore, simply comparing the trip length distribution may not adequately capture the top links. That is, on average a particular model may produce a good fit between the predicted and observed distributions, but may do this by minimizing error across the entire matrix of trip pairs without necessarily minimizing the error for the top links.

Consequently, it is important to also compare the fit of the model for the top links. One of the lines in the dialogue for the “Compare observed and predicted trip lengths” is “Compare top links”. The user should specify the number of top links to be compared; the default is 100. The top links are the trip pairs that have the most number of actual trips, starting from the pair with the most trips and sorting in descending order. The routine calculates a pseudo-Chi square test on just those links. Since the top links will all have a sufficient number of trips, it is possible to calculate a Chi square statistic. However, since not all links are being considered in this test, a significance test of this statistic cannot be calculated since the sampling error is not known.

Figure 28.16:

Comparing Observed and Predicted Crime Trip Lengths

Empirical Impedance Function

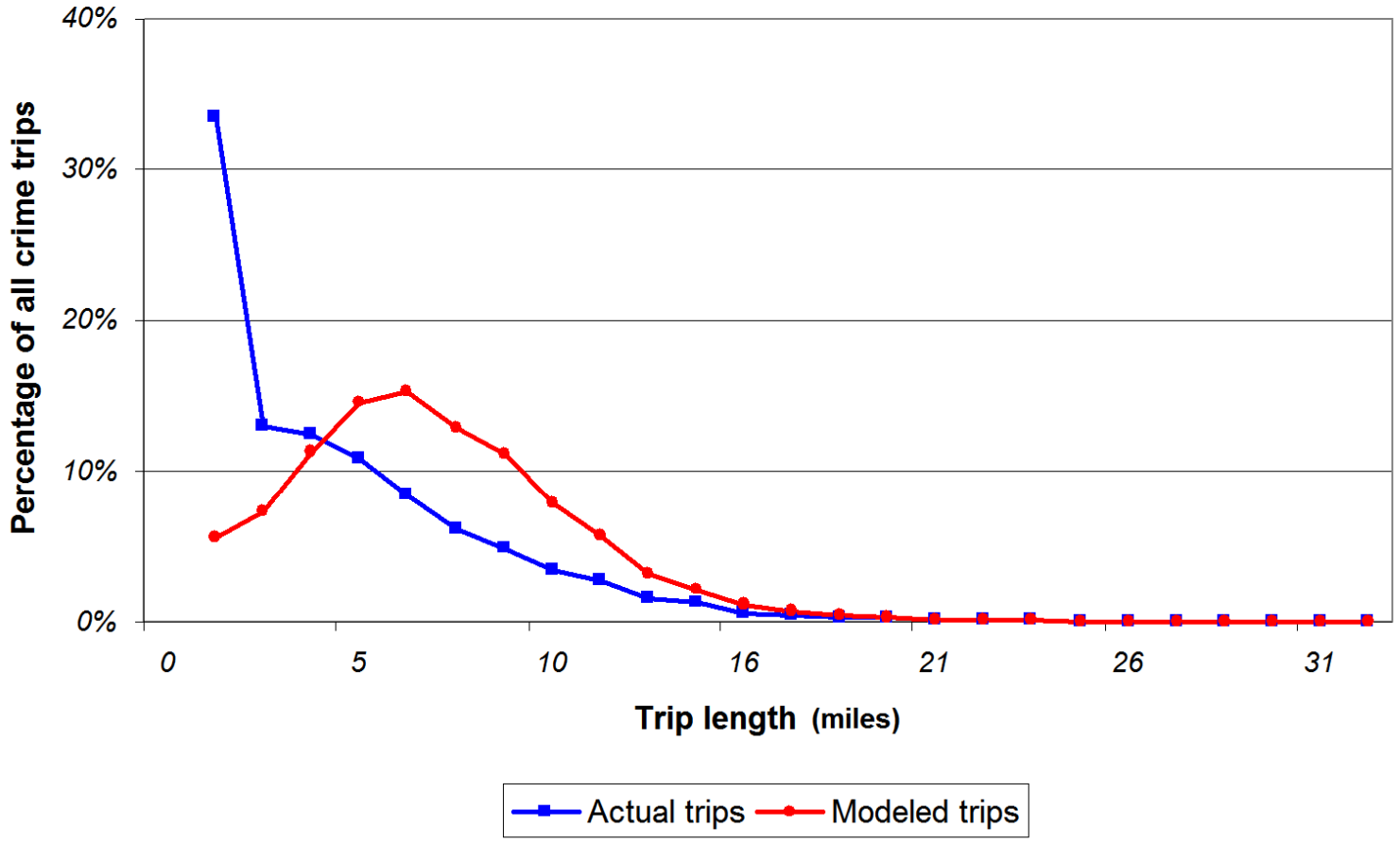


Figure 28.17:

Comparing Observed and Predicted Crime Trip Lengths

Negative Exponential Impedance Function

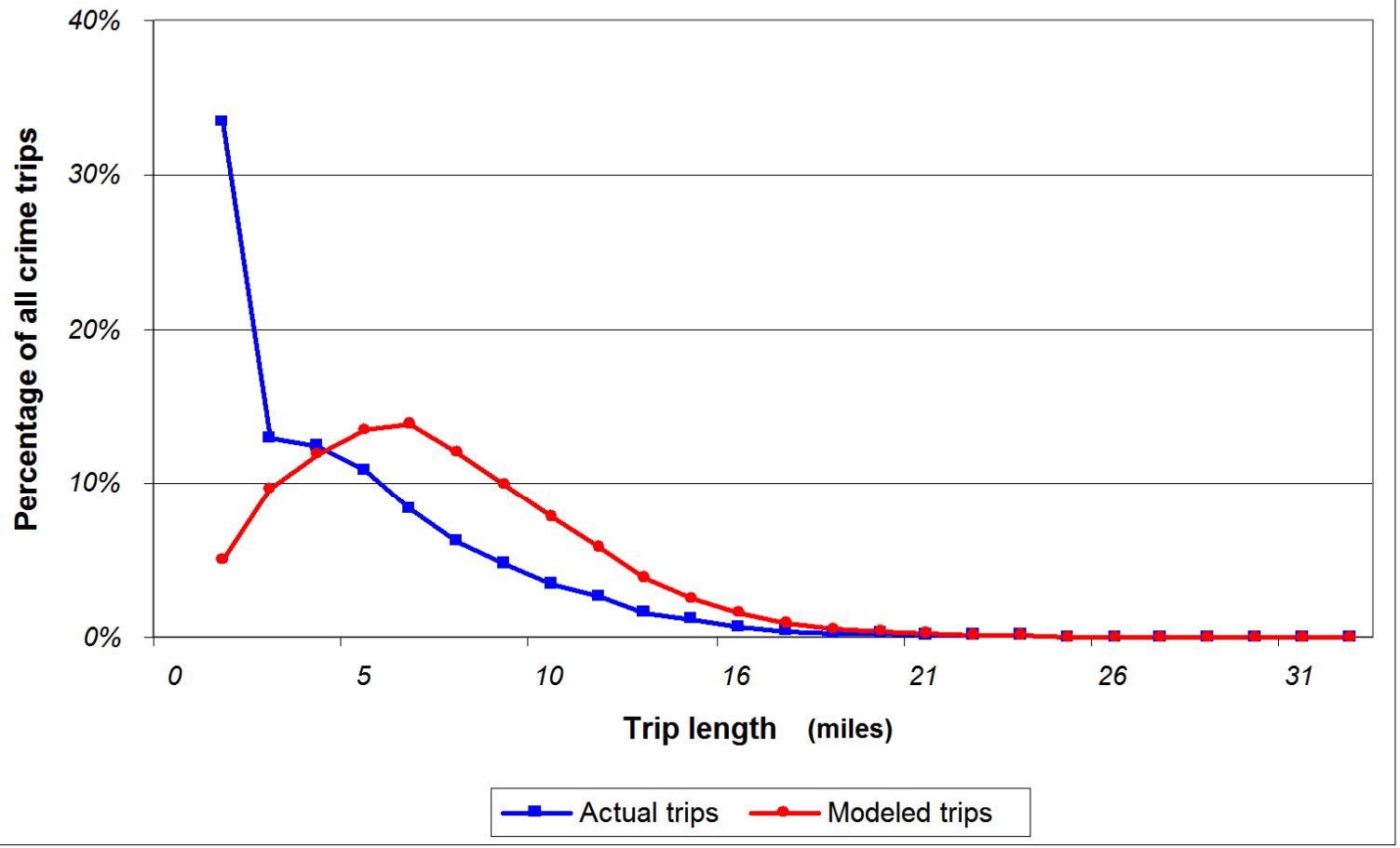


Figure 28.18:

Comparing Observed and Predicted Crime Trip Lengths

Truncated Negative Exponential Impedance Function

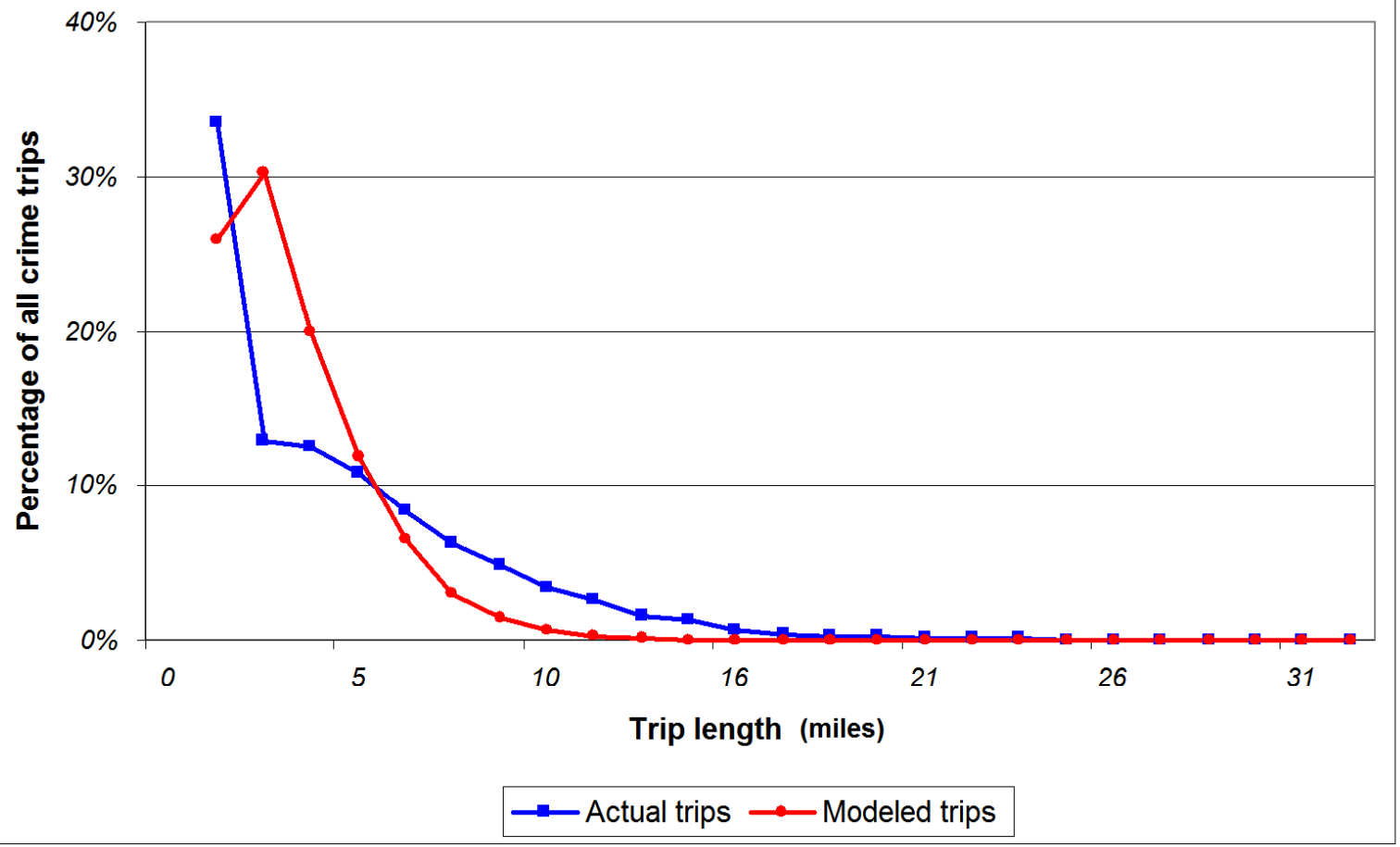
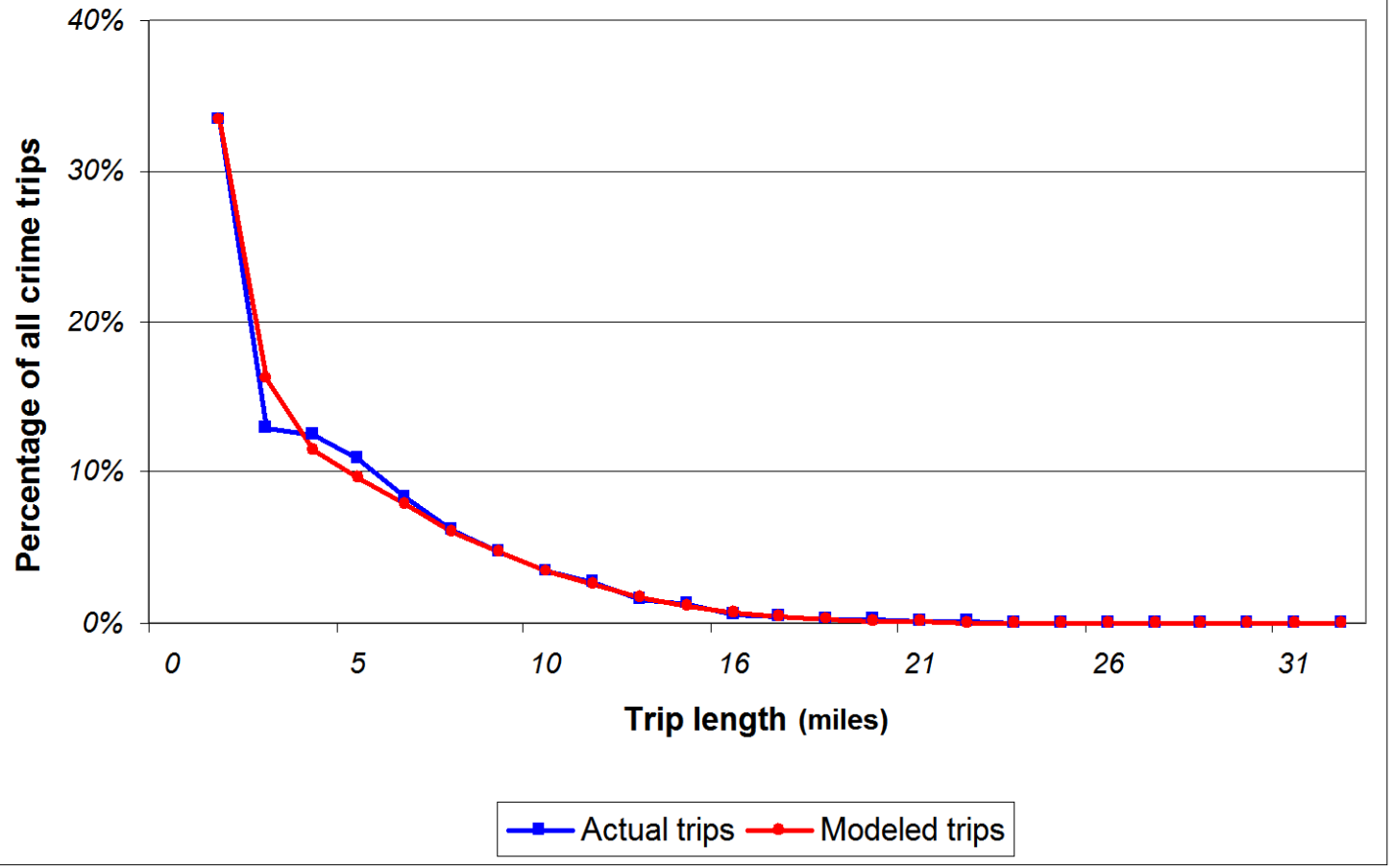


Figure 28.19:

Comparing Observed and Predicted Crime Trip Lengths

Lognormal Impedance Function



Using the observed (actual) links as the reference, the test calculates:

$$Pseudo - \chi^2 = \sum_{k=1}^{K} \frac{(O_i - P_i)^2}{O_i} \quad (28.28)$$

where O_i is the observed (actual) number of trips for trip pair, i, P_i is the predicted number of trips for trip pair, i, and i is the number of trip pairs that are compared up to K comparisons, where K is selected by the user.

Number of links to test

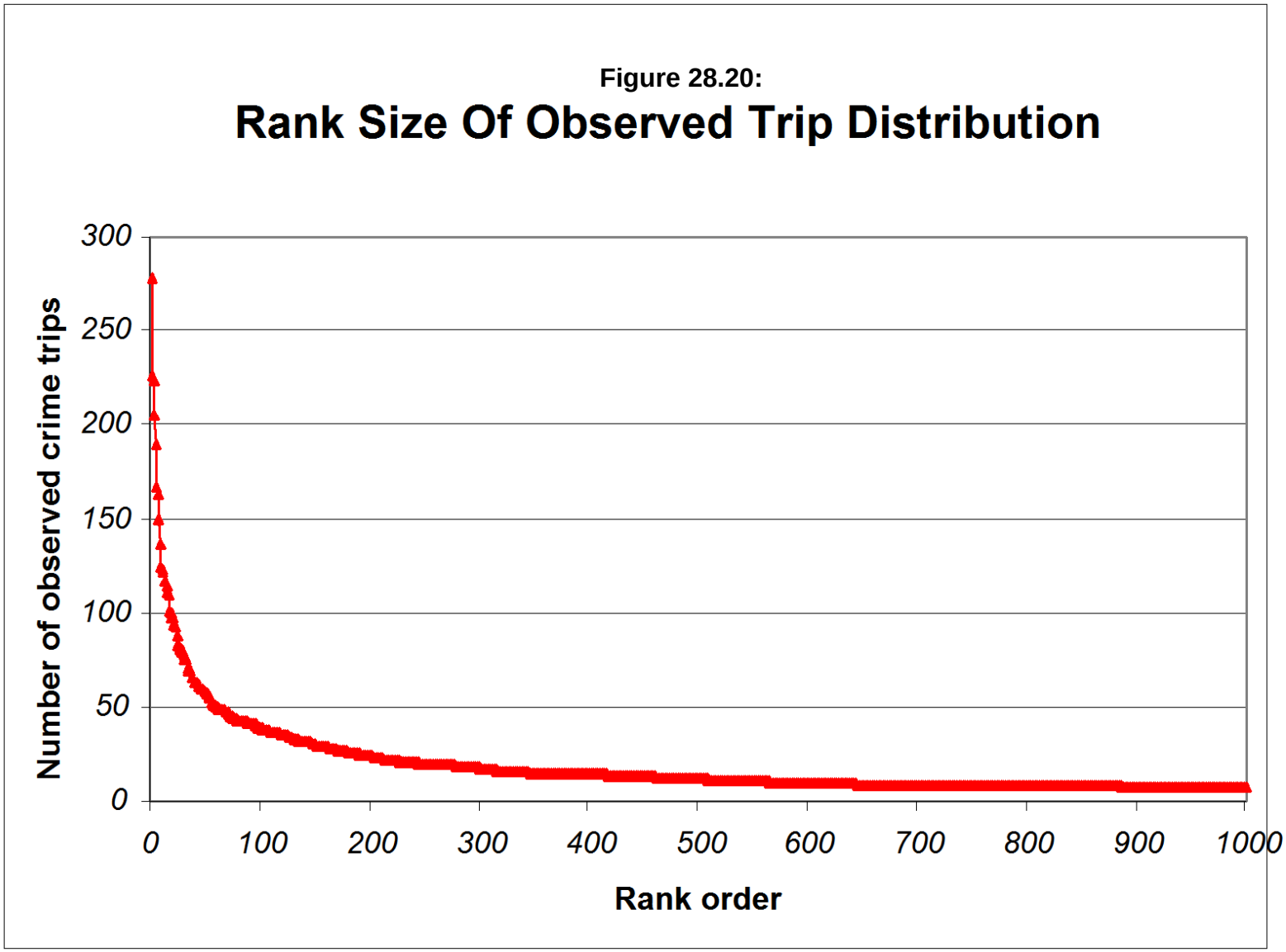
The number of top links that are to be compared depends on how skewed is the distribution. One good way to look at this is to plot the *rank size* distribution of the observed trips. Using the output 'dbf' file for the observed trip distribution (see "Calculate observed origin-destination trips" above), import the file into a spreadsheet. Sort the file in descending order of the trip frequency and create a new variable called "Rank order", which is simply the descending order of the trip frequencies. Then, plot the frequency of trips (FREQ) on the Y axis against the rank order of the trip pairs on the X axis.

Figure 28.20 below shows the rank size distribution of the Baltimore County crime trips. Notice how the distribution is very skewed for the top crime trip pairs, but declines substantially after that. That is, the top trip link (which was an intra-zonal trip pair - zone 654 to itself) had 278 trips. The second top link (also an intra-zonal pair - zone 714 to itself) had 226 trips. The third had 223; the fourth had 205; and so forth. As mentioned above, the top 1000 trip links account for about 47% of all the trips in the matrix, but the first 176 account pairs account for half of that. In other words, if the top 150 to 200 trip pairs are examined, the highest volume links will be included and most of the skewness in the distribution will be accounted for. The remaining distribution, which is not fitted, will be less skewed.

Illustration

An illustration of how comparing the top links can modify a trip distribution model can be given. The same model as shown in Figure 28.12 was run. The pseudo-Chi square test for the first 176 pairs was 5,832 (rounding-off to the nearest integer). However, by modifying the mean distance of the lognormal function a lower Chi square value was obtained. After several iterations, the lowest Chi square value was obtained for a mean distance of 5.2 miles (χ^2 = 5,448). Again, the top links represents only one criterion out of the three mentioned. A good model should balance all three of these.

Figure 28.20:


Rank Size Of Observed Trip Distribution
Number of observed crime trips
300
250
200
150
100
50
0
0
100
200
300
400
500
600
700
800
900
1000
Rank order

Optimizing the Three Evaluation Criteria

The ideal solution would be to have all three evaluation criteria minimized. That is, with an ideal model, there should be very little error between the predicted model and the observed distribution for the number of intra-zonal trips, the trip length distribution, and the top links.

In practice, it is unlikely that any one model will minimize all three types of errors. Thus, a balance (a compromise) must be obtained in order to produce an optimal solution. Since a balance can be obtained in different ways, there are multiple solutions possible.

> **Hint**: In CrimeStat, it is very easy to run through different models. The parameters are input on the "Setup origin-destination model page". The coefficients are calibrated in the "Calibrate origin-destination model" routine on the "Origin-Destination Model" page. The coefficient file which is output is then input into the "Apply predicted origin-destination model" routine on the same page. The comparison between the observed and predicted values is found in the "Compare observed and predicted origin-destination trip lengths" routine. Once set up, iterations of the models can be run very easily. A change is made on the setup page. The model is calibrated. It is then applied to the calibration data set. Finally, a comparison is made. Since the file names remain constant, an entire iteration will take less than a minute on a fast computer.

To illustrate the multiple criteria, Table 28.2 shows the best models for each of the three tests with variations on the mean distance in the model shown in Figure 28.12. All other parameters were held constant. Many models were run to produce this table including testing other functions. These are the three best.

As seen, different models produce the lowest error for each of the criteria. For obtaining the closest fit to the number of intra-zonal trips, the mean distance of the lognormal function was 3.5 miles. For producing the best fit to the top 176 links, the mean distance for the best model was 5.2 models. For producing the best fit for the entire trip length distribution, the mean distance of the best model was 6.0 miles. The question is which one to use?

Table 28.2:

Multiple Criteria in Selecting a Distribution Function

Lognormal function
Standard deviation = 4.7 miles
Coefficient = 1
Origin exponent = 1.0
Destination exponent = 1.06

Mean distance	**Number of Intra-zonal Trips**	**Chi square for top 176 Links**	**Coincidence Ratio**
Observed	8272	-	-
6.0	5463	5814	**0.93**
5.2	6296	**5777**	0.87
3.5	**8275**	5986	0.74

One solution for optimizing decisions

One possible solution is to optimize in the following way:

1. *If* the trip distribution matrix is highly skewed (which will occur with most crime data sets), then it is essential that the top links be replicated closely. This would take priority over the second criterion which is minimizing the error for the trip length distribution, and the third criterion which is minimizing the error in predicting intra-zonal trips.

2. Next fit the model to minimize the Chi square value for the top links. In the example above, this would be the top 176 pairs. Typically, the mean distance has the biggest impact for a lognormal or normal function and this would be adjusted first. For a negative exponential function, the exponent has the strongest impact. For a linear function, the slope has the strongest impact and for a truncated negative exponential, both the peak distance, for the near distance, and the exponent, for the far distance, has the biggest impacts (see Chapter 13). Again, the aim is to produce the Chi square for the top links with the lowest value.

4. Then, while trying to maintain a Chi square value as close to this minimal value as possible, adjust the model to minimize the error in the trip length comparison. In this case, the model with the highest Coincidence Ratio is that which minimizes the error. For lognormal and normal functions, the standard deviation

is the next parameter to adjust. For a negative exponential function, the coefficient should be adjusted next. For a linear function, the intercept would be adjusted next and for a truncated negative exponential the slope would be adjusted next. Again, the aim should be to obtain the highest Coincidence Ratio without losing the fit for the top links.

5. Finally, if it is possible, adjust the exponents of the origins and destinations and the other parameters (e.g., the coefficient in the lognormal and normal distributions) to reduce the error in the total number of intra-zonal trips. Typically, however, these do not alter the results very much. They can be thought of as "fine tuning" adjustments.

Notice that this hierarchy fits the highest volume trip links first, then fits the overall trip length distribution, and finally fits the number of intra-zonal trips.

Illustration

To illustrate, we first start with the model that produced the lowest Chi square. That model used a lognormal function with a mean distance of 5.2 miles, a standard deviation of 4.7 miles, a coefficient of 1, an origin exponent of 1.0 and a destination exponent of 1.06. Varying the standard deviation of the lognormal function produced the following results (Table 28.3).

Table 28.3:
Minimizing the Second Criteria in Selecting a Distribution Function

Lognormal function
Mean distance = 5.2 miles
Standard Deviation = 4.6 miles
Coefficient = 1
Origin exponent = 1.0
Destination exponent = 1.06

Standard deviation	**Number of Intra-zonal Trips**	**Chi square for top 176 Links**	**Coincidence Ratio**
4.5	5809	5789	0.90
4.6	**6057**	**5779**	**0.88**
4.7 (baseline)	6296	5777	0.87
4.8	6526	5780	0.86
4.9	6746	5788	0.84

As the standard deviation was increased, the Coincidence Ratio decreased while the number of intra-zonal trips increased. Of these five different standard deviations, 4.5 produced the highest Coincidence Ratio, but also increased the Chi square statistic for the 176 top links. Since that criterion was set first, we do not want to loosen it substantially during the second adjustment. Consequently, a standard deviation of 4.6 was selected because this increased the Coincidence Ratio slightly while not substantially worsening the Chi square test.

Subsequent tests varying the coefficient of the lognormal function and the exponents of the origin and destination terms did not alter these values. Consequently, the final model that was selected is listed in Table 28.4.

Table 28.14:
Baltimore County Crime Trips: 1993-1997
Optimal Model Selected

Lognormal function
Mean distance = 5.2 miles
Standard deviation = 4.6
Coefficient = 1
Origin exponent = 1.0
Destination exponent = 1.06

The model was re-run with the new parameters used. The top 176 predicted trip links were output and were compared to the top 179 observed trip links (which exceeded 176 because of tied values). The top predicted 176 links accounted for 7,241 trips, or 17.3% of the total number of trips. The top observed 179 links accounted for 9,900 trip, or 23.6% of the total. Compared to the observed distribution, the top 176 predicted links accounted for a smaller proportion of the total trips.

However, the fit was generally better. Figure 28.21 shows the top predicted inter-zonal trip links and compares them to the top observed links while Figure 28.22 shows the top predicted intra-zonal (local) trip links and compares them to the top observed intra-zonal links. Comparing these maps to Figure 28.12 and 28.13 (which mapped the top 1000 links, not the top 176), the fit is a bit better for the major links, which is what we optimized. The fit is not perfect; it probably will never be. But, it is reasonably close.

Of course, this is not the only way to optimize and different users might approach it differently (e.g., minimizing the intra-zonal trips first, then the overall trip length distribution, and finally the top links). It has to be realized that optimizing in a different order will probably produce varying results; there is not, unfortunately, a single optimum solution to these three

Figure 28.21:

Comparison of Predicted and Observed Crime Trips
Top Zone-to-Zone Trips from Optimized Model
All Crime Types

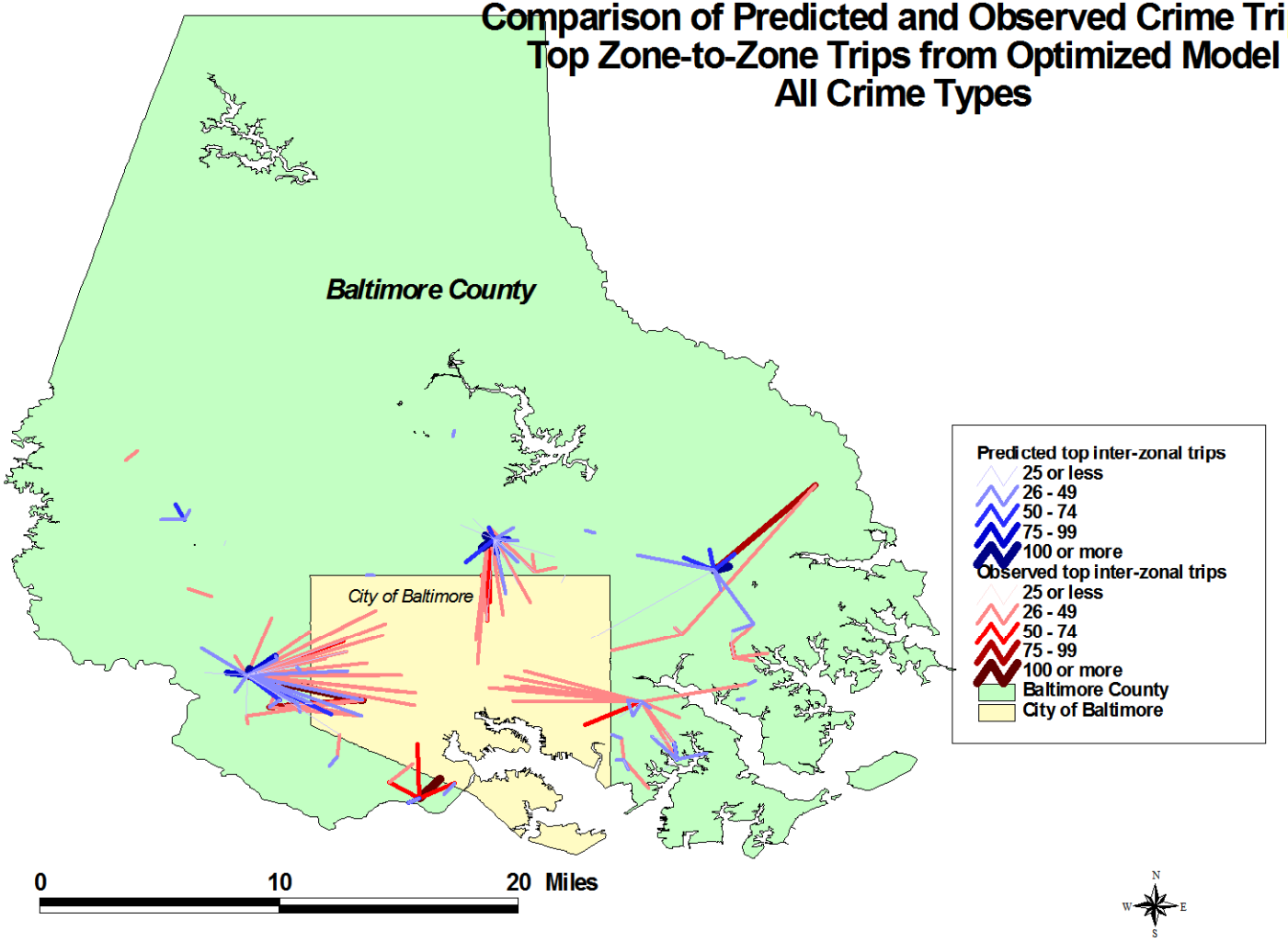
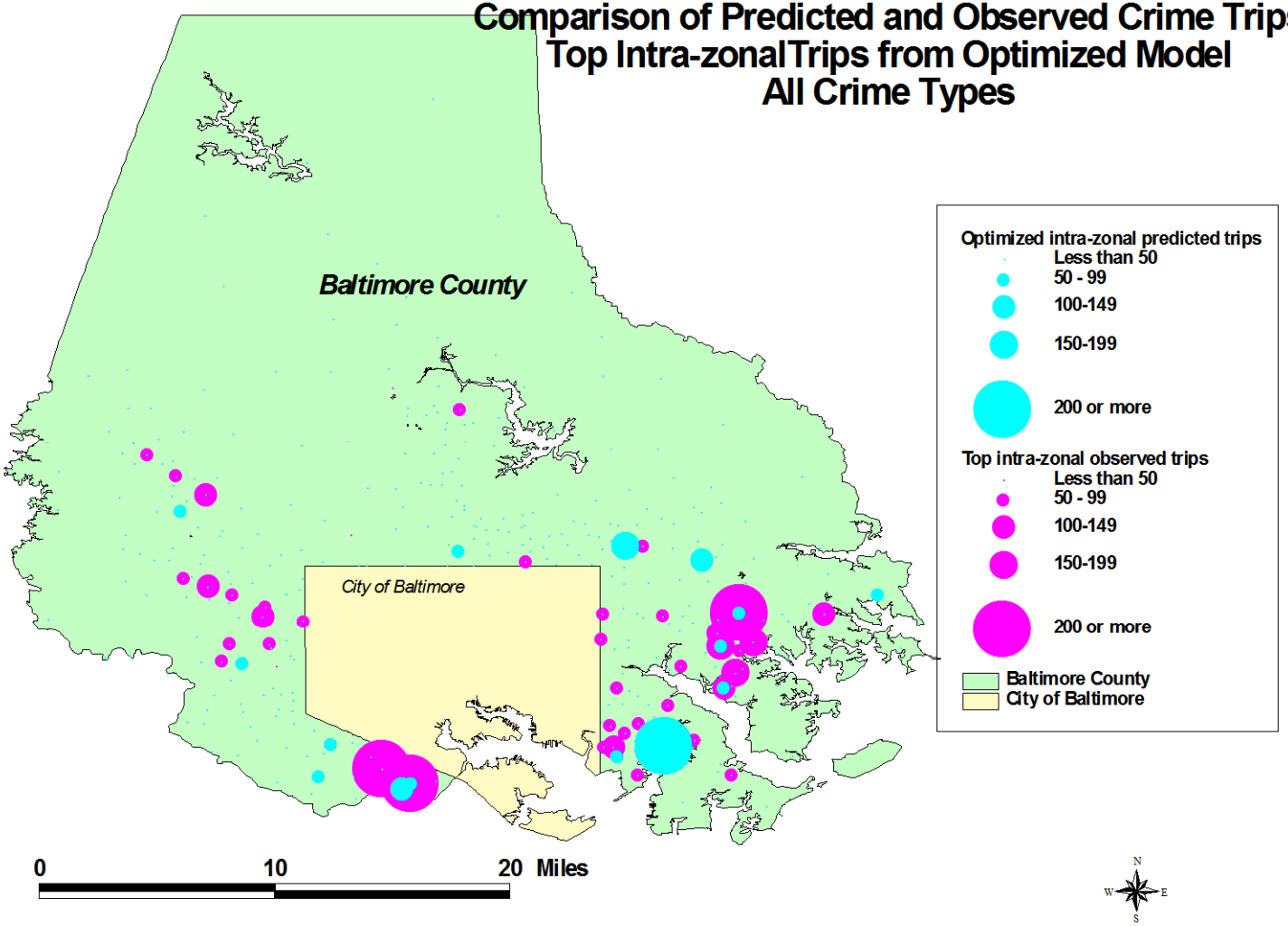


Figure 28.22:

Comparison of Predicted and Observed Crime Trips
Top Intra-zonalTrips from Optimized Model
All Crime Types



criteria. That is why it is important to explicitly define how an optimal solution will be obtained. In that way, users of the model can be cognizant of where the model is most accurate and where it is probably less accurate.

Implementing the Comparisons in *CrimeStat*

The mechanics of conducting the tests is fairly straightforward. The three tests are implemented in the "Compare Observed and Predicted Trip Lengths" routine on the last page of the Trip distribution module.

Observed trip file

Select the observed trip distribution file by clicking on the Browse button and choosing the appropriate file.

Observed number of origin-destination trips

Specify the variable for the observed number of trips. The default name is FREQ.

Orig_ID

Specify the ID name for the origin zone. The default name is ORIGIN. Note that the ID's used for the origin zones must be the same as in the destination file and the same as in the predicted trip file if the top links are to be compared.

Orig_X

Specify the name for the X coordinate of the origin zone. The default name is ORIGINX.

Orig_Y

Specify the name for the Y coordinate of the origin zone. The default name is ORIGINY.

Dest_ID

Specify the ID name for the destination zone. The default name is DEST. Note that all destination ID's should be in the origin zone file and must have the same names and the same as in the predicted trip file if the top links are to be compared.

Dest_X

Specify the name for the X coordinate of the destination zone. The default name is DESTX.

Dest_Y

Specify the name for the Y coordinate of the destination zone. The default name is DESTY.

Predicted trip file

Select the predicted trip distribution file by clicking on the Browse button and choosing the appropriate file.

Predicted number of origin-destination trips

Specify the variable for the observed number of trips. The default name is PREDTRIPS.

Orig_ID

Specify the ID name for the origin zone. The default name is ORIGIN. Note that the ID's used for the origin zones must be the same as in the destination file and the same as in the observed trip file if the top links are to be compared.

Orig_X

Specify the name for the X coordinate of the origin zone. The default name is ORIGINX.

Orig_Y

Specify the name for the Y coordinate of the origin zone. The default name is ORIGINY.

Dest_ID

Specify the ID name for the destination zone. The default name is DEST. Note that all destination ID's should be in the origin zone file and must have the same names and the same as in the observed trip file if the top links are to be compared.

Dest_X

Specify the name for the X coordinate of the destination zone. The default name is DESTX.

Dest_Y

Specify the name for the Y coordinate of the destination zone. The default name is DESTY.

Select bins

Specify how the bins (intervals) will be defined. There are two choices. One is to select a fixed number of bins. The other is to select a constant interval.

Fixed number

This sets a fixed number of bins. An interval is defined by the maximum distance between zone divided by the number of bins. The default number of bins is 25. Specify the number of bins.

Constant interval

This defines an interval of a specific size. If selected, the units must also be chosen. The default is 0.25 miles. Other distance units are nautical miles, feet, kilometers, and meters. Specify the interval size.

Compare top links

The "Compare top <value> links" dialogue implements a comparison of the top links. The user specifies the number of links to be compared. The default is 100. The routine

calculates a Chi square statistic for these links. Note that in order to make the comparison, the origin and destination ID's must be the same for both the observed and predicted trip files.

Save comparison

The output is saved as a 'dbf' file specified by the user.

Table output

The table output includes summary information and:

1. The number of trips in the observed origin-destination file
2. The number of trips in the predicted origin-destination file
3. The number of intra-zonal trips in the observed origin-destination file
4. The number of intra-zonal trips in the predicted origin-destination file
5. The number of inter-zonal trips in the observed origin-destination file
6. The number of inter-zonal trips in the predicted origin-destination file
7. The average observed trip length
8. The average predicted trip length
9. The median observed trip length
10. The median predicted trip length
11. The Coincidence Ratio (an indicator of congruence varying from 0 to 1)
12. The D value for the Komolgorov-Smirnov two-sample test
13. The critical D value for the Komolgorov-Smirnov two-sample test
14. The p-value associated with the D value of Komolgorov-Smirnov two-sample test relative to the critical D value.
15. The pseudo-Chi square test for the top links

and for each bin:

16. The bin number
17. The bin distance
18. The observed proportion
19. The predicted proportion

File output

The saved file includes:

1. The bin number (BIN)

2. The bin distance (BINDIST)
3. The observed proportion (OBSERVPROP)
4. The predicted proportion (PREDPROP)

Graph

While the output page is open, clicking on the graph button will display a graph of the observed and predicted trip length proportions on the Y-axis by the trip length distance on the X-axis. This would produce a similar graph to that seen in Figures 28.16 through 28.19 above.

Uses of Trip Distribution Analysis

There are a number of uses for the trip distribution analysis. First, for policing, an analysis of the actual (observed) trip distribution can be valuable. Second, the predicted model has value, above-and-beyond the analysis of the actual distribution.

Utility of Observed Trip Distribution Analysis

This information by itself can be very useful for police. Two applications will be discussed.

Crime prevention efforts

A major application is using the data shown in a trip distribution map to guide enforcement efforts. For example, in Baltimore County, with the crimes occurring at the five shopping malls, the origin locations can be more easily seen. This has utility for police. First, the police can intervene more effectively on the routes leading from likely origin locations. They can patrol those routes more heavily and, perhaps, intervene more frequently. By using the information from the trip distribution analysis, they make their enforcement efforts smarter. Second, they can conduct crime prevention efforts more effectively. By knowing the likely origin of offenders, intervention efforts in the origin zones may head off some of these incidents. Programs such as *weed-and-seed* and after-school programs depend on providing alternative facilities for youth, hoping to redirect them to more constructive activities. These facilities can be placed in locations where many crimes originate.

Improved Journey-to-crime analysis

A second application is in guessing the likely origin of a serial offender. In Chapter 13, theories of travel behavior by a serial offender was discussed. The resulting analysis (geographic profiling, Journey-to-crime analysis) utilized information on the distribution of

incidents committed by the offender. On the other hand, the trip distribution pattern seen in Figure 28.4 provides a probability map of offender locations and gives more information than was evident in the Journey-to-crime model. That model assigned a likelihood of the offender living at a location (the origin) on the basis of the distribution of the incidents. There was no additional information used about likely origin locations. This trip distribution map, on the other hand, points to certain zones as being the likely origin for offenses committed at the major destination locations. There is more 'structure' in this analysis than in the Journey-to-crime logic. This is the basis for the Bayesian Journey-to-crime approach discussed in Chapter 14.

One can think of this in terms of a quasi-Bayesian approach to guessing the likely origin of an offender. The geographic profiling/Journey-to-crime logic assumes no *prior probabilities*. The only information that is used is the distribution of crimes committed by a serial offender and a model of crime travel distance (essentially, an impedance function). The trip distribution map, on the other hand, points to certain locations as being the likely origin for incidents. Admittedly, this is based on a large sample of cases rather than one particular serial offender. But, the map points to certain prior probabilities for an origin location. The Bayesian Journey-to-crime routine combines those two pieces of information. As mentioned in Chapter 14, tests on more than 1000 serial offenders in four cities (in three countries) showed that the method was 10-15% more accurate than the traditional journey-to-crime approach and as precise.

In other words, the empirical description of crime travel patterns is useful for policing, above-and-beyond any modeling that is developed.

Utility of Predicted Trip Distribution Analysis

The model also has a lot of utility for both policing and crime analysis. A number of examples will be given. First, it can be used for **forecasting**. By calibrating the model on one data set, it be applied to a future data set. As mentioned in Chapter 26, much of the population and employment data that form the basis of a trip generation model comes from a Metropolitan Planning Organization (MPO). Most MPOs in the United States also make forecasts of future population and employment. Those forecasts can be, in turn, converted into forecasts of future crime origins and crime destinations. Thus, on the assumption that the distribution trends will remain the same over time, the trip distribution model can be applied to the forecast set of origins and destinations. This could allow an examination of possible changes in the crime distribution (assuming that the future forecasts are correct and that the trip distribution coefficients remain constant).

Second, a model of crime trip distribution can be useful for modeling **changes in land uses**. For example, if a new shopping mall is being planned, one can take the existing trip generation model and adjust it to fit the planned situation (e.g., adding 500 retail jobs to the zone

in which the mall is being developed). Then, the trip generation model is re-run with the new expected data, and the trip distribution model is applied to the predicted crime origins and crime destinations. The result would be a model of likely crime trips to the new shopping mall. This can be useful to the mall developers, to future businesses, and to the police. If it turns out that the model forecasts there will be a sizeable number of crime trips to that mall, then preventive actions can be developed before the mall is built (e.g., improving security design in the mall; improving the parking lot arrangement).

Third, a model of crime trip distribution can help in analyzing **future interventions**. For example, increasing police patrols in a high crime attraction area can be examined as to possible effectiveness before taking the trouble to reorganize deployment. Or, adding a new drug treatment center or a new youth center can be modeled as to its possible effectiveness in changing the nature of crime trips. Again, the input is at the data level, which affects the trip generation model. But the trip distribution model is applied to the new outputs from the trip generation model. The advantage of a model is that it explores a set of interventions without having to actually having to implement them; it is a 'thinking' tool for planning change.

Fourth, and finally, a crime trip distribution model is helpful in developing **crime theory**. As indicated in Chapter 25, the theory of crime travel has been very elementary up to now. The primary focus of analysis has been only on the destinations and on the trip lengths as measured by distance traveled. A trip distribution model, on the other hand, analyzes both trip destinations and trip origins, and can include a more sophisticated measure of impedance than simple distance. Because analysis is conducted over a larger area (a jurisdiction or a metropolitan area), the hierarchy of crime trips can be analyzed as an interaction between origins and destinations. In short, a crime trip distribution model is a 'quantum leap' in sophistication and complexity compared to the usual Journey-to-crime types of models. Hopefully, it will generate even more sophisticated types of models. The attachment illustrates how the crime travel demand model was used to examine possible interventions to reduce DWI trips ending in crashes in Baltimore County.

The next chapter continues the travel demand model by examining how crime trip links are split into different travel modes. That is, the trip distribution model estimates the number of trips flowing from each origin zone to each destination zone. The mode split model then breaks these trips into distinct travel modes.

References

Andersson, T. (1897). *Den Inre Omflyttningen*. Norrland: Mälmo.

Bernasco, W. & Block, R. (2009). Where offenders choose to attack: A discrete choice model of robberies in Chicago. *Criminology* 47(1): 93-130.

Bossard, E. G. (1993). RETAIL: Retail trade spatial interaction. In Richard E. Klosterman, Richard K. Brail & Earl G. Bossard, *Spreadsheet Models for Urban and Regional Analysis*. Center for Urban Policy Research, Rutgers University: New Brunswick, NJ, 419-448.

Bright, M. L. & Thomas, D. S. (1941). Interstate migration and intervening opportunities, *American Sociological Review*, 6, 773-783.

Carnegie-Mellon University (1975). *Security of Patrons on Urban Public Transportation Systems*. Transportation Research Institute, Carnegie-Mellon University: Pittsburgh, PA.
Cliff, A. D. & Haggett, P. (1988). *Atlas of Disease Distributions*. Blackwell Reference: Oxford.

Domencich, T. & McFadden, D. (1975). *Urban Travel Demand: A Behavioral Analysis*. North Holland Publishing Company: Amsterdam & Oxford (republished in 1996). Also found at http://emlab.berkeley.edu/users/mcfadden/travel.html. Accessed April 28, 2012.

FHWA (1997). *Model Validation and Reasonableness Checking Manual*. Prepared by Barton-Aschman Associates, Inc and Cambridge Systematics, Inc for the Travel Model Improvement Program, Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://ops.fhwa.dot.gov/freight/publications/qrfm2/sect08.htm. Accessed May 31, 2012.

Field, B. & MacGregor, B. (1987). *Forecasting Techniques for Urban and Regional Planning*. UCL Press, Ltd: London.

Foot, D. (1981). *Operational Urban Models*. Methuen: London.

Hägerstrand, T. (1957). Migration and area: survey of a sample of Swedish migration fields and hypothetical considerations on their genesis. *Lund Studies in Geography, Series B, Human Geography*, 4, 3-19.

Huff, D. L. (1963). A probabilistic analysis of shopping center trade areas. *Land Economics*, 39, 81-90.

Isbel, E. C. (1944). Internal migration in Sweden and intervening opportunities, *American Sociological Review*, 9, 627-639.

References (continued)

Isard, W. (1979). *Location and Space-Economy: A General Theory Relating to Industrial Location, Market Areas, Land Use, Trade, and Urban Structure* (originally published 1956). Program in Urban and Regional Studies, Cornell University: Ithaca, NY.

Johnson, M.A. (1978). Attribute importance in multiattribute transportation decisions, *Transportation Research Record*, 673, 15-21.

Kanji, G. K. (1993). *100 Statistical Tests*. Sage Publications: Thousand Oaks, CA.

Levine, N. & Canter, P. (2011). "Linking origins with destinations for DWI Motor Vehicle Crashes: An application of crime travel demand modeling". *Crime Mapping*, 3, 7-41.

Levine, N. & Wachs, M. (1986). Bus Crime in Los Angeles: II - Victims and Public Impact. *Transportation Research*. 20 (4), 285-293.

Massey, F. J., Jr (1951). The distribution of the maximum deviation between two sample cumulative step functions. *Annuals of Mathematical Statistics*, 22, 125-128.

NCHRP (1995). *Travel Estimation Techniques for Urban Planning*. Project 8-29(2). National Cooperative Highway Research Program, Transportation Research Board: Washington, DC. http://www.trb.org/main/blurbs/160284.aspx. Accessed May 29, 2012.

Oppenheim, N. (1980). *Applied Models in Urban and Regional Analysis*. Prentice-Hall, Inc.: Englewood Cliffs, NJ.

Ortuzar, J. D. & Willumsen, L. G. (2001). *Modeling Transport* (3rd edition). J. Wiley & Sons: New York.

Porojan, A. (2000). Trade flows and spatial effects: the Gravity Model revisited. Conference on Managing Economic Transition in Eastern Europe: Emerging Research Issues. The Manchester Metropolitan University: Manchester, England, January.

Ravenstein, E. G. (1885). The laws of migration. *Journal of the Royal Statistical Society*. 48.

Reilly, W. J. (1929). Methods for the study of retail relationships. *University of Texas Bulletin*, 2944.

Roemer, F. & Sinha, K. (1974). Personal security in buses and its effects on ridership in Milwaukee, *Transportation Research Record*, 487, 13-25.

References (continued)

Schnell, J. B., A. J. Smith, K. R. Dimsdale, & L. J. Thrasher (1973). *Vandalism and Passenger Security: A Study of Crime and Vandalism on Urban Mass Transit Systems in the United States and Canada*. Prepared by the American Transit Association for the Urban Mass Transportation Administration (now Federal Transit Administration), U. S. Department of Transportation. National Technical Information Service: Springfield, VA. PB 236-854.

Siegel, S. (1956). *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill: New York.

Smirnov, N. V. (1948). Table for estimating the goodness of fit of empirical distributions. *Annuals of Mathematical Statistics*, 19, 279-281.

Stewart, J. Q. (1950). The development of social physics. *American Journal of Physics*, 18, 239-53.

Stouffer, S. A. (1940). Intervening opportunities: a theory relating mobility and distance. *American Sociological Review*, 5, 845-67.

Wachs, M., Taylor, B., Levine, N. & Ong, P. (1993). The Changing Commute: A Case Study of the Jobs/Housing Relationship Over Time. *Urban Studies*. 30 10, 1711-1729.

WASHCOG (1974). *Citizen Safety and Bus Transit*. Metropolitan Washington Council of Governments. National Technical Information Service, Springfield, VA. PB 237-740/AS.

Wilson, A. G. (1970). *Entropy in Urban and Regional Planning*. Leonard Hill Books: Buckinghamshire.

Zhao, F., Chow, L-F, Li, M-T, Gan, A., & Shen, D. L. (2001). *Refinement of FSUTMS Trip Distribution Methodology*. Lehman Center for Transportation Research, Florida International University: Miami, FL..

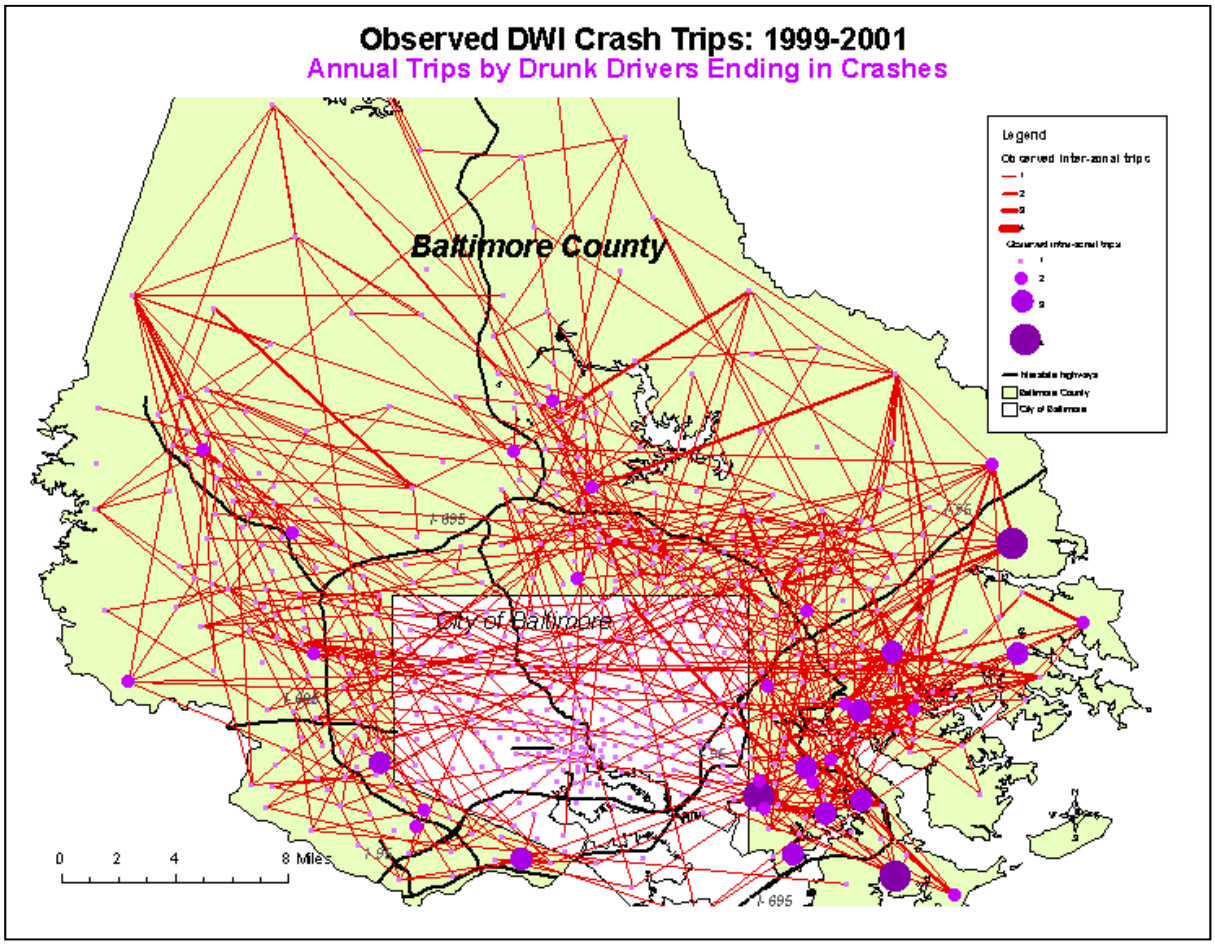
Zipf, G. K. (1949). *Human Behaviour and the Principle of Least Effort*. Cambridge.

Modeling DWI Trips That End in Crashes in Baltimore County, MD

Ned Levine
Ned Levine & Associates
Houston, TX

Phil Canter
Towson University
Towson, MD

A crime travel demand study was conducted on 862 Driving While Intoxicated (DWI) motor vehicle crash trips that occurred in Baltimore County, Maryland between 1999 and 2001. Factors associated with both the residence location of the drivers and the crash location were identified. The crime travel demand model was used to simulate the likely outcome of concentrating on a few zones with targeted interventions. It was estimated that a 7.5% reduction in DWI crashes could be obtained by targeting 3% of the origin zones and 6% of the destination zones with anti-DWI efforts. The full study can be found in Levine, N. & Canter, P. (2011), Linking origins with destinations for DWI Motor Vehicle Crashes: An application of crime travel demand modeling". *Crime Mapping*, 3, 7-41.

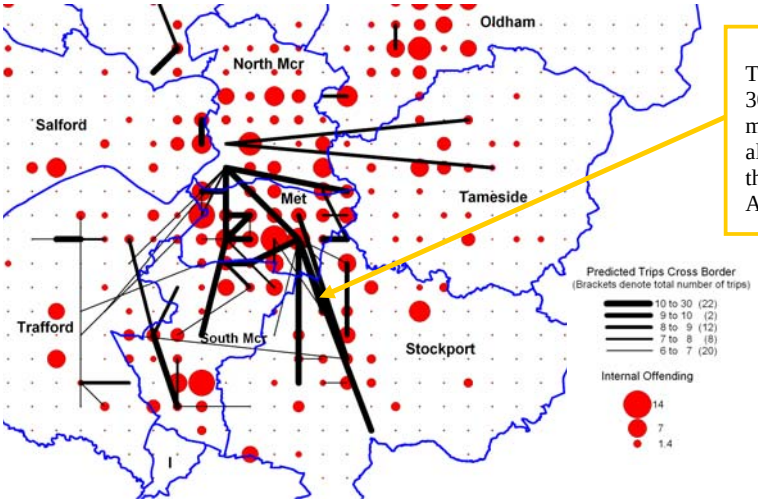


Targeting Crime on Public Transport:
An Example from Greater Manchester, England

Daisy Smith & Steph Winstanley
Strategic Analytical Partnership Co-ordinators
Greater Manchester Against Crime Central Team

The aim of the Greater Manchester Against Crime Central Team was to provide GMPTE (Greater Manchester Passenger Transport Executive) with an evidence base for their resources to address incidents of crime and anti-social behaviour on public transport during a Greater Manchester Partnership Day of Action. The analysis made use of the *Crimestat* Crime Travel Demand module to map the 'journey to crime' (home address to offence location) taken by personal robbery offenders within Greater Manchester. As a result GMPTE were able to identify their role in the partnership operation as they could easily visualise the bus and Metrolink tram system routes that ran coterminous with the most frequent journeys taken by offenders.

Personal Robbery: Internal Offending and Predicted Cross Border Trips



The map shows a high number of predicted trips (10 to 30) between Stockport and central Manchester. This mirrors the route of the 192 bus service, which travels along the A6 between Stockport and Manchester that was then identified as a main target for the Partnership Day of Action.

During the Day of Action, Gateway checks were conducted on the key public transport routes (bus routes and the Metrolink tram system) that were identified through *Crimestat* analysis. The Gateway checks consisted of staff from a range of agencies deployed on static and mobile patrols in order to identify fare evasion/ fraud and conduct intelligence checks. The agencies involved included Greater Manchester Police, GMPTE, Carlisle Security (independent enforcement agency) and the UK Border Agency. The public transport routes identified through *Crimestat* were targeted with much success and resulted in 7058 passengers being checked, 496 buses boarded, 76 people identified without valid tickets, 28 intelligence checks, 22 Bus Operator penalties issued and 22 arrests (including possession of illegal substances, robbery, fraud, outstanding warrant). The total fraud prevented through the Gateway Checks was estimated to be £3784.50 and extremely positive feedback was received from all agencies involved.

Chapter 29:

Crime Mode Split

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Chapter 29:

Crime Mode Split

In this chapter, the third modeling step in the crime travel demand model is discussed, mode split. *Mode split* involves separating (splitting) the predicted trips from each origin zone to each destination zone into distinct travel modes (e.g., walking, bicycle, driving, train, bus).

This model has both advantages and disadvantages for crime analysis. At a theoretical level, it is the most developed of the four stages since there has been extensive research on travel mode choice. For crime analysis, on the other hand, it represents the 'weakest link' in the analysis since there is very little available information on travel mode by offenders. Since researchers cannot interview the general public in order to document crimes committed by respondents nor, in most cases, even interview offenders after they have been caught, there is very little information on travel mode by offenders that has been collected.[1] Consequently, we have to depend on the existing theory of travel mode choice and adapt it intuitively to crime data. The approach is solely theoretical and depends on the validity of the existing theory and on the intuitiveness of guesses. Hopefully, in the future, there will be more information collected that would allow the model to be calibrated against some real data. But, for the time being, we are limited in what can be done.

Theoretical Background

The theoretical background behind the mode split module is presented first. Next, the specific procedures are discussed with the model being illustrated with data from Baltimore County.

Utility of Travel and Mode Choice

The key aim of mode choice analysis is to distinguish the travel mode that travelers (or, in the case of crime, offenders) use in traveling between an origin location and a destination location. In the travel demand model, the choice is for travel between a particular origin zone and

---

[1] There is no reason this data could not be collected. Typically, many police departments collect information on 'Method of departure' from a crime scene. When a police report is taken, the victim is sometimes asked how the offender left the scene. In most cases, the information is not recorded on the police forms, or at least those that have been examined. This information is probably unreliable in any case since many offenders will take the bus or leave their car nearby while they walk/run to the crime scene. Still, if police departments were to put more effort into collecting this information and, perhaps, to validating it with arrested offenders, then it is possible to build up reliable data sets that can be used to model mode split. Until then, unfortunately, we have to rely on theory rather than evidence.

a particular destination zone. Thus, the trips that are distributed from each origin zone to each destination zone in the trip distribution module are further split into distinct travel modes.

With few exceptions, the assumption behind the mode split decision is for a two-way trip. That is, if an offender decides on driving to a particular crime location, we normally assume that this person will also drive back to the origin location. Similarly, if the offender takes a bus to a crime location, then that person will also take the bus back to the origin location. There are, of course, exceptions. A car thief may take a bus to a crime location, then steal a car and drive back. But, in general, without information to the contrary, it is assumed that the travel mode is for a round trip journey.

Underlying the choice of a travel mode is assumed to be a *utility function*. Chapter 21 discusses the economic theory of utility choice, so the discussion here will be brief. Essentially, mode split utility is a function that describes the benefits and costs of travel by that mode (Ortuzar & Willumsen, 2001). This can be written with a conceptual equation:

$$Utility = f(benefits, costs) \tag{29.1}$$

where 'f' is some function of the benefits and the costs. The benefits have to do with the advantages in traveling to a particular destination from a particular origin while the costs have to do with the real and perceived costs of using a particular mode. Since the benefits of traveling to a particular destination from a particular origin are probably equal, the differences in utility between travel modes essentially represent differences in costs (Train, 2009). Thus, Equation 29.1 breaks down to:

$$Utility = F(costs) \tag{29.2}$$

where 'F' is another function but this time of only the costs. If different travel modes (e.g., driving, biking, walking) are each represented by a separate utility cost function, then they can be compared:

$$Utility_1 = F_1(cost_1 + cost_2 + cost_3 + \ldots . + cost_k) \tag{29.3a}$$
$$Utility_2 = F_1(cost_1 + cost_2 + cost_3 + \ldots . + cost_k) \tag{29.3b}$$
$$Utility_3 = F_1(cost_1 + cost_2 + cost_3 + \ldots . + cost_k) \tag{29.3c}$$
$$\vdots$$
$$Utility_l = F_l(cost_1 + cost_2 + cost_3 + \ldots . + cost_k) \tag{29.3l}$$

where Utility_1 through Utility_L represents l distinct travel modes, cost_1through cost_k represent k cost components and are variables, and F_1 through F_l represent l different utility functions (one for each mode).

There are several observations that can be made about this representation. First, each of the cost components can be applied to all modes. However, the cost components are variables in that the values may or may not be the same. For example, if cost_1 is the operating cost of traveling from an origin to a destination, the cost for a driver is, of course, a lot higher than for a bus passenger since the latter person shares that cost with other passengers. Similarly, if cost_2 is the travel time from a particular origin zone to a particular destination zone, then travel by private automobile may be a lot quicker than by public bus. Time differences can be converted into costs by applying some type of hourly wage/price to the time. To take one more example, for driving mode, there could be a cost in parking (e.g., in a central business district); for transit use, on the other hand, this cost component is zero. In other words, each of the travel modes has a different cost structure. The same costs can be enumerated, but some of them will not apply (i.e., they have a value of 0).

Second, the costs can be perceived costs as well as real costs. For example, a number of studies have demonstrated that private automobile use is seen by most people as far more convenient to than bus or train (e.g., see Schnell, Smith, Dimsdale, & Thrasher, 1973; Roemer & Sinha, 1974; WASHCOG, 1974; Carnegie-Mellon University, 1975; Johnson, 1978; Levine & Wachs, 1986). 'Convenience' is defined in terms of ease of access and effort involved in travel (e.g., how long it takes to walk to a bus stop from an origin location, the number of transfers that have to made to reach a final destination, and the time it takes to walk from the last bus stop to the final destination). While it is sometimes difficult to separate the effects of convenience from travel itself, it is clear that most people perceive this as a dimension in travel choice. In turn, convenience can be converted into a monetary value in order to allow it to be calculated in a cost equation, for example how much people are willing to pay in time savings to yield an equivalent amount of convenience (e.g., asking how many more minutes in travel time by bus an individual would be willing to absorb in order to give up having to drive).

Third, these costs can be considered at an aggregate as well as individual level. At an aggregate level, they represent average or median costs (e.g., the average time it takes to travel between zone A and zone B by private automobile, bus, train, walking, or biking; the average dollar value assigned by a sample of survey respondents to the convenience they associate in traveling by car as opposed to bus).

On the other hand, at an individual level, the costs are specific to the individual. For example, travel time differences between car and bus can be converted into an hourly wage using

the individual's income (i.e., someone making $100,000 a year is going to price that time savings differently than someone making only $25,000 a year).

Fourth, a more controversial point, the specific mathematical function that ties the costs together into a particular utility function may also differ. Typically, most travel demand models have assumed that a similar mathematical function is used for all travel modes; this is the negative exponential function described below (Ortuzar and Willumsen, 2001; Domencich & McFadden, 1975). However, there is no reason why different functions cannot be used. Thus, the equations above identify different functions for the modes, F_1 through F_L. One can think of this in terms of *weights*. Each of the different mathematical function weigh the cost components differently.

It is an empirical question whether individuals apply different functions to evaluating the different modes. For example, most people would not drive just to travel one block (unless it was pouring rain or unless a heavy object had to be delivered or picked up). Even though it is convenient to get into a vehicle and drive the one block, most people see the effort involved (and, most likely, the fuel and oil costs) as not being worth it.

In other words, it appears that a different utility function is being applied to walking as opposed to driving (i.e., walk up to a certain distance; drive thereafter). A strict utility theorist might disagree with this interpretation saying that the per minute cost of walking the one block and back was less than the monetized per minute cost of operating the vehicle (which may include opening a garage door, getting into the vehicle, starting the vehicle, driving out of the parking spot, closing the garage door, and then driving the one block). In other words, it could be argued that the difference in behavior has to do with the values of the different cost components, rather than the way they are *weighed* together (the mathematical function). In retrospect, one can explain any difference. It is argued in this chapter, however, that crime trips appear to show different likelihoods by travel mode and that treating each of these functions as distinct allows more flexibility in the framework.

Discrete Choice Analysis

No matter how the utility functions are defined, they have to be combined in such a way as to allow a discrete choice. That is, an offender in traveling from zone A to zone B makes a discrete choice on travel mode. There may be a probability for travel by each mode, for example 60% by car and 40% by bus. But, for an individual, the choice is car or bus, not a probability. The probabilities are obtained by a sample of individuals, for example of 10 individuals 6 went by car and 4 went by bus. But, still, at the individual level, there is a distinct choice that was made.

Multinomial Logit Function

A common mathematical framework that used is for mode choice modeling at an aggregate level is the *multinomial logit function* (Ortuzar & Willumsen, 2001; Oppenheim, 1980; Domincich & McFadden, 1975; Stopher & Meyburg, 1975). Chapter 21 on discrete choice modeling discussed this model extensively but for individual decision makers. If there was data available on the individual travel mode choices made by offenders, then an individual level discrete choice model could be constructed. However, till now, such data has not been available. Consequently, the modeling has to occur for zone-to-zone interactions rather than for individuals choosing among zones.

The multinomial logit model used is for aggregate zone-to-zone flows. That is, for all trips from zone i to zone j, a multinomial model can be defined as:

$$P_{ijL} = \frac{e^{V_{ij_L}}}{\sum_{j=1}^{J} e^{V_{ijL}}} \tag{29.4}$$

where P_{ijL} is the probability of using a mode for any particular trip link (particular origin zone i to particular destination j), L is the travel mode, V_{ni} *is the representative utility* (that observed by the researcher/analyst as opposed to total utility which includes unobserved factors) for zone i, among j alternative destinations. The representative utility, in turn, is seen as a linear combination of independent variables (predictors) for travel from origin zone i, to destination zone j:

$$V_{ijL} = \boldsymbol{\beta}'_{jL}\boldsymbol{X}_i = \sum_{j=1}^{J} \beta_{jL} X_{ijL} \tag{29.5}$$

where V_{ijL} is the utility of traveling from origin zone i, to destination zone j, L is the travel mode, and β_{jL} are coefficients.

Substituting equation 29.5 into equation 29.4, we have:

$$P_{ijL} = \frac{e^{\sum_{j=1}^{J} \beta_{jL} X_{ijL}}}{\sum_{L=1}^{J} e^{\sum_{j=1}^{J} \beta_{jL} X_{ijL}}} \tag{29.6}$$

which relates the linear combination for any one mode to the sum of the linear combinations of all modes.

Several observations can be made about this function. First, the multinomial logit model relates the choice of alternatives to differences in the *characteristics of the zones*, both

origin and destination (the X_{ijl}), rather than to the differences in the destination zones themselves. This is different than the more general conditional logit model which relates the choices to the characteristics of the alternatives (destinations) and to interactions between the origin zones and the destination zones (and is also discussed in Chapter 21).

Second, each travel mode, *L*, has its own costs and benefits and can be evaluated by itself. That is, each origin zone have a different set of destination alternatives according to the characteristics of both the origin and destination zones. That is, there is a distinct utility function for each mode. This is the numerator of Equation 29.6. However, the choice of any one mode is dependent on its utility value relative to other modes (the denominator of the equation). The more choices that are available, obviously, the lower the probability that a particular origin-destination zone combination will have for that mode. But the value associated with the mode (the utility) does not change. As mentioned above, we generally assume that the benefit of traveling between any two zones is identical for all modes and, hence, any differences are due to costs.

Third, the mathematical form is the negative exponential. The exponential function is a growth function in which growth occurs at a constant *rate* (either positive - growth, or negative - decline). The use of the negative exponential assumes that the costs are related to the likelihood as a function that declines at a constant rate. It is actually a 'disincentive' or 'discount' function rather than a utility function, *per se*. That is, as the costs increase, the probability of using that mode decreases, all other things being equal. Still, for historical reasons, it is still called a utility function.

Fourth, for any one mode, the total cost is a logarithmic function of individual cost components:

$$Utility_{iL} = e^{\sum_{j=1}^{J} \beta_{jL} X_{ijL}} \tag{29.7}$$

$$Ln(Utility_{iL}) = \sum_{j=1}^{J} \beta_{jL} X_{ijL} \tag{29.8}$$

where the cumulative cost is made up of independent predictors X_1, X_2 through X_J, and β_1 through β_k are coefficients for the individual cost components. Thus, we see that the utility function is a loglinear model, as was seen in Chapter 13. Thus, the utility function is Poisson distributed, declining at a constant *rate* with increasing cumulative costs. Domincich and McFadden (1975) point out that the error term is not Poisson distributed, but skewed as a Type I extreme value distribution (sometimes called a Gumbel distribution; Train, 2009). As discussed in Chapters 16-17, there are a variety of different Poisson models that incorporate skewed error

terms (Poisson-Gamma, Poisson-lognormal, COM-Poisson) and which could also be used to fine-tune the fit. Nevertheless, the mean utility is a Poisson-type function.

Generalized Relative Utility Function

One can generalize this further to allow any type of mathematical function. While the Poisson has a long history and is widely used, allowing other non-linear functions allows greater flexibility. It is possible that individuals apply different *weighting* systems in evaluating different modes (e.g., a negative exponential for walking, but a lognormal function for driving). We certainly see what appear to be different functions when the actual travel behavior of individuals are examined (e.g., homeless individuals don't walk everywhere even though the cost of walking long distances is cheaper in travel time than taking a bus[2]; people don't drive or take a bus for very short distances, say a block or two). Therefore, if we allow that there are different travel functions for different modes, then more flexibility is possible than by assuming a single mathematical function.

We can, therefore, write a *generalized relative utility function* as:

$$P_{ijL} = \frac{F_L(V_{ijL})}{\sum_{L=1}^{L} F_L(V_{ijL})} = \frac{I_{ijL}}{\sum_{L=1}^{L} I_{ijl}} \tag{29.9}$$

where the terms are the same as in 29.4 except the function, F_L, is some function that is specific to the travel mode, *L*. The numerator is defined as the impedance of mode *L* in traveling from origin zone i to destination zone j while the denominator is the sum of all impedances.

Notice that the ratio of the cost function for one mode relative to the total costs is also the ratio of the impedance for mode *L* relative the total impedance. The total impedance was defined in Chapter 28 as the disincentive to travel as a function of separation (distance, travel time, cost). We see that the share of a particular mode, therefore, is the proportion of the total impedance of that mode. This share will vary, of course, with the degree of separation. For any given separation, there will usually be a different share for each mode. For example, at low separation between zones (e.g., zones that are next to each other), walking and biking are much more attractive than taking a bus or a train and, perhaps even driving. At greater separation (e.g., zones that are 5 miles apart), walking and biking are almost irrelevant choices and the likelihood of driving or using public transit is much greater. In other words, the share that any one mode occupies is not constant, but varies with the impedance function.

---

[2] In a survey of the travel behavior of homeless persons, it was noted that most homeless walked very short distances over the day even though the value of their time was very low. For longer trips, they still tended to take the bus rather than walk. Survey on the travel behavior of very low income individuals. Urban Planning Program, University of California at Los Angeles, 1987 (with Martin Wachs).

Why then cannot the mode split be estimated directly at the trip distribution stage? If the trip distribution function is:

$$T_{ij} = \alpha P_i^{\lambda} \beta \frac{A_j^{\tau}}{I_{ij}} \qquad \text{(28.12 repeat)}$$

and if these trips, in turn, are split into distinct modes using equation 29.9, could not 28.12 be re-written as:

$$T_{ijL} = \alpha P_i^{\lambda} \beta \frac{A_j^{\tau}}{I_{ijL}} \qquad (29.10)$$

where T_{ijL} is the number of trips traveling from origin zone i to destination zone j by mode L, P_i is the production capacity of zone i, A_j is the attraction of zone j, α and β are constants that are applied to the productions and attractions respectively, λ and τ are 'fine tuning' exponents of the productions and attractions respectively, and I_{ijL}, is the impedance of using mode L to travel between the two zones? The answer is, yes, it could be calculated directly. If I_{ijL} was a perfectly defined impedance function (with no error), then the mode share could be calculated directly at the distribution stage instead of separating the calculations into two distinct stages. The problem, however, is that the impedance functions are never perfect (far from it, in fact) and that re-scaling is required both to get the origins and destinations balanced in the trip distribution stage and to ensure that the probabilities in equation 29.10 add to 1.0. The effect of these adjustments generally throws off a model such as equation 29.10. Consequently, the trip distribution and mode split stages are usually calculated as separate operations.

Measuring Travel Costs

The next question is what types of travel costs are there that define impedance? As mentioned above, there are real as well as perceived costs that affect a travel mode decision. Some of these can be measured easily, while others are very difficult requiring detailed surveys of individuals. Among these costs are:

1. Distance or travel time. As mentioned throughout this discussion, distance is only a rough indicator of cost since it is invariant with respect to time. Actual travel time is a much better indicator because it varies throughout the day and can be easily converted into a *travel time value*, for example by multiplying by an average unit wage.

2. Other real costs, such as the operating costs of a private vehicle (fuel, oil, maintenance), parking, and insurance. Some of these can be subsumed under travel time value by working out an hourly price for travel.

3. Perceived costs, such as convenience, fear of being caught by an offender, ease of escape from a crime scene, difficulties in moving stolen goods, and fear of retaliation by other offenders or gangs).

Some of these costs can be measured and some cannot. For example, the value of travel time can be inferred from the median household income of a zone for aggregate analysis or from the actual household income for individual-level analysis. Parking can be averaged by zone. Insurance costs can be estimated from zone averages *if* the data can be obtained.

Many perceived costs also can be measured. Convenience, for example, could be measured from a general survey. The fear of being caught can be inferred from the amount of surveillance in a zone (e.g., the number of police personnel, security guards, security cameras). Even though it may be a difficult enumeration process, it is still possible to measure these costs and come up with some average estimate.

Other perceived costs, on the other hand, may not be easily measured. For example, the fear an offender belonging to one gang has about retaliation from another gang is not easily measured. If one could map the 'territory' of a gang, then one could members of one territory would not commit a crime in another territory (Bernasco & Block, 2009). Similarly, the cost of moving stolen goods by a thief is not easily measured; one would need to know the location of the distributors of these goods.

In practice, travel modelers make simple assumptions about costs because of the difficulty in measuring many of them. For example, travel time is taken as a proxy for all the operating costs. Parking costs can be incorporated through simple assumptions about the distribution across zones (e.g., zones within the central business district - CBD, are given an average high parking costs; zones that are central, but not in the CBD, are assigned moderate parking costs; zones that are suburban are assigned low parking costs). It would be just too time consuming to document each and every cost affecting travel behavior, particularly if we are developing a model of offender travel.

Nevertheless, theoretically, these are all potentially measurable costs. They are real and probably have an impact in the travel decisions that offenders make. As researchers and analysts, we have to work towards articulating as many of these costs as possible in order to produce a realistic representation of offender travel.

Aggregate and Individual Utility Functions

One of the big debates in travel modeling is whether to use aggregate or individual utility functions to calculate mode share. The aggregate approach measures common costs for each

zone, assuming an average value. The disaggregate approach (sometimes called 'second generation' models) measures unique costs for individuals, then sums upward to yield values for each zone pair. Even though the end result is an allocation of costs to each zone pair, the articulation of unique costs at the individual level can, in theory, allow a more realistic assessment of the utility function that is applied to a region.

The aggregate approach will measure costs by averages. Thus, a typical equation for driving mode might be:

$$Total\ cost_{ij} = \beta_1 T_{ijk} + \beta_2 P_{ijk} \quad (29.11)$$

where T_{ij} is the average travel time between two zones, i and j, and P_{ij} is the average parking cost for parking in zone j. Notice that there are a limited number of variables in an aggregate model (in this case, only two) and that the assigned average is for an entire zone. Notice also that the parking cost is applied only to the destination zone. It is assumed that any traveler will pay that fee in that zone irrespective of which origin zone he/she came from.

A disaggregate approach can allow more cost components, if they are measured. Thus, a typical equation for driving mode might be:

$$Total\ cost_{ijk} = \beta_1 T_{ijk} + \beta_2 P_{ijk} + \beta_3 C_{ijk} + \beta_4 M_{ijk} + \beta_5 S_{ijk} \quad (29.12)$$

where T_{ijk} is the travel time for individual k between two zones, i and j, P_{ij} is the average parking cost for parking in zone j, C_{ijk} is the convenience of traveling to zone j from zone i for individual k, M_{ijk} is the comfort and privacy experienced by individual k in traveling from zone i to zone j, and S_{ijk} is the perceived safety experienced by individual k in traveling from zone i to zone j. Notice that there are more cost variables in the equation and that the model is targeted specifically to the individual, k. Two individuals who live next door to each other and who travel to the same destination may evaluate these components differently. If these individuals have substantially different incomes, then the value of the travel time will differ. If one values privacy enormously while the other does not, then the cost of driving for the first is less than for the second. Similarly, convenience is affected by both travel time and the ease of getting in and out of vehicle. Finally, the perception of safety may differ for these two hypothetical individuals. There are many studies that have documented the significant role played by safety in affecting, particularly, transit trips (Levine & Wachs, 1986).

In other words, the aggregate approach applies a very elementary type of utility function whereas the disaggregate approach allows much more complexity and individual variability.

Of course, one has to be able to measure the individual cost components, a difficult task under most circumstances.

There is also a question about which approach is more accurate for correctly forecasting actual mode splits. Historically, most Metropolitan Planning Organizations have used the aggregate method because it's easier. However, more recent research (McFadden, 2002; Ben-Akiva & Lerman, 1985; Domincich & McFadden, 1975) has suggested that disaggregate modeling may be more accurate. At the very minimum, the disaggregate model is more amenable to policy interpretations because it is more behavioral. *If* one could interview travelers with a survey, then it is possible to explore the variety of cost factors that affect a decision on both destination and mode split, and a more realistic (if not unique) utility function derived.

But, as mentioned above, with crime trips, this is very difficult, if not impossible, to do. Consequently, for the time being, we are stuck with an aggregate approach towards modeling the utility of travel by offenders.

Tools for Estimating Mode Split in *CrimeStat*

CrimeStat has two sets of tools for estimating the mode split model. First, if individual level data on travel modes can be obtained, then the multinomial logit model in the Discrete Choice module can be used (see Chapters 21 and 22). That is, if data on actual mode choices taken by offenders could be obtained with characteristics of both the offenders and the zones in which they lived or committed crimes in being associated with those choices, then the preferred method would be to use the multinomial logit to model the predictors of mode choice.

Second, if individual level data on travel modes is not available (the usual circumstances in most police department), then an approximation to a utility function can be made. The approach, in this case, is to estimate a *relative accessibility* function and then apply that function to the predicted trip distribution. The relative accessibility function is a mathematical approximation to a utility function, rather than a measured utility function by itself. Because the cost components cannot be measured, at least for offenders, we use an inductive approach. Reasonable assumptions are made and mathematical functions are found that fit these assumptions.

Relative Accessibility

The relative accessibility approach produces a plausible model, not an analytical one. The plausibility comes by making reasonable assumptions about actual travel behavior. One can assume that walking trips will occur for short trips, say under two miles. Bicycle trips, on the

other hand, could occur over longer distances, but will still be relatively short (also, there is always the risk of traffic on the safety of bicycle trips). Transit trips (bus and train) will be used for moderately long distances but require an actual transit network. Finally, driving trips are the most flexible because they can occur over any size distance and road network. They are less likely to be used for very short trips, on the other hand, due to reasons discussed above.

Hierarchical Approach to Estimating Mode Accessibility

Using this approach, specific steps can be defined to produce a plausible accessibility model. The Mode Split tab in *CrimeStat* allows the estimation of a relative accessibility model. Figure 29.1 shows the setup page for the mode split module. There are four tabs in the mode split module:

1. The setup tab
2. Calibrate mode split: I
3. Calibrate mode split: II
4. Calibrate mode split: III

The setup tab defines the three files that are used in the module. There is predicted origin file, a predicted destination file, and a predicted origin-destination trip file. Both the predicted origin file and the predicted destination file have to be defined as the Primary file or Secondary file and ID fields have to be defined for each. The predicted origin-destination trip file is input separately on this tab. The variable identifying the predicted number of trips must be defined. Figure 29.1 illustrates this one data set.

The next three tabs define up to five separate travel modes. The first Calibrate mode split tab defines modes 1 and 2. The second tab defines modes 3 and 4 and the third defines mode 5. The user can assign any one mode to each of these available slots. For example, mode 1 could be walking; mode 2 could be driving; mode 3 could be bus; mode 4 could be train; and mode 5 could be bicycle. There is no particular order to the assignment and not all five available modes have to be used. Figure 29.2 illustrates the defining of two modes on the 'Calibrate mode split: I' tab where mode 1 is walking and mode 2 is by bicycle.

For each of the modes that are used, the user must define an impedance function. The impedance function is a mathematical function that approximates the discounting of that mode with distance. The user can use a pre-defined mathematical function or else estimate it using an already-calibrated impedance function (see Chapter 28, page 28.24). In the example in Figure

Figure 29.1:

Mode Split Module

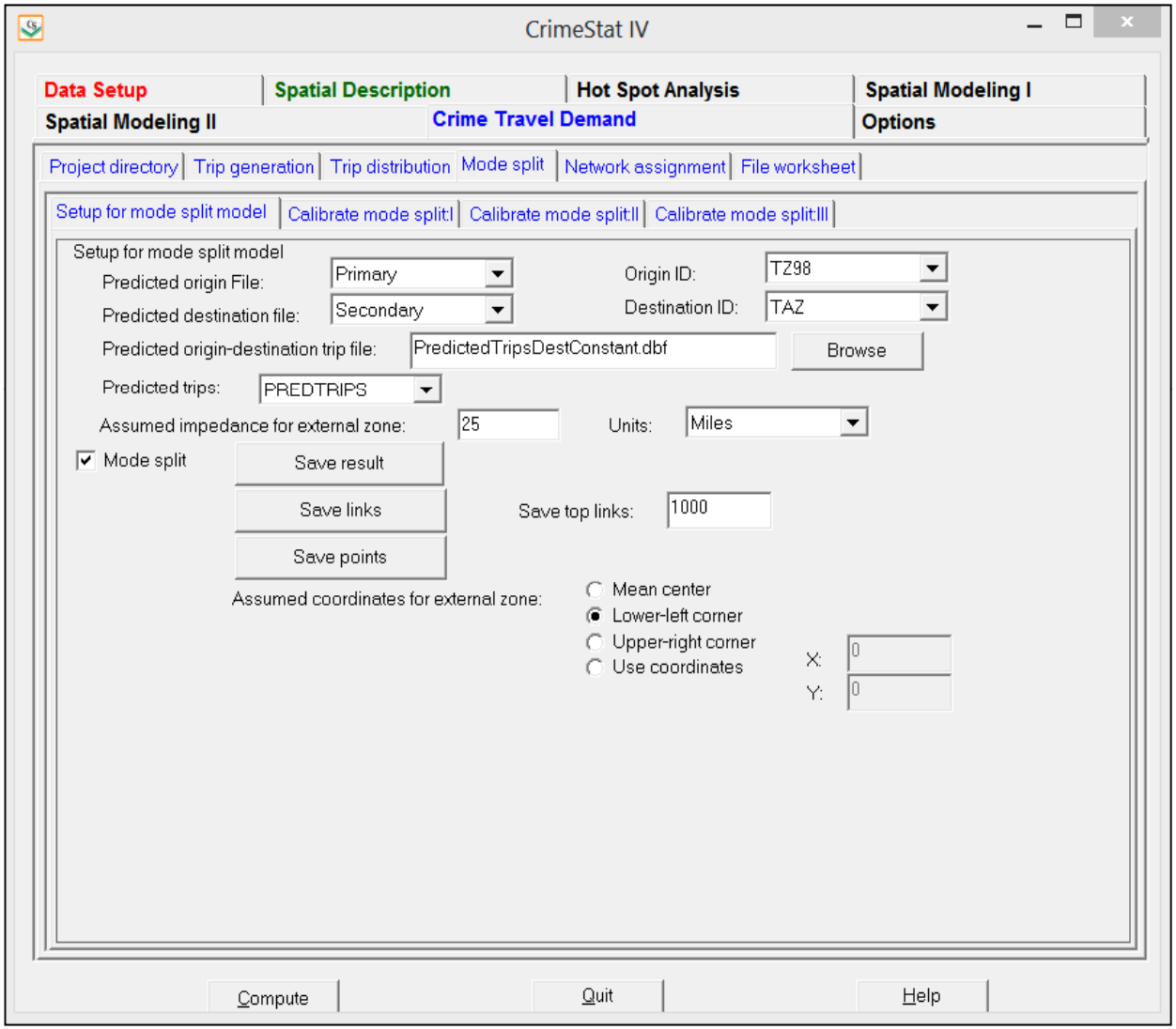
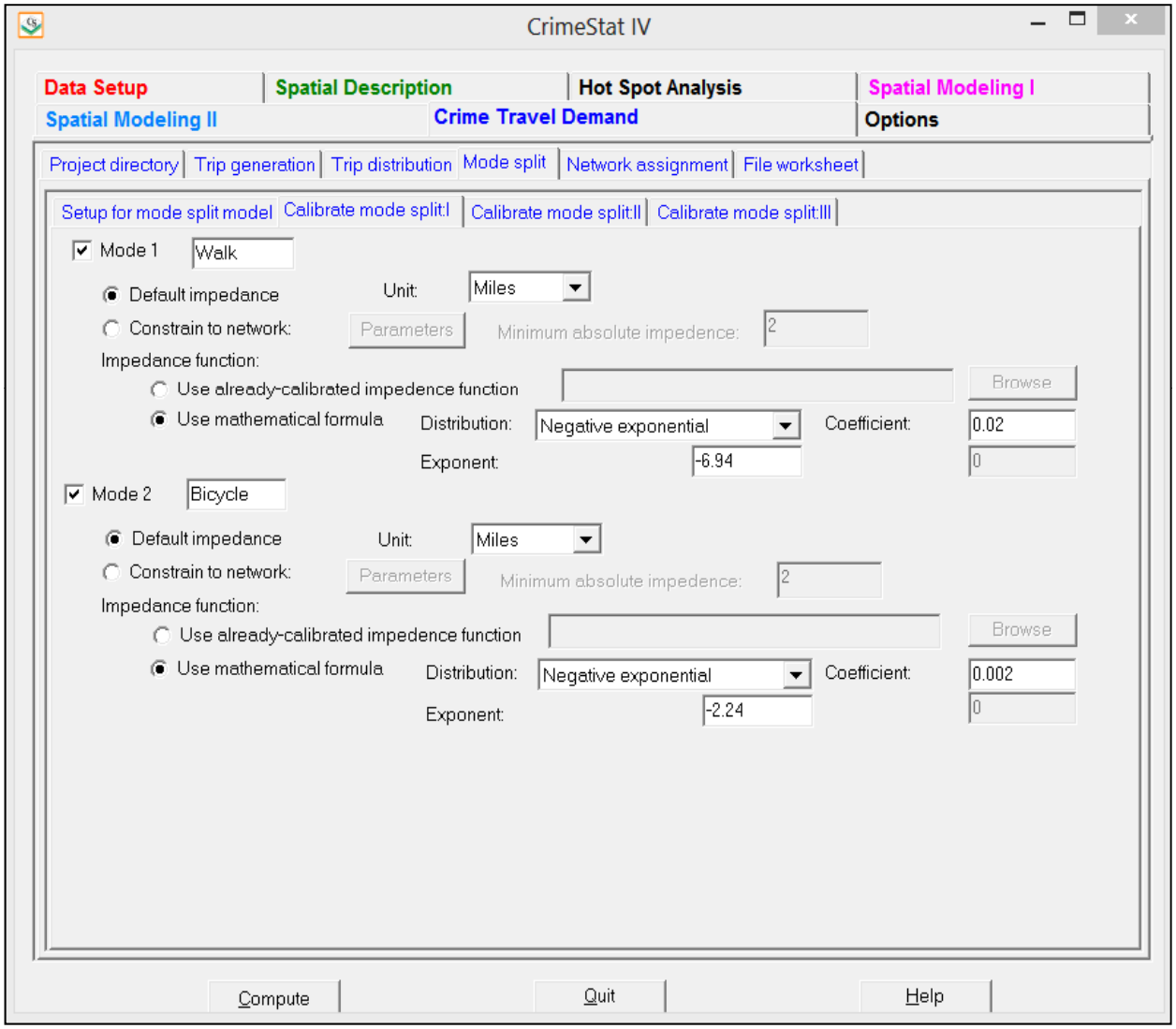


Figure 29.2:

The 'Calibrate Model Split: I' Tab



29.2, a mathematical function is being used for both the walking mode (mode 1) and the bicycle mode (mode 2). For the walking mode, the following negative exponential function is used:

$$Y_{ijL} = 0.02e^{-6.94} \quad (29.13)$$

while bicycle mode uses a negative exponential function of the following form:

$$Y_{ijL} = 0.002e^{-2.24} \quad (29.14)$$

Note that these are not probabilities but frequencies. That is, equation 29.13 estimates the number of walking trips as a function of distance between two zones, i and j. Each equation discounts the likelihood of using that mode as a function of distance between the zones. The probabilities are estimated later when the frequency of all modes have been defined.

Spreadsheet for Estimating Mode Split Impedance Values

To identify the parameters that produce a plausible model of mode frequency, an Excel spreadsheet has been developed for making these calculations (*Mode Split Impedance Defaults.xls*). It is part of the "Crime Travel Demand Sample Data.zip" file and can be downloaded from the *CrimeStat* download page. Figure 29.3 shows part of the spreadsheet.

The spreadsheet has been defined with distance, but it can be adapted for travel time or travel cost as well. A spreadsheet has been used because it is more flexible than incorporating it as a routine in *CrimeStat* to estimate the parameters. There is not a single solution to the parameters estimates and the different choices can be seen more easily.

Define travel modes

The following provides instructions on estimating the impedance values with the spreadsheet. In the *CrimeStat* mode split routine, up to five different modes are allowed. First, the user should define the travel modes to be modeled. In the spreadsheet, these have default names of "Walk", "Bike", "Drive", "Bus", and "Train". The user is not required to use these names nor all five modes. Clearly, if there is not a train system in the study area, then the "Train" mode does not apply. Travel modelers use variations on these, such as "drive alone","carpool", "automobile", "motorcycle", and so forth.

Figure 29.3:

Estimating Mode Split Impedance Values

Estimating Mode Split Impedance Defaults

Instructions

1. Define target mode split proportions
2. Identify mathematical model for each mode
3. Define order in which mathematical functions are fit
4. Iteratively estimate parameters over distance range, one model at a time
5. Examine mode graphs to check model (below)
6. Examine accessibility graph for final review

Step 1:
Define target mode split proportions:

	Mode	Proportions
1	Drive	0.90
2	Bus	0.04
3	Train	0.01
4	Walk	0.04
5	Bike	0.01
	Total	**1.00**

Step 2:
Identify mathematical model for each mode

		Walk Mode	(target=.04)	
	Intercept	Slope		
Linear	0.0366	-0.0665		
	Mean	SD	Coefficient	
Normal	0.5	0.0404	0.00405	
	Mean	SD	Coefficient	
Lognormal	0.5	0.25	0.0735	
	Coefficient	Exponent		
Negative Exponential	0.0366	-4.95		
	Peak likelihood	Peak distance	Exponent	
Truncated Negative Exponential	0.0366	0.5	-4.95	

		Bike Mode	(target=.01)	
	Intercept	Slope		
Linear	0.00211	-0.0005		
	Mean	SD	Coefficient	
Normal	2	1	0.00505	
	Mean	SD	Coefficient	
Lognormal	2	1	0.024	
	Coefficient	Exponent		
Negative Exponential	0.0033	-0.8		
	Peak likelihood	Peak distance	Exponent	
Truncated Negative Exponential	0.00284	1	-0.8	

Step 3:
Define order in which mathematical functions are fit

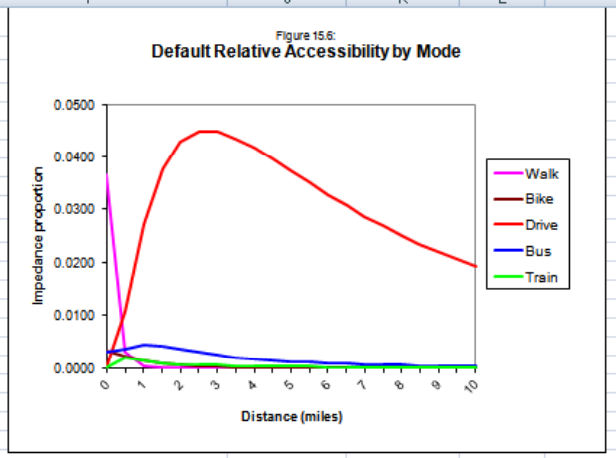
Order	Function
1. Walk	Negative exponential
2. Bike	Negative exponential
3. Drive	Lognormal
4. Bus	Lognormal
5. Train	Lognormal

Step 4:
Iteratively estimate parameters over distance range, one model at a time

Sum of test =	0.0400	0.0400	0.0400	0.0400	0.0400	
Mean distance (approx) =	0.0419	0.5000	1.3022	0.0459	0.5459	
Max =	0.0366	0.0400	0.0241	0.0366	0.0366	
Distance	**Linear**	**Normal**	**Lognormal**	**Negative Exponential**	**Truncated Negative Exponential**	
0	0.0366	0.0000	0.0000	0.0366	0.0000	

Sum of test =	0.0100	0.0100	0.0100	
Mean distance (approx) =	1.2492	2.0307	2.4002	
Max =	0.0021	0.0020	0.0021	
Distance	**Linear**	**Normal**	**Lognorma**	**Negative**
0	0.0021	0.0003	0.0000	

Step 6:
Accessibility graph



Define target proportions

Second, define the *target proportions*. These are the expected proportions of travel for each mode. Where would such proportions come from? There have been many studies of driving and transit behavior, but relatively few studies of bicycle and pedestrian use (Schwartz et al, 1999; Porter, Suhrbier & Schwartz, 1999; Turner, Shunk, & Hottenstein, 1998). There are not simple tables that one can look up default values.

Other studies

To solve this problem, examples were sought from different size metropolitan areas. Estimates of travel mode share for all trip purposes (work and non-work) were obtained from Ottawa (McCormick Rankin, 2011; Ottawa, 2008), Portland (Portland, 1998); and Houston[3]. Table 29.1 shows the estimated shares. The Houston data does not include walking and biking shares, and transit trips are not distinguished by mode in the Portland and Ottawa data.

Table 29.1:
Estimated Mode Share for Three Metropolitan Areas
All Trip Purposes

	Ottawa	**Portland**	**Houston**
Population:	725 thousand (1995)	2.0 million (2001)	4.6 million (2000)
Percent of trips by:	(1995)	(1994)	(2025 forecast)
Driving	73.5%	88.6%	98.3%
Transit	15.2%	3.0	1.7% (bus 1.1%; rail 0.6%)
Walking	9.6%	4.6%	-
Bicycle	1.7%	1.0%	-
Other	-	2.8%	-

While it is difficult to generalize, walking is very much dependent on both the compactness of the city and the existence of an extensive transit system. In Houston, the transit system is primarily a commuter system whereas in Portland and Ottawa, it serves multiple purposes. Clearly, the more compact is the urban area, the more likely that trips will occur by transit, walking or biking. But, even in the case of Ottawa where almost 10% of trips are by

[3] Houston-Galveston Area Council. Personal communication. 2004.

walking, the majority of trips are by private vehicle. In the United States and Canada, for metropolitan areas with extensive transit facilities (New York, Chicago, Boston, Montreal), a majority of regional trips are still by automobile.

Based on this, some default values were selected and put into the spreadsheet. The spreadsheet requires that they are entered as proportions (not percentages). The defaults values were (Table 29.2):

The user can modify these in the spreadsheet. It is important that a user contact the local Metropolitan Planning Organization to find out what would be reasonable values for the urban area. The default values are guesses based on a limited amount of data.

Table 29.2:
Default Mode Share Values
Proportions

Mode	Share
Driving	.90
Bus	.04
Train	.01
Walk	.04
Bicycle	.01

Journey to work census

An alternative approach is to use the Journey to Work data of the U.S. Census Bureau (2004). During every census, the Census Bureau documents home-to-work 'commute' trips and breaks down these data by mode share. They release these data under the title "Journey to Work". The 2010 Journey to Work data set has not been released. However, in 2000 in the United States, 87.9% of all home-to-work trips were by private vehicle (automobile, van, truck), 4.7% were by public transit (bus 2.5%; rail 2.1%; other 0.1%), 2.9% were by walking, 0.4% were by bicycle, 0.1% were by motorcycle, 0.7% were by other means, and 3.3% worked at home.

National journey to work statistics for 1990 and 2000 and for metropolitan areas in 1990 can be found at U.S. Census, 2009). Data on metropolitan areas for 2000 can be found in McGuckin and Srinivasan (2003). In 2000, the home-to-work mode share for a sample of large metropolitan (including the 15 largest) areas is shown in Table 29.3. They are rank-ordered by the 2000 population of the metropolitan area.

As can be seen, the larger metropolitan areas generally have a higher share of transit use and walking than smaller metropolitan areas, but the differences are not that dramatic. Further,

Table 29.3:

Mode Share of Journey to Work Trips: 2000

(From McGuckin & Srinivasan, 2003)

Greater Metropolitan Area	2000 Pop (M)	Mode Share					
		Walk	Bicycle	Drive	Bus	Rail	Other*
New York	21.1	5.6%	0.3%	65.7%	6.8%	17.1%	4.5%
Los Angeles	16.4	2.6%	0.6%	87.6%	4.3%	0.3%	4.6%
Chicago	9.2	3.1%	0.3%	81.5%	4.6%	6.6%	3.9%
Washington DC	7.6	3.0%	0.3%	83.2%	4.1%	5.0%	4.4%
San Francisco	7.0	3.3%	1.1%	81.0%	5.7%	3.5%	5.4%
Philadelphia	6.2	3.9%	0.3%	83.6%	5.3%	3.3%	3.6%
Detroit	5.5	1.8%	0.2%	93.4%	1.7%	0.0%	2.9%
Boston	5.8	4.1%	0.4%	82.7%	3.2%	5.5%	4.1%
Dallas	5.2	1.5%	0.1%	92.7%	1.6%	0.1%	4.0%
Houston	4.7	1.6%	0.3%	91.3%	3.1%	0.0%	3.7%
Atlanta	4.1	1.3%	0.1%	90.6%	2.4%	1.1%	4.5%
Miami	3.9	1.8%	0.5%	90.1%	3.2%	0.5%	3.9%
Seattle	3.6	3.2%	0.6%	84.4%	6.2%	0.0%	5.6%
Phoenix	3.3	2.1%	0.9%	90.0%	1.9%	0.0%	5.1%
Minneapolis/ St Paul	3.0	2.4%	0.4%	88.4%	4.4%	0.0%	4.4%
Cleveland	2.9	2.1%	0.2%	91.1%	3.1%	0.3%	3.2%
San Diego	2.8	3.4%	0.6%	86.9%	3.1%	0.2%	5.8%
St Louis	2.6	1.6%	0.1%	92.5%	2.1%	0.2%	3.5%
Denver	2.6	2.4%	0.7%	87.1%	4.2%	0.1%	5.5%
Pittsburgh	2.4	3.6%	0.1%	87.1%	6.0%	0.1%	3.1%
Portland	2.3	3.0%	0.8%	85.2%	5.1%	0.5%	5.4%
Cincinnati	2.0	2.3%	0.1%	91.4%	2.8%	0.0%	3.4%
Sacramento	1.8	2.2%	1.4%	88.9%	2.4%	0.3%	4.8%
Kansas City	1.8	1.4%	0.1%	93.2%	1.2%	0.0%	4.1%
Milwaukee	1.7	2.8%	0.2%	90.0%	3.9%	0.0%	3.1%
Indianapolis	1.6	1.7%	0.2%	93.3%	1.2%	0.0%	3.6%
Orlando	1.6	1.3%	0.4%	92.7%	1.6%	0.0%	4.0%
San Antonio	1.6	2.4%	0.1%	90.9%	2.8%	0.0%	3.8%
Norfolk	1.6	2.7%	0.3%	91.0%	1.7%	0.0%	4.3%
Las Vegas	1.6	2.4%	0.5%	89.5%	3.9%	0.0%	3.7%
Charlotte	1.5	1.2%	0.1%	93.8%	1.3%	0.0%	3.6%

Table 29.3: (continued)

Greater Metropolitan Area	2000 Pop (M)	Mode Share					
		Walk	Bicycle	Drive	Bus	Rail	Other*
New Orleans	1.3	2.7%	0.6%	87.7%	5.2%	0.0%	3.8%
Salt Lake City	1.3	1.8%	0.4%	90.3%	2.7%	0.3%	4.5%
Memphis	1.1	1.3%	0.1%	93.9%	1.6%	0.0%	3.1%
Rochester	1.1	3.5%	0.2%	90.9%	1.9%	0.0%	3.5%
Oklahoma City	1.1	1.7%	0.2%	93.8%	0.5%	0.0%	3.8%
Louisville	1.0	1.7%	0.2%	92.9%	2.2%	0.0%	3.0%

* Includes taxi, ferry, and working at home

for even the largest metropolitan areas, the majority of the home-to-work trips are by private vehicle.

The problem with these data, however, is that they only examine work trips. Nationally, home-to-work trips represent only about 15% of all daily trips (BTS, 2002). On the other hand, 45% of daily trips are for shopping and errands and 27% are social and recreational. Further, non-work trips are even more likely to occur by automobile, and are generally shorter. For example, in Houston, for home-based non-work trips, only 1% of trips were by transit compared to 3.1% for home-to-work trips in 2004. These home-based non-work trips may be a better analogy to crime trips than work trips since they tend to be of similar trips lengths as crime trips.

Thus, unless the user is willing to assume that a crime trip is like a work trip (which is questionable), then the Journey to Work tables are probably not the best guide for the target proportions. Nevertheless, an examination of them is valuable to see how work trips are split among the various travel modes.

Select mode functions

Third, select mathematical functions that approximate accessibility utility. Again, some plausible assumptions need to be made. In *CrimeStat*, the user can select among five different mathematical functions (linear, negative exponential, normal, lognormal, truncated negative exponential). The default functions are shown in Table 29.4 below.

Table 29.4:
Default Mode Share Functions

Mode	Function
Walk	Negative exponential
Bicycle	Negative exponential
Driving	Lognormal
Bus	Lognormal
Train	Lognormal

The reasoning behind this is that walking and biking are relatively short trips whereas transit modes involve intermediate length trips. Finally, driving can be used for any length trip other than very short trips (e.g., less than one or two blocks). Thus, it is unlikely that an automobile will be used for very short trips (less than a quarter mile) and it is very unlikely that transit will be used for short trips (less than a half mile or more). Nevertheless, the user can modify these choices and examine the appropriate column in the spreadsheet.

Select model priorities

Fourth, select the priorities for modeling the target. Unfortunately, there may not be a single solution that will yield the target proportions. Therefore, a decision needs to be made on which **order** the spreadsheet will be calculated. The default order is shown Table 29.5.

Table 29.5:
Default Mode Share Functions

Mode	Order of Iteration
Walk	1
Bicycle	2
Driving	3
Bus	4
Train	5

The reasoning is that the offender first makes a decision on the length of the trip (short, medium, long, or the equivalent in travel time). Then, within each category, the offender makes a decision on which mode to choose. For very short trips, the default mode is walking. For intermediate to long trips, the default choice is driving. However, the user can change this order.

Iteratively estimate parameters

Fifth, in the spreadsheet, iteratively adjust the parameters until the target proportion is reached. Do this in the order selected in the above step. Again, there is not a single solution that will produce the target proportion. For example, each of the mathematical functions has two or three parameters that can be adjusted:

1. For the negative exponential, the coefficient and exponent
2. For the normal distribution, the mean distance, standard deviation and coefficient
3. For lognormal distribution, the mean distance, standard deviation and coefficient
4. For the linear distribution, an intercept and slope
5. For the truncated negative exponential, a peak distance, peak likelihood, intercept, and exponent.

The target proportion can be achieved by adjusting any or all of the parameters. For example, to achieve a target proportion of 0.05 (i.e., 5%) using the negative exponential, an infinite number of models can yield this, for example coefficient=0.0366, exponent=-2.63; coefficient=0.0459 or exponent=-5; coefficient=0.01966, exponent=-1; and so forth. Therefore, there must be additional criteria to constrain the choices.

One criterion is to set an approximate mean distance. For example, with walking trips, the mean distance can be set to a half mile or for driving, the mean distance can be set to 6 miles. Then, check the approximate mean distance of the selected function. Though rarely will the exact mean distance be replicated, the calculated mean distance should be close to the ideal. The one exception is for very short trips. Since the intervals in the spreadsheet are a half mile each, there is considerable error for very short distances.

Examine the graphs in the spreadsheet

Another diagnostic tool is to examine the graph of the function in the spreadsheet (below the calculations). Does the typical trip approximate the expected mean distance? Does the selected function produce something that looks intuitive? Admittedly, these are subjective decisions. But, if the function looks strange, it can be caught and re-calculated.

In short, the aim should be to produce a function that not only captures the target proportion, but looks plausible. Several examples are shown below. Figure 29.4 shows the default walking model using a negative exponential. Figure 29.5 shows the default biking model, also using a negative exponential. Figure 29.6 shows the default driving mode using a lognormal function. Figure 29.7 shows the default bus mode, also using a lognormal function, and Figure 29.8 shows the default train mode using a lognormal function.

Figure 29.4:

Negative Exponential Function: Walk Mode

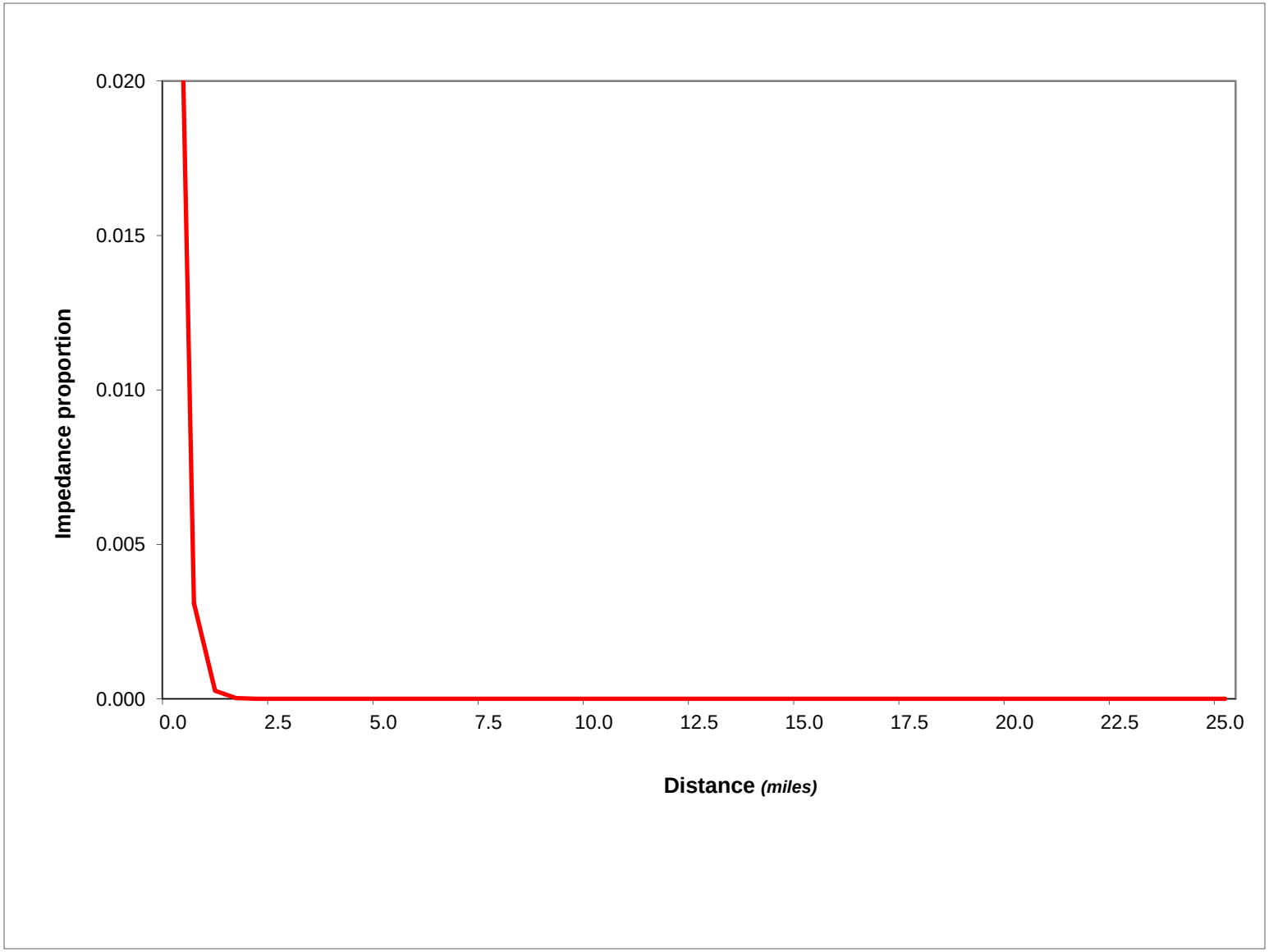


Figure 29.5:

Negative Exponential Function: Bike Mode

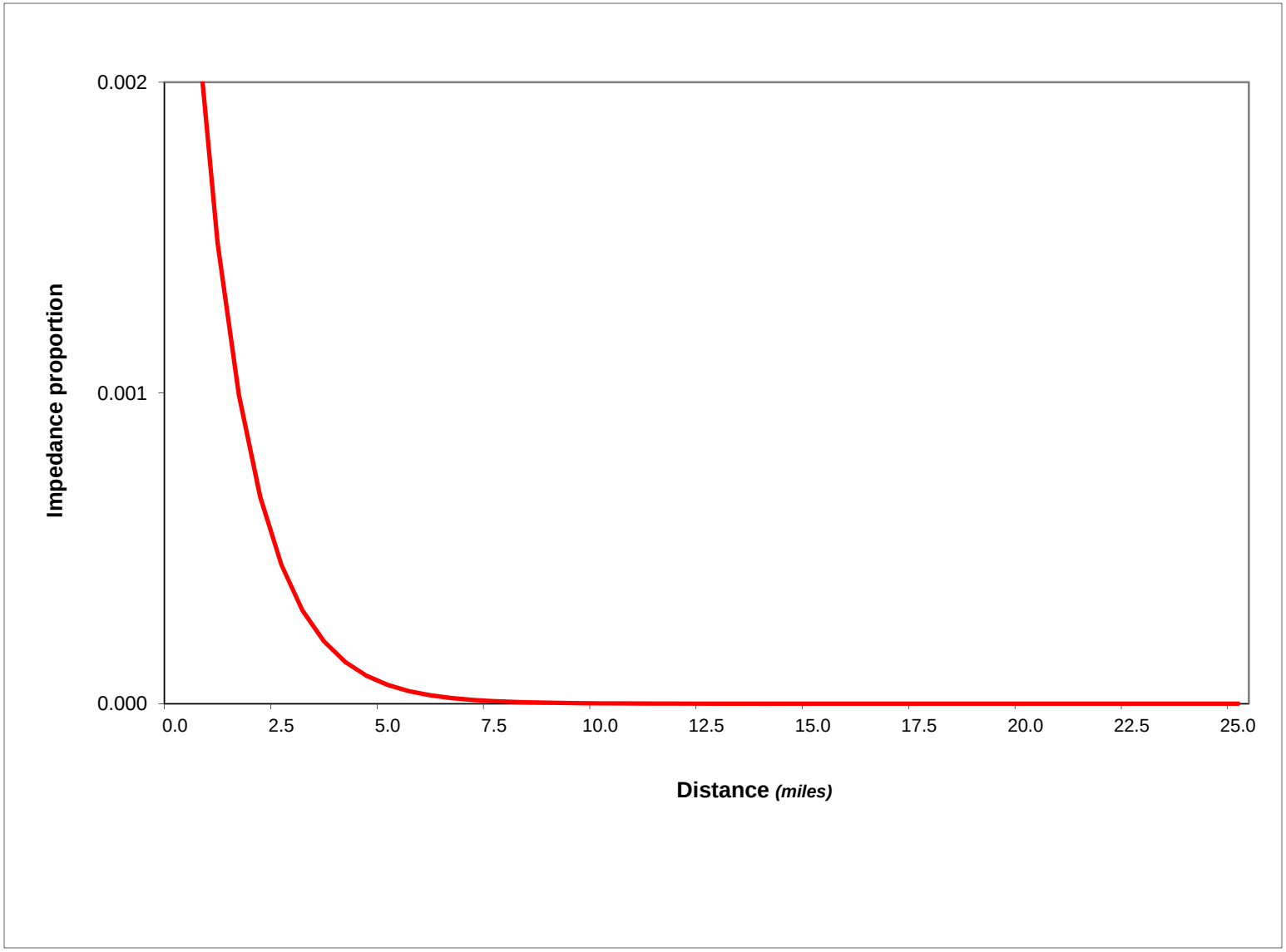


Figure 29.6:

Lognormal Function: Drive Mode

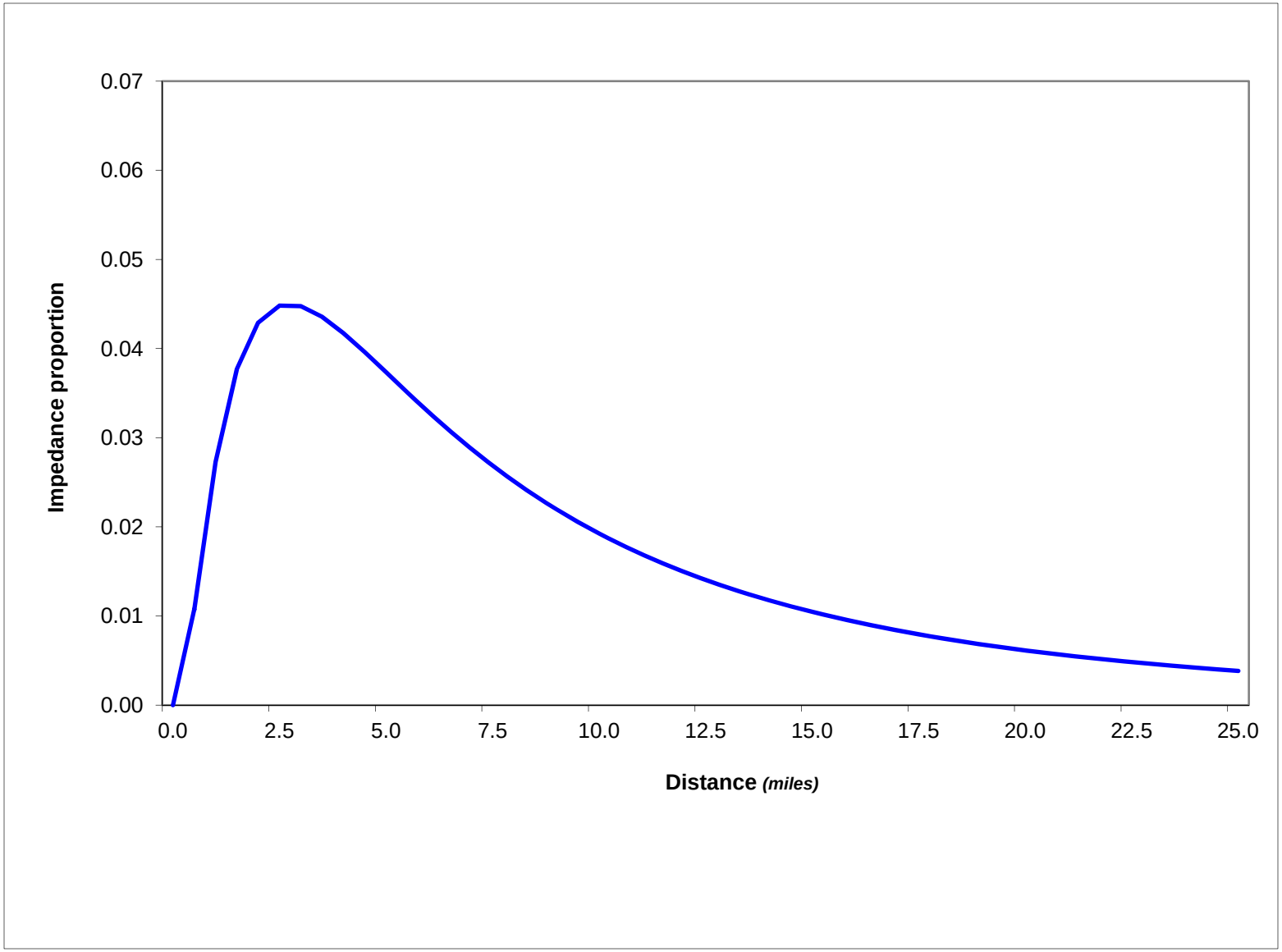


Figure 29.7:

Lognormal Function: Bus Mode

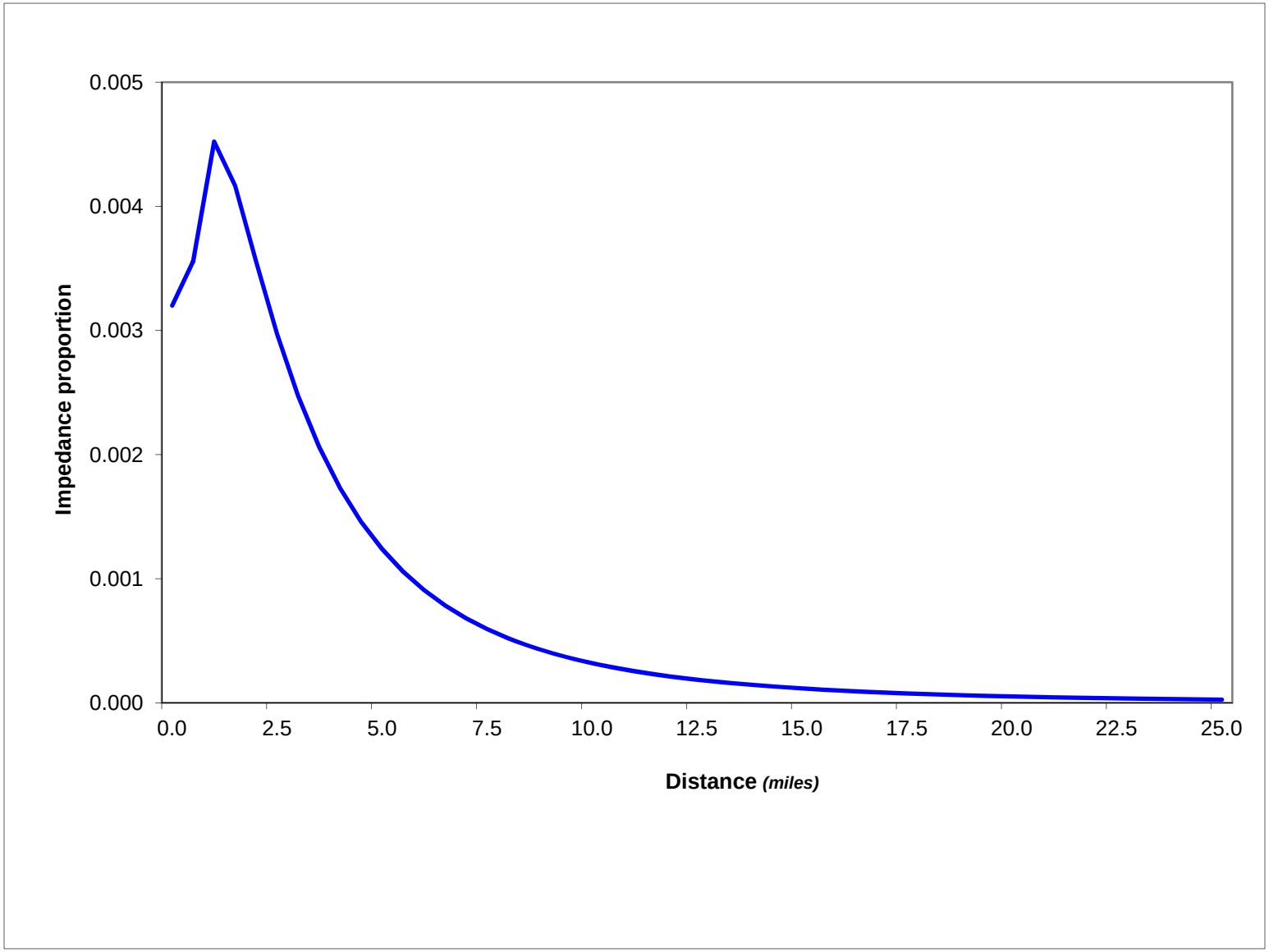


Figure 29.8:

Truncated Negative Exponential Function: Train Mode

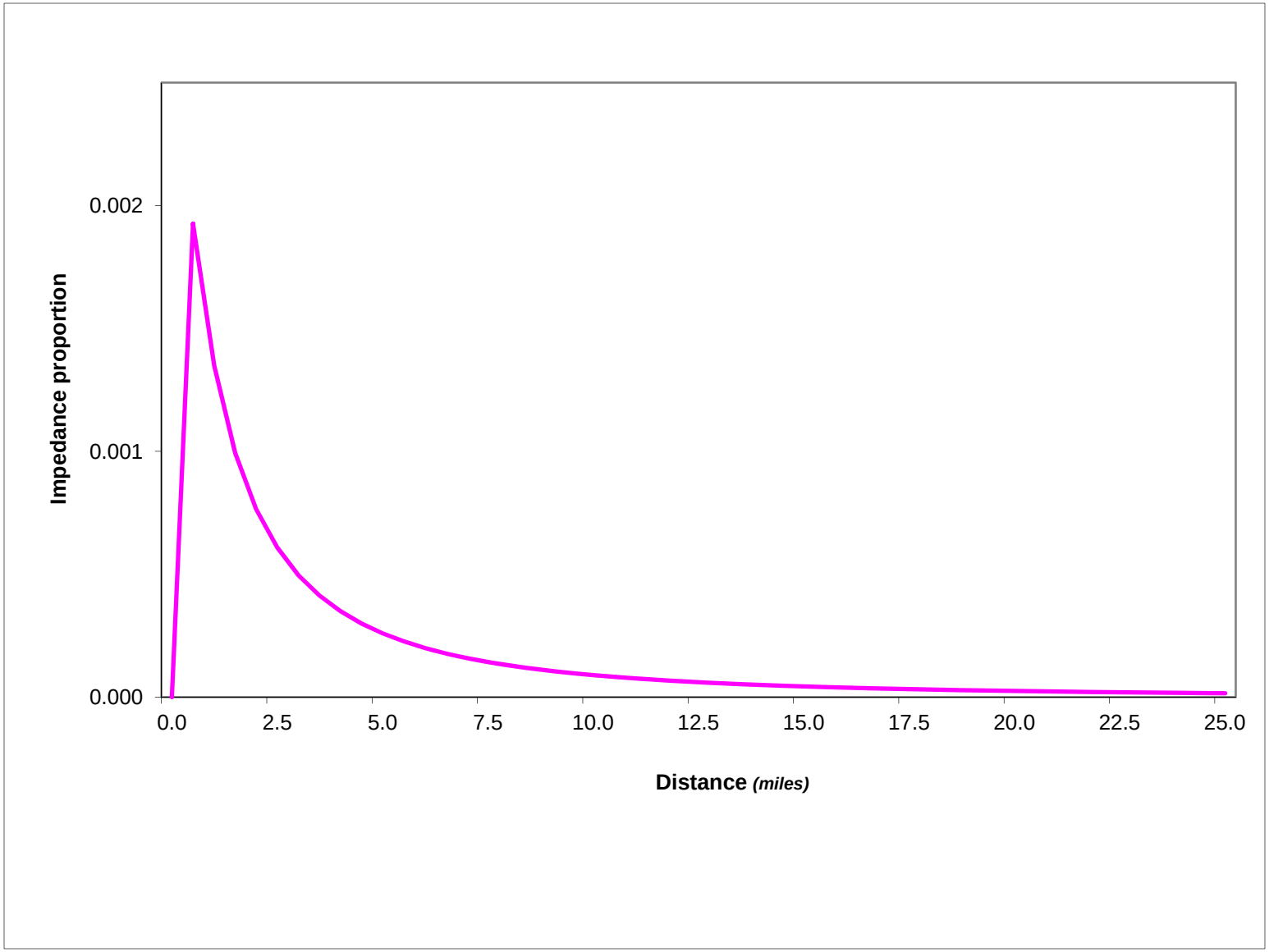


Figure 29.9 shows the cumulative results of the default values. This is also graphed in the spreadsheet, starting in cell I1. Notice how the relative accessibility function works. As distance increases, the mode proportions change. At very short distances, walking trips predominate with biking trips also getting a moderate share. As the distance increases, the proportions increasingly shift toward driving. Even though the likelihood of driving declines with distance, the other modes decline even faster. In other words, the relative accessibility function is estimating the relative shares of each mode as a function of the impedance (in this case, distance). Note also the relative differences in the frequency of trips. Driving trips are far more frequent than any other mode. Thus, compared to the individual graphs (figure 29.4-29.8), the other modes are more muted than driving.

Adapting spreadsheet for travel time or travel cost

The illustrations to this point have used distance as an impedance unit. However, other impedance units, such as travel time and generalized travel cost, can also be used. These generally require a network (see below) in that weights have to be assigned to segments. Nevertheless, the same logic applies. For each travel mode, a specific impedance function is estimated and then applied to the trip distribution matrix.

Empirically Estimating the Mode-specific Impedance

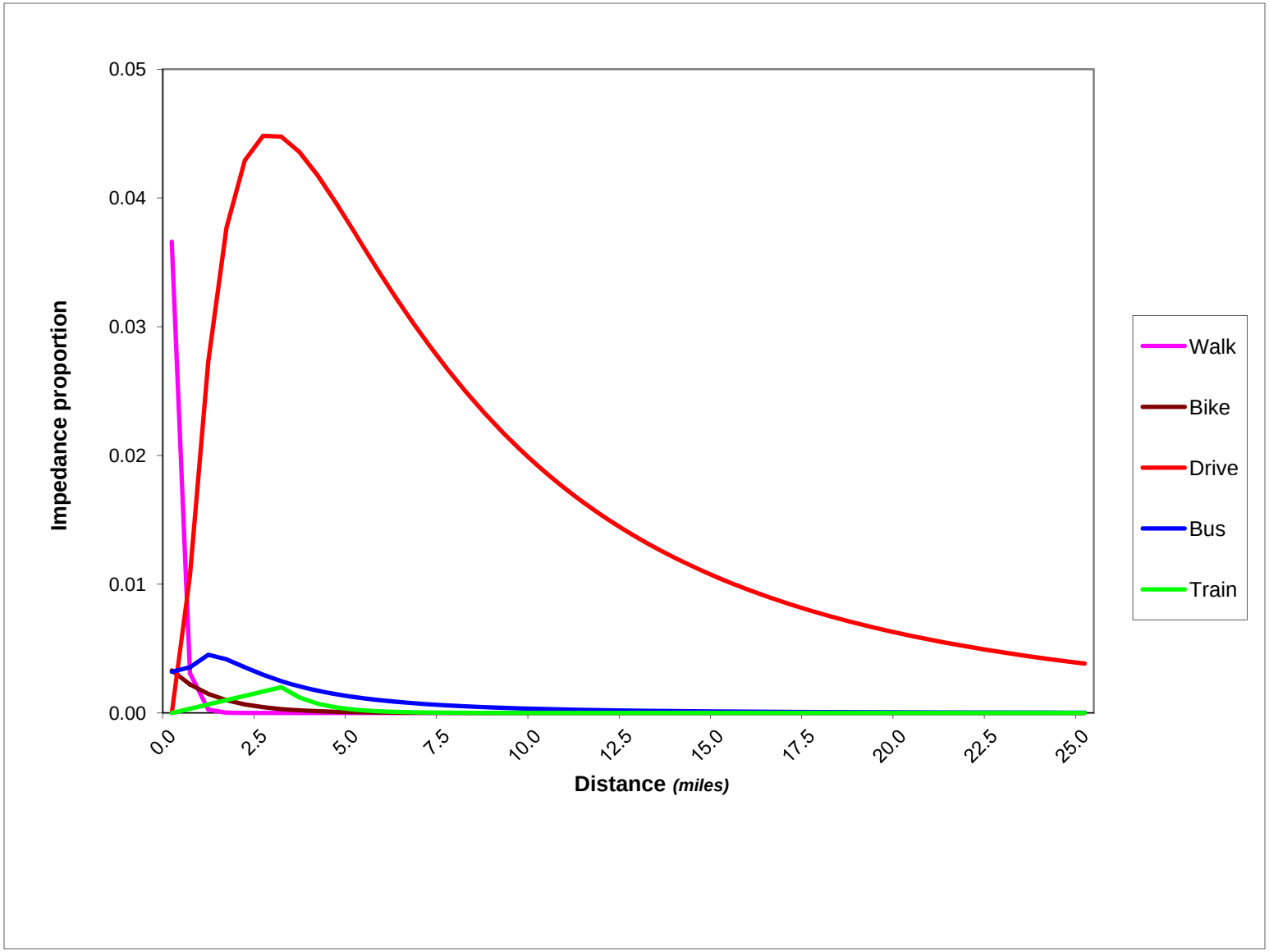
As mentioned at the beginning of this chapter, the lack of information about offender travel modes has necessitated the use of mathematical 'guesses' about travel behavior. However, if it were possible to obtain actual information on travel modes by offenders, then this information could be utilized directly to estimate a much more accurate impedance function. The multinomial logit model in the Discrete Choice module could be used for this purpose (see Chapters 21 and 22). The advantage would be enormous. Instead of guesses about likely impedance functions of specific travel modes, the user would have a function that was based on real data. There should be a substantial improvement in modeling accuracy. However, these data have to be first collected.

CrimeStat IV Mode Split Tools

The *CrimeStat* mode split module allows the relative accessibility function to be calculated. The following provides detailed instructions on running the model. Figure 29.2 above showed the setup page for the mode split routine and Figure 29.3 showed the setup for modes 1 and 2, in the example "Walk" and "Bicycle". The setup for modes 3, 4, and 5 are similar.

Figure 29.9:

Relative Accessibility by Mode



Mode Split Setup

On the mode split setup page, the predicted origin and predicted destination files must be input as the primary and secondary files. If the origin and destination files are identical (i.e., all the origin zones are included in the destination zones), then the file must be input as the primary file.

In addition, the user must input a predicted origin-destination trip file from the trip distribution module. Finally, an assumed impedance value for trips from the "External zone" must be specified. The default is 25 miles. Choose a value that would represent a 'typical' trip from outside the study region.

For each mode, the user must provide a label for the name and define the mathematical function which is to be applied and specify the parameters. The first time the routine is opened, the default values are listed. However, the user can change these.

> Hint: Once the parameters are entered, they can be saved on the Options page. Then, they can be re-entered by loading the saved parameters file.

Constrain Transit Trips to Network

The impedance will be calculated either directly or is constrained to a network. The default impedance is defined with the type of distance measurement specified on the Measurement Parameters page (under Data setup). On the other hand, if the impedance is to be constrained to a network, then the network has to be defined.

Default

The default impedance is that specified on the Measurement parameters page. If direct distance is the default distance (on the measurement parameters page), then all impedances are calculated as a direct distance. If indirect distance is the default, then all impedances are calculated as indirect (Manhattan) distance. If network distance is the default, then all impedances are calculated using the specified network and its parameters; travel impedance will automatically be constrained to the network under this condition.

Constrain to network

An impedance calculation should be constrained to a network when there are limited choices. For example, a bus trip requires a bus route; if a particular zone is not near an existing bus route, then a direct distance calculation will be misleading since it will probably

underestimate true distance. Similarly, for a train trip, there needs to be an existing train route. Otherwise, the routine will assign transit trips where those are not possible (i.e., it will assign train trips where there are no train stations and it will assign bus trips where there are no bus routes). The routine does not 'know' whether there are transit routes and must be told where they are. Even for walking, bicycling and driving trips, an existing network might produce a more realistic travel impedance than simply assuming a direct travel path.

If the impedance calculation is to be constrained to a network, then the network must be defined. A more extensive discussion of a network is provided in Chapter 3 (under Type of distance measurement on the Measurement Parameters page) and in Chapter 30 in the discussion of the Trip Assignment module. Essentially, a network is a series of connected segments that specify possible routes. Each segment has two end nodes (in *CrimeStat*, they are called 'FromNode' and "ToNode). Depending on the type of network, the segments can be bi-directional (i.e., travel is allowed in either direction) or single directional (i.e., travel is allowed only from the "FromNode" to the "ToNode").

A critical component of a network for the mode split routine is that travel can only pass through nodes. This means that two segments that are connected can allow a trip to pass over those two segments whereas two segments that are not connected cannot allow a trip to pass directly from one to the other. From outside the network, a trip connects to it at a node. For a transit network, this can be critical. For a bus route, it may or may not be important. A precise bus network defines nodes by bus stops so that a trip can 'enter' or 'leave' the bus system at a real stop. A less precise bus network defines nodes by the ends of segments (e.g., the end nodes of a TIGER segment). The routine will not know whether the node it enters or leaves from is a real bus stop or not. In the case of bus routes, it probably does not matter since they generally make very regular stops (every two or three blocks).

Accurately defined transit networks

For train networks, however, it is absolutely critical that the network be defined accurately. The nodes must be legitimate stations; a trip can only enter or leave the train system through a station (i.e., it cannot enter or leave a train network at the end of an arbitrary segment node). Most travel demand models use very precise bus and train networks that have been carefully checked; where errors occur, the networks are edited and updated.

Utility for creating transit network

If the user does not have an edited transit network, one can be made in the trip assignment module. There is a "Create a transit network from primary file" routine that will draw segments between input primary file points; the user inputs the station or bus stop locations

as the primary file and the routine creates a network from one point to the next in the *same* order as in the primary file (i.e., the primary file needs to be properly sorted in order to travel). See Chapter 30 for more information about creating a transit network.

Entering the network parameters

The network is input by selecting "Constrain to network" and click on the 'Parameters' button. A dialogue is brought up that allows the user to specify the network to be used. The network file can be either a shape line or polyline file (the default) or another file, either dBase IV 'dbf', Microsoft Excel 'xls/xlsx',Microsoft Access 'mdb', Ascii 'dat', or an ODBC-compliant file. If the file is a shape file, the routine will know the locations of the nodes. All the user needs to do is identify a weighting variable, if used, and possible one way routes ('flags'). For a dBase IV or other file, the X and Y coordinate variables of the end nodes must be defined. These are called the "From" node and the "End" node, though there is no particular order.

An optional weight variable is allowed for both a shape or dbf file. The routine identifies nodes and segments and finds the shortest path. By default, the shortest path is in terms of distance though each segment can be weighted by travel time, travel speed, or generalized cost; in the latter case, the units are minutes, hours, or unspecified cost units.

Finally, the number of graph segments to be calculated is defined as the network limit. The default is 50,000 segments. This can be changed, but be sure that this number is greater than the number of segments in your network.

Minimum absolute impedance

If a mode is constrained to a network, an additional constraint is needed to ensure realistic allocations of trips. This is the minimum absolute impedance between zones. The default is 2 miles. For any zone pair that is closer together than the minimum specified (in distance, time interval, or cost), no trips will be allocated to that mode. This constraint is to prevent unrealistic transit trips being assigned to intra-zonal trips or trips between nearby zones.

CrimeStat uses three impedance components for a constrained network:

1. The impedance from the origin zone to the nearest node on the network (e.g., nearest rail station);

2. The impedance along the network to the node nearest to the destination; and

3. The impedance from that node to the destination zone.

Since most impedance functions for a mode constrained to a network will have the highest likelihood some distance from the origin, it's possible that the mode would be assigned to, essentially, very short trips (e.g., the distance from an origin zone to a rail network and then back again might be modeled as a high likelihood of a train trip even though such a trip is very unlikely).

For each mode that is constrained to a network, specify the minimum absolute impedance. The units will be the same as that specified by the measurement units. The default is 2 miles. If the units are distance, then trips will only be allocated to those zone pairs that are equal to or greater in distance than the minimum specified. If the units are travel time or speed, then trips will only be allocated to those zone pairs that are farther apart than the distance that would be traveled in that time at 30 miles per hour. If the units are cost, then the routine calculates the average cost per mile along the network and only allocates trips to those zone pairs that are farther apart than the distance that would be traveled at that average cost.

Applying the Relative Accessibility Function

To apply the relative accessibility function, the parameter choices for each mode are entered into the mode split routine. All transit modes are then constrained
Once the mode split setup has been defined and all transit modes have been constrained to a proper network, the mode split routine can be run.

Figure 29.10 shows the top 300 walking crime trips in Baltimore County estimated with the default accessibility functions. As seen, the vast majority of walking trips are intra-zonal (local). There are only a couple of inter-zonal walking trip links shown. The default impedance function assigned approximately 4% of the trips to this mode and the result is many intra-zonal trips.

Figure 29.11 shows the top 300 bicycle crime trips in Baltimore County. There are fewer trips by bicycle and they also tend to be quite local. The impedance function used for bicycle trips allocated approximately 1% of all trips to this mode. Thus, it's less frequent than walking mode. There are proportionately more inter-zonal trips among the top 300 than for walking trips, but these tend to be quite short (travel between adjacent zones).

On the other hand, driving is the predominant travel mode for the crime trips (Figure 29.12). The impedance function used allocated approximately 90% of the trips to driving. Among the top 300 links, there were no intra-zonal driving trips. The use of a lognormal function minimized intra-zonal travel.

Figure 29.10:

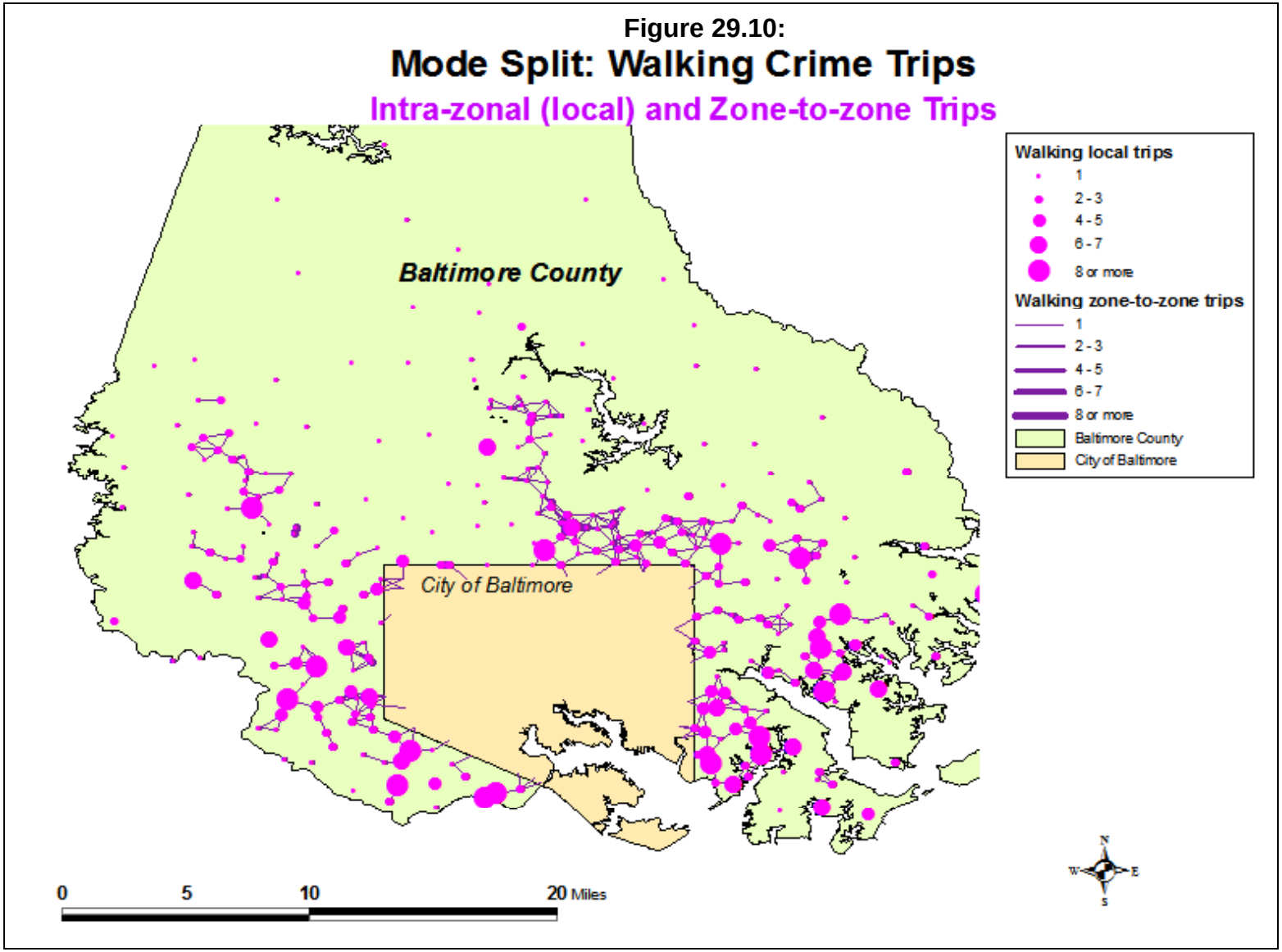

Mode Split: Walking Crime Trips
Intra-zonal (local) and Zone-to-zone Trips
Baltimore County
City of Baltimore
Walking local trips
1
2 - 3
4 - 5
6 - 7
8 or more
Walking zone-to-zone trips
1
2 - 3
4 - 5
6 - 7
8 or more
Baltimore County
City of Baltimore
0
5
10
20 Miles
N
W
E
S

Figure 29.11:

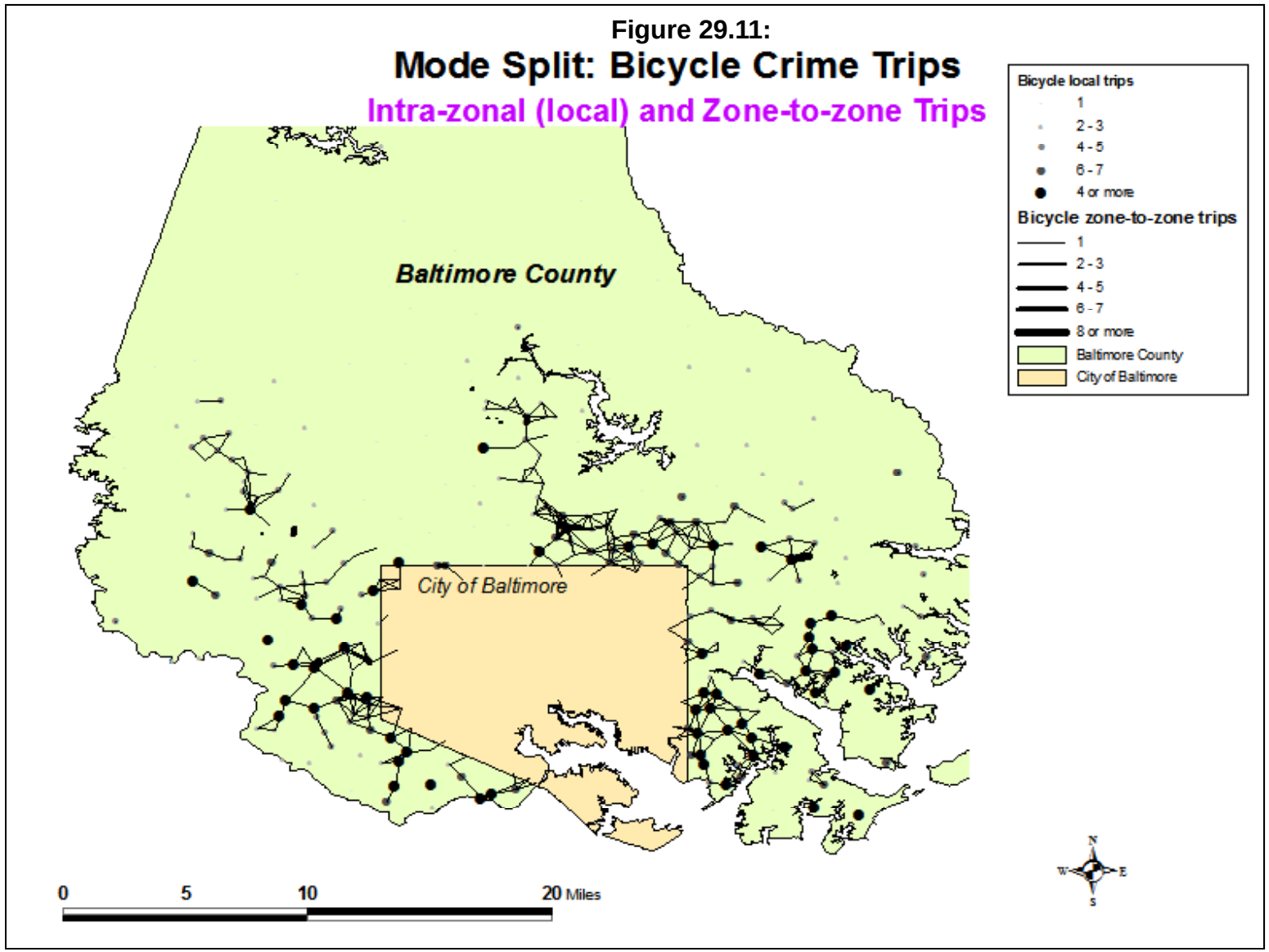
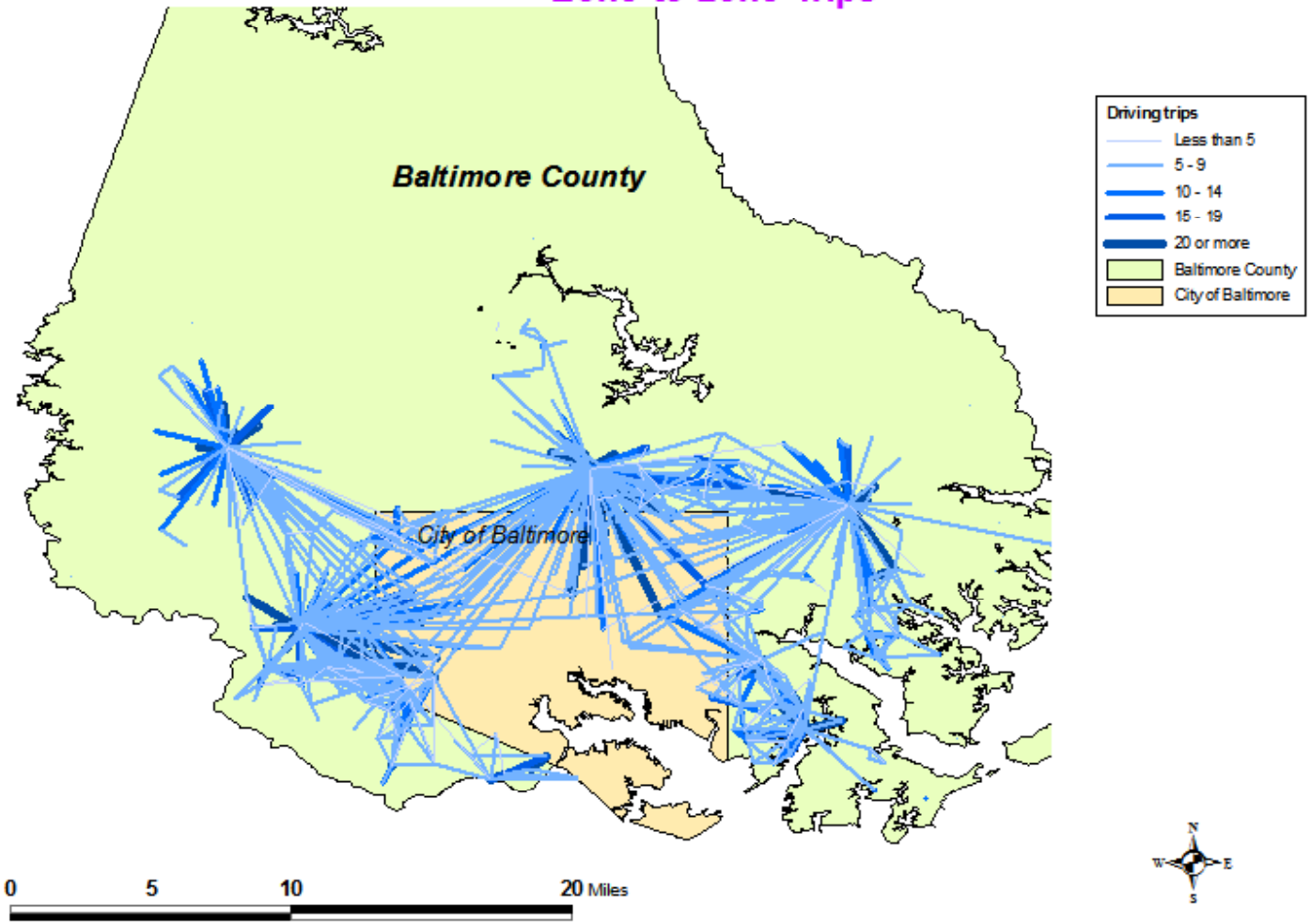

Mode Split: Bicycle Crime Trips
Intra-zonal (local) and Zone-to-zone Trips
Baltimore County
City of Baltimore
Bicycle local trips
1
2 - 3
4 - 5
6 - 7
4 or more
Bicycle zone-to-zone trips
1
2 - 3
4 - 5
6 - 7
8 or more
Baltimore County
City of Baltimore
0
5
10
20 Miles
N
W
E
S

Figure 29.12:

Mode Split: Driving Crime Trips

Zone-to-zone Trips



To allocate bus and train trips, however, it was necessary to constrain them to a network. Separate bus and train networks were obtained from the Baltimore Metropolitan Council. Figure 29.13 shows the Baltimore bus network and Figure 29.14 shows the predicted bus trips superimposed over the bus network. Overall, about 4% of the total trips were allocated to the bus mode by the accessibility function. As seen, the trips tend to be moderate distances and tend to be close to the bus network. Constraining these trips by the network decreased the likelihood that the routine would assign a particular trip link that was far from the bus work to a bus trip.

Finally, train crime trips were constrained to the train network. Figure 29.15 superimposes the assigned train trips over the intra-urban rail network. Overall, only 1% of the total trips were allocated to train mode. Therefore, the number of trips for any zone pair is quite small. The trips are generally longer than the bus trips, as might be expected, and they also tend to fall along the major rail lines. Some of the trips start quite far from the rail lines, so it's possible that these are not realistic representations. Keep in mind that this is a mathematical model and is far from perfect.

Overall, the mode split routine has produced a reasonable approximation to travel modes for crime trips. Since there was no data upon which to calibrate the functions, reasonable guesses were made about the accessibility function. The mathematical model produced a plausible representation of these assumptions, generally fitting into what we know about crime travel patterns.

Usefulness of Mode Split Modeling of Crime Trips

The mode split model is a logical extension of the travel demand framework. For transportation planning, it is an important step in the process. But, it also is important for crime analysis. First, it addresses the complexity of travel by separating the trips from specific origins to specific destinations into distinct modes. In this sense, it adds more realism to our understanding of criminal travel behavior. The Journey-to-crime literature, which has been used by crime analysts and criminal justice researchers to "understand" criminal travel behavior, is simplistic in this respect. It assumes a single mode, though that is rarely articulated by the researchers. By pointing out typical travel distances by offenders circumvents the critical question of how they made the trip. This was, perhaps, not as critical 50-60 years ago when most crimes were committed within a smaller community and it could be assumed that most offenders walked to the crime location. But in the post- World War era, automobile travel has become increasingly dominate. This model assumes that the vast majority of crime trips are taken by automobile. While there is currently no data to prove that assertion, it follows from the transportation patterns that have become widespread in the U.S. and elsewhere.

Figure 29.13:

Baltimore Bus Network

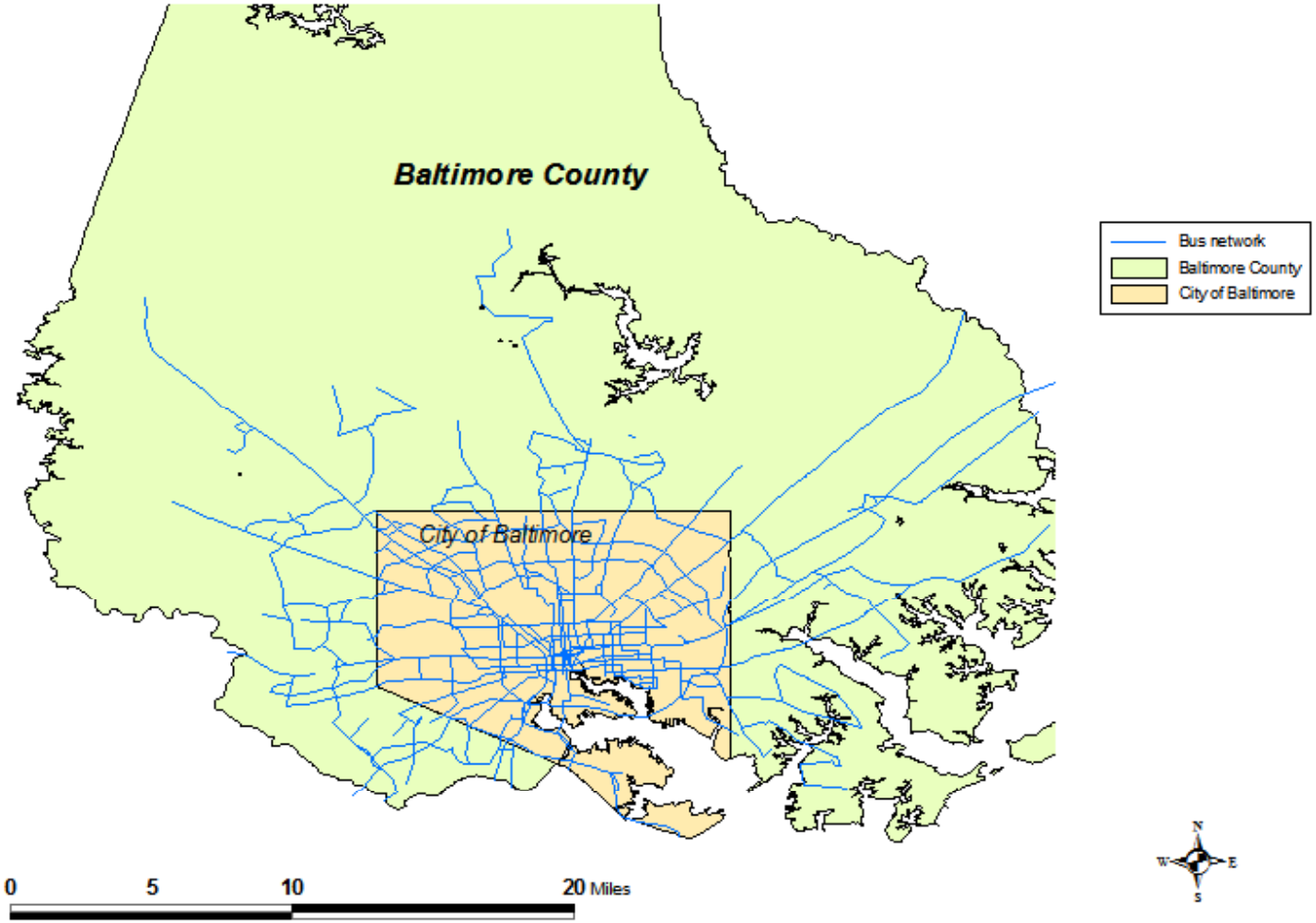


Figure 29.14:

Mode Split: Bus Crime Trips

Zone-to-zone Trips

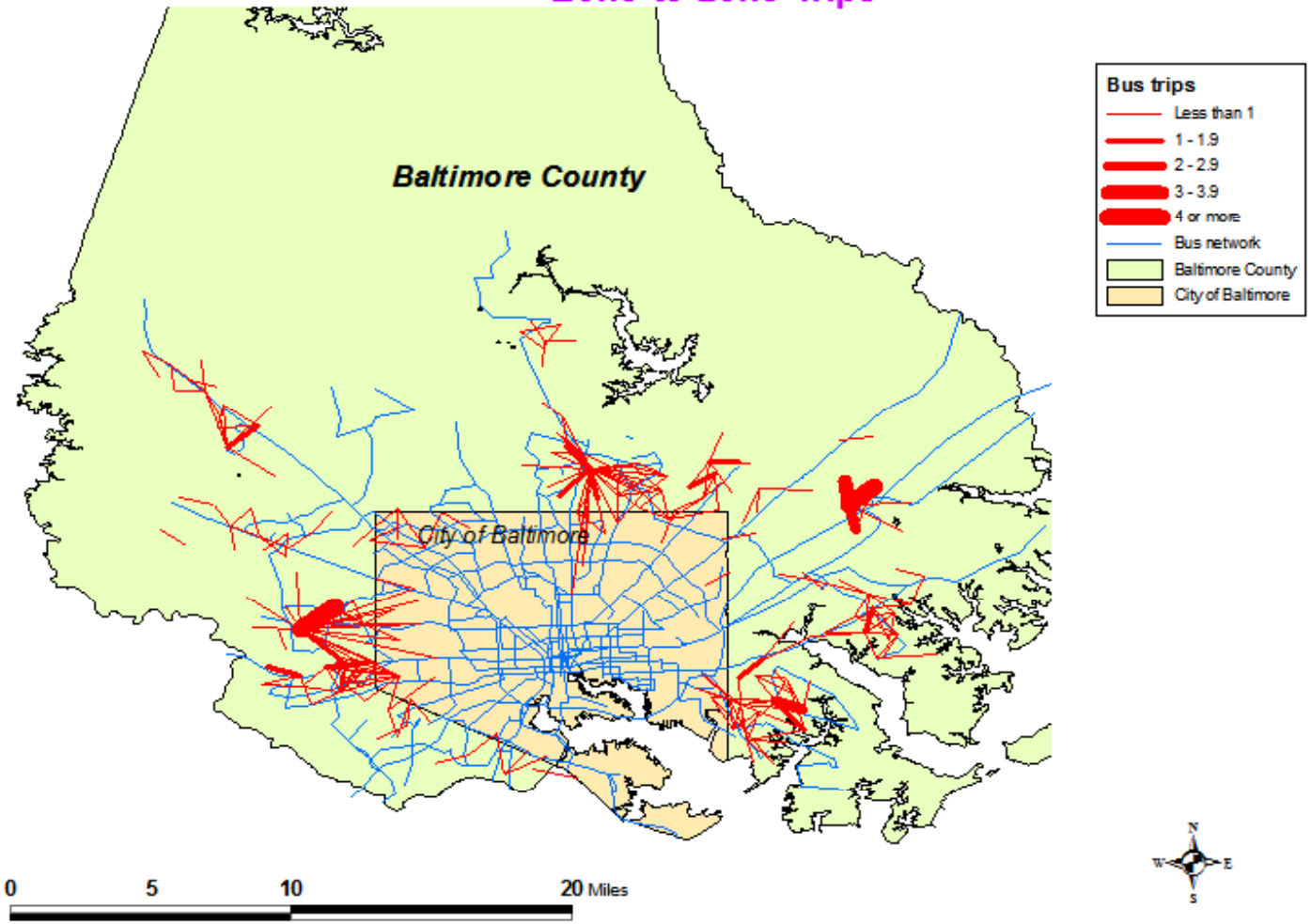
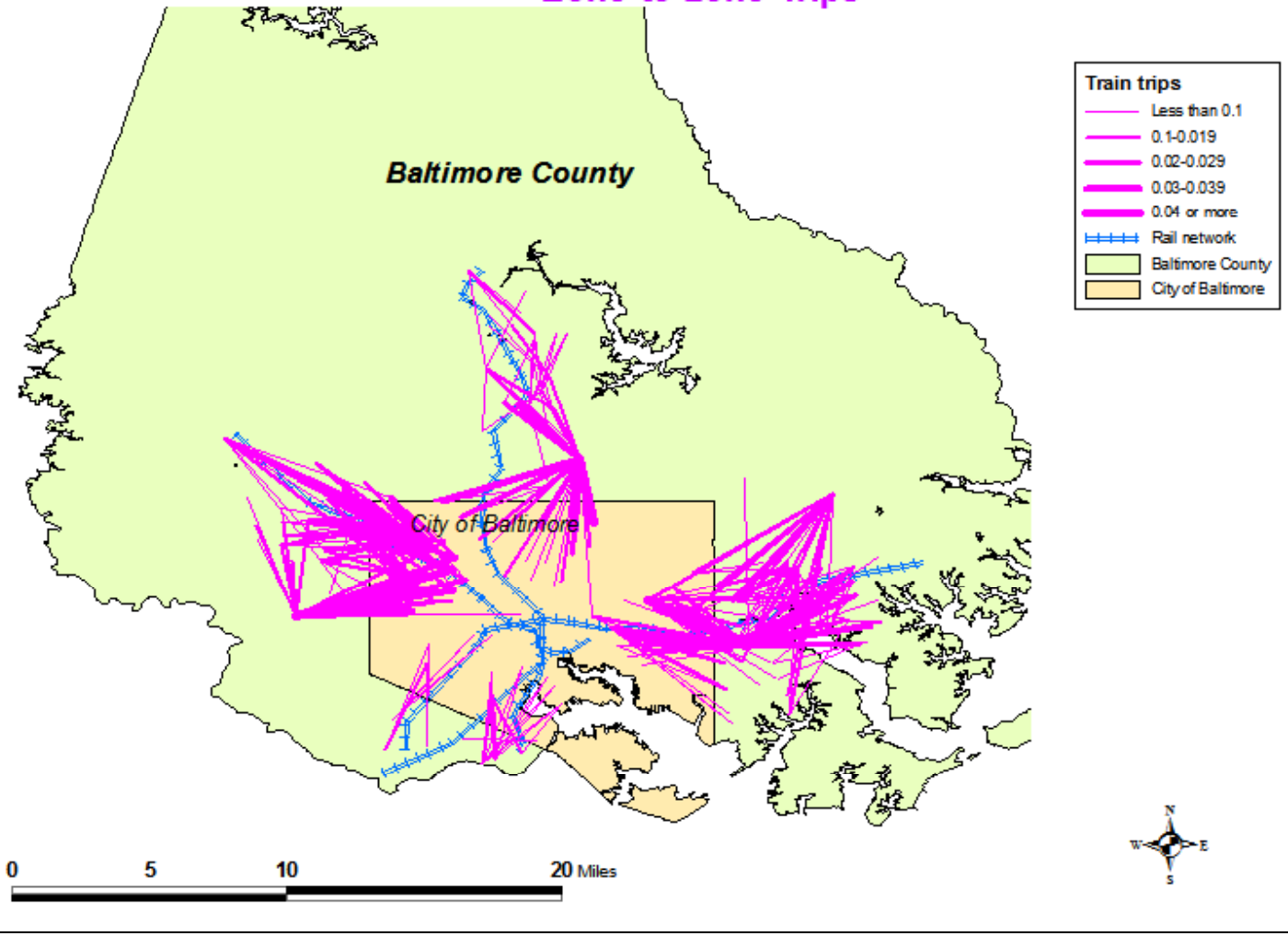


Figure 29.15:

Mode Split: Intra-urban Train Crime Trips

Zone-to-zone Trips



There is a second reason why an analysis of crime travel mode can be important. *If* the limitations of travel mode information could be improved through better and more careful data collection by police and other law enforcement agencies, this type of analysis could be very useful for policing. For one thing, it could allow more focused police deployment. For neighborhoods with a predominance of walking crime trips, then a police foot patrol could be most effective. Conversely, for neighborhoods with a predominance of driving crime trips, then patrol cars are probably the most effective. Police intuitively understand these characteristics, but the crime mode split model makes this more explicit.

For another thing, a mode split analysis of crime can better help crime prevention efforts. As the Baltimore data suggest, many of the local (intra-zonal) crime trips are committed around housing projects and in very low income communities. Most likely, this is a by-product of poverty, lack of local employment opportunities, deteriorated housing, and even poor surveillance. Since teenagers are more likely to *not* own vehicles, it might be expected that the majority of these local crime trips are committed by very young offenders (see Levine & Lee, 2012). This can be useful in crime prevention. Again, "Weed and Seed" and after-school programs are generally targeted to youth from very low income neighborhoods. What is shown by the mode split analysis is probably the crime patterns associated with these neighborhoods. Even though it is intuitively understood, the mode split analysis quantifies these relationships in an explicit manner.

In short, a mode split analysis of crime trips is an important tool for crime analysts and criminal justice researchers. If correctly calibrated, it can help focus police enforcement and crime prevention efforts more specifically and can improve the theory of criminal travel behavior.

Hopefully, police departments will start to improve the quality of data in capturing likely travel modes while taking incident reports. Even though most police departments have an item similar to "Method of departure", there has not been a lot of emphasis on this information and most crime data sets are deficient on this information. However, with improved data will come more accurate accessibility functions and, hopefully, even real utility functions where actual costs are measured. The expectation is that this will happen and we should work towards accelerating the process.

Limitations to the Mode Split Methodology for Crime Analysis

There are also limitations to the method, particularly the aggregate approach. First, the aggregate approach does not consider individuals, only properties associated with zones (e.g., average travel time between two zones). As mentioned earlier, the accessibility function used

(or the underlying utility theory) is much simpler for zones than for individuals. Consequently, the analysis is cruder at an aggregate level than at an individual level. Policy scenarios are much more limited with aggregate mode split than with individual-level models. For example, if an analyst wanted to explore what was the likely effect of increased public surveillance on walking behavior by pickpockets, it is more difficult to do with aggregate data than with individual data. For example, it could be hypothesized that actual pickpockets are more sensitive to increased public surveillance than, say, car thieves, but this cannot be tested at the aggregate level. Instead, some general characteristics are assigned to all individuals (e.g., the number of security personnel in a zone).

Second, the zonal model for mode split (as with trip distribution) cannot explain intra-zonal travel. For intra-zonal trips, it is inaccurate and generally defaults to simple choices (e.g., walking, biking or driving). For example, bus or train mode can rarely be applied at an intra-zonal level because there are usually too few network segments that traverse a zone and the segments rarely stop within the zone. While this deficiency also applies to the trip distribution model, the dependence on a network for transit modes, particularly, leads to underestimation of transit use for very short trips.

Third, the zonal mode split model cannot explain individual differences. This goes back to the first point that a single utility function is being applied at the zonal level. Thus, the value of time to different individuals living in the same zone cannot be examined; instead, everyone is given the same value.

Fourth, the aggregate mode split model does not analyze time of day very well. The probabilities are assigned to all trips, rather than to trips taken at particular times of the day. To conduct that analysis, an analyst has to break down crimes by time of day and model the different periods separately. Aside from being awkward, the summed trips need to be balanced to ensure that they sum to the total number of trips.

Fifth, and finally, the mode split model, both aggregate and disaggregate, cannot explain *linked trips* (sometimes called *trip chaining*). An offender might leave home one day, go out to eat, visit a friend, commit a street robbery, go to a 'fence' to distribute the goods, buy drugs from a drug dealer, and then finally go home. The mode split model treats each of these as separate trips; in the case of crime mode split, there are three distinct crime trips - committing the robbery, selling the stolen goods to the 'fence', and buying the drugs from the drug dealer. The model does not understand that these are related events, but instead assigns separate mode probabilities to each trip. Thus, it is possible to produce absurd choices, such as driving to the crime scene, taking the bus to the drug dealer, and then biking home. In this respect, the disaggregate approach is equally flawed as the aggregate since both treat each trip as independent events. The solution to this lies in a 'third generation' of travel modeling in which individual trip makers are simulated over a day; *activity-based modeling*, as it is known, is still in

a research stage (FHWA, 2009; Culp & Lee, 2005; Miller & Salvini, 1999). But, it will eventually emerge as the dominant travel demand modeling algorithm.

Conclusions

Nevertheless, mode split modeling can be a very useful analysis step for crime analysis. It represents a new approach for crime analysis and one with many useful possibilities. It will require building more systematic databases in order to document travel modes. But, the possibilities that it offers up can be important for crime analysts and criminal justice researchers alike.

References

Ben-Akiva, M. & Lerman, S. (1985). *Discrete Choice Analysis: Theory and Application to Travel Demand*. MIT Press: Cambridge.

Bernasco, W. & Block, R. (2009). Where offenders choose to attack: A discrete choice model of robberies in Chicago. *Criminology* 47(1): 93-130.

BTS (2002). *National Household Travel Survey: Daily Travel Quick Facts*. Bureau of Transportation Statistics, U.S. Department of Transportation: Washington, DC. http://nhts.ornl.gov/download.shtml#2009. Accessed June 1, 2012.

Carnegie-Mellon University (1975). *Security of Patrons on Urban Public Transportation Systems*. Transportation Research Institute, Carnegie-Mellon University: Pittsburgh, PA. Johnson, M.A. (1978). Attribute importance in multiattribute transportation decisions, *Transportation Research Record*, 673, 15-21.

Culp, M. & Lee, E. J. (2005). Improving travel models through peer review. *Public Roads*, 68 (6), FHWA-HRT-05-005. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/publications/publicroads/05may/07.cfm. Accessed April 28, 2012.

Domencich, T. & McFadden, D. (1975). *Urban Travel Demand: A Behavioral Analysis*. North Holland Publishing Company: Amsterdam & Oxford (republished in 1996). Also found at http://emlab.berkeley.edu/users/mcfadden/travel.html. Accessed April 28, 2012.

FHWA (2009). Integrated Urban Systems Modeling, *The Exploratory Advanced Research Program Fact Sheet*, FHWA-HRT-09-042. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/advancedresearch/pubs/interurbsys.pdf. Accessed April 28, 2012.

Levine, N. & Lee, P. (2009). Bayesian journey to crime modeling of juvenile and adult offenders by gender in Manchester. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 237-251.

Levine, N. & Wachs, M. (1986). Bus Crime in Los Angeles: II - Victims and Public Impact. *Transportation Research*. 20 (4), 285-293.

McCormick Rankin (2011). *Transportation Demand Management Plan: Final Report*. Ottawa. http://ottawa.ca/cs/groups/content/@webottawa/documents/pdf/mdaw/mdc3/~edisp/cap078202.pdf. Accessed June 1, 2012.

McFadden, D. L. (2002). The path to discrete-choice models. *Access*, No. 20, Spring. 20-25. http://www.uctc.net/access/access20.shtml. Accessed April 28, 2012.

References (continued)

McGuckin, N. A. & Srinivasan, N. (2003). *Journey to Work in the United States and its Major Metropolitan Areas*. FHWA-EP-03-058, Office of Planning, Federal Highway Administration: Washington, DC.

Miller, E. J. & Salvini, P. A. (1999). Activity-based travel behavior modeling in a microsimulation framework. Paper presented at IATBR Conference, Austin, TX. December. http://www.civ.utoronto.ca/sect/traeng/ilute/downloads/conference_papers/miller-salvini_iatbr-97.pdf. Accessed May 4, 2012.

Oppenheim, N. (1980). *Applied Models in Urban and Regional Analysis*. Prentice-Hall, Inc.: Englewood Cliffs, NJ.

Ortuzar, J. D. & Willumsen, L. G. (2001). *Modeling Transport* (3rd edition). J. Wiley & Sons: New York.

Ottawa (2008). *Transportation Master Plan*. Regional Municipality of Ottawa-Carleton. http://ottawa.ca/en/city_hall/planningprojectsreports/master_plans/tmp/. Accessed June 1, 2012.

Porter, C., Suhrbier, J. & Schwartz, W. L. (1999). Forecasting bicycle and pedestrian travel: State of the practice and research needs. *Transportation Research Record*, 1674, 94-101.

Portland (1998). *Bicycle Master Plan*. Resolution 35515, Office of Transportation, City of Portland: Portland, OR. http://www.portlandonline.com/transportation/index.cfm?a=369990&c=49304. Accessed June 1, 2012.

Roemer, F. & Sinha, K. (1974). Personal security in buses and its effects on ridership in Milwaukee, *Transportation Research Record*, 487, 13-25.

Schnell, J. B., A. J. Smith, K. R. Dimsdale, & L. J. Thrasher (1973). *Vandalism and Passenger Security: A Study of Crime and Vandalism on Urban Mass Transit Systems in the United States and Canada*. Prepared by the American Transit Association for the Urban Mass Transportation Administration (now Federal Transit Administration), U. S. Department of Transportation. National Technical Information Service: Springfield, VA. PB 236-854.

Schwartz, W.L., C. D. Porter, G.C. Payne, J.H. Suhrbier, P.C. Moe, & W.L. Wilkinson III (1999). *Guidebook on Methods to Estimate Non-Motorized Travel: Overview of Methods*. Turner-Fairbanks Highway Research Center, Federal Highway Administration: McLean, VA. July. http://www.fhwa.dot.gov/publications/research/safety/pedbike/98165/index.cfm. Accessed June 1, 2012.

Stopher, P. R. & Meyburg, A. H. (1975). *Urban Transportation Modeling and Planning.* Lexington, MA: Lexington Books.

References (continued)

Turner, S., Shunk, G. & Hottenstein, A. M. (1998). *Development of a Methodology to Estimate Bicycle and Pedestrian Travel Demand*. Report 1723-S, Texas Transportation Institute: College Station. http://tti.tamu.edu/publications/catalog/record/?id=146. Accessed April 28, 2012.

Train, K. (2009). *Discrete Choice Methods with Simulation* (2nd edition). Cambridge University Press: Cambridge.

U.S. Census Bureau (2009). Commuting (Journey to Work) Main. U.S. Census Bureau, U.S. Department of Commerce: Washington, DC. http://www.census.gov/hhes/commuting/. Accessed June 1, 2012.

WASHCOG (1974). *Citizen Safety and Bus Transit*. Metropolitan Washington Council of Governments. National Technical Information Service, Springfield, VA. PB 237-740/AS.

Chapter 30:

Crime Network Assignment

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 30:
Crime Network Assignment

In this chapter, the fourth, and last, component of the crime travel demand model will be described. *Network assignment* involves the assigning of predicted trips to particular routes. The predicted trips are those that are either predicted from the trip distribution stage or from the mode split stage. In the former case, all trips from each origin zone to each destination zone are assigned to a particular travel route, usually on the assumption that they all travel with the same mode of travel (usually walking, biking or driving). In the latter case, the predicted trips from each origin-destination zone pair by specific travel modes are assigned to a particular route which is mode specific. Thus, bus trips are assigned to bus routes; train trips are assigned to train routes; driving trips are assigned to a road network; walking trips are assigned to a more limited road network; and biking trips are assigned to a mixture of roads and bike paths. In other words, the assignment of travel modes is specific to a particular network.

Once the trips are assigned to routes, several statistics can be calculated. First, the predicted path from an origin zone to a destination zone can be displayed. This can very useful for police who could increase their patrol on high crime routes. Second, the entire *trip load* on road segments can be calculated. Since many crime trips pass over the same network segments (e.g., freeways, major arterial roads), the total number of predicted trips on individual segments can be enumerated. The result is a map of the most heavily traversed segments in the network. Again, this can be very useful for police.

Thus, the network assignment completes the four stage modeling process of the crime travel demand framework. To summarize, in the first stage - trip generation, separate models of the number of crimes originating in each zone and the number of crimes ending in each zone are developed. In the second stage - trip distribution, the predicted number of crimes originating in each zone are allocated to each destination zone; the result is a prediction of the number of trips that occur between each origin-destination zone pair. In the third stage - mode split, each predicted origin-destination trip pair is separated (split) into distinct travel modes (e.g., walking, biking, driving, bus, train) with the result being a mode-specific origin-destination zone pair. Finally, the fourth stage - network assignment, assigns these trips to specific routes.

Theoretical Background

To understand the background, we need to look, first, at the nature of networks and second, at types of routing algorithms.

Networks

The most fundamental element of assignment is, of course, a network. The network can be a road network, a bus network (e.g., bus routes with stops), a train network (e.g., train lines with stations), or even a bicycle network (e.g., a mixture of roads and bicycle paths). Other kinds of networks can also be considered, for example telecommunication lines or even trade routes. We will concentrate on urban transportation networks, however.

The mathematical properties of networks are known as *graph theory* (Sedgewick, 2002). A network (or graph) is a set of nodes (or vertices) and a set of segments (or edges) that connect pairs of nodes. If there are V nodes (vertices), then there are V^2 pairs of nodes, including the distance from a node to itself. A graph with V nodes has, *at most*, $V(V-1)/2$ segments (edges); if multiple segments share nodes, then there will be even fewer.

Figure 30.1 illustrates a simple network. Travel occurs along the segments through the connecting nodes. A path is a sequence of nodes in which each successive node is connected to its predecessor in the path. hus, in the figure, there cannot be direct travel between node A and node C, but must go through an intermediate node (e.g., through B or through a path from D to E to C).

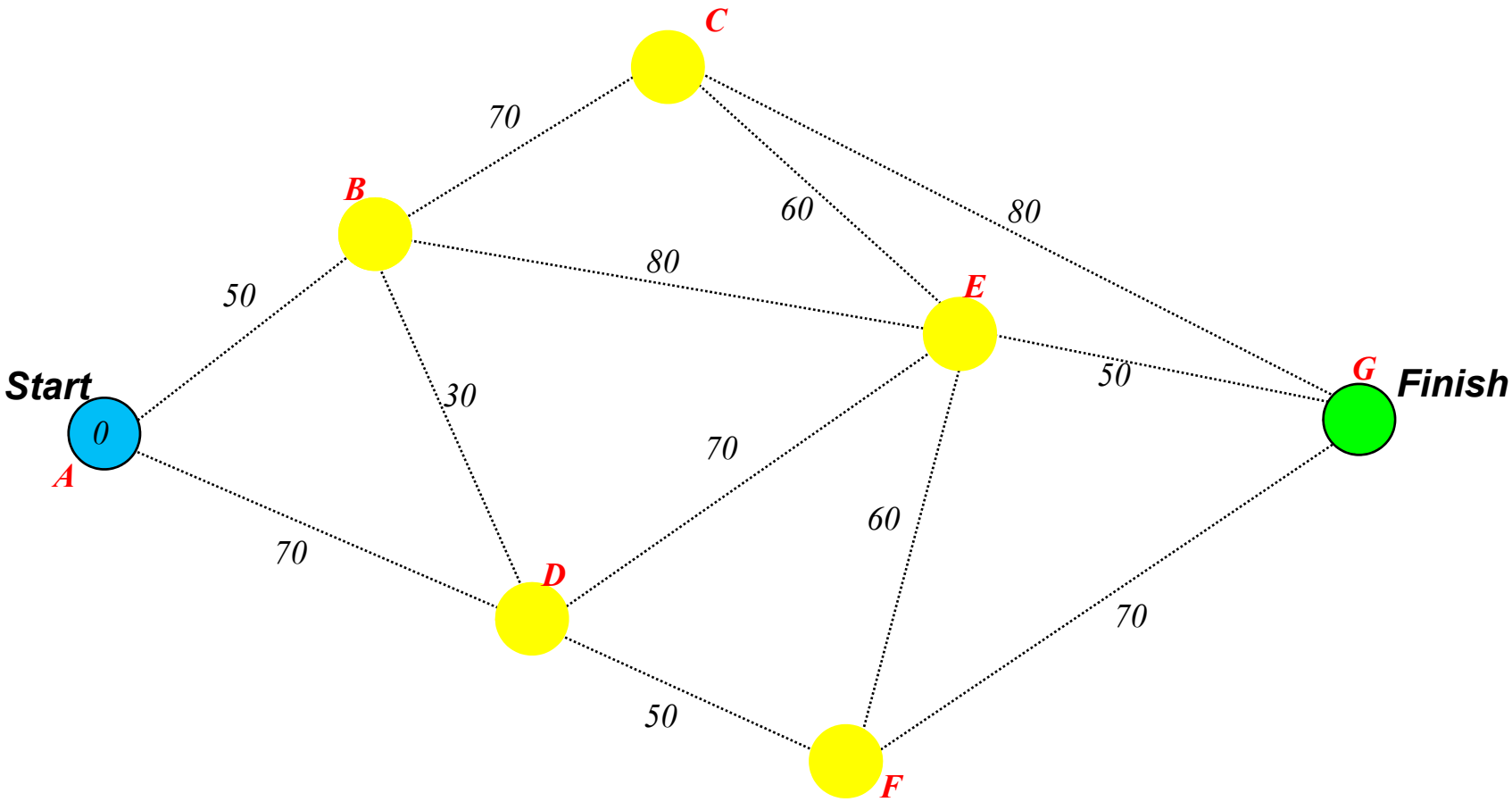
Impedance of a Network

There are several properties of a network that are important for travel modeling. First, the *length* of a segment is proportional to its impedance (see Chapters 28 and 29). The simplest kind of impedance is distance in which each unit length of the network corresponds to some unit of distance in the real world (e.g., one inch = 1 miles; one centimeter = 5 kilometers). This is analogous to the *scale* used in mapping systems. More complex types of impedance involve travel time, speed, or even generalized cost (a collection of several cost elements). Thus, to use the example in Figure 30.1, node A is connected to nodes B and D. The path from A to B is 50 units long; similar lengths are found for the other segments in the example. This could represent distance (e.g., 50 miles), travel time (e.g., 50 minutes), or generalized cost (e.g., $50).[1] To a graph, the units are irrelevant. As long as the user is explicit about these and consistent, path calculations will work properly.

[1] Speed could be used, but it is inversely proportional to impedance (i.e., the higher the speed, the less the impedance). Most shortest path algorithms treat the weight as proportional. However, speed can be converted into travel time by dividing distance by speed. To use the example, if the length is 1 mile long and the speed is 50 miles per hour, then the travel time is 1/50 hours (or 1.2 minutes).

Figure 30.1:

A Simple Network



Bi-directional and Single Directional Networks

Bi-directional networks

Second, typical transportation networks are either *bi-directional* or *single directional*. In a bi-directional network, travel can occur in either direction. Again, using Figure 30.1, if the network is bi-directional, then travel can occur from A to B or from B to A. A well known example of a bi-directional network is the TIGER system of the U.S. Census Bureau (2011). This is a representation of all major urban lines, including streets, railroad lines, census geography boundaries, jurisdictional boundaries, Congressional boundaries, and other features. It is used to map out census areas for the purpose of collecting the decennial Census. Virtually the entire United States is now mapped in the TIGER system. Depending on how carefully each jurisdiction updates the database for new roads and changes in existing roads, the TIGER system can be a very accurate spatial representation of the an urban road system. It is a widely available system and is often the first network that most police departments use when they create a crime mapping system. Figure 30.2 shows a TIGER network for Baltimore County and the City of Baltimore. There are 49,015 road segments in the TIGER map shown in the figure.

Problems with the TIGER system for travel modeling

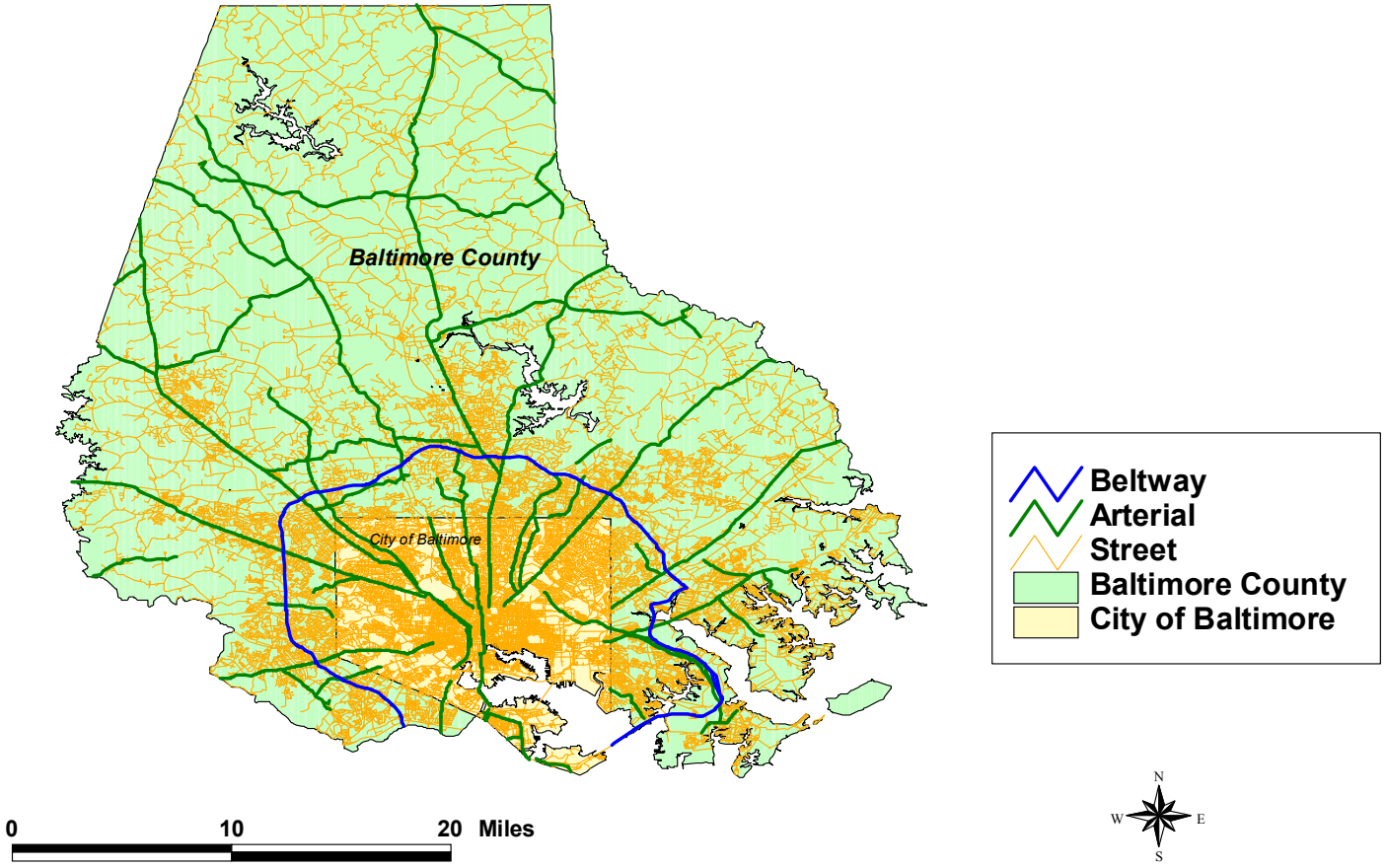
On the other hand, for travel modeling, there are substantial problems with bi-directional networks and with TIGER in particular. A major problem is that *connectivity* is often not tested. Since the aim of the TIGER system is to represent a metropolitan area for the purpose of collecting the Census, connectivity is not guaranteed since it is irrelevant for that purpose. It is not clear that all roads are properly represented, a feature that could substantially alter a shortest path algorithm. For example, in Figure 30.1, if the segment from A to B was not connected, then travel from A to C would have to take a circuitous path from A to D to E to C. Having an accurate and edited network is critical for modeling travel behavior. With a large number of segments in a TIGER system, it is often not clear where in a file connectivity is not properly linked.

Another problem is that TIGER is typically less accurate with respect to rail lines and has virtually no information about bus routes, which are local in nature. Depending on how diligent the local government is in updating the database, the representation may not be as accurate as possible (though, in general, it is getting better over time).

Another major deficiency of the TIGER system is the lack of information about travel time or travel cost. Travel along a TIGER network is defined by distance, which does not change by time of day. It does not have cost information either, which makes it less flexible for examining alternative routes as a function of additional cost factors (e.g., an analysis of travel

Figure 30.2:

TIGER Street Network
49,015 Road Segments



through an area with high surveillance compared to travel through an area with low security presence even if travel through the first area is shorter in time than through the second area). The TIGER system does have information about functional class of road (interstate, state highway, collector road) and it is possible to assign *a priori* speeds to the different segments based on these classes (e.g., 35 miles per hour for Interstate highways, 25 miles per hour for principal arterial roads). But, because the network is bi-directional, it is impossible to assign speeds for travel in opposite directions; in reality, there are usually differences in travel speeds in opposite directions (e.g., travel into the central business district in the morning might be at 15 miles per hour whereas travel in the opposite direction might be at 35 miles per hour).

Another major problem with TIGER and with a bi-directional network in general is in the representation of one-way streets. The TIGER system does not provide this information. Consequently, in using a TIGER file for modeling travel, a shortest path could easily travel up a one-way street in the wrong direction. To make the system work properly, there needs to be an additional field in the database that identifies a segment as one-way.

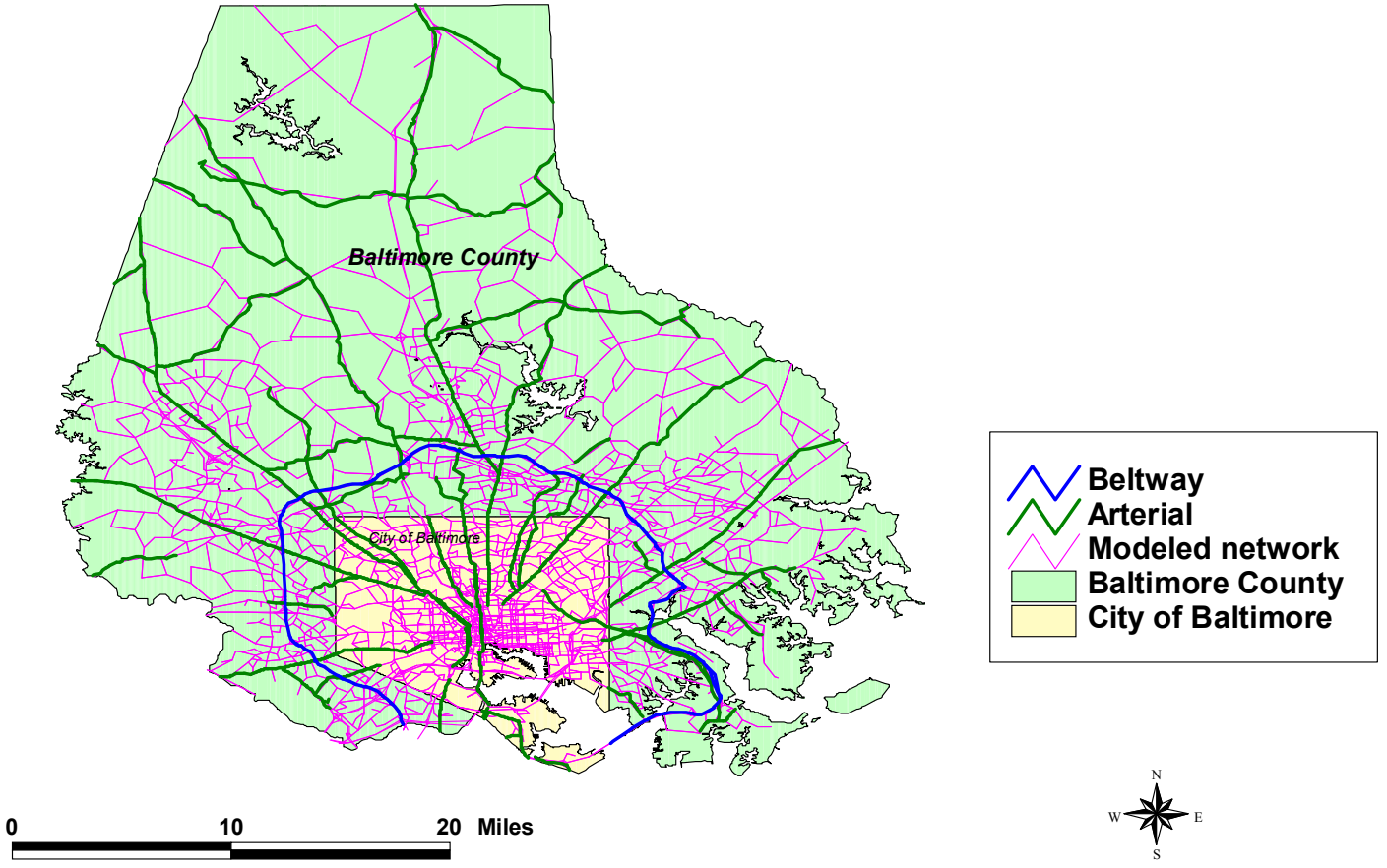
Single directional networks

A single directional (or uni-directional) network, on the other hand, allows travel in only one direction. This has the advantage of keeping travel consistently defined. Two-way travel is represented by two segments, one in each direction (e.g., one for travel from A to B and one for travel from B to A). One-way streets can be characterized by only one of the paired directions. Most transportation modeling networks are single directional since an accurate representation of travel is critical. Travel times, speeds or costs can be assigned to the different directions of travel between two nodes and can be further assigned to different times of the day (e.g., 20 miles per hour in the morning peak period, 15 miles per hour in the afternoon peak period, 30 miles per hour in the off-peak daytime period, and 45 miles per hour at nighttime).

An example of a single directional network is that used for travel demand modeling by most Metropolitan Planning Organizations (MPO). These are used to model travel over an entire metropolitan area (regional travel) and are generally updated regularly; connectivity is continuously tested and errors are few in number. The travel modeling network is usually a 'skeleton' network, covering all the major roads - freeways, principal arterial roads, minor arterial roads, and some collector roads. They usually do not include much information about local or neighborhood streets since these are not very relevant for regional travel modeling. Figure 30.3 shows a modeling network used by the Baltimore Metropolitan Council for their travel demand model. There are only 11,045 road segments in the file, less than one fourth the size of the corresponding TIGER network. Considering that each segment in a single direction, effectively only about 5,000-6,000 actual roads are being represented in the file.

Figure 30.3:

Modeled Street Network
11,045 Road Segments



Most importantly, modeling networks usually include information about travel time or travel speed (which can be converted to travel time by dividing distance by speed) and are usually broken down into different time periods. Thus, it becomes possible to analyze travel at different times of the day to account for the major congestion effects that occur at the peak travel periods, particularly the afternoon peak. Some modeling networks also include information on travel costs, which include parking, toll roads, and other costs that impact a trip. As mentioned in Chapter 26, any analyst wishing to develop a crime travel demand model should contact the local MPO about obtaining a copy of the modeling network used.

> Hint: A single directional network can also be treated as bi-directional. In this case, all the trips on that roadway will generally be assigned to only one of the paired segments (for a two-way pair). For the network load output, particularly, this can be useful for showing the total number of trips on a road segment, independent of direction. Otherwise, if defined as a single directional network, the loads in each direction will be displayed separately.

Problems with modeling networks

Modeling networks also have their problems. The biggest one is that they do not include all roads, but only the more important regional ones. This can lead to unrealistic paths being modeled in a neighborhood (e.g., entering or leaving a neighborhood from a centroid, rather than from a real street; taking circuitous routes to travel a short distance in space when, in fact, there are connecting local roads that actually exist but are not included in the file). However, neighborhood roads can usually be added to the network to provide more detail at the neighborhood level and to correct modeling errors. It is a tedious process, but a police department could slowly update such a system over time and improve its accuracy. Care must be taken in doing this, however, to ensure that connectivity is correctly portrayed.

Another problem, which may or may not be critical, is that the representation of roads in a modeling network is spatially simplified. Road segments are straight lines, rather than having curvature. In the TIGER system, the basic record of a segment is a straight line connecting two nodes, but also includes up to 10 intermediate 'shape grammar' nodes that define curvature (integrated with spatially more accurate information from the U.S. Geological Survey). Thus, a modeling network looks a little 'unreal' at a neighborhood level since there are only straight lines. But additional segments can be added to the file to improve local connectivity as well as familiarity.

Transportation Networks

The third property of a network for travel modeling is the type of network. Road networks were mentioned above. But there are also transit networks (e.g., bus routes, train routes) and even bicycle networks (e.g., bike paths). If a trip distribution matrix of trips from origins to destinations is analyzed by travel mode, then it is critical to have a mode-specific network. Using TIGER or a simple modeling network will imply that all trips occur by the existing road system. For transit trips (bus and rail) particularly, but also for biking trips and possibly walking trips, features that are specific to the travel mode must be included. Bus routes will use the existing road system, but they do not use all roads, typically only the major arterial roads. Train systems rarely use the existing road system, but usually have dedicated tracks. There are exceptions. Some light rail systems do run on arterial roads. Other rail systems will run on an arterial road, but with a grade separation. Depending on how the MPO conceptualizes this, there may be separate lines for the rail or not.

Thus, it is very important to check and edit all networks that are used. For transit networks, in particular, the lines need to be connected and thoroughly tested. Figure 30.4 repeats the Baltimore bus network map from Chapter 29 (figure 29.13). Each of the lines on the map represent bus routes; there can be (and usually are) more than one bus route running on any one line. Typically, these are drawn as separate line objects and are overlaid on each other. This particular network does not have information about bus stops. Consequently, a shortest path algorithm will choose the end nodes of segments to allow a trip to "enter" or "leave". Thus, it is possible that a bus trip would start at a location where there is not a bus stop. However, given that buses in Baltimore and elsewhere stop very frequently (every two or three blocks on average), the amount of error introduced is quite small.

With trains, however, it is absolutely critical that station locations be used to define the rail lines; people cannot enter or leave a train between stations. Figure 30.5 shows each of the four intra-urban rail lines with the station locations. Later in this chapter, there will be a discussion of a utility for creating rail lines from station locations. But a critical point is that each of the end points of the rail segments be associated rail stations. In the figure, each of the four rail lines is shown in separate color. For modeling in *CrimeStat*, however, the individual lines need to be merged into a single file in order for the shortest path routine to be able to move between rail lines (i.e., if there are separate line objects for each line, the routine will not know how to move from one line to another). Figure 30.6 shows the full rail line network.

Shortest Path Algorithms

Once a network has been created, edited and thoroughly tested for accurate connectivity, it can be used for a shortest path analysis. In a *shortest path* for a single trip (from an origin

Figure 30.4:

Baltimore Bus Network

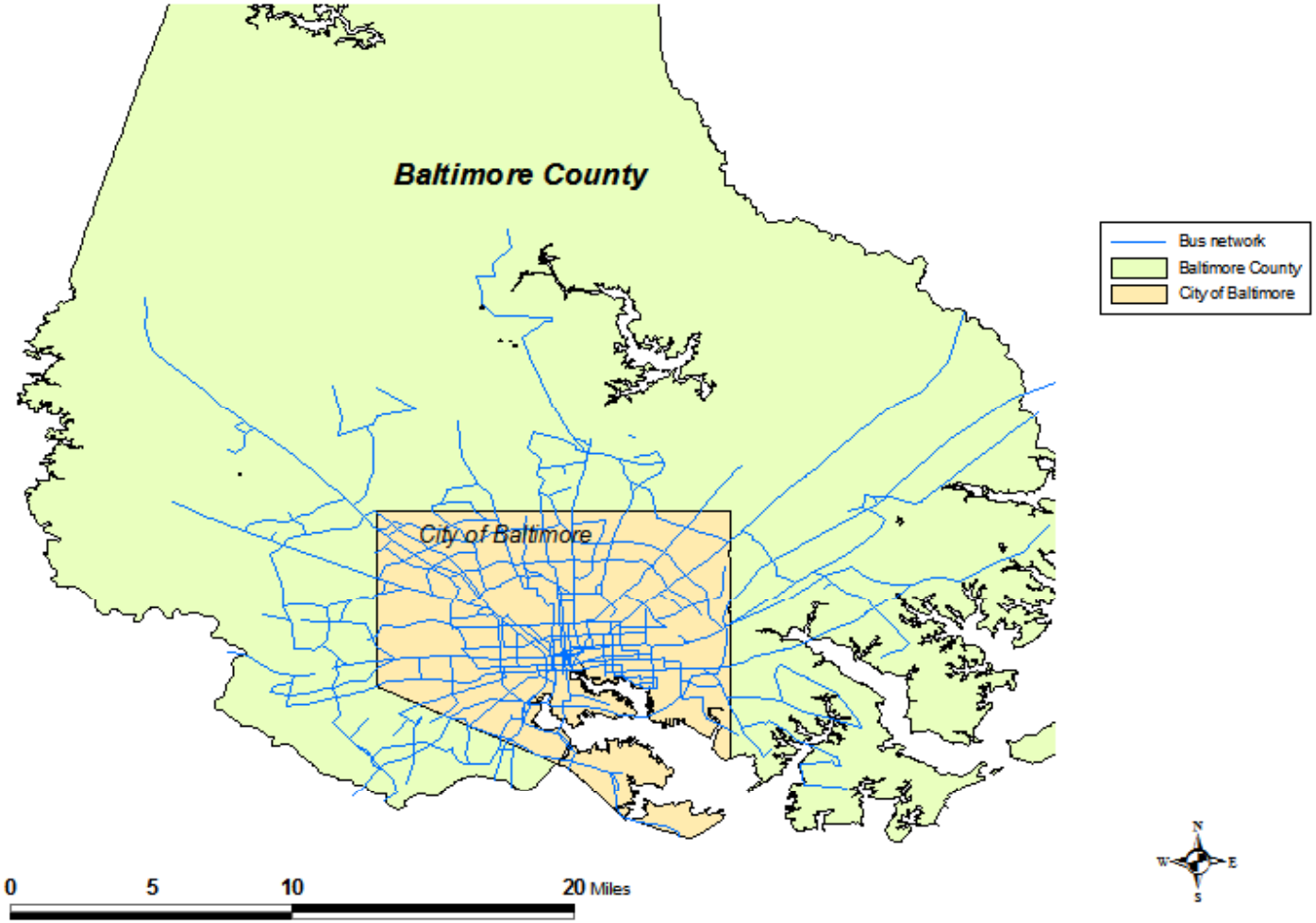


Figure 30.5:

Baltimore Intra-urban Rail Network: 2004
Intra-urban Lines and Rail Stations

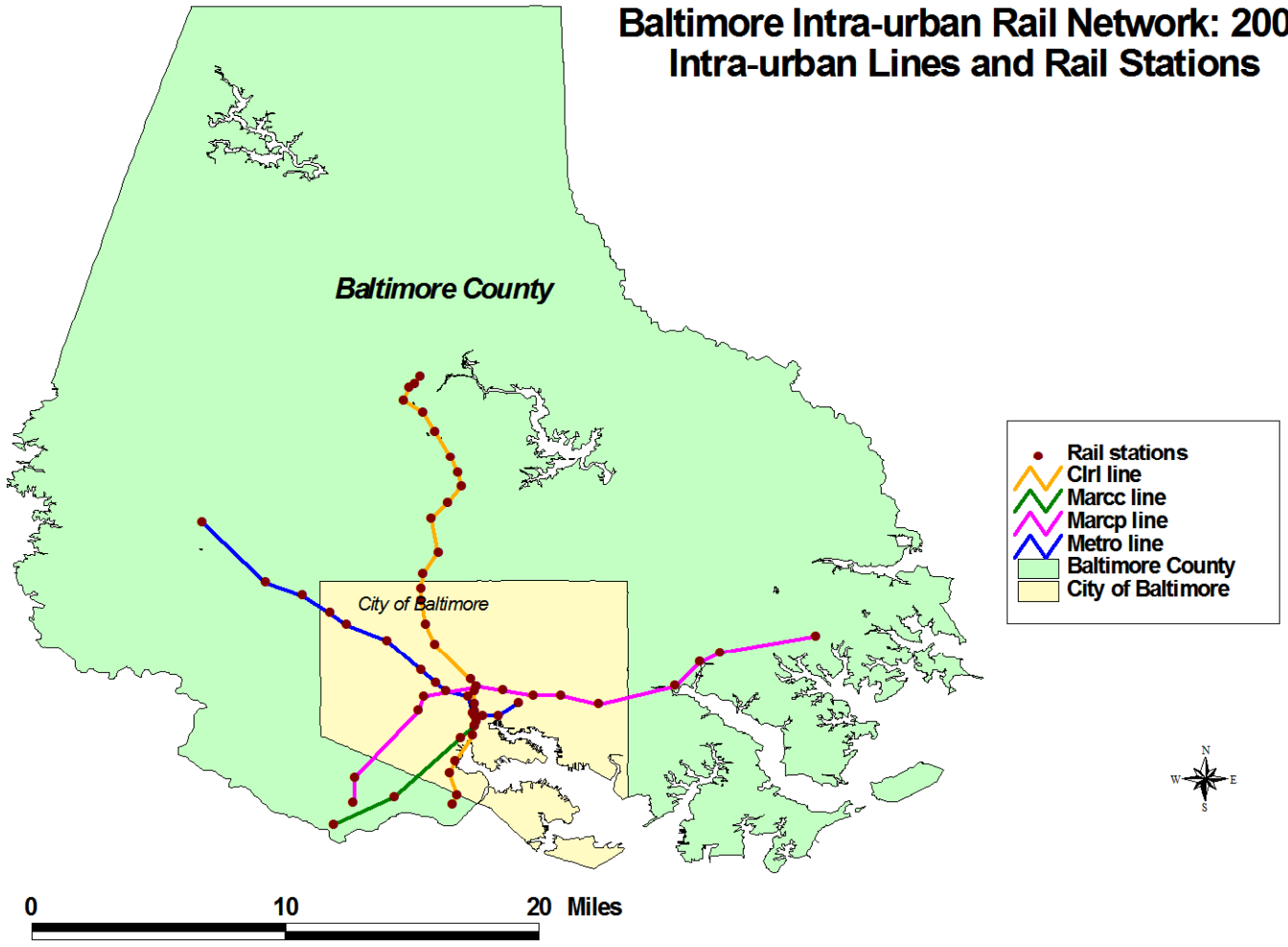
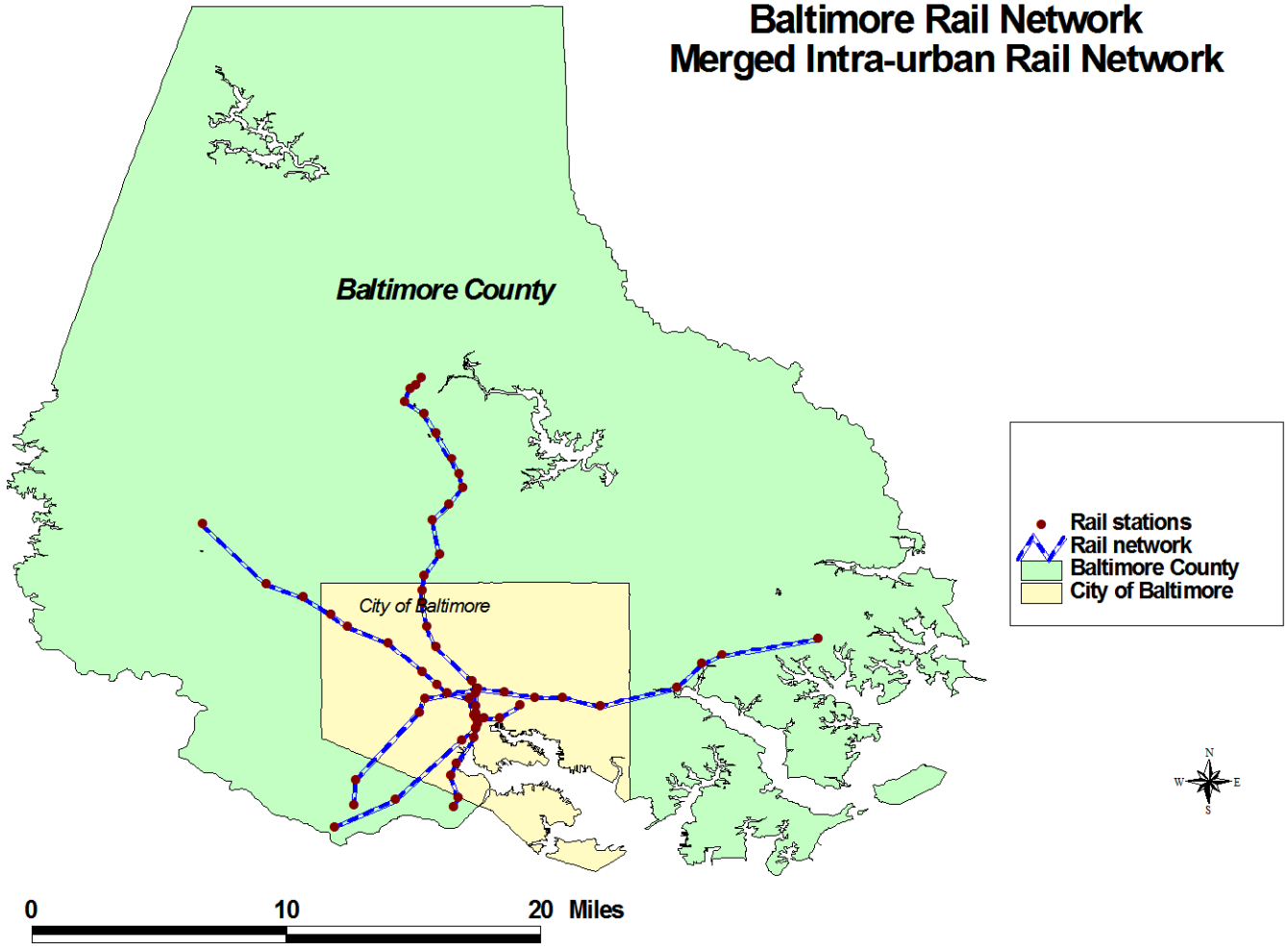


Figure 30.6:

Baltimore Rail Network
Merged Intra-urban Rail Network



zone to a destination zone), the route with the lowest overall impedance is selected. As mentioned, impedance can be defined in terms of distance, travel time, speed, or generalized cost.

There are a number of shortest path algorithms that have been developed (Sedgewick, 2002). They differ in terms of whether they are breadth-first (i.e., search all possibilities) or depth-first (i.e., go straight to the target) algorithms and whether they examine a one-to-many relationship (i.e., from a single origin node to many nodes) or a many-to-many relationship (All pairs; from each node to every other node).

The algorithm that is most commonly used for shortest path analysis of moderate-sized data sets (up to a million cases) is called *A**, which is pronounced "A-star" (Nilsson, 1980; Stout, 2000; Rabin 2000a, 2000b; Sedgewick, 2002). It is a one-to-many algorithm but is an improvement over another commonly-used algorithm called *Dijkstra* (Dijkstra, 1959). Therefore, I will start first by describing the Dijkstra algorithm before explaining the A* algorithm.

Dijkstra Algorithm

The Dijkstra algorithm is a one-to-many search strategy in which a shortest path from a single node to all other nodes is calculated. The routine is a breadth-first algorithm in that it searches all possible paths, but builds the path one segment at a time. Starting from an origin location (node), it identifies the node that is nearest to it **and** which has not already been identified on the shortest path. After each node has been identified to be on the shortest path, it is removed from the search possibilities. The algorithm proceeds until the shortest path to all nodes has been determined. In terms of a matrix of origin nodes (on the vertical) and destination nodes (on the horizontal - see figure 28.1 in Chapter 28), the search algorithm estimates the shortest path for any one row (i.e., from a particular origin to all destinations).

The algorithm can also be structured to find the shortest path between a particular origin node and a particular destination node. In this case, it will quit once the destination node has been identified on the shortest path. The algorithm can also be structured to find the shortest path from each origin node to each destination node. It does this one path at a time (e.g., it finds the shortest path from node A to all other nodes; then it finds the shortest path from node B to all other nodes; and so forth).

The network in Figure 30.1 will be used as an example. Figure 30.7 presents the network in terms of a particular origin node (A = Start) and a particular destination node (G = Finish). In the first step (not shown), the algorithm finds the node that is closest to A that has not already been put on the shortest path. In this case, it is to itself (i.e., A to A is the shortest path at this point). It thus removes A from the list of possible nodes and puts it in a shortest path

Figure 30.7:

Example of Dijkstra Algorithm

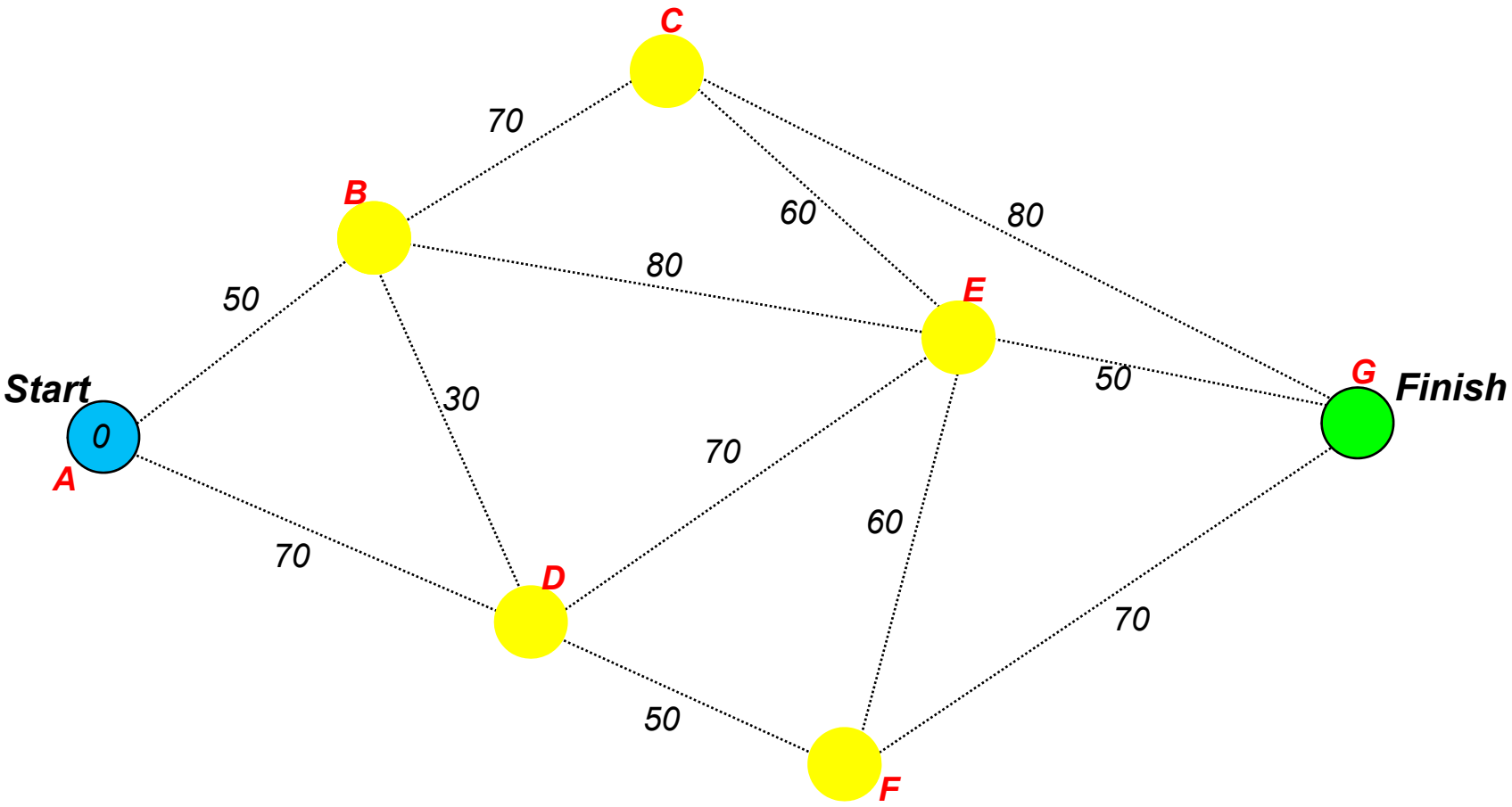
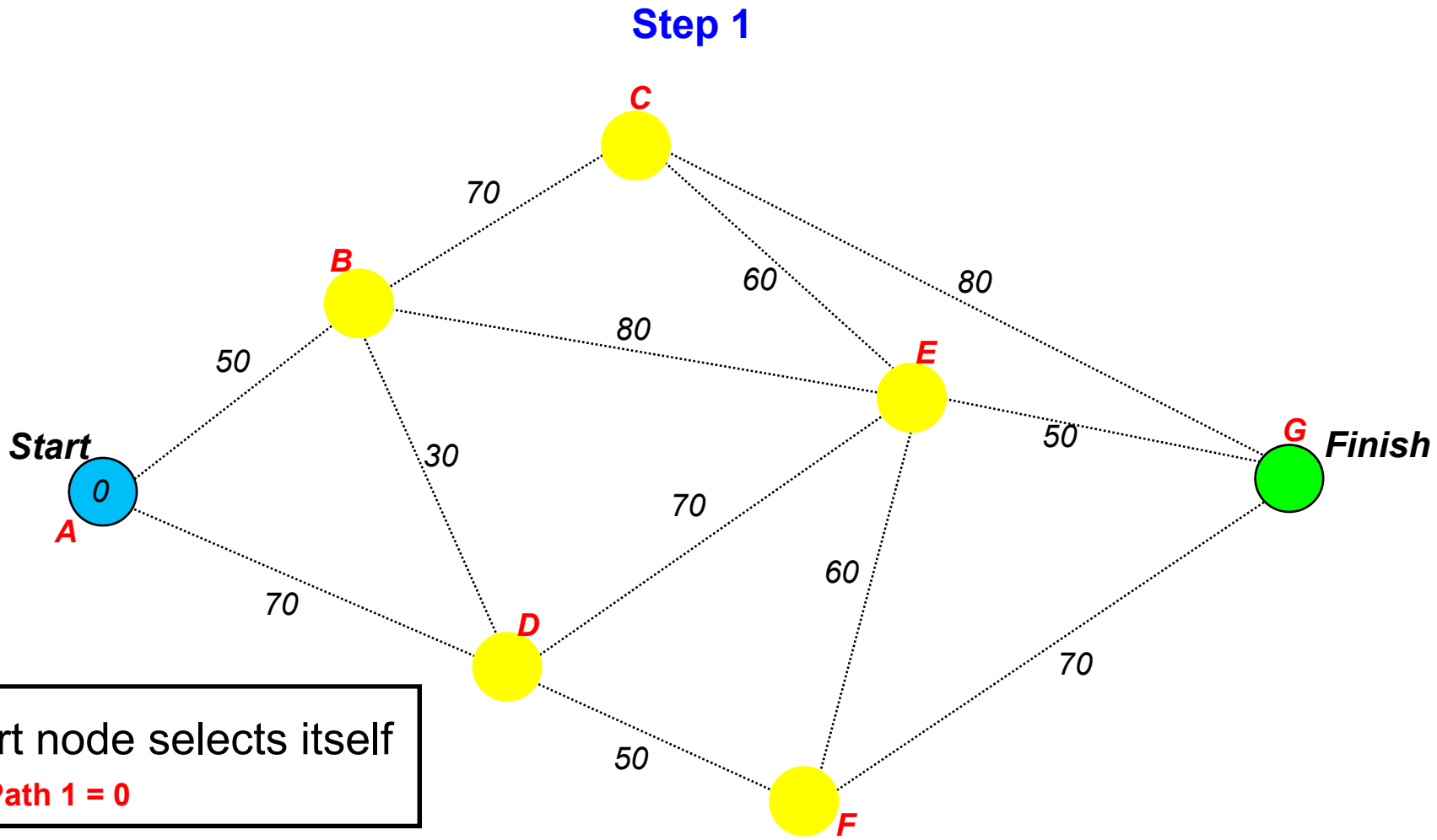


Figure 30.8:

Example of Dijkstra Algorithm



node list. Next, the routine finds the node that is closest to A that has not already been put on the shortest path list. This will be node B, which is 50 units distance from A (Figure 30.8). Thus, the shortest path now goes from A to B. Subsequently, node B is removed from the list of possible new nodes and is put on the shortest path list.

In step 2, the routine finds the node that is closest to one of the existing nodes on the shortest path list but which has not already been put on that list. This will be node D, which is 70 units from A (Figure 30.9). That is, if A and B have already been put on the shortest path list, then only two nodes are connected to these - C and D. The distance from A to C is 120 (50 + 70) while the distance from A to D is 70. Thus, the routine selects node D next. Subsequently, node D is removed from the list of possible new nodes and is put on the shortest path list.

In step 3 (Figure 30.10), the routine determines the node that is closest to A and which has not yet been put on the shortest path. There are two possibilities - C and F; both are 120 units distance from A. In the case of a tie, the routine 'flips a coin' and chooses one, in this case node F. Subsequently, node F is removed from the list of possible new nodes and is put on the shortest path list.

In step 4 (Figure 30.11), the routine adds node C to the shortest path. Note that had the 'coin flip' in step 3 chosen node C instead of F, in this stage node F would have been selected; thus, the routine produces the same solution, just in a different order. Both nodes C and F are 120 units distance from node A. Node C is now removed the list of possible new nodes and is put on the shortest path list.

In step 5 (Figure 30.12), the routine adds node E to the shortest path list because the distance to E through B is shorter than any other route that has not yet been determined (130 units from A). Notice that the path to E through C or D would have been longer than through B (180 and 140 units respectively).

Finally, in step 6 (Figure 30.13), the routine goes to the finish, node G. The path through B and E is shorter than by any other path to G (180 total units). Thus, the Dijkstra algorithm has searched every node in the network and determined a shortest path from node A to each of them (Figure 30.14). Even though we are only interested in the path from A to G, the algorithm solves all shortest paths from A to all nodes.

A* Algorithm

The biggest problem with the Dijkstra algorithm is that it searches the path to every single node. If the purpose were to find the shortest path from a single node to all other nodes,

Figure 30.9:

Example of Dijkstra Algorithm

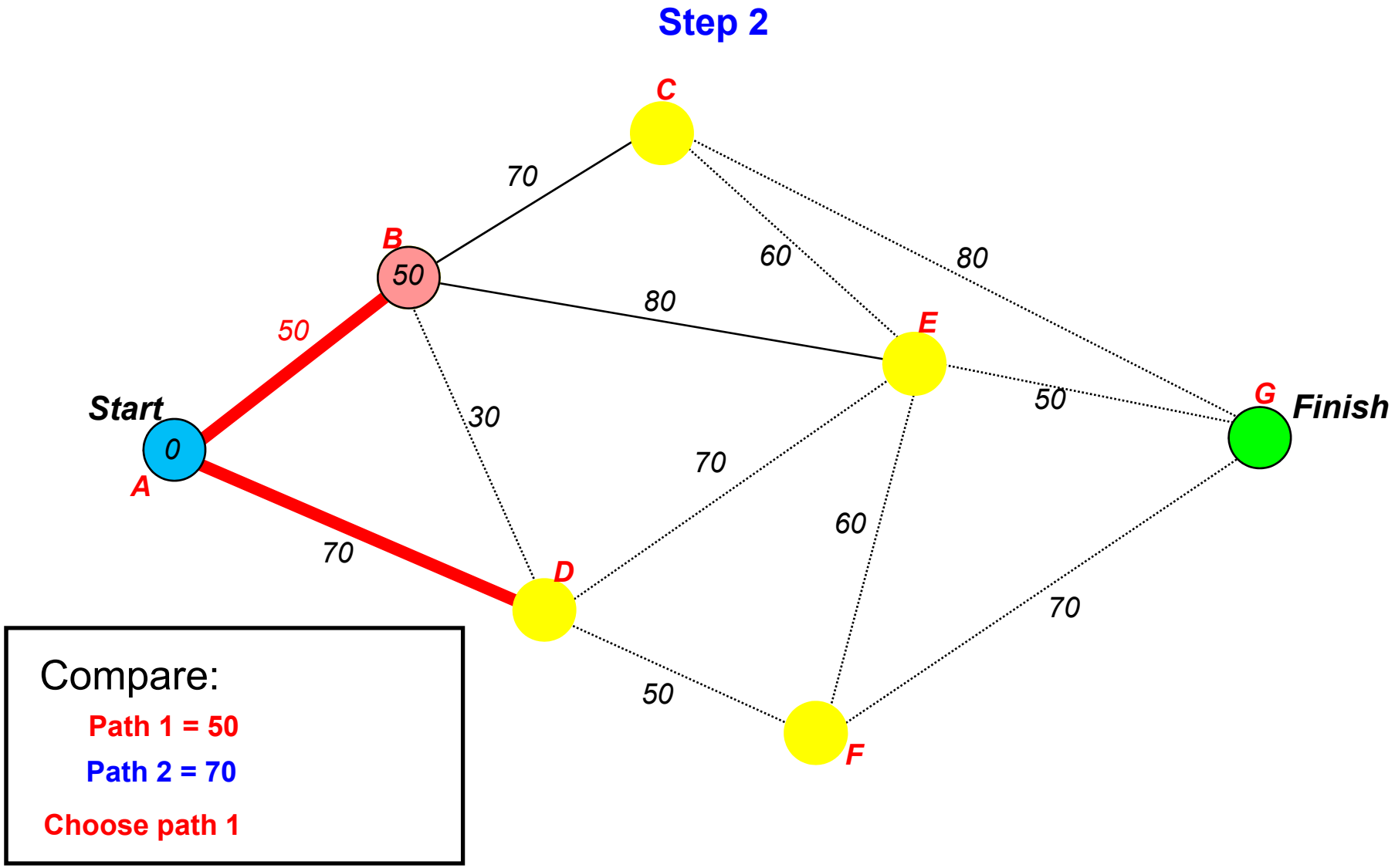


Figure 30.10:

Example of Dijkstra Algorithm

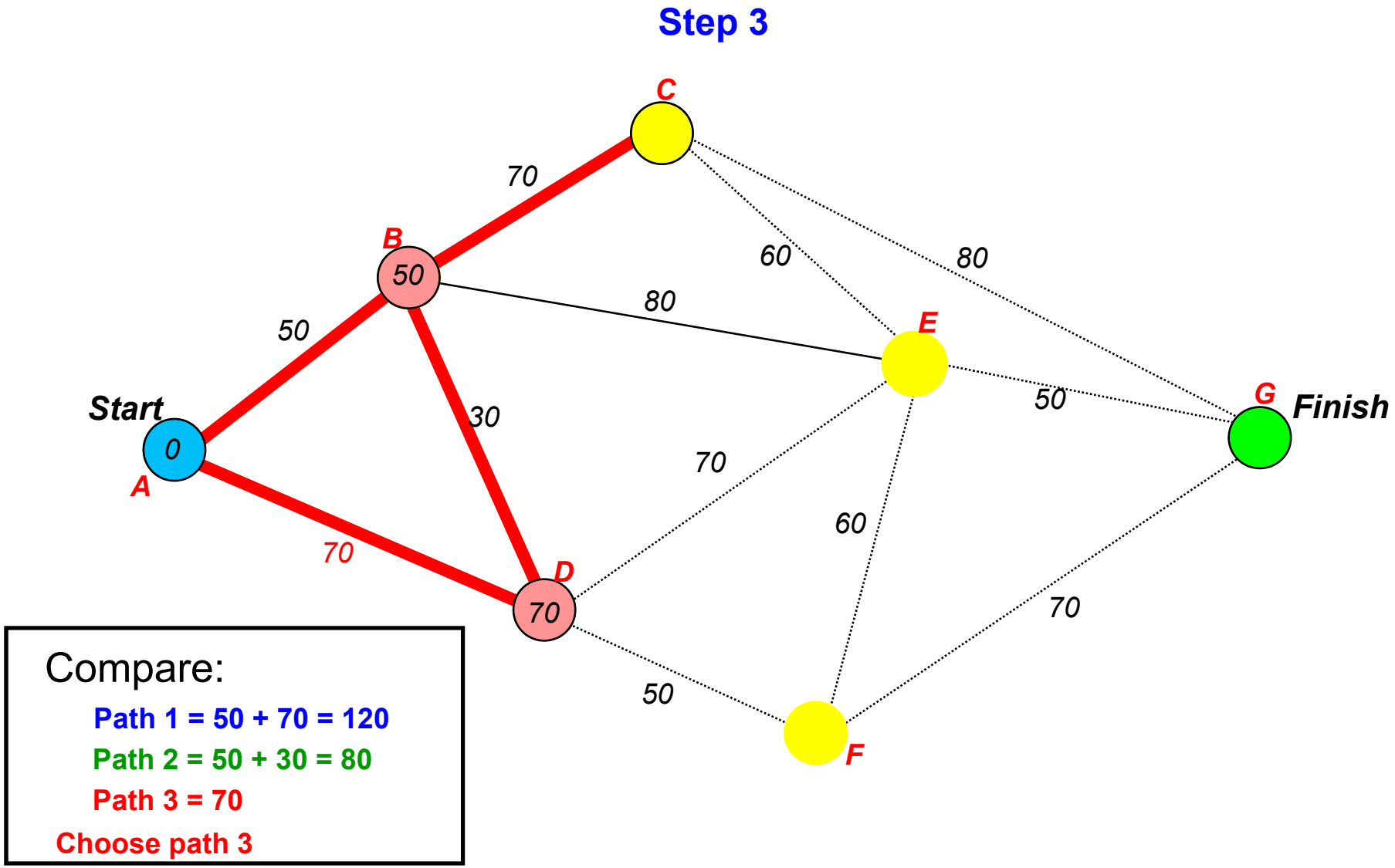


Figure 30.11:

Example of Dijkstra Algorithm

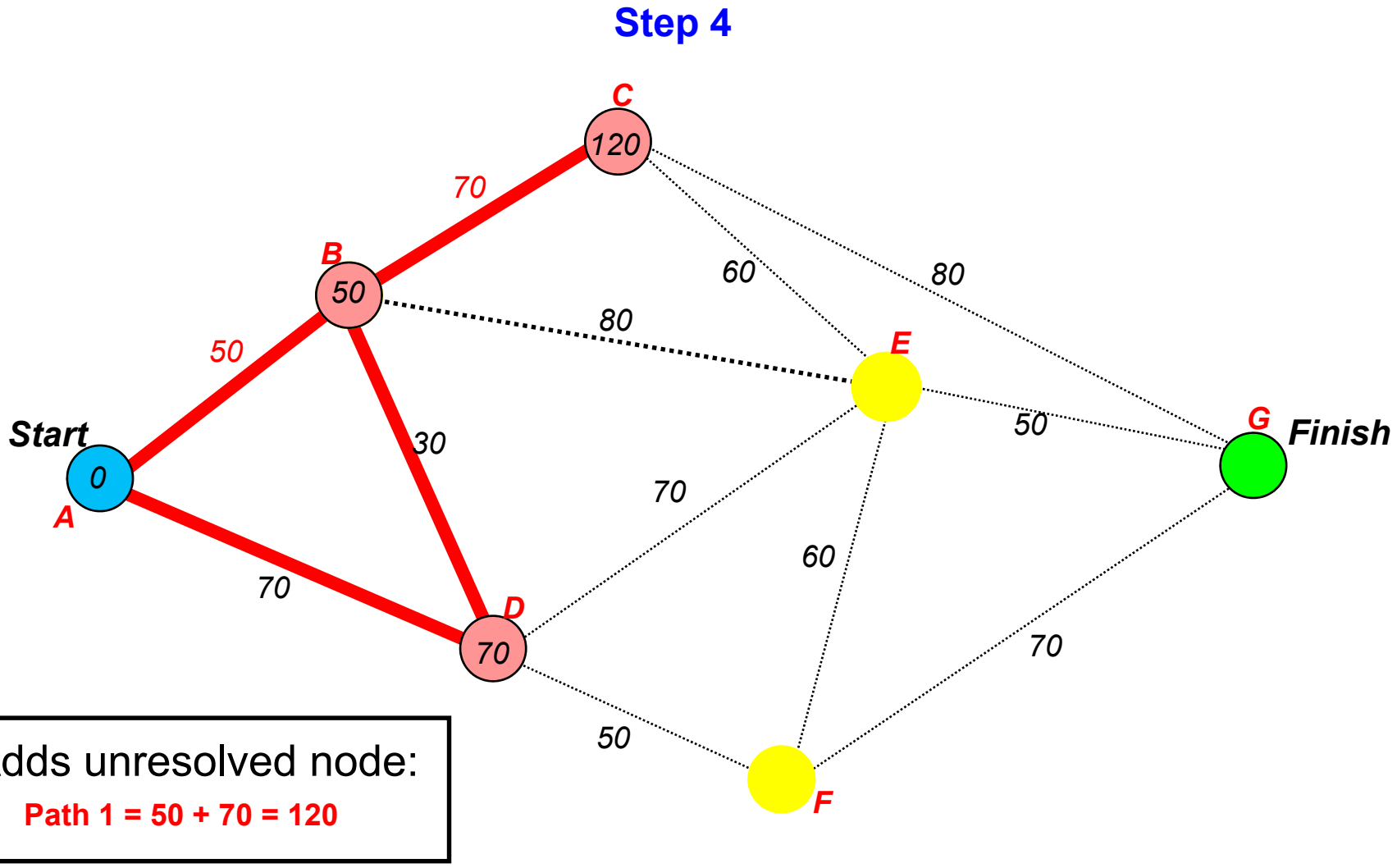


Figure 30.12:

Example of Dijkstra Algorithm

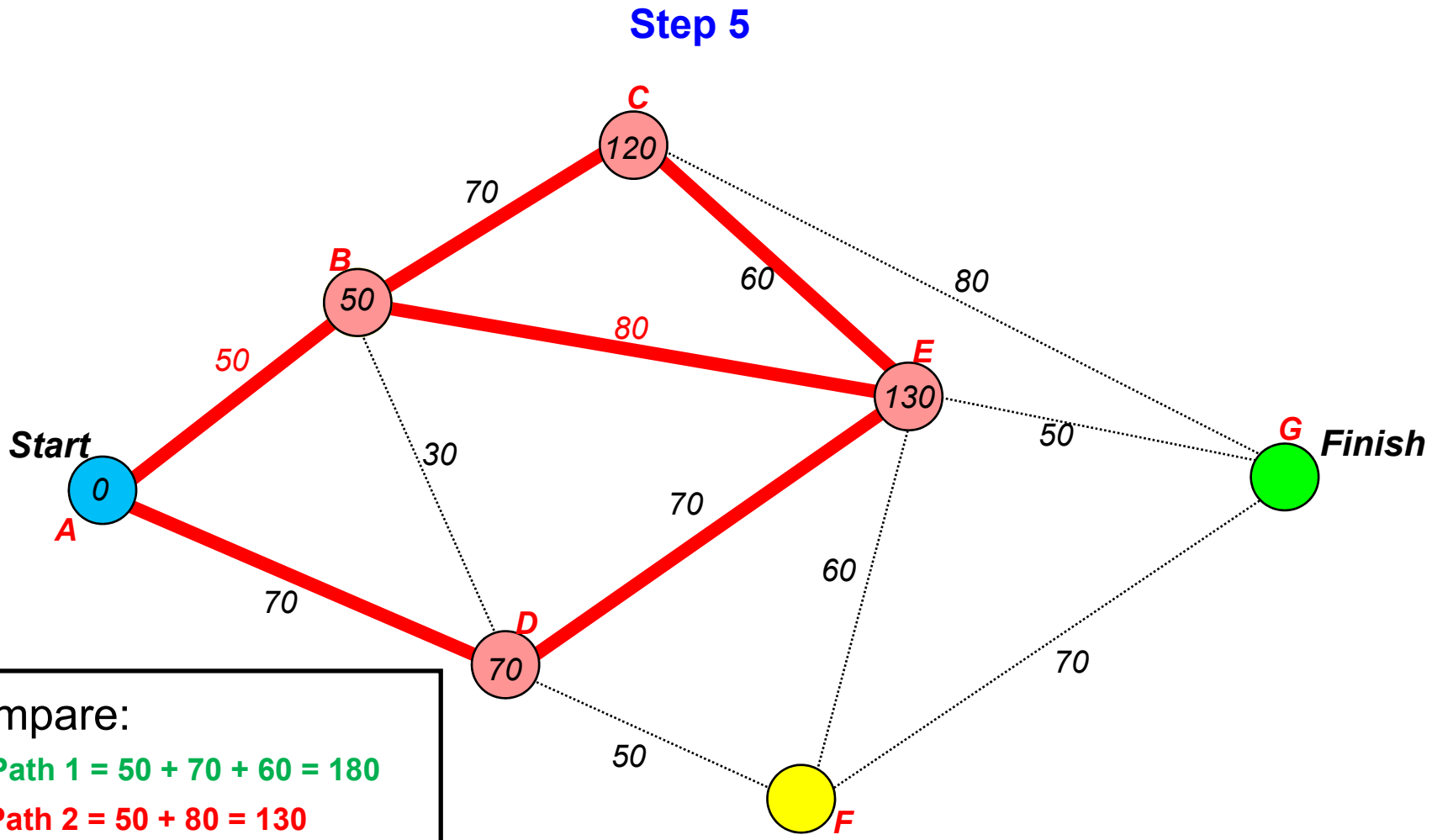


Figure 30.13:

Example of Dijkstra Algorithm

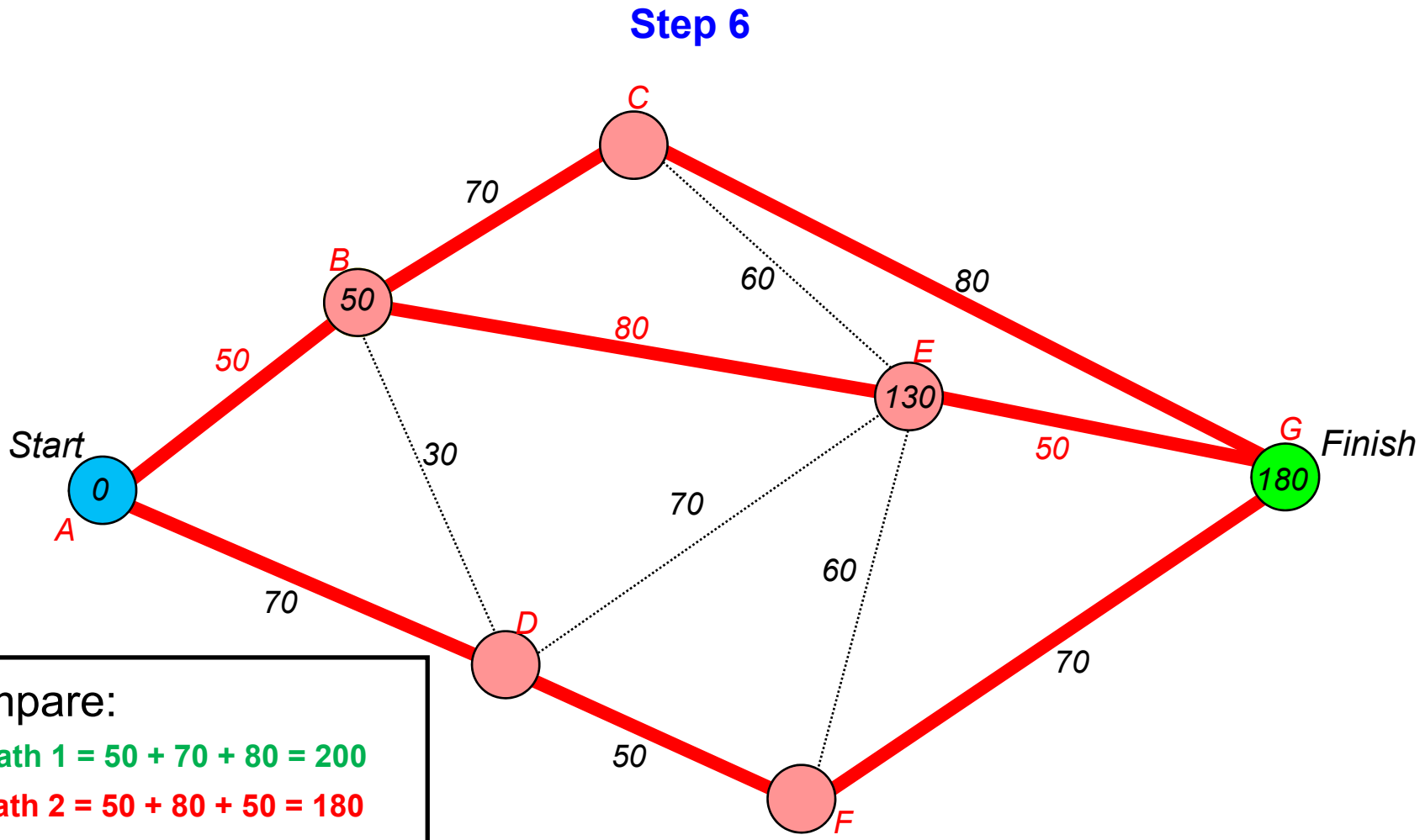
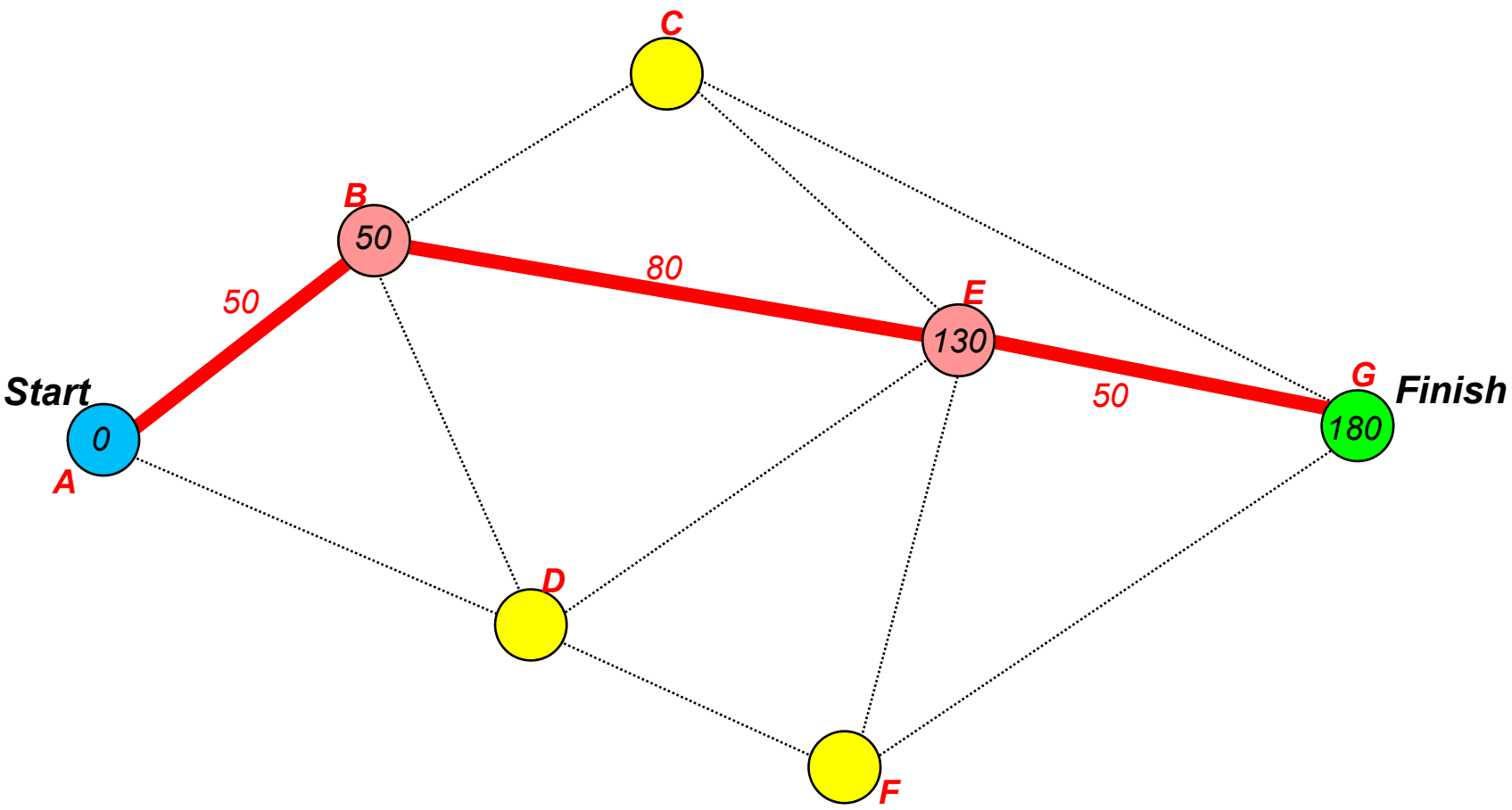


Figure 30.14:

Example of Dijkstra Algorithm



then this would produce the best solution. However, with an origin-destination matrix, we really want to know the distance between a pair of nodes (one origin and one destination). Consequently, the Dikjstra algorithm is very, very slow compared to what we need. It would be a lot quicker if we could find the distance from each origin-destination pair but quit the algorithm as soon as that distance has been determined.

This is where the A* algorithm comes in. A* was developed within the artificial intelligence research area as a means for developing a *heuristic* rule for solving a problem (Nilsson, 1980). In this case, the heuristic rule is the remaining distance from a solved node to the final destination. That is, at every step in the Dijkstra routine, an estimate is made of the remaining distance from each possible choice to the final destination. The node that is chosen for the shortest path is that which has the least total *combined* distance from the previously determined node to the final goal. Thus, for any step, if D_{i1} is the distance to a node, i, which has not already been put on the shortest path and D_{i2} is an estimate of the distance from that node to the final destination, the estimated total distance for that node is:

$$d_i = d_{i1} + d_{i2} \tag{30.1}$$

Of all the nodes that could be chosen, the node, i, which has the shortest total distance is selected next for the shortest path. There are two caveats to this statement. First, the node, i, cannot have already been selected for the shortest path; this is just re-stating the rules by which we search for nodes which have not yet been put on the shortest path list. Second, the estimate of the remaining distance to the final destination must be less than or equal to the actual distance to the final destination. In other words, the estimated distance, D_{i2}, cannot be an overestimate (Nilsson, 1980). However, the closer the estimated distance is to the real distance, the more efficient will be the search.

How then do we determine a reasonable estimate for D_{i2}? The answer is a straight line from the possible node to the final destination since the shortest distance between two points is a straight line (or, on a sphere, a Great Circle distance since the shortest distance between two points is an arc). If we simply calculate the straight-line (or straight arc if spherical distance is being used) from the node that we are exploring to the final node, then the heuristic will work.

Figure 30.15 displays the example network again. Like the Dijkstra algorithm, the routine first finds a node closest to A, which is itself. Next, it finds a node that has the least total distance from A to the final destination, G (Figure 30.16). There are two possibilities, go through B or go through D. The distance from A to B is 50 and the remaining distance from B to G is 130. Thus, the total distance through B would be 180. On the other hand, the distance from A to D is 70 and the remaining distance from D to G is 120. Thus, the total distance through D would be 190. Since 180 is smaller than 190, we choose node B.

Figure 30.15:

A* Modifies the Dijkstra Algorithm

Adding an Estimate of the Remaining Distance to the Dijkstra Distance

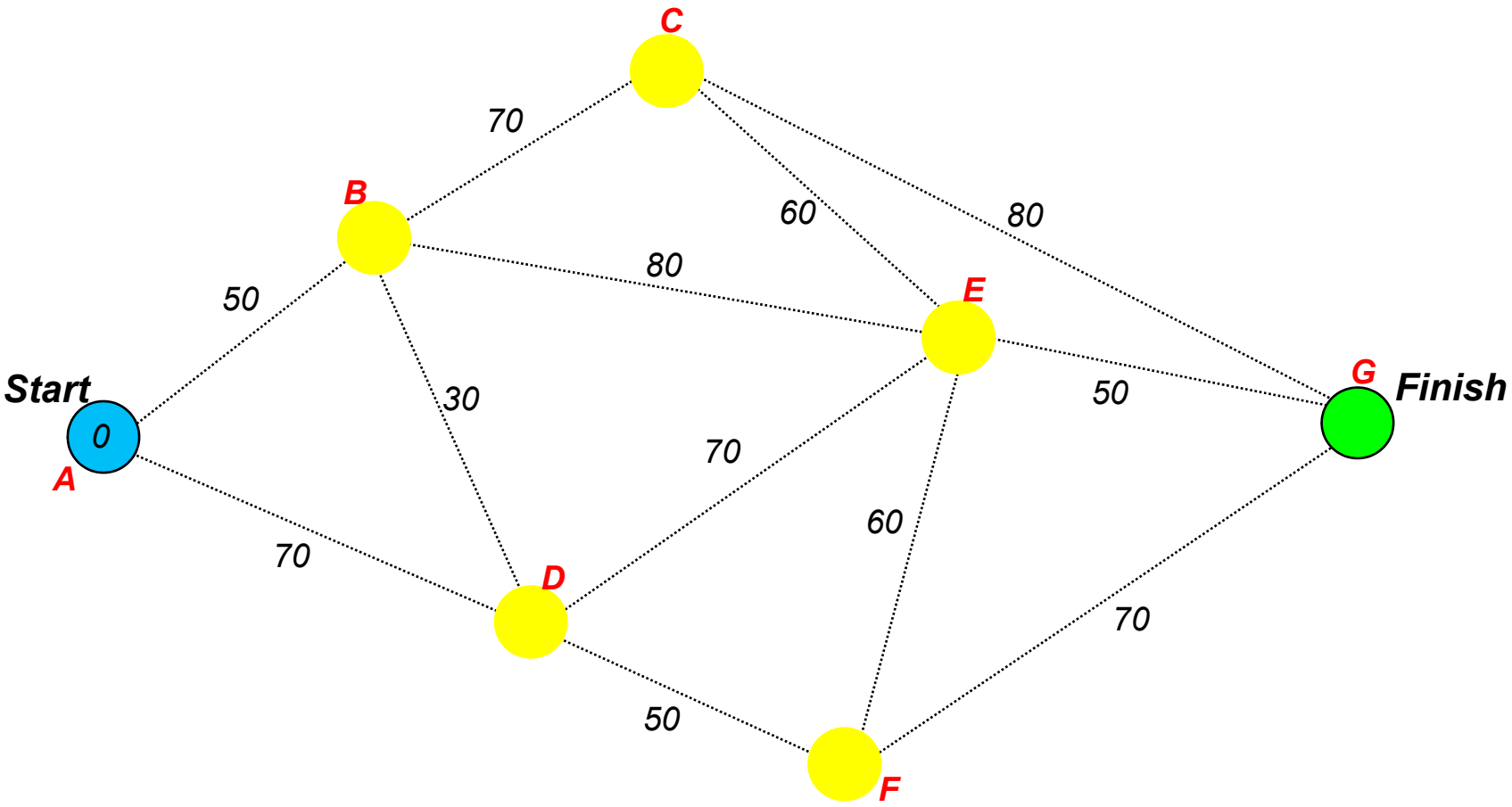
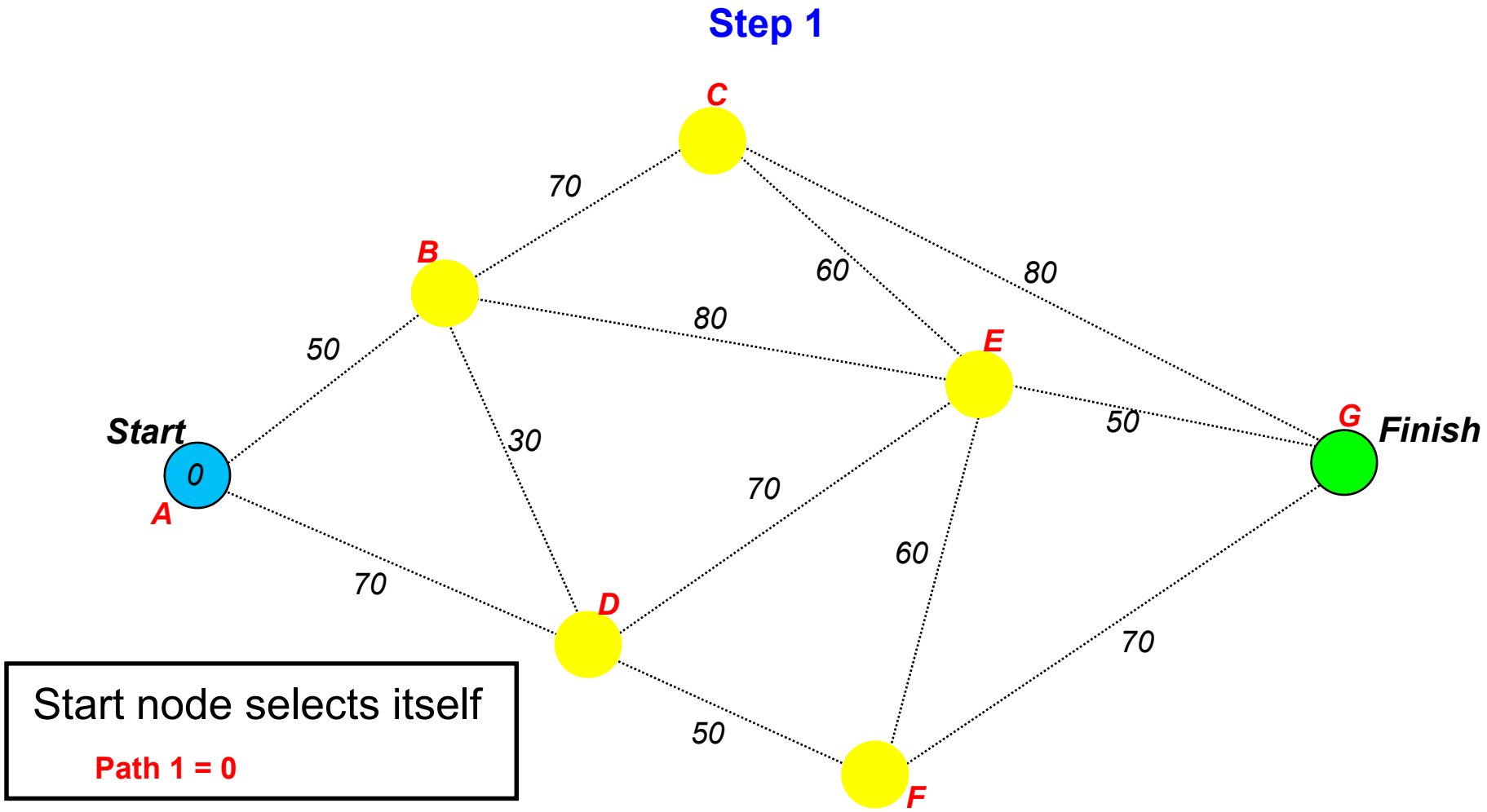


Figure 30.16:

A* Algorithm



In step 2 (Figure 30.17), three possibilities are explored for reaching G from A - through B and E; through B and C; and through D. The total distance through B and E is 180 (50 + 80 + 50) while the total distance through B and C is 200 (50 + 70 + 80) and through D is 190 (70 + 120). Thus, the routine chooses through B and E.

In step 3 (Figure 30.18), it is but a short path from E to the final destination G. The total distance through B and E to G is 180 while the total distance through B and C is 200 and through D is 190. Thus, the A* algorithm has determined a shortest path in three steps, rather than the 6 it took the Dijkstra algorithm (Figure 30.19).

In general, if V is the number of nodes in the network, the Dijkstra algorithm requires V^2 searches whereas the A* algorithm requires only V searches (Sedgewick, 2002). As can be seen, this is much more efficient than having to search every single possible node, which is what Dijkstra requires.

Applying A to multiple origins*

As with the Dijkstra algorithm, A* can be applied to multiple origins. It does it one origin-destination combination at a time. If an origin-destination matrix is represented by the origins as rows and the destinations as columns, then the A* algorithm takes each origin-destination combination and finds the shortest path. Since it does not search all possible nodes (only those in which the total distance to the destination is shortest), it cannot determine in one step the distance from an origin to all possible destinations. However, it is so quick as an algorithm that it can be applied to each cell of the origin-destination matrix and still come out much faster than a Dijkstra search.

Weighting of Segments

As mentioned above, the units of the network can be any type of impedance - distance, travel time, or cost. These can be thought of as *weights* applied to a segment. The A* algorithm does not really care what are the units of the segments as long as they are consistent and proportional to cost. The algorithm will determine the path with the shortest total cost (or total weight).

Thus, this algorithm can be applied to a trip distribution or mode split matrix of origin-destination pairs. It will determine the shortest path from each origin zone to each destination zone and can do this in the measurement units that are selected for weighting.

The advantages for travel demand modeling are enormous. It means that if the weighting variable is travel time, then the algorithm will find the shortest time path through the

Figure 30.17:

A* Algorithm

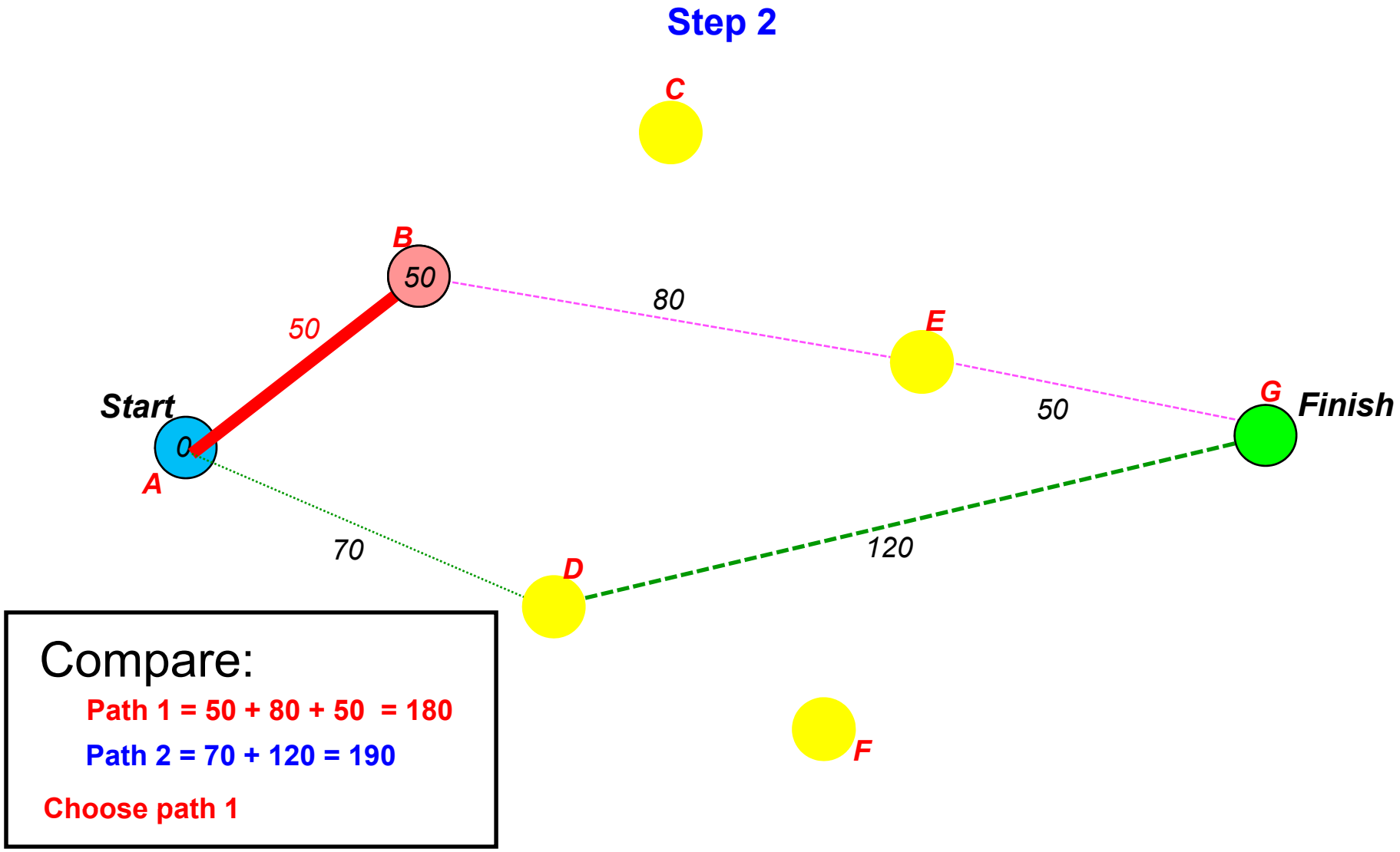


Figure 30.18:

A* Algorithm

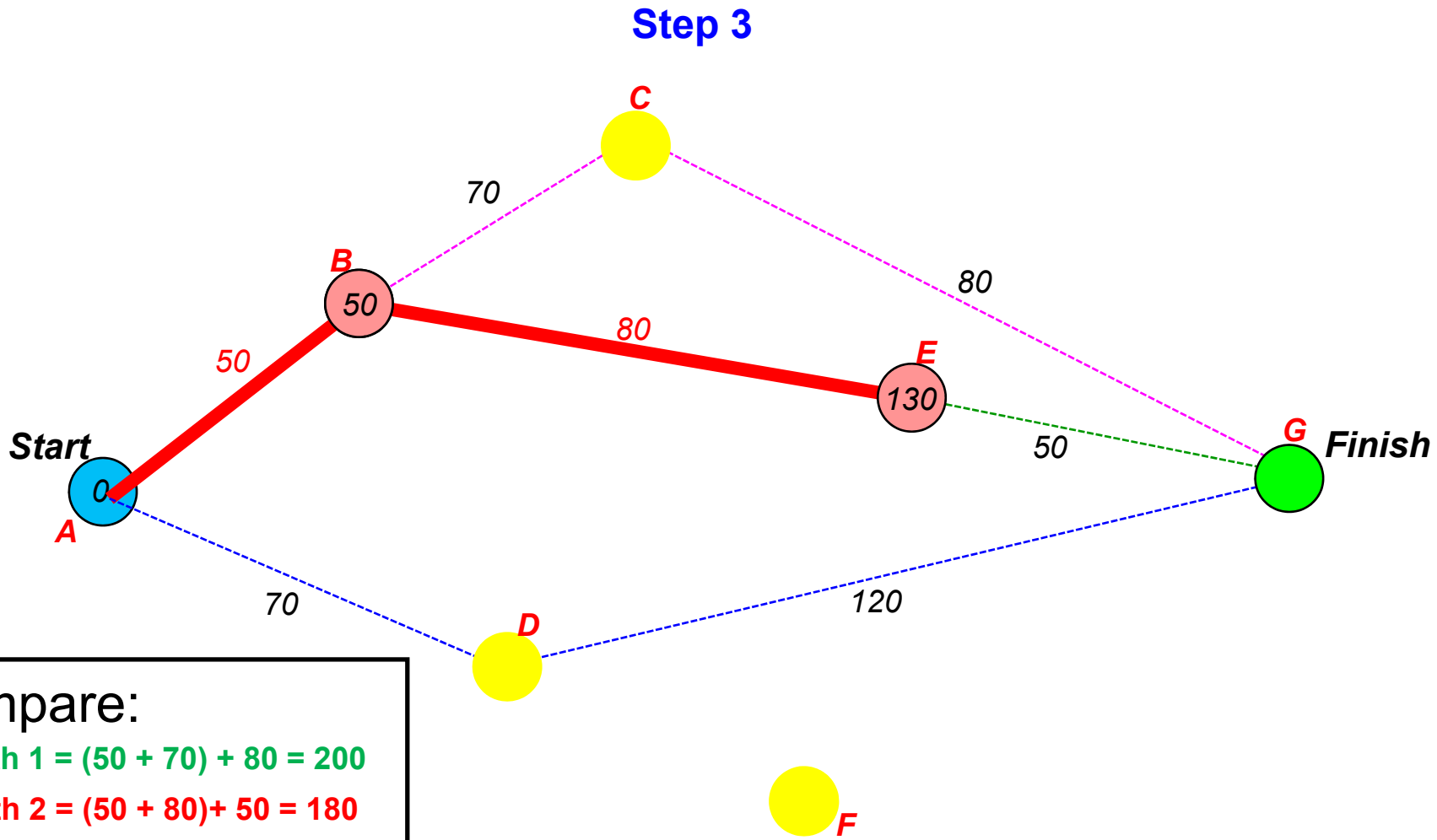
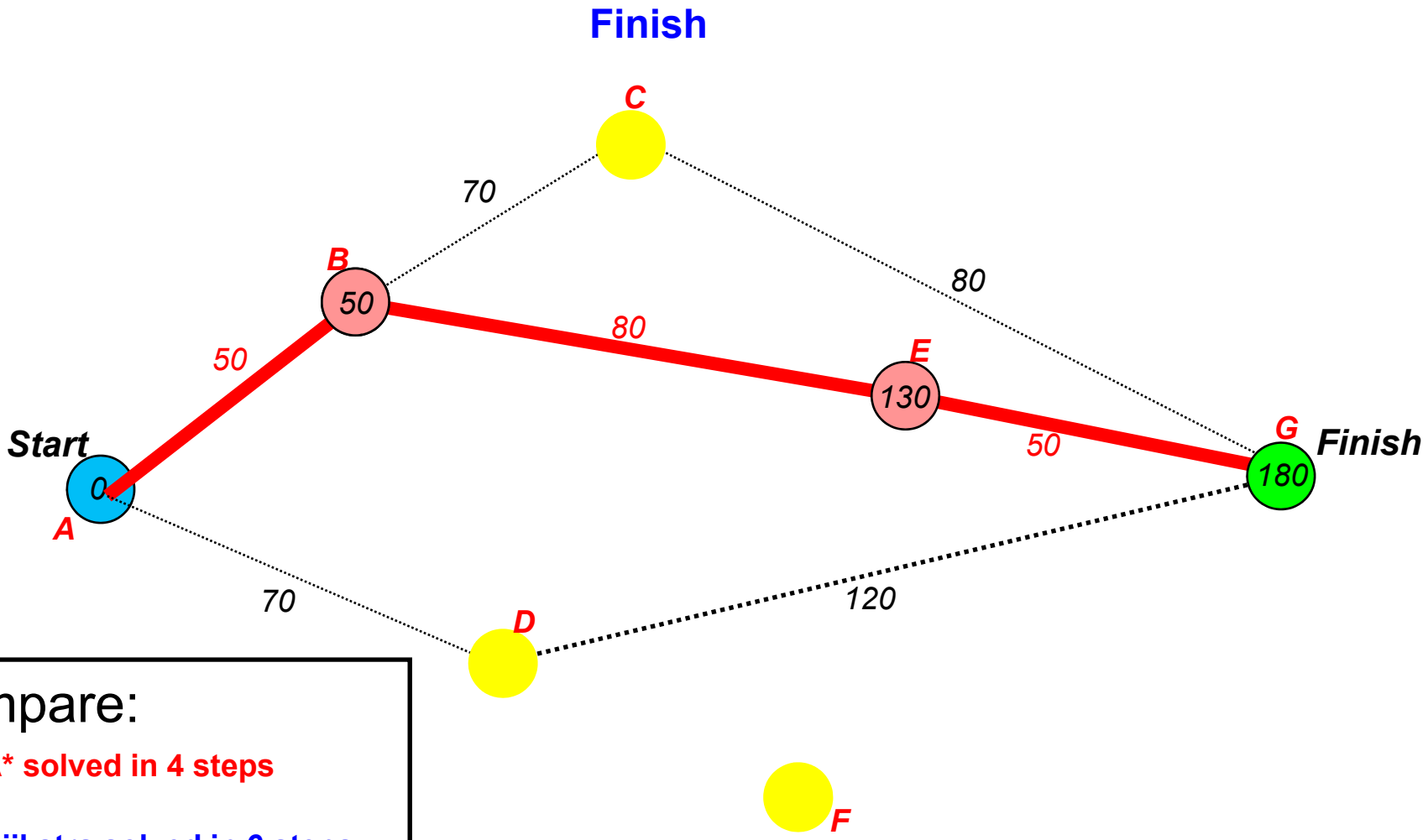


Figure 30.19:

A* Algorithm



Compare:

A* solved in 4 steps

Dijkstra solved in 6 steps

network for each origin-destination pair. If the weighting variable is generalized cost, then the algorithm will find the shortest cost path through the network. Finally, if the weighting variable is speed, then this must be converted into an impedance weight by dividing the distance of the segment by the speed to yield travel time. In short, the A* algorithm is an amazing one and allows the building of a routing algorithm.[2]

Routing Algorithms

In applying a shortest path analysis to a network assignment, several assumptions have to be made. As mentioned earlier, network assignment involves assigning trips to particular routes. Given a network of segments (e.g., road segments, train segments), a *routing algorithm* allocates the predicted number of trips to one or more routes. In other words, the network assignment is done through a routing algorithm. What makes this complex is that there are a number of different routing algorithms of which a shortest path is only one. Most of them are based on the assumption of travel cost relative to network capacity (Ortuzar & Willumsen, 2001).

The simplest type of routing algorithm is an *All or None* assignment. For each origin-destination pair (either for all trips or trips by specific travel mode), the algorithm calculates the shortest path through the network and assigns *all* trips to that path. This is the most rational model in that the cost of travel (whether measured by distance, travel time, or some cost measure) is minimized.

A second routing algorithm is a *stochastic path* in which each route has a certain probability of being selected. Multiple paths can be selected, but with a probability inversely proportional to their cost. The shortest path will be selected most often; the second shortest path next most often; the third shortest path third most often; and so forth. This type of algorithm attempts to capture the variability in travel behavior that can come from traveler's perceptions or incomplete information about the choice of path.

A third routing algorithm is a *congested assignment* in which there is feedback from the capacity of the network to the choice of route. In the classic case, as travel volumes increase on network segments, the capacity of the segment to absorb traffic is approached. The higher the ratio of the volume-to-capacity (V/C), the slower traffic becomes on the segment. In other words, the cost of travel increases. Eventually, if the volume keeps increasing, the speed slows so much as to eventually reduce the amount of traffic that can enter the segment (ITE, 2010). In theory, if there is so much traffic volume relative to the capacity, traffic comes to a complete

[2] For larger databases greater than, say, 1 million records, however, A* is too slow. An algorithm that is appropriate for very large databases can be found in Shekhar and Chawla (2003).

halt (gridlock). However, in practice this does not happen as drivers take other routes. Consequently, with high V/C ratios, other routes become more desirable and some traffic spills over on to those segments. his type of model is frequently used in metropolitan travel demand models for transportation since congestion is a major factor in most urban areas.

There are advantages and disadvantages to each of these approaches. The "All or none" assignment is the closest to a rational choice model; the route with the lowest total cost is chosen. On the other hand, this algorithm will continue to assign trips to a route even if the road segment becomes extremely congested, which is not realistic. A stochastic model has the advantage of accounting for variability. If individual-level data could be obtained that measured individual choices and perceptions of routes, then it is possible that a realistic proportional split among routes could be detected. More often, however, such information is lacking and a variation on the mode split model is used to proportion the trips among the different possible routes (see Ortuzar & Willumsen, 2001, Chapter 10 for more information).

The "Congested assignment" algorithm can be seen as a more realistic variation on the "All or none" in that the costs of travel change as the network capacity is reached. Most transportation models use that type of model because it is a more realistic representation.

Lack of Information about Crime Trips

The problem with crime trips, however, is that the number of trips is liable to represent only a very small proportion of the total trips on any segment of a network. Thus, there is not liable to be any feedback from the capacity limits of segments to crime trips *per se*. Any feedback is liable to apply to all trips, of which the crime trips are a sub-set. It might be possible to link a crime trip route choice algorithm to a general congested assignment in order to approximate this situation, but the amount of information that would be necessary to be obtained and the complexity of the modeling algorithm would probably not produce much tangible benefits beyond what a simple model predicts.

Further, there could be feedback from surveillance and other policing practices that might increase the cost to an offender of traveling along a particular route. However, without any detailed information about perceived costs of particular routes, it is difficult to postulate any type of model for choosing alternatives. This would be a very valuable area of research in understanding the travel behavior of offenders. At the end of this chapter, there is a brief discussion of an article that modeled the likely escape routes taken by bank robbers in Baltimore County, MD (Levine, 2007).

But, short of that information, an "All or none" assignment routine is probably the easiest to implement for allocating the predicted crime trips to routes.

The CrimeStat Network Assignment Module

The *CrimeStat* network assignment routine implements an "All or none" assignment based on the A* shortest path algorithm. Figure 30.20 shows the setup page for network assignment. On the page, there is a network assignment routine and there are some network utilities.

Network Used

The first input that needs to be made is which network is to be used. The choices are the network specified on the Measurement parameters page (the default) or an alternative network.

Network on measurement parameters page

Check the 'Network on Measurement parameters page' box to use that network. All the parameters will have been defined for that setup (see Measurement parameters page).

Alternative network

If an alternative network is to be used, it must be defined. Check the 'Alternative network' box and click on the 'Parameters' button. Figure 30.21 shows the dialogue box for the alternative network.

> Note: if a network is also used on the Measurement Parameters page, then it must be defined there as well. *CrimeStat* will check whether that file exists; if it does not, the routine will stop and an error message will be issued. Therefore, if an alternative network is used, the user should probably change the distance measurement on the Measurement Parameters page to direct or indirect distance.

Type of network

Network files can be *bi-directiona*l (e.g., a TIGER file) or *single directional* (e.g., a transportation modeling file). In a bi-directional file, travel can be in either direction. In a single directional file, travel is only in one direction. Specify the type of network to be used.

Input file

The network file can either be a shape file (line, polyline, or polylineZ file) or another file, either dBase IV 'dbf', Microsoft Excel 'xls/xlsx', Microsoft Access 'mdb', Ascii 'dat', or an

Figure 30.20:

Network Assignment Module

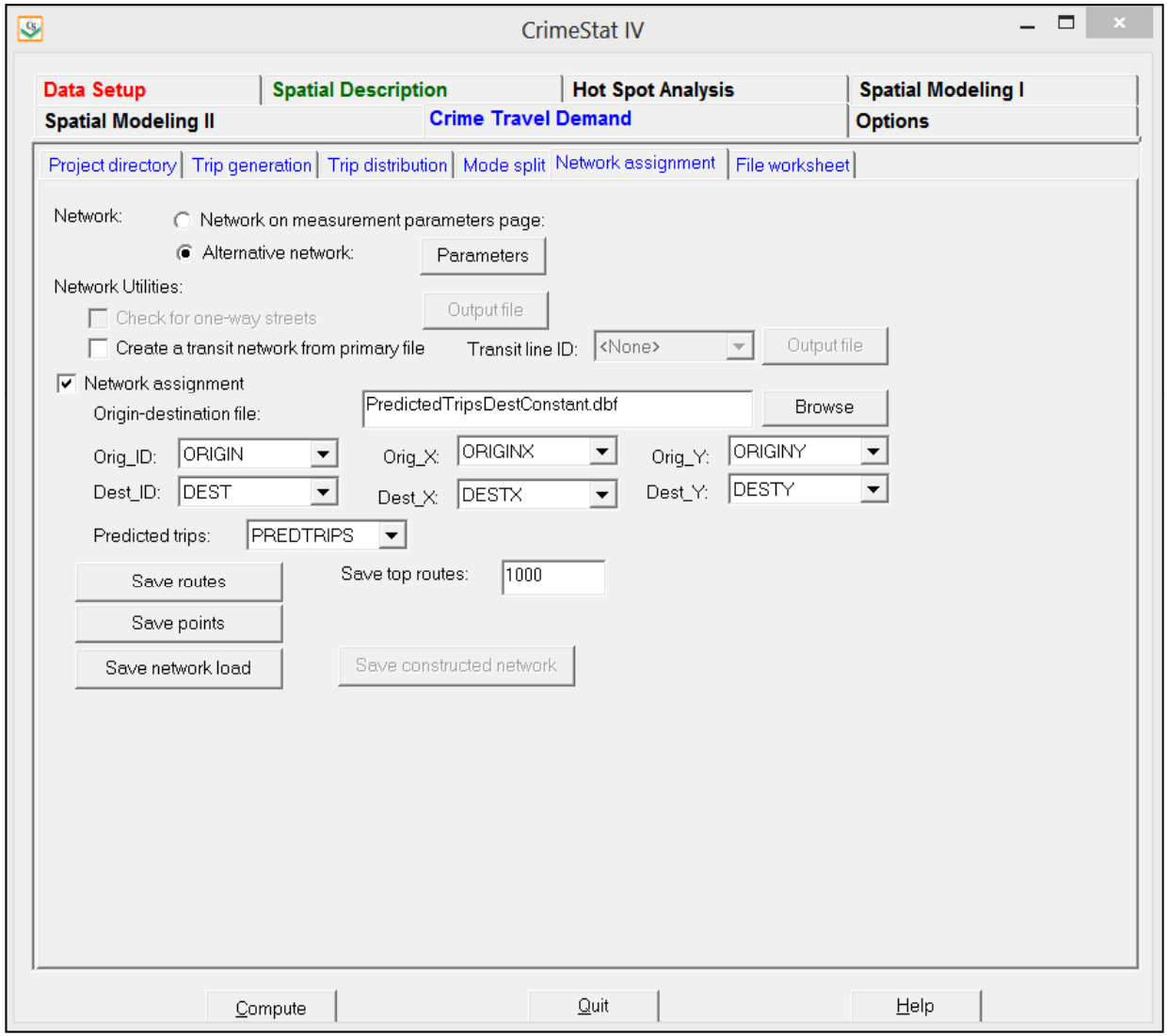
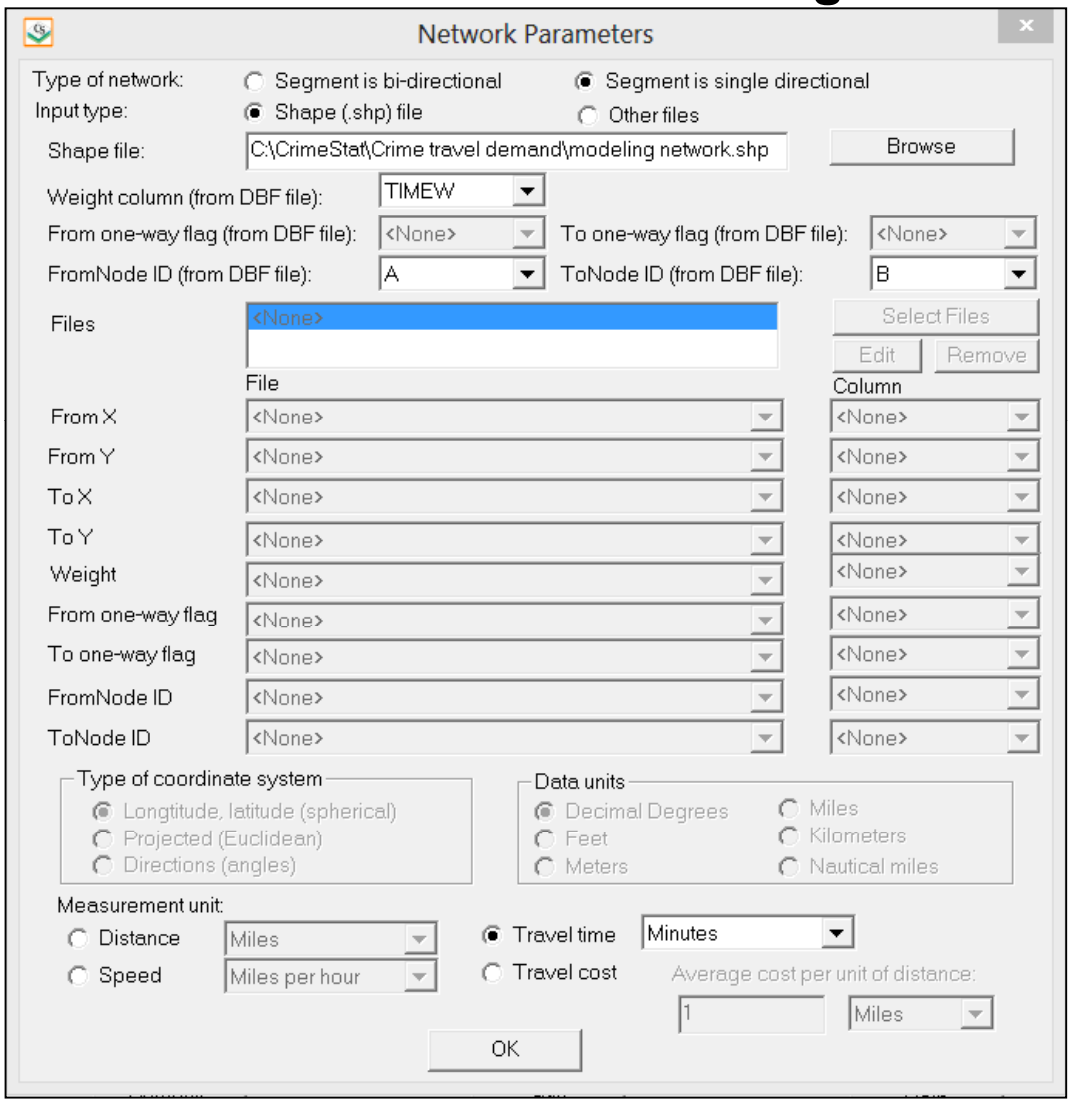


Figure 30.21:

Alternative Network Dialogue



ODBC-compliant file. The default is a shape file. If the file is a shape file, the routine will know the locations of the nodes. For a dBase IV, Excel or another file type, the X and Y coordinate variables of the end nodes must be defined. These are called the "From" node and the "End" node. An optional weight variable is allowed for both a shape or dbf file. The routine identifies nodes and segments and finds the shortest path. If there are one-way streets in a bi-directional file, the flag fields for the "From" and "To" nodes should be defined.

Weight field

By default, each segment in the network is not weighted. In this case, the routine calculates the shortest distance between two points using the distance of each segment. However, each segment can be weighted by travel time, speed or travel costs. If travel time is used for weighting the segment, the routine calculates the shortest time for any route between two points. If speed is used for weighting the segment, the routine converts this into travel time by dividing the distance by the speed. Finally, if travel cost is used for weighting the segment, the routine calculates the route with the smallest total travel cost. Specify the weighting field to be used and be sure to indicate the measurement units (distance, speed, travel time, or travel cost) at the bottom of the page. If there is no weighting field assigned, then the routine will calculate using distance.

From one-way flag and To one-way flag

One-way segments can be identified in a bi-directional file by a 'flag' field (it is not necessary in a single directional file). The 'flag' is a field for the end nodes of the segment with values of '0' and '1'. A '0' indicates that travel can pass through that node in either direction whereas a '1' indicates that travel can only pass from the other node of the same segment (i.e., travel cannot occur from another segment that is connected to the node). The default assumption is for travel to be allowed through each node (i.e., there is a '0' assumed for each node). There is a 'From one-way flag' field and a 'To one-way flag' field. For each one-way street, specify the flags for each end node. A '0' allows travel from any connecting segments whereas a '1' only allows travel from the other node of the same segment. Flag fields that are blank are assumed to allow travel to pass in either direction.

FromNode ID, ToNode ID

If the network is single directional, there are individual segments for each direction. Typically, two-way streets have two segments, one for each direction. On the other hand, one-way streets have only one segment. The FromNode ID and the ToNode ID identify from which end of the segment travel should occur. If no FromNode ID and ToNode ID is defined, the

routine will chose the first segment of a pair that it finds, whether travel is in the right or wrong direction. To identify correctly travel direction, define the FromNode and ToNode ID fields.

Type of coordinate system

The type of coordinate system for the network file is the same as for the primary file.

Measurement unit

By default, the shortest path is in terms of distance. However, each segment can be weighted by travel time, travel speed, or travel cost.

1. For travel time, the units are minutes, hours, or unspecified cost units. For speed, the units are miles per hour and kilometers per hour. In the case of speed as a weighting variable, it is automatically converted into travel time by dividing the distance of the segment by the speed, keeping units constant.

2. For travel cost, the units need to be defined as cost per unit distance (e.g., per mile, per kilometer). The routine will then indentify routes by those with the smallest total cost.

Network Utilities

There are two network utilities that can be used.

Check for one-way streets

First, there is a routine that will identify one-way streets *if* the network is single directional. In a single directional file, one-way streets do not have a reciprocal pair (i.e., a segment traveling in the opposite direction). This is indicated by a reciprocal pair of ID's for the “From” and “To” nodes. If checked, the routine identifies those segments that do not have reciprocal node ID's. The network is saved with a new field called “**Oneway**”. One-way segments are assigned a value of ‘1’ value and two-way segments are assigned a value of ‘0’. The output is saved as an *ArcGIS* '.shp', *MapInfo* '.mif' or various *Ascii* file types.

Create a transit network from primary file

Second, there is a routine that will create a transit network from the primary file. This is useful for creating a transit network from a collection of bus stops (bus network) or rail stations (rail network). If checked, the routine will read the primary file and will draw lines from one

point to another *in the order* in which the points appear in the primary file. Note, it is essential to order the points in the same order in which the network should be drawn (otherwise, an illogical network will be obtained). It is easy to do this in a spreadsheet program.

Transit Line ID

The routine can handle multiple lines, for example different rail lines or bus routes (e.g., Line A, Line B, Route 1, Route 2). In the primary file, the points must be grouped by lines, however, and must be classified by a Transit Line ID field. Within each group, the points must be arranged in order of occurrence; the routine will draw lines from one point to another in that order. In the Transit Line ID field, indicate which variable is the classification variable. The output is saved as an *ArcGIS* '.shp', *MapInfo* '.mif' or various *Ascii* file types.

Figure 30.5 above showed the effect of creating four separate rail lines from the station locations while Figure 30.6 showed the merged four lines implemented with the Group ID.

Network Output

There are three types of output for the network assignment routine. First, the most frequent inter-zonal routes (i.e., trips between different zones) can be output as polylines. Second, the most frequent intra-zonal routes (i.e., trips within the same zone) can be output as points. Third, the entire network can be output in terms of the total number of trips that occur on each segment (*network load*).

Save inter-zonal routes

The shortest routes can be saved as separate **polyline** objects for use in a GIS. Specify the output file format (*ArcGIS* '.shp', *MapInfo* '.mif' or various *Ascii* file types) and the file name.

Save top inter-zonal routes

Because the output file is very large (number of origin zones x number of destination zones), the user can select a zone-to-zone route with the most predicted trips. The default is the top 100 origin-destination combinations. Each output object is a line from the origin zone to the destination zone with a Route prefix. The prefix is placed before the output file name. The graphical output includes:

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (ROUTE)
3. The origin zone (ORIGIN)

4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of trips on that particular route (FREQ)
10. The distance between the origin zone and the destination zone (DIST).

Figure 30.22 shows the top 300 routes calculated with the modeling network. The assignment was weighted by travel time and the thickness and color of the line is proportional to the number of predicted trips.

To see how this differs from the trip distribution matrix, Figure 30.23 zooms into a high volume route in eastern Baltimore County. The modeling streets are displayed as are the predicted links from the trip distribution for that area. As seen, the trip distribution simply produces straight-line links between origins and destinations. In this case, the crime trips come into to the centroid of the Traffic Analysis Zone (TAZ) in the middle of this hot spot of crimes (TAZ 610). The actual routes, on the other hand, follow the streets (in this case, the modeling network) and are more circuitous. Several of the streets are used much more heavily than others, according to the assignment.

An additional point should be noted, however. Since the modeling network was used rather than the TIGER network, the trips into and from the centroid of the TAZ do not follow any particular road; the algorithm simply draws a straight line from the centroid to the nearest road segment. In subsequent modeling, it might be worthwhile to digitize additional streets in this neighborhood since there are many crimes being attracted to it. A crime mapping analyst can easily add the additional features to improve resolution. The model would have to re-run, however, to get a more accurate display.

Save intra-zonal routes

Intra-zonal routes (trips in which the origin and destination are the same zone) can be output as separate **point** objects as an *ArcGIS* '.shp', *MapInfo* '.mif' or various *Ascii* file types. Again, the top K points are output (default=100). Each output object is a point representing an intra-zonal trip with a RoutePoints. The prefix is placed before the output file name.

The graphical output for each includes:

1. An ID number from 1 to K, where K is the number of links output (ID)
2. The feature prefix (ROUTEPoints)

Figure 30.22:

Predicted Baltimore County Crime Trips: 1993-1997

Routes and LInks for Zone-to-zone Trips: All Crimes

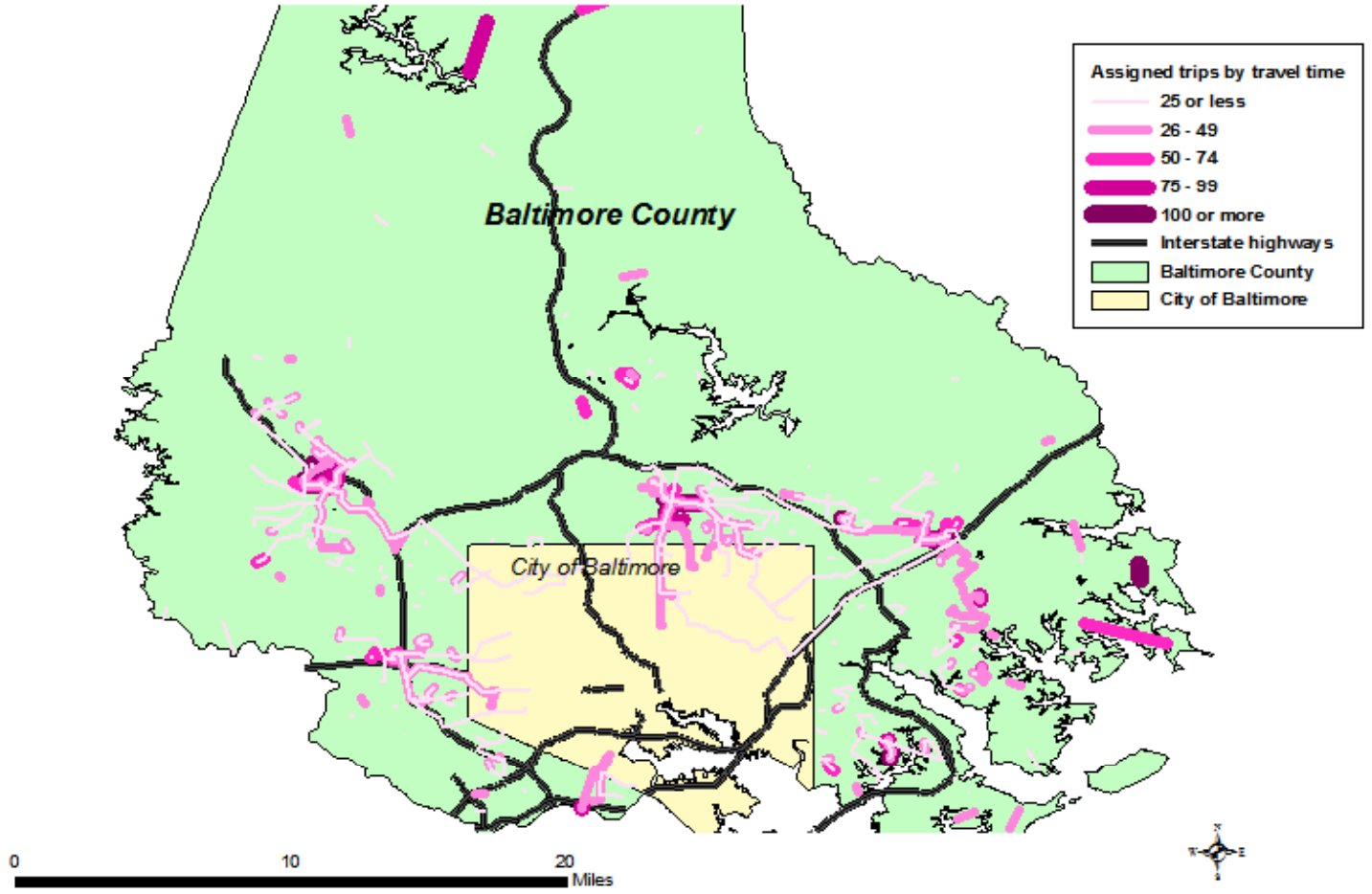
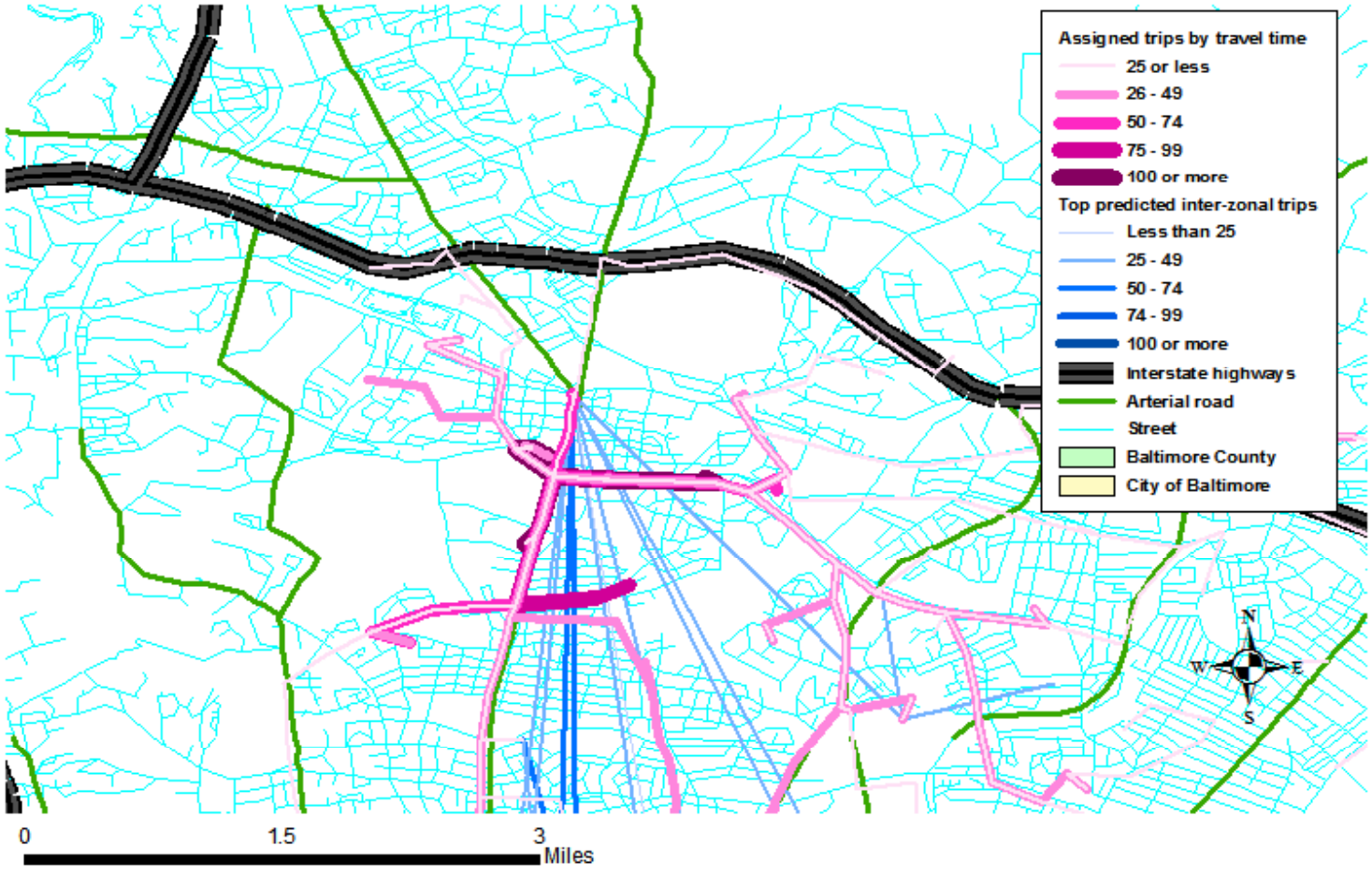


Figure 30.23:

Predicted Baltimore County Crime Trips: 1993-1997

Routes and LInks for Zone-to-zone Trips: All Crimes



3. The origin zone (ORIGIN)
4. The destination zone (DEST)
5. The X coordinate for the origin zone (ORIGINX)
6. The Y coordinate for the origin zone (ORIGINY)
7. The X coordinate for the destination zone (DESTX)
8. The Y coordinate for the destination zone (DESTY)
9. The number of trips on that particular route (FREQ)

These are not illustrated in this chapter because they are identical to the intra-zonal output of the trip distribution module (see Chapter 28).

Save network load

It is also possible to save the total network *load* as an *ArcGIS* '.shp', *MapInfo* '.mif' or various *Ascii* file types. This is the total number of trips on each segment of the network. The routine takes every origin zone to destination zone combination and sums the number of trips that occur on each segment of the network. Click on the "Save output network" box and specify a file name for the output.

Figure 30.24 shows the entire crime trip volume on the network (network load). The assignment was weighted by travel time. Notice how there are many trips on the circular Baltimore Beltway (I-695). Because the road is a freeway, travel is generally much faster than on most arterial roads. Consequently, there are many crime trips being assigned to the freeway even though it is longer than many direct links.

To see how this differs from a shortest distance assignment, the routine was re-run using only distance as the weighting variable. Figure 30.25 displays the results. As seen, the routine does not use the Beltway very much, but instead uses the arterial roads more, particularly the diagonal arterial roads coming out of the City of Baltimore. Since the routine was determining the shortest path on the basis of distance only, it will inevitably find the most direct routes in terms of distance. In terms of travel time, however, many of those routes will be much slower because of traffic lights, cross-traffic, drivers pulling in and out of parking spaces, and so forth. Thus, the freeway is almost always quicker for travel than an arterial road except at peak rush hour conditions. This points out the importance of using travel time and, better yet, travel cost as an impedance variable. Distance is much too simple an indicator of it.

The network load routine can even be used for specific travel modes (and usually is for transportation travel demand modeling). Figure 30.26, for example, shows the network volumes (load) of bus crime trips, again weighted by travel time. According to the model, many of these trips originate in the City of Baltimore. But at the high crime locations, multiple

Figure 30.24:

Crime Volume by Road Segment Weighted by Travel Time

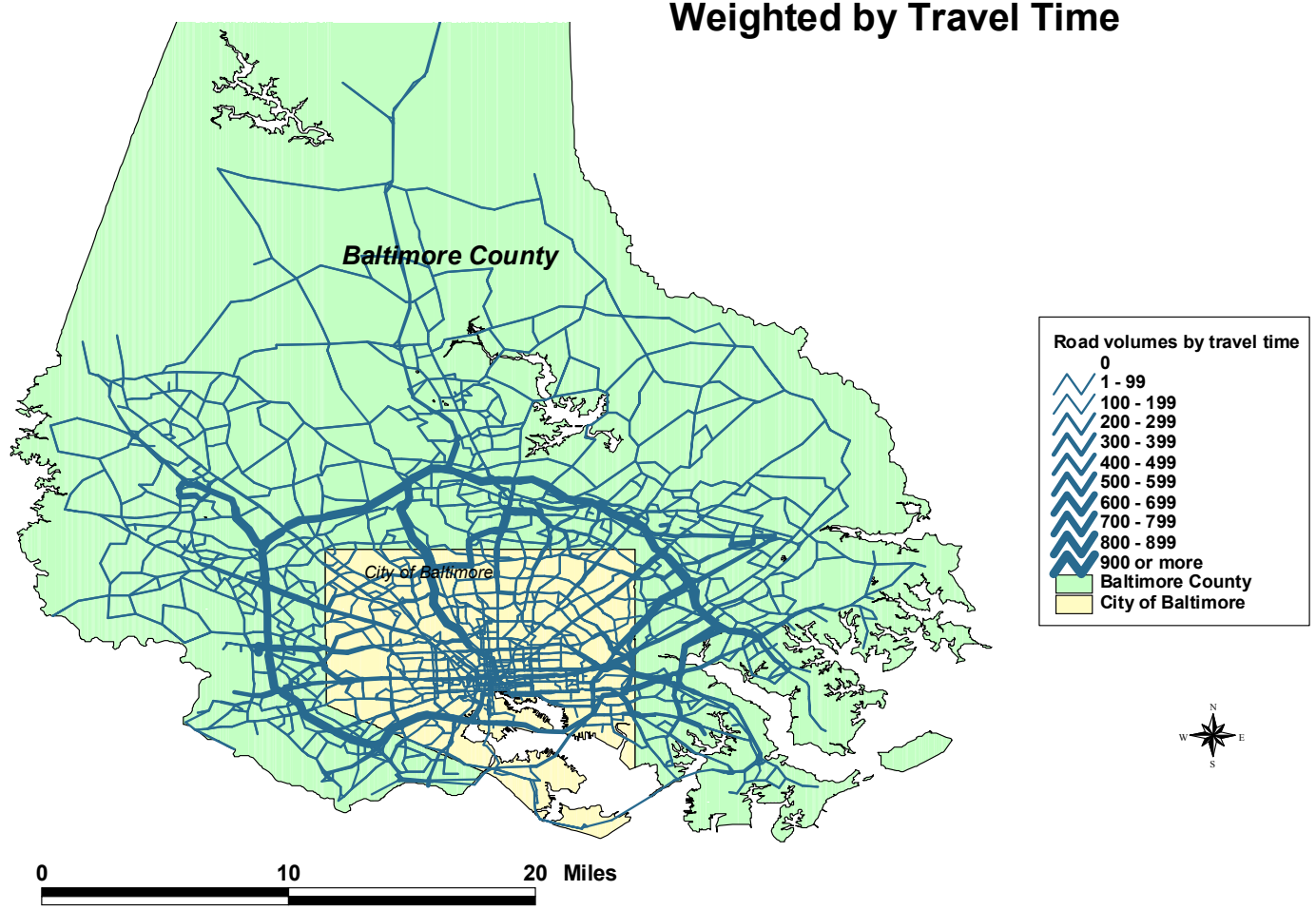


Figure 30.25:

Crime Volume by Road Segment Weighted by Distance

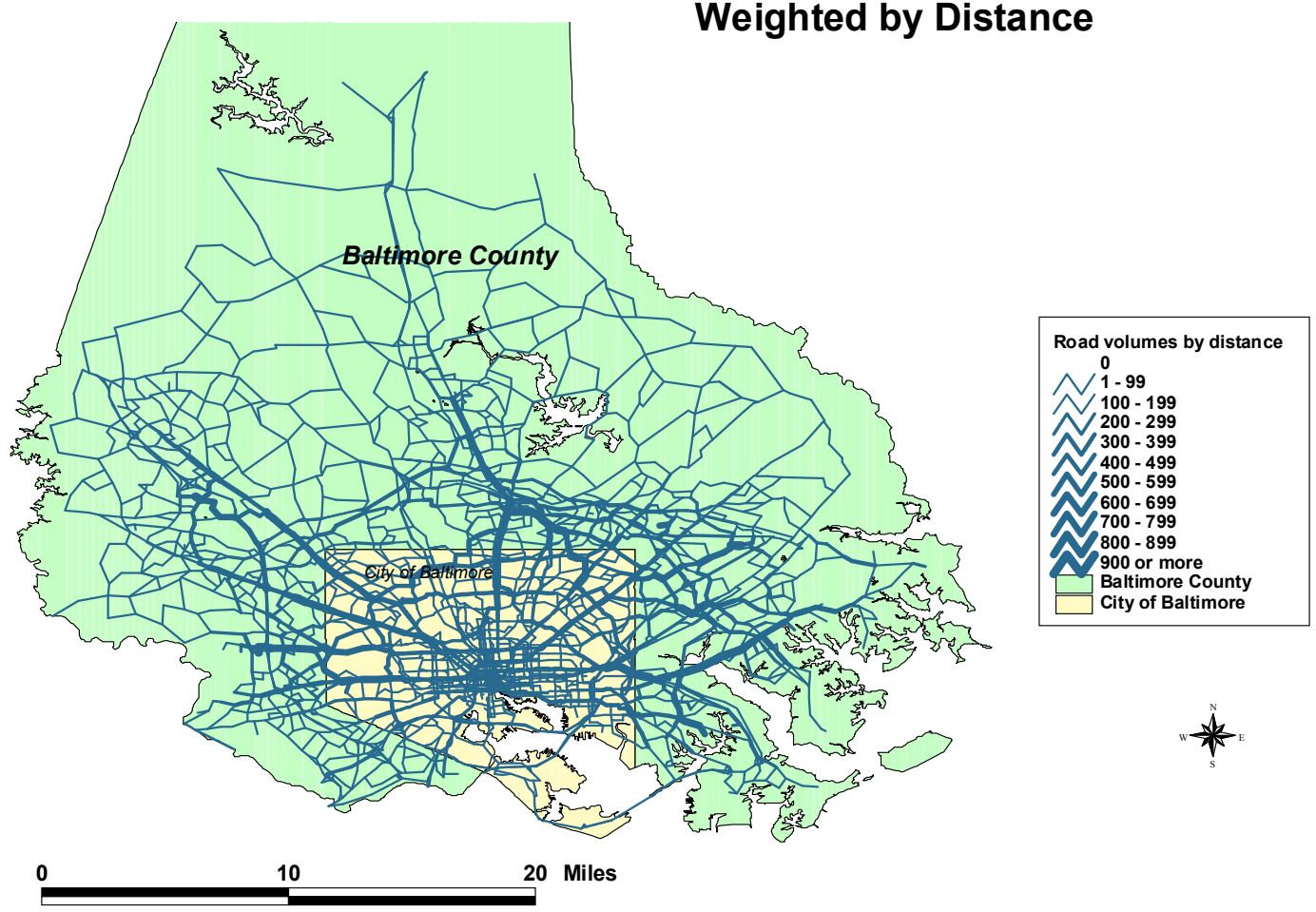
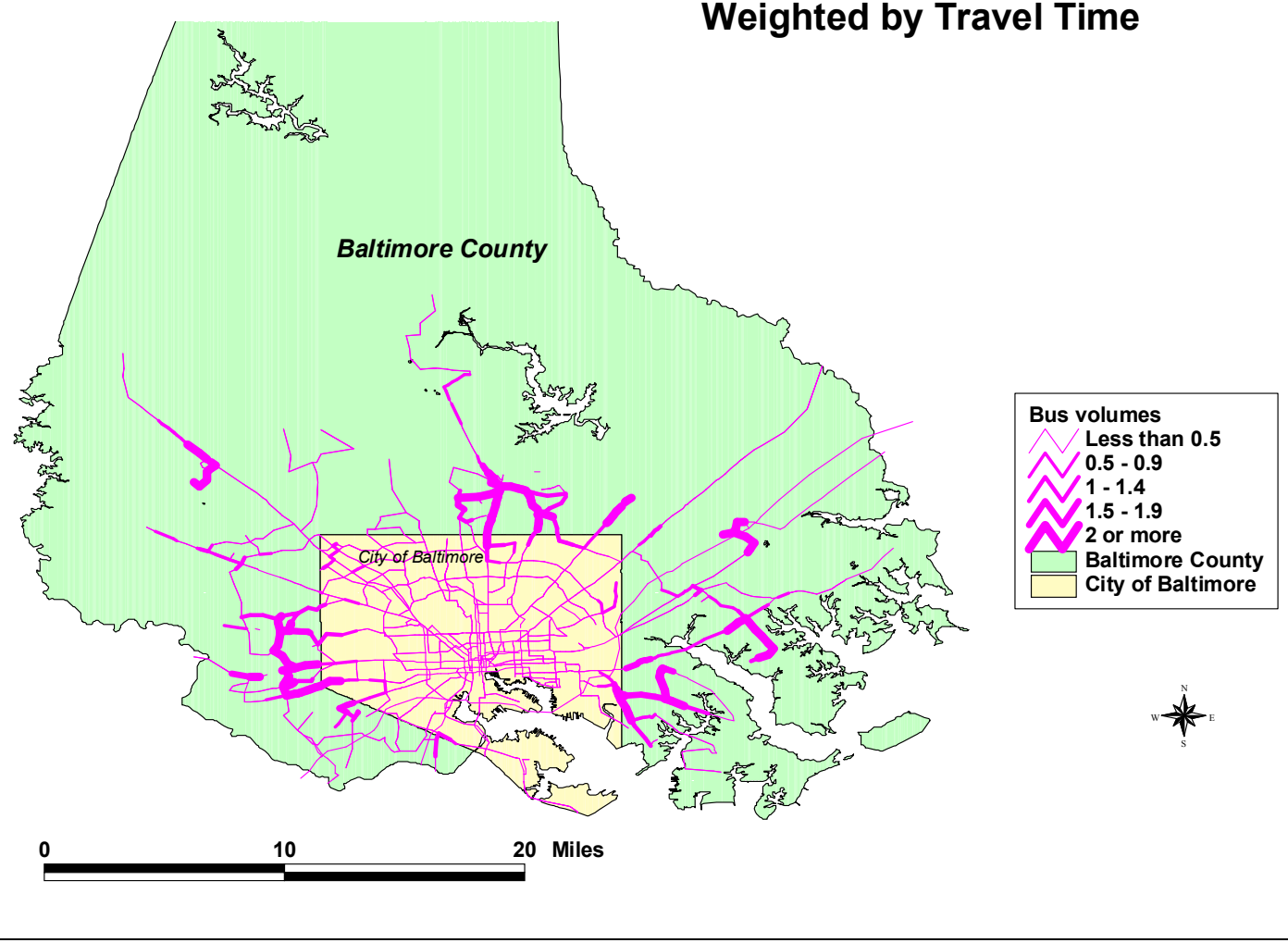


Figure 30.26:

Crime Volume by Bus Route Segment Weighted by Travel Time



bus routes tend to converge producing a high bus trip volume on the adjacent streets. Because of the very small number of bus crime trips predicted by the mode split model, the volumes are not high, even for the highest volume links. Also, notice how the Beltway is not used very much for bus trips, compared to the total network load in Figure 30.24. The reason is that most bus routes do not use the freeway but stay on arterial roads (express buses would be an exception, but those tend to be used primarily for commuting).

Figure 30.27 shows the network volumes of train trips. Since there was no data on travel times along each train segment, the volumes are weighted only by distance. The number of crime trips by train, of course, are limited as was noted in Chapter 29. Also, notice how most of the crime trips taken by train occur on two lines, the Metro line to the west and the MarcP line to the east. In both cases, the train trips start in the City of Baltimore and travel to Baltimore County. These, of course, are predictions of crime travel volumes on the rail network, not empirical verifications.

Modeling Network Assignment of Crime Types

The network assignment routine can be applied to specific crime types. In general, it is a good idea to calibrate a general assignment for all crimes before analyzing specific crimes. The reason is that there are volume dimensions that assign most crime trips to the same segments. Still, some differences can be observed. Figure 30.28 shows the likely routes for vehicle thefts (in blue) and compares it to the likely routes for all crimes (in red). There are similarities and differences. There is overlap in the predicted routes in the southeast and southwest edges of the County with the City of Baltimore, and there is some overlap at the northwest border with the City of Baltimore. At the same time, though, some differences are visible, particularly at the western border with the City of Baltimore.

In other words, the network assignment model shows different routes for vehicle thefts than for crimes in general. This difference, of course, represents differences in the trip distribution matrix of the vehicle thefts compared to all crimes.[3]

Uses of Network Assignment of Crime

A network assignment routine is the culmination of the crime travel demand modeling process. Essentially, it assigns predicted trips (whether for entire origin-destination trip pairs or for mode-specific trip pairs) to an actual network and usually on the basis of least cost. The

[3] The differences could be due to the mode split routine as well as the trip distribution matrix. However, in the case of vehicle thefts, the travel mode is not very relevant since the return trip is always by vehicle - the stolen vehicle, at least to the disposal location.

Figure 30.27:

Crime Volume by Rail Segment Weighted by Distance

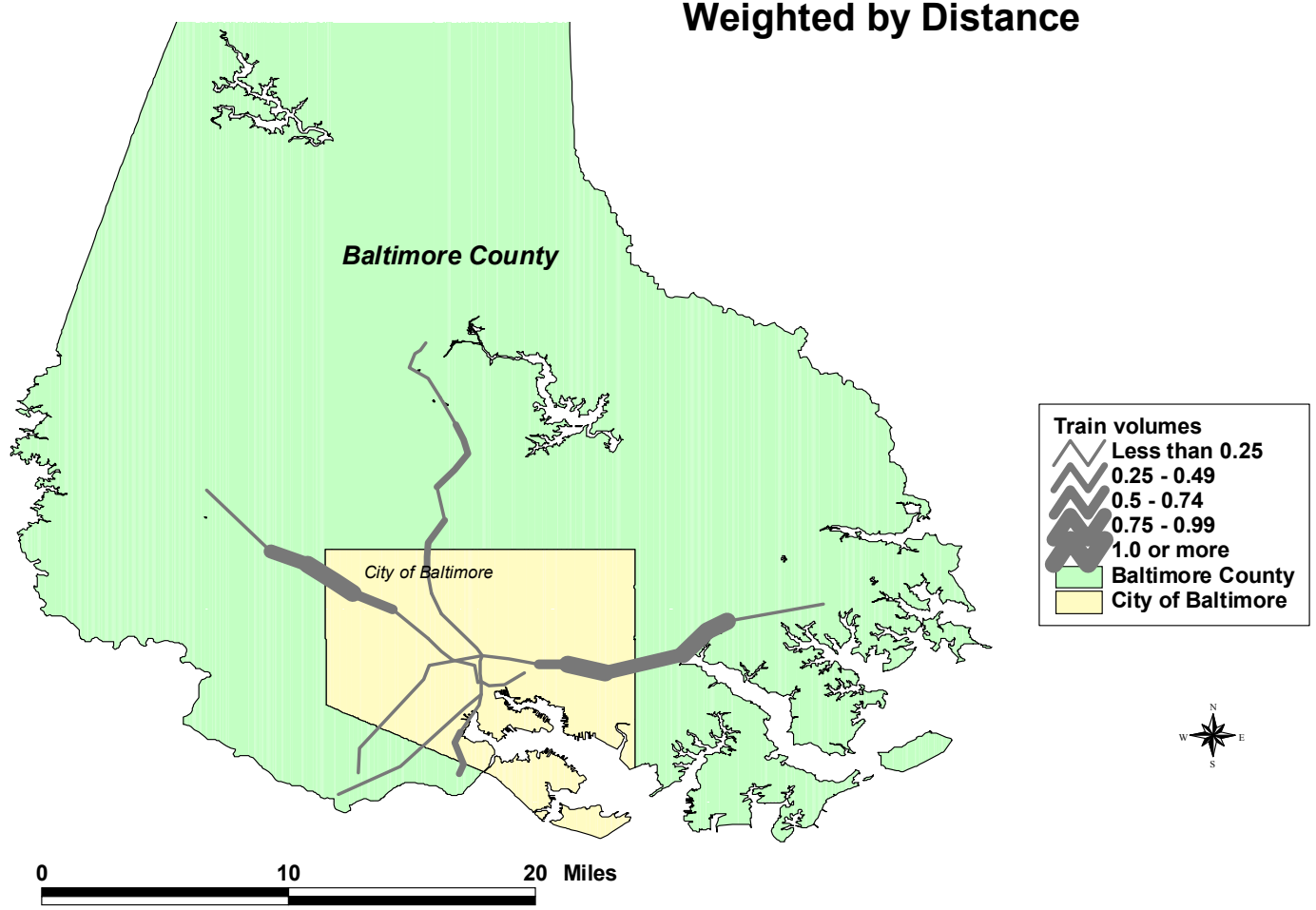
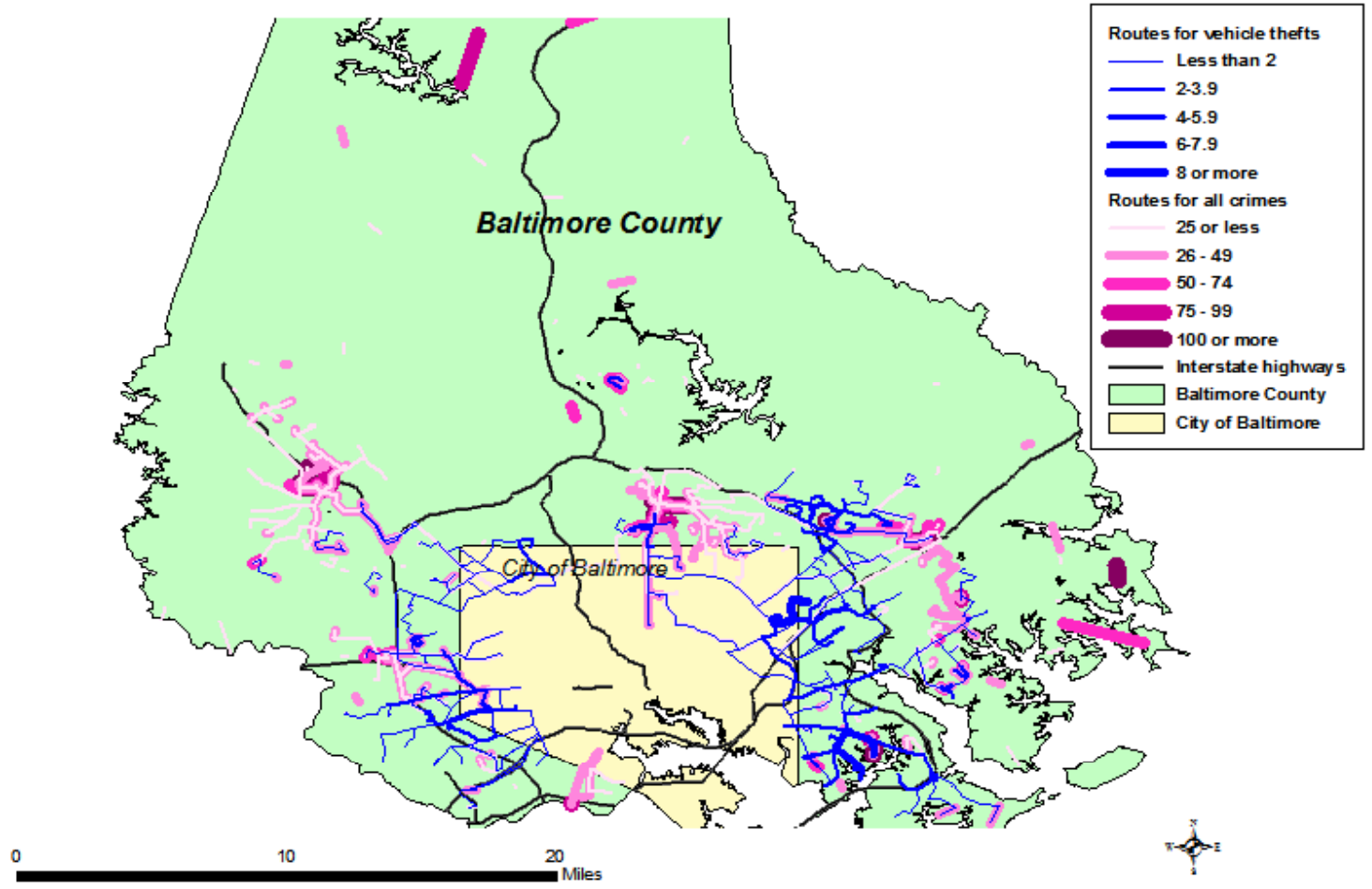


Figure 30.28:

Baltimore County Network Assignment for Crime Trips: 1993-1997

Routes for Zone-to-zone Trips: All Crimes and Vehicle Thefts



algorithm used in the *CrimeStat* network assignment routine calculated the shortest path (in terms of distance, travel time, or cost) and assigned all the trips for each origin-destination pair to this route. The representation is more complex than a simple trip link (which is a straight line) since it uses information on the actual network used. The result is a prediction of routes that are taken to commit crimes and a prediction of the total crime trip volume on each network segment. This is clearly an advance on the geographic profiling/journey-to-crime approach, which has simply analyzed travel distance as an explanatory variable.

Network assignment also has many uses for police. First, it can point out where police need to focus their deployment. In this sense, the progression of the four modeling stages represents adding information to the knowledge of the crime events. Simply mapping the crime events tells a police department where the crimes are occurring. Analyzing the trip distribution tells the department from where the crimes might be originating.

Splitting the distribution by travel model provides information about the likely travel mode used. Finally, assigning the predicted trips to actual routes gives information about how offenders may have traveled to the crime location. The model provides a lot more information than a simple description of a high crime area.

Second, knowing the likely routes of offenders can allow for increased surveillance' and target hardening. Not only can police patrol the likely routes in a more focused manner, but other surveillance tools can be used, too. For example, surveillance cameras that monitor traffic can be used for a variety of purposes. In the U.S., they have tended to be used for monitoring traffic signals for red-light running (IIHS, 2012). However, in Europe they are widely used for a variety of traffic monitoring purposes - speed enforcement, bus lane enforcement, entering the London congestion zone, as well as monitoring traffic signals. In London, for example, the entire monitoring process is automated. For a vehicle making a violation, the camera takes a picture and a software package identifies the license plate. The license number is then matched against a database of vehicles and a traffic citation is sent to the owner. There is no reason why this type of technology could not be structured to also look for stolen vehicles or vehicles belonging to individuals for which outstanding citations have been issued. In short, knowing on which roads high crime trips volumes are likely to occur can help police focus a range of surveillance tools on those locations.

Conclusion

In short, network assignment is a logical step in the modeling of crime trips and one that brings the trips down to actual routes that are used. It is a more realistic representation of travel behavior and one that can allow focused deployment by police.

References

Dijkstra, E. W. (1959). A note on two problems in connection with graphs, *Numerische Mathematik*, 1, 269-271.

IIHS (2012). *Q&A: Red Light Cameras*. Insurance Institute for Highway Safety: Arlington, VA. http://www.iihs.org/research/qanda/rlr.html. Accessed June 5, 2012.

ITE (2010). *Highway Capacity Manual* (5th edition) Institute of Transportation Engineers: Washington, DC. http://www.ite.org/emodules/scriptcontent/orders/ProductDetail.cfm?pc=LP-674. Accessed June 5, 2012.

Levine, N. (2007). Crime travel demand and bank robberies: Using CrimeStat III to model bank robbery trips. *Social Science Computer Review*, 25(2), 239-258.

Nilsson, N. J. (1980). *Principles of Artificial Intelligence*. Morgan Kaufmann Publishers, Inc.: Los Altos, CA.

Ortuzar, J. D. & Willumsen, L. G. (2001). *Modeling Transport* (3rd edition). J. Wiley & Sons: New York.

Rabin, S. (2000a). A* aesthetic optimizations. In DeLoura, Mark. *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 264-271.

Rabin, S. (2000b). A* speed optimizations. In DeLoura, Mark. *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 272-287.

Sedgewick, R. (2002). *Algorithms in C++: Part 5 Graph Algorithms* (3rd edition). Addison-Wesley: Boston.

Shekhar, S. & Chawla, S. (2003). *Spatial Databases: A Tour*. Prentice-Hall: Upper Saddle River, NJ.

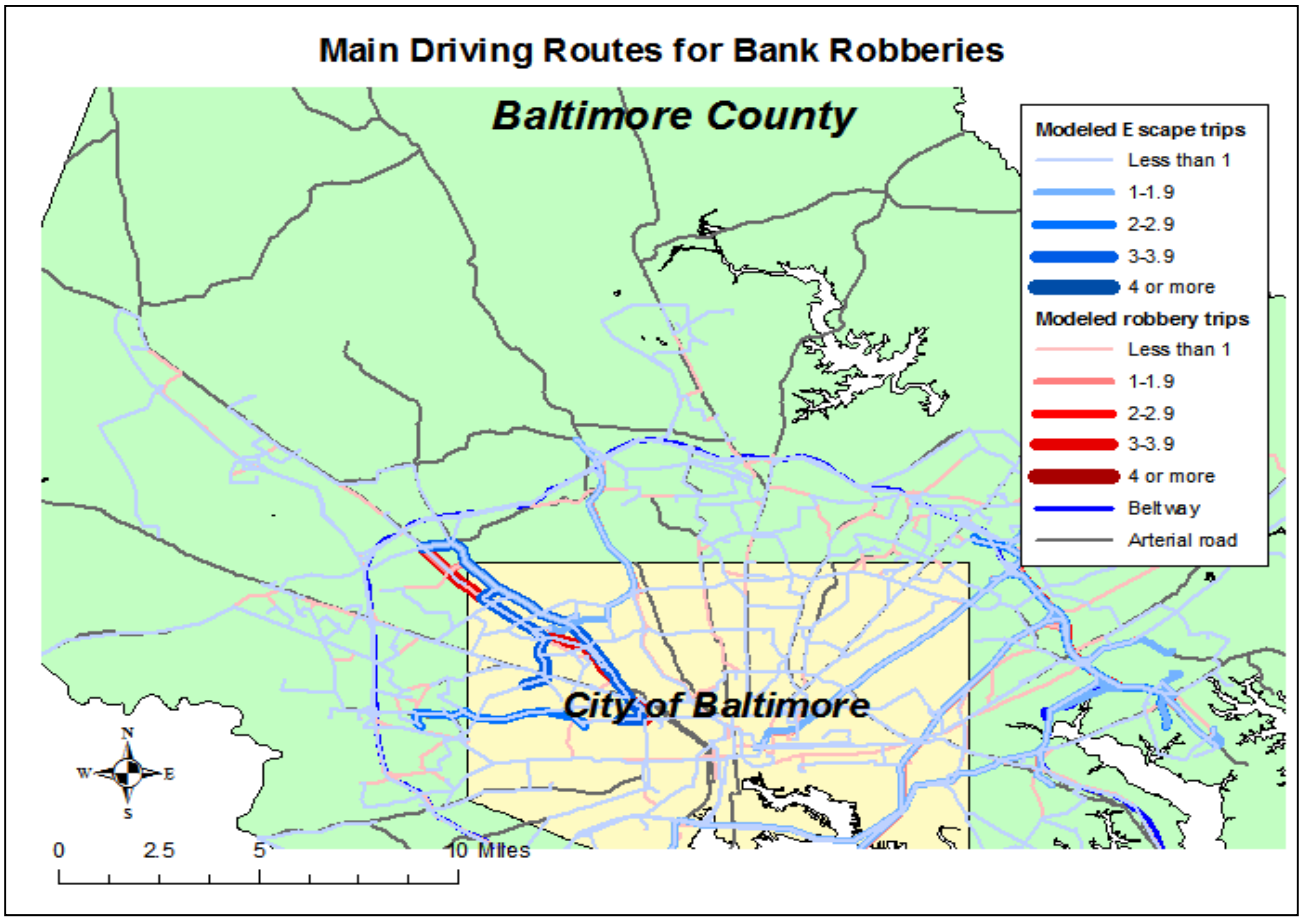
Stout, B. (2000). The basics of A* for path planning. In DeLoura, Mark. *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 254-263.

U.S. Census Bureau (2011). *Tiger Products*. U.S. Census Bureau: Washington, DC. http://www.census.gov/geo/www/tiger/. Accessed May 8, 2012.

Modeling Bank Robbery Trips in Baltimore County, MD

Ned Levine
Ned Levine & Associates
Houston, TX

A study was conducted of 258 bank robberies that occurred in Baltimore County, MD, from 1993 to 1997. The crime travel demand model showed that the bank robbery trips tended to originate in poorer, denser neighborhoods and, in general, rob banks that were close to the offender's residence. Possible travel routes to the banks were modeled as well as escape routes on the assumption that the impedance of using the same routes would be higher after the robberies than before. The alternative routes can provide insights to the police for surveillance after bank robberies have occurred. The full study can be found at Levine, N. (2007). Crime travel demand and bank robberies: Using CrimeStat III to model bank robbery trips. *Social Science Computer Review*, 25(2), 239-258.



Chapter 31:

Case Studies in Crime Travel Demand Modeling:

I - Travel Patterns of Chicago Robbery Offenders

Richard Block
Professor Emeritus
Loyola University Chicago
Chicago, IL

Table of Contents

Chapter 31:
Case Studies in
Crime Travel Demand Modeling:
I - Travel Patterns of Chicago Robbery Offenders

In this chapter, a case study of the crime travel demand model for Chicago, IL, robbery is discussed. Originally written in 2004, it is presented to illustrate the application of the model to a compact city with substantial transit services.

Travel Patterns of Chicago Robbery Offenders

Some neighborhoods are dangerous others are safe. Crime clusters in specific areas. So too do criminals. Criminologists, police, and civilians have known this for nearly 150 years. However, relatively little research has been done on the travel patterns of offenders. Using a modification of standard transportation models, *CrimeStat IV* allows police and researchers to describe and predict travel patterns based on four sequential models.

The object of research presented here is to test the usefulness and feasibility of CrimeStat's Crime Travel Demand model utilizing police reports of all robberies occurring in Chicago in 1997 and 1998 that had at least one known offender who lived in the city. In sum, the objectives of this study of robbery in Chicago are:

1. To test the *CrimeStat IV* crime travel demand model in a mature central city;

2. To describe the travel patterns of robbery offenders based upon offenders home and location of incident;

3. To predict the travel patterns of robbers in 1998 based upon characteristics of the offender's resident neighborhood and the incident neighborhood and a gravity model of the relationship between the two;

4. To predict the travel patterns of robbers in 1998 based upon the patterns of 1997; and

5. To assess the quality of the predictions and their value to the police.

Two Models: Econometric and Opportunistic

As outlined in Chapter 25, a travel demand model is a four-step sequential model. The first stage is trip generation, whereby the number of crimes originating in a neighborhood and the number of crimes ending in a neighborhood are modeled. The second stage is trip distribution which summarizes the number of trips that go from each origin zone to each destination zone. The third stage is mode split, which models the number of trips for each zone pair (origin zone and destination zone) that travels by a particular mode of travel. The fourth, and final stage, is network assignment which models the likely routes taken by offenders in traveling between particular zone pairs.

This mapping of links assumes that travel decisions are based upon minimizing costs to get to a valued destination. When I go to work, I weigh costs and benefits. I choose the route that will get me there quickest with the fewest problems. Early theories of criminology assumed that criminal activity was no different than other behavior. It was determined rationally. By extension, travel routes and crime locations are also determined rationally.

Trips of offenders are similar to any repeated activity. Most of our activities occur near where we live or work or on the path in between. This is our knowledge space. Trips within it maximize our efficiency and minimize costs. Daily purchases occur close to home with a rapid fall off with distance. But major purchases are an exception. They may occur far away. This distance decay can be generalized to travel cost decay. The more expensive in time, money, and distance, the less likely a trip will occur. Applied to robbery, most incidents occur close to home, but a bank robber might incur greater costs to find a good target. In addition most previous research has found that predatory criminals avoid incidents too close to home for fear that they will be recognized. Combined with distance decay, this creates a buffer zone of few criminal incidents (Rossmo, 2000).

Environmental criminology assumes that most activity occurs in a knowledge space that includes nodes of residence work and play and the routes between these (Brantingham & Brantingham, 1984; 1990). However, the components of travel for criminals may not be the same as other people. For example, for someone with a full time job, getting to work as quickly as possible is important; time is money. For a jobless criminal, time may be less important.

Routine activities theory assumes that both targets and offenders choose their activities based on a weighing of costs and benefits. Offenders seek out targets in locations where they are likely to congregate (e.g. bars at closing time, rapid transit stations). A crime occurs when an offender and a target converge in the absence of a capable guardian (Felson, 2002). The routine activities of offenders may mostly be hanging out rather than rationally seeking targets. What is the basis of convergence? Chance or the decisions of offenders? Any potential robber's

decision is effected by both chance and cost. Time and distance are both measures of cost. However, within a short distance of home time and distance costs are near to zero.

An alternative hypothesis is that robbers do not weigh costs and benefits of travel. Rather, they may see an opportunity for crime and take it. Because much of their day to day activity is near home, many incidents occur near the robber's home. Travel patterns are irrelevant for these crimes. The number of robberies decline with distance from the offender's home because fewer of the robber's daily activities occur far from home. On the other hand, more professional robbers may seek out specific areas or locations where lucrative targets are found and may be willing to travel great distances.

In Chicago, an opportunistic robber's knowledge of good targets may be limited to the isolated area around his residence. In addition, trips within the area cost almost nothing, although other costs, such as risk of capture may be relatively high. The difference between Chicago and Baltimore County (or between Chicago and its suburbs) has to do as much with knowledge of the distribution of opportunities as with the cost of travel. Chicago's neighborhoods are so isolated that some offenders may have little knowledge of opportunities outside their resident area. The crime travel demand model holds that in the aggregate offenders appear to weigh costs and benefits. However, the data analyzed here says nothing about individual decisions. Decisions may be made with other factors not captured by shortest distance or time.

In one of the few studies of non-arrested robbers Wright and Decker (1997) found that most St. Louis robbers are opportunistic and rob close to home. Rationality and careful cost calculation have little to do with their decisions. These are people who have a need for quick money. If they saw an opportunity near home, they would take it. Opportunities were most likely to occur as the potential offender and victim went about their daily routine activities. Most of these happened close to home. Therefore, robbery occurred close to home.

The closer to an offender's home that an incident occurs, the more likely the incident has resulted from a chance meeting. The further away that it occurs the more likely that it was planned. Part of the planning is transportation costs. It is difficult to calculate this for offenders. The best we can do is estimate travel time.

Crime Travel Demand Modeling in Chicago

The Offender Travel Model is a new application of the Travel Demand Model. The travel demand model has been in development since the 1950's. It is used in every metropolitan area in the United States. *CrimeStat*'s crime travel demand model was outlined in Chapter 25.

As applied to robbery in Chicago, description is as important as prediction. While the Chicago Police Department (CPD) has long collected information on the location of the incident

and residence of the offender, these were not linked in any systematic way. In meetings with the department, credible descriptive maps proved to be the most convincing reason to use the new *CrimeStat* travel demand module. Before a new technique is tested, its potential credibility must be demonstrated. Therefore, the last phase, in the Chicago Travel Demand Model emphasized both the predicted travel demand model and the observed travel of offenders.

Analysis of Chicago's Crime Travel Demand proceeds in three stages. The first step (trip generation) is a prediction of variables associated with the number of crimes originating in each zone and the number of crimes ending in each zone.

The second step is the prediction of links between zones based on zonal characteristics of incident locations and offender residences and a measure of the attraction between the two zones. These predictive models are compared to the observed links and trips and the previous year's trips used as a prediction.

The mode split step was not run because of the lack of data. Unfortunately, the Chicago police data does not permit an analysis by different modes of transportation (see Chapter 29). Data on whether the offender drove, walked, or rode rapid transit to the incident are not collected.

The final step is the description of probable travel routes from the offender's home zone to the incident zone based on shortest time or distance along a transportation network. The links modeled in the second step can be converted to a probable route between home and incident zones over a road network or a summary network load which aggregates travel of all offenders along a transportation network.

Data for the Chicago Study

Incident and Arrest Files

The analysis presented here merged information from many sources. This research is based on incident and arrest records from the CPD. Excluding O'Hare Airport, the city of Chicago is divided into 946 traffic analysis zones. Incidents are assigned to these zones for both residence location (the origin) and the crime location (the destination). These include all Chicago robberies in 1997 and 1998 that had at least one known offender who lived in Chicago. These were geocoded by the address of the incident and the home address of all known offenders. Offenders who traveled longer distances were probably under-represented (Block, 2007). About 20% of all reported robberies were included. In 1997, there were 25,000 robberies reported to the police. Of these robberies, 4,636 resulted in the arrest of at least one Chicago resident. Including robberies with multiple offenders, there were 6,643 crime trips.

Traffic Analysis Zones

These incidents and offenders are counted in 946 Traffic Analysis Zones (TAZ). O'Hare Airport is excluded. Chicago's traffic analysis zones are mostly based on a uniform grid of 1/2 mile squares. These are not based on census tracts or other city divisions. However, some census data was available for these zones along with information on employment. About 100 of them had no census population and therefore were unlikely to include the residence of an offender. Land use, employment, population, and robbery incident and offender residence counts were available for all zones. Land use goes beyond the standard census measures to include characteristics from many data sources that might be related to crime. Among these are code violations, vacant parcels, fires, liquor licenses, pawn shops, entertainment venues, distance from the central business district and other potentially criminogenic characteristics.[1] These traffic analysis zones were the unit of analysis. Trips were defined from the center of a zone.

Chicago's Road Network

The base of Chicago's road network is a grid with 1/8 mile between blocks, a feeder street every half mile, and a main street every mile. Layered on top of this grid is a series of diagonal streets that tend to be major shopping streets and a relatively small number of expressways that converge at the edge of the central city. A semi-expressway, Lake Shore Drive, runs along the lakefront for 25 miles. Chicago has a well developed rapid transit system that, unfortunately, could not be included in the current analysis.

Two street networks were available for analysis:

1. **Modified TIGER Line File**: A mostly complete map of all streets and rail lines. Following police practice, the modified TIGER file allows for geo-coding in non-addressed areas, such as parks, by extending the base grid. All public streets are included, but one-way streets are not taken into account and the shortest distance may be on a route that no one would travel. Some areas of the city were not well mapped.

2. **Modeling network:** This includes Expressways, principal arterials and collector roads. Each road segment is uni- (or single-) directional; that is, it expresses travel in only one direction. Thus, for a two-way road, there will be two records for every segment, one in each direction. This has the advantage that one-way streets can examined since there will not be an opposite direction pair. On the

1 In contrast to many cities, Chicago has a large population living in the central business district and lacks a ring of impoverished communities surrounding downtown.

other hand, a modeling network is less complete since minor streets are ignored. This type of map is useful for capturing trips that occur over a mile or more, but is not very useful for the many trips of less than 1/2 mile that occur in Chicago. It does take into account one-way streets. Using distance, the network will over-emphasize surface diagonal streets and will under-emphasize expressways.

One of the advantages of the modeling network is that street segments can be weighted by speed or travel time, rather than just distance. There are eight distinct time periods with the travel time on each segment by period being indicated. Each street segment can be weighted by its travel time in minutes during a specific time period (e.g.; 7- 9 AM) to allow a more realistic description of travel behavior. Further, travel in opposite directions can be treated differently since travel times can be different for each direction. During rush hour, travel in one direction may be much quicker than travel in the other direction. Weighting by travel time will allow larger arterial roads and expressways to be chosen more because travel speeds will generally be faster on the larger capacity roads. This network tends to be most realistic for longer trips but, again, is not useful for very short 'local' trips since the local, neighborhood road network is not included. A greater percentage of the travel is on expressways.

Trip Generation

Using the arrest data, events were aggregated to the TAZ's for both the origins and the destinations. As expected, the distribution of crimes by origin zone and by destination zone were highly skewed. For example, 419 zones had no robberies originate in them while one zone had 27 and another had 24 originate in them. A similar condition held for the number of crimes by destination. For example, no robberies occurred in 409 zones while one zone had 24 robberies and two had 23.

Separate models of these incident were developed at the zone level. The regression analysis tools in CrimeStat are excellent, but choosing regression predictors requires both skill and theory. Many explanatory variables were tested. The independent variables chosen for analysis were based on those previously found to be important predictors of violent crime in Chicago. Significant variables were:

1. POP2000 The most important was the 2000 population because the dependent variable was a predicted count of origins or destination. Other variables that were included were:

2. ETHNICPER The percentage of the dominant racial or ethnic group within the TAZ. Recent research (Sampson & Raudenbush, 2001) has found that racial isolation and poverty predicted high community levels of violence.

3. POVPERCENT The percent of the households below the poverty level. Sampson and Raudenbush (2001) found this to be a dominant variables explaining community disorder.

4. VENUE The number of entertainment venues (clubs, theaters, bowling allies) in a TAZ. This is information gathered from the MetroMix and the Reader in 2002. It was negatively related to the residence of the offender and was probably more a measure of perceived neighborhood safety than availability of targets.

5. PAWNSHOP The number of pawnshops is included in several regressions. A pawnshop is both a focus for potential targets and a good place to get cash.

6. VACANT: Count of vacant buildings in the TAZ. Perhaps this is an indicator of general neighborhood dilapidation (Broken Windows).

The variables that were not significantly related to origins or destinations included many that are typically related to travel demand including employment and distance from the central business district. In addition, variables that are often associated with robbery, such as counts of drug arrests, convenience stores, liquor licenses, banks and currency exchanges were unrelated to origins or destinations after poverty and population were accounted for. Few TAZ characteristics that might attract an offender to commit a crime were significantly related to the number of robbery incidents in a TAZ. In general the results of the regression models and the resulting travel demand matrix supported the depiction of robbery in Chicago as occurring in or near the offender's relatively isolated home neighborhood.

Poisson regression models for origin and destination zone counts for overnight trips were similar in 1997 and 1998. Table 31.1 presents the final Poisson regression model for the resident zone of robbers in 1998.

The Likelihood Ratio was good and an analysis of the residual errors did not reveal any major outliers. Given the large number of zones (n=946) the regression predicted variations in the count of origins fairly well.

Table 31.1:
Overnight 1998 Robbery Origin Model

Data file:	Chicago TAZ with Time.dbf
Type of model:	Origin
DepVar:	**Robbery Origins 8PM-5:59AM**
N:	946
Df:	940
Type of regression model:	Poisson with over-dispersion correction
Log Likelihood:	-2,011.35
Likelihood ratio(LR):	2,962.73 P-value of LR: 0.0001
AIC:	4,034.71
SC:	4,063.82
Dispersion multiplier:	1.00

Predictor	DF	Coefficient	Stand Error	Tolerance	z-value	p-value
CONSTANT	1	**-2.072610**	0.170828	.	-12.132746	0.001
POP2000	1	**0.000235**	0.000011	0.876420	22.156415	0.001
ETHNICPER	1	**0.015786**	0.001746	0.909463	9.042151	0.001
POVPERCENT	1	**0.037134**	0.002144	0.872974	17.321707	0.001
VACANT	1	**0.016970**	0.002528	0.835809	6.712064	0.001
VENUE	1	**-0.115182**	0.033458	0.933336	-3.442566	0.001

Similarly with the destination model (Table 31.2), the Likelihood Ratio of the destination model was reasonably good, though not as strong as with the origin model. There were not any apparent major outliers. Given the large number of zones (n=946) the regression predicted variations in the count of destinations fairly well.

In both regression models, population had a positive relationship to the number of crimes. Similarly, the poverty variable and the ethnic homogeneity variable were positively related to the number of crimes, both origins and destinations.

Table 31.2:
Overnight 1998 Robbery Destination Model

Data file:	Chicago TAZ with Time.dbf
Type of model:	Destination
DepVar:	**Robbery Destinations 8PM-5:59AM**
N:	946
Df:	941
Type of regression model:	Poisson with over-dispersion correction
Log Likelihood:	-2,041.56
Likelihood ratio(LR):	2,661.30 P-value of LR: 0.0001
AIC:	4,093.11
SC:	4,117.37
Dispersion multiplier:	1.00

Predictor	DF	Coefficient	Stand Error	Tolerance	z-value	p-value
CONSTANT	1	**-1.946591**	0.032370	.	-60.135432	0.001
POP2000	1	**0.000218**	0.000008	0.898680	26.418877	0.001
ETHNICPER	1	**0.015913**	0.000874	0.944910	18.201093	0.001
PAWNSHOP	1	**0.335678**	0.029184	0.954563	11.501940	0.001
POVPERCENT	1	**0.035707**	0.001888	0.989400	18.913079	0.001

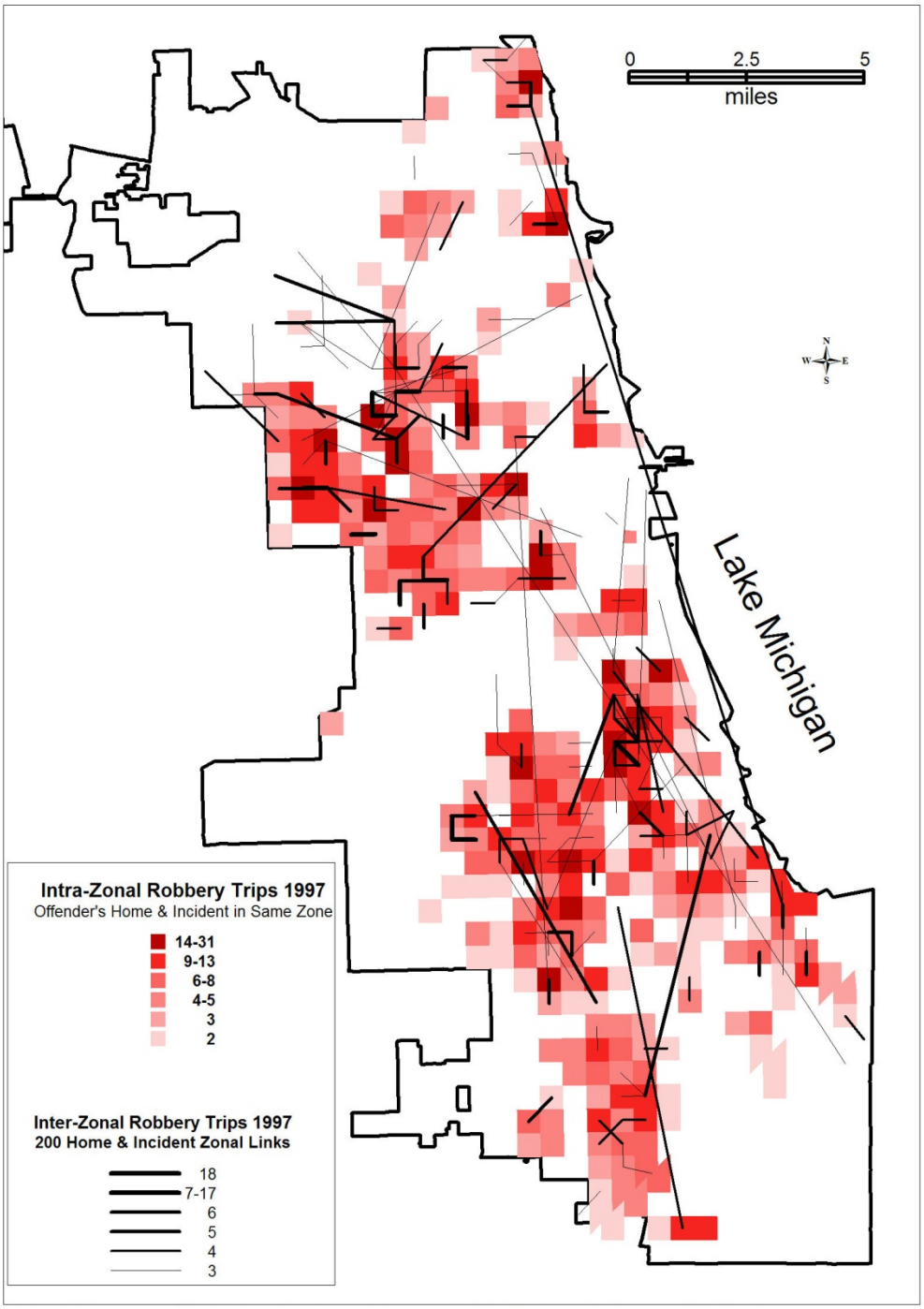
Trip Distribution

After the two predicted models were developed, the trip distribution was predicted, in other words the modeled number of trips that went from each origin zone to each possible destination zone was estimated (trip distribution). The inputs were the predicted origins and predicted destinations for robberies in 1998 from Tables 31.1 and 31.2.

The test of CrimeStat's crime travel demand module began with an analysis of 1997. Preparatory analysis indicated that 29% of robbery trips occurred in the offender's home zone. While the number of intra-zonal trips can be mapped and predicted, travel within a zone cannot be described.

Using observed crime trips, the actual number of trips from each zone to every other zone was calculated. Figure 31.1 depicts the volume of observed inter- and intra-zonal trip links in 1997. The zone shadings indicate the number of intra-zonal trips. The width of the links indicates the frequency of trip links for zones with 3 or more links.

Figure 31.1:



Robbery 1997: Intra-Zonal & Inter-Zonal Links

Source: Chicago Police Department Cartography: Richard Block, Loyola University Chicago

Impoverished areas of the west and south side dominate this analysis. Most inter-zonal links are quite short (Figure 31.2). Many begin in zones that also have many intra-zonal trips. In Las Vegas and Baltimore County many links are associated with specific sites such as shopping malls or entertainment areas. Within the City of Chicago, the links lack a clear focal zone for incidents. However, few robbery trips are made to the central business district.

From a police perspective, even the distribution of crime trips can be of value for tactical purposes and for planning interventions. However, the description of 1998 night time robberies south side dominate this analysis. Most inter-zonal links are quite short. Many begin in zones that also have many intra-zonal trips. In Las Vegas and Baltimore County many links are associated with specific sites such as shopping malls or entertainment areas. Within Chicago, the links lack a clear focal zone for incidents. However, few robbery trips are made to the central business district.

A trip distribution analysis includes both inter- and intra-zonal trips in a single analysis. The analysis is not of travel from home to destination, but from a home zone to a destination zone. For transportation planners inter-zonal trips are more important than intra-zonal trips because these predict changing transportation needs. The volume of within zone travel can be predicted but not specific routes. However, many Chicago robberies (29% in 1997, 26% in 1998) are intra-zonal.

Therefore, two techniques were tested to account for the many intra-zonal trips. First, both inter- and intra-zonal overnight robberies trips were included in the same model. Second, to see whether different variables were predicting incidents close to the offender's home address from those further away, inter- and intra-zonal trips were analyzed separately. Ultimately, I concluded that there was little to be gained by separating the two types of trips.

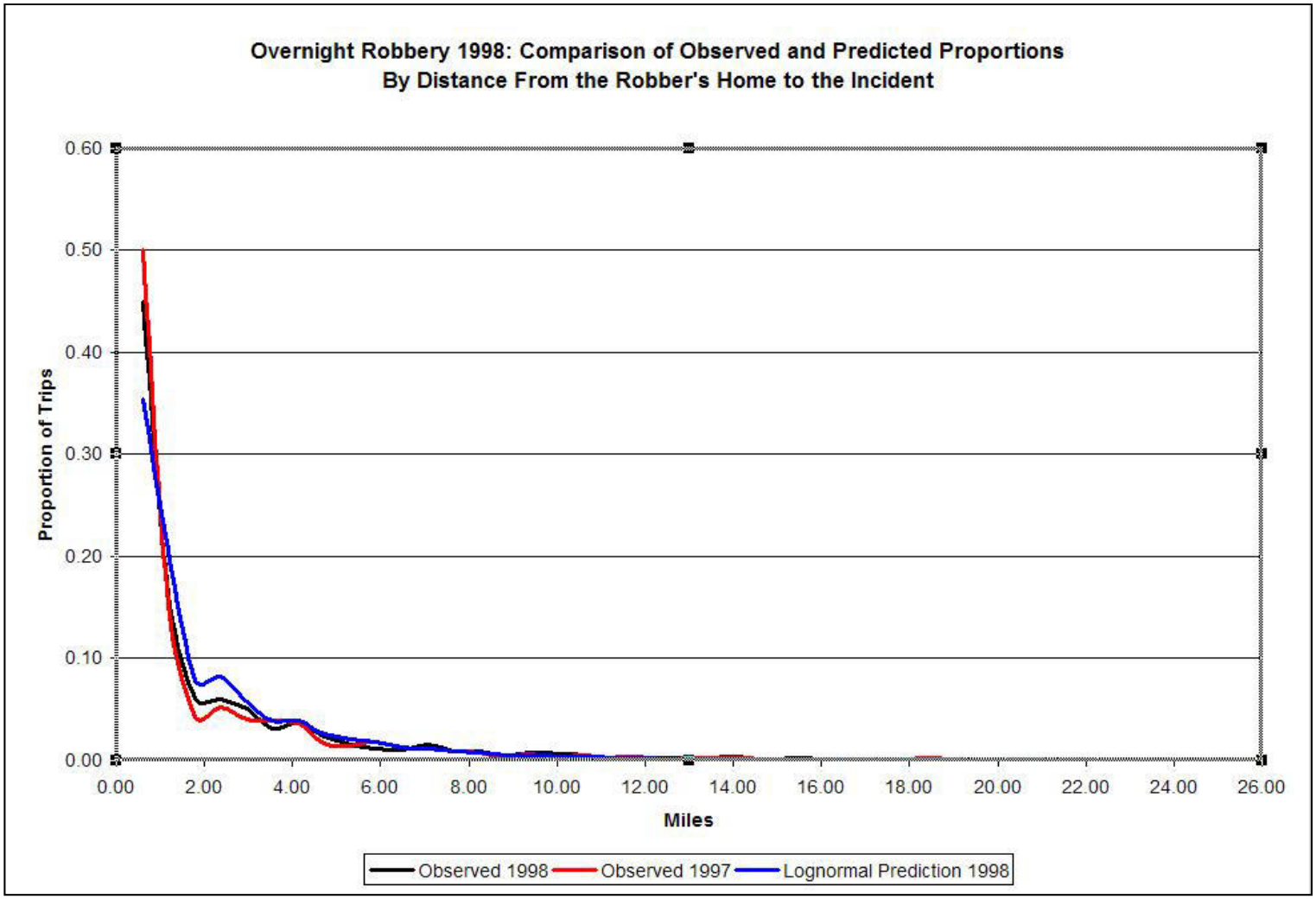
Gravity Model of Chicago Robbers

The gravity model that underlies CrimeStat's trip distribution model assumes that travel between or within zones is dependent upon the offender pool, opportunities, and costs. Conceptually, this can be written as:

$$T_{ij} = \frac{\alpha P_i \beta A_j}{C_{ij}^{\lambda}} \qquad (31.1)$$

where T_{ij} is the number of trips from zone i to zone j, P_i is the number of offenders in zone i (the offender pool), A_j is the number of attractions or opportunities in zone j, C_{ij}^{λ} is cost of travel from zone i to zone j, α and β are coefficients and λ is an exponent. The impedance (or 'cost')

Figure 31.2:


Overnight Robbery 1998: Comparison of Observed and Predicted Proportions
By Distance From the Robber's Home to the Incident
Proportion of Trips
0.60
0.50
0.40
0.30
0.20
0.10
0.00
0.00
2.00
4.00
6.00
8.00
10.00
12.00
14.00
16.00
18.00
20.00
22.00
24.00
26.00
Miles
Observed 1998
Observed 1997
Lognormal Prediction 1998

component is modeled with a mathematical function. After experimentation, I found that the best impedance function was a lognormal distribution with a mean of 2 miles and a standard deviation of 5. The resulting model fit the actual trip length distribution quite well.

Predicting 1998 Trips From 1997 Trips

Can the 1998 distribution be successfully predicted from the 1997 model? In time series analysis, the best prediction of one period is generally the period that immediately preceded it. In spatial analysis, this is also likely to be true, especially in a mature city. However, while neighborhood characteristics change slowly in Chicago, they do change. During the late 1990's many public housing projects were emptied and most were torn down. While few neighborhoods deteriorated, many gentrified. Any of these might cause a change in the distribution of robbery trips.

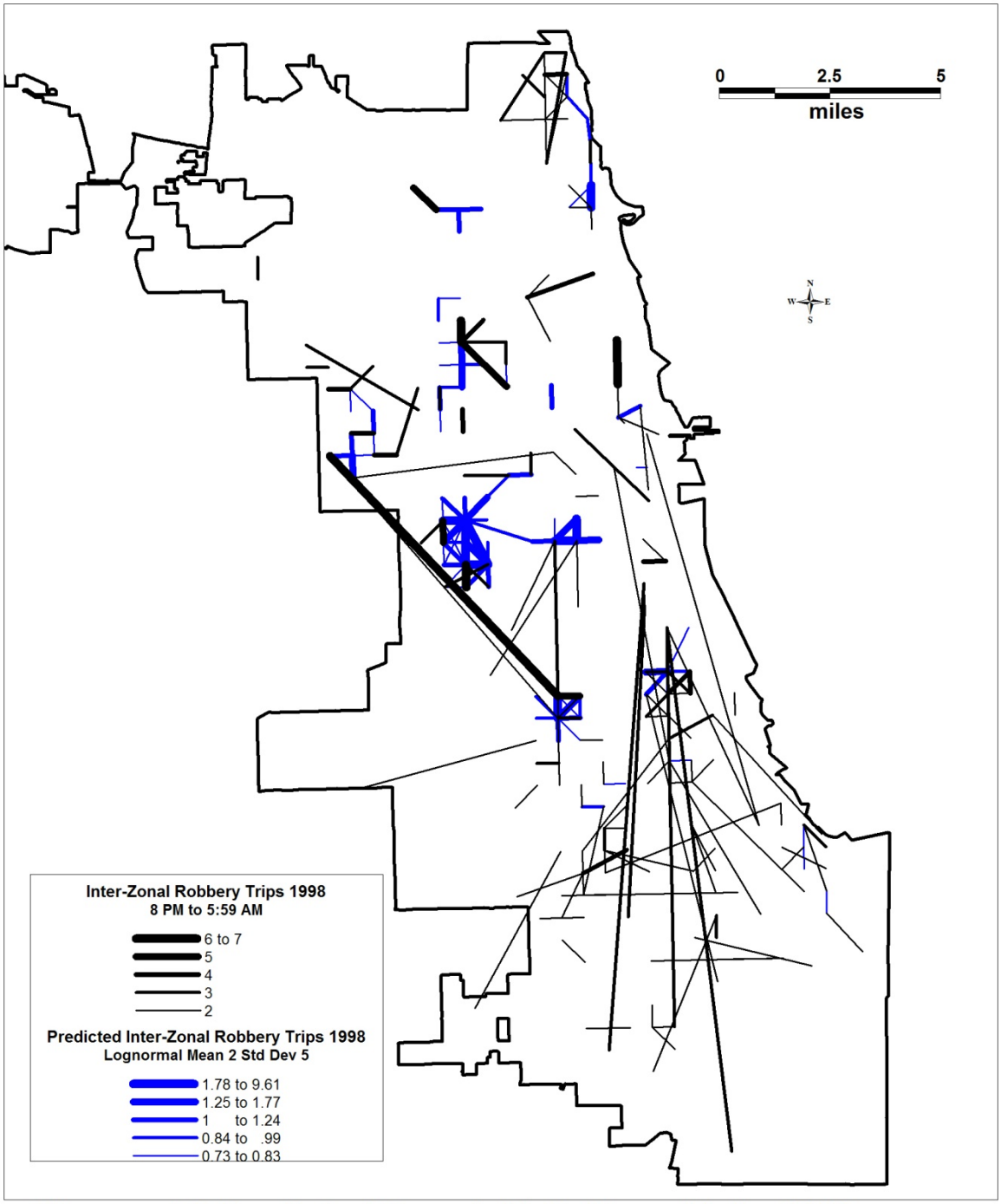
Nevertheless, to test the model, the 1997 observed robbery travel matrix was used to predict observed travel in 1998 (Figure 31.3). *CrimeStat IV,* in conjunction with a GIS and a statistical package, provides several comparison tools. Comparing 1997 and 1998, the fit was quite good. Including street segments that had no trips in either year, 55% of the trip links in 1998 were predicted by the trip links in 1997. The coincidence ratio of .86 for 1998 and the distance distribution in Figure 31.2 above indicated a high degree of similarity. However, a comparison of the top 300 trip links illustrated that, while zones with many intra-zonal incidents were fairly well predicted, inter-zonal trips were not as well predicted. Mapping these made clear that 1997 inter-zonal links did not accurately predict specific 1998 links (Figure 31.4). However, specific links may be less important from a police perspective than knowledge of the frequency of offender travel on specific streets. The coincidence ratio was about the same for both the 1997 and 1998 comparisons (Figure 31.2 for night time robbery trips).

In figure 31.3, predicted and observed overnight robbery trips in 1998 are pictured. To graphically indicate the trips, straight lines are used to indicate links between zones and widths to indicate volume. An inspection of Figure 31.3 shows that many specific links were not well predicted. In general, the prediction underestimated very short trips but overestimated middle distance trips (2-4 miles).

Predicting Overnight Robbery Trips

After selecting only those 1998 robberies that occurred from 8 PM to 5:59 AM, a zone to zone matrix was constructed. This matrix included both intra-zonal (31.5% of the total) and inter-zonal trips. As shown in Figure 31.5, zones with many intra-zonal overnight trips also had

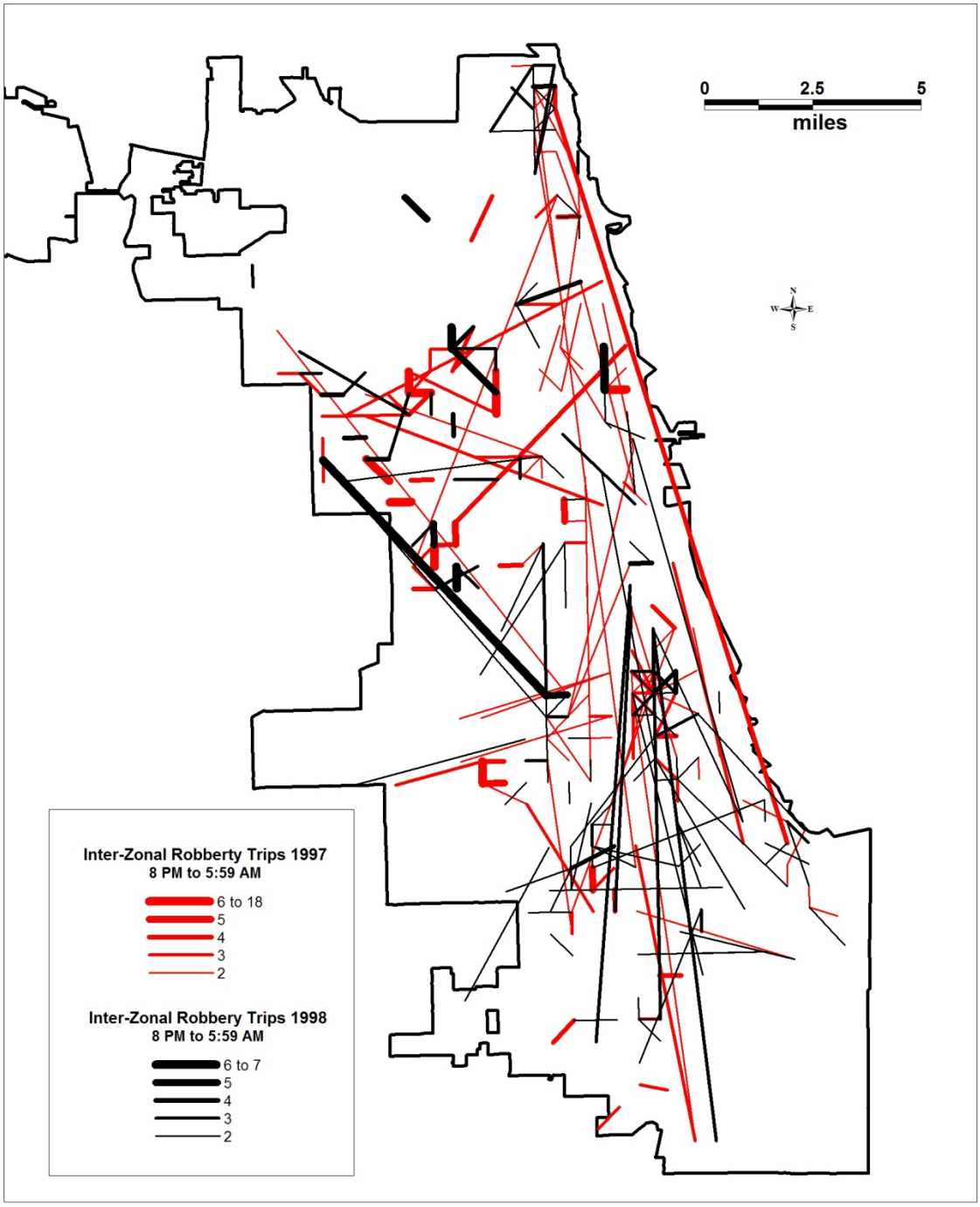
Figure 31.3



Observed & Predicted Overnight Robbery Links 1998

Source: Chicago Police Department Cartography: Richard Block, Loyola University Chicago

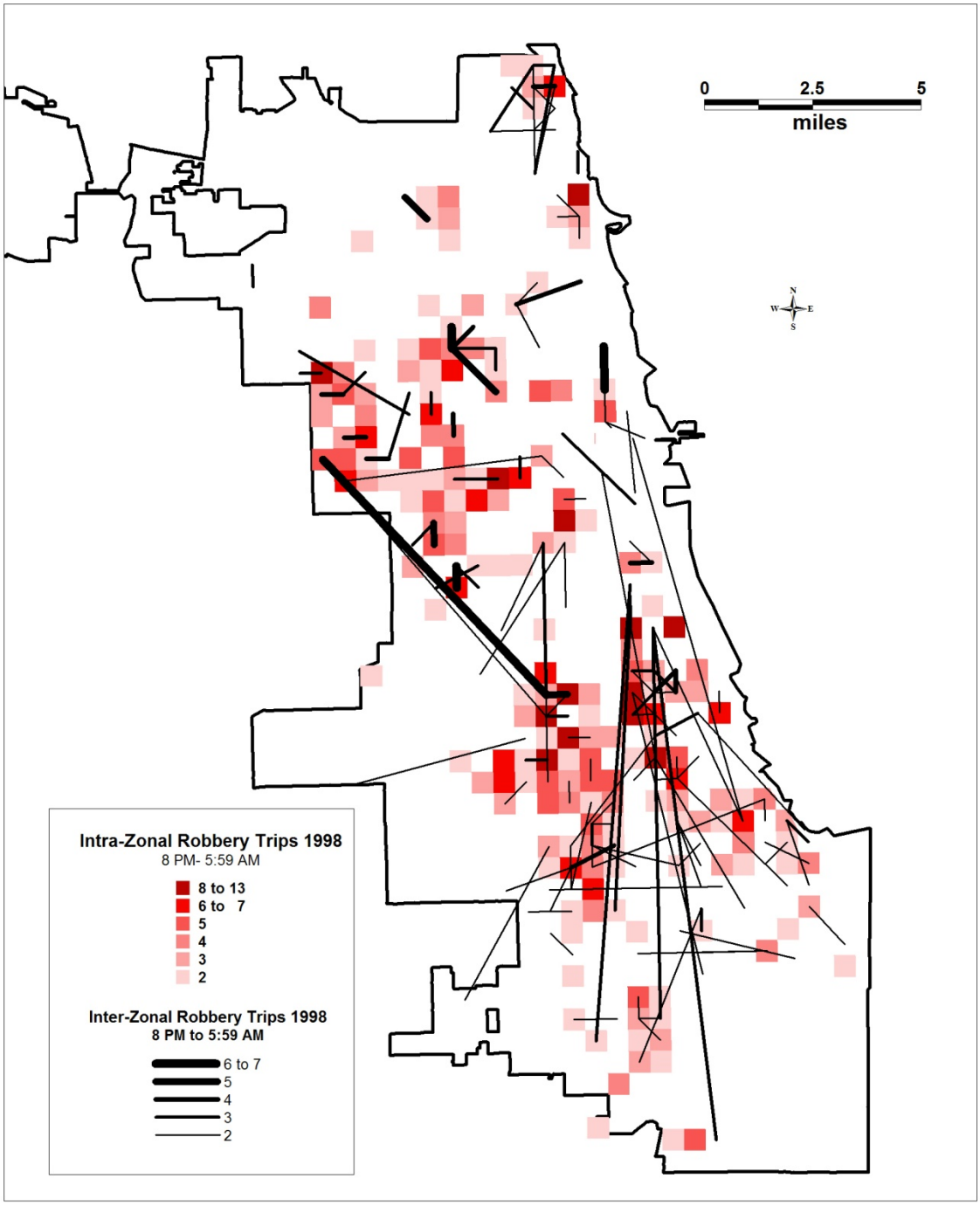
Figure 31.4:



Robbery 1998 & 1997:Observed Links

Source: Chicago Police Department Cartography: Richard Block, Loyola University Chicago

Figure 31.5:



Robbery 1998: Overnight Intra-Zonal & Inter-Zonal Links

Source: Chicago Police Department Cartography: Richard Block, Loyola University Chicago

many inter-zonal trips. Intra-zonal links were widely dispersed throughout the city with an area of concentration on the west side, but there was no clear pattern.

Mode Split

Because of the lack of information about travel mode, the mode split model was not run. It is hoped that, with better information, this type of model could be run in the future.

Network Assignment

The third, and final, step in the analysis was to examine the likely routes taken as well as the total demand placed on the road network. Network assignment is an especially useful tool for police work because it can suggest possible locations for intervention. Because it is based on the actual street network, it is more concrete than a depiction of links. Therefore, I tested several ways to depict network assignment for 1997 robbery travel before proceeding to the 1998 analysis.

The network assignment routine in *CrimeStat IV* outputs two results:

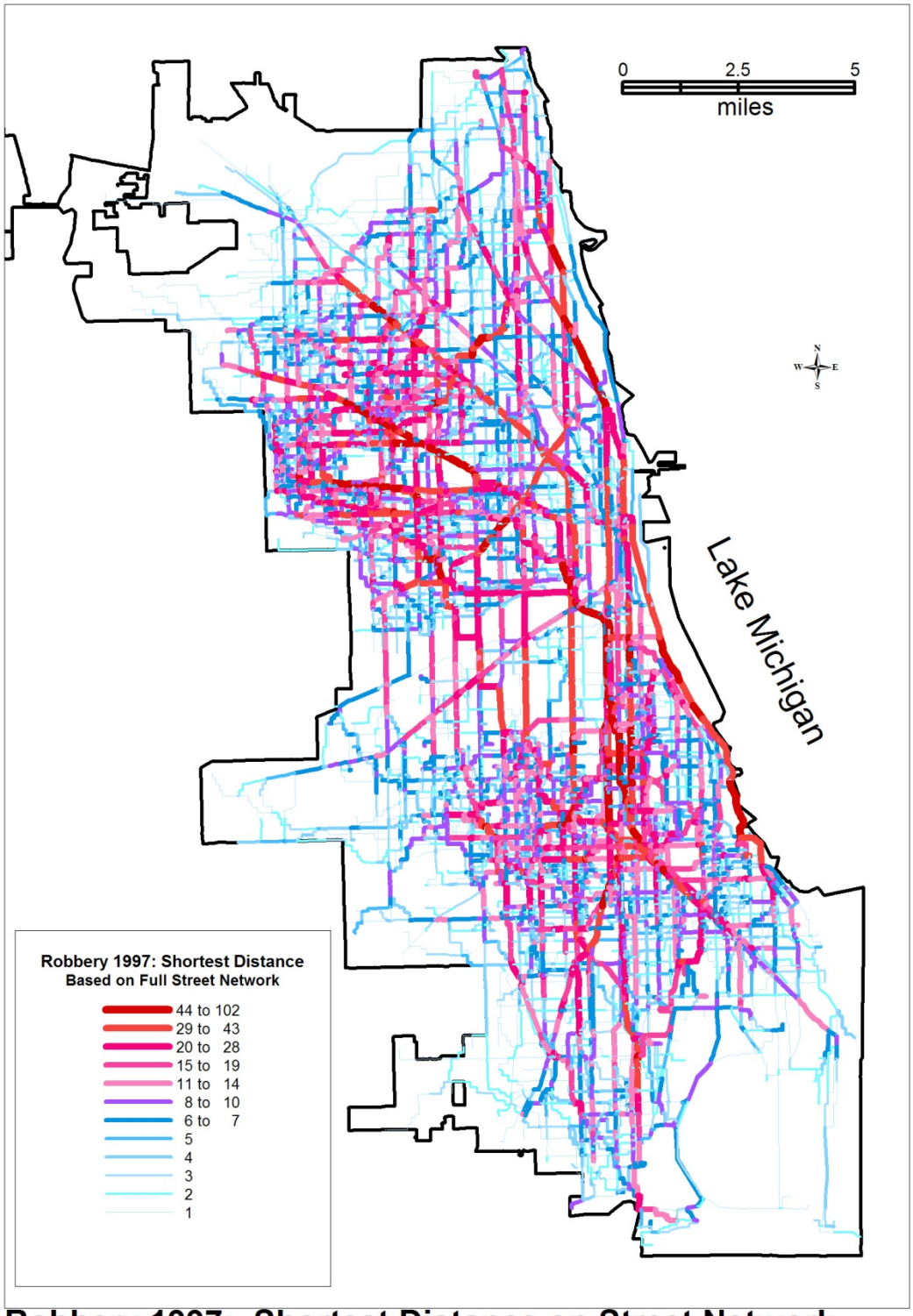
1. The shortest routes on a street network. For each zone-to-zone pair, the shortest path was calculated.

2. The Network load. Network load counts the number of trips over each street segment regardless of origin or destination and sums these.

Both the shortest routes and the total network load can be based on time or cost rather than distance.

First, all inter-zonal robberies in 1997 were mapped along Chicago's street network by shortest distance (Figure 31.6). The 4000 trips were counted along each of Chicago's 51,000 street segments and mapped as a network load (see Chapter 30). As the width and color changes from blue to red in Figure 31.6, the number of trips that passed over a segment increased. However, this map is difficult to interpret and lacks credibility. Much of the load is along small side-streets. Diagonal streets are emphasized and expressways are ignored because they usually are not the shortest route in terms of distance. Also, travel in the wrong direction on a one-way street is possible since only distance was used to calculate the shortest path. The CPD did not believe this to be a useful map.

Figure 31.6:



Robbery 1997: Shortest Distance on Street Network

Source Chicago Police Department Cartography: Richard Block, Loyola University Chicago

The same inter-zonal links were mapped again along using the Chicago modeling network, but weighting segments only by distance (Figure 31.7). While this resulted in a greatly simplified map, it still lacked some credibility. Expressways are rarely the shortest distance, therefore, their use is under emphasized. The algorithm resulted in an over emphasis on diagonal main streets. Some connected segments looked like a stair case following along Chicago's grid of main and secondary streets from one high incident neighborhood to another on the west and southwest sides.

Distance did not seem to be a good representation of travel routes. Given that police records include time of incident and travel time along Chicago's road network is available, and that *CrimeStat* allows for analysis by travel time, I re-conceptualized travel cost as shortest time rather than distance.

Shortest Time or Shortest Distance?

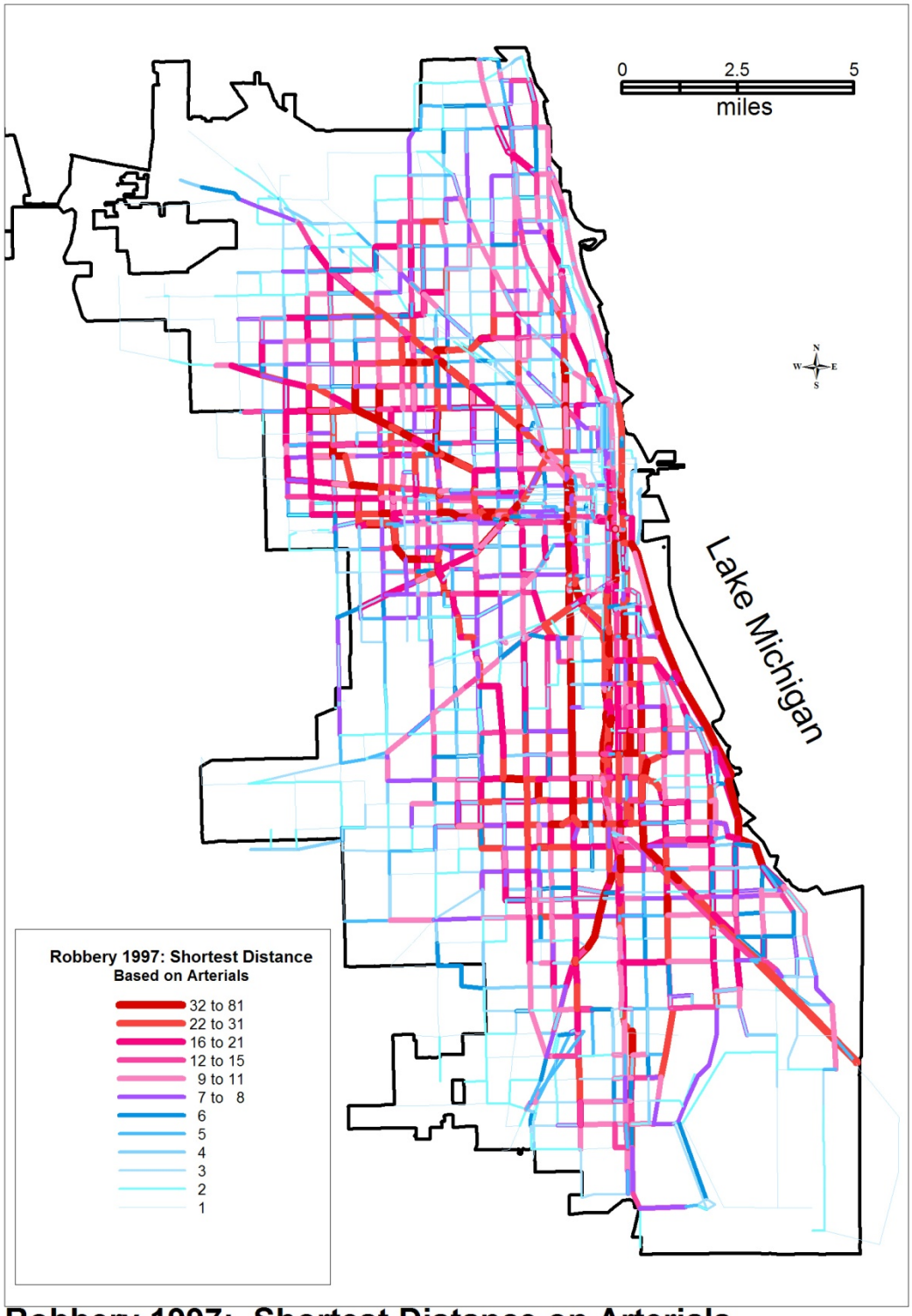
What does distance measure? Traveling ten miles during Chicago's evening rush is quite different than at midnight. However, the two blocks from my house to the nearest convenience store is unaffected by the time of day and little effected by the mode of transportation. While distance appears to be a straightforward measure, it is not. For close distances, it specifies knowledge space or the location of routine activities. Further from home, it is related to a lack of knowledge but is also an inaccurate measure of the cost of travel. Better measures than distance are often available. All U.S. major metropolitan areas map travel time by time of day on major streets, feeder streets, and expressways using modeling networks (see chapter 30). These maps along with police data on time of incident can be combined to realistically describe shortest travel time rather than shortest distance.

The Chicago Area Transportation Survey (CATS) divides the day into eight time periods based on travel demand. Whether a crime trip was intra- or inter-zonal was unaffected by time of day (χ^2=7.07 sig=.421 in 1998). Not surprisingly, the robber's daily travel cycle was different than the general population. In 1998, robbers showed little demand for travel in the morning rush hour period (6 AM to 10 AM). Of the remaining trips, about half (46% in 1998) occurred from 8 PM and 5:59 AM. These overnight trips are the subject of the analysis presented here.

Overnight robbery trip load

Overnight network load was mapped on Chicago's arterial roads and expressways according to both shortest distance (Figure 31.8 left) and shortest time (Figure 31.8 right).

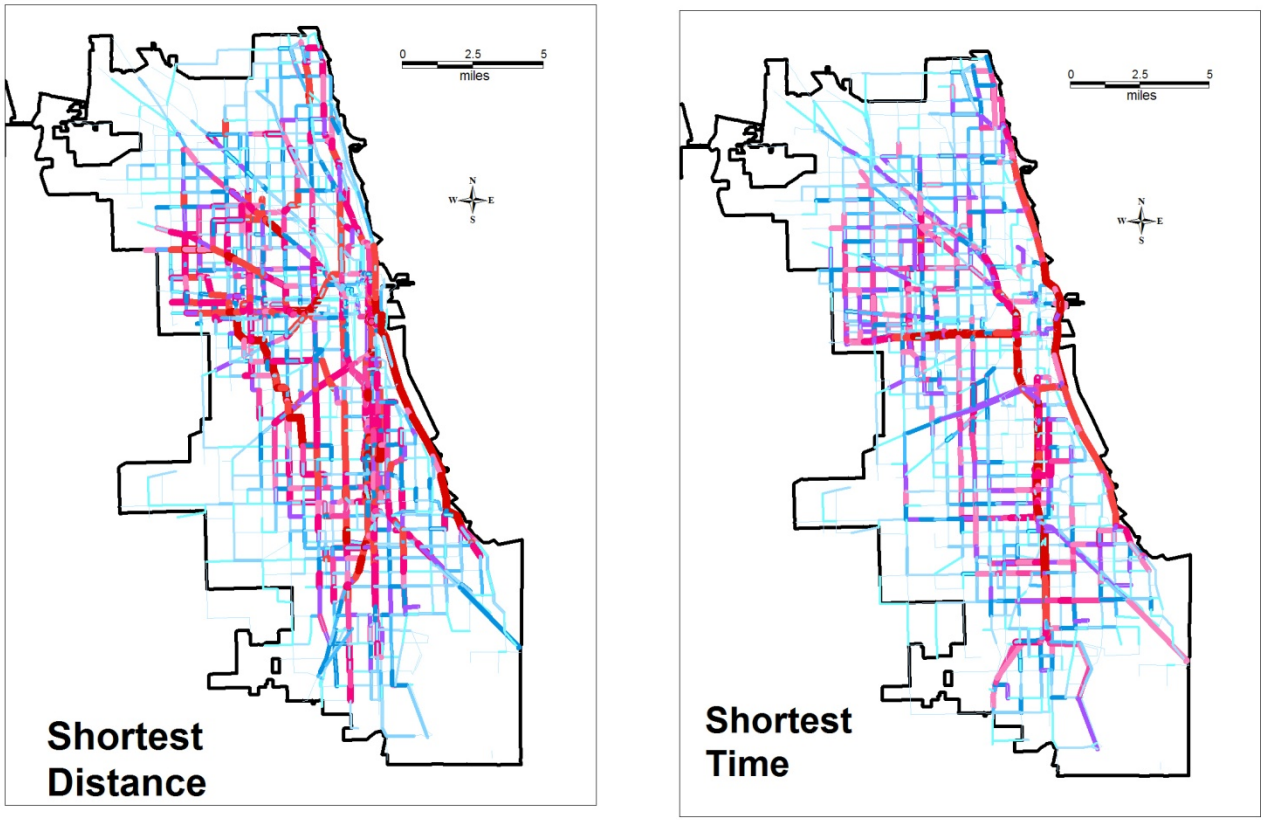
Figure 31.7:



Robbery 1997: Shortest Distance on Arterials

Source Chicago Police Department Cartography: Richard Block, Loyola University Chicago

Figure 31.8



Robbery 1998, Offender Travel Network for Incidents Occuring from 8 PM to 5:59 AM. Shortest Travel Time & Distance Compared

The two maps are very different. Expressways are rarely included in the shortest distance between zones. Much of the travel is on diagonal surface streets. However, if travel time is taken into account, many of the trips are on expressways and on Lake Shore Drive. This is probably a more realistic description of longer distance trips.

In moving from a complete street network to a simplified network using distance as an impedance to a time-based network, the description moves from an unrealistic and probably un-interpretable map to one that probably corresponds to the routes taken by offenders. Does this add to police knowledge? Of the 10,763 mapped segments in the network, 65.1% had no predicted trips assigned to them. Two percent of the road segments, those with 15 or more trips, contributed 20.2% of the 16,162 robber's movements across road segments. These were typically arterial roads or expressways. By identifying these streets as those most likely to carry crime trips, these 'hot street' segments could become a focus for police patrol or for intervention to prevent crime.

Conclusions

Feasibility & Advantages of Crime Travel Demand Modeling

The police already collect information on the location and time of incidents and the home address of arrested offenders. Can this information be utilized to describe and predict the travel patterns of Chicago robbers? *CrimeStat*'s trip distribution module was used to describe zonal patterns of travel for all known 1997 Chicago robbery offenders. Around 30% of Chicago robberies were committed near to the offender's home. For these a zonal model cannot predict travel patterns. For other robberies, a time-weighted travel pattern resulted in a more credible description than one based on distance.

The key to analyzing the robber's travel pattern is to reconsider the meaning of distance. Close to home or work, distance represents a knowledge space and an opportunity space, a place the offender knows in which he or she spends a lot time. This is an area where the benefits of knowledge may outweigh the costs of possible capture or it may simply be where the offender hangs out.. Further away, shortest distance is a poor representation of travel cost. In major metropolitan areas, a better representation is shortest travel time. Combining travel time of day with time of incident, results in a more realistic travel pattern.

These intra- and inter-zonal links are a new way to look at the relationship between offender and incident. However, they need some representation before they are useful to the police for tactical analysis or crime prevention. In my discussion with the Chicago Police

Department, a network load map seemed to be most useful. Network load summarizes the number of crime trips that passed over each segment in a road network.

Limiting analysis to robberies occurring overnight (8PM to 5:59 AM), 1997 travel patterns were a good predictor of travel distances, intra-zonal robberies, and network load in 1998. However, 1997 travel patterns only weakly predicted specific links between traffic analysis zones. For 1998 incidents, a trip distribution model (using Poisson regression of the zonal count of robbers' homes and incident locations, and an impedance function) modeled the overnight travel links between home and incident. Substituting a lognormal impedance function that better matched the observed overnight robbery pattern resulted in predictions that were nearly as good as the 1997 observed travel patterns. A combination of these predictions with analysis of travel patterns over several years might eventually result in an excellent zonal prediction of crime travel patterns.

Crime travel demand analysis is complex and time consuming and requires a relatively powerful PC with a large memory capacity. Is it worth it? Yes. Information on crime trips is automatically gathered by the police, but it is not fully utilized. However, unlike transportation planners, police are generally concerned with the short term and with acute rather than chronic problems. They work on an existing street network rather than planning for the future. Crime travel demand models may better serve the police as short term descriptions rather than long term predictions and can probably be used to describe the effect of specific police interventions such as road blocks or drug interdictions. The crime travel demand model along with a GIS can identify hot street segments–those segments that are most likely to be on the travel routes of offenders and most useful for intervention to prevent crime.

For researchers, on the other hand, a crime travel demand model is a good way to ask long-term, structural questions. If the travel patterns remain relatively constant over time, then these relationships can be modeled using a limited number of variables. The result is a way to compare different metropolitan areas as well as a way to look at the same metropolitan area over different time periods. It is a framework for analysis that is broader than just a journey-to-crime type of description.

L**imitations to Crime Travel Demand Modeling**

There are also limitations to the model:

1. Only crimes with at least one known offender are analyzed. To the extent that offender travel patterns in unsolved crimes are different than those with known offenders, travel patterns will be misrepresented.

2. The model works best if records are gathered in such a way that the address of an offender home can be linked to the address of an incident.

3. The travel demand model assumes that the offender's home address is accurate. Offenders may not have a stable address or may give a false address.

4. The travel demand model assumes that offenders travel directly from home neighborhood to incident neighborhood; many probably do not.

5. The crime travel demand model is an aggregate model, not a individual one. It predicts travel from the center of one zone to the center of another. It cannot predict specific trips or the behavior of specific offenders and cannot predict travel within a zone.

6. The model must be crime and city specific. Chicago robbers were much more likely to attack close to home than those in Baltimore County (Chapter 28) or Las Vegas (Chapter 32). Because these homes were distributed throughout the city, the travel patterns of Chicago robbers were much less focused on single target zones than in the other test sites.

7. The study of Chicago was limited to incidents that occurred in the city of Chicago. It does not model travel patterns of incidents occurring outside the city and can say nothing about them.

8. The data available from the Chicago Police Department did not allow for a test of travel mode used. It cannot be assumed that criminal trips use the same modes of transportation as non-criminal trips.

Chicago is a city of isolated neighborhoods. Even nearby neighborhoods may be *terra incognita*. Crime travel follows the pattern of neighborhoods. In Chicago, many robberies occur very close to the home address of the offender. The crime travel demand model cannot analyze these crime trips because each zone is represented by a single point. In some impoverished neighborhoods, robbery is very common. An offender can opportunistically attack on any block. Even when offenders travel they tend to stay nearby their home neighborhood. The isolation of robbery in the a few neighborhoods results in a downtown that is relatively free of incidents and crime trips that are relatively short.

Chicago is a mature city. Neighborhoods change slowly. Large scale changes in housing, poverty, or attractors do occur and include the destruction of public housing, widespread gentrification and the replacement of rail yards with upscale housing. With these

changes come new opportunities for crime and changing crime travel patterns. These may be predicted with the crime travel demand module.

References

Block, R. Brice, D, & Galary, A. (2007) The Journey to Crime: Victims and Offenders Converge in Violent Index Offenses in Chicago. *Security Journal*, April 2007.

Brantingham, P. & Brantingham, P. J. (1984). *Patterns in Crime*. Macmillan Publishing: New York.

Brantingham, P. & Brantingham, P.J. (1990). *Environmental Criminology*. Waveland Press: Long Grove IL.

Felson, M. (2002). *Crime & Everyday Life* (3rd Ed). Sage: Thousand Oaks, CA.

Rossmo, D. Kim (2000). *Geographic Profiling*. CRC Press: Boca Raton Fl.

Sampson & Raudenbush, (2001). *Disorder in Urban Neighborhoods: Does it Lead to Crime?* National Institute of Justice, Washington D.C.

Wright, R. T. & Decker, S. H. (1997). *Armed Robbers in Action: Stickups and Street Culture*. Northeastern University Press, Boston

Chapter 32:

Case Studies in Crime Travel Demand Modeling II:

Application of Travel Demand Behavior Model to Crime Data from Las Vegas, Nevada

Dan Helms
Scytale Consulting
Reston, VA

Table of Contents

Chapter 32:

Case Studies in Crime Travel Demand Modeling:

II - Application of Travel Demand Behavior Model on Crime Data from Las Vegas, Nevada

In this chapter, a case study of crime travel demand in Las Vegas, NV is discussed. Originally written in 2004, it is presented in order to illustrate how crime travel demand modeling can be applied to a primarily auto-oriented city.

Introduction

Strategic crime forecasting has for many years relied on a limited and simplistic suite of methods to predict approximately where future events may occur in broad strokes. Extrapolation of percentile change is probably the most commonly used means of forecasting future crime frequencies, based on the notion fundamental to all predictions, that the future will resemble the past. Unfortunately, this method is completely unable to cope with changes in the demographics, population, and social makeup of a jurisdiction.

For a number of years, innovative crime analysts and criminologists have looked to other disciplines outside the study of criminal behavior for methods of predicting how the future will unfold. Economics, epidemiology, meteorology, and biology have all offered significant contributions as their more sophisticated and creative methods for foretelling future frequencies have been adapted to criminology with varying degrees of success.

Transportation modeling is the most recent external science to suggest potential means of predicting criminal behavior. The success of travel demand modeling in the civilian world of transportation behavior has presented us with another possible technique which could be adapted to forecasting crime. Travel demand modeling offers a set of algorithms for estimating not only how much activity will occur in a given region, but also how offenders will travel across the jurisdiction to commit their crimes. This model has been implemented in the *CrimeStat* software application for use against crime data.

In this study, we will review the application of this model against data from the metropolitan Las Vegas area over a period of three years.

The Las Vegas Metropolitan Area

The Las Vegas metropolitan area is comprised of Clark County, Nevada, and several independent municipalities within it. The Las Vegas Metropolitan Police Department (LVMPD) serves Clark County (in the capacity of a Sheriff's Office) as well as the City of Las Vegas (in the capacity of a municipal police department). Although the vast majority of the land area, population, and businesses within this area are policed by the LVMPD, there are three other significant jurisdictions - the City of North Las Vegas, the City of Henderson, and the City of Boulder City, each having their own police department.

In addition to these important sibling agencies, several other law enforcement agencies have overlapping jurisdiction within areas principally policed by the LVMPD - the Paiute Tribal Police, the Southern Pacific Railway Police, the Nevada Highway Patrol, U. S. Air Force Security Police, U. S. Air Force Office of Special Investigations, Federal Bureau of Investigation, Veteran's Administration Police, and others. Although these agencies perform valuable police functions, the LVMPD unquestionably deals with the vast majority of crime in the area, making it an attractive candidate for offender travel research.

In many ways, Las Vegas resembles an island. Surrounded by barren desert, with very few roads entering or leaving the city, it is an urban oasis in a sparsely populated desert wilderness consisting of largely impassable terrain. This geographic position and isolation make Las Vegas highly interesting from the perspective of a transportation (or crime trip movement) modeler.

Another unique feature of the Las Vegas area is the highly transient nature of the population, which falls into three discrete categories:

1. First, the resident population consists of some one million persons, approximately 880,000 of which live in the jurisdiction of the LVMPD (the remainder being served primarily by Henderson and North Las Vegas). These permanent residents are the mainstay of the community and the source for demographic data used by the census bureau and planning agencies.

2. Second, we must consider the visitor population, consisting of some 35,000,000 - 40,000,000 persons per year. On any given day, between 100,000 and 500,000 visitors will be staying in the Las Vegas area, a critical factor in transportation, demography, and crime! These tourists sometimes act as crime importers (e.g., criminal street gangs from neighboring Californian cities often visit Las Vegas for weekend mayhem or more professional criminal purposes); in most instances, however, they serve as a pool of prey for local criminals.

3. Third, and finally, there is a substantial homeless population in Las Vegas, drawn by the seasonally warm climate and the ease with which this city can be reached as a destination. Although not famous for a "friendly" attitude toward the homeless, these persons are protected by law enforcement in Las Vegas and are well served by many charitable social institutions and services. Because Las Vegas is also an easy place to sin, homeless individuals with drug, alcohol, and gambling addictions often gravitate there; the possibility of "winning big" and instantly reversing a life of misfortune also weighs in the consideration of many homeless who choose to make their base in Las Vegas. However, due to the inability to accurately measure a "home" location for these persons when they do commit crimes, few of these have been represented in this study.

This study will focus on the criminal movement behavior of the resident population of the greater Las Vegas metropolitan area.

Source Data Provenance and Organization

Data concerning the Las Vegas metropolitan area was provided by the Las Vegas Metropolitan Police Department's Investigative Division. Often, researchers underestimate the severe difficulties and chronic shortcomings of law enforcement data. Thanks to a first-rate records management system (RMS) and a voluminous tactical database repository, the Las Vegas Metropolitan Police Department's data presented relatively few problems; however, geocoding accuracy issues, missing data fields from *modus operandi* tables, and erroneous arrestee home locations result in sources of error that can contaminate analysis. These had to be overcome before any analysis or testing of new methods was possible.

Crime report data for the LVMPD is maintained in an SQL-Server 7.0 database constructed by the Printrak (now owned by Motorola) company, makers of the Law RMS (LRMS) police records management system used by Las Vegas, among others. This repository currently houses many hundreds of thousands of crime reports, field interviews, and other critical police data in a well-organized, relational database.

Crime reports are filled out by either sworn officers (when taken in the field) or by station personnel (when reported in person at an LVMPD substation or city hall). These paper reports include ample MO detail and descriptive information in compartmentalized, "force-choice" fields, as well as substantial expository narratives. "Forced-choice" fields are also typically supplemented by "Other" options which can then be individually explained, to deal with very unusual crime behaviors, descriptions, or details.

At the end of each shift, officers submit their reports to their sergeant for review. After a quick check to ensure the most basic levels of data quality and integrity, the reports are then placed in a mailbox for pickup, which occurs several times each day and night. Reports are transferred by intradepartmental couriers to city hall where they are collected by the Records Section. Professional data entry specialists then meticulously type each report into the LRMS database.

The data entry process includes several validation and error trapping elements. These usually greatly enhance the completeness and accuracy of each report, but are sometimes bypassed by busy clerks. Perhaps the most significant validity check which can be bypassed is the address verification system which performs a 'brute force' match against a geofile of known, valid locations. When a matching address is entered into the system, geographic coordinates and other useful data is automatically propagated into the file. Because many crimes do not occur at valid, documented physical street addresses (crimes in remote or desert areas or in new construction zones or on buses or in taxi cabs, for example), however, data entry clerks have grown accustomed to overriding the address verification module. This is also sometimes done in the interests of speed and expediency, even when a valid, matchable address is provided in the crime report. When this happens, the resulting address must be cleaned using a data cleaning application prior to successfully matching in a geocoding operation. Once entered into the LRMS database, crime report information may be extracted through a variety of standard methods.

The LVMPD routinely downloads crime reports on a daily basis into an ATAC analytical database where crime analysts and investigators can examine and study the data without creating any drag on the primary server. The ATAC database is streamlined for analysis, and is much easier to query and analyze than the LRMS repository itself. The ATAC databases are Microsoft-compliant relational databases very similar to the MS Access database.

Data used for the Next-Generation Offender Crime Travel Model project were derived from records stored in several ATAC analytical databases created and maintained by the LVMPD Crime Analysis Section. These databases are archived by calendar year and by crime category. The archive dates for calendar year are assigned based on the year of occurrence. Crime categories are: auto crimes (including motor vehicle thefts, burglaries from motor vehicles, and criminal damage to automobiles); burglaries (including all burglary statutes); Larcenies (including all Larceny/Theft statutes); and personal crimes (including all sexual offenses, assaults and aggravated assaults, robberies and home invasions, kidnappings, and homicides).

These databases contain MO, Persons, and Vehicles tables related by event number. The MO table contains all information pertinent to the location, timing, category, and methods of each crime event; the Persons table all information on personal identification, description, and

histories, not only for suspect and arrestees, but also victims, witnesses, reporting parties, etc.; the Vehicle table all information concerning any vehicles which may be involved in the offense, including descriptive and identification information, whether the vehicle relates to the criminals, victims, or has some other relationship to the crime.

For purposes of this project, the LVMPD authorized access and transmission of the contents of the complete ATAC database inventory for the Crime Analysis Section. Of the fifty-odd databases provided, the Personal crimes databases for the years 1996 - 2002 were initially selected.

Data Screening

Three broad categories were selected from the complete data inventory provided:

1. Confrontational
2. Burglary, and
3. Vehicular crimes.

These intentionally disparate data were selected in the interests of increasing the latitude of the study. It was hypothesized that travel behavior would vary between these categories of events. Confrontational crimes included sexual assaults, robberies, kidnappings, and murders. These crimes were included in a single group as part of this initial appraisal of the effectiveness of travel demand modeling on criminal behavior even though it is obvious that the behaviors exhibited by offenders across these crime types are likely to vary. These crimes were grouped in spite of these likely differences because similarities in targeting behavior across these crimes might make them amenable to collective analysis; a hypothesis which can be tested using the techniques built into the travel demand module.

Burglaries used in this analysis included both residential and commercial burglaries, but not burglaries from motor vehicles. Only crimes in which a building or property was illegally entered for the purpose of theft were included in this study, thereby eliminating the prolific larceny category.

Vehicular crimes included both auto thefts and burglaries from motor vehicles. "Carjackings" were not specifically included, but some auto thefts in which the modus operandi followed the confrontational "carjacking" pattern may have been included when specifically statutory designations were missing to differentiate these from more typical auto thefts.

Some operational definitions of these crimes are in order:

1. Sexual assaults used in this analysis included forcible rapes with victims of either sex, as well as any other physical, sexual abuse of another person of either sex - such as digital or objective penetration, fondling, etc. - and also open and gross lewdness (e.g., "flashing"). Statutory sexual seduction ("statutory rape") was excluded.

2. Robberies used in this analysis included all robbery-related statutes in the Nevada Revised Statutes as of 2002 including home invasions (see State of Nevada, 2012).

3. Kidnappings were included in confrontational crimes, but the application of kidnapping as a statutory offense by law enforcement in Las Vegas (and elsewhere) may be counter-intuitive to some readers. Kidnapping is often attached as an additional offense to other crimes, such as robberies or sexual assaults, in any case in which the victim is forcibly moved from one location to another. This practice is used primarily as an adjunct to prosecution because kidnapping (unlike either robbery or sexual assault) is a federal crime and, in some cases, may be easier to prove in court.

4. Homicides used in this analysis included all murder statutes, as well as all manslaughter statutes. No justified homicides were included.

Once the target crime categories had been defined, separate databases for each of the three categories were compiled. Although data for several years was made available, all but three years of data were excluded from the study. Data prior to 1997 was often relatively poorly maintained and prepared and sometimes contained serious omissions which made it unreliable. Data for the year 2002 was incomplete when this study was commissioned. Although crime data for the years 1997 and 1998 was functionally reliable, socio-economic and transportation data for these years was not readily obtainable at the time this study commenced; since these data were necessary for implementation of this model, these years, too, were excluded from analysis. Therefore, only the years 1999, 2000, and 2001 were included in this study.

Because this study focuses on spatial relationships between crime event locations and criminal home locations, only solved crimes could be used. Crimes were included as "Solved" when an arrest was made - unfortunately, difficulties in obtaining data from the justice system and the long delays inevitable in the prosecutorial process made it impossible to identify crimes in which a conviction had been obtained; an arrest was the closest approximation to a reliable solution possible for this research.

Of those "solved" crimes in which an arrest was made, only those in which the offender's home address and the precise location of the crime itself were both known could be used. Even when crimes were closed by arrest and adequate data was available to geographically plot and analyze the case, some have still been excluded. Instances in which the offender and victim both live at the scene of the crime have been excluded from these analyses since no travel was involved. However, instances in which either party lived at the scene of the crime but the other did not have been retained. The reasoning behind this decision is that the decision to commit a crime at a given place does include the decision to commit a crime in one's own home. Therefore, the spatial travel (none) component of this decision should still be reflected in the model if we hope to eventually derive a valid statistical representation of offender travel behavior.

Also, crime in which the offender lived outside the study area (Clark County, Nevada) have been excluded in most cases, but not all. In some cases, "tourist" offenders may have been included when their temporary "base of operations" (i.e., local lodgings) had been recorded. In these instances, the hotel, motel, resort, or private dwelling they lived in has been used as a "home" location for purposes of originating a crime trip.

The number of cases usable for each category of crime varied significantly from year to year (Table 32.1).

Table 32.1:
Confrontational Crimes Available for Analysis

Year	Total Offenses	Usable Offenses
1999	5,272	1,080
2000	7,560	1,643
2001	3,588	991

The large increase in number of offenses from 1999 to 2000 is difficult to explain; the following substantial drop from 2000 to 2001 (52%!) is even more troubling. A similar, but inverted, discrepancy emerges in the frequency of burglaries reported during those years (Table 32.2).

Table 32.2:
Burglary Crimes Available for Analysis

Year	Total Offenses	Usable Offenses
1999	17,234	2,520
2000	12,899	2,040
2001	16,403	2,733

A final enigma, most significant of all, is obvious when we look at the frequency of auto crimes over the same three-year period (Table 32.3).

Table 32.3:
Vehicular Crimes Available for Analysis

Year	Total Offenses	Usable Offenses
1999	6,871	646
2000	15,025	1,219
2001	8,349	894

These disparities are hard to account for. On the whole, 1999 had a middling number of auto thefts and confrontations but a high number of burglaries; in 2000, on the other hand, the confrontations and auto thefts radically increased (auto crimes more than doubled!), but burglaries dropped notably. Finally, in 2001, confrontational crimes dropped to the lowest levels as did auto crimes while burglaries leaped to nearly 1999 levels!

How can we explain these strange fluctuations? Given the large percentages involved, it's tempting to imagine some change in counting or reporting procedures in 2000; however, a scrutiny of the policies and procedures for the LVMPD does not seem to bear this out. Previous years (1996 - 1999) did not evince a similar wide degree of variation. The reason for these crime reporting changes remains unknown.

Is there reason, therefore, to distrust these data? For purposes of this study, the answer appears to be, "No." That is, the data used for these analyses should, even allowing for as yet-unexplained vagaries in reporting, comprise a representative sample of the reported crime activity in Las Vegas over these years.

Since forecasting the frequency of crime is a relatively minor component of the travel demand model, these numeric variations should not cause too much concern. Instead, since the focus of this model is the effective explanation and representation of the distribution of crime

trip generators and crime trip destinations (and, as a function thereof, of the crime trip paths between them), the frequencies themselves should matter little.

Reference Data

The Traffic Analysis Zone (TAZ) file for Las Vegas was selected as the optimum polygonal reference theme for this study (Figure 32.1). This file was provided by the Metropolitan Planning Office for Las Vegas, the Regional Transportation Commission. The data provided included historical data for 1999, 2000, and 2001 enabling more accurate modeling of the importance of various factors longitudinally across time. The TAZ dataset was provided in ESRI shapefile format, which is intrinsically legible to the CrimeStat application on which the model is to be built.

The TAZ shapefile includes information on housing, employment, income, population, road mileage, and a variety of subset data specific to particular types of employment (e.g., "Strip" jobs, Nellis Air Force Base employment, entertainment-related jobs, vacant properties, number of pawn shops, etc.).

An additional reference theme was needed to apply the final step in the travel demand model, the network assignment method. The Major Street Centerline file (LVMAJSCL.shp) in ESRI shapefile format was selected (Figure 32.2). Although only including arterial streets, freeways, and major thoroughfares, this transportation network layer is all that is needed to describe the vast majority of trips (of any sort) in Las Vegas. The addition of bus route information may prove a useful supplementary network to future analyses using this model.

Assignment of Crime Trips

Data from each year, by category, was assigned to a simple tabular database consisting of an identifying variable (Event Number as primary key), origination coordinates (coordinates of the offender's home address, or local base of operations in the case of external offenders), and destination coordinates (coordinates of the crime scene). These data were then combined into an *MS Access 97®* database for analysis using CrimeStat. Figures 32.3 and 32.4 show the assigned origins and destinations.

Each origin-destination pair is termed a "Crime Trip." Following the reasoning of transportation modelers, it is understood that offenders to not leave their homes, travel directly to a crime scene to commit an attack, and then return home. Instead, each "sortie" is likely to consist of several stages.

Figure 32.1:

Traffic Analysis Zones in Las Vegas

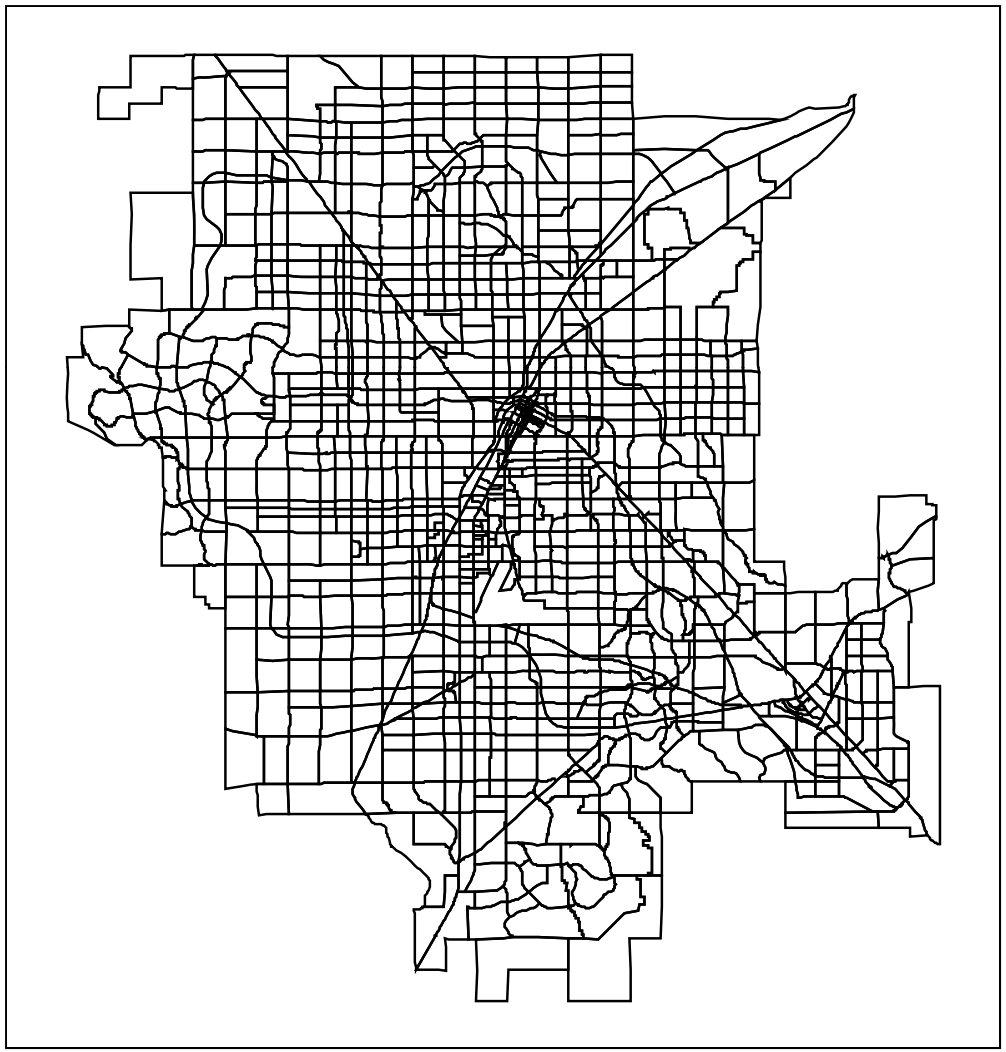


Figure 32.2:

Las Vegas Major Street Centerline Network

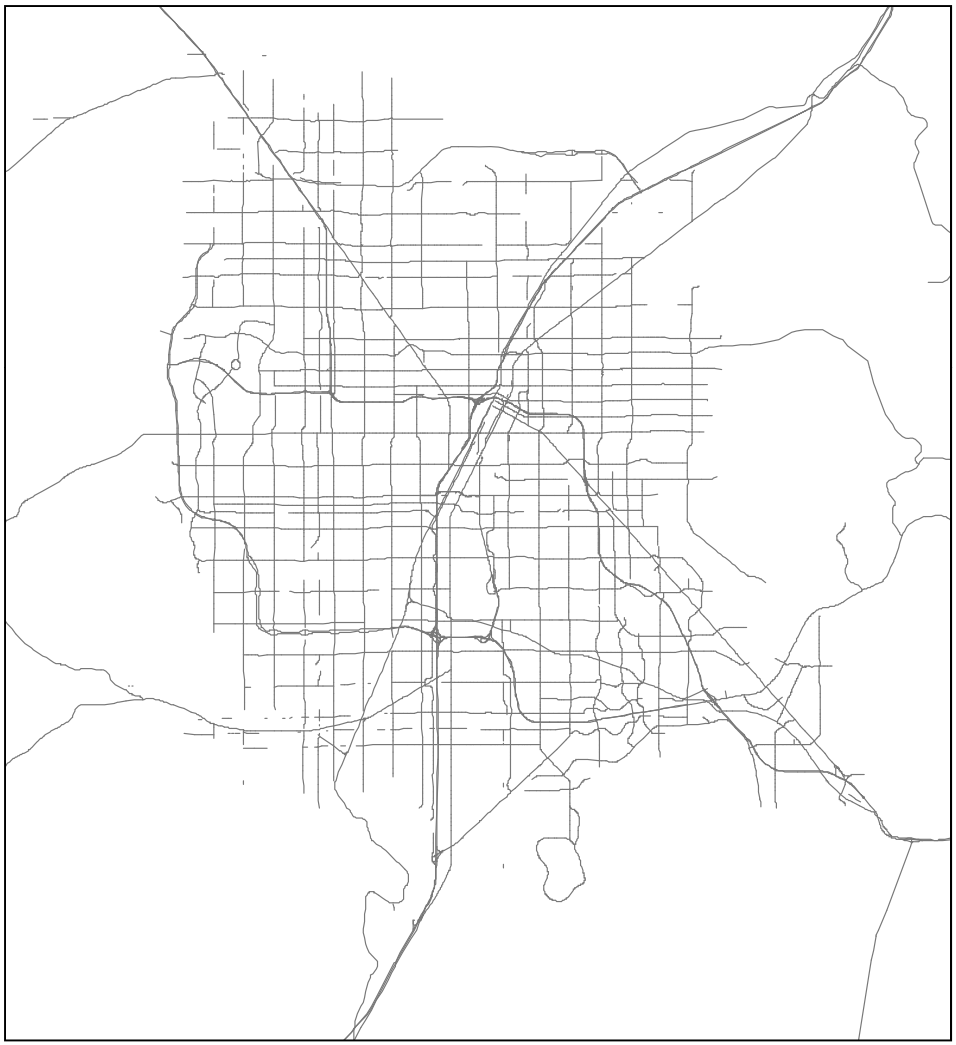


Figure 32.3:

Trip Origins: All Confrontational Crimes, 1999-2001

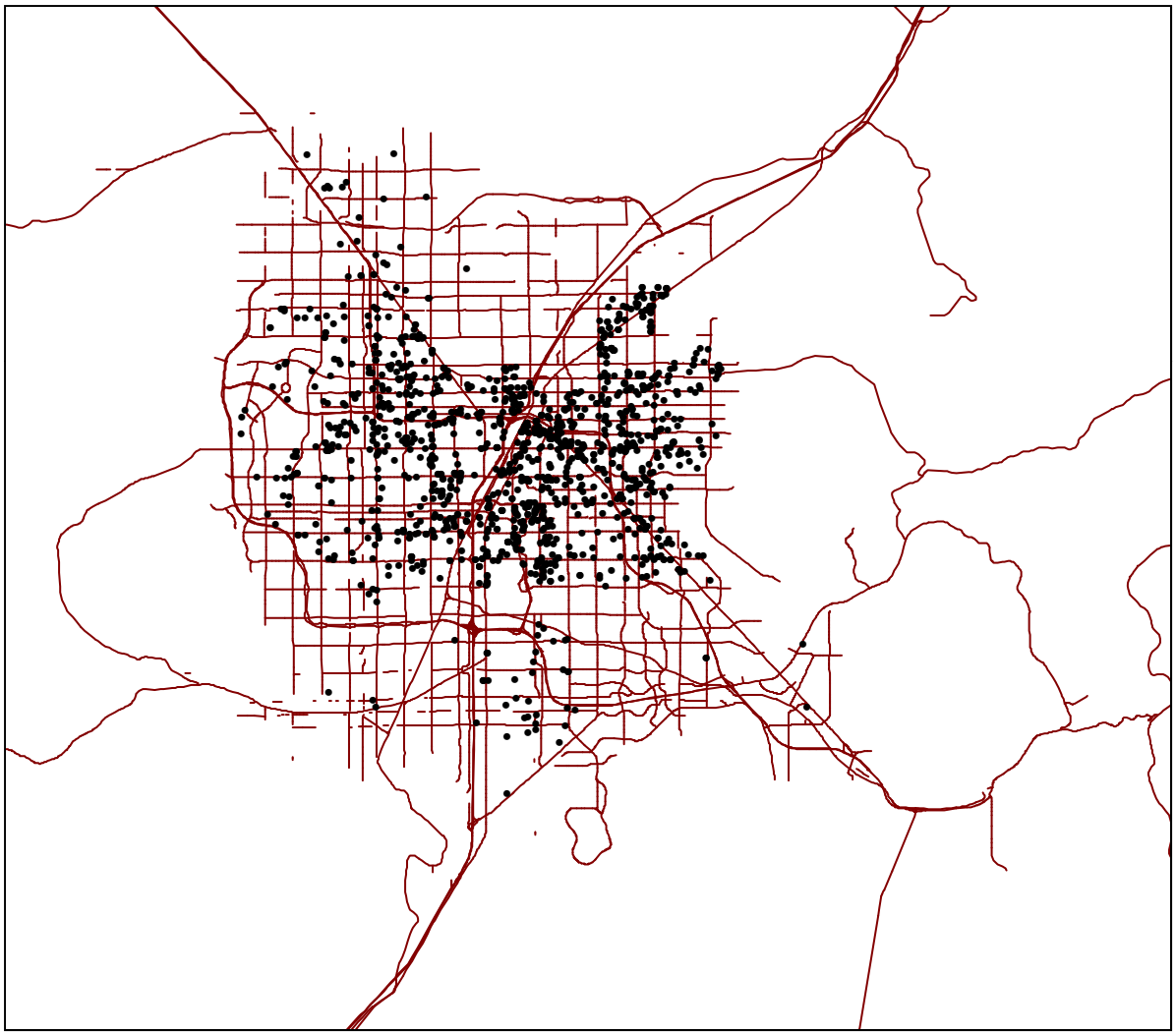
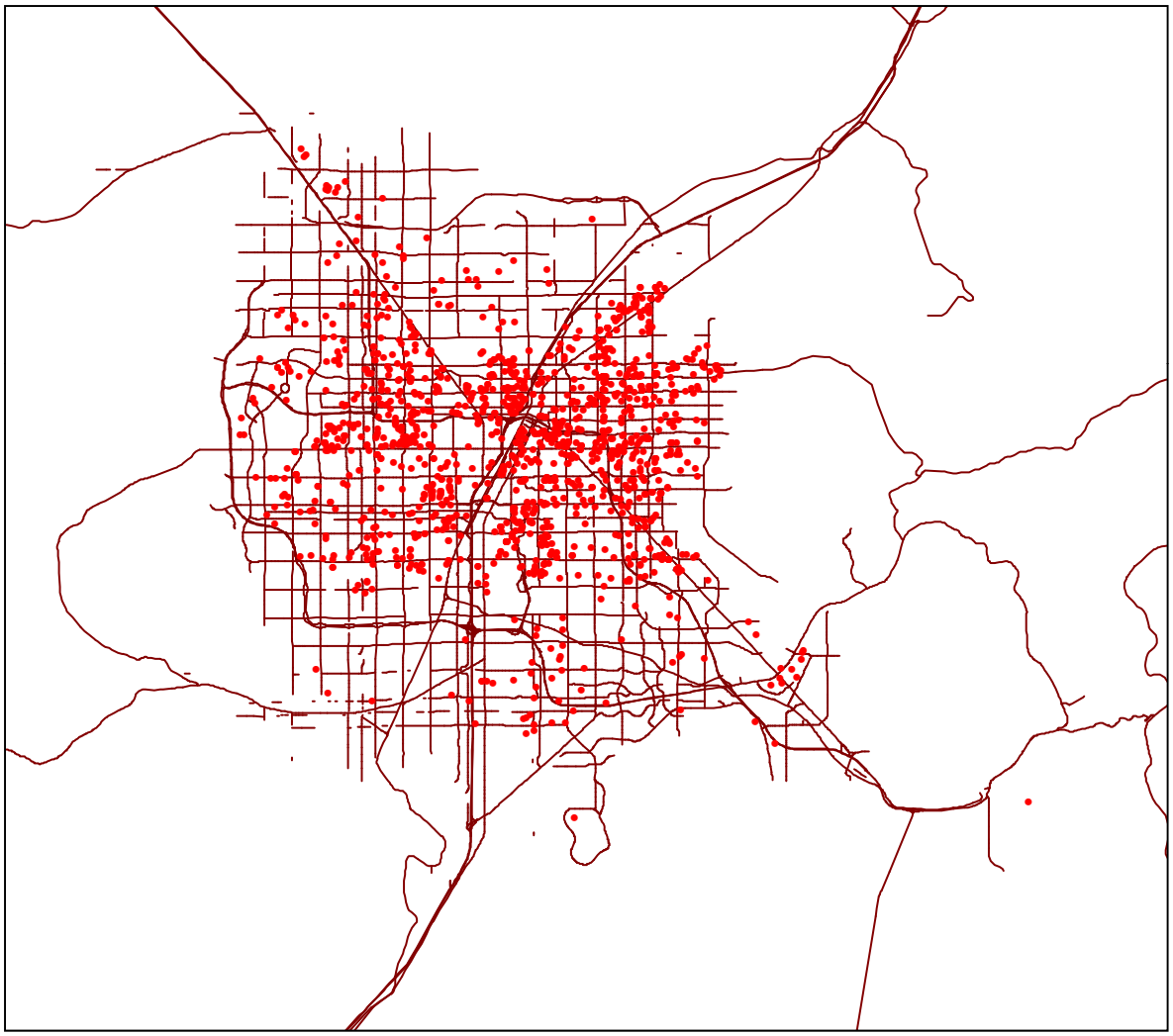


Figure 32.4:

Trip Destinations: All Confrontational Crimes, 1999-2001



For example, a sexually predatory offender may get up in the morning, leave home, drive to work (stopping for coffee along the way), then go out to lunch before returning to the office, then on his way home depart from his usual route to drive through a residential neighborhood, looking for targets for potential victims. If a promising target is observed, he may then commit an attack, then drive back toward his home area, stopping off for gas or at a drive-through restaurant on the way, before parking at his house. Although this round-trip from home to home consists of multiple destinations, some of which are repeated throughout the day, the whole journey is considered to be a single "Crime Trip".

In some cases, a single offender was responsible for many crimes. When this happens, the single origin is paired with multiple destinations, resulting in separate Crime Trips. In other cases, one crime may be perpetrated by multiple offenders. When this happens, each offender's origin is paired with the single destination, again resulting in separate Crime Trips.

While it is possible to distinctly model each Crime Trip based on precise spatial locations, the type of model used is an aggregate one. Thus, both origins and destinations of each crime trip were aggregated to the centroid of each Traffic Analysis Zone. This enables the spatial assignment of TAZ variables such as income and population to the aggregate frequencies of both origins and destinations.

This assignment is performed in CrimeStat by centroid allocation - the nearest TAZ centroid is used to assign the TAZ data to each origin and destination. This method is faster and simpler than "point-in-polygon" spatial aggregation and assignment, but should result in comparatively few mistaken assignments due to unusual TAZ polygon shape or distribution. Since crime trip data is aggregated to the zonal level, therefore, the resulting analyses and forecasts are only applicable to this level and cannot meaningfully disaggregated to a more refined resolution.

The accepted travel demand model framework contains a built-in "error factor" for external trips - that is, crime trips originating outside the study area but having internal destinations. These "external trips" were culled from the crime database during the data screening process; therefore, "External Zone" data is inapplicable to the trip generation stage of the analysis.

Trip Generation

Each origin/destination pair having been aggregated to the TAZ polygon layer, it is now possible to evaluate the relationship between socio-economic variables available in the TAZ database with the frequency of crime origins and destinations. This is accomplished through

regression modeling and may prove one of the most useful single features in the new modeling capabilities of the *CrimeStat* application.

There are two main regression options available in the software at present: Ordinary Least Squares (OLS) and Poisson.[1] The Poisson estimation also includes a separate option which allows backward elimination of variables. This option, Poisson Regression with backward elimination, was the most effective of the techniques evaluated resulting in consistently better visual fits to the data and lower residuals. This very useful step examines each variable element suggested by the analyst for its predictive value as a coefficient in estimating the frequency of either origins or destinations by TAZ.

In every case, three variables within the TAZ database for Las Vegas proved consistently useful as predictive measures:

1. Income,
2. Population, and
3. Total Employment.

The measurable successfulness of these variables to account for the predictable distribution of both origins and destinations was somewhat counter-intuitive. It was suspected prior to the application of this model that other variables would be critical predictors of crime, in particular the number of pawn shops, the number of Strip employment opportunities, and the number of Nellis AFB employment opportunities. In fact, however, all of these variables demonstrated strong multicolinearity with the three primary variables listed above. When these other, extraneous factors were excluded from the regression process, the effectiveness of the model's predictive capabilities was substantially improved.

A suggested and accepted travel demand modeling techniques widely implemented by transportation planners is the adoption of “special generator” variables to explain unusual or unique factors implicit in some areas. It was expected that Nellis AFB, the Las Vegas Strip itself, and some other seemingly significant factors would likely fill the role of "special generator;" however, results indicated that none of these were as effective in a predictive or explanatory role as income, population, and total employment.

Latitudinal forecasting of crime trip origins and destinations performed fairly well; comparison of expected versus observed trip numbers did not match particularly well but the relative distribution by TAZ was a very close match (Figures 32.5 through 32.8).

---

[1] Since this was first written, the regression capabilities have been expanded to include a variety of Poisson-type models including Poisson-Gamma, Poisson-lognormal, and Conway-Maxwell Poisson all with a spatial component. See Chapter 20.

Figure 32.5:

Relative Distribution of Observed Crime Trip Origins

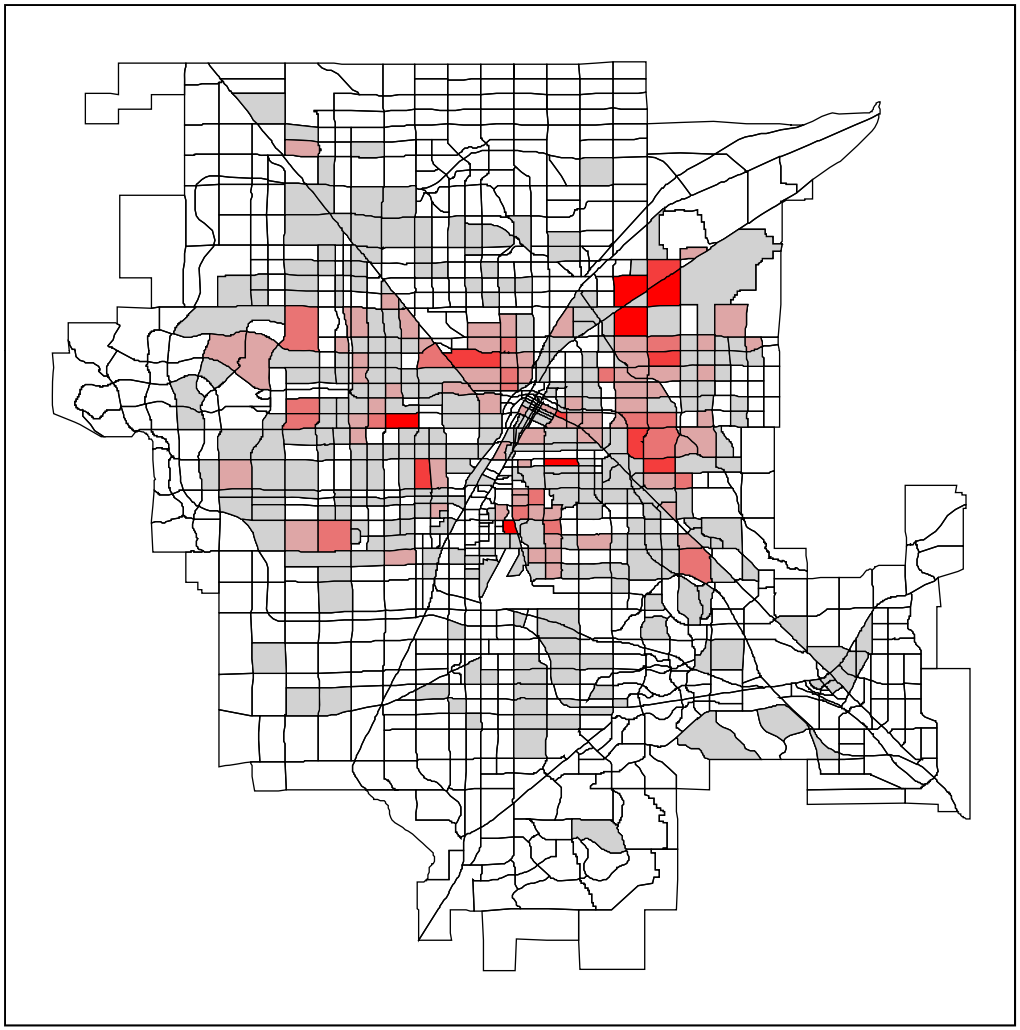


Figure 32.6:

Relative Distribution of Observed Crime Trip Destinations

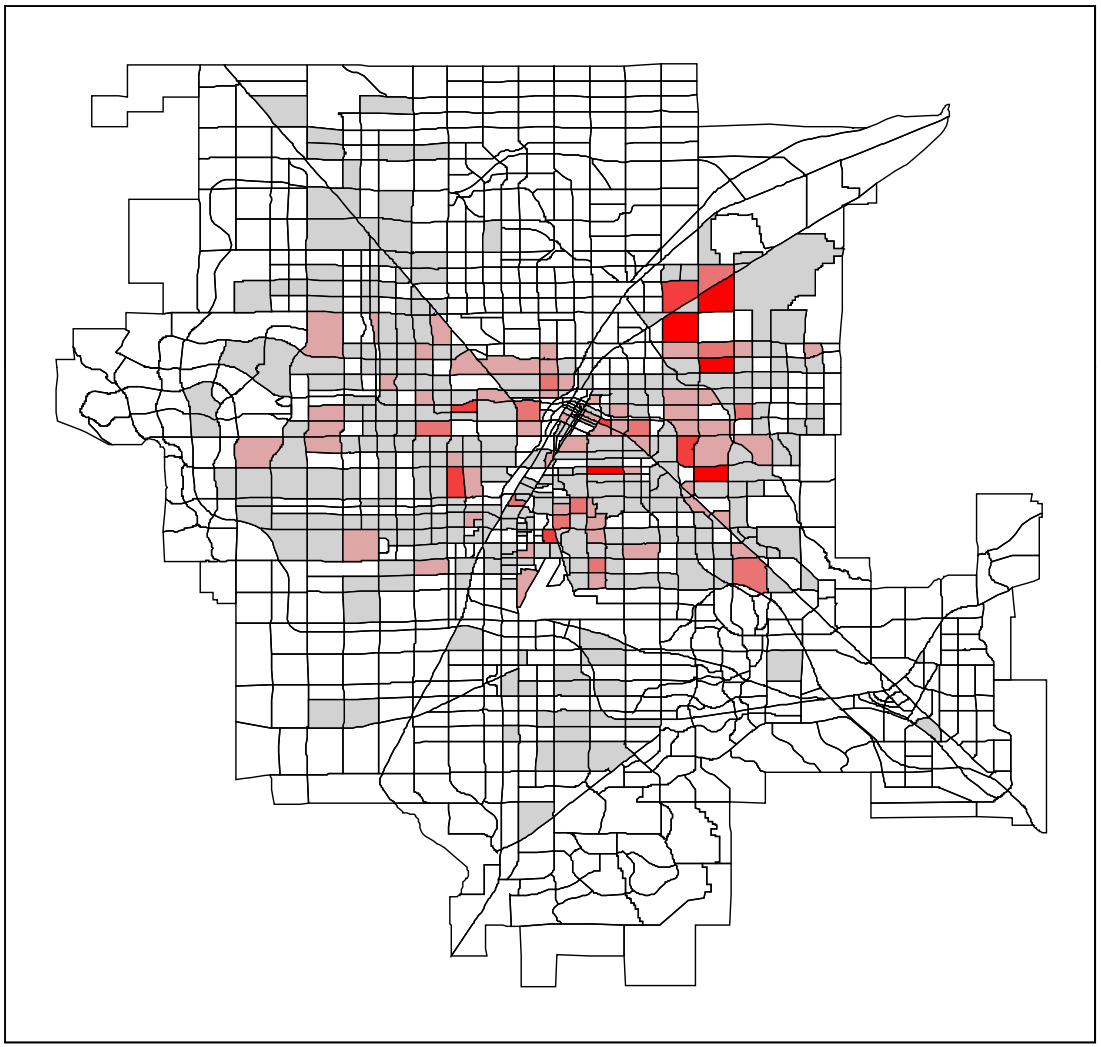


Figure 32.7:

Relative Distribution of Predicted Crime Trip Origins

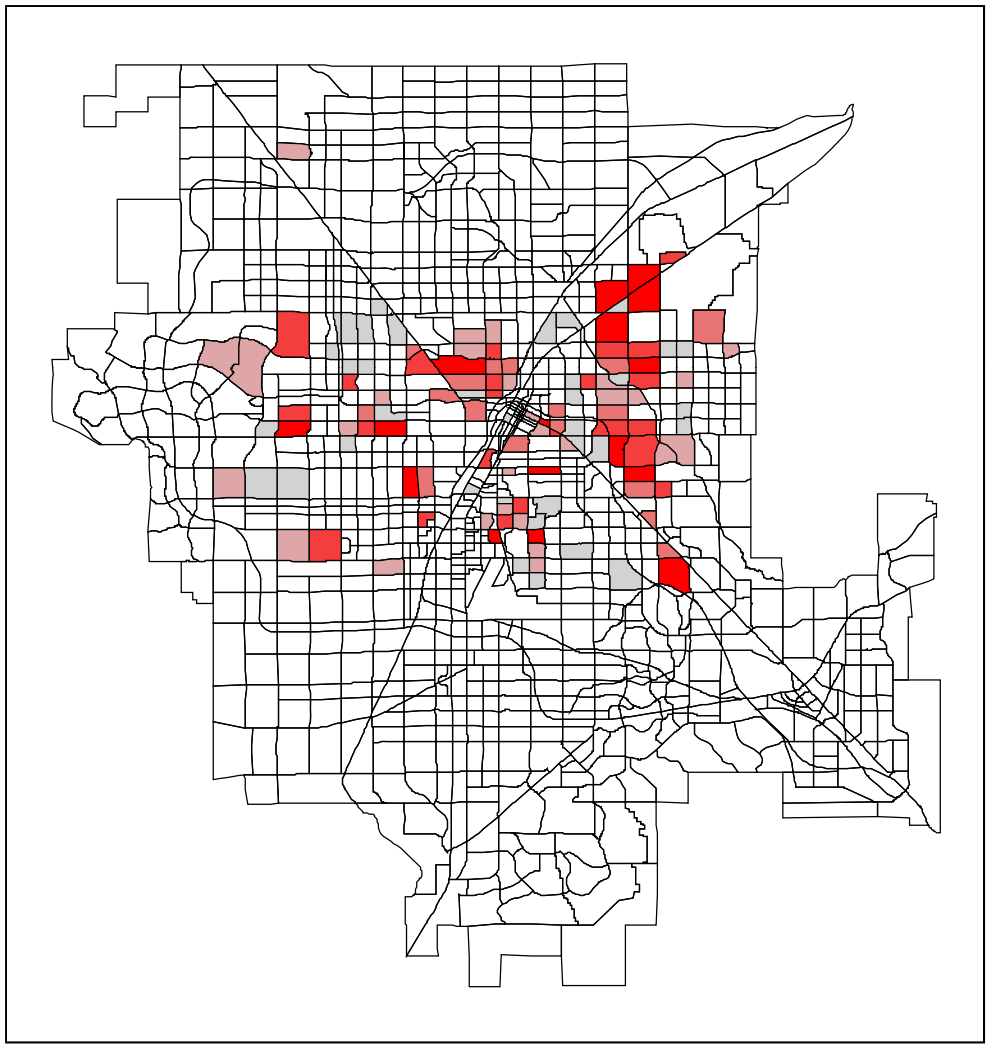
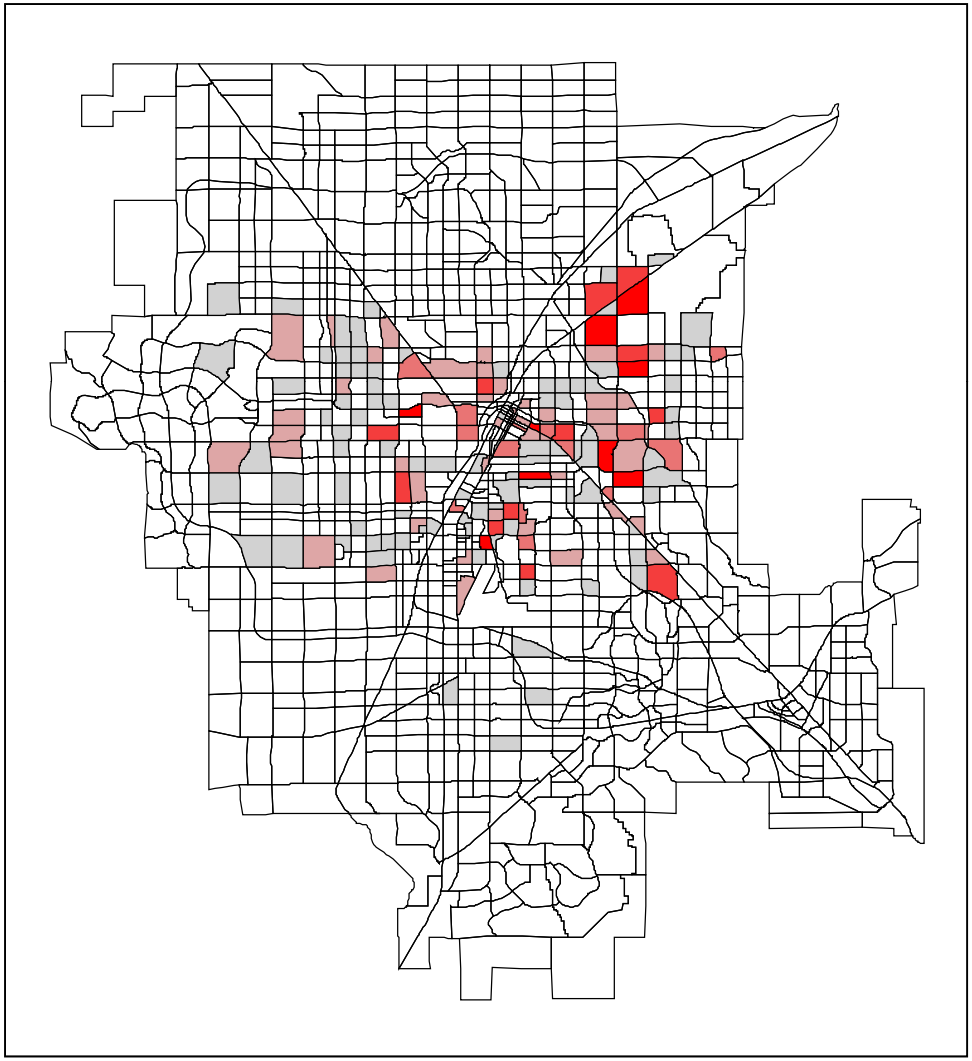


Figure 32.8:

Relative Distribution of Predicted Crime Trip Destinations



Longitudinal forecasting of crime trip frequency by data from one year to the next year performed very poorly; this is probably an artifact of the still-unexplained drastic variation in frequency between the three years considered in this study. Results from other years may exemplify very different findings.

Side-by-side comparisons of observed and predicted crime trip origins reveal some persuasive similarities, but significant discrepancies, also (Figure 32.9 and Figure 32.10). In general, relative proportions are very accurately described, but smaller producing zones are somewhat underestimated (the model seems to perform better on zones with higher productions).

A side-by-side comparison of observed and predicted crime trip destinations suggests that, proportionally, the model again performs very well, particularly on zones with higher production scores. Zones with very weak crime trip destination productions (of one or two crimes) are not as accurately depicted.

Trip Distribution

Assignment of trip links between TAZ polygons performed very well (Figure 32.11). Originally, some concern was felt that the assignment of crime events to TAZ centroids (rather than using the actual crime scene and home address coordinates) might result in significant distortion; however, this does not appear to have occurred. Compare the raw (actual) crime trip lines with the centroid-corrected trip lines to see how neatly they match (Figure 32.12). The resulting distance decay and impedance functions perform perfectly well. There are almost no discrepancies visible to the naked eye.

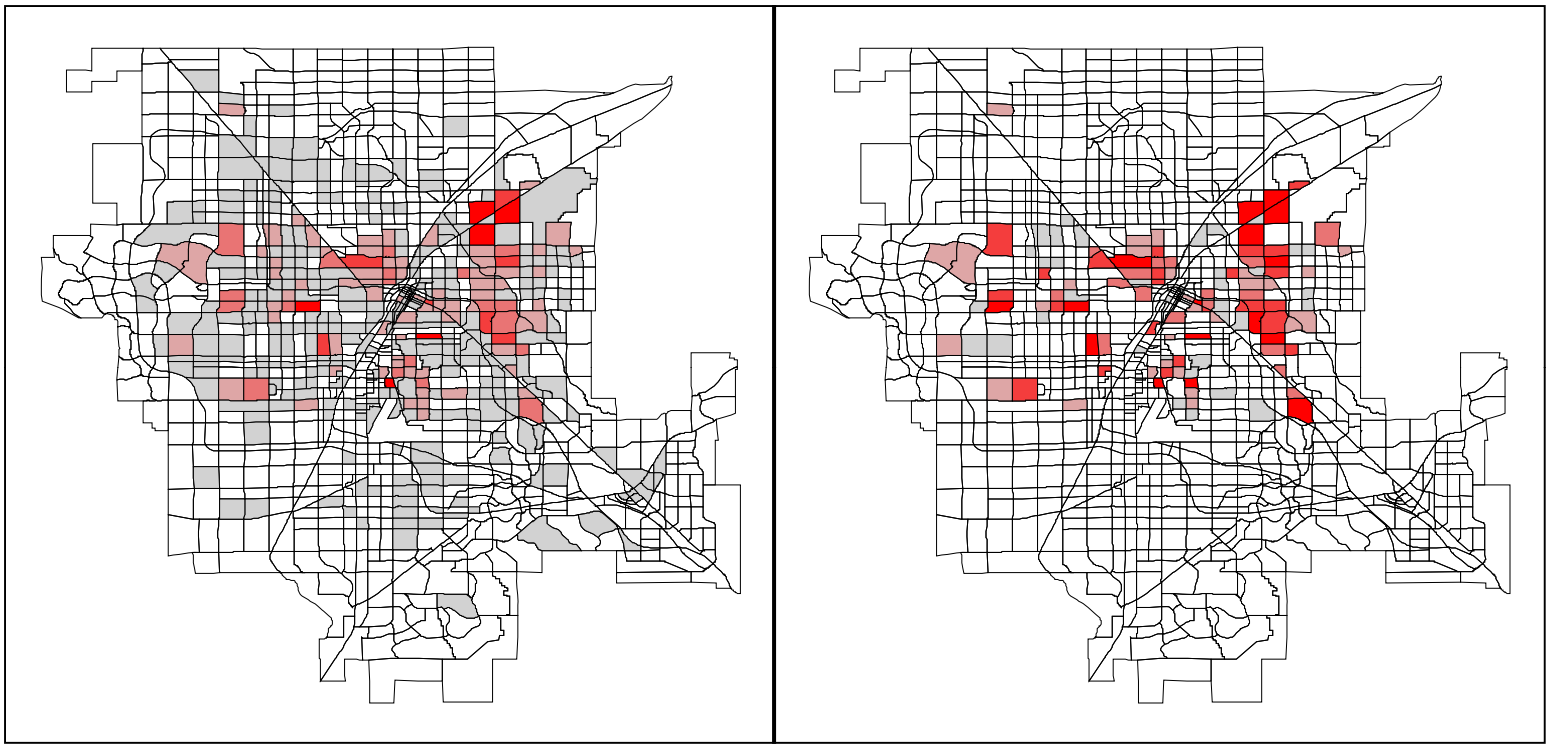
Various impedance function calculations were attempted in the course of this study. Eventually, an adaptive (100-bin) normal interpolation with 100 minimum samples was selected as the best fit. However, a negative exponential impedance function also fit well, similar to the Baltimore County and Chicago models.

Intra-zonal crime trips - those having both origin and destination within the same TAZ, cannot be displayed as lines since they have no length. Instead, they can be represented by points (Figure 32.13). Inter-zonal crime trips, on the other hand, are better displayed by lines (Figure 32.14).

Overall, intra-zonal crime trips accounted for 42% of all crime trips but only 12% of robberies, indicating a much longer "hunting range" for robbers; this may be in keeping with the hypothesis that the tourist corridors draw robbery crime trips as destinations which originate in other neighborhoods. More than 50% of sexual assaults were intra-zonal, indicating a

Figure 32.9:

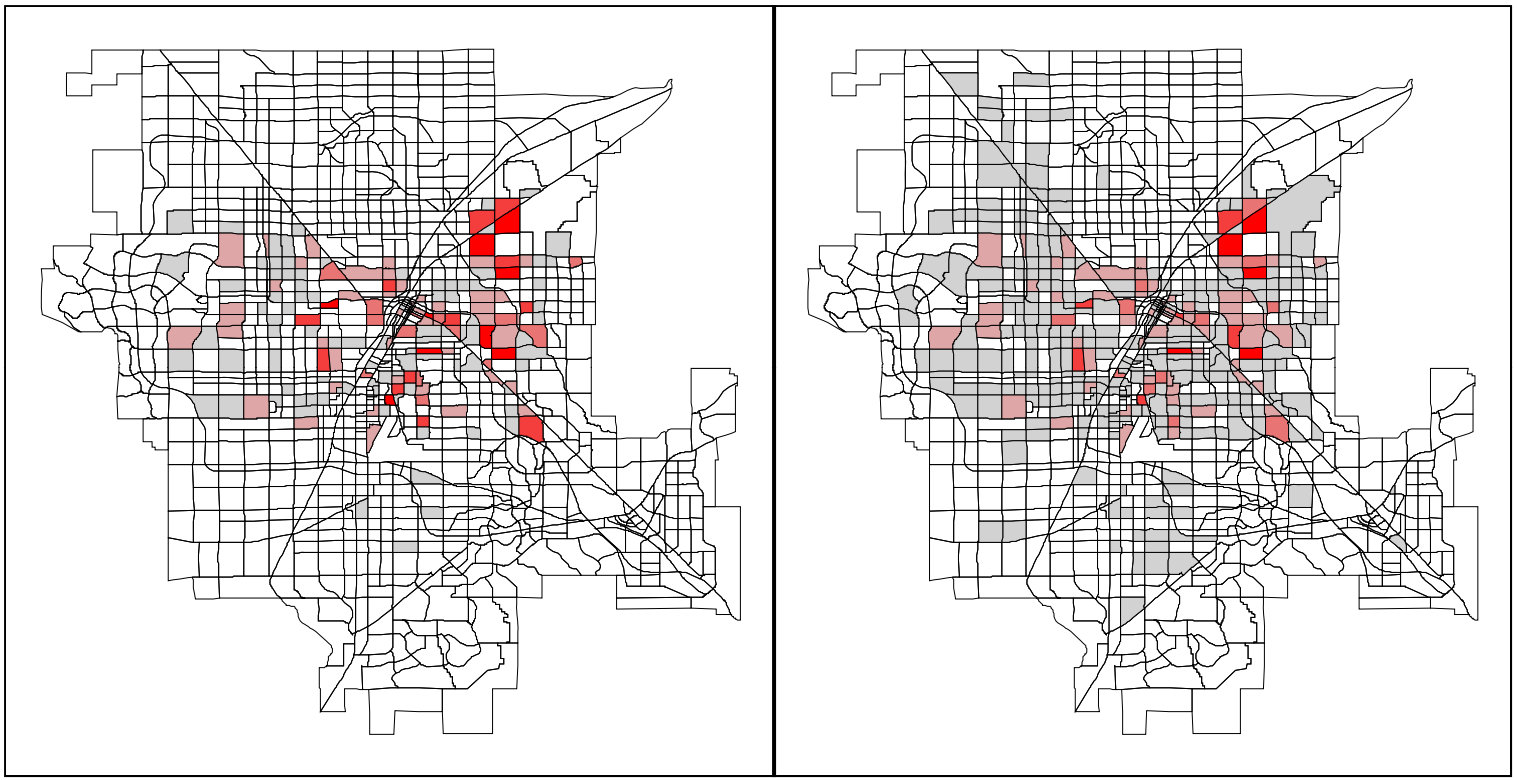
Comparison of Observed and Predicted Crime Trip Origins



Observed **Predicted**

Figure 32.10:

Comparison of Observed and Predicted Crime Trip Destinations



Observed **Predicted**

Figure 32.11:

Observed Crime Trips

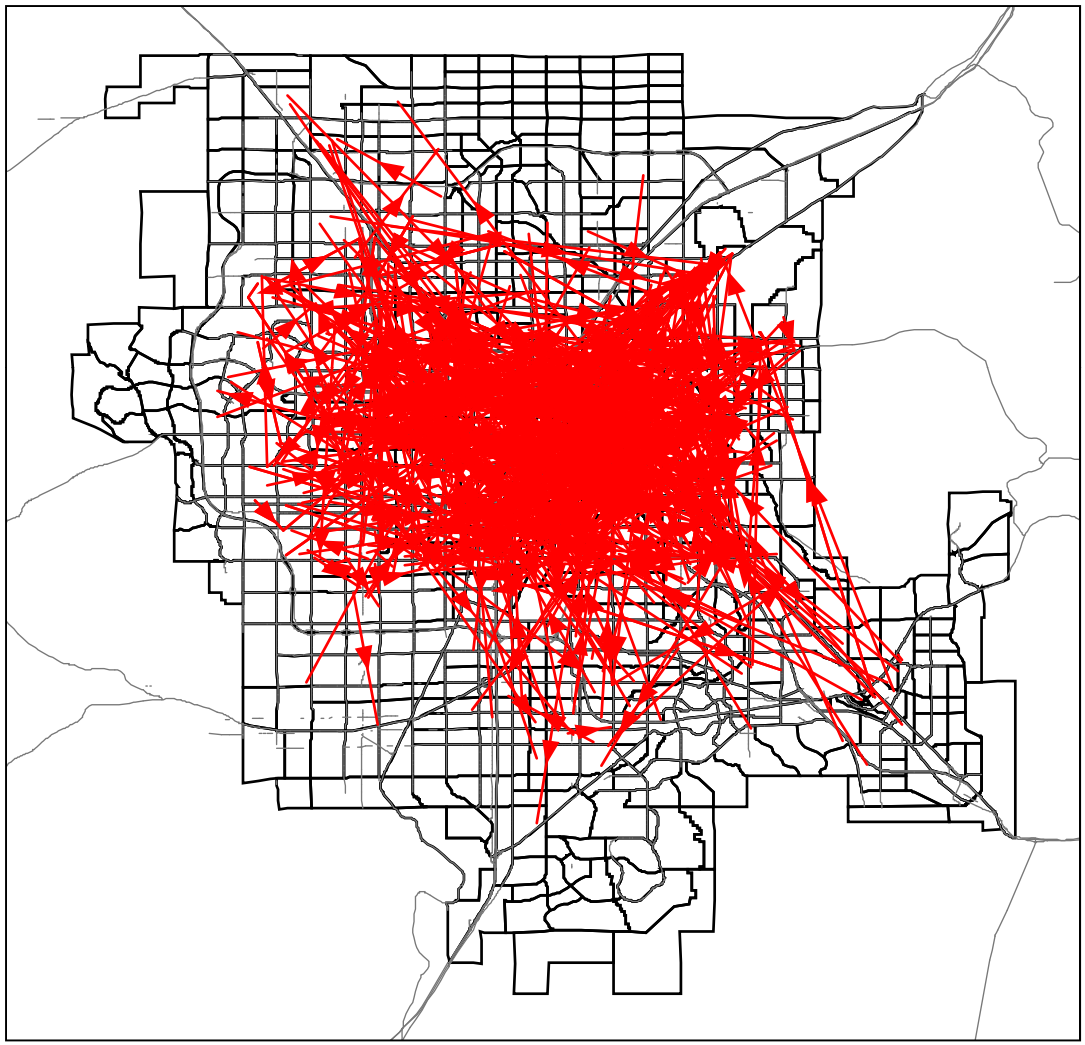


Figure 32.12:

Comparison of Observed and Predicted Crime Trip Links

Observed **Predicted**

Figure 32.13:

Predicted Intra-zonal Crime Trips

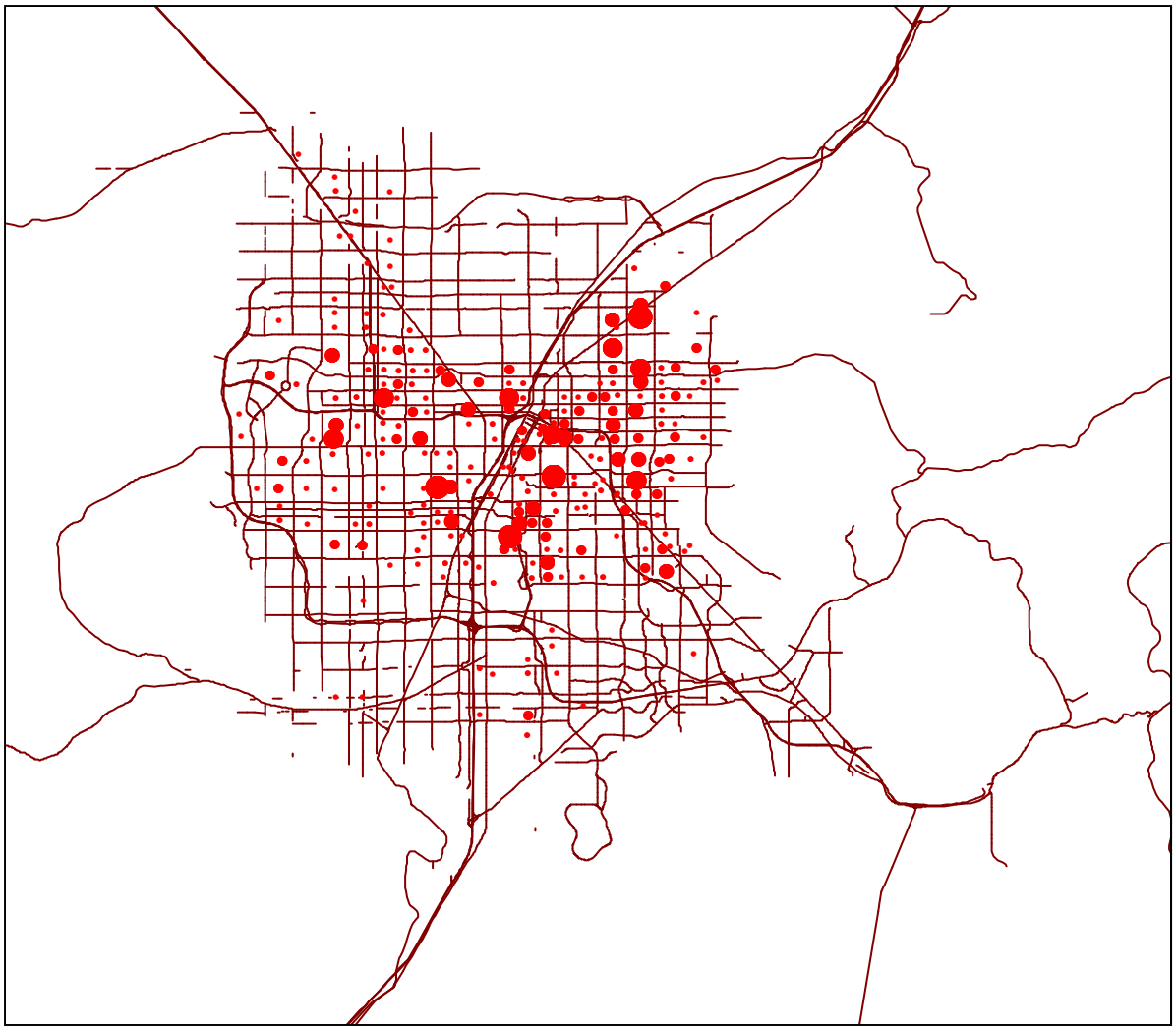
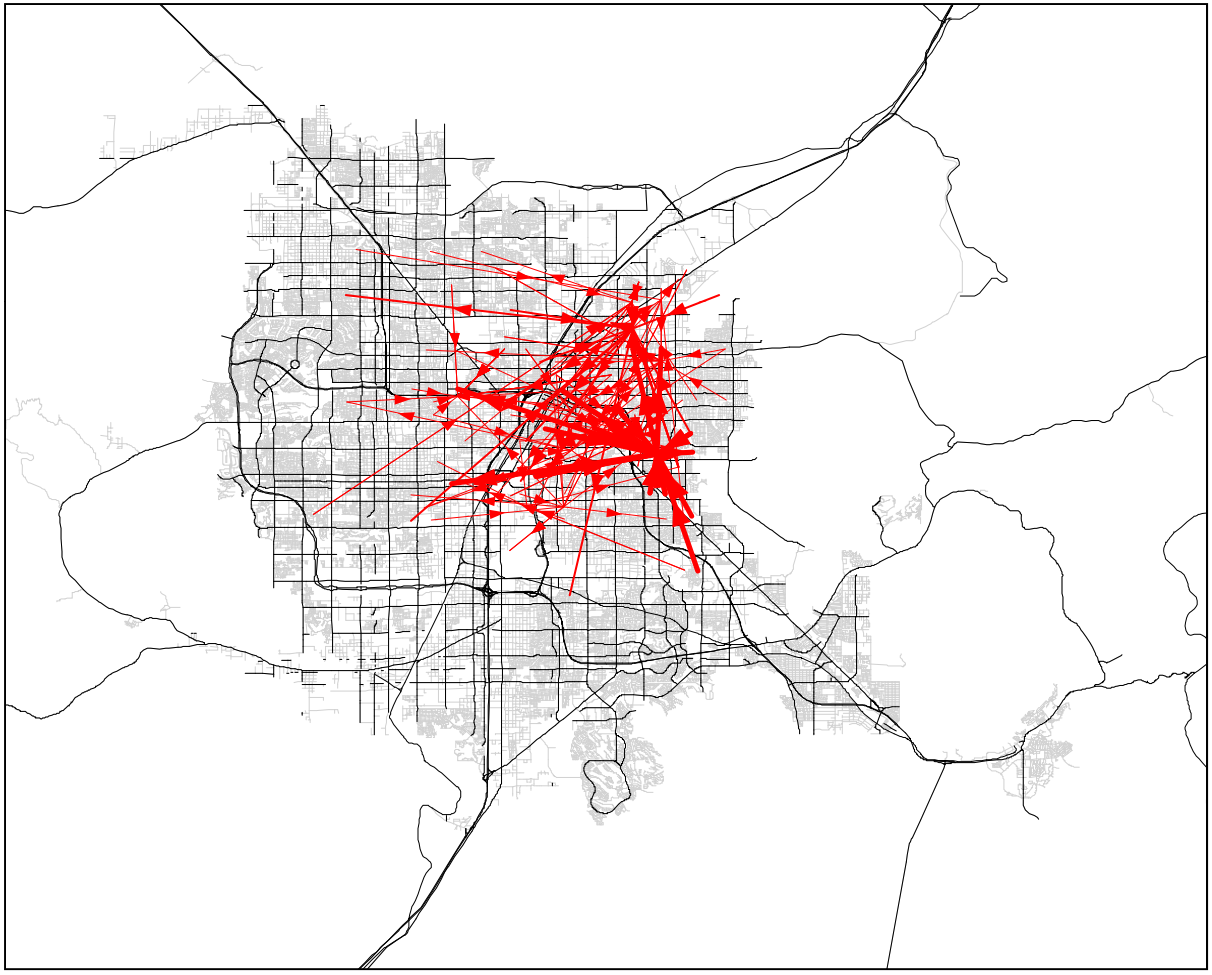


Figure 32.14:

Top 100 Predicted Inter-zonal Crime Trips

Allocated to All Streets



shorter-than-usual hunting range for sexual attackers, who seem to prefer striking in their home neighborhoods.

Mode Split

Unfortunately, the mode split portion of the travel demand model is the weakest element for the Las Vegas data. Transportation modes across metropolitan Las Vegas are varied. Typical of a western city, the overwhelming majority of residents rely on private automobiles for transportation as do many tourist visitors. However, this mainstay is supplemented by a robust bus system as well as alternate personal transportation for short trips (i.e., walking, bicycling, or scooters). The picture of automobile transportation is somewhat muddled by the higher than usual dependency on taxi-cabs and limousines for transportation by out-of-state visitors.

Data provided by the LVMPD included a field called "Method of Departure" which was intended to contain information about how the offender departed the scene of the crime which, in turn, would have been an effective way of calculating probable mode split for crime trips sampled. Unfortunately, this data field was blank in the overwhelming majority of cases (approximately 4% contained entries, and only 75% of these - 3% overall - contained apparently valid data).

Therefore, any empirical estimation of mode split for these data requires inference from other data. For example, auto theft crimes may safely be assumed to use a car to provide transportation for at least some portion of the crime trip. In other cases, the plain-text narrative includes vehicle descriptions or statements about how the offender moved that were not distilled into the correct field. Unfortunately, the large volume of cases makes recovering information from these free narratives impractical for the small number of cases in which mode split information can beneficially be derived.

Due to this lack of reliable data, only two mode split options were included in this analysis: Walking and Driving. Default impedance functions proved very acceptable for both modes: Inverse Exponential for walking trips and Lognormal for driving.

Network Assignment

The complete street centerline (SCL) file for the metropolitan Las Vegas area was available in a routable format (topologically rectified ESRI Shapefile); however, this file proved prohibitively large and unwieldy for the A* shortest-path/least-cost algorithm implemented in *CrimeStat*. Instead of the complete SCL data layer, a layer consisting only of arterial streets and freeways was used instead. This major roads file proved adequate to neatly explaining the

probable transportation path choices made by the top 100 and top 300 inter-zonal crime trips (Figures 32.15 and 32.16).

In general, the visual goodness-of-fit for predicted crime trips improved as the category of crime was narrowed. Predictions from one year to the next remained weak, probably as a result of the as-yet-unexplained radical variance in crime frequencies across all study categories. However, within discrete crime categories predictive capabilities were sometimes visually impressive.

Modeling Different Crime Types

Auto Theft Site to Recovery Site

In the case of auto thefts, an attempt was made to isolate the movement from vehicle theft site to vehicle recovery site rather than use the theft site and offender home location as the destination and origin, respectively, of the crime trip. It was hoped that this variation of the travel demand model for crime trip analysis might prove more useful for this type of data than home-based crime trips partly because more accurate location information was available for recovery sites than for home locations. Also, it was hypothesized that the theft/recovery "trip" segment might prove more representative than the home/theft trip.

Results for auto thefts appeared weak with predicted crime trips much longer than the observed (Figure 32.17). While the observed trips focused tightly on the central core areas and densely-populated residential zones, the predicted trips seemed to skirt the edges of the metropolitan area. This is possibly due to an implied overemphasis on freeway travel which may be correctible with better network allocation parameters. The median distance for observed crime trips was 2.3 miles.

Residential Burglaries

Differentiation of residential from commercial or auto burglaries was accomplished by three filtering criteria: Statute, Premise, and Zoning. Some specific Nevada Revised Statutes have been reserved for residential burglaries; burglaries in which these statues were cited were therefore accepted as residential in nature. Categorical Premise type data was provided in the MO data for each crime; when this data explicitly noted a residential site, these cases were also accepted as residential.

Some burglaries did not specifically include a residential statue or explicitly residential premise code; but were spatially located in areas of the jurisdiction reserved for residential rather

Figure 32.15:

Top 100 Predicted Inter-zonal Crime Trips

Allocated to Freeways and Major Streets

Figure 32.16:

Top 300 Predicted Inter- and Intra-zonal Crime Trips

Allocated to Freeways and Major Streets

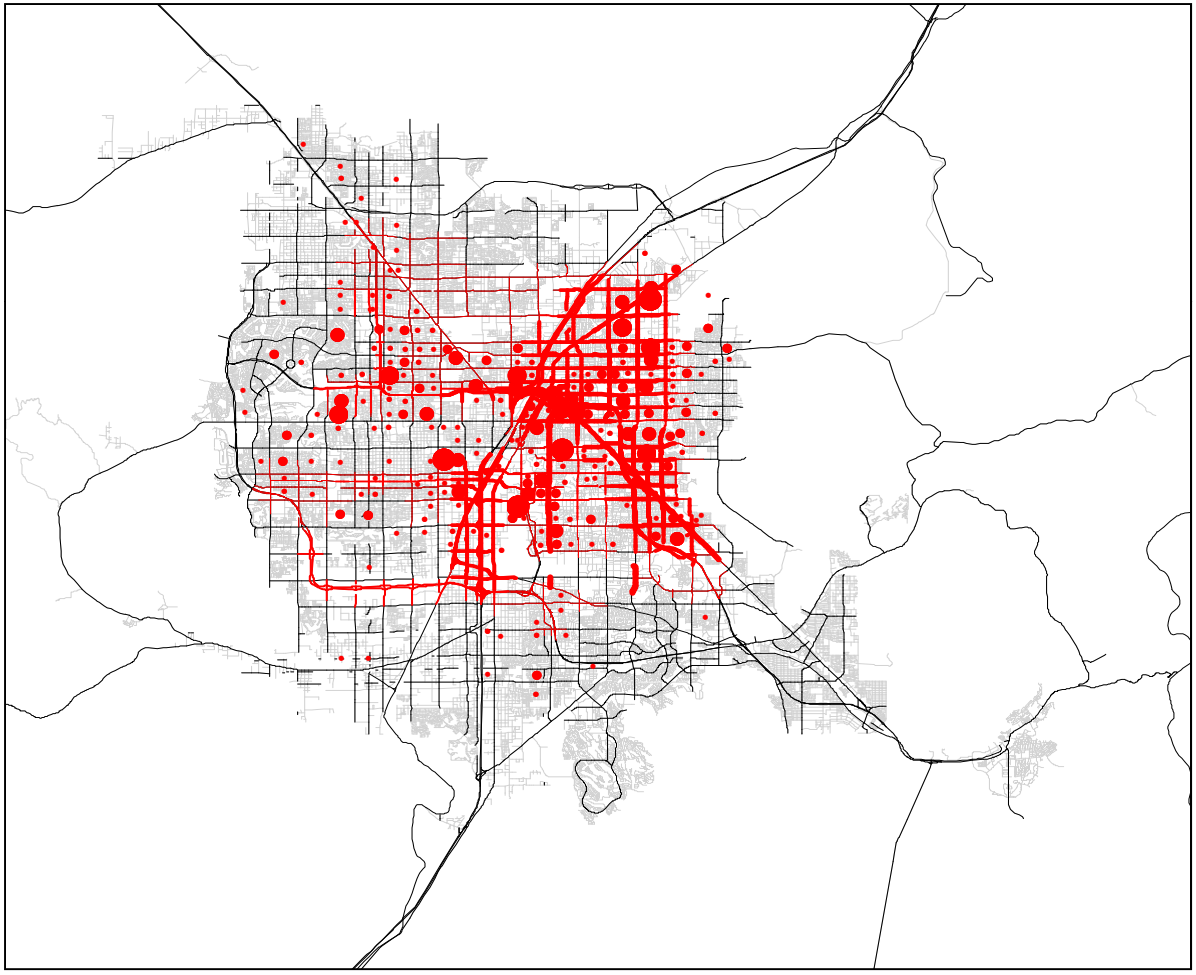
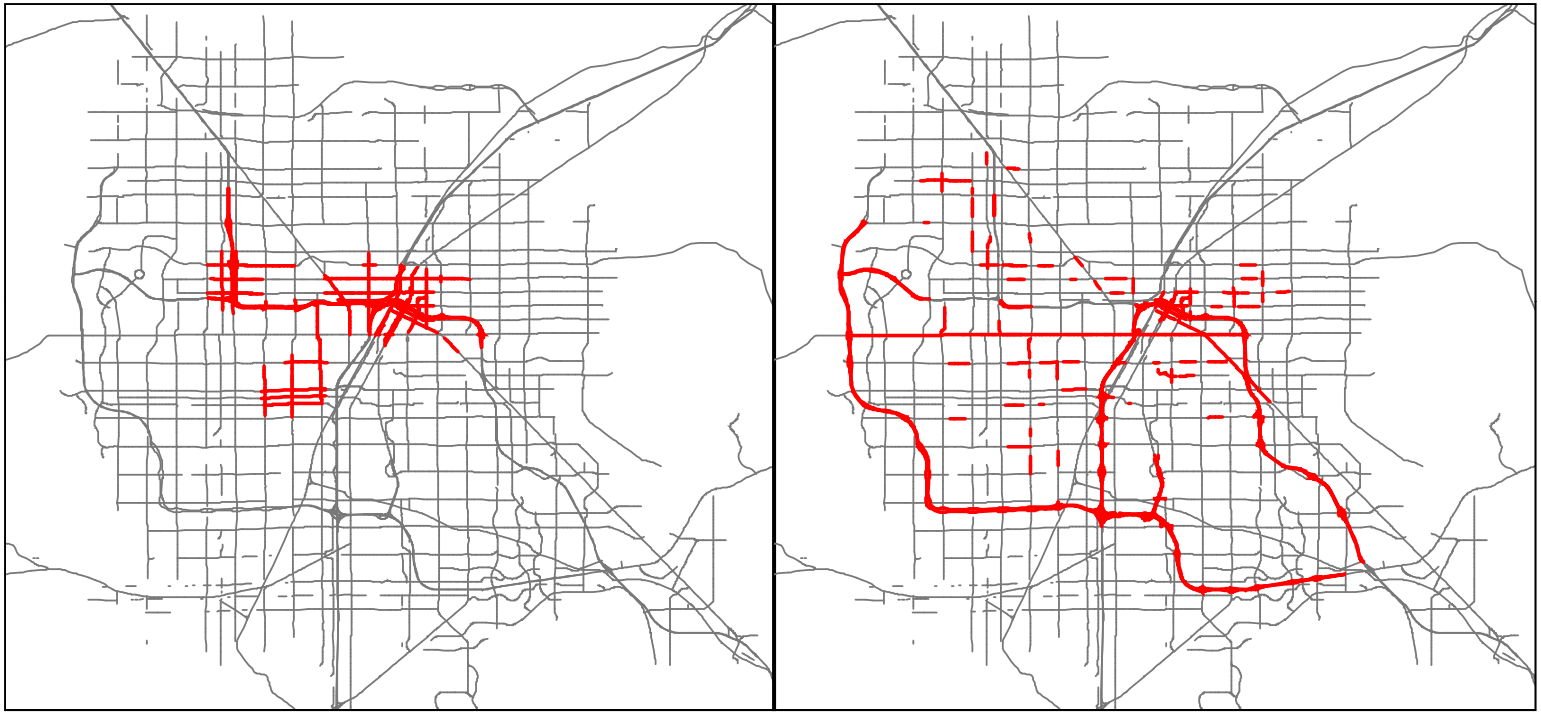


Figure 32.17:

Top 100 Observed & Predicted Auto Theft Crime Trips

Allocated to Freeways and Major Streets



Observed **Predicted**

than commercial, industrial, or other zoning purposes. These cases were therefore also accepted as residential in character.

Results for analysis of residential burglaries was more promising than for auto thefts, or for burglaries overall (Figure 32.18). While, again, observed crime trips focused on the most densely-populated residential neighborhoods and predicted crime trips were much longer and spread more far afield, this spread was much smaller than that seen in auto thefts and more closely conformed to the observed distribution. The median distance for residential burglary crime trips was 1.1 miles.

Sexual Assaults

The spatial distribution of sexual assault crime trips in many ways seemed to invert the problems seen in the predicted crime trips for auto thefts and residential burglaries. In the previous examples, an observed tendency toward centrality seemed to be confused with a predicted tendency toward dispersion toward outlying areas. In this case, however, a very nebulous, outlying distribution of observed crime trips (centering in three faint clusters around the perimeter of the central metropolitan region) was observed. The predicted crime trip distribution mistakenly emphasized central areas, and seemed to completely fail to predict the southeastern-most "cluster" of crime trips (Figure 32.19).

The large median crime trip length for sexual assaults - 3.2 miles, may help explain the relatively poor predictions of these results. Different impedance functions will probably help improve the reliability of this model against these types of crimes.

Robberies

Robbery crime trips in Las Vegas appear to closely parallel the major gaming and transportation corridors running north to south through the center of the metropolitan area (Figure 32.20). The visual fit of predicted against observed crime trips was most impressive against these cases. Although the predicted crime trip distribution appears more compact and centralized than the observed, the directionality and polarity of the two parallel nicely, and make a striking visual match. The median crime trip distance for robberies was 2.3 miles.

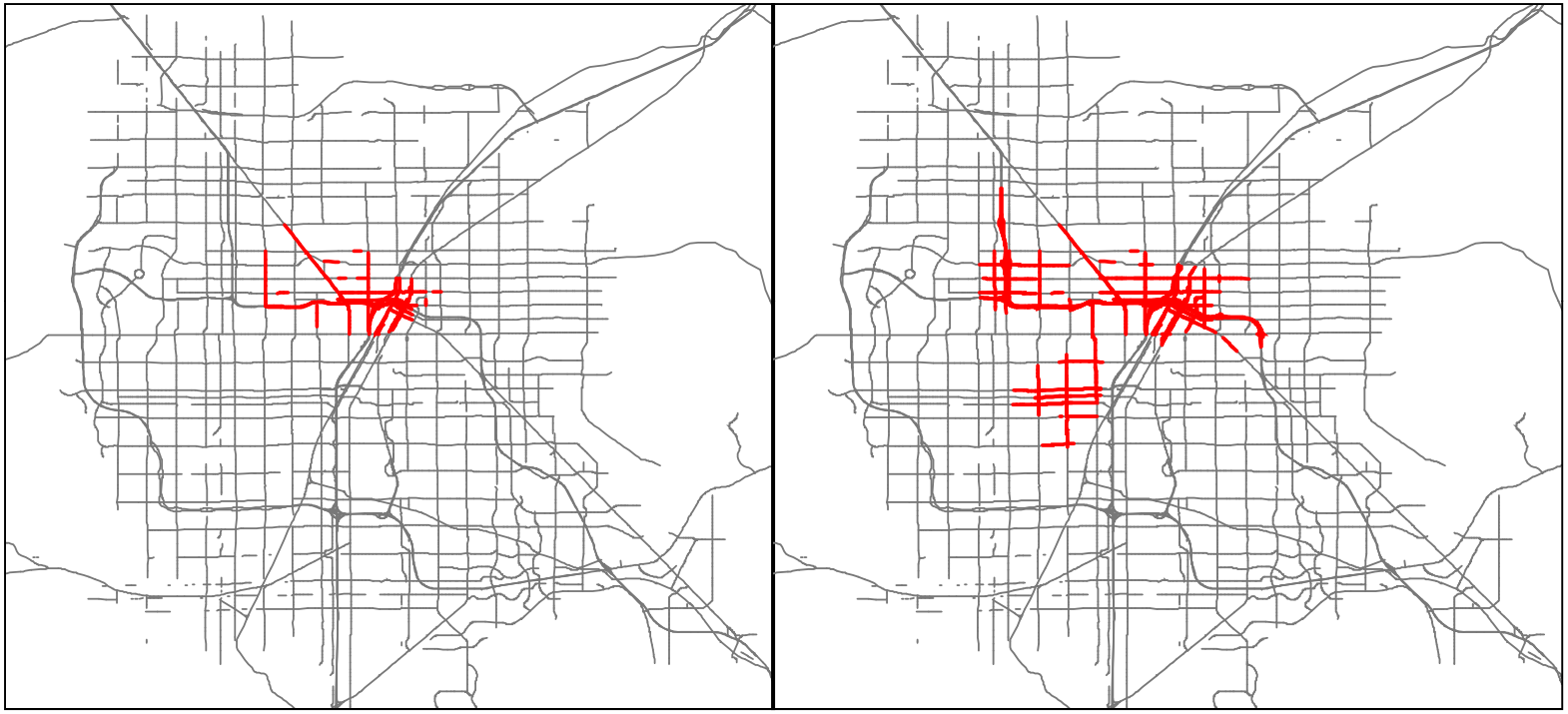
Conclusions

Overall, the model appeared to perform well for some crime types but weaker for others. One of the most troubling problems facing the evaluation of the network assignment stage of the model is the lack of any good final metric other than visual approximation for determining the

Figure 32.18:

Top 100 Observed & Predicted Residential Burglary Crime Trips

Allocated to Freeways and Major Streets

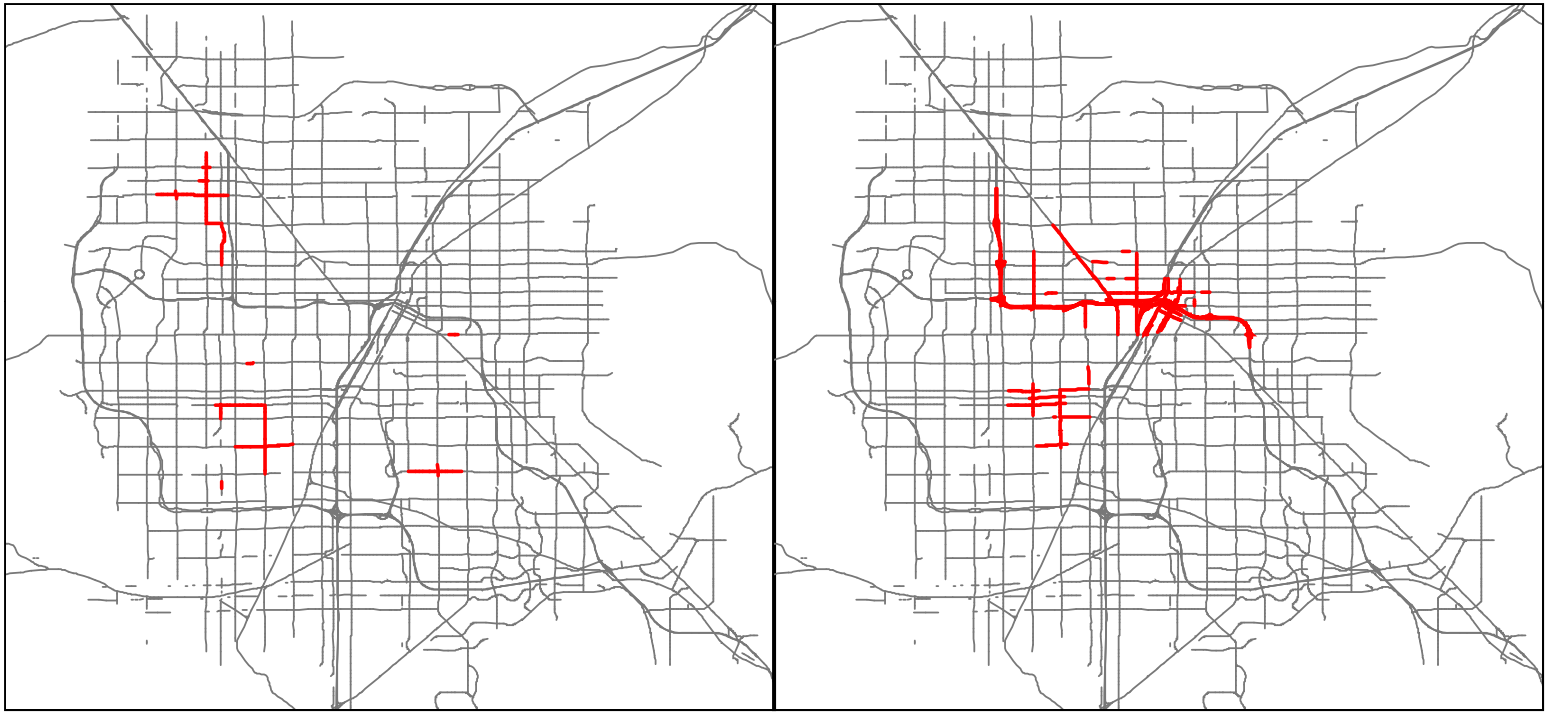


Observed **Predicted**

Figure 32.19:

Top 100 Observed & Predicted Sexual Assault Crime Trips

Allocated to Freeways and Major Streets

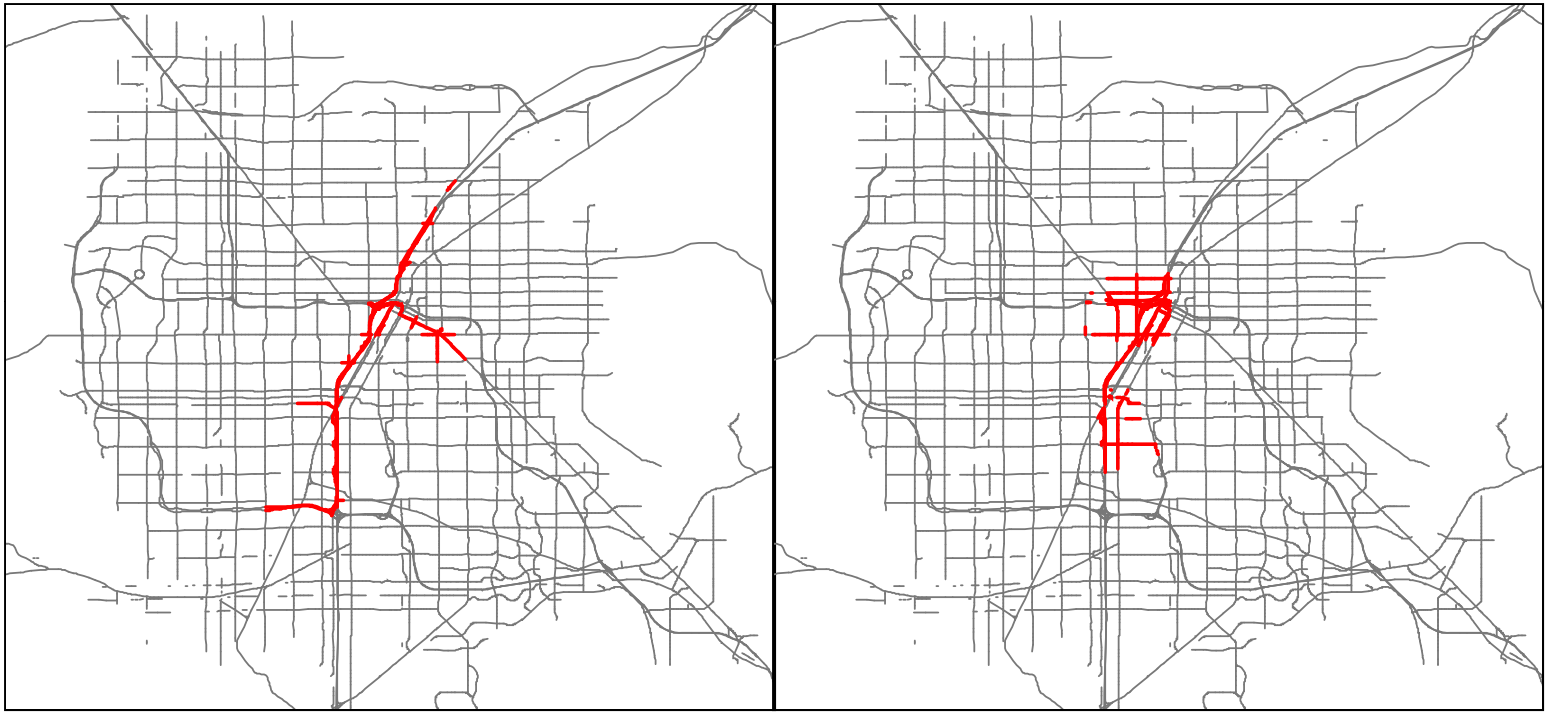


Observed **Predicted**

Figure 32.20:

Top 100 Observed & Predicted Robbery Crime Trips

Allocated to Freeways and Major Streets



Observed **Predicted**

value of the resulting prediction. Some measurement of congruence is needed to make the determination of usefulness reliable and valid.

The first stage of the model - crime trip generation, is arguably the most useful to law enforcement. This elegantly simple model can readily be adapted to different types of data, and with the inclusion of additional regression methods (specifically the negative binomial distribution model) to supplement the existing ordinary least squares (OLS) and Poisson variants, this feature is likely to remain useful for the foreseeable future.[2]

The second stage of the model - crime trip distribution, is also potentially highly useful. The analysis not merely of where offenders live or where crimes are committed but of the travel and transportation decisions linking the two locations, may have significant repercussions for crime analysts. This type of analysis will be particularly useful for strategic and administrative analysts when recommending manpower allocation, beat boundaries and precinct/district configuration schemes, and assessing the impact of major developments such as transportation corridors, shopping malls, or sports complexes on the distribution of crime.

The mode split stage of the model was difficult to apply meaningfully to the Las Vegas data in this study because of deficiencies in the data itself. Either transportation choice values were not recorded, or were recorded in irretrievable formats, making an empirical evaluation of offenders' transportation choice proclivities impractical. Failing the availability of empirical data, falling back on overall trends in public transportation choice are all that is possible for the analyst. Since it is possible that crime trips may be qualitatively different than other types of trips on which these statistical models have been based, further research is required to assess whether or not these standards will be applicable to criminal behavior.

The final stage of the model - network assignment, functioned mechanically as expected, but did result in some potentially weak results (such as overemphasis on the speed of freeways apparent in some results) which may be overcome with better mode split and network choice parameters.

One aspect of the model that caused for initial concern, the aggregation of crimes to the Traffic Analysis Zone polygon level, proved to have no significant impact on the resulting analysis. The TAZ structure seems admirably suited to analysis of this sort of movement - as indeed one might expect from its provenance.

The most successful predictive variables for estimating crime trip production, whether of origins or destinations, were infallibly total population, total employment, and income. Inclusion

---

[2] This has been implemented since this chapter was initially written.

of additional variables distorted rather than improved the predictive value of the model, most of the time with measurable multicolinearity which was not always apparent a priori.

With the mechanical aspects of the model - as implemented in the latest version of *CrimeStat*, complete and functioning correctly, it remains to be learned how to better calibrate and implement the model to make it an effective tool for law enforcement analysis and planning.

References

State of Nevada (2012). Laws of the State of Nevada. Nevada Law Library: Reno. http://www.leg.state.nv.us/law1.cfm.

References Used in *CrimeStat* Manual

(All Versions)

Abraham, B. & Ledolter, J. (2006). *Introduction to Regression Modeling*. Thompson Brooks/Cole: Belmont, CA.

Aizcorbe, A. & Starr-McCluer, M. (1996). Vehicle Ownership, Vehicle Acquisitions, and the Growth of Auto Leasing: Evidence from Consumer Surveys. Finance and Economic Discussion Series, Federal Reserve Board of Governors: Washington, DC. http://www.federalreserve.gov/pubs/feds/1996/199635/199635pap.pdf. Accessed April 28, 2012.

Aldenderfer, M. & Blashfield, R. (1984). *Cluster Analysis*. Sage: Beverly Hills, CA.

Alonso, W. (1964). *Location and Land Use: Towards a General Theory of Land Rent*. Harvard University Press: Cambride, MA.

Amir, Menachim (1971). *Patterns in Forcible Rape*. The University of Chicago Press: Chicago, 87-95.

AMPO (2012). *AMPO: Highlights & What's New*. Association of Metropolitan Planning Organizations: Washington, DC. http://www.ampo.org/. Accessed May 7, 2012.

Andersson, T. (1897). *Den Inre Omflyttningen*. Norrland: Mälmo.

Anselin, L. (2008). "Personal note on the testing of significance of the local Moran values".

Anselin, L. (2002). Under the hood: Issues in the specification ad interpretation of spatial regression models, *Agricultural Economics*, 17(3), 247-267.

Anselin, L. (1995). Local indicators of spatial association - LISA. *Geographical Analysis*. 27, No. 2 (April), 93-115.

Anselin, L.. (1992). *SpaceStat: A Program for the Statistical Analysis of Spatial Data*. Santa Barbara, CA: National Center for Geographic Information and Analysis, University of California.

References (continued)

Anselin, L. & Madden, M.s (1990). *New Directions in Regional Analysis*. Belhaven Press: New York.

Aplin, G. (1983). *Order-Neighbour Analysis*. Concepts and Techniques in Modern Geography No. 36. Institute of British Geographers, Norwich, England: Geo Books.

Bachi, R. (1957). *Statistical Analysis of Geographical Series*. Central Bureau of Statistics, Kaplan School, Hebrew University: Jerusalem.

Bailey, T. C. & Gatrell, A. C. (1995). *Interactive Spatial Data Analysis*. Longman Scientific & Technical: Burnt Mill, Essex, England.

Banez, L., Prasanna, P., Sun, L., Ali, A., Zhiqiang, Z., & Adam, B. (2003). Diagnostic potential of serum proteomic patterns in prostate cancer, *The Journal of Urology*, 170, 442–446.

Barber, C., Dobkin, D. & Huhdanpaa, H. (1997). The Quickhull algorithm for convex hulls. *ACM Trans. Mathematical Software*, 22, 469-483.

Ball, G. H. & Hall, D. J. (1970). A clustering technique for summarizing multivariate data. *Behavioral Science*, 12, 153-155.

Barnard, G. A. (1963). Comment on 'The Spectral Analysis of Point Processes' by M. S. Bartlett, *Journal of the Royal Statistical Society*, Series B, 25, 294.

Beale, E. M. L. (1969). *Cluster Analysis.* Scientific Control Systems: London.

Beimborn, E. A. (1995). A transportation modeling primer. In *Inside the Blackbox, Making Transportation Models Work for Livable Communities*. http://www4.uwm.edu/cuts/utp/models.pdf. Accessed April 28, 2012.

Ben-Akiva, M. E., & Bierlaire, M. (1999). Discrete Choice Methods and their Applications to Short Term Travel Decisions. In R. W. Hall (Ed.), *Handbook of Transportation Science* (pp. 5-34). Norwell, MA: Kluwer.

Ben-Akiva, M. & Lerman, S. (1985). *Discrete Choice Analysis: Theory and Application to Travel Demand*. MIT Press: Cambridge.

References (continued)

Berk, K. N. (1977). Tolerance and condition in regression computations, *Journal of the American Statistical Association*, 72 (360), 863-866.

Bernasco, W. (2010a). Modeling Micro-Level Crime Location Choice: Application of the Discrete Choice Framework to Crime at Places. *Journal of Quantitative Criminology, 26*(1), 113-138.

Bernasco, W. (2010b). A Sentimental Journey to Crime; Effects of Residential History on Crime Location Choice. *Criminology, 48,* 389-416.

Bernasco, W. (2007). The usefulness of measuring spatial opportunity structures for tracking down offenders: A theoretical analysis of geographic offender profiling using simulation studies. *Psychology, Crime & Law, 13,* 155-171.

Bernasco, W. (2006). Co-Offending and the Choice of Target Areas in Burglary. *Journal of Investigative Psychology and Offender Profiling, 3,* 139-155.

Bernasco, W. & Block, R. (2009). Where offenders choose to attack: A discrete choice model of robberies in Chicago. *Criminology* 47(1): 93-130.

Bernasco, W., & Kooistra, T. (2010). Effects of Residential History on Commercial Robbers' Crime Location Choices. *European Journal of Criminology, 7*(4), 251-265.

Bernasco, W. & Nieuwbeerta, P. (2005). How do residential burglars select target areas?. *British Journal of Criminology* 44: 296-315.

Bernasco, W. & Luykx, F. (2002). Using random utility models to explain location choice of offenders. Sixth Annual International Crime Mapping Research Conference, December, Denver CO.

Besag, J., 1974. Spatial interaction and the statistical analysis of lattice systems (with discussion). *J. Roy.Statist. Soc. Ser. B 36*, 192–236.

Besag, J. & Newell, J. (1991). The detection of clusters in rare diseases. Journal of the Royal Statistic Society A, 154, Part I, 143-155.

References (continued)

Besag, J., Green, P., Higdon, D., & Mengersen, K. (1995). Bayesian computation and stochastic systems (with discussion), *Statistical Science*, 10, 3-66.

Betlyon, B. & Culp, M. (2001). *Overview of Travel Demand Forecasting*. Presentation. Federal Highway Adminstration, U.S. Department of Transportation: Washington, DC. http://tmip.fhwa.dot.gov/conf_courses/presentations/fmt_traveldemand/traveldemand_files/v3_document.htm.

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press: New York.

Bishop, Y. M. M., Feinberg, S. E. & Holland, P. W. (1975). *Discrete Multivariate Analysis: Theory and Practice*. MIT Press: Cambridge, MA.

Block, C. R. (1994). STAC hot spot areas: a statistical tool for law enforcement decisions. In *Proceedings of the Workshop on Crime Analysis Through Computer Mapping*. Criminal Justice Information Authority: Chicago, IL.

Block, R. & Bernasco, W. (2009). Finding a serial burglar's home using distance decay and conditional origin-destination patterns: A test of Empirical Bayes journey to crime estimation in The Hague. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 187-211.

Block, R. & Block, C. R. (1999). Risky places: a comparison of the environs of rapid transit stations in Chicago and the Bronx. In Mollenkopf, J. (ed), *Analyzing Crime Patterns: Frontiers of Practice*, Sage Publishing: Beverly Hills, CA.

Block, R. & Block, C. R. (1995). Space, place and crime: hot spot areas and hot places of liquor-related Crime in Eck, J. E. & Weisburd, D. (eds.), *Crime and Place*. Crime Prevention Studies, Volume 4. Criminal Justice Press: Monsey, NY. 147-185.

Block, C. R. & Green, L. A. (1994). *The GeoArchive Handbook: A Guide for Developing a Geographic Database an Information Foundation for Community Policing*. Illinois Criminal Justice Information Authority: Chicago, IL.

Blumin, D. (1973). *Victims: A Study of Crime in a Boston Housing Project*. City of Boston, Mayor's Safe Street Act, Advisory Committee: Boston.

Boggs, S. L. (1965). Urban crime patterns, *American Sociological Review*, 30, 899-908.

References (continued)

Bodnar, P. M. (2007). A new approach to geographic profiling. Ninth Crime Mapping Research Conference, National Institute of Justice. Pittsburgh, PA. March.

Borland.Com (1998). *dBase IV 2.0*. Inprise Corporation: Scotts Valley, CA.

Bossard, E. G. (1993). RETAIL: Retail trade spatial interaction. In Richard E. Klosterman, Richard K. Brail & Earl G. Bossard, *Spreadsheet Models for Urban and Regional Analysis*. Center for Urban Policy Research, Rutgers University: New Brunswick, NJ, 419-448.

Boswell, M. T. & Patil, G. P. (1970). Chance mechanisms generating negative binomial distributions. In *Random Counts in Scientific Work*, Vol. 1, G. P. Patil, ed., Pennsylvania State University Press:University Park, PA, 3-22.

Bowers, K. & Hirschfield, A. (1999). Exploring links between crime and disadvantage in North-West England: An analysis using Geographic Information Systems. *International Journal of Geogrpahical Information Science*, 13, 159-184.

Bowman, A. W. & Azzalini, A. (1997). *Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations*. Oxford Science Publications, Oxford University Press: Oxford, England.

Box, E.P., Jenkins, G.M., & Reinsel, G.C. (2008). *Time series analysis: forecasting and control*, Wiley: Hoboken.

Braga, A. & Weisburd, D. (2010). Policing Problem Places: Crime Hot Spots and Effective Prevention. Oxford: Oxford University Press.

Brantingham, P. & Brantingham, P. (1999). A Theoretical Model of Crime Hot Spot Generation, *Studies on Crime and Crime Prevention*, 8 (1), 7-26.

Brantingham, P. & Brantingham, P. J. (1984). *Patterns in Crime*. Macmillan Publishing: New York.

Brantingham, P. L. & Brantingham, P. J. (1981). Notes on the geometry of crime. In Brantingham, P. J. & Brantingham, P. L. *Environmental Criminology*. Waveland Press, Inc.: Prospect Heights, IL, 27-54.

References (continued)

Bright, M. L. & Thomas, D. S. (1941). Interstate migration and intervening opportunities, *American Sociological Review*, 6, 773-783.

Brown, R. G. (1959). *Statistical forecasting for inventory control*, New York: McGraw-Hill.

BTS (2007). Table 2: A matrix of transportation expenditure by source of finance and type of expenditure, *Government Transportation Financial Statistics*, Bureau of Transportation Statistics, U.S. Department of Transportation: Washington, DC., http://www.bts.gov/publications/government_transportation_financial_statistics/2007/html/table_02.html. Accessed April 28, 2012.

BTS (2003). U.S. Vehicle-miles (millions), Table 1-32. *National Transportation Statistics 2004*, Bureau of Transportation Statistics, U.S. Department of Transportation: Washington, DC. http://www.bts.gov/publications/national_transportation_statistics/2004/html/table_01_32.html. Accessed April 28, 2012.

BTS (2002). *National Household Travel Survey: Daily Travel Quick Facts*. Bureau of Transportation Statistics, U.S. Department of Transportation: Washington, DC. http://nhts.ornl.gov/download.shtml#2009. Accessed June 1, 2012.

BUGS (2008). *The BUGS (Bayesian Inference Using Gibbs Sampling) Project*. MRC Biostatistics Unit, University of Cambridge: Cambridge. http://www.mrc-bsu.cam.ac.uk/bugs. Accessed March 23, 2010.

Burgess, E. W. (1925). The growth of the city: an introduction to a research project. In Park, R. E., Burgess, E. W. & Mackensie, R. D. (ed), *The City*. University of Chicago Press: Chicago, 47-62.

Bursik, R. J., Jr. & Grasmick, H. G. (1993). Economic deprivation and neighborhood crime rates, 1960-1980. *Law and Society Review*, 27, 263-268.

Burt, J. E. & Barber, G. M. (1996). *Elementary Statistics for Geographers* (second edition). The Guilford Press: New York.

Cameron, A. C. & Trivedi, P.K. (1998). *Regression Analysis of Count Data*. Cambridge University Press: Cambridge, U.K.

References (continued)

Cameron, A. C. & Windmeijer, F. A. G. (1996). R-squared measures for count data regression models with applications to health-care utilization. *Journal of Business & Economic Statistics*, 14(2), 209-20.

Can, A. & Megbolugbe, I. (1996). The geography of underserved mortgage markets. Paper presented at the American Real Estate and Urban Economics Association meeting. May.

Canter, D. (2009). Developments in geographical offender profiling: Commentary on Bayesian journey-to-crime modeling. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 161-166.

Canter, D. (2003). *Dragnet: A Geographical Prioritisation Package*. Center for Investigative Psychology, Department of Psychology, The University of Liverpool: Liverpool, UK. http://www.i-psy.com/publications/publications_dragnet.php.

Canter, D. (1994). *Criminal Shadows: Inside the Mind of the Serial Killer*. Harper Collins Publishers: London.

Canter, D. (1999). Modelling the home location of serial offenders. Paper presented at the 3rd Annual International Crime Mapping Research Conference, Orlando, December.

Canter, D. V. & Hammond, L. (2007). A comparison of the efficacy of different decay functions in geographical profiling for a sample of US serial killers. In press *Journal of Investigative Psychology and Offender Profiling*. 3.

Canter, D.V, Coffey, T., Huntley, M., & Missen, C. (2000). Predicting serial killers' home base using a decision support system. Journal of Quantitative Criminology, 16, 457-478.

Canter, D. & Larkin, P. (1993). The environmental range of serial rapists, *Journal of Environmental Psychology*, 13, 63-69.

Canter, D. & Tagg, S. (1975). Distance estimation in cities, *Environment and Behaviour*, 7, 59-80.

Canter, D. & Gregory, A. (1994). Identifying the residential location of rapists, *Journal of the Forensic Science Society*, 34 (3), 169-175.

References (continued)

Canter, D. V., & Snook, B. (1999). *Modelling the home location of serial offenders*. Paper presented at the meeting of the Crime Mapping Research Center, Orlando, FL. December.

Capone, D. L. & Nichols Jr, W. W. (1975). Crime and distance: an analysis of offender behaviour in space, *Proceedings, Association of American Geographers*, 7, 45-49.

Carlin, B. P. & Louis, T. A. (2000). *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman & Hall/CRC: Boca Raton Fl.

Carmichael, J. W., George, L.A. & Julius, R.S. (1968). Finding natural clusters. *Systematic Zoology*, 17, 144-150.

Carnegie-Mellon University (1975). *Security of Patrons on Urban Public Transportation Systems*. Transportation Research Institute, Carnegie-Mellon University: Pittsburgh, PA.

Cattell, R. B. & Coulter, M.A. (1966). Principles of behavioural taxonomy and the mathematical basis of the taxonome computer program. *British Journal of Mathematical and Statistical Psychology*, 19, 237-269.

Chainey, S. & Ratcliffe, J. (2005). *GIS and Crime Mapping*, John Wiley & Sons, Inc.:Chichester, Sussex, England.

Chainey, S., Thompson, L. & Uhlig, S. (2008). The utility of hotspot mapping for predicting spatial patterns of crime. *Security Journal*, **21**, 4-28.

Chaitin, G. (1990). *Information, Randomness and Incompleteness* (second edition). World Scientific: Singapore.

Chand, D & Kapur, S. (1970). An algorithm for convex polytopes. *J. ACM*, 17, 78-86.

Chen, A. & Renshaw, E. (1994) The general correlated random walk. *Journal of Applied Probability*, 31, 869-884.

Chen, A. & Renshaw, E. (1992). The Gillis-Domb-Fisher correlated random walk. *Journal of Applied Probability*, 29, 792-813.

Chiricos, T. (1987). Rates of Crime and Unemployment *Social Problems*, 34, 187-211

References (continued)

Chrisman, N. (1997) *Exploring Geographic Information Systems*. John Wiley & Sons, Inc.: New York.

Citro, C. F. & Michael, R. T. (eds) (1995). *Measuring Poverty : A New Approach.* Panel on Poverty and Family Assistance, Committee on National Statistics, Commission on Behavioral and Social Sciences and Education, National Research Council: Washington, DC.http://www.census.gov/hhes/www/img/povmeas/ack.pdf. Accessed May 7, 2012.

Clare, J., Fernandez, J., & Morgan, F. (2009). Formal Evaluation of the Impact of Barriers and Connectors on Residential Burglars' Macro-Level Offending Location Choices. *Australian and New Zealand Journal of Criminology*, *42*, 139-158.

Clark, P. J. & Evans, F. C. (1954). Distance to nearest neighbor as a measure of spatial relationships in populations. *Ecology*, 35, 445-453.

Clayton, D. & Kaldor, J. (1987). Empirical Bayes estimates of age-standardized relative risks for use in disease mapping. *Biometrics*, 43, 671-681.

Cleveland, W. S., Grosse, E. & Shyu, W. M. (1993). Local regression models. In John M. Chambers & Trevor J. Hastie, *Statistical Models in S*. Chapman & Hall: London.

Cliff, A. D. & Haggett, P. (1988). *Atlas of Disease Distributions*. Blackwell Reference: Oxford.

Cliff, A. & Ord, J. (1973). *Spatial Autocorrelation*. Pion: London.

Cohen, J., Garman, S. & Gorr, W. L. (2009). Empirical calibration of time series monitoring methods using receiver operating characteristic curves, *International Journal of Forecasting*, 2009, 25(3), 484–497.

Cohen, L.E. & Felson, M. (1979) Social change and crime rate trends: a routine activity approach, *American Sociological Review*, 44: 588-608.

Cohen, L. E. 1981 Modeling crime trends: a criminal opportunity perspective, *Journal of Research in Crime and Delinquency*, 18:138-163.

Cole, A. J. & Wishart, D. (1970). An improved algorithm for the Jardine-Sibson method of generating overlapping clusters. *Comparative Journal*, 13, 156-163.

References (continued)

Committee on Map Projections (1986). *Which Map is Best*, American Congress on Surveying and Mapping, Falls Church, VA., 1986.

Conway, R. W & Maxwell, W. L. (1962), A queuing model with state dependent service rates. *Journal of Industrial Engineering* **12**: 132–136.

Cormen, T. H., Leiserson, C. E., Rivest, R. L. & Stein, C. (2009). Ch. 16: Greedy algorithms, *Introduction to Algorithms*, MIT Press: Cambridge, MA.

Cornish, D. & Clarke, R. (1986). *The Reasoning Criminal*. Springer-Verlag: New York.

Cressie, N. (1991). *Statistics for Spatial Data.* New York: J. Wiley & Sons, Inc.

Cromley, R. G. (1992). *Digital Cartography*. Prentice Hall: Englewood Cliffs, NJ.

Culp, M. & Lee, E. J. (2005). Improving travel models through peer review. *Public Roads*, 68 (6), FHWA-HRT-05-005. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/publications/publicroads/05may/07.cfm. Accessed April 28, 2012.

Curtis, L. A. (1974). *Criminal Violence*. Lexington Books: Lexington, MA.

D'andrade, R. (1978). U-Statistic Hierarchical Clustering *Psychometrika*, 4,58-67.

de Berg, M., van Kreveld, M., Overmans, M. & Schwarzkopf, O. (2000). Convex hulls: mixing things. In *Computational Geometry: Algorithms and Applications*, 2nd rev. ed. Springer-Verlag: Berlin, 235-250.

Demographia (1999). *U.S. Central Cities and Suburban Crime Rates Ranked: 1999*. Wendell Cox Consultancy: Belleville, IL. http://www.demographia.com/db-crime99r.htm.

Demographia (1998). *U. S. Metropolitan Areas: 1998 Central City and Suburban Population*. Wendell Cox Consultancy: Belleville, IL. http://www.demographia.com/db-usmsacc98.htm.

Denison, D.G.T., Holmes, C.C., Mallick, B. K., & Smith, A. F. M. (2002). *Bayesian Methods for Nonlinear Classification and Regression*. John Wiley & Sons, Ltd: Chichester, Sussex.

References (continued)

De Smith, M., Goodchild, M. F., & Longley, P. A. (2007). *Geospatial Analysis* (second edition). Matador: Leicester, U.K.

Diggle, P. J. (2003). *Statistical Analysis of Spatial Point Patterns.* Arnold: London.

Dijkstra, E. W. (1959). A note on two problems in connection with graphs, *Numerische Mathematik*, 1, 269-271.

Domencich, T. & McFadden, D. (1975). *Urban Travel Demand: A Behavioral Analysis*. North Holland Publishing Company: Amsterdam & Oxford (republished in 1996). Also found at http://emlab.berkeley.edu/users/mcfadden/travel.html. Accessed April 28, 2012.

Draper, N. & Smith, H. (1981). *Applied Regression Analysis, Second Edition*. John Wiley & Sons: New York.

Durkheim, E. (1895). *The Rules of Sociological Method*. Free Press: New York. 1964.

Dwass, M (1957). Modified randomization tests for nonparametric hypotheses. *Annals of Mathematical Statistics*, 28, 181-187.

Ebdon, D. (1988). *Statistics in Geography* (second edition with corrections). Blackwell: Oxford.

Ehrlich, I. (1975). On the relation between education and crime. In Juster, F. T. (ed), *Education, Youth and Human Behavior*. McGraw-Hill: New York, 313-337.

El-Basyouny K. & Sayed, T. (2009). Collision prediction models using multivariate Poisson-lognormal regression. *Accident Analysis & Prevention*, 41(4), 820-828.

Eldridge, J. D. & Jones, J. P. (1991). Warped space: a geography of distance decay, *Professional Geographer*, 43 (4), 500-511.

Elmore, J. G., Miglioretti, D. M., Reisch, L. M., Barton, M. B., Kreuter, W., & Christiansen, C. L., (2002). Screening mammograms by community radiologists: Variability in false positive rates. Journal of the National Cancer Institute, 94, 1373–1380.

Engelen, R. E. (1986). Transportation planning. In So, F. S. *The Practice of State and Regional Planning*. American Planning Association: Chicago, Ch. 17, 431-453.

References (continued)

ESRI (2012). *ArcGIS 10.0*. Environmental Systems Research Institute: Redlands, CA. http://www.esri.com/software/arcgis/index.html.

ESRI (1998a). *ArcView GIS 3.1*. Environmental Systems Research Institute: Redlands, CA.

ESRI (1998b). *ArcInfo 7.2.1*. Environmental Systems Research Institute: Redlands, CA.

ESRI (1998c). *Atlas*GIS 4.0*. Environmental Systems Research Institute: Redlands, CA.

ESRI (1997). *ArcView Spatial Analyst.* Environmental Systems Research Institute: Redlands, CA.

Everitt, B. S. (2011). *Cluster Analysis* (5th edition). J. Wiley: London.

Everitt, B. S., Landau, S. & Leese, M. (2001). *Cluster Analysis*. 4th Edition. Oxford University Press: New York.

Farewell, D. (1999). Specifying the bandwidth function for the kernel density estimator. http://www.iph.cam.ac.uk/bugs/documentation/coda03/node44.html.

Fazel, S. 2006. The population impact of severe mental illness on violent crime. *American Journal of Psychiatry*, 163, 1397-1403.

Felson, M. (2002). *Crime & Everyday Life* (3rd Ed). Sage: Thousand Oaks, CA.

FHWA (2006). *Highway Statistics: 2005*. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/policy/ohim/hs05/national_household_info.htm.

FHWA (2009). Integrated Urban Systems Modeling, *The Exploratory Advanced ResearchProgram Fact Sheet*, FHWA-HRT-09-042. Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://www.fhwa.dot.gov/advancedresearch/pubs/interurbsys.pdf. Accessed April 28, 2012.

References (continued)

FHWA (1997). *Model Validation and Reasonableness Checking Manual*. Prepared by Barton-Aschman Associates, Inc and Cambridge Systematics, Inc for the Travel Model Improvement Program, Federal Highway Administration, U.S. Department of Transportation: Washington, DC. http://ops.fhwa.dot.gov/freight/publications/qrfm2/sect08.htm. Accessed May 31, 2012.

FHWA (1996). Latest VMT growth estimates, *Highway Information Update*, 1(1), Federal Highway Administration, U.S. Department of Transportation: Washington, DC., http://www.fhwa.dot.gov//ohim/vol1no1.html. Accessed April 28, 2012.

Field, B. & MacGregor, B. (1987). *Forecasting Techniques for Urban and Regional Planning*. UCL Press, Ltd: London.

Findley, D. F. (1993). *The Overfitting Principles Supporting AIC*. Statistical Research Division Report Series, SRD Research Report no. CENSUS/SRD/ RR-93/04, U.S. Bureau of the Census: Washington, DC. http://www.census.gov/srd/papers/pdf/rr93-04.pdf.

Fisher, W. (1958). On grouping for maximum homogeneity. *Journal of the American Statistical Association*. **53**, 789-798.

Foot, D. (1981). *Operational Urban Models*. Methuen: London.

Fotheringham, A. S., Brunsdon, C. & Charlton, M. (2002). *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. John Wiley & Sons: New York.

Fotheringham, A. S. & O'Kelly, M. E. (1989). *Spatial Interaction Models: Formulations and Applications*. Kluwer Academic Publishers: Boston.

Fowles, R. & Merva, M.. (1996). Wage Inequality and Criminal Activity, *Criminology*, 34, 163-82.

Frank, L., Kerr, J. Chapman, J. & Sallis, J. (2007). Urban Form Relationships With Walk Trip Frequency and Distance Among Youth. *American Journal of Health Promotion*. March/April 2007, V21, I4 Supplement, 305.

Freedman, D. A. (1999). Ecological inference and ecological fallacy. *International Encyclopedia of the Social and Behavioral Sciences*, Technical Report No. 549, October. http://www.stanford.edu/class/ed260/freedman549.pdf. Accessed March 26, 2012.

References (continued)

Friedman, H. P. & Rubin, J. (1967). On some invariant criteria for grouping data, *Journal of the American Statistical Association*, **62**, 1159-1178.

Fritzon, K. (2001). An Examination of the Relationship between Distance Travelled and Motivational Aspects of Firesetting Behaviour. *Journal of Environmental Psychology*, 21, 45-60.

Furfey, P. H. (1927). A note on Lefever's 'Standard deviational ellipse'. *American Journal of Sociology*. XXIII, 94-98.

Gaile, G. L. & Burt, J. E. (1980). *Directional Statistics*. Concepts and Techniques in Modern Geography No. 25. Institute of British Geographers, Norwich, England: Geo Books.

Geary, R. (1954). The contiguity ratio and statistical mapping. *The Incorporated Statistician*, 5, 115-145.

Gelman, A. (1996). Inference and monitoring convergence. In Gilks, W. R., Richardson, S. & Spiegelhalter, D. J. (eds), *Markov Chain Monte Carlo in Pratice*, Chapman & Hall: London.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2004). *Bayesian Data Analysis* (second edition). Chapman & Hall/CRC: Boca Raton, FL.

Gelman, A. & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences (with discussion), *Statistical Science*, 7, 457-511.

Gersho, A. & Gray, R. (1992). *Vector Quantization and Signal Compression*. Kluwer Academic Publishers: Dordrecht, Netherlands.

Getis, A. (1991). Spatial interaction and spatial auto-correlation: a cross-product approach. *Environment and Planning A*, 23, 1269-1277.

Getis, A. & Ord, J. K. (1996). Local spatial statistics: an overview. In Longley, P. & Batty, M. (eds), *Spatial Analysis: Modelling in a GIS Environment*. GeoInformation International: Cambridge, England, 261-277.

Getis, A. & Ord, J. K. (1992). The analysis of spatial association by use of distance statistics, *Geographical Analysis*, 24, 189-206.

References (continued)

Getis, A. & Boots, B. (1978). *Models of Spatial Processes: An Approach to the Study of Point, Line and Area Patterns*. London: Cambridge University Press.

Gitman, I. & Levine, M. D. (1970). An algorithm for detecting uniomodal fuzzy sets and its application as a clustering technique. *IEE Transactions on Computers*, 19, 583-593.

Golden Software. 2008. *Surfer® for Windows (Ver. 10)*. Golden Software, Inc.: Golden, CO.

Goldfield, S. M., Quandt, R. E., & Trotter, H. F. (1966). Maximization by quadratic hill-climbing, *Econometrica*, 34 (3), 541-551.

Gorr, W. L. (2009). Forecast accuracy measures for exception reporting using receiver operating characteristic curves, *International Journal of Forecasting*, 2009, Vol. 25(1), 48–61.

Gorr, W. L. & Kurland, K. S. (2012). *GIS Tutorial for Crime Analysis*, Esri Press, Redlands.

Gorr, W.L. & Lee, Y. J. (2013). Early warning system for crime hot spots, Heinz College, Carnegie Mellon University Working Paper Series (http://www.heinz.cmu.edu/faculty-and-research/research/research-details/index.aspx?rid=482).

Gorr, W. L., Olligschlaeger, O. M. & Thompson, Y. (2003). Short-term forecasting of crime, *International Journal of Forecasting, Special Section on Crime Forecasting*, 19(4), 579–594.

Gowers, J. C. (1967). A comparison of some methods of cluster analysis. *Biometrics*, 23, 623-628.

Graham, R (1972). An efficient algorithm for determining the convex hull of a finite planar point set. *Info. Proc. Letters*, 1, 132-133.

Greenfeld, L. A. (1998). *Alcohol and Crime: An Analysis of National Data on the Prevalence of Alcohol Involvement in Crime*. NCJ 168632, Bureau of Justice Statistics, U.S. Department of Justice: Washington, DC. http://www.ojp.usdoj.gov/bjs/pub/pdf/ac.pdf.

Greenhood, D. (1964). *Mapping*. The University of Chicago Press: Chicago.

References (continued)

Greenwood, M. & Yule, G. U. (1920). An inquiry into the nature of frequency distributions of multiple happenings, with particular reference to the occurrence of multiple attacks of disease or repeated accidents. *Journal of the Royal Statistical Society*, 83, 255-279.

Griffith, D. A. (1987). *Spatial Autocorrelation: A Primer*. Resource Publications in Geography, The Association of American Geographers: Washington, DC.

Groff, E. R. (2002). Modeling the spatial dynamics of homicide. Paper presented at Mapping and Analysis for Public Safety annual conference. Denver, CO., December. http://www.ojp.usdoj.gov/nij/maps/Conferences/02conf/Groff.ppt.

Groff, E. R. & McEwen, J. T (2005). Disaggregating the Journey to Homicide. In Wang, F. (ed.), *Geographic Information Systems and Crime Analysis*. Idea Group Publishing: Hershey, PA.

Grubesic, T. H. & Murray, A. T. (2001). Detecting hot spots using cluster analysis and GIS. Paper presented at Annual Conference of the Crime Mapping Research Center, Dallas, TX. http://www.ojp.usdoj.gov/cmrc.

Guikema, S.D. & Coffelt, J. P. (2008). A flexible count data regression model for risk analysis, *Risk Analysis*, 28 (1), 213–223.

Guo, F., Wang, X. & Abdel-Aty, M. A. (2009). Modeling signalized intersection safety with corridor-level spatial correlations, *Accident Analysis and Prevention*, In press.

Hagan, J. & Peterson, R. (1994). *Inequality and Crime*. Stanford University Press: Palo Alto, CA.

Hägerstrand, T. (1957). Migration and area: survey of a sample of Swedish migration fields and hypothetical considerations on their genesis. *Lund Studies in Geography, Series B, Human Geography*, 4, 3-19.

Haggett, P. & Arnold, E. (1965). *Locational Analysis in Human Geography* (1st edition). Edward Arnold: London.

Haggett, P., Cliff, A. D. & Frey, A. (1977). *Locational Analysis in Human Geography* (2nd edition). Edward Arnold: London.

References (continued)

Hall, D. B. (2000). Zero-inflated Poisson and binomial regression with random effects: a case study. *Biometrics*, 56, 1030-1039.

Hammond, R. & McCullagh, P. (1978). *Quantitative Techniques in Geography: An Introduction*. Second Edition. Clarendon Press: Oxford, England.

Hansen, K. (1991). Head-banging: robust smoothing in the plane. *IEEE Transactions on Geoscience and Remote Sensing*, 29 (3), 369-378.

Härdle, W. (1991). *Smoothing Techniques with Implementation in S.* Springer-Verlag: New York.

Harlow, C. W. 1999. Prior abuse reported by inmates and probationers, *Bureau of Justice Statistics Selected Findings*. NCJ 172879, Bureau of Justice Statistics, U.S. Department of Justice: Washington, DC. http://www.ojp.usdoj.gov/bjs/pub/pdf/parip.pdf.

Harries, K. (1999). *Mapping Crime: Principle and Practice*. NCJ 178919, National Institute of Justice, U. S. Department of Justice: Washington, DC., http://www.ncjrs.org/html/nij/mapping/pdf.html.

Harries, K. (1980). *Crime and the Environment*. Charles C. Thomas Press: Springfield.

Harries, K. & Canter, P. (1998). The use of GPS in geocoding crime incidents. Personal Communication.

Hartigan, J. A. (1975). *Clustering Algorithms*. John Wiley & Sons, Inc.: New York.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov Chains and their applications, *Biometrika*, 57, 97-109.

Hauer, E. (2002). *Observational Before-After Studies in Road Safety*. Pergamon: Oxford.

Henderson, R. (1981). *The Structural Root Systems of Sitka Spruce and Related Stochastic Processes*. PhD Thesis, University of Edinburgh: Edinburgh.

Henderson, R, Renshaw, E. & Ford, D. (1984). A correlated random walk model for two-dimensional diffusion. Journal of Applied Probability, 21, 233-246.

References (continued)

Henderson, R., Renshaw, E & Ford, D. (1983). A note on the recurrence of a correlated random walk. Journal of Applied Probability, 20, 696-699.

Henderson, R., Ford, D., Renshaw, E. & Deans, J. D. (1983). Morphology of the structural root system of Sitka Spruce 1. Analysis and Quantitative Description. *Forestry*, 56 (2), 121-135.

Hensher, D. A. & Button, K. J. (2002). *Handbook of Transport Modeling*. Elsevier Science: Cambridge, UK.

Heskin, A., Levine, N. & Garrett, M. (2000). "Rent control and vacancy control: a spatial analysis of four California cities". *Journal of the American Planning Association*. 66 (2), 162-176.

H-GAC (2010). Transportation and air quality program, *Houston-Galveston Area Council*. http://www.h-gac.com/taq/.

Hibon M. & Makridakis S. (2000). The M3 Competition: results, conclusions and implications, *International Journal of Forecasting*, 16, 451-476.

Hilbe, J. M. (2008). *Negative Binomial Regression (with corrections)*. Cambridge University Press: Cambridge.

Hipp, J. R. (2007). Block, Tract, and Levels of Aggregation: Neighborhood Structure and Crime and Disorder as a Case in Point. *American Sociological Review* 72:659-680.

Hodge, S. & Canter, D. (1998) Victims and Perpetrators of Male Sexual Assault. *Journal of Interpersonal Violence*, 1 (April), 222-239.

Horowitz, J. L., Koppelman, F. S. & Lerman, S. R. (1986). *A Self-instructing Course in Disaggregate Mode Choice Modeling*. Federal Transit Administration, U.S. Department of Transportation: Washington, DC. http://ntl.bts.gov/DOCS/381SIC.html. Accessed April 28, 2012.

Huff, D. L. (1963). A probabilistic analysis of shopping center trade areas. *Land Economics*, 39, 81-90.

References (continued)

Hultquist, J., Brown, L. & Holmes, J. (1971). Centro: a program for centrographic measures. Discussion paper no. 21, Department of Geography, Ohio State University: Columbus, OH.

Husmeier, D. & McGuire, G. (2002). Detecting recombination in DNA sequence alignments: A comparison between maximum likelihood and Markov Chain Monte Carlo. Biomathematics and Statistics Scotland, SCRI: Dundee. http://www.bioss.ac.uk/~dirk/software/BARCEtdh/Manual/em/em.html.

Huxhold, W. E. (1991). *An Introduction to Geographic Information Systems*. Oxford University Press: Oxford, New York, 147-184.

Hyndman, R. J. & Athanasopoulos, G. (2012). *Forecasting: principles and practice: An online textbook*, http://otexts.com/fpp/.

IIHS (2012). *Q&A: Red Light Cameras*. Insurance Institute for Highway Safety: Arlington, VA. http://www.iihs.org/research/qanda/rlr.html. Accessed June 5, 2012.

Insightful Corporation (2001). *S-PLUS 6.0 Professional for Windows*. Insightful Corporation: Seattle, WA.

Isard, W. (1979). *Location and Space-Economy: A General Theory Relating to Industrial Location, Market Areas, Land Use, Trade, and Urban Structure* (originally published 1956). Program in Urban and Regional Studies, Cornell University: Ithaca, NY.

Isard, W. (1960). *Methods in Regional Analysis*. John Wiley & Sons: New York.

Isbel, E. C. (1944). Internal migration in Sweden and intervening opportunities, *American Sociological Review*, 9, 627-639.

\ITE (2003). *Trip Generation* (7th edition). Institute of Transportation Engineers: Washington, DC.

ITE (2010). *Highway Capacity Manual* (5th edition) Institute of Transportation Engineers: Washington, DC. http://www.ite.org/emodules/scriptcontent/orders/ProductDetail.cfm?pc=LP-674. Accessed June 5, 2012.

References (continued)

Jardine, N. & Sibson, R. (1968). The construction of hierarchic and non-hierarchic classifications. *Comparative Journal*, 11, 117-184.

Jefferis, E. (1998). A multi-method exploration of crime hot spots. Crime Mapping Research Center, National Institute of Justice: Washington, DC.

Jepsen, L., & Jepsen, C. (2002). An empirical analysis of the matching patterns of same-sex and opposite-sex couples. *Demography, 39*(3), 435-453.

Jessen, R. J. (1979). *Statistical Survey Techniques*. John Wiley & Sons: New York.

Johnson, M.A. (1978). Attribute importance in multiattribute transportation decisions, *Transportation Research Record*, 673, 15-21.

Johnson, S.C. (1967), Hierarchical Clustering Schemes *Psychometrika*, 2,241-254

Jones, K. S. & Jackson, D. M. (1967). Current approaches to classification and clump finding at the Cambridge Language Research Unit. *Comparative Journal*, 10, 29-37.

Kafadar, K. (1996). Smoothing geographical data, particularly rates of disease. *Statistics in Medicine* 15(23), 2539-2560.

Kallay, M. (1984). The complexity of incremental convex hull algorithms in Rd, *Info. Proc. Letters* 19, 197.

Kaluzny, S. P., Vega, S. C., Cardoso, T. P., & Shelly, A. A. (1998). *S+ Spatial Stats: User Manual for Windows and Unix*. Springer: New York.

Kanji, G. K. (1993). *100 Statistical Tests*. Sage Publications: Thousand Oaks, CA.

Kelsall, J. E. & Diggle, P.J. (1995a). Kernel estimation of relative risk, *Bernoulli*, 1, 3-16.

Kelsall, J. E. & Diggle, P.J. (1995b). Non-parametric estimation of spatial variation in relative risk. *Statistical Medicine*, 14, 2335-2342.

References (continued)

Kent, J., Leitner, M., & Curtis, A. (2006). Evaluating the usefulness of functional distance measures when calibrating journey-to-crime distance decay algorithms. *Computers, Environment and Urban Systems, 30* (2), 181-200.

Khan, G., Qin, X., & Noyce, D. A. (2006). Spatial analysis of weather crash patterns in Wisconsin. 85th Annual meeting of the Transportation Research Board: Washington, DC.

Kim, K. E. & Parke, M. (1996). The use of GPS and GIS in traffic safety. Report to Motor Vehicle Safety Office, State of Hawaii Department of Transportation: Honolulu.

Kind, S. S. (1987). Navigational ideas and the Yorkshire Ripper investigation. *Journal of Navigation*, 40 (3), 385-393.

King, B. F. (1967). Step wise clustering procedures. *Journal of the American Statistical Association*. 62, 86-101.

Kitamura, R., Yoshii, T., & Yamamoto, T. (2009). The Expanding Sphere of Travel Behaviour Research: Selected Papers from the 11th International Conference on Travel Behaviour Research. Emerald Group Publishing, Ltd: Bingley, U.K. http://books.google.com/books?id=fFqEnNOWKw8C&pg=PA375&lpg=PA375&dq=microsimulation+of+travel+behavior&source=bl&ots=ArxmN7EIZl&sig=rIUukRBjCApH22qDQ0UXp5dUOGs&hl=en&sa=X&ei=jRmkT_3aFIOi8ATImsS5CQ&ved=0CGQQ6AEwCA#v=onepage&q=microsimulation%20of%20travel%20behavior&f=false. Accessed May 4, 2012.

Kneebone, E. & Raphael, S. (2011). *City and Suburban Crime Trends in Metropolitan America*. Metropolitan Opportunity Series, Metropolitan Policy Program, Brookings Institution: Washington, DC. http://www.brookings.edu/papers/2011/0526_metropolitan_crime_kneebone_raphael.aspx. Accessed April 28, 2012.

Kohfeld, C. W. & Sprague, J. (1988). Urban unemployment drives crime. *Urban Affairs Quarterly*, 24, 215-241.

Knox, E. G. (1988). Detection of clusters. In Elliott, P. (ed), *Methodology of Enquiries into Disease Clustering*, London School of Hygiene and Tropical Medicine: London.
Knox, E. G. (1964). The detection of space-time interactions. *Applied Statistics*, 13, 25-29.

References (continued)

Knox, E. G. (1963). Detection of low intensity epidemicity: Application in cleft lip and palate. *British Journal of Preventive and Social Medicine*, 18, 17-24.

Krebs, J. R., & Davies, N. B. (1993). *An Introduction to Behavioural Ecology* (3th ed.). Oxford: Blackwell.

Krueckeberg, D. A. & Silvers, A. L. (1974). *Urban Planning Analysis: Methods and Models*. John Wiley & Sons: New York.

Kuhn, H.W. & Kuenne, R. E. (1962). An efficient algorithm for the numerical solution of the generalized Weber problem in spatial economics, *Journal of Regional Science* 4, 21-33.

Kulldorff, M. (1997). A spatial scan statistic, *Communications in Statistics - Theory and Methods*, 26, 1481-1496.

Kulldorff, M. & Williams, G. (1997). *SaTScan v 1.0: Software for the Space and Space-Time Scan Statistics*, Bethesda, MD: National Cancer Institute.

Kulldorff, M. & Nagarwalla, N. (1995). Spatial disease clusters: Detection and inference, *Statistics in Medicine*, 14, 799-810.

Lam, N. S. & De Cola, L. (1993). *Fractals in Geography*. The Blackburn Press: Caldwell, NJ.

Lander, B. (1954). *Toward an Understanding of Juvenile Delinquency*. Columbia University Press: New York.

Langbein, L. I. & Lichtman, A. J. (1978). *Ecological Inference*. Sage University Paper series on Quantitative Applications in the Social Sciences, series no. 07-010. Beverly Hills and London: Sage Publications.

Langworthy, R. H. & Jefferis, E. (1998). The utility of standard deviational ellipses for project evaluation. Discussion paper, National Institute of Justice: Washington, DC.

Laukkanen, M., P. Santtila, P. Jern, & K. Sandnabba, K. 2008. Predicting offender home location in urban burglary series. *Forensic Science International*, 176, 224-235.

References (continued)

LaVigne, N. & Wartell, J. (2000). *Crime Mapping Case Studies: Success in the Field (volume 2).* Police Executive Research Forum and National Institute of Justice, U. S. Department of Justice: Washington, DC., http://www.mn-8.com/Merchant2/merchant.mvc?Screen=PROD&Product_Code=841&Category_Code=CAR.

LaVigne, N. & Wartell, J. (1998). *Crime Mapping Case Studies: Success in the Field (volume 1).* Police Executive Research Forum and National Institute of Justice, U. S. Department of Justice: Washington, DC.

LeBeau, J. L. (1997). *Demonstrating the Analytical Utility of GIS for Police Operations: A final report*, NCJ 187104, National Institute of Justice, U. S. Department of Justice: Washington, DC., http://www.ncjrs.org/pdffiles1/nij/187104.pdf.

LeBeau, J. L. (1992). Four case studies illustrating the spatial-temporal analysis of serial rapists. *Police Studies*, 15(3), 124-145.

LeBeau, J. L. (1987a). The journey to rape: geographic distance and the rapist's method of approaching the victim, *Journal of Police Science and Administration*, 15 (2), 129-136.

LeBeau, J. L. (1987b). The methods and measures of centrography and the spatial dynamics of rape, *Journal of Quantitative Criminology*, 3 (2), 125-141.

Lee, Jay & Wong, D. W. S. (2001). *Statistical Analysis with ArcView GIS*. J. Wiley & Sons, Inc.: New York.

Lee, P. M. (2004). *Bayesian Statistics: An Introduction* (third edition). Hodder Arnold: New York.

Lees, B. (2006). "The spatial analysis of spectral data: Extracting the neglected data", *Applied GIS*, 2 (2), 14.1-14.13.

Lefever, D. (1926). Measuring geographic concentration by means of the standard deviational ellipse. *American Journal of Sociology*, 32(1): 88-94.

References (continued)

Leitner, M. (2007). Assessment and evaluation of individually calibrated journey to crime geographic profiling models. Ninth Crime Mapping Research Conference, National Institute of Justice. Pittsburgh, PA. March.

Leitner, M. & Kent, J. (2009). Bayesian journey to crime modeling of single- and multiple crime type series in Baltimore County, MD. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 213-236.

Leonard, T. & Hsu, J. S. J. (1999). *Bayesian Methods: An Analysis for Statisticians and Interdisciplinary Researchers*. Cambridge University Press: Cambridge.

Levine, N. (2011a). Spatial variation in motor vehicle crashes by gender in the Houston Metropolitan Area. *Proceedings of the 4th International Conference on Women's Issues in Transportation. Volume II: Technical Papers*, Transportation Research Board: Washington, DC. 12-25. http://onlinepubs.trb.org/onlinepubs/conf/cp46v2.pdf. Accessed May 7, 2012.

Levine, N. (2011b). "Spatial variation in motor vehicle crashes by gender in the Houston Metropolitan Area". *Proceedings of the 4th International Conference on Women's Issues in Transportation. Volume II: Technical Papers*, Transportation Research Board: Washington, DC. 12-25. http://onlinepubs.trb.org/onlinepubs/conf/cp46v2.pdf.

Levine, N. (2009a). "A motor vehicle safety planning support system: The Houston experience". In S. Geertman and J. Stillwell, *Planning Support Systems: Best Practice and New Methods*. Springer. 93-111.

Levine, Ned (2009b). Introduction to the special issue on Bayesian Journey-to-crime modeling. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 167-185.

Levine, N. (2008). "The 'hottest' part of a crime hotspot: Comments on "The utility of hotspot mapping for predicting spatial patterns of crime" by Spencer Chainey, Lisa Tompson, and Sebastian Uhlig". *Security Journal*, 21, 295-302.

Levine, N. (2007a).*CrimeStat: A Spatial Statistics Program for the Analysis of Crime Incident Locations* (version 3.1). Ned Levine & Associates, Houston, TX, National Institute of Justice, Washington, DC.

References (continued)

Levine, N. (2007b). Crime travel demand and bank robberies: Using CrimeStat III to model bank robbery trips. *Social Science Computer Review*, 25(2), 239-258.

Levine, N. (2005). *"The evaluation of geographic profiling software: Response to Kim Rossmo's critique of the NIJ methodology".* http://www.nedlevine.com/Response to Kim Rossmo Critique of the GP Evaluation Methodology.May 8 2005.doc

Levine, N. (2004).*CrimeStat: A Spatial Statistics Program for the Analysis of Crime Incident Locations* (version 3.0). Ned Levine & Associates, Houston, TX, National Institute of Justice, Washington, DC.

Levine, N. (2002).*CrimeStat: A Spatial Statistics Program for the Analysis of Crime Incident Locations* (version 2.0). Ned Levine & Associates, Houston, TX, National Institute of Justice, Washington, DC.

Levine, N. (2000). *CrimeStat: A Spatial Statistics Program for the Analysis of Crime Incident Locations* (version 1.1). Ned Levine & Associates, Annandale, VA., National Institute of Justice, Washington, DC.

Levine, N. (1999a) *CrimeStat: A Spatial Statistics Program for the Analysis of Crime Incident Locations* (version 1.0). Ned Levine & Associates, Annandale, VA., National Institute of Justice, Washington, DC.

Levine, N. (1999b). The effects of local growth management on regional housing production and population redistribution in California, *Urban Studies*. 1999. 36 12, 2047-2068.

Levine, N. (1996). Spatial statistics and GIS: software tools to quantify spatial patterns. *Journal of the American Planning Association*. 62 (3), 381-392.

Levine, N. & Block, R. (2010). Bayesian Journey-to-Crime Estimation: An Improvement in Geographic Profiling Methodology. *The Professional Geographer*. 63(2), 213-229.

Levine, N. & Canter, P. (2011). Linking origins with destinations for DWI motor vehicle crashes: An application of crime travel demand modeling. *Crime Mapping*, 3, 7-41.

References (continued)

Levine, N. & Kim, K. E. (1999). The spatial location of motor vehicle accidents: A methodology for geocoding intersections. *Computers, Environment, and Urban Systems*. 22 (6), 557-576.

Levine, N., Kim, K. E.. & Nitz, L. H. (1995a). Spatial analysis of Honolulu motor vehicle crashes: I. Spatial patterns. *Accident Analysis & Prevention*, 27(5), 663-674.

Levine, N., Kim, K. E..& Nitz, L. H. (1995b). Spatial analysis of Honolulu motor vehicle crashes: II. Generators of crashes. *Accident Analysis & Prevention*, 27(5), 675-685.

Levine, N. & Lee, P. (2013). Crime travel of offenders by gender and age in Manchester, England. Leitner, M. (ed), *Crime Modeling and Mapping Using Geospatial Technologies*, Springer. 145-178.

Levine, N. & Lee, P. (2009). Bayesian journey to crime modeling of juvenile and adult offenders by gender in Manchester. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 237-251.

Levine, N. & Wachs, M. (1986a). Bus Crime in Los Angeles: I - Measuring The Incidence. *Transportation Research*. 20 (4), 273-284.

Levine, N. & Wachs, M. (1986b). Bus Crime in Los Angeles: II - Victims and Public Impact. *Transportation Research*. 20 (4), 285-293.

Levine, N. Wachs, M. & Shirazi, E. (1986). Crime at Bus Stops: A Study of Environmental Factors. *Journal of Architectural and Planning Research*. 3 (4), 339-361.

Levinson, D. & Kumar, A. (2007). Density and journey to work. Manuscript, University of Minnesota. http://ideas.repec.org/p/nex/wpaper/density.html

Lind, A. W. (1930). Some ecological patterns of community disorganization in Honolulu. *American Journal of Sociology*, 36 (2). 206-220.

Lord, D., Geedipally, S. R., & Guikema, S. D. (2010). Extension of the application of Conway-Maxwell-Poisson Models: Analyzing traffic crash data exhibiting under-dispersion, *Risk Analysis*, 30 (8), 1268-1276.

References (continued)

Lord, D., Guikema, S. D., & Geedipally, S. R. (2008). Application of the Conway–Maxwell–Poisson Generalized Linear Model for analyzing motor vehicle crashes, *Accident Analysis & Prevention*, 40 (3), 1123–1134.

Lord, D. & Miranda-Moreno, L. F. (2008). Effects of low sample mean values and small sample size on the estimation of the fixed dispersion parameter of Poisson-gamma models for modeling motor vehicle crashes: A Bayesian Perspective. *Safety Science*, 46 (5), 751-770.

Lord, D. (2008) Methodology for estimating the variance and confidence intervals of the estimate of the product of baseline models and AMFs. *Accident Analysis & Prevention*, 40 (3), 1013-1017.

Lord, D. (2006). Modeling motor vehicle crashes using Poisson-gamma models: Examining the effects of low sample mean values and small sample size on the estimation of the fixed dispersion parameter. *Accident Analysis and Prevention*, 38, 751-766.

Los Angeles Times (1998). *Eye on the Sky*. Business section, July 20.

Lottier, S. (1938). Distribution of criminal offences in metropolitan regions, *Journal of Criminal Law, Criminology, and Police Science*, 29, 37-50.

Lundrigan, S., & Canter, D., (2001) A multivariate analysis of serial murderers' disposal site location choice in Journal of Environmental Psychology, 21, 423-432.

Lynch, S. M. (2007). *Introduction to Applied Bayesian Statistics and Estimation for Social Scientists*. Springer: New York.

Ma, J., Kockelman, K. M., & Damien, P. (2008). A multivariate Poisson-lognormal regression model for prediction of crash counts by severity, using Bayesian methods. *Accident Analysis & Prevention*, 40 (3), 964-975.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *5th Berkeley Symposium on Mathematics, Statistics and Probability*. Vol 1, 281-298.

McBratney, A. B. & deBruijter, J. J. (1992). A continuum approach to soil classification by modified fuzzy k-means with extragrades, *Journal of Soil Science*, 43, 159-175.

References (continued)

McCullagh, P. & Nelder, J. A. (1989). *Generalized Linear Models* (2nd edition). Chapman & Hall/CRC: Boca Raton, FL.

McCormick Rankin (2011). *Transportation Demand Management Plan: Final Report*. Ottawa. http://ottawa.ca/cs/groups/content/@webottawa/documents/pdf/mdaw/mdc3/~edisp/cap078202.pdf. Accessed June 1, 2012.

McClain, J. O. (1988). Dominant time series monitoring methods, *International Journal of Forecasting*, 4, 563–572.

McDonnell, P. W. Jr. (1979). *Introduction to Map Projections.* New York: Marcel Dekker, Inc.

McCullagh, P. & Nelder, J. A. (1989). *Generalized Linear Models* (2nd edition). Chapman & Hall/CRC: Boca Raton, FL.

McFadden, D. L. (2002). The path to discrete-choice models. *Access*, No. 20, Spring. 20-25. http://www.uctc.net/access/access20.shtml. Accessed April 28, 2012.

McFadden, D. (1980). Econometric Models for Probabilistic Choice Among Products. *The Journal of Business, 53*(3), S13-S29.

McFadden, D. (1973). Conditional Logit Analysis of Qualitative Choice Behavior, in Zarembka, P. (ed.), *Frontiers in Econometrics*, New York, Academic.

McFadden, D. L. & Train, K. (2000). Mixed MNL model for discrete response, *Journal of Applied Econometrics*, 15 (5), 447-470.

McFadden, D. L. & Train, K. (1986). *Qualitative Choice Analysis: Theory, Econometrics, and an Application to Automobile Demand*, MIT Press: Cambridge.

McGuckin, N. A. & Srinivasan, N. (2003). *Journey to Work in the United States and its Major Metropolitan Areas.* FHWA-EP-03-058, Office of Planning, Federal Highway Administration: Washington, DC.

McQuitty, L. L. (1960). Hierarchical syndrome analysis. *Educational and Psychological Measurement*, 20, 293-304.

References (continued)

Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, E. & Winkler, R. (1982), The accuracy of extrapolation (time series) methods: Results of a forecasting competition, *Journal of Forecasting*, 1, 111-153.

Maling, D. H. (1973). *Coordinate Systems and Map Projections* (1973). George Philip & Sons, London.

Malkiel, B. G. (1999). *A Random Walk Down Wall Street* (revised edition). W. W. Norton & Company: New York.

Maltz, M. D., Gordon, A. C., & Friedman, W. (1990). *Mapping Crime in Its Community Setting: Event Geography Analysis*. Springer-Verlag: New York.

Mantel, N. (1967). The detection of disease clustering and a generalized regression approach. *Cancer Research*, 27, 209-220.

Mantel, N. & Bailar, J. C. (1970). A class of permutational and multinomial test arising in epidemiological research, *Biometrics*, 26, 687-700.

MapInfo (1998). *MapInfo Professional 5.0.1*. MapInfo Corporation: Troy, NY.

Marcon, E. & Puech, F. (2003). Evaluating the geographic concentration of industries using distance-based methods. *Journal of Economic Geography*, 3, 409-428.

Mardia, K.V. (1972). *Statistics of Directional Data*. Academic Press: New York.

Massey, F. J., Jr (1951). The distribution of the maximum deviation between two sample cumulative step functions. *Annuals of Mathematical Statistics*, 22, 125-128.

Mather, A. S. (1986). *Land Use*. John Wiley & Sons: New York.

Messner, S. (1986). Economic inequality and levels of urban homicide, *Criminology*, 23, 297-317.

Messner, S. & Tardiff, K. (1986). The social ecology of urban homicide: an application of the 'Routine Activities approach'. *Criminology*, 22, 241-267.

References (continued)

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, & E. Teller (1953). Equations of state calculations by fast computing machines, *Journal of Chemical Physics*, 21, 1087-91.

Miaou, S. P. (2006). Coding instructions for the spatial regression models in CrimeStat. Unpublished manuscript. College Station, TX.

Miaou, S. P., Song, J. J., & Mallick, B. K. (2003). Roadway traffic crash mapping: a space-time modeling approach, *Journal of Transportation and Statistics*, 6 (1), 33-57.

Miaou, S. P. (1996). *Measuring the Goodness-of-Fit of Accident Prediction Models*. FHWA-RD-96-040. Federal Highway Administration, U.S. Department of Transportation: Washington, DC.

Microsoft (2012). *Excel*^{TM}. Microsoft Corporation: Redmond, WA.

Microsoft (1999). *Welcome to the ODBC Section of the Microsoft Universal Data Access Web Site*. Microsoft: Redmond, WA. http://www.microsoft.com/data/obdc.

Microsoft (2007). *Windows Vista*. Microsoft: Redmond, WA.

Microsoft (2002). *Windows XP*. Microsoft: Redmond, WA.

Microsoft (2012). SKEW - skewness function, *Microsoft Office Excel 2010, Microsoft: Redmond, WA. http://office.microsoft.com/en-us/excel-help/skew-HP005209261.aspx. Accessed May 21, 2012.*

Microsoft (1998a). *Windows NT Workstation 4.0*. Microsoft: Redmond, WA.

Microsoft (1998b). *Windows NT Server 4.0*. Microsoft: Redmond, WA.

Microsoft (1998c). *Windows 98*. Microsoft: Redmond, WA.

Microsoft (1995). *Windows 95*. Microsoft: Redmond, WA.

References (continued)

Miller, E. J. & Salvini, P. A. (1999). Activity-based travel behavior modeling in a microsimulation framework. Paper presented at IATBR Conference, Austin, TX. December. http://www.civ.utoronto.ca/sect/traeng/ilute/downloads/conference_papers/miller-salvini_iatbr-97.pdf. Accessed May 4, 2012.

Mitra, S. & Washington, S. (2007). On the nature of over-dispersion in motor vehicle crash prediction models, *Accident Analysis and Prevention*, 39, 459-468.

Moran, P. A. P. (1950). Notes on continuous stochastic phenomena. *Biometrika*, 37, 17-23.

Moran, P. A. P. (1948). The interpretation of statistical maps. *Journal of the Royal Statistical Society B*, 10, 243-251.

Mungiole, M., Pickle, L. W., & Simonson, K. H. (2002). Application of a weighted Head-Banging algorithm to Mortality data maps, *Statistics in Medicine*, 18, 3201-3209.

Mungiole, M. & Pickle, L. W. (1999). Determining the optimal degree of smoothing using the weighted head-banging algorithm on mapped mortality data. In ASC '99 - Leading Survey & Statistical Computing into the New Millennium, Proceedings of the ASC International Conference, September. Available at http://srab.cancer.gov/headbang.

Murray, A.T. & Grubesic, T. H. 2002. Identifying Non-hierarchical Clusters. *International Journal of Industrial Engineering*, 9, 86-95.

Myers, R. H. (1990) Classical and Modern Regression with Applications, 2nd edition, Duxbury Press, Belmont, CA.

Nannen, V. (2003). *The Paradox of Overfitting*. Artificial Intelligence, Rijksuniversitat: Groningen, Netherlands. http://volker.nannen.com/pdf/the_paradox_of_overfitting.pdf. Accessed March 11, 2010.

NARC (2012). *Welcome to NARC*. National Association of Regional Councils: Washington, DC. http://www.narc.org/. Accessed May 7, 2012.

References (continued)

NCHRP (1998). *Integration of Land Use Planning with Multimodal Transportation Planning*. Project 8-32(3). Prepared by Parsons Brinkerhoff Quade & Douglas, Inc. for the National Cooperative Highway Research Program, Transportation Research Board, National Research Council: Washington DC. October.

NCHRP (1995). *Travel Estimation Techniques for Urban Planning*. Project 8-29(2). National Cooperative Highway Research Program, Transportation Research Board: Washington, DC. http://www.trb.org/main/blurbs/160284.aspx. Accessed May 29, 2012.

Needham, R. M. (1967). Automatic classification in linguistics. *The Statistician*, 17, 45-54.

Neft, D. S. (1962). *Statistical Analysis for Areal Distributions*. Ph.D. dissertation, Columbia University: New York.

Neill, D. B. (2009). Expectation-based scan statistics for monitoring spatial time series data. *International Journal of Forecasting* 25: 498–517.

Newell, A., Shaw, J. C. & Simon, H. A. (1957). Empirical Explorations of the Logic Theory Machine, Proceedings of the Western Joint Computer Conference, pp. 218-239.

Newman, O. (1972). *Defensible Space: Crime Prevention Through Urban Design*. Macmillan: New York.

Nilsson, N. J. (1980). *Principles of Artificial Intelligence*. Morgan Kaufmann Publishers, Inc.: Los Altos, CA.

NIST (2004). Gallery of distributions. *Engineering Statistics Handbook*. National Institute of Standards and Technology: Washington, DC. http://www.itl.nist.gov/div898/handbook/eda/section3/eda366.htm.

Normandeau, A. (1967). *Trends and Patterns in Robbery: Philadelphia, PA, 1960-66*. Ph.D. Dissertation, University of Pennsylvania: Philadelphia.

Ntzourfras, I. (2009). *Bayesian Modeling using WinBugs*. Wiley Series in Computation Statistics, Wiley: New York.

References (continued)

Oh, J., Lyon, C., Washington, S., Persaud, B., & Bared , J. (2003). Validation of FHWA crash models for rural intersections: lessons learned. Transportation Research Record 1840, 41-49.

Oppenheim, N. (1980). *Applied Models in Urban and Regional Analysis*. Prentice-Hall, Inc.: Englewood Cliffs, NJ.

O'Leary, M. (2009). The mathematics of geographical profiling. *Journal of Investigative Psychology & Offender Profiling*. 6(3), 253-265.

Openshaw, S. (1984). *The Modifiable Areal Unit Problem.* Norwich: Geo Books. ISBN 0-86094-134-5.

Openshaw, S. A., Craft, A. W., Charlton, M., & Birch, J. M. (1988). Invetigation of leukemia clusters by use of a geographical analysis machine, *Lancet*, 1, 272-273.

Openshaw, S. A., Charlton, M., Wymer, C. & Craft, A. W. (1987). A mark 1 analysis machine for the automated analysis of point data sets, *International Journal of Geographical Information Systems*, 1, 335-358.

Ord, J.K. & Getis, A. (1995). Local spatial autocorrelation statistics:Distributional Issues and an Application. *Geographical Analysis*, Vol. 27, 1995, 286-306.

Ortuzar, J. D. & Willumsen, L. G. (2001). *Modeling Transport* (3rd edition). J. Wiley & Sons: New York.

Ottawa (2008). *Transportation Master Plan*. Regional Municipality of Ottawa-Carleton. http://ottawa.ca/en/city_hall/planningprojectsreports/master_plans/tmp/. Accessed June 1, 2012.

Palfrey, T. R., & Poole, K. T. (1987). The Relationship between Information, Ideology, and Voting Behavior. *American Journal of Political Science*, *31*(3), 511-530.

Papachristos, A. (2003). The social structure of gang homicides in Chicago. Annual conference of the American Society of Criminologists, Denver.

Park, B. J. (2009). Note on the Bayesian analysis of count data. From Park, Byung-Jung PhD thesis, Texas A & M University: College Station, TX.

References (continued)

Park, B. J. & Lord, D. (2008). Adjustment for the maximum likelihood estimate of the negative binomial dispersion parameter. *Transportation Research Record,* 2061, 9-19.

Park, B. J. & Lord, D. (2008). Adjustment for the maximum likelihood estimate of the negative binomial dispersion parameter. *Transportation Research Record,* 2061, 9-19.

Park, E.S., & Lord, D. (2007). Multivariate Poisson-Lognormal Models for Jointly ModelingCrash Frequency by Severity. In Transportation Research Record 2019: Journal of theTransportation Research Board, TRB, National Research Council, Washington, D.C., 1-6.

Park, R. & Burgess, E. (1924). *Introduction to the Science of Sociology*. Chicago University Press: Chicago.

Parzen, E. (1962). On the estimation of a probability density and mode. *Annuals of Mathematical Statistics*, 33, 1065-1076.

Paulsen, D. (2007). Improving geographic profiling through commuter/marauder prediction.. *Police Practice and Research* 8: 347-357

Paulsen, D. (2006a). Connecting the dots: assessing the accuracy of geographic profiling software. *Policing: An International Journal of Police Strategies and Management*. 20 (2), 306-334.

Paulsen, D. (2006b). Human versus machine: A comparison of the accuracy of geographic profiling methods. *Journal of Investigative Psychology and Offender Profiling* 3: 77-89.

Pettiway, L. E. (1995). Copping crack: The travel behaviour of crack users. *Justice Quarterly, 12*(3), 499-524.

Phillips, P.D. (1980) Characteristics and typology of the journey to crime. In Georges-Abeyie, D. E. & Harries, K. D. (eds), *Crime: A Spatial Perspective*, Columbia Univ. Press: New York, 156-166.

Phillips, S. (2003). The Social Structure of Vengeance: A Test of Black's Model. *Criminology, 41*(3), 673-708.

References (continued)

Pickle, L. W. & Su, Y. (2002). Within-State geographic patterns of health insurance coverage and health risk factors in the United States, *American Journal of Preventive Medicine*, 22 (2), 75-83.

Pickle, L. W., Mungiole, M., Jones, G. K., & White, Andrew A. (1996). *Atlas of United States Mortality*. National Center for Health Statistics: Hyattsville, MD.

Porojan, A. (2000). Trade flows and spatial effects: the Gravity Model revisited. Conference on Managing Economic Transition in Eastern Europe: Emerging Research Issues. The Manchester Metropolitan University: Manchester, England, January. http://www.business.mmu.ac.uk/research/met/papers/aporojan.pdf .

Portland (1998). *Bicycle Master Plan*. Resolution 35515, Office of Transportation, City of Portland: Portland, OR. http://www.portlandonline.com/transportation/index.cfm?a=369990&c=49304. Accessed June 1, 2012.

Porter, C., Suhrbier, J. & Schwartz, W. L. (1999). Forecasting bicycle and pedestrian travel: State of the practice and research needs. *Transportation Research Record*, 1674, 94-101.

Preparata, F. & Hong, S. J. (1977). Convex hulls of finite sets of points in two and three dimensions, *Comm. ACM*, 20, 87-93.

Pyle, G. F. (1974). *The Spatial Dynamics of Crime*. Department of Geography Research Paper No. 159, University of Chicago: Chicago.

Pyle, G. F., Hanten. E. W., Williams, P. G., Pearson, II, A. L. Doyle, J. G. & Kwofie, K. (1974). *The Spatial Dynamics of Crime*. Department of Geography, University of Chicago: Chicago.

Rabin, S. (2000a). A* aesthetic optimizations. In DeLoura, M.. *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 264-271.

Rabin, S. (2000b). A* speed optimizations. In DeLoura, M., *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 272-287.

References (continued)

Radford, N. (2006). The problem of overfitting with maximum likliehood . CSC 411: Machine Learning and Data Mining, University of Toronto: Toronto, CA. http://www.cs.utoronto.ca/~radford/csc411.F06/10-nn-early-nup.pdf Accessed March 11, 2010.

Radford, N. (2003). Slice sampling, *Annals of Statistics*, 31(3), 705-767.

Rand, A. (1986). Mobility triangles. In Figlio, R. M., Hakim, S. & Rengert, G. (ed), *Metropolitan Crime Patterns*. Criminal Justice Press: Monsey, NY, 117-126.

Ratcliffe, J.H. (2008). The magnitude of the crime challenge (Chapter 3). *Intelligence-Led Policing*, Willan Publishing: Cullompton.

Ravenstein, E. G. (1885). The laws of migration. *Journal of the Royal Statistical Society*. 48.

Recker, W. (2000). A bridge between travel demand modeling and activity-based travel analysis. *Center for Activity Systems Analysis*. Paper UCI-ITS-AS-WP-00-11. http://repositories.cdlib.org/itsirvine/casa/UCI-ITS-AS-WP-00-11/. Accessed May 23, 2012.

Reilly, W. J. (1929). Methods for the study of retail relationships. *University of Texas Bulletin*, 2944.

Rengert, G., Piquero, A. R., & Jones, P. R. (1999). Distance decay re-examined, *Criminology*, 37 (2), 427-445.

Rengert, G. F. (1995). Comparing cognitive hot spots to crime hot spots. In Block, C. R., Dabdoub, M. & Fregly, S., *Crime Analysis Through Computer Mapping*. Police Executive Research Forum: Washington, DC., 33-47.

Rengert, G. F. (1981). Burglary in Philadelphia: a critique of the opportunity structure model. In Brantingham, P. J. & Brantingham, P. L., *Environmental Criminology*. Waveland Press, Inc.: Prospect Heights, IL, 189-202.

Rengert, G. F. (1975). Some effects of being female on criminal spatial behavior. *The Pennsylvania Geographer*, 13 (2), 10-18.

References (continued)

Renshaw, E. (1985). Computer simulation of sitka spruce: spatial branching models for canopy growth and root structure. *Journal of Mathematics Applied in Medicine and Biology*, 2, 183-200.

Repetto, T. A. (1974). *Residential Crime*. Ballinger: Cambridge, MA.

Rhodes, W. M. & Conly, C. (1981). Crime and mobility: an empirical study. In Brantingham, P. J. & Brantingham, P. L., *Environmental Criminology*. Waveland Press, Inc.: Prospect Heights, IL, 167-188.

Rich, T. (2001). Crime mapping and analysis by community organizations in Hartford, Connecticut, *Research in Brief*. National Institute of Justice, U. S. Department of Justice: Washington, DC., http://www.ncjrs.org/pdffiles1/nij/185333.pdf.

Rich, T., & Shively, M. (2004). *A Methodology for Evaluating Geographic Profiling Software*. Final Report for the National Institute of Justice, Abt Associates: Cambridge, MA. http://www.ojp.usdoj.gov/nij/maps/gp.pdf

Ripley, B. D (1981). *Spatial Statistics*. John Wiley & Sons: New York.

Ripley, B. D. (1976). The second-order analysis of stationary point processes. *Journal of Applied Probability* 13: 255-66.

Robinson, A. H., Sale, R. D., Morrison, J. L. & Muehrcke, P. C. (1984). *Elements of Cartography* (5th edition). J. Wiley & Sons: New York.

Roemer, F. & Sinha, K. (1974). Personal security in buses and its effects on ridership in Milwaukee, *Transportation Research Record*, 487, 13-25.

Rosenblatt, M. (1956). Remarks on some non-parametric estimates of a density function. *Annuals of Mathematical Statistics*, 27, 832-837.

Rossmo, D. K. (2005a). Geographic heuristics or shortcuts to failure?: Response to Snook et al. *Applied Cognitive Psychology* 19: 651-654.

Rossmo, D. K. (2005b). Response to NIJ's methodology for evaluating geographic profiling software. http://www.ojp.usdoj.gov/nij/maps/gp.htm.

References (continued)

Rossmo, D. K. & Filer, S. (2005). Analysis versus guesswork. *Blue Line Magazine*, , August / September, 24:26.

Rossmo, D. K. (2000). *Geographic Profiling*. CRC Press: Boca Raton Fl.

Rossmo, D. K. (1997). Geographic profiling. In Jackson, J. L. & Bekerian, D. A. *Offender Profiling: Theory, Research and Practice*. John Wiley & Sons: Chicester, 159-175.

Rossmo, D. K. (1995). Overview: multivariate spatial profiles as a tool in crime investigation. In Block, C. R., Dabdoub, M. & Fregly, S., *Crime Analysis Through Computer Mapping*. Police Executive Research Forum: Washington, DC., 65-97.

Rossmo, D. K. (1993a). Multivariate spatial profiles as a tool in crime investigation. In Block, C. R. & Dabdoub, M. (eds), *Workshop on Crime Analysis Through Computer Mapping: Proceedings*. Illinois Criminal Justice Information Authority and Loyola University Sociology Department: Chicago. (Library of Congress HV7936.C88 W67 1993).

Rossmo, D. K. (1993b). Target patterns of serial murderers: a methodological model. *American Journal of Criminal Justice*, 17, 1-21.

Rushton, G. (1979). *Optimal Location of Facilities*. COMPress: Wentworth, NH.
SPSS, Inc. (1999). *SPSS 9.0 for Windows*. SPSS, Inc.: Chicago.

SAS Institute Inc. (1998). *Statistical Analysis System, Version 7*. Cary, NC.

Schachter, J. (2001). Geographical mobility: March 1999 to March 2000. *Current Population Reports*, P20-538, March. U.S. Census Bureau: Hyattsville, MD.

Schnell, J. B., Smith, A. J., Dimsdale, K. R. & Thrasher, L. J. (1973). *Vandalism and Passenger Security: A Study of Crime and Vandalism on Urban Mass Transit Systems in the United States and Canada*. Prepared by the American Transit Association for the Urban Mass Transportation Administration (now Federal Transit Administration), U. S. Department of Transportation. National Technical Information Service: Springfield, VA. PB 236-854.

References (continued)

Schwartz, W.L., Porter, C. D., Payne, G. C., Suhrbier, J. H., Moe, P. C. & Wilkinson III, W. L. (1999). *Guidebook on Methods to Estimate Non-Motorized Travel: Overview of Methods*. Turner-Fairbanks Highway Research Center, Federal Highway Administration: McLean, VA. July. http://www.fhwa.dot.gov/publications/research/safety/pedbike/98165/index.cfm. Accessed June 1, 2012.

Scott, D. W. (1992). *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley & Sons: New York.

Sedgewick, R. (2002). *Algorithms in C++: Part 5 Graph Algorithms* (3rd edition). Addison-Wesley: Boston.

Sellers, K. S. & Shmueli, G. (2010), A flexible regression model for count data, *Annals of Applied Statistics*, 4 (2), 943-961.

Shaw, C. R. (1929). *Delinquency Areas*. University of Chicago Press: Chicago.

Shaw, C. R. & McKay, H. D. (1942). *Juvenile Delinquency in Urban Areas*. Chicago: University of Chicago Press.

Shaw, C. & McKay, H.D. (1972). *Juvenile Delinquency and Urban Areas* (revised edition). University of Chicago Press: Chicago.

Shekhar, S. & Chawla, S. (2003). *Spatial Databases: A Tour*. Prentice-Hall: Upper Saddle River, NJ.

Sherman, L.W. & Weisburd, D. (1995). General deterrent effects of police patrol in crime hot spots: a randomized controlled trial. *Justice Quarterly*. 12, 625-648.

Sherman, L. W.., Gartin, P. R. & Buerger, M. E. (1989). Hot spots of predatory crime: routine activities and the criminology of place. *Criminology*, 27(1), 27-56.

Shifton, Y., Ben-Akiva, M., Proussaloglu, K., de Jong, G., Popuri, Y., Kasturirangan, K., & Bekhor, S. (2003). Activity-based modeling as a tool for better understanding travel behaviour. *Conference Proceedings*. 10th International Conference on Travel Behaviour Research, Lucerne, Switzerland. August. http://www.ivt.ethz.ch/news/archive/20030810_IATBR/shiftan.pdf. Accessed May 23, 2012.

References (continued)

Shmueli G., Minka T., Kadane J.B., Borle S., & Boatwright, P.B (2005). A useful distribution for fitting discrete data: revival of the Conway–Maxwell–Poisson distribution. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 54(1), 127–142.

Shoup, D. (2002). Roughly right vs. precisely wrong. *Access*, No. 20, Spring. 20-25.

Siegel, S. (1956). *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill: New York.

Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman & Hall: London.

Simon, H. A. & Newell, A. (1963). The uses and limitations of models. In Marx, M. (ed), *Theories of Contemporary Psychology*, Macmillan: New York, 89-104.

Skiena, S. S. (1997). Convex hull. §8.6.2 in *The Algorithm Design Manual*. Springer-Verlag: New York, 351-354.

Smirnov, N. V. (1948). Table for estimating the goodness of fit of empirical distributions. *Annuals of Mathematical Statistics*, 19, 279-281.

Smith, T. S. (1976). Inverse distance variations for the flow of crime in urban areas. *Social Forces*, 25(4), 804-815.

Smith, W., Glave, S. & Davison, E. (2000). Furthering the integration of routine activity and social disorganization theories: Small units of analysis and the study of street robbery as a diffusion process. *Criminology*, 38, 489-523.

Sneath, P. H. A. (1957). The application of computers to taxonomy. *Journal of General Microbiology*, 17, 201-226.

Snook, B. (2004). Individual differences in distance travelled by serial burglars. *Journal of Investigative Psychology and Offender Profiling, 1*, 53-66.

Snook, B., Cullen, R. M., Mokros, A., & Harbort, S. (2005). Serial murderers' spatial decisions: factors that influence crime location choice. *Journal of Investigative Psychology and Offender Profiling, 2*, 147-164.

References (continued)

Snook, B., Zito, M., Bennell, C. & Taylor, P. J. (2005). On the complexity and accuracy of geographic profiling strategies. *Journal of Quantitative Criminology*, 21 (1), 1-26.

Snook, B., Taylor, P. & Bennell, C. (2004). Geographic profiling: the fast, fugal and accurate way. *Applied Cognitive Psychology* 18: 105-121.

Snyder, J. P. (1987). *Map Projections - A Working Manual*. U.S. Geological Survey Professional Paper 1395. U. S. Government Printing Office: Washington, DC.

Snyder, J. P. & Voxland, P. M. (1989). *An Album of Map Projections*. U.S. Geological Survey Professional Paper 1453. U. S. Government Printing Office: Washington, DC.

Sokal, R. R. & Sneath, P. H. A. (1963). *Principles of Numerical Taxonomy*. W. H. Freeman & Co.: San Francisco.

Sokal, R. R. & Michener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38, 1409-1438.

Son, D., Tsutakawa, R. K., Kim, H. & He, Z. (2000). Spatio-temporal interaction with disease mapping. *Statistics in Medicine*, 19, 2015-2035.

So, A. M., Ye, Y., & Zhang, J. (2007). Greedy algorithms for metric facility location problems. In Gonzalez, T. F. (ed), *Handbook of Approximation Algorithms and Metaheuristics*, CRC Computer & Information Sciences Series, Chapman & Hall/CRC: Boca Raton, FL, Chapter 39.

Springer (2001). Polya distribution, *Encyclopedia of Mathematics*, Springerlink: London, http://eom.springer.de/p/p073540.htm.

Spitzer, F. (1976). *Principles of Random Walk* (second edition). Springer: New York.

Stack, S. (1984). Income inequality and property crime, *Criminology*, 22, 229-257.

StatSoft (2010). Tolerance, *StatSoft Electronic StatisticsTextbook*, StatSoft:Tulsa, OK. http://www.statsoft.com/textbook/statistics-glossary/t/button/t/

Stephenson, L. (1980). Centrographic analysis of crime. In George-Abeyie, G. & Harries, K. D. (eds), *Crime, A Spatial Perspective*, Columbia University Press: New York.

References (continued)

Stewart, J. Q. (1950). The development of social physics. *American Journal of Physics*, 18, 239-53.

Stoe, D., Watkins, C. R. Kerr, J., Rost, L. & Craig, T. (2003). *Using Geographic Information Systems to Map Crime Victim Services: A Guide for State Victims of Crime Act Adminstrators and Victim Service Providers*. National Institute of Justice, U. S. Department of Justice: Washington, DC., http://www.ojp.usdoj.gov/ovc/publications/infores/geoinfosys2003/welcome.html.

Stopher, P. R. & Meyburg, A. H. (1975). *Urban Transportation Modeling and Planning.* Lexington, MA: Lexington Books.

Stouffer, S. A. (1940). Intervening opportunities: a theory relating mobility and distance. *American Sociological Review*, 5, 845-67.

Stout, B. (2000). The basics of A* for path planning. In DeLoura, M.. *Game Programming Gems*. Charles River Media, Inc.: Rockland, MA., 254-263.

Systat, Inc. (2008). *Systat 13: Statistics I*. SPSS, Inc.: Chicago.

Tabachnick, B. G. & Fidell, L. S.(1996). *Using Multivariate Statistics* (3rd ed). Harper Collins: New York.

Taylor, P. J. (1970). *Interaction and Distance: An Investigation into Distance Decay Functions and a Study of Migration at a Microscale*. PhD thesis, University of Liverpool: Liverpool.

Thompson, H. R. (1956). Distribution of distance to nth neighbour in a population of randomly distributed individuals. *Ecology*, 37, 391-394.

Thorndike, R. L. (1953). Who belongs in a family?. *Psychometrika*, 18, 267-276.

Thrasher, F. M. (1927). *The Gang*, University of Chicago Press: Chicago.

Thünen, J. H. Von (1826). *Der Isolierte Staat in Beziehung auf Landwirtshaft und Nationalökonomie* (The Isolated State in Relation to Agriculture). Hamburg.

Thurstone, L. L. (1947). Multiple-Factor Analysis. Chicago: University of Chicago Press.

References (continued)

Tita, G., & Griffiths, E. (2005). Traveling to Violence: The Case for a Mobility-Based Spatial Typology of Homicide. *Journal of Research in Crime and Delinquency*, *42*, 275-308.

Train, K. (2009). *Discrete Choice Methods with Simulation* (2nd edition). Cambridge University Press: Cambridge.

Train, K. E. (1980). A Structured Logit Model of Auto Ownership and Mode Choice. *The Review of Economic Studies, 47*(2), 357-370.

Trigg, D. W. (1964). Monitoring a forecasting system, *Operational Research Quarterly*, 15, 271–274.

Tukey, P. A. & Tukey, J. W. (1981). Graphical display of data sets in 3 or more dimensions. In Barnett, V. (ed), *Interpreting Multivariate Data*. John Wiley & Sons: New York.

Turnbull, B. W., Iwano, E.J., Burnett, W. S., Howe, H. L. & Clark, L. C. (1990). Monitoring for clusters of disease: application to leukemia incidence in upstate New York, *American Journal of Epidemiology*, 132, S136-S143.

Turner, S. (1969). Delinquency and distance. In Wolfgang, M. E. & Sellin, T. (eds), *Delinquency: Selected Studies*. John Wiley & Sons: New York.

Turner, S., Shunk, G. & Hottenstein, A. M. (1998). *Development of a Methodology to Estimate Bicycle and Pedestrian Travel Demand*. Report 1723-S, Texas Transportation Institute: College Station. http://tti.tamu.edu/publications/catalog/record/?id=146. Accessed April 28, 2012.

U.S. Census Bureau (2012). *Commuting (Journey to Work)*. U.S. Census Bureau: Washington, DC. http://www.census.gov/hhes/commuting.

U.S. Census Bureau (2011). Summary File 3 (SF3). U.S. Census Bureau: Washington, DC. http://www.census.gov/census2000/sumfile3.html.

U.S. Census Bureau (2009). Commuting (Journey to Work) Main. U.S. Census Bureau, U.S. Department of Commerce: Washington, DC. http://www.census.gov/hhes/commuting/. Accessed June 1, 2012.

References (continued)

U.S. Census Bureau (2004a). *TIGER/Line 2004*. Bureau of the Census, U. S. Department of Commerce: Washington, DC.

U.S. Census Bureau (2004b). *Journey to Work and Place of Work*. Bureau of the Census, U.S. Department of Commerce: Washington, DC. http://www.census.gov/population/www/socdemo/journey.html

U.S. Census (2003). Net Worth and Asset Ownership of Households: 1998 and 2003 (Table A). *Current Population Reports*, P70-88. U. S. Census Bureau, U. S. Department of Commerce: Washington, DC. http://www.census.gov/prod/2003pubs/p70-88.pdf. Accessed April 28, 2012.

U.S. Census Bureau (2000). All across the USA: Population distribution, 1999, In *Population Profile of the United States: 1999.* Bureau of the Census, U. S. Department of Commerce: Washington, DC., chapter 2.

USDOJ (2000). *Regional Crime Analysis Geographic Information System (RCAGIS)*. Criminal Division, U.S. Department of Justice: Washington, DC. http://www.usdoj.gov/criminal/gis/rcagishome.htm.

USDOT (2003). *Title XXIII, Part 450*. Code of Federal Regulations. Code of Federal Regulations, Title 23, Part 450, Volume 1. 23CFR450. Washington, DC.

van Koppen, P. J, van der Kemp, J. J. & Christianne J. P. (2002) Geografische daderprofilering (Geographic offender profiling) in van Koppen et al, *Het Recht Van Binnen: Psychologie Van Het Recht* (The Law from Inside: Psychology of the Law), Deventer, Netherlands Kluwer.

van Koppen, P. J., Christanne J. P. & Koppen, V. V. (2000). Cirkels van delicten: over pleegplaatsen van misdrijven en de woonplaats van de daders (Circles of crime: incident location and the residences of the offenders), *De Psycholoog: Psychologie en Recht* (The Psychologist, Psychology and Law), Oktober, 435-442.

van Koppen, P. J. & Jansen, R. W. J. (1998). The Road to robbery: travel patterns in commercial robberies. *British Journal of Criminology*, 38 (2), 230-246.

van Koppen, P. J. & de Keijser, J. W. (1997). Desisting distance decay: on the aggregation of individual crime trips. *Criminology*, 35 (3), 505-516.

References (continued)

Venables, W. N. & Ripley, B. D. (1997). *Modern Applied Statistics with S-Plus (second edition)*. Springer-Verlag: New York.

von Thünen, J. (1826). *The Isolated State in Relation to Agriculture and Political Economy*. English edition, van Suntum, Ulrich. Palgrave Macmillan:Houndsmills, Basingstoke, Hampshire, England, 2009.

Wachs, M., Taylor, B., Levine, N. & Ong, P. (1993). The Changing Commute: A Case Study of the Jobs/Housing Relationship Over Time. *Urban Studies*. 30 10, 1711-1729.

Wachter, S. M. & Cho, M. (1991). "Interjurisdictional price effects of land use controls". *Washington University Journal of Urban and Contemporary Law*, 40, 49-63.

Waller, L. A. & Gotway, C. A. (2004). *Applied Spatial Statistics for Public Health Data*. John Wiley & Sons: Hoboken, NJ.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*. 58, 236-244.

Warren, J., Reboussin, R., Hazelwood, R., Cummings, A., Gibbs, N. & Trumbetta, S. (1998). The distance correlates of serial rape. *Journal of Quantitative Criminology*, 14, 35-58.

Wartell, J. & McEwen, T. (2001). *Privacy in the Information Age: A Guide for Sharing Crime Maps and Spatial Data*. National Institute of Justice, U. S. Department of Justice: Washington, DC., http://www.ncjrs.org/pdffiles1/nij/188739.pdf.

WASHCOG (1974). *Citizen Safety and Bus Transit*. Metropolitan Washington Council of Governments. National Technical Information Service, Springfield, VA. PB 237-740/AS.

Weber, A. (1909). *Über den Standort der Industrien* (Theory of Location of Industries).

Weisburd, D., Groff, E. R., & Yang, S-M (2012). *The Criminology of Place*. Oxford University Press: New York.

Weisburd, D., Bushway, S., Lum, C. & Yang, S. (2004). Trajectories of crime at places: A longitudinal study of street segments in the City of Seattle. *Criminology*, 42 (2), 283-321.

References (continued)

Weisburd, D. & McEwen, T. (1998). *Crime Mapping Crime Prevention*. Criminal Justice Press: Monsey, NY.

Weisburd, D. & Green, L. (1995). Policing drug hot spots: the Jersey City drug market analysis experiment. *Justice Quarterly*. 12 (4), 711-735.

Weisburd, D., Maher, L.,& Sherman, L. (1992). Contrasting crime general and crime specific theory: the case of hot-spots of crime. *Advances in Criminological Theory*, 4, 45-70.

Weishart, D. (1969). Mode analysis. In Cole, A. J. (ed), *Numerical Taxonomy*, Academic Press: New York.

White, R. Clyde (1932). The relationship of felonies to environmental factors in Indianapolis. *Social Forces*, 10 (4), 488-509.

Whittle, P. (1958). On the smoothing of probability density functions. *Journal of the Royal Statistical Society, Series B*, 55, 549-557.

Whittle, P., 1954. On stationary process in the plane. *Biometrika*, 41, 434–449.

Wikipedia (2013a). Instrumental variable. Wikipedia. http://en.wikipedia.org/wiki/Instrumental_variable. Accessed January 31, 2013.

Wikipedia (2013b). Specification (regression). Wikipedia. http://en.wikipedia.org/wiki/Specification_(regression). Accessed January 31, 2013.

Wikipedia (2012a). Condition number. *Wikipedia.* *http://en.wikipedia.org/wiki/Condition_number*. Accessed March 19, 2010

Wikipedia (2012b). Geometric mean. http://en.wikipedia.org/wiki/Geometric_mean and "Weighted geometric mean" http://en.wikipedia.org/wiki/Weighted_geometric_mean.

Wikipedia (2012c). Greedy algorithm, *Wikipedia*. http://en.wikipedia.org/wiki/Greedy_algorithm. Accessed March 12, 2010.

Wikipedia (2012d). Harmonic mean. http://en.wikipedia.org/wiki/Harmonic_mean and "Weighted harmonic mean" http://en.wikipedia.org/wiki/Weighted_harmonic_mean.

References (continued)

Wikipedia (2012e). Maximum likelihood, *Wikipedia*. http://en.wikipedia.org/wiki/Maximum_likelihood. Accessed March 12, 2010.

Wikipedia (2012f). Negative binomial distribution, *Wikipedia*, http://en.wikipedia.org/wiki/Negative_binomial_distribution Accessed February 24, 2010.

Wikipedia (2012g). Modifiable Area Unit Problem. Wikipedia. http://en.wikipedia.org/wiki/Modifiable_areal_unit_problem

Wiles, P. & Costello, A. (2000). The 'road to nowhere': The evidence for travelling criminals, *Home Office Research Study, No. 207*. Research, Development and Statistics Directorate, London. http://www.homeoffice.gov.uk/rds/prgpdfs/brf400.pdf

Wilson, A. G. (1970). *Entropy in Urban and Regional Planning*. Leonard Hill Books: Buckinghamshire.

Wilson, J.Q. & Kelling, G. (1982) Broken Windows: The Police and Neighborhood Safety. *Atlantic Monthly*, March. 29-38.

White, R. C. (1932). The relation of felonies to environmental factors in Indianapolis. *Social Forces*, 10(4), 498-509.

Wooldridge, J. (2002). Examining the (Ir)Relevance of Aggregation Bias for Multilevel Studies of Neighborhoods and Crime with an Example Comparing Census Tracts to Official Neighborhoods in Cincinnati. *Criminology* 40:681-710.

Wong, D. W. S. & Lee, J. (2005). *Statistical Analysis of Geographic Information with ArcView GIS and ArcGIS*. J. Wiley & Sons, Inc.: New York.

Wright, R. T. & Decker, S. H. (1997). *Armed Robbers in Action: Stickups and Street Culture*. Northeastern University Press, Boston.

Xie, X. L. & Beni, G. (1991). A validity measure for fuzzy clustering. IEEE Trans. Pattern Analysis Machine Intell., 13, 841-847.

References (continued)

Zhao, F., Chow, L-F, Li, M-T, Gan, A., & Shen, D. L. (2001). *Refinement of FSUTMS Trip Distribution Methodology*. Lehman Center for Transportation Research, Florida International University: Miami, FL.

Appendix A:

Some Notes on the Statistical Comparison of Two Samples

By

Ned Levine

Ned Levine & Associates

Houston, TX

The following presents methods for testing the spatial differences between two distributions. At this point, *CrimeStat* does not include routines for testing the differences between two or more samples. The following is provided for the reader's information. Chapter 4 discussed the calculation of these statistics as a single distribution.

Differences in the Mean Center of Two Samples

For differences between two samples in the mean center, it is necessary to test both differences in the X coordinate and differences in the Y coordinates. Since *CrimeStat* outputs the mean X, the mean Y, the standard deviation of X, and the standard deviation of Y, a simple t-test can be set up. The null hypothesis is that the mean centers are equal

$$H_0: \quad \mu_{XA} = \mu_{XB} \tag{A.1}$$
$$\mu_{YA} = \mu_{YB} \tag{A.2}$$

and the alternative hypothesis is that the mean centers are not equal

$$H_1: \quad \mu_{XA} \neq \mu_{XB} \tag{A.3}$$
$$\mu_{YA} \neq \mu_{YB} \tag{A.4}$$

Because the true standard deviations of sample A, σ_{XA} and σ_{YA}, and sample B, σ_{XB} and σ_{YB}, are not known, the sample standard deviations are taken, S_{XA}, S_{YA}, S_{XB} and S_{YA}. However, since there are two different variables being tested (mean of X and mean of Y for groups 1 and 2), the alternative hypothesis has two fundamentally different interpretations:

Comparison I: That EITHER $\mu_{XA} \neq \mu_{XB}$ OR $\mu_{YA} \neq \mu_{YB}$ is true (A.5)

Comparison II: That BOTH $\mu_{XA} \neq \mu_{XB}$ AND $\mu_{YA} \neq \mu_{YB}$ are true (A.6)

In the first case, the mean centers will be considered not being equal if either the mean of X *or* the mean of Y are significantly different. In the second case, both the mean of *and* the mean of Y must be significantly different for the mean centers to be considered not equal. The first case is clearly easier to fulfill than the second.

Significance levels

By tradition, significance tests for comparisons between two means are made at the $\alpha \leq .05$ or $\alpha \leq .01$ levels, though there is nothing absolute about those levels. The significance levels are selected to minimize *Type 1 Errors*, inadvertently declaring a difference in the means when, in reality, there is not a difference. Thus, a test establishes that the likelihood of falsely rejecting the null hypothesis be less than one-in-twenty (less strict) or one-in-one hundred (more strict).

However, with multiple comparisons, the chances increase for finding 'significance' due to the multiple tests. For example, with two tests - a difference in the means of the X coordinate and a difference in the means of the Y coordinate, the likelihood of rejecting the first null hypothesis ($\mu_{XA} \neq \mu_{XB}$) is one-in-twenty and the likelihood of rejecting the second null hypothesis ($\mu_{YA} \neq \mu_{YB}$) is also one-in-twenty, then the likelihood of rejecting either one null hypothesis or the other is actually one-in-ten.

To handle this situation, comparison I - the 'either/or' condition, a Bonferoni test is appropriate (Anselin, 1995; Systat, 1996). Because the likelihood of achieving a given significance level increases with multiple tests, a 'penalty' must be assigned in finding either the differences in means for the X coordinate or differences in means for the Y coordinates significant. The Bonferoni criteria divides the critical probability level by the number of tests. Thus, if the $\alpha \leq .05$ level is taken for rejecting the null hypothesis, the critical probability for each mean must be .025 (.05/2); that is, differences in either the mean of X or mean of Y between two groups must yield a significance level less than .025.

For comparison II - the 'both/and' condition, on the other hand, the test is more stringent since the differences between the means of X and the means of Y must both be significant. Following the logic of the Bonferoni criteria, the critical probability level is multiplied by the number of tests. Thus, if the $\alpha = .05$ level is taken for rejecting the null hypothesis, then *both* tests must be significant at the $\alpha \leq .10$ level (i.e., .05*2).[1]

[1] There are limits to the Bonferoni logic. For example, if there were 10 tests, having a threshold significance

Tests

The statistics used are for the t-test of the difference between means (Kanji, 1993).

a. First, test for equality of variances by taking the ratio of the variances (squared sample standard deviations) of both the X and Y coordinates:

$$F_X = \frac{S_{XA}^2}{S_{XB}^2} \quad \text{(A.7)}$$

$$F_Y = \frac{S_{YA}^2}{S_{YB}^2} \quad \text{(A.8)}$$

with (N_A - 1) and (N_B - 1) degrees of freedom for groups *A* and *B* respectively. This test is usually done with the larger of the variances in the numerator. Since there are two variances being compared (for X and Y, respectively), the logic should follow either *I* or *II* above (i.e., if either are to be true, then the critical α will be actually α/2 for each; if both must be true, then the critical α will be actually 2*α for each).

b. Second, if the variances are considered equal, then a t-test for two group means with unknown, but equal, variances can be used (Kanji, 1993; 28). Let

$$S_{XAB} = \sqrt{\frac{\sum_{i=1}^{N_A}[(X_{iA}-\bar{X}_A)^2]+\sum_{i=1}^{N_B}[(X_{iB}-\bar{X}_B)^2]}{N_A+N_B-2}} \quad \text{(A.9)}$$

$$S_{YAB} = \sqrt{\frac{\sum_{i=1}^{N_A}[(Y_{iA}-\bar{Y}_A)^2]+\sum_{i=1}^{N_B}[(Y_{iB}-\bar{Y}_B)^2]}{N_A+N_B-2}} \quad \text{(A.10)}$$

where the summations are for i=1 to N within each group separately. Then the test becomes

---

level of .005 (.05 / 10) for the 'either/or' conditions and a threshold significance level of .50 (.05 * 10) for the 'both/and' would lead to an excessively difficult test in the first case and a much too easy test in the second. Thus, the Bonferoni logic should be applied to only a few tests (e.g., 5 or fewer).

$$t_X = \frac{(\bar{X}_A - \bar{X}_B) - (\mu_{XA} - \mu_{XB})}{S_{XAB}\sqrt{[\frac{1}{N_A} + \frac{1}{N_B}]}} \quad \text{(A.11)}$$

$$t_Y = \frac{(\bar{Y}_A - \bar{Y}_B) - (\mu_{YA} - \mu_{YB})}{S_{YAB}\sqrt{[\frac{1}{N_A} + \frac{1}{N_B}]}} \quad \text{(A.12)}$$

with ($N_A + N_B - 2$) degrees of freedom for each test.

c. Third, if the variances are not equal, then a t-test for two group means with unknown and unequal variances should be used (Kanji, 1993; 29).

$$S_{XA} = \sqrt{\frac{\sum_{i=1}^{N_A}[(X_{iA} - \bar{X}_A)^2]}{N_A - 1}} \quad \text{(A.13)}$$

$$S_{XB} = \sqrt{\frac{\sum_{i=1}^{N_B}[(X_{iB} - \bar{X}_B)^2]}{N_B - 1}} \quad \text{(A.14)}$$

$$S_{YA} = \sqrt{\frac{\sum_{i=1}^{N_A}[(Y_{iA} - \bar{Y}_A)^2]}{N_A - 1}} \quad \text{(A.15)}$$

$$S_{YB} = \sqrt{\frac{\sum_{i=1}^{N_B}[(Y_{iB} - \bar{Y}_B)^2]}{N_B - 1}} \quad \text{(A.16)}$$

$$t_X = \frac{(\bar{X}_A - \bar{X}_B) - (\mu_{XA} - \mu_{XB})}{\sqrt{[\frac{S_{XA}}{N_A} + \frac{S_{XB}}{N_B}]}} \quad \text{(A.17)}$$

$$t_Y = \frac{(\bar{Y}_A - \bar{Y}_B) - (\mu_{YA} - \mu_{YB})}{\sqrt{[\frac{S_{YA}}{N_A} + \frac{S_{YB}}{N_B}]}} \quad \text{(A.18)}$$

with degrees of freedom

$$v_X = \left\{ \frac{[\frac{S_{XA}^2}{N_A} + \frac{S_{XB}^2}{N_B}]}{\frac{S_{XA}^4}{N_A^2(N_A+1)} + \frac{S_{XB}^4}{N_B^2(N_B+1)}} \right\} - 2 \quad \text{(A.19)}$$

$$v_Y = \left\{ \frac{\left[\frac{S_{YA}^2}{N_A} + \frac{S_{YB}^2}{N_B}\right]}{\frac{S_{YA}^4}{N_A^2(N_A+1)} + \frac{S_{YB}^4}{N_B^2(N_B+1)}} \right\} - 2 \qquad \text{(A.20)}$$

for both the X and Y test. Even though this latter formula is cumbersome, in practice, if the sample size of each group is greater than 100, then the t-values for infinity can be taken as a reasonable approximation and the above degrees of freedom need not be tested:

i. t=1.645 for α=.05; t=2.326 for α=.01 for a one-tail test

ii. t=1.645 for α=.10; t=1.960 for α=.05; t=2.327 for α=.02; t=2.576 for α=.01 for a two-tail test

d. The significance levels are those selected above. For comparison I - that **either** differences in the means of X or differences in the means of Y are significant, the critical probability level is α/2 (e.g., .05/2 = .025; .01/2 = .005). For comparison II - that **both** differences in the means of X and differences in the means of Y are significant, the critical probability level is α*2 (e.g., .05*2 = .10; .01*2 - .02).

e. Reject the null hypothesis if:

Comparison I: Either tested t-value (t_x or t_y) is greater than the Critical t for α/2

Comparison II: Both tested t-values (t_x and t_y) are greater than the critical t for α*2

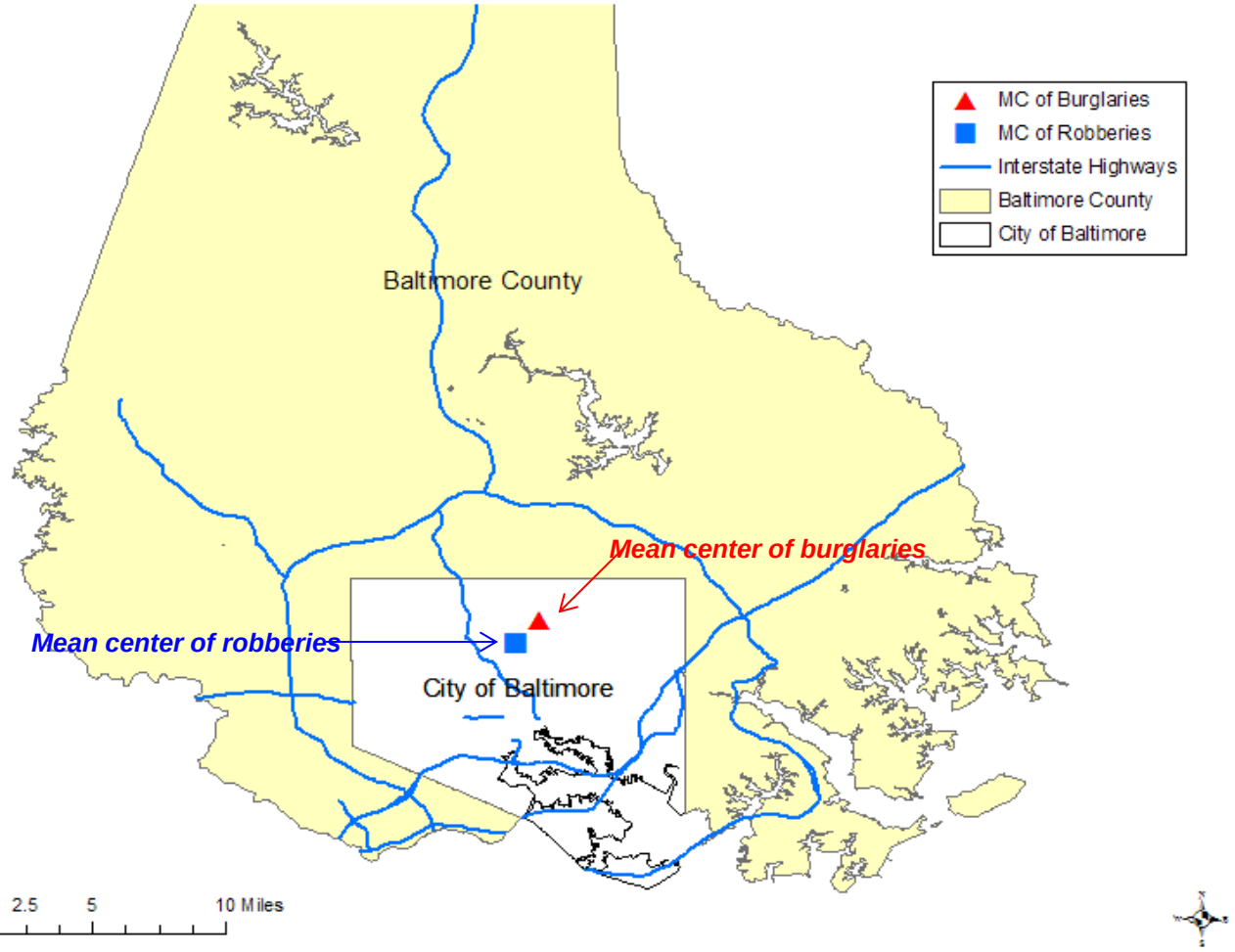
Example 1: Burglaries and Robberies in Baltimore County

To illustrate, compare the distribution of burglaries in Baltimore County with those of robberies, both for 1996. Figure A.1 shows the mean center of all robberies (blue square) and all residential burglaries (red triangle). As can be seen, the mean centers are located within Baltimore City, a property of the unusual shape of the county (which surrounds the city on three sides). Thus, these mean centers cannot be considered an unbiased estimate of the metropolitan area, but unbiased estimates for the County only. When the relative positions of the two mean centers are compared, the center of robberies is south and west of the center for burglaries. Is this difference significant or not?

Figure A.1:

1996 Burglaries and Robberies in Baltimore County, MD

Comparison of Mean Centers (MC)



To test this, the standard deviations of the two distributions are first compared and the F-test of the larger to the smaller variance is used (equations A.1 and A.2). *CrimeStat* provides the standard deviation of both the X and Y coordinates; the variance is the square of the standard deviation. In this case, the variance for burglaries is slightly larger than for robberies for both the X and Y coordinates.

$$F_X = \frac{S_{XA}^2}{S_{XB}^2} = \frac{0.0154}{0.0145} = 1.058 \quad \text{(A.21)}$$

$$F_Y = \frac{S_{YA}^2}{S_{YB}^2} = \frac{0.0058}{0.0029} = 2.007 \quad \text{(A.22)}$$

Because both samples are fairly large (1180 robberies and 6051 burglaries respectively), the degrees of freedom are also very large. The F-tables are a little indeterminate with large samples, but the variance ratio approaches 1.00 as the sample reaches infinity. An approximate critical F-ratio can be obtained by the next largest pair of values in the table (1.22 for $p \leq .05$ and 1.32 for $p \leq .01$). Using this criterion, differences in the variances for the X coordinate are probably not significant while that for the Y coordinates definitely are significant. Consequently, the test for a difference in means with unequal variances is used (equations A.17 and A.18).

$$t_X = \frac{(\bar{X}_A - \bar{X}_B) - (\mu_{XA} - \mu_{XB})}{\sqrt{[\frac{S_{XA}}{N_A} + \frac{S_{XB}}{N_B}]}} = \frac{\text{-76.608482} - (\text{-76.620838})}{\sqrt{[\frac{0.0154}{6051} + \frac{0.0145}{1180}]}} = \frac{0.0124}{0.0039} = 3.21 \ (\leq .005) \quad \text{(A.23)}$$

$$t_Y = \frac{(\bar{Y}_A - \bar{Y}_B) - (\mu_{YA} - \mu_{YB})}{\sqrt{[\frac{S_{YA}}{N_A} + \frac{S_{YB}}{N_B}]}} = \frac{39.348368 - 39.334816}{\sqrt{[\frac{0.0058}{6051} + \frac{0.0029}{1180}]}} = \frac{0.0136}{0.0018} = 7.36 \ (p \leq .005) \quad \text{(A.24)}$$

Therefore, whether we use the 'either/or' test (critical $\alpha \leq .025$) or the 'both/and' test (critical $\alpha \leq .1$), we find that the difference in the mean centers is highly significant. Burglaries have a different center of gravity than robberies in Baltimore County.

Differences in the Standard Distance Deviation of Two Samples

Since the standard distance deviation, S_{XY} (equation 4.6 in Chapter 4) is a standard deviation, differences in the standard distances of two groups can be compared with an equality of variance test (Kanji, 1993, 37),

$$F = \frac{S_{XYA}^2}{S_{XYB}^2} \quad \text{(A.25)}$$

with (N_A - 1) and (N_B - 1) degrees of freedom for groups A and B, respectively. This test is usually done with the larger of the variances in the numerator. Since there is only one variance being compared, the critical α are as listed in the tables.

Baltimore County Burglary Example (continued)

From *CrimeStat*, we find that the standard distance deviation of burglaries is 8.44 miles while that for robberies is 7.42 miles. The F-test of the difference is calculated by

$$F = \frac{S_{XYA}^2}{S_{XYB}^2} = \frac{8.44^2}{7.42^2} = 1.29 \quad \text{(A.26)}$$

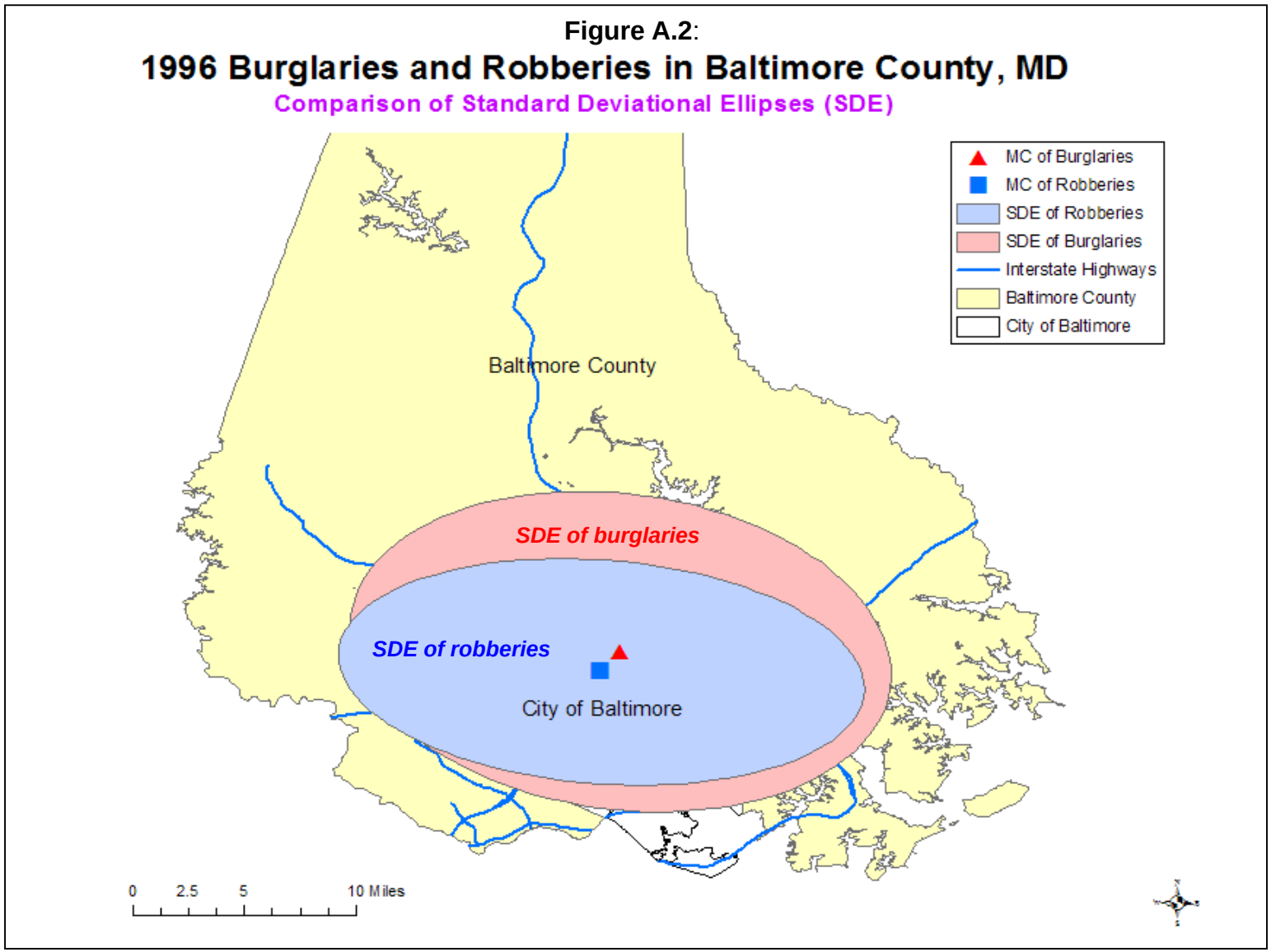
with 6050 and 1180 degrees of freedom respectively. Again, the F-tables are slightly indeterminate with respect to large samples, but the next largest F beyond infinity is 1.25 for $p \leq .05$ and 1.38 for $p \leq .01$. Thus, it appears that burglaries have a significantly greater dispersion than robberies, at least at the $p \leq .05$ level.

Differences in the Standard Deviational Ellipse of Two Samples

Figure A.2 shows the standard deviational ellipse of all robberies (light blue) and all residential burglaries (light red). As can be seen, the dispersion of incidents, as defined by the standard deviational ellipse, is greater for burglaries than for robberies. In a standard deviational ellipse, there are actually six variables being compared:

1. Mean of X
2. Mean of Y
3. Angle of rotation
4. Standard deviation along the transformed X axis
5. Standard deviation along the transformed Y axis
6. Area of the ellipse

Figure A.2:


1996 Burglaries and Robberies in Baltimore County, MD
Comparison of Standard Deviational Ellipses (SDE)
MC of Burglaries
MC of Robberies
SDE of Robberies
SDE of Burglaries
Interstate Highways
Baltimore County
City of Baltimore
Baltimore County
SDE of burglaries
SDE of robberies
City of Baltimore
0
2.5
5
10 Miles

Differences in the mean centers

Comparisons between the two mean centers can be tested with equations A.9 through A.12 if the variance test of equations A.7 and A.8 show equality or equation A.13 through A.20 if the variances are unequal.

Differences in the angle of rotation

Unfortunately, to our knowledge, there is not a formal test for the difference in the angle of rotation. Until this test is developed, we have to rely on subjective judgment.

Differences in the standard deviations along the transformed axes

The differences in the standard deviations along the transformed axes (X and Y) can be tested with an equality of variance test (Kanji, 1993, 37),

$$F_{S_X} = \frac{S_{XA}^2}{S_{XB}^2} \tag{A.27}$$

$$F_{S_Y} = \frac{S_{YA}^2}{S_{YB}^2} \tag{A.28}$$

with N_A*-1* and N_B*-1* degrees of freedom for groups A and B respectively. This test is usually conducted with the larger of the variances in the numerator. The example above for comparing the mean centers of Baltimore County burglaries and robberies illustrated the use of this test.

Differences in the areas of the two ellipses

Since an area is a variance, the differences in the areas of the two ellipses can be compared with an equality of variance test (Kanji, 1993, 37),

$$F = \frac{Area_A}{Area_B} \tag{A.29}$$

with N_A*-1* and N_B*-1* degrees of freedom for groups A and B respectively. This test is conducted with the larger of the variances in the numerator.

Significance levels

The testing of each of these parameters for the difference between two ellipses is even more complicated than the difference between two mean centers since there are up to six parameters which must be tested (differences in mean X, mean Y, angle of rotation, standard deviation along transformed X axis, standard deviation along transformed Y axis, and area of ellipse). However, as with differences in mean center of two groups, there are two different interpretations of differences.

Comparison I: That the two ellipses differ on ANY of the parameters (A.30)

Comparison II: That the two ellipses differ on ALL parameters (A.31)

In the first case, the critical probability level, α, must be divided by the number of parameters being tested, α/p. In theory, this could involve up to six tests, though in practice some of these may not be tested (e.g., the angle of rotation). For example, if five of the parameters are being estimated, then the critical probability level at $\alpha \leq .05$ is actually $\alpha \leq .01$ (.05/5).

In the second case, the critical probability level, α, is multiplied by the number of parameters being tested, α*p, since *all* tests must be significant for the two ellipses to be considered as different. For example, if five of the parameters are being estimated, then the critical probability level, say, at $\alpha \leq .05$ is actually $\alpha \leq .25$ (.05*5).

Differences in the Mean Direction Between Two Groups

Statistical tests of different angular distributions can be made with the directional mean and variance statistics. To test the difference in the angle of rotation between two groups, a Watson-Williams test can be used (Kanji, 1993; 153-54). The steps in the test are as follows:

1. All angles, θ_i, are converted into radians

$$\text{Radian}_i = \text{Angle}_i * \pi/180 \quad \text{(A.32)}$$

2. For each sample separately, *A* and *B*, the following measures are calculated

$$C_A = \sum_{A=1}^{N_A} cos\theta_A\,,\;\; S_A = \sum_{A=1}^{N_A} sin\theta_A \quad \text{(A.33)}$$

$$C_B = \sum_{B=1}^{N_B} cos\theta_B \, , \; S_B = \sum_{B=1}^{N_B} sin\theta_B \qquad \text{(A.34)}$$

where θ_j and θ_k are the individual angles for the respective groups, *A* and *A*.

3. Calculate the resultant lengths of each group

$$R_A = \sqrt{(C_A^2 + S_A^2)} \qquad \text{(A.35)}$$

$$R_B = \sqrt{(C_B^2 + S_B^2)} \qquad \text{(A.36)}$$

4. Resultant lengths for the combined sample are calculated as well as the length of the resultant vector.

$$C = C_A + C_B \qquad \text{(A.37)}$$

$$S = S_A + S_B \qquad \text{(A.38)}$$

$$R = \sqrt{(C^2 + S^2)} \qquad \text{(A.39)}$$

$$N = N_A + N_B \qquad \text{(A.40)}$$

$$R^* = \frac{R_A + R_B}{N} \qquad \text{(A.41)}$$

5. An F-test of the two angular means is calculated with

$$F = g(N-2)\frac{R_A + R_B - R^*}{N - (R_A + R_B)} \qquad \text{(A.42)}$$

where

$$g = 1 - \frac{3}{8k} \qquad \text{(A.43)}$$

with *k* being identified from a maximum likelihood Von Mises distribution by referencing R^* with 1 and N-2 degrees of freedom (Gaile & Burt, 1980; Mardia, 1972). Some of the reference *k*'s are given in Table A.1 below (from Kanji, 1993, table 38; Mardia, 1972).

6. Reject the null hypothesis of no angular difference if the calculated F is greater than the critical value $F_{1, N-2}$.

Example 2: Angular comparisons between two groups

A second example is that of sets of angular measurements from two different groups, A and B. Table A.2 provides the data for the two sets. The angular mean for Group A is 144.83^0 with a directional variance of 0.35 while the angular mean for Group B is 258.95^0 with a directional variance of 0.47. The higher directional variance for Group B suggests that there is more angular variability than for Group A.

Table A.1:

Maximum Likelihood Estimates for Given R^* in the Von Mises Case

(Kanji, 1993, table 38; Mardia, 1972)

R^*	k
0.00	0.00000
0.05	0.10013
0.10	0.20101
0.15	0.30344
0.20	0.40828
0.25	0.51649
0.30	0.62922
0.35	0.74783
0.40	0.87408
0.45	1.01022
0.50	1.15932
0.55	1.32570
0.60	1.51574
0.65	1.73945
0.70	2.01363
0.75	2.36930
0.80	2.87129
0.85	3.68041
0.90	5.3047
0.95	10.2716
1.00	infinity

Using the Watson-Wheeler test, we compare these two distributions.

1. All angles are converted into radians (equation A.32).

2. The cosines and sines of each angle are taken and are summed within groups (equations A.33 and A.34).

$C_A =$ -3.1981 $S_A = 2.2533$
$C_B =$ -.8078 $S_B =$ -4.1381

Table A.2:
Comparison of Two Groups for Angular Measurements
Angle of Deviation From Due North

Group A		**Group B**	
Measured Incident	**Angle**	**Measured Incident**	**Angle**
1	160	1	196
2	184	2	212
3	240	3	297
4	100	4	280
5	95	5	235
6	120	6	353
7	190		
8	340		

3. The resultants are calculated (equations A.35 and A.36).

$R_A = 3.9121$
$R_B = 4.2162$

4. Combined sample characteristics are defined (equations A.37 through A.41).

   C = -4.0059
   S = -1.8848
   R = 4.4271
   N = 14
   $R^* = 0.5806$

5. Once the parameter, *k*, is obtained (approximated from Table A.1 or obtained from Mardia, 1972 or Kanji, 1993), g is calculated, and an F-test is constructed (equations A.42 and A.43).

   k = 1.44
   g = 0.7396
   F = 5.59

6. The critical F for 1 and 12 degrees of freedom is 4.75 for $p \leq .05$ and F=9.33 for $p \leq .01$. Since F=5.59 is between these two critical F values, the test is significant at the $p \leq .05$ level, but not at the $p \leq .01$ level. Nevertheless, we reject the null hypothesis of no angular differences between the two groups. Group A has a different angular distribution than Group B.

References

Anselin, L. (1995). Local indicators of spatial association - LISA. *Geographical Analysis*. 27, No. 2 (April), 93-115.

Gaile, G. L. & Burt, J. E. (1980). *Directional Statistics*. Concepts and Techniques in Modern Geography No. 25. Institute of British Geographers, Norwich, England: Geo Books.

Kanji, G. K. (1993). *100 Statistical Tests*. Sage Publications: Thousand Oaks, CA.

Mardia, K.V. (1972). *Statistics of Directional Data*. Academic Press: New York.

Systat, Inc. (2008). *Systat 13: Statistics I*. SPSS, Inc.: Chicago.

Appendix B:

Ordinary Least Squares and Poisson Regression Models

by
Luc Anselin
Arizona State University
Tempe, AZ

This note provides a brief description of the statistical background, estimators and model characteristics for a regression specification, estimated by means of both Ordinary Least Squares (OLS) and Poisson regression.

Ordinary Least Squares Regression

With an assumption of normality for the regression error term, OLS also corresponds to Maximum Likelihood (ML) estimation. The note contains the statistical model and all expressions that are needed to carry out estimation and essential model diagnostics. Both concise matrix notation as well as more extensive full summation notation are employed, to provide a direct link to "loop" structures in the software code, except when full summation is too unwieldy (e.g., for matrix inverse). Some references are provided for general methodological descriptions.

Statistical Issues

The classical multivariate linear regression model stipulates a linear relationship between a *dependent* variable (also called a response variable) and a set of *explanatory* variables (also called independent variables, or covariates). The relationship is stochastic, in the sense that the model is not exact, but subject to random variation, as expressed in an *error* term (also called disturbance term).

Formally, for each observation i, the value of the dependent variable, Y_i, is related to a sum of K explanatory variables, X_{ih}, with h=1,...,K, each multiplied with a regression *coefficient*, β_h, and the random error term, ε_i:

$$Y_i = \sum_{h=1}^{K} X_{ih}\beta_h + \varepsilon_i \qquad \text{(B.1)}$$

Typically, the first explanatory variable is set equal to one, and referred to as the *constant term*. Its coefficient is referred to as the *intercept*, the other coefficients are

slopes. Using a constant term amounts to extracting a mean effect and is equivalent to using all variables as deviations from their mean. In practice, it is highly recommended to *always* include a constant term.

In matrix notation, which summarizes all observations, *i=1,...,N*, into a single compact expression, an *N by 1* vector of values for the dependent variable, y is related to an *N by K* matrix of values for the explanatory variables, *X*, a *K by 1* vector of regression coefficients, β, and an *N by 1* vector of random error terms, ε:

$$Y = X\beta + \epsilon \tag{B.2}$$

This model stipulates that on average, when values are observed for the explanatory variables, *X*, the value for the dependent variable equals $X\beta$, or:

$$E(Y|X) = X\beta \tag{B.3}$$

where E[|] is the conditional expectation operator. This is referred to as a specification for the conditional mean, conditional because *X* must be observed. It is a theoretical model, built on many assumptions. In practice, one does not know the coefficient vector, β, nor is the error term observed.

Estimation boils down to finding a "good" value for the β, with known statistical properties. The statistical properties depend on what is assumed in terms of the characteristics of the distribution of the unknown (and never observed) error term. To obtain a Maximum Likelihood estimator, the complete distribution must be specified, typically as a normal distribution, with mean zero and variance, σ^2. The mean is set to zero to avoid systematic under- or over-prediction. The variance is an unknown characteristic of the distribution that must be estimated together with the coefficients, β. The estimate for β (Greek letter) will be referred to as *b* (Latin letter with b_h as the estimate for the individual coefficient, β_h).

The *estimator* is the procedure followed to obtain an estimate, such as OLS, for b_{OLS}, or ML, for b_{ML}. The *residual* of the regression is the difference between the observed value and the *predicted value*, typically referred to as *e*. For each observation,

$$e_i = Y_i - \sum_{h=1}^{K} X_{ih}\beta_h \tag{B.4}$$

or, in matrix notation, with $\hat{Y} = Xb$ as short hand for the vector of predicted values,

$$e = Y - \hat{Y} \tag{B.5}$$

Note that the residual is *not* the same as the error term, but only serves as an estimate for the error. What is of interest is not so much the individual residuals, but the properties of the (unknown) error distribution. Within the constraints of the model assumptions, some of the characteristics of the error distribution can be estimated from the residuals, such as the error variance, σ^2, whose estimate is referred to as s^2.

Because the model has a random component, the observed *y* are random as well, and any "statistic" computed using these observed data will be random too. Therefore, the estimates *b* will have a distribution, intimately linked to the assumed distribution for the error term. When the error is taken to be normally distributed, the regression coefficient will also follow a normal distribution. Statistical inference (significance tests) can be carried out once the characteristics (parameters) of that distribution have been obtained (they are never known, but must be estimated from the data as well). An important result is that OLS is *unbiased*. In other words, the mean of the distribution of the estimate *b* is β, the true, but unknown, coefficient, such that "on average," the estimation is on target. Also, the variance of the distribution of *b* is directly related to the variance of the error term (and the values for the *X*). It can be computed by replacing σ^2 by its estimate, s^2.

An extensive discussion of the linear regression model can be found in most texts on linear modeling, multivariate statistics, or econometrics, for example, Rao (1973), Greene (2000), or Wooldridge (2002).

Ordinary Least Squares Estimator

In its most basic form, OLS is simply a fitting mechanism, based on minimizing the sum of squared residuals or residual sum of squares (RSS). Formally, b_{OLS} is the vector of parameter values that minimizes

$$RSS = \sum_{i=1}^{N} e_i^2 = \sum_{i=1}^{N} (Y_i - \sum_{h=1}^{K} X_{ij} b_h)^2 \qquad \text{(B.6)}$$

or, in matrix notation,

$$RSS = e'e = (Y - Xb)'(Y - Xb) \qquad \text{(B.7)}$$

The solution to this minimization problem is given by the so-called *normal equations*, a system of *K* equations of the form:

$$\sum_{i=1}^{N}(Y_i - \sum_{h=1}^{K} X_{ih} b_h) X_{ih} = 0 \quad \text{(B.8)}$$

for *h=1 to K*, or, in matrix notation,

$$X'(Y - Xb) = 0 \quad \text{(B.9)}$$

$$X'Xb = X'Y \quad \text{(B.10)}$$

The solution to this system of equations yields the familiar matrix expression for b_{OLS}:

$$b_{OLS} = (X'X)^{-1}X'Y \quad \text{(B.11)}$$

An estimate for the error variance follows as

$$s^2_{OLS} = \frac{\sum_{i=1}^{N}(Y_i - \sum_{h=1}^{K} X_{ij} b_{OLS,h})^2}{N-K} \quad \text{(B.12)}$$

or, in matrix notation,

$$s^2_{OLS} = \frac{e'e}{N-K} \quad \text{(B.13)}$$

It can be shown that when the *X* are *exogenous*[1] only the assumption that E[ε]=0 is needed to show that the OLS estimator is *unbiased*. With the additional assumption of a fixed error variance, s^2, OLS is also most *efficient*, in the sense of having the smallest variance among all other linear and unbiased estimators. This is referred to as the BLUE (Best Linear Unbiased Estimator) property of OLS. Note, that in order to obtain these properties, no additional assumptions need to be made about the distribution of the error term. However, to carry out statistical inference, such as significance tests, this is insufficient, and further characteristics of the error distribution need to be specified (such as assuming a normal distribution) or asymptotic assumptions need to be invoked in the form of laws of large numbers (typically yielding a normal distribution).

[1] In practice, this means that each explanatory variable must be uncorrelated with the error term. The easiest way to ensure this is to assume that the *X* are fixed. But even when they are not, this property holds, as long as the randomness in *X* and ε are not related. In other words, knowing something about the value of an explanatory variable should *not* provide any information about the error term. Formally, this means that *X* and ε must be orthogonal, or $E[X'\varepsilon]=0$. Failure of this assumption will lead to so-called simultaneous equation bias.

Maximum Likelihood Estimator

When the error terms are assumed to be independently distributed as normal random variables, OLS turns out to be equivalent to ML.

Maximum Likelihood estimation proceeds as follows. First, consider the density for a single error term:

$$\varepsilon \sim N(0, \sigma^2) \tag{B.14}$$

or

$$f([\varepsilon]_i|s^2) = \frac{1}{\sqrt{2\Pi\sigma^2}} e^{-(1/2)(\varepsilon_i^2/\sigma^2)} \tag{B.15}$$

A subtle, but important, point is that the error itself is not observed, but only the "data" (y and X) are. We move from a model for the error, expressed in unobservables, to a model that contains observables and the regression parameter by means of a standard "transformation of random variables" procedure. Since Y_i is a linear function of ε it will also be normally distributed. Its density is obtained as the product of the density of ε and the "Jacobian" of the transformation, using $\varepsilon_i = y_i - x_i\beta$ (with x_i as the i-th row in the X matrix). As it turns out, the Jacobian is one, so that

$$f([y_i|\beta]_i|s^2) = \frac{1}{\sqrt{2\Pi\sigma^2}} e^{-(1/2)((y_i - x_i\beta)^2/\sigma^2)} \tag{B.16}$$

The likelihood function is the joint density of all the observations, given a value for the parameters β and σ^2. Since independence is assumed, this is simply the product of the individual densities from equation B.16. The log-likelihood is then the log of this product, or the sum of the logs of the individual densities. The contribution to the log likelihood of each observation follows from equation B.16:

$$Log(f(Y_i|\beta, \sigma^2) = L_i = -0.5\log(2\pi) - 0.5\log(\sigma^2) - 0.5(\frac{Y_i - X_i\beta)^2}{\sigma^2}) \tag{B.17}$$

The full log-likelihood follows as:

$$L = \sum_{i=1}^{N} L_i = -\frac{N}{2}\log(2\pi) - \frac{N}{2}\log(\sigma^2) - \frac{\sigma^2}{2}\sum_{i=1}^{N}(Y_i - X_i\beta)^2 \tag{B.18}$$

or, in matrix notation,

$$L = -\frac{N}{2}\log(2\pi) - \frac{N}{2}\log(\sigma^2) - \frac{\sigma^2}{2}(Y - X\beta)'(Y - X\beta) \quad \text{(B.19)}$$

A Maximum Likelihood estimator for the parameters in the model finds the values for β and σ^2 that yield the highest value for equation B.19. It turns out that minimizing the residual sum of squares (or, least squares), the last term in equations B.18 and B.19, is equivalent to maximizing the log-likelihood. More formally, the solution to the maximization problem is found from the first-order conditions (setting the first partial derivatives of the log-likelihood to zero), which yield the OLS estimator for *b* and

$$s_{ML}^2 = \sum_{i=1}^{N} \frac{e_i^2}{N} \quad \text{(B.20)}$$

or, in matrix notation,

$$s_{ML}^2 = \frac{e'e}{N} \quad \text{(B.21)}$$

Inference

With estimates for the parameters in hand, the missing piece is a measure for the precision of these estimates, which can then be used in significance tests, such as t-tests and F-tests. The estimated variance-covariance matrix for the regression coefficients is

$$Var(b) = s^2(X'X)^{-1} \quad \text{(B.22)}$$

where s^2 is either s_{OLS}^2 or s_{ML}^2. The diagonal elements of this matrix are the variance terms, and their square root the standard error. Note that the estimated variance using s_{ML}^2 will always be smaller than that based on the use of s_{OLS}^2. This may be spurious, since the ML estimates are based on asymptotic considerations (with a "conceptual" sample size approaching infinity), whereas the OLS estimates use a "degrees of freedom" (*N*-*K*) correction. In large samples, the distinction between OLS and ML disappears (for very large *N* as *N* and *N*-*K* will be very close).

Typically, interest focuses on whether a particular population coefficient (the unknown b_h) is different from zero, or, in other words, whether the matching variable contributes to the regression. Formally, this is a test on the null hypothesis that $b_h = 0$. This leads to a t test statistic as the ratio of the estimate over its standard error (the square root of the *h*,*h* element in the variance-covariance matrix), or

$$t = \frac{b_h}{\sqrt{s^2 (X'X)^{-1}_{hh}}} \tag{B.23}$$

This test statistic follows a Student t distribution with *N-K* degrees of freedom. If, according to this reference distribution, the probability that a value equal to or larger than the t-value (for a one-sided test) occurs is very small, the null hypothesis will be rejected and the coefficient deemed "significant."[2]

Note that when s_{ML}^2 is used as the estimate for s^2, the t-test is referred to as an "asymptotic" t-test. In practice, this is a standard normal variate. Hence, instead of comparing the t test statistic to a Student t distribution, its probability should be evaluated from the standard normal density.

A second important null hypothesis pertains to all the coefficients taken together (other than the intercept). This is a test on the significance of the regression as a whole, or a test on the null hypothesis that, jointly, $b_h = 0$, for h=2,...,K (note that there are K-1 hypotheses). The F test statistic for this test is constructed by comparing the residual sum of squares (RSS) in the regression to that obtained without a model. The latter is referred to as the "constrained" (i.e., with all the β except the constant term set to zero) residual sum of squares (RSS_C). It is computed as the sum of squares of the y_i in deviation from the mean, or

$$RSS_C = \sum_{i=1}^{N} (Y_i - \bar{Y})^2 \tag{B.24}$$

where $\bar{Y} = Y_i/N$. The F statistic then follows as:

$$F = \frac{\frac{(RSS_C - RSS)}{(K-1)}}{\frac{RSS}{(N-K)}} \tag{B.25}$$

It is distributed as an F-variate with K-1,N-K degrees of freedom.

Model Fit

The most common measure of fit of the regression is the R^2, which is closely related to the F-test. The R^2 departs from a decomposition of the total sum of squares, or the RSS_C from equation B**Error! Reference source not found.**, into the "explained" sum of

[2] Any notion of significance is always with respect to a given p-value, or Type I error. The Type I error is the chance of making a wrong decision, i.e., of rejecting the null hypothesis when in fact it is true.

squares (the sum of squares of predicted values, in deviations from the mean), and the residual sum of squares, RSS.

The R^2 is a measure of how much of this decomposition is due to the "model." It is easily computed as:[3]

$$R^2=1-RSS/RSS_C \tag{B.26}$$

In general, the model with the highest R^2 is considered best. However, this may be misleading since it is always possible to increase the R^2 by adding another explanatory variable, irrespective of whether this variable contributes "significantly." The adjusted R^2 (R_a^2) provides a better guide that compensates for "over-fitting" the data by correcting for the number of variables included in the model. It is computed by rescaling the numerator and denominator in equation B.26, as

$$R_a^2 = 1 - \frac{\frac{RSS}{(N-K)}}{\frac{RSS_C}{(N-1)}} \tag{B.27}$$

For very large data sets, this rescaling will have negligible effect and the R^2 and R_a^2 will be virtually the same.

When OLS is viewed as a ML estimator, an alternative measure of fit is the value of the maximized log-likelihood. This is obtained by substituting the estimates b_{ML} and s_{ML}^2 into expression B.18 or B.19. With $e = y - Xb_{ML}$ as the residual vector and $s_{ML}^2 = e'e/N$, the log-likelihood can be written in a simpler form:

$$L = -\frac{N}{2}\log(2\pi) - \frac{N}{2}\log\left(\frac{e'e}{N}\right) - 0.5\frac{e'e}{N}(e'e) \tag{B.28}$$

$$L = -\frac{N}{2}\log(2\pi) - \frac{N}{2} - \frac{N}{2}\log\left(\frac{e'e}{N}\right) \tag{B.29}$$

Note that the only term that changes with the model fit is the last one, the logarithm of the average residual sum of squares. Therefore, the constant part is not always reported. To retain comparability with other models (e.g., spatial regression models), it is important to be consistent in this reporting. The model with the *highest* maximized log-likelihood is considered to be best, even though the likelihood, as such, is technically not a measure of fit.

[3] When the regression specification does not contain a constant term, the value obtained for the R^2 using equation 26 will be incorrect. This is because the constant term forces the residuals to have mean zero. Without a constant term, the RSS must be computed in the same waysameway as *in equation **Error! Reference source not found.** by subtracting the average residual $\hat{e} = \Sigma e_i/N$.

Similar to the R_a^2, there exist several corrections of the maximized log-likelihood to take into account potential over-fitting. The better-known measures are the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC)/Schwartz Criterion (SC), familiar in the literature on Bayesian statistics. They are easily constructed from the maximized log-likelihood. They are, respectively:

$$AIC = -2L + 2K \tag{B.30}$$

$$BIC/SC = -2L + Klog(N) \tag{B.31}$$

The model with the *lowest* information criterion value is considered to be best.

Poisson Regression

Next, the Poisson regression model is examined.

Likelihood Function

In the Poisson regression model, the dependent variable for observation *i* (with *i*=1,...,*N*), Y_i is modeled as a Poisson random variate with a mean λ_i that is specified as a function of a *K* by 1 (column) vector of explanatory variables x_i, and a matching vector of parameters β. The probability of observing y_i is expressed as:

$$Prob(Y_i) = \frac{e^{-\lambda}\lambda_i Y_i}{Y_i!} \tag{B.32}$$

The conditional mean of y_i, given observations on x_i is specified as an exponential function of *x*:

$$E(Y_i|X_i) = l_i = e^{X_i'b} \tag{B.33}$$

where x_i' is a row vector. Equivalently, this is sometimes referred to as a *loglinear* model, since

$$\ln(l_i) = X_i'b \tag{B.34}$$

Note that the mean in B.33 is nonlinear, which means that the effect of a change in X_i will depend not only on β (as in the classical linear regression), but also on the value of X_i. Also, in the Poisson model, the mean equals the variance (equidispersion) so that there is no need to separately estimate the latter.

There is a fundamental difference between a classical linear regression model and the specification for the conditional mean in the Poisson regression model in that the latter does not contain a random error term (in its "pure" form). Consequently, unlike the approach taken for the linear regression, the log-likelihood is not derived from the joint density of the random errors, but from the distribution for dependent variable itself, using B.32. Also, there is no need to estimate a residual variance, as in the classical regression model.

Assuming independence among the count variables (e.g., *excluding* spatial correlation), the log-likelihood for the Poisson regression model follows as:

$$L = \sum_{i=1}^{N} Y_i X_i^b - e^{X_i' b} - lnY_i! \qquad \text{(B.35)}$$

Note that the third term is a constant and does not change with the parameter values. Some programs may not include this term in what is reported as the log-likelihood. Also, it is not needed in a Likelihood Ratio test, since it will cancel out.

The first order conditions, $\partial L/\partial \beta = 0$, yield a system of K equations (one for each β) of the form:

$$\sum_{i=1}^{N} (y_i - e^{x_i'b}) x_i = 0 \qquad \text{(B.36)}$$

Note how this takes the usual form of an orthogonality condition between the "residuals" ($y_i - e^{x_i'b}$) and the explanatory variables, X_i. This also has the side effect that when X contains a constant term, the sum of the predicted values, $e^{x_i'b}$ equals the sum of the observed counts.[4] The system B.36 is nonlinear in β and does not have an analytical solution. It is typically solved using the Newton-Raphson method (see below).

Once the estimates of β are obtained, they can be substituted into the log-likelihood (equation B.36) to compute the value of the maximum log-likelihood. This can then be inserted in the AIC and BIC information criteria in the usual way.

[4] A different way of stating this property is to note that the sum of the residuals equals zero. As for the classical linear regression model, this is not guaranteed without a constant term in the regression.

Predicted Values and Residuals

The predicted value, $\widehat{Y_i}$, is the conditional mean or the average number of events, given the X_i. This is also denoted as λ_i and is typically not an integer number whereas the observed value Y_i is a count. The use of the exponential function guarantees that the predicted value is non-negative. Specifically:

$$\widehat{\lambda_i} = e^{X_i'\hat{b}} \tag{B.37}$$

The "residuals" are simply the difference between observed and predicted:

$$e_i = Y_i - e^{e^{X_i'\hat{b}}} = Y_i - \widehat{\lambda_i} \tag{B.38}$$

Note that, unlike the case for the classical regression model, these residuals are not needed to compute estimates for error variance (since there is no error term in the model).

Estimation Steps

The well known Newton-Raphson procedure proceeds iteratively. Starting from a set of estimates $\hat{\beta}_t$ the next value is obtained as:

$$\hat{\beta}_{t+1}=\hat{\beta}_t - \hat{H}_t^{-1}\hat{g}_t \tag{B.39}$$

where $\hat{g}_t$ is the first partial derivative of the log-likelihood, evaluated at $\hat{\beta}_t$ and $\hat{H}_t$ is the Hessian, or second partial derivative, also evaluated at $\hat{\beta}_t$.

In the Poisson regression model,

$$g= \sum_{i=1}^{N} x_i(y_i-\hat{\lambda}_i) \tag{B.40}$$

$$H= - \sum_{i=1}^{N} \hat{\lambda}_i x_i x_i' \tag{B.41}$$

In practice, one can proceed along the following lines.

1. **Set initial values for parameters**, say b_0[h], for h=1,...,K. One can set $b_0[1] = \bar{y}$, the overall average count as the constant term, and the other b_0[h]=0, for h=2,...,K.

2. **Compute predicted values** for each i, the value of $\hat{\lambda}_i = e^{x_i' b0}$.

3. **Compute gradient**, g, using the starting values. Note that $g[h]$ is a K by 1 vector. Each element of this vector is the difference between:

$$O_i = \sum_{i=1}^{N} X_{ih} Y_i \quad \text{(B.42)}$$

$$P_i = \sum_{i=1}^{N} X_{ih} \lambda_i \quad \text{(B.43)}$$

$$g_i = O_i - P_i \quad \text{(B.44)}$$

Note that B.42 does not contain any unknown parameters and needs only to be computed once (provided there is sufficient storage). As the Newton-Raphson iterations proceed, the values of g will become very small.

4. **Compute the Hessian**, H, using the starting values. H is a K by K matrix (B.41) that needs to be inverted at each iteration in B.39. It is *not* the $X'X$ of the classical model, but rather more like $X' \Sigma X$, where Σ is a diagonal matrix. One way to implement this is to multiply each row of the X matrix by $\sqrt{\hat{\lambda}_i}$, e.g., $xs[i][h]=x[i][h]$*sqrt($\hat{\lambda}[i]$), where xs is the new matrix (X^*), i is the observation (row) and h the column of X. The Hessian then becomes the cross product of the new matrices, or, $H=X^{*'} X^*$. This needs to be done in each iteration. There is no need to take a negative since the negative in B.41 and in B.39 cancel.

5. **Update the estimate** for the $b[h]$, say b_1[h] is obtained using the updating equation B.39 except that the product $H^{-1}g$ is added to the initial value. In general, for iteration t, the new estimates are obtained as b_{t+1}. After checking for convergence, the old b_t is set to b_{t+1} and inserted in the computation of the predicted values, in step 2 above.

6. **Convergence**. Stop the iterations when the difference between b_{t+1} and b_t becomes below some tolerance level. A commonly used criterion is the norm of the difference vector or $\Sigma_h (b_{t+1}[h] - b_t[h])^2$. When the norm is below a preset level, stop the iterations and report the last b_t as the result.

The reason for not using b_{t+1} is that the latter would require an extra computation of the Hessian needed for inference.

Inference

The asymptotic variance matrix is the inverse Hessian obtained at the last iteration (i.e., using b_t). The variance of the estimates are the diagonal elements, the standard errors their square roots. The asymptotic t-test is constructed in the usual way, as the ratio of the estimate over its standard error. The only difference with the classic linear regression case is that the p-values must be looked up in a standard normal distribution, not a Student t distribution.

Likelihood Ratio Test

A simple test on the overall fit of the model, as an analogue to the F-test in the classical regression model is a Likelihood Ratio test on the "slopes". The model with only the intercept is nothing but the mean of the counts, or

$$\lambda_I = \bar{Y}\forall \tag{B.45}$$

with $\bar{Y} = \sum_{i=1}^{N} Y_i/N$.

The corresponding log-likelihood is:

$$L_R = -N\bar{Y} + \ln(\bar{Y})\,(\sum_{i=1}^{N} Y_i) - \sum_{i=1}^{N} \ln(Y_i)\,! \tag{B.46}$$

where the *R* stands for the "restricted" model, as opposed to the "unrestricted" model with *K-1* slope parameters. The last term in B.46 can be dropped, as long as it is also dropped in the calculation of the maximized likelihood (B.35) for the unrestricted model (L_U), using $l_i = e^{x_i' b_t}$. The Likelihood Ratio test is then:

$$LR = 2(L_U - L_R), \tag{B.47}$$

and follows a χ^2 distribution with *K*-1 degrees of freedom.

References

Gentle, J. E. (1998). *Numerical Linear Algebra for Applications in Statistics*. Springer-Verlag, New York, NY.

Gentleman, W. M. (1974). Algorithm AS 75: Basic procedures for large, sparse or weighted linear least problems. *Applied Statistics*, 23: 448–454.

Greene, W. H. (2000). *Econometric Analysis, 4th Ed.* Prentice Hall, Upper Saddle River, NJ.

Miller, A. J. (1992). Algorithm AS 274: Least squares routines to supplement those of Gentleman. *Applied Statistics*, 41: 458–478.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). *Numerical Recipes in C. The Art of Computing (Second Edition)*. Cambridge University Press, Cambridge, UK.

Rao, C. R. (1973). *Linear Statistical Inference and its Applications*. Wiley, New York, 2nd edition.

Wooldridge, J. M. (2002). *Econometric Analysis of Cross Section and Panel Data*. MIT Press, Cambridge, MA.

Appendix C:

Negative Binomial Regression Models and Estimation Methods

Dominique Lord
Zachry Dept. of
Civil Engineering
Texas A & M University
College Station, TX

Byung-Jung Park
Korea Transport Institute
Goyang, South Korea

This appendix presents the characteristics of Negative Binomial regression models and discusses their estimating methods.

Probability Density and Likelihood Functions

The properties of the negative binomial models with and without spatial intersection are described in the next two sections.

Poisson-Gamma Model

The Poisson-Gamma model has properties that are very similar to the Poisson model discussed in Appendix B, in which the dependent variable y_i is modeled as a Poisson variable with a mean λ_i where the model error is assumed to follow a Gamma distribution. As its name implies, the Poisson-Gamma is a mixture of two distributions and was first derived by Greenwood and Yule (1920). This mixture distribution was developed to account for over-dispersion that is commonly observed in discrete or count data (Lord et al., 2005). It became very popular because the conjugate distribution (same family of functions) has a closed form and leads to the negative binomial distribution. As discussed by Cook (2009), "the name of this distribution comes from applying the binomial theorem with a negative exponent." There are two major parameterizations that have been proposed and they are known as the NB1 and NB2, the latter one being the most commonly known and utilized. NB2 is therefore described first. Other parameterizations exist, but are not discussed here (see Maher and Summersgill, 1996; Hilbe, 2007).

NB2 Model

Suppose that we have a series of random counts that follows the Poisson distribution:

$$g\left(y_i;\lambda_i\right)=\frac{e^{-\lambda_i}\lambda_i}{y_i!} \tag{C.1}$$

where y_i is the observed number of counts for $i=1,2,\ldots,\text{n}$; and λ_i is the mean of the Poisson distribution. If the Poisson mean is assumed to have a random intercept term and this term enters the conditional mean function in a multiplicative manner, we get the following relationship (Cameron and Trivedi, 1998):

$$\begin{aligned}
\lambda_i &= \exp\left(\beta_0+\sum_{j=1}^{K}x_{ij}'\beta_j+\varepsilon_i\right)\\
\lambda_i &= e^{\sum_{j=1}^{K}x_{ij}'\beta_j}e^{(\beta_0+\varepsilon_i)}\\
\lambda_i &= e^{\left(\beta_0+\sum_{j=1}^{K}x_{ij}'\beta_j\right)}e^{\varepsilon_i}\\
\lambda_i &= \mu_i\nu_i
\end{aligned} \tag{C.2}$$

where $\exp\left(\beta_0+\varepsilon_i\right)$ is defined as a random intercept; $\mu_i=\exp\left(\beta_0+\sum_{j=1}^{K}x_{ij}'\beta_j\right)$ is the log-link between the Poisson mean and the covariates or independent variables xs ; and the βs are the regression coefficients. As discussed in Appendix B, the relationship can also be formulated using vectors, such that $\mu_i=\exp(\mathbf{x}_i'\boldsymbol{\beta})$.

The marginal distribution of y_i can be obtained by integrating the error term, ν_i ,

$$\begin{aligned}
f\left(y_i;\mu_i\right) &= \int_o^{\infty} g\left(y_i;\mu_i,\nu_i\right)h\left(\nu_i\right)d\nu_i\\
f\left(y_i;\mu_i\right) &= E_\nu\left[g\left(y_i;\mu_i,\nu_i\right)\right]
\end{aligned} \tag{C.3}$$

where $h\left(\nu_i\right)$ is a mixing distribution. In the case of the Poisson-Gamma mixture, $g\left(y_i;\mu_i,\nu_i\right)$ is the Poisson distribution and $h\left(\nu_i\right)$ is the Gamma distribution. This distribution has a closed form and leads to the NB distribution.

Assume that the variable ν_i follows a two-parameter Gamma distribution:

$$k(\nu_i;\psi,\delta)=\frac{\delta^{\psi}}{\Gamma(\psi)}\nu_i^{\psi-1}e^{-\nu_i\delta},\qquad \psi>0,\ \delta>0,\ \nu_i>0 \tag{C.4}$$

where, $E[\nu_i]=\psi/\delta$ and $VAR[\nu_i]=\psi/\delta^2$. Setting $\psi=\delta$ gives us the one-parameter Gamma where $E[\nu_i]=1$ and $VAR[\nu_i]=1/\psi$. We can transform the Gamma distribution as a function of the Poisson mean, which gives the following *probability density function* (PDF; Cameron and Trivedi, 1998):

$$k(\lambda_i;\psi,\mu_i)=\frac{(\psi/\mu_i)^{\psi}}{\Gamma(\psi)}\lambda_i^{\psi-1}e^{-\frac{\lambda_i}{\mu_i}\delta} \tag{C.5}$$

Combining equations C-1 and C-5 with equation C-3 yields the marginal distribution of y_i:

$$f(y_i;\mu_i,\psi)=\int_o^{\infty}\frac{\exp(-\lambda_i)\lambda_i^{y_i}}{y_i!}\frac{(\psi/\mu)^{\psi}}{\Gamma(\psi)}\lambda_i^{\psi-1}e^{-\frac{\lambda_i}{\mu_i}\delta}d\lambda_i \tag{C.6}$$

Using the properties of the Gamma function, it can be shown that equation C-6 can be expressed as:

$$\begin{aligned}
f(y_i;\mu_i,\psi)&=\frac{(\psi/\mu_i)^{\psi}}{\Gamma(\psi)\Gamma(y_i+1)}\int_o^{\infty}\exp\left(-\lambda_i\left(1+\frac{\psi}{\mu_i}\right)\right)\lambda_i^{y+\psi-1}d\lambda_i\\
f(y_i;\mu_i,\psi)&=\frac{(\psi/\mu_i)^{\psi}\left(1+\frac{\psi}{\mu_i}\right)^{-(\psi+y_i)}\Gamma(\psi+y_i)}{\Gamma(\psi)\Gamma(y_i+1)}\\
f(y_i;\mu_i,\psi)&=\frac{\Gamma(y_i+\psi)}{\Gamma(y_i+1)\Gamma(\psi)}\left(\frac{\psi}{\mu_i+\psi}\right)^{\psi}\left(\frac{\mu_i}{\mu_i+\psi}\right)^{y_i}
\end{aligned} \tag{C-7}$$

The PDF of the NB2 model is therefore the last part of Equation C-7:

$$f(y_i;\mu_i,\psi)=\frac{\Gamma(y_i+\psi)}{\Gamma(y_i+1)\Gamma(\psi)}\left(\frac{\psi}{\mu_i+\psi}\right)^{\psi}\left(\frac{\mu_i}{\mu_i+\psi}\right)^{y_i} \tag{C.8}$$

Note that the PDF has also been defined in the literature as:

$$f\left(y_i;\psi,\mu_i\right)=\begin{pmatrix} y_i+\psi-1 \\ \psi-1 \end{pmatrix}\left(\frac{\psi}{\mu_i+\psi}\right)^{\psi}\left(\frac{\mu_i}{\mu_i+\psi}\right)^{y_i} \tag{C.9}$$

The first two moments of the NB2 are the following:

$$E\left[y_i;\mu_i,\psi\right]=\mu_i \tag{C.10}$$

$$VAR\left[y_i;\mu_i,\psi\right]=\mu_i+\frac{\mu_i^2}{\psi} \tag{C.11}$$

The next step consists of defining the ***log-likelihood*** function of the NB2. It can be shown that:

$$\ln\left(\frac{\Gamma\left(y_i+\psi\right)}{\Gamma\left(\psi\right)}\right)=\sum_{j=0}^{y-1}\ln\left(j+\psi\right) \tag{C.12}$$

By substituting equation C-12 into C-8, the log-likelihood can be computed using the following equation:

$$\ln L\left(\psi,\beta\right)=\sum_{i=1}^{n}\left\{\left(\sum_{j=0}^{y-1}\ln\left(j+\psi\right)\right)-\ln y_i!-\left(y_i+\psi\right)\ln\left(1+\psi^{-1}\mu_i\right)+y_i\ln\psi^{-1}+y_i\ln\mu_i\right\} \tag{C.13}$$

Note also that the log-likelihood has also been expressed as:

$$\ln L\left(\psi,\beta\right)=\sum_{i=1}^{n}\left\{y_i\ln\left(\frac{\psi\mu_i}{1+\psi\mu_i}\right)-\psi^{-1}\ln\left(1+\psi\mu_i\right)+\ln\Gamma\left(y_i+\psi^{-1}\right)-\ln\Gamma\left(y_i+1\right)-\ln\Gamma\left(\psi^{-1}\right)\right\} \tag{C.14}$$

Recall that $\mu_i=\exp(\mathbf{x}_i'\boldsymbol{\beta})$.

In the statistical literature, the Poisson-Gamma model has also been defined as:

$$y_i \mid \lambda_i = Poisson(\lambda_i) \quad \text{i} = 1, 2, \ldots, \text{I} \tag{C.15}$$

where the mean of the Poisson is structured as:

$$\lambda_i = f(\mathbf{X};\boldsymbol{\beta})\exp(\varepsilon_i) = \mu_i \exp(\varepsilon_i) \tag{C.16}$$

and where, $f(.)$ is a function of the covariates, $\mathbf{X}$ (Miaou and Lord, 2003). As before, $\boldsymbol{\beta}$ is a vector of coefficients and ε_i is the model error independent of all the covariates with mean equal to 1 and a variance equal to $1/\psi$.

NB1 Model

The NB1 is very similar to the NB2, but the parameterization of the variance (the second moment) is slightly different than in equation C-11.

$$E\left[y_i;\mu_i,\psi\right] = \mu_i \tag{C.17}$$

$$VAR\left[y_i;\mu_i,\psi\right] = \mu_i + \frac{\mu_i}{\psi} \tag{C.18}$$

The log-likelihood of the NB1 is given by:

$$\ln L\left(\psi,\beta\right) = \sum_{i=1}^{n}\left\{\left(\sum_{j=0}^{y-1}\ln\left(j+\psi\mu_i\right)\right) - \ln y_i\,! - \left(y_i+\psi\mu_i\right)\ln\left(1+\psi^{-1}\right) + y_i \ln\psi^{-1}\right\} \tag{C.19}$$

The NB1 is usually less flexible in capturing the variance and is not used very often by analysts and statisticians. Interested readers are referred to Cameron and Trivedi (1998) for additional information about this parameterization.

Poisson-Gamma Model with Spatial Interaction

The Poisson-Gamma (or negative binomial model) can also incorporate data that are collected spatially. To capture this kind of data, a spatial autocorrelation term needs to be added to the model. Using the notation described in Equation C-15, the NB2 model with spatial interaction can be defined as:

$$y_i \mid \lambda_i = Poisson(\lambda_i) \tag{C.20}$$

with the mean of Poisson-Gamma organized as:

$$\lambda_i = \exp(\mathbf{x}_i'\boldsymbol{\beta} + \varepsilon_i + \phi_i) \tag{C.21}$$

The assumption on the uncorrelated error term ε_i is the same as in the Poisson-Gamma model described above; same as before, namely $\mu_i = \exp(\mathbf{x}_i'\boldsymbol{\beta})$. The third term in the expression, ϕ_i, is a *spatial random effect*, one for each observation. Together, the spatial effects are distributed as a complex *multivariate normal* (or Gausian) density function. In other words, the second model is a spatial regression model within a negative binomial model.

There are two common ways to express the spatial component, either as a *Conditional Autoregressive* (CAR) or as a *Simultaneous Autoregressive* (SAR) function (De Smith et al., 2007). The CAR model is expressed as:

$$E\left(y_i \mid \text{all } y_{j\neq i}\right)_{i\neq j} = \mu_i + \rho\sum_{ij}\left[w_{ij}\left(y_i - \mu_j\right)\right] \tag{C.22}$$

where μ_i is the expected value for observation *i*, w_{ij} is a spatial weight between the observation, *i*, and all other observations, *j* (and for which all weights sum to 1.0), and ρ is a spatial autocorrelation parameter that determines the size and nature of the spatial neighborhood effect. Note that there are different weight factors that have been proposed, such as the inverse distance weight function, exponential distance decay weight function and the Gaussian weighting function among others. The summation of the spatial weights times the difference between the observed and predicted values is over all other observations ($i \neq j$). The reader is referred to Haining (1990) and LeSage (2001) for further details.

The SAR model has a simpler form and can be expressed as:

$$E\left(y_i \mid \text{all } y_{j\neq i}\right)_{i\neq j} = \mu_i + \rho\sum_{ij}\left[w_{ij}y_j\right] \tag{C.23}$$

where the terms are as defined above. Note that in the CAR model the spatial weights are applied to the difference between the observed and expected values at all other locations whereas in the SAR model, the weights are applied directly to the observed value. In practice, the CAR and SAR models produce very similar results. Additional information about the Poisson-Gamma-CAR is described below.

Estimation Methods

This section describes two methods that can be used for estimating the coefficients of the regression NB models. The two methods are the maximum likelihood estimates (MLE) and the Monte Carlo Markov Chain (MCMC).

Maximum Likelihood Estimation

The characteristics of the MLE method were described in Appendix B for the normal and Poisson regression. The same characteristics apply here. The coefficients of the NB regression model are estimated by taking the first-order conditions and making them equal to zero. There are two first-order equations, one for the model's coefficients and one for the dispersion parameter (Lawson, 1987). The two for the NB2 are as follows:

$$\sum_{i=1}^{n} \frac{y_i - \mu_i}{1 + \psi^{-1}\mu_i} \mathbf{x_i} = \mathbf{0} \tag{C.24a}$$

$$\sum_{i=1}^{n} \left\{ \frac{1}{\left(\psi^{-1}\right)^2} \left(\ln\left(1 + \psi^{-1}\mu_i\right) - \sum_{j=0}^{y_i - 1} \frac{1}{\left(j + \psi\right)} \right) + \frac{y_i - \mu_i}{\psi^{-1}\left(1 + \psi^{-1}\mu_i\right)} \right\} = 0 \tag{C.24b}$$

where $\mathbf{x_i}$ is a vector of covariates.

Similar to the Poisson model, the series of equations can be solved using the Newton-Raphson procedure or the scoring algorithm. The confidence intervals on the βs and ψ^{-1} can be calculated using the covariance matrix that is assumed to be normally distributed:

$$\begin{bmatrix} \hat{\boldsymbol{\beta}} \\ \alpha \end{bmatrix} \sim N\left(\begin{bmatrix} \boldsymbol{\beta} \\ \alpha \end{bmatrix}, \begin{bmatrix} VAR[\boldsymbol{\beta}] & 0 \\ 0 & VAR[\alpha] \end{bmatrix} \right) \tag{C.25}$$

where,

$$VAR[\boldsymbol{\beta}] = \left(\sum_{i=1}^{n} \frac{\mu_i}{1 + \psi^{-1}\mu_i} \mathbf{x_i} \mathbf{x_i'} \right)^{-1} \tag{C.26a}$$

$$VAR[\alpha] = \left(\sum_{i=1}^{n} \frac{i}{\left(\psi^{-1}\right)^4} \left(\ln\left(1 + \psi^{-1}\mu_i\right) - \sum_{j=0}^{y_{j-1}} \frac{1}{\left(j + \psi\right)} \right)^2 + \frac{\mu_i}{\left(\psi^{-1}\right)^2 \left(1 + \psi^{-1}\mu_i\right)} \right)^{-1} \tag{C.26b}$$

It should be pointed out that the NB2 with spatial interaction model (Poisson-Gamma-CAR) cannot be estimated using the MLE method. It needs to be estimated using the MCMC technique, which is described next.

Monte Carlo Markov Chain Estimation

This section discusses how to draw samples from the posterior distribution of the Poisson-Gamma model and Poisson-Gamma-Conditional Autoregressive (CAR) model using the MCMC technique.

MCMC Poisson-Gamma Model

The Poisson-Gamma model can be formulated from a two-stage hierarchical Poisson model:

$$\text{(Likelihood)} \quad y_i \mid \lambda_i \sim Poisson\,(\lambda_i) \tag{C.27a}$$

$$\text{(First-stage)} \quad \lambda_i \mid \psi \sim \pi_\lambda(\psi)$$

(C.27b)

$$\text{(Second-stage)} \quad \psi \sim \pi_\psi(\cdot) \tag{C.27b}$$

where $\pi_\lambda(\psi)$ is the *prior distribution* imposed on the Poisson mean, λ_i, with a prior parameter ψ, and $\pi_\psi(\cdot)$ is the *hyper-prior* on ψ with known *hyper-parameters* (a, b, for example).

In Equations C-27a and C-27b, if we specify $\lambda_i = \nu_i \mu_i$ (where $\nu_i (= e^{\varepsilon_i}) \sim Gamma\ (\psi, \psi)$ in the first stage and $\psi \sim Gamma\ (a, b)$ in the second stage), these result in exactly the NB2 regression model described in the previous section. With this specification, it is also easy to show that λ_i in the first stage follows $Gamma\,(\psi, \psi / \mu_i)$ as shown in Equation C-5. Note that $\mu_i = \exp(\mathbf{x}_i'\boldsymbol{\beta})$ as described above.

For simplicity, if a *flat uniform prior* is assumed for each β_j ($j = 0, 1, \cdots, J$) and the parameters βs and ψ are mutually independent, the joint posterior distribution for the Poisson-Gamma model is defined as:

$$\pi(\boldsymbol{\lambda}, \boldsymbol{\beta}, \psi \mid \mathbf{y}) \propto f(\mathbf{y} \mid \boldsymbol{\lambda}) \cdot \pi(\boldsymbol{\lambda} \mid \boldsymbol{\beta}, \psi) \cdot \pi(\beta_0) \cdots \pi(\beta_J) \cdot \pi(\psi \mid a, b) \tag{C.28a}$$

$$= \left(\prod_{i=1}^{n} \frac{e^{-\lambda_i} (\lambda_i)^{y_i}}{y_i!} \right) \times \left(\prod_{i=1}^{n} \frac{\left(\psi e^{-\mathbf{x}_i'\boldsymbol{\beta}}\right)^{\psi}}{\Gamma(\psi)} \lambda_i^{\psi-1} e^{-(\psi e^{-\mathbf{x}_i'\boldsymbol{\beta}})\lambda_i} \right) \times \left(\psi^{(a-1)} e^{-b\psi} \right) \tag{C.28b}$$

The parameters of interest are $\boldsymbol{\lambda} = (\lambda_1, \cdots \lambda_n)$, $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots \beta_J)$, and the inverse dispersion parameter ψ (or the *dispersion parameter* $\gamma = 1/\psi$). Ideally, samples need to be drawn of each parameter from the joint posterior distribution. However, the form in Equation C-28b is very complex and it is difficult to draw samples from such a distribution. Consequently, samples are drawn from the full

conditional distribution *sequentially* (that is, one at a time). This iterative process is called the Gibbs sampling method.

Therefore, once the full conditionals are known for each parameter, Gibbs sampling can be implemented by drawing samples of each parameter sequentially. The full conditional distributions for each parameter for the Poisson-Gamma model can be easily derived from Equation C-28b and are given as (Park, 2010):

$$\pi(\lambda_i \mid \boldsymbol{\beta}, \psi, y_i) \propto f(y_i \mid \lambda_i) \cdot \pi(\lambda_i \mid \boldsymbol{\beta}, \psi)$$
$$= Gamma\,(y_i + \psi, 1 + \psi e^{-\mathbf{x}_i'\boldsymbol{\beta}}), \text{ for } i = 1, 2, \cdots, n \qquad \text{(C.29a)}$$

$$\pi(\beta_j \mid \boldsymbol{\lambda}, \boldsymbol{\beta}_{-j}, \psi) \propto \pi(\boldsymbol{\lambda} \mid \boldsymbol{\beta}_{-j}, \psi) \cdot \pi(\beta_j)$$
$$= \exp\left\{-\psi\left[\left(\sum_{i=1}^{n} x_{ij}\right)\beta_j + \sum_{i=1}^{n} \lambda_i e^{-\mathbf{x}_i'\boldsymbol{\beta}}\right]\right\}, \text{ for } j = 0, 1, \cdots, J \qquad \text{(C.29b)}$$

$$\pi(\psi \mid \boldsymbol{\lambda}, \boldsymbol{\beta}, a, b) \propto \pi(\boldsymbol{\lambda} \mid \boldsymbol{\beta}, \psi) \cdot \pi(\psi \mid a, b) \qquad \text{(C.29c)}$$
$$= \exp\left\{-n \ln(\Gamma(\psi)) + \psi\left(n \ln(\psi) - \sum_{i=1}^{n}\left(\mathbf{x}_i'\boldsymbol{\beta} + \ln(\lambda_i) + \lambda_i e^{-\mathbf{x}_i'\boldsymbol{\beta}}\right)\right) + (a-1)\ln(\psi) - b\psi\right\}$$

However, unlike Equation C-29a, the full conditional distributions for the βs and ψ (Equations C-29b and C-29c) do not belong to any standard distribution family so it is not easy to draw samples directly from their full conditional distributions. While there are several approaches to sampling from such a complex distribution, the particular sampling algorithm used in *CrimeStat* is a Metropolis-Hastings (or MH) algorithm with *slice sampling* of individual parameters.

The MCMC sampling procedure using the slice sampling algorithm within Gibbs sampling, therefore, can be summarized as follows:

1. Start with initial values $\boldsymbol{\lambda}^{(0)}$, $\boldsymbol{\beta}^{(0)}$ and $\psi^{(0)}$. Repeat the following steps for $t = 1, \cdots, T_0, \cdots, T_0 + T$.
2. *Step 1*: Conditional on knowing $\boldsymbol{\beta}^{(t\text{-}1)}$ and $\psi^{(t\text{-}1)}$, draw $\boldsymbol{\lambda}^{(t)}$ from Equation C-29a independently for $i = 1, 2, \cdots, n$.
3. *Step 2*: Conditional on knowing $\boldsymbol{\lambda}^{(t)}$ and $\psi^{(t\text{-}1)}$, draw $\boldsymbol{\beta}^{(t)}$ from Equation C-29b independently for $j = 0, 1, \cdots, J$ using the slice sampling method.
4. *Step 3*: Conditional on knowing $\boldsymbol{\lambda}^{(t)}$ and $\boldsymbol{\beta}^{(t)}$, draw $\psi^{(t)}$ from Equation C-29c using the slice sampling method.

5. *Step 4*: Store the values of all parameters (i.e., $\boldsymbol{\lambda}^{(t)}$, $\boldsymbol{\beta}^{(t)}$ and $\psi^{(t)}$). Increase t by one and return to Step 1.
6. *Step 5*: Discard the first *k* draws as a *burn-in* period, where *k* is defined by the user.

After equilibrium is reached at the k^{th} iteration, sampled values are averaged to provide the consistent estimates of the parameters:

$$\hat{E}[h(\theta)] = \frac{\sum_{t=T_0+1}^{T} h(\theta)^{(t)}}{T} \tag{C.30}$$

where θ denotes any parameter of interest in the model.

MCMC Poisson-Gamma-CAR Model

For the Poisson-Gamma-CAR model, the only difference from the Poisson-Gamma model is the way λ_i is structured. The mean of Poisson-Gamma-CAR is organized as:

$$\lambda_i = \exp(\mathbf{x}_i'\boldsymbol{\beta} + \varepsilon_i + \phi_i) \tag{C.31}$$

where ϕ_i is a spatial random effect, one for each observation. As in the Poisson-Gamma model, we specify $e^{\varepsilon_i} \sim Gamma(\psi, \psi)$ to model the independent error term. To model the spatial effect, ϕ_i, we assume the following:

$$p(\phi_i \mid \boldsymbol{\Phi}_{-i}) \propto \exp\left(-\frac{w_{i+}}{2\sigma_\phi^2}\left[\phi_i - \rho \sum_{j \neq i} \frac{w_{ij}}{w_{i+}} \phi_j \right]^2 \right) \tag{C.32}$$

where $p(\phi_i \mid \boldsymbol{\Phi}_{-i})$ is the probability of a spatial effect given a lagged spatial effect, $w_{i+} = \sum_{i \neq j} w_{ij}$ which sums all over all records, j (i.e., all other zones) except for the record of interest, i. This formulation gives a conditional normal density with mean $= \rho \sum_{j \neq i} \frac{w_{ij}}{w_{i+}} \phi_j$ and variance $= \frac{\sigma_\phi^2}{w_{i+}}$. The parameter ρ determines the direction and overall magnitude of the spatial effects. The term w_{ij} is a spatial weight function between zones i and j. In the algorithm, the term for the variance is $\sigma_\phi^2 = 1/\tau_\phi$ and the same variance is used for all observations.

We define the spatial weight matrix $\mathbf{W}$ with the entries w_{ij} and the diagonal entries $w_{ii} = 0$. The matrix $\mathbf{D}$ is defined as a diagonal matrix with the diagonal entries, w_{i+}. Sun, Tsutakawa, and Speckman (1999) show that if $\kappa_{\min}^{-1} < \rho < \kappa_{\max}^{-1}$ where $\kappa_{\min}$ and $\kappa_{\max}$ are the smallest and largest eigenvalues of $\mathbf{WD}^{-1}$ respectively, then $\mathbf{\Phi}$ has a multivariate normal distribution with mean **0** and nonsingular covariance matrix $\sigma_\phi^2(\mathbf{D} - \rho\mathbf{W})^{-1}$.

$$\mathbf{\Phi} = (\phi_1, \cdots \phi_n)' = MNV_n\left(\mathbf{0}, \sigma_\phi^2 \mathbf{A}^{-1}\right) = \frac{|\mathbf{A}|^{1/2}}{(2\pi\sigma_\phi^2)^{n/2}} \exp\left(-\frac{1}{2\sigma_\phi^2}\mathbf{\Phi}'\mathbf{A}\mathbf{\Phi}\right) \tag{C.33}$$

where $\mathbf{A} = (\mathbf{D} - \rho\mathbf{W})$ and $\kappa_{\min}^{-1} < \rho < \kappa_{\max}^{-1}$.

Prior Distributions for MCMC Poisson-Gamma-CAR

For the prior distributions, we assume the following distributions for each parameter:

Parameter	Prior distribution
$\beta_j \ (j = 0, 1, \cdots, J)$	$Uniform(-\infty, \infty)$
ψ	$Gamma\,(a_\psi, b_\psi)$
$\tau_\phi (= \sigma_\phi^{-2})$	$Gamma\,(a_\phi, b_\phi)$
ρ	$Uniform(\kappa_{\min}^{-1}, \kappa_{\max}^{-1})$

The parameters in the Poisson-Gamma-CAR model are $\boldsymbol{\lambda} = (\lambda_1, \cdots \lambda_n)$, $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots \beta_J)$, ψ, $\mathbf{\Phi} = (\phi_1, \cdots \phi_n)$, τ_ϕ and ρ. Then, the random samples can be drawn from the full conditional distributions of each parameter. It can be shown that the full conditional distributions for each parameter are given as follows:

$$\pi(\lambda_i \mid others) \sim Gamma\,(y_i + \psi, 1 + \psi e^{-\mathbf{x}_i'\boldsymbol{\beta} - \phi_i}), \text{ for } i = 1, 2, \cdots, n \tag{C.34a}$$

$$\pi(\beta_j \mid others) \propto \exp\left\{-\psi\left[\left(\sum_{i=1}^{n} x_{ij}\right)\beta_j + \sum_{i=1}^{n} \lambda_i e^{-\mathbf{x}_i'\boldsymbol{\beta} - \phi_i}\right]\right\}, \text{ for } j = 0, 1, \cdots, J \tag{C.34b}$$

$$\pi(\psi \mid others) \tag{C.34c}$$

$$\propto \exp\left\{-n\ln(\Gamma(\psi))+\psi\left(n\ln(\psi)-\sum_{i=1}^{n}\left(\mathbf{x}_i'\boldsymbol{\beta}-\phi_i+\ln(\lambda_i)-\lambda_i e^{-\mathbf{x}_i'\boldsymbol{\beta}-\phi_i}\right)\right)+(a_\psi-1)\ln(\psi)-b_\psi\psi\right\}$$

$$\pi(\phi_i \mid others) \propto \exp\left\{-\psi\phi_i-\psi\lambda_i e^{-\mathbf{x}_i'\boldsymbol{\beta}-\phi_i}-\frac{\tau_\phi}{2}(\boldsymbol{\Phi}^T\mathbf{A}\boldsymbol{\Phi})\right\},\ \text{for } i=1,2,\cdots,n \qquad \text{(C.34d)}$$

$$\pi(\tau_\phi \mid others) \propto Gamma\left(a_\phi+\frac{n}{2}, b_\phi+\frac{1}{2}\boldsymbol{\Phi}^T\mathbf{A}\boldsymbol{\Phi}\right) \qquad \text{(C.34e)}$$

$$\pi(\rho \mid others) \propto \exp\left\{\frac{1}{2}\sum_{i=1}^{n}\ln(1-\rho\kappa_i)-\frac{\tau_\phi}{2}(\boldsymbol{\Phi}^T\mathbf{A}\boldsymbol{\Phi})\right\} \qquad \text{(C.34f)}$$

where $\kappa_1,\cdots,\kappa_n$ are the eigenvalues of $\mathbf{WD}^{-1}$.

Since the full conditional distributions were specified, the Gibbs sampling method can be applied sequentially. It is easy to generate random samples from Equations C-34a and C-34e. The other full conditional distributions are not of closed form, so the slice sampling method should be applied.

Likelihood Statistics

There are many measures that can be used for estimating how well the model fits the data. Some of them have already been discussed in Appendix B but are also included here for the sake of completeness. They fall into three groups. First, there are statistics for indicating the likelihood level of a model, that is, how well the model maximizes the likelihood function. Among these statistics are:

Akaike Information Criterion (AIC)

The AIC is another measure of fit that can be used to assess models. This measure also uses the log-likelihood, but add a penalizing term associated with the number of variables. It is well known that by adding variables, one can improve the fit of models. Thus, the AIC tries to balance the goodness-of-fit versus the inclusion of variables in the model. The AIC is computed as:

$$AIC = -2\ln L + 2p \qquad \text{(C.37)}$$

where p is the number of unknown parameters included in the model (this also includes the inverse dispersion parameter ψ and random spatial effect f_i) and $\ln L$ is the log-likelihood described in Equations C-13 or C-14. Smaller values are better.

Bayes Information Criterion (BIC)

Similar to the AIC, the BIC also employs a penalty term associated with the number of parameters (p) and the sample size (n). This measure is also known as the Schwarz Information Criterion. It is computed the following way:

$$AIC = -2\ln L + p\ln n \tag{C.38}$$

Again, smaller values are better.

Deviance Information Criterion (DIC)

When the Bayesian estimation method is used, the DIC is often used as a goodness-of-fit (GOF) measure instead of the AIC or BIC. The latter ones are generally used for the maximum likelihood method. The DIC is defined as follows:

$$DIC = \hat{D} + 2(\overline{D} - \hat{D}) \tag{C.39}$$

where $\overline{D}$ is the average of the deviance ($-2\ln L$) over the posterior distribution, and $\hat{D}$ is the deviance calculated at the posterior mean parameters. As with the AIC and BIC, the DIC uses $p_D = \overline{D} - \hat{D}$ (effective number of parameters) as a penalty term on the goodness of fit. Differences in DIC from 5-10 indicate that one model is clearly better (Spiegelhalter et al., 2002).

Deviance

The deviance is a measure of goodness of fit that can be used to assess models. It is defined as twice the difference between the maximum likelihood achievable ($y_i = \hat{\mu}_i$) and the likelihood of the fitted model (the ^ refers to the estimate of the variable that is based on the data):

$$D(\mathbf{y}, \mathbf{u}) = 2\left\{L(\mathbf{y}) - L(\hat{\boldsymbol{\mu}})\right\} \tag{C.35}$$

For the NB2 model, the deviance can be computed as:

$$D = 2\sum_{i=1}^{n}\left\{y_i \ln\left(\frac{y_i}{\hat{\mu}_i}\right) - \left(y_i + \psi^{-1}\right)\ln\left[\frac{y_i + \psi^{-1}}{\hat{\mu}_i + \psi^{-1}}\right]\right\} \tag{C.36}$$

Smaller values mean that the model fits the data better.

Pearson Chi-Square

Another useful likelihood statistic is the *Pearson Chi*-square and is defined as

$$Pearson - \chi^2 = \sum_{i=1}^{N}\frac{(y_i - \hat{\mu}_i)^2}{VAR(y_i)} \tag{C.37}$$

If the mean and the variance are properly specified, then $E\left[\sum_{i=1}^{n}\left(y_i-\mu_i\right)^2/VAR\left(y_i\right)\right]=n$ (Cameron and Trivedi, 1998). Values closer to n (the sample size) show a better fit. Recall that the variance for the NB2 model is $VAR\left(y_i\right)=\hat{\mu}_i+\hat{\mu}_i^2/\psi$.

Model Error Estimates

Second, there are statistics for estimating how well the model fit the data and the converse, how much error was in the model. Two error statistics are particularly useful.

Mean Absolute Deviation (MAD)

This criterion has been proposed by Oh et al. (2003) to evaluate the fit of models. The Mean Absolute Deviance (MAD) calculates the absolute difference between the estimated and observed values

$$MAD=\frac{1}{n}\sum_{i=1}^{n}\left|\hat{\mu}_i-y_i\right| \tag{C.38}$$

Mean Squared Prediction Error (MSPE)

The Mean Squared Prediction Error (MSPE) is a traditional indicator of error and calculates the difference between the estimated and observed values squared.

$$MPSE=\frac{1}{n}\sum_{i=1}^{n}\left(\hat{\mu}_i-y_i\right)^2 \tag{C.39}$$

A value closer to 1 means the model fits the data better.

Over-dispersion Tests

Third, there are statistics for indicating the degree of over-dispersion in the model, including:

Adjusted Deviance

The *adjusted deviance* is defined as the deviance divided by the degrees of freedom (N-K-1). A value closer to 1 indicates a satisfactory GOF. Usually, values greater than 1 indicate signs of over-dispersion, while values below 1 show signs of under-dispersion.

Adjusted Pearson Chi-Square

The *adjusted Pearson Chi-square* is defined as the Pearson Chi-square divided by the degrees of freedom. A value closer to 1 indicates a satisfactory goodness-of-fit.

Dispersion Multiplier

The *dispersion* multiplier, γ, measures the extent to which the conditional variance exceeds the conditional mean (conditional on the independent variables and the intercept term) and is defined by $Var(y_i) = \mu_i + \gamma\mu_i^2$

Inverse Dispersion Multiplier

The *inverse dispersion multiplier* (ψ) is simply the reciprocal of the dispersion multiplier $(\psi = 1/\gamma)$; some users are more familiar with it in this form.

It should be pointed out that many GOF measures are not useful when a single model is evaluated. The measures are therefore relevant when several models are compared with each other (i.e., different functional forms or when different variables are included in the models).

There are other measures that can be used for estimating the goodness-of-fit and the amount of error in models, but are not presented here. Readers can find additional measures in Mitra and Washington (2007) and Lord and Park (2008).

References

Cameron, A.C. & Trivedi, P.K. (1998). *Regression Analysis of Count Data*. Cambridge University Press. Cambridge, UK.

Cook, J.D. (2009). Notes on the Negative Binomial Distribution. www.johndcook.com/negative_binomial.pdf (accessed on Aug. 15th, 2010).

De Smith, M., Goodchild, M.F., & Longley, P.A. (2007). *Geospatial Analysis* (second edition). Matador: Leicester, U.K.

Greenwood, M., & Yule, G.U. (1920). An Inquiry into the Nature of Frequency Distributions of Multiple Happenings, with Particular Reference to the Occurrence of Multiple Attacks of Disease or Repeated Accidents. Journal of the Royal Statistical Society A, Vol. 83, pp. 255-279.

Haining, R. (1990). *Spatial Data Analysis in the Social and Environmental Sciences*. Cambridge University Press, Cambridge, U.K.

Hilbe, J.M. (2007) *Negative Binomial Regression*. Cambridge University Press, Cambridge, U.K.

Lawless, J.F. (1987). Negative binomial and mixed Poisson regression. The Canadian Journal of Statistics, Vol. 15, No. 3, pp. 209-225.

LeSage, J.P. (2001). A Family of Geographically Weighted Regression Models. Working Paper. Department of Economics, University of Toledo, Toledo, OH.

Lord, D., & Park, P.Y.J. (2008). Investigating the effects of the fixed and varying dispersion parameters of Poisson-Gamma models on empirical Bayes estimates. Accident Analysis & Prevention, Vol. 40, No. 4, pp. 1441-1457.

References (continued)

Lord, D., Washington, S.P., & Ivan, J.N. (2005). Poisson, Poisson-Gamma and Zero Inflated Regression Models of Motor Vehicle Crashes: Balancing Statistical Fit and Theory. Accident Analysis & Prevention. Vol. 37, No. 1, pp. 35-46.

Maher M.J., & Summersgill, I. (1996) A Comprehensive Methodology for the Fitting Predictive Accident Models. Accident Analysis & Prevention, Vol. 28, No. 3, pp.281-296.

Miaou, S.-P., & Lord, D. (2003). Modeling TraffiC.Flow Relationships at Signalized Intersections: Dispersion Parameter, Functional Form and Bayes vs Empirical Bayes. Transportation Research Record 1840, pp. 31-40.

Mitra, S., & Washington, S. (2007). On the nature of over-dispersion in motor vehicle crash prediction models. Accident Analysis & Prevention, Vol. 39, No. 3, pp. 459-468.

Oh, J., Lyon, C., Washington, S., Persaud, B., & Bared, J. (2003). Validation of FHWA crash models for rural intersections: lessons learned. Transportation Research Record 1840, pp. 41-49.

Park, B.-J. (2010). Application of the Finite Mixture Models for Vehicle Crash Data Analysis. PhD Dissertation, Texas A&M University, College Station, TX.

Spiegelhalter, D.J., Best, N.G., Carlin, B.P., van der Linde, A. (2002) Bayesian measures of model complexity and fit. Journal of the Royal Statistical Society, Series B, Vol. 64, No. 4, pp. 583-639.

Sun, D., Tsutakawa, R. K., & Speckman, P. L. (1999). Posterior distribution of hierarchical models using CAR(1) distributions. Biometrika, Vol. 86, No. 2, pp. 341-350.