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

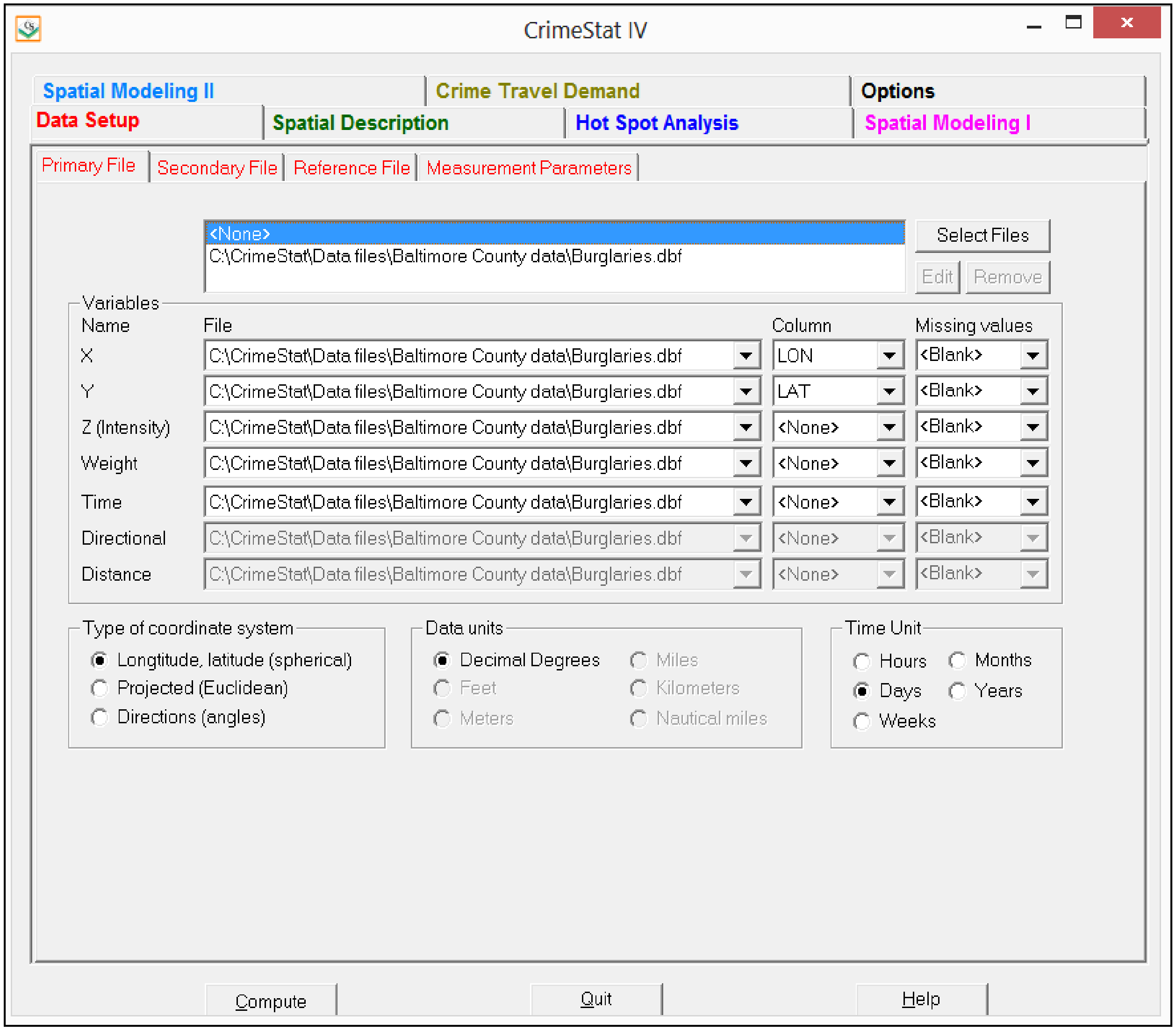
Spatial Description

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.

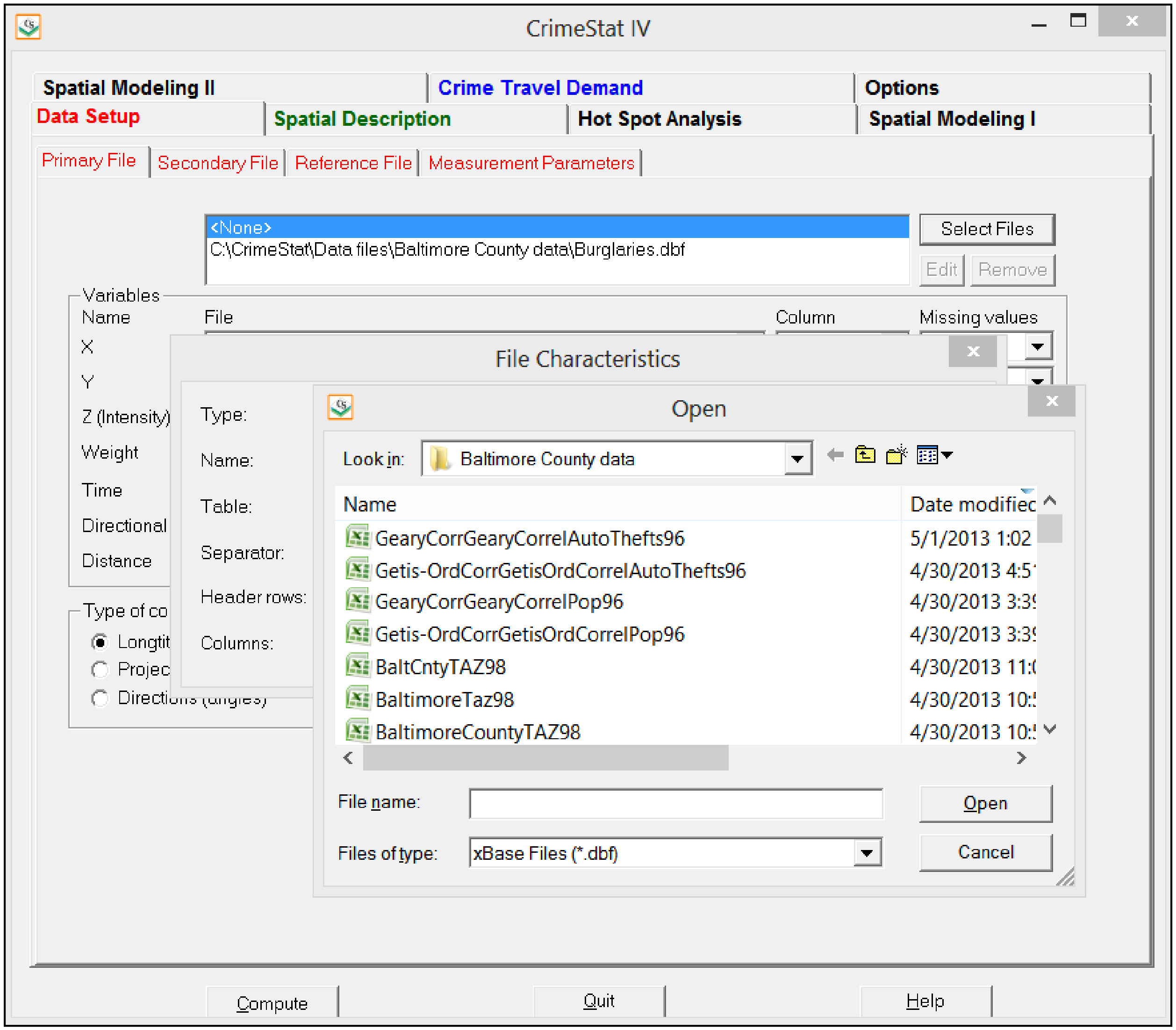
ASCII

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

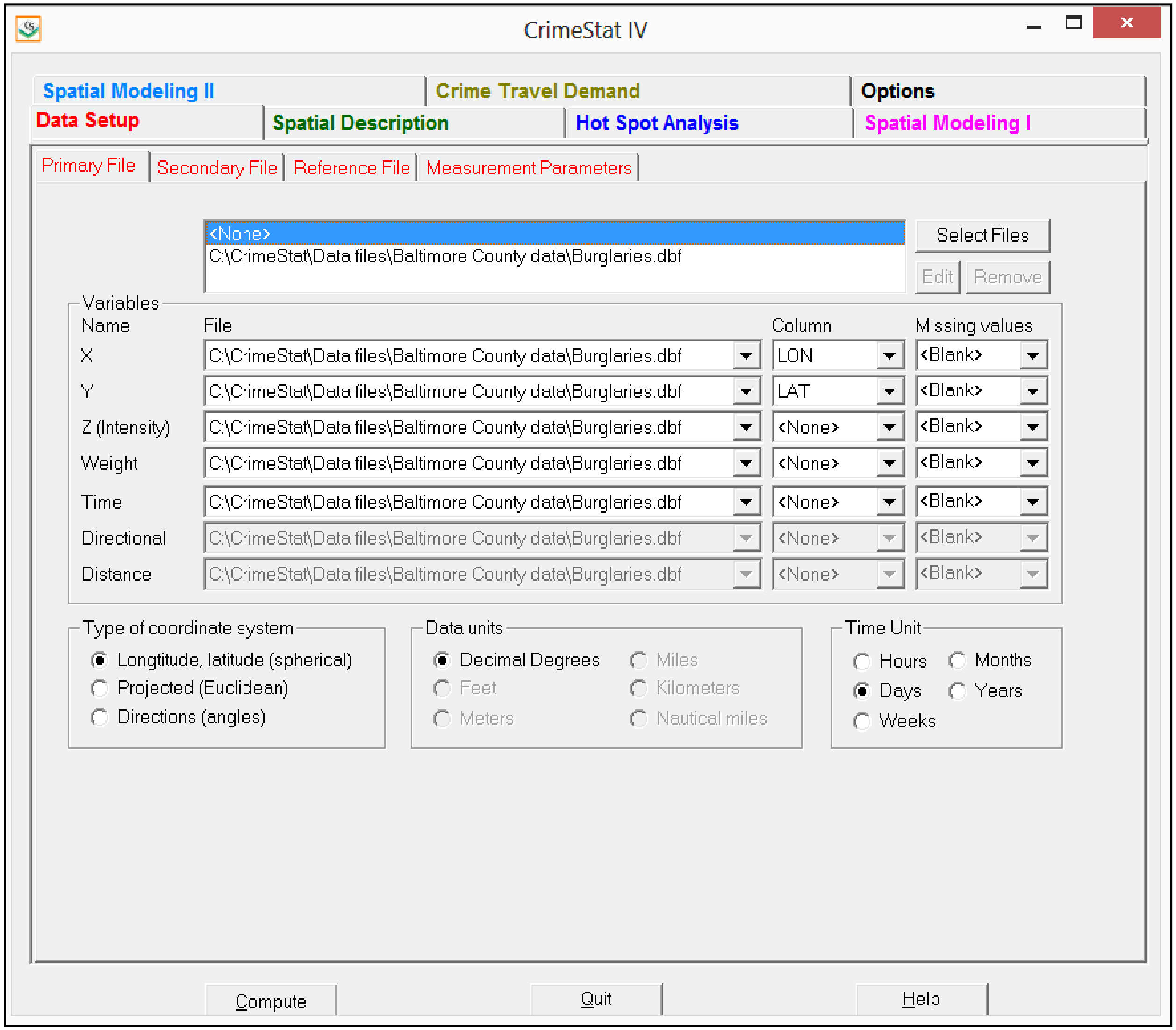
After defining a file and its format, either 'dbf', 'shp' or ASCII, it is necessary to identify the variables. Two variables are required and two are optional. The required variables are the X and Y coordinates. The user should indicate the file name that contains the coordinates by clicking on the drop 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

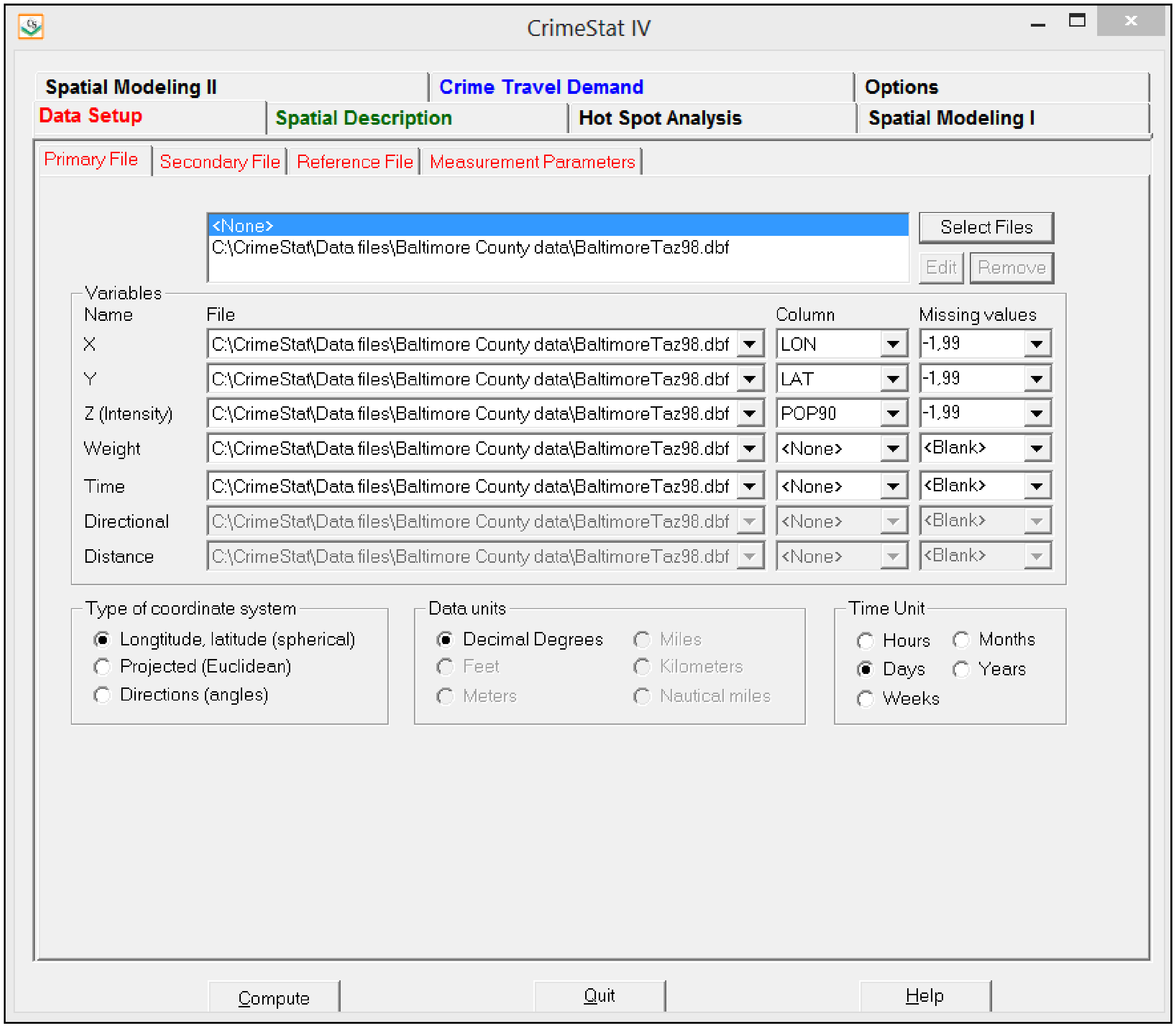
Finally, a time variable can be defined for use in the special Space-time analysis tools under Spatial modeling. *CrimeStat* allows five different time references:

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*).

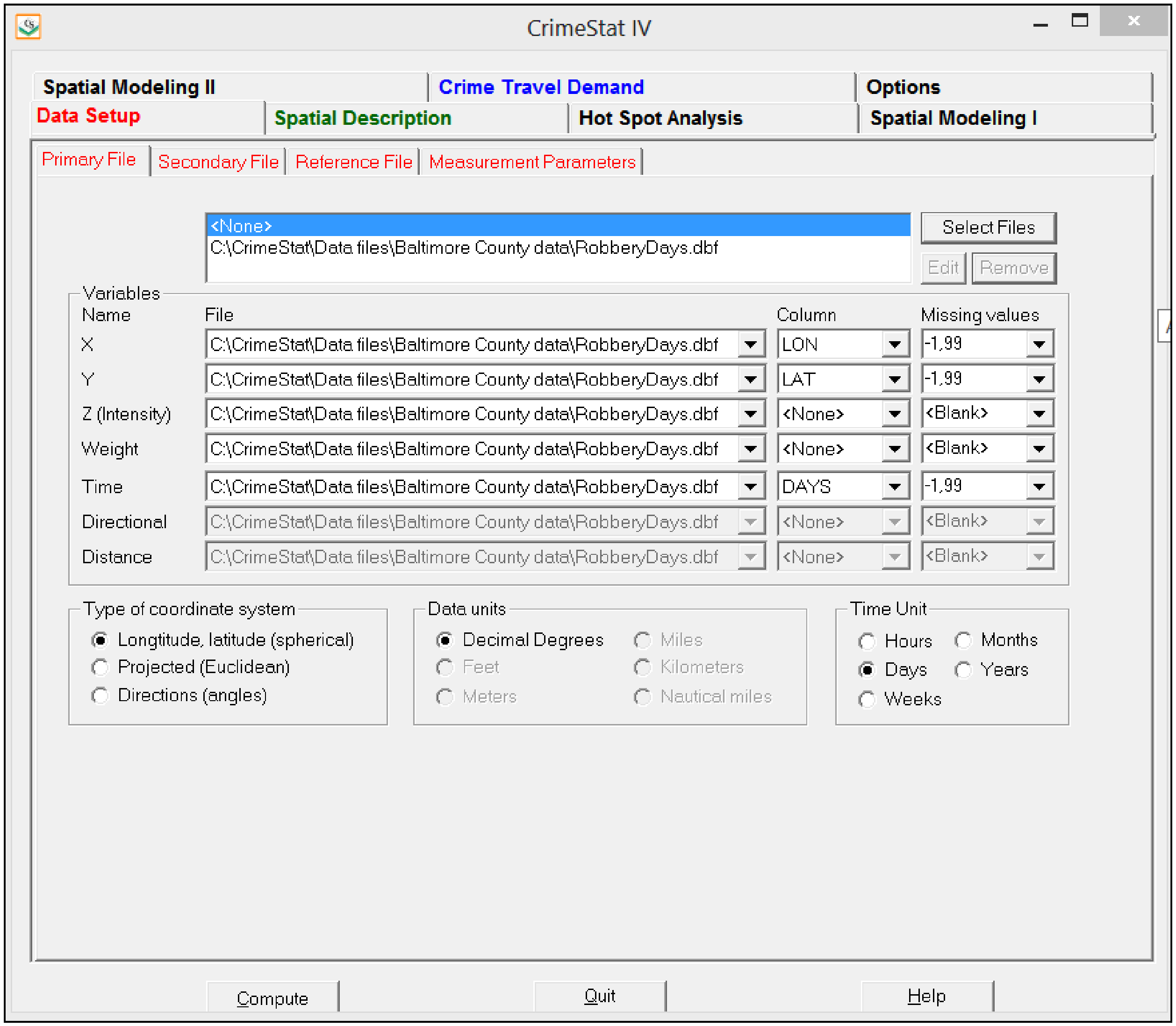
CrimeStat can work with either spherical or projected coordinates. On the primary file tab, the user indicates which coordinate system is being used. If the coordinate system is spherical, then units are automatically assumed to be latitude and longitude in decimal degrees. If the coordinate system is 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)

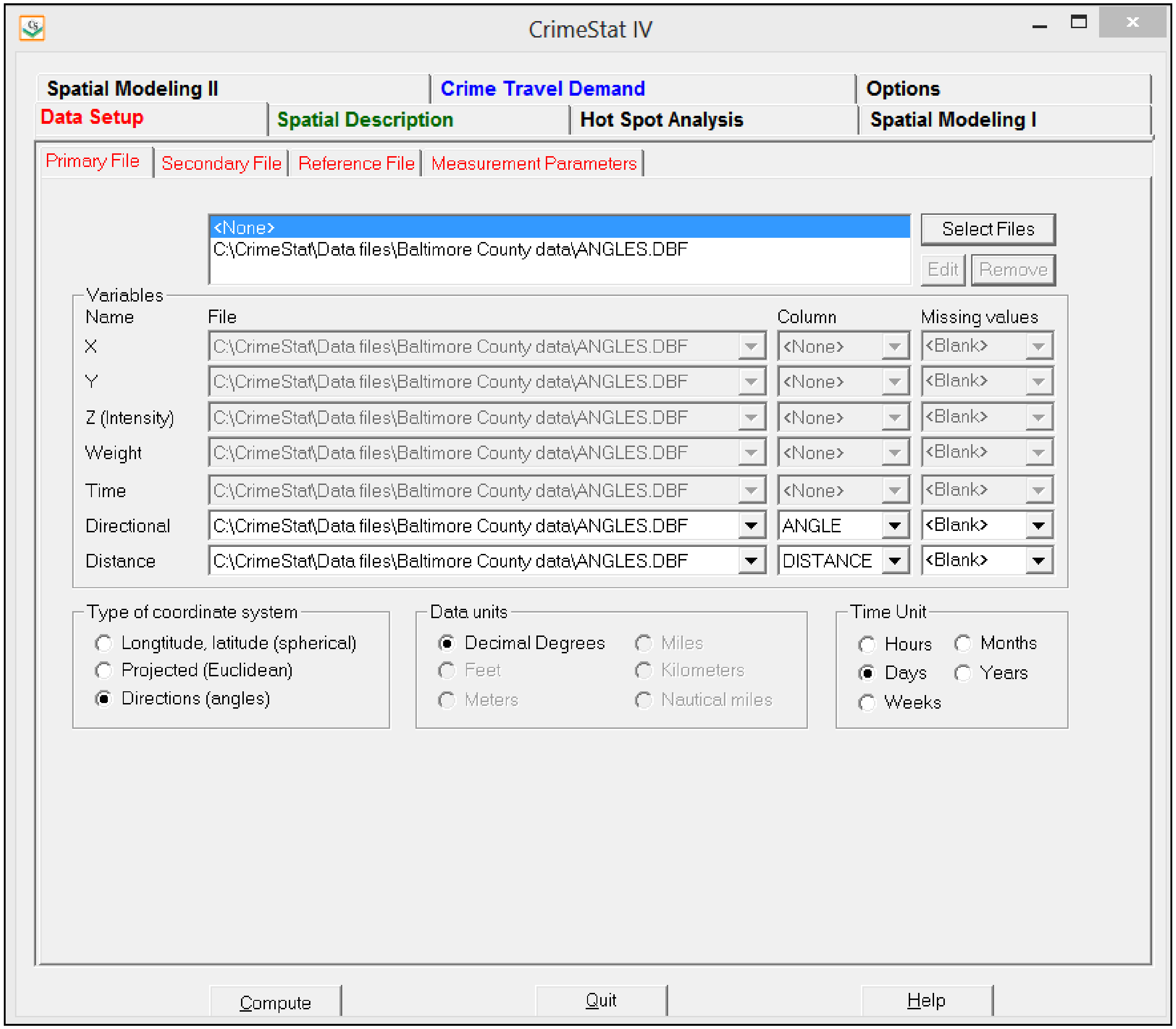


Figure 3.7:

Ascii File Selection of Secondary File

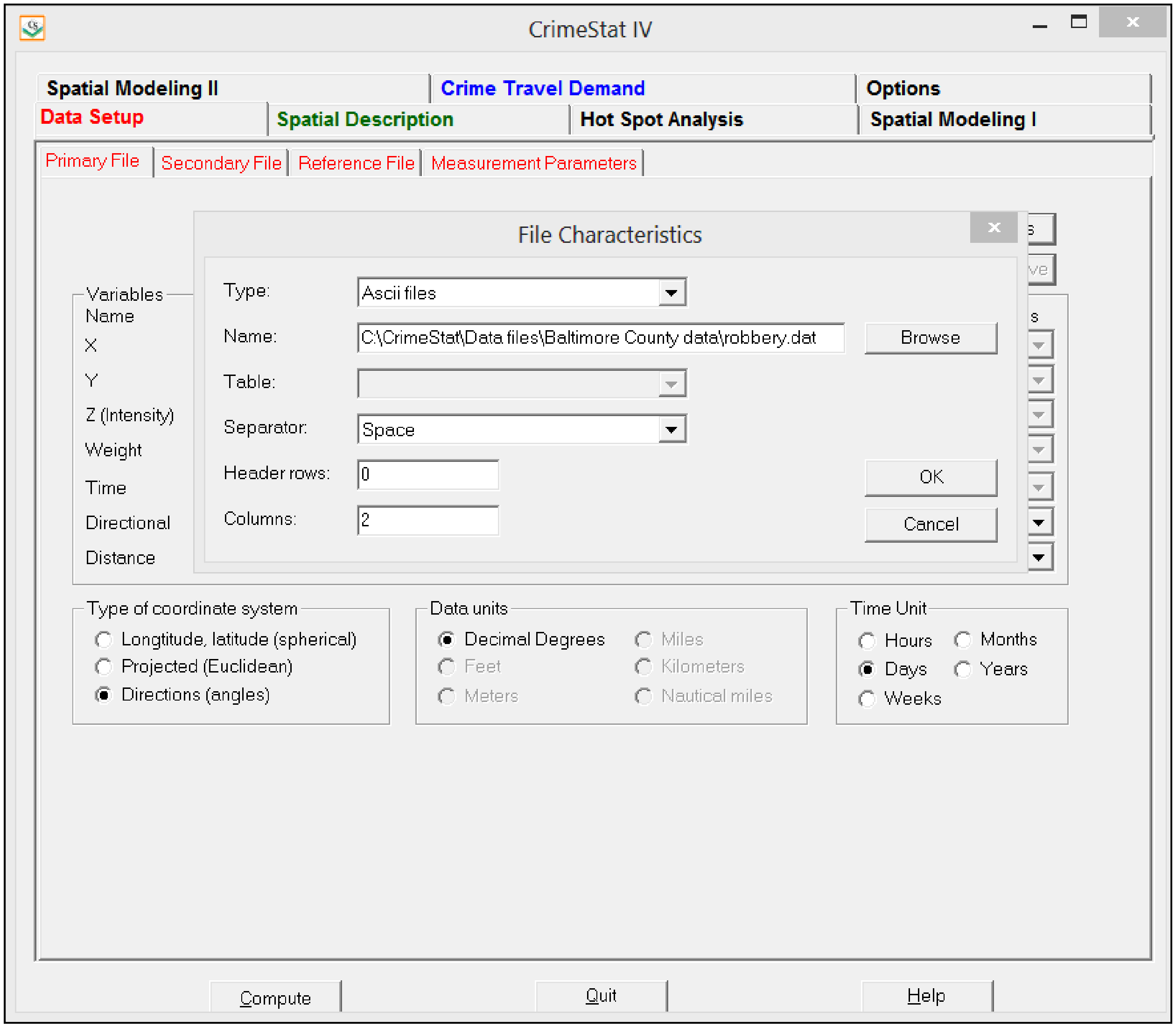
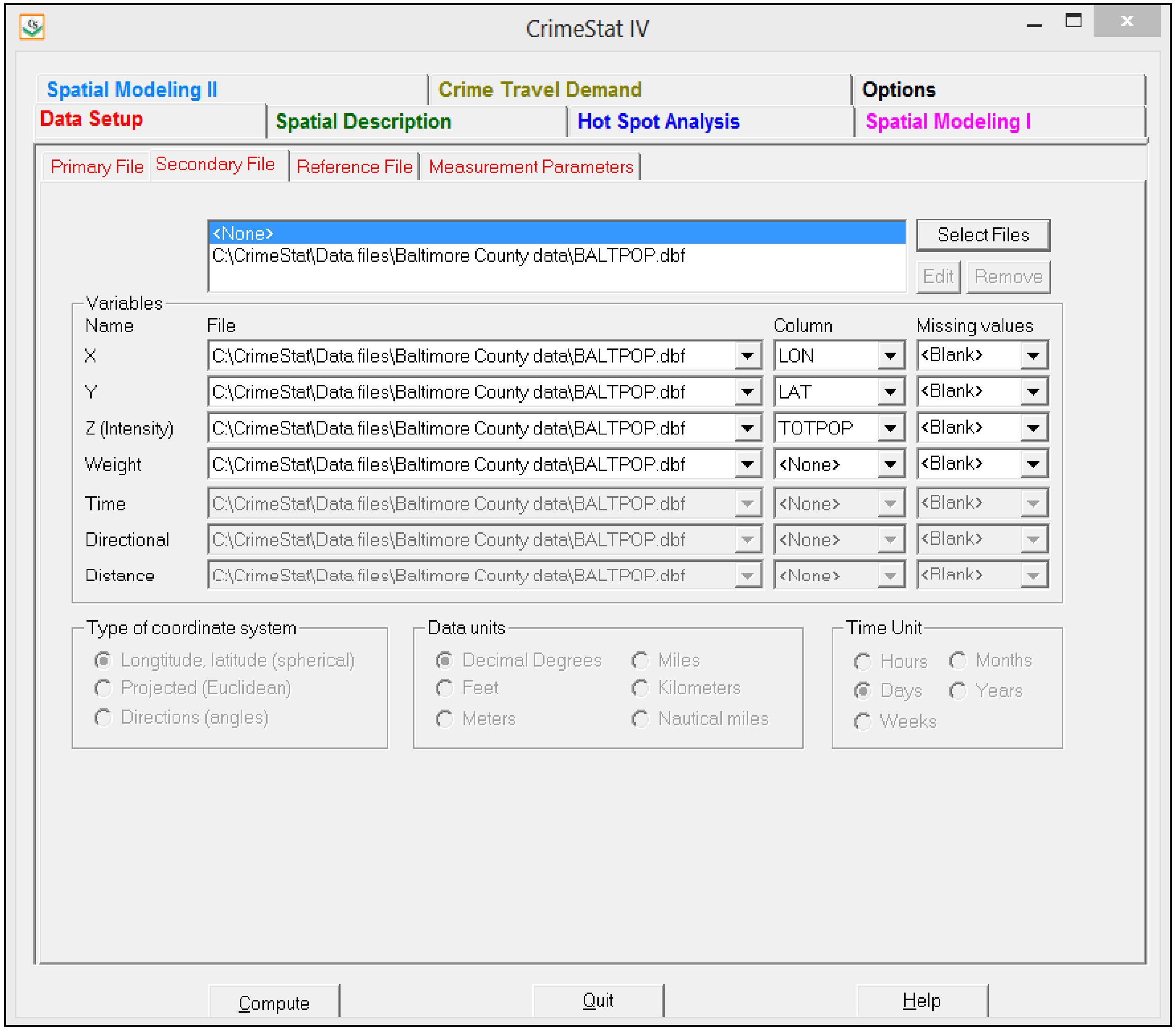


Figure 3.8:

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.

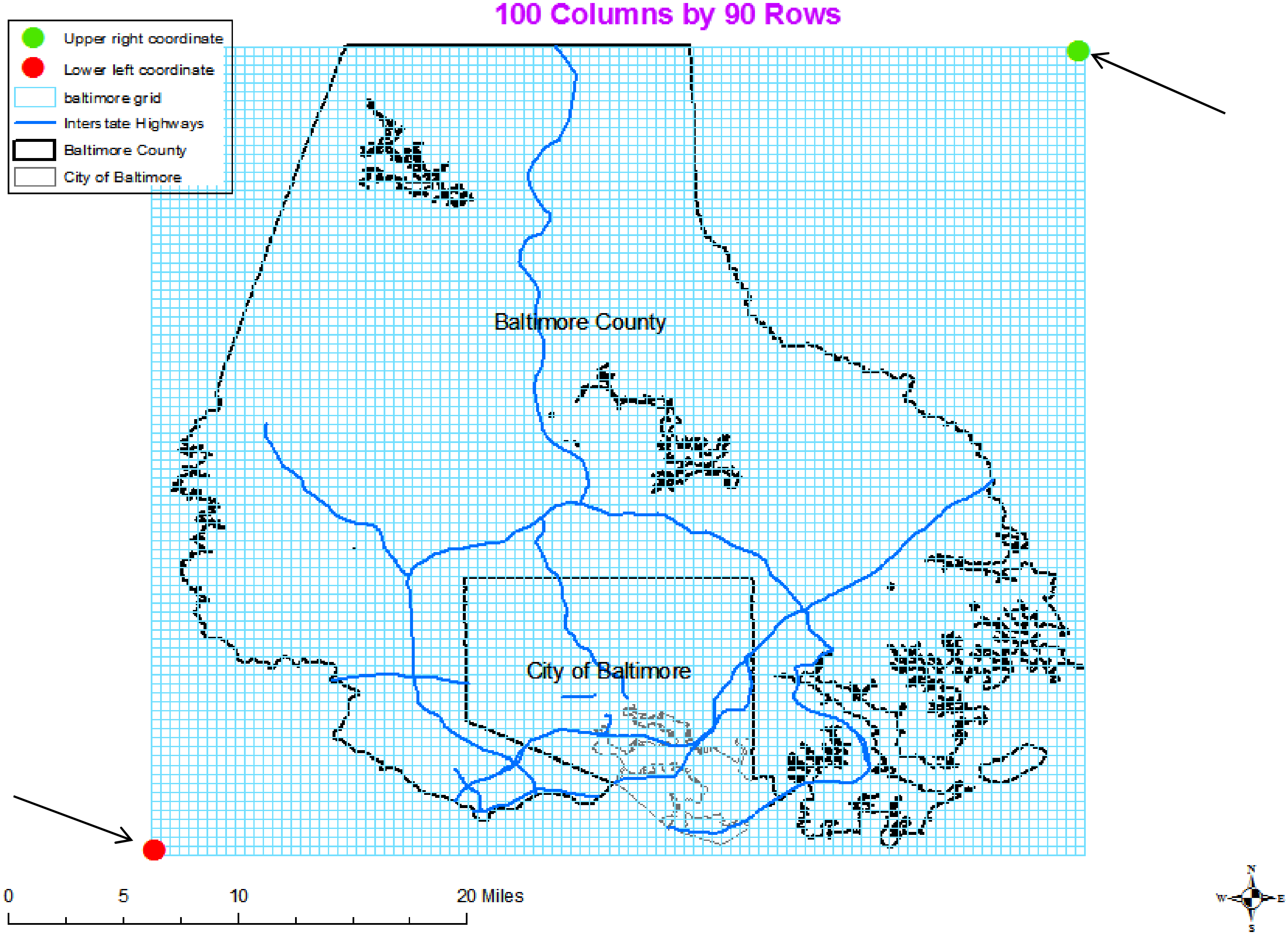
2. The user selects whether the grid is to be created by cell spacing or by the number of columns.

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



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

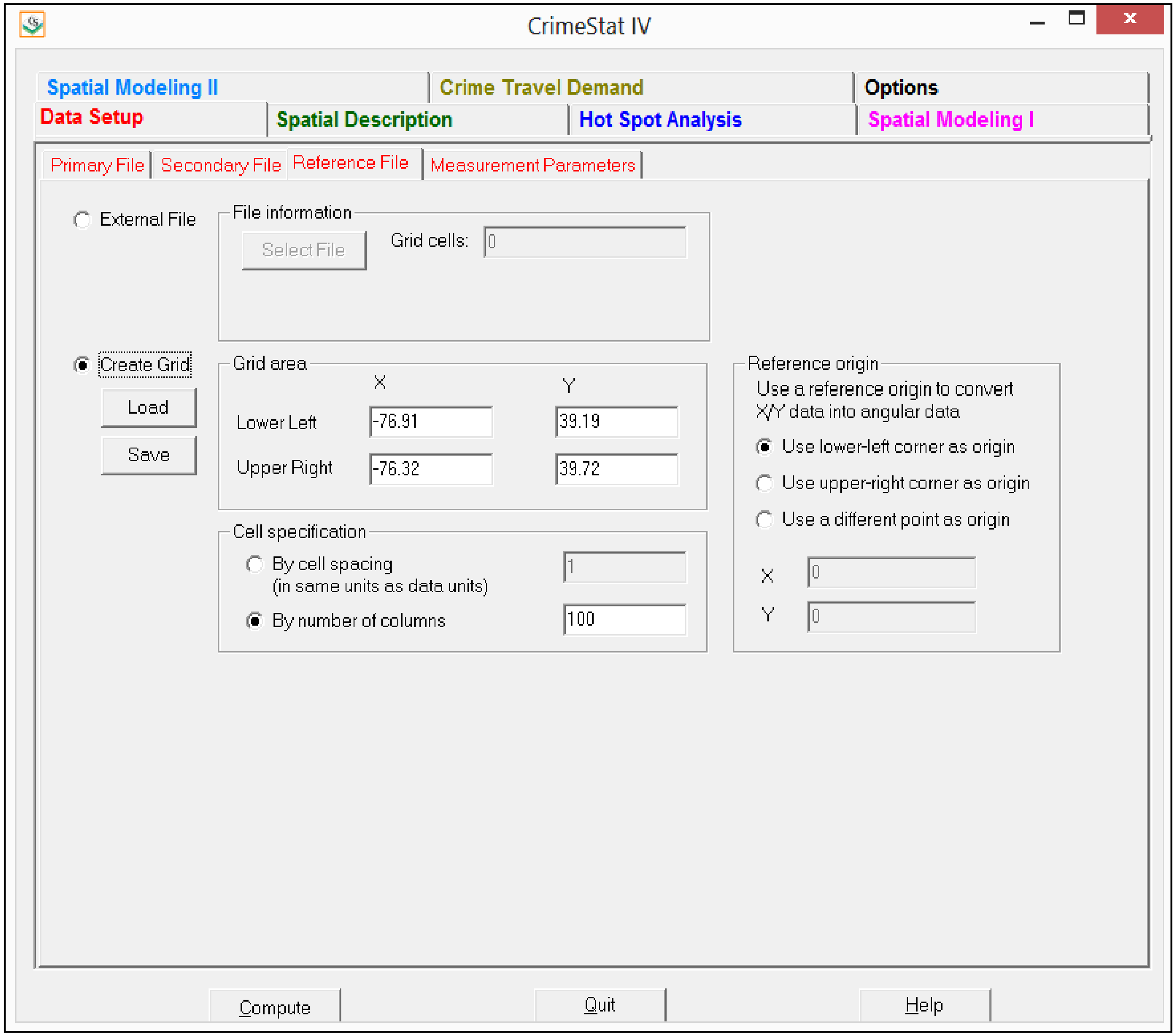


Figure 3.11 shows a correctly defined reference file using an existing grid file. One must be careful in using a file which is not a grid. *CrimeStat* can output the results of the interpolation routines in several GIS formats - *Surfer for Windows*, *ArcGIS Spatial Analyst*®, *ArcGIS*®, *MapInfo*® and 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

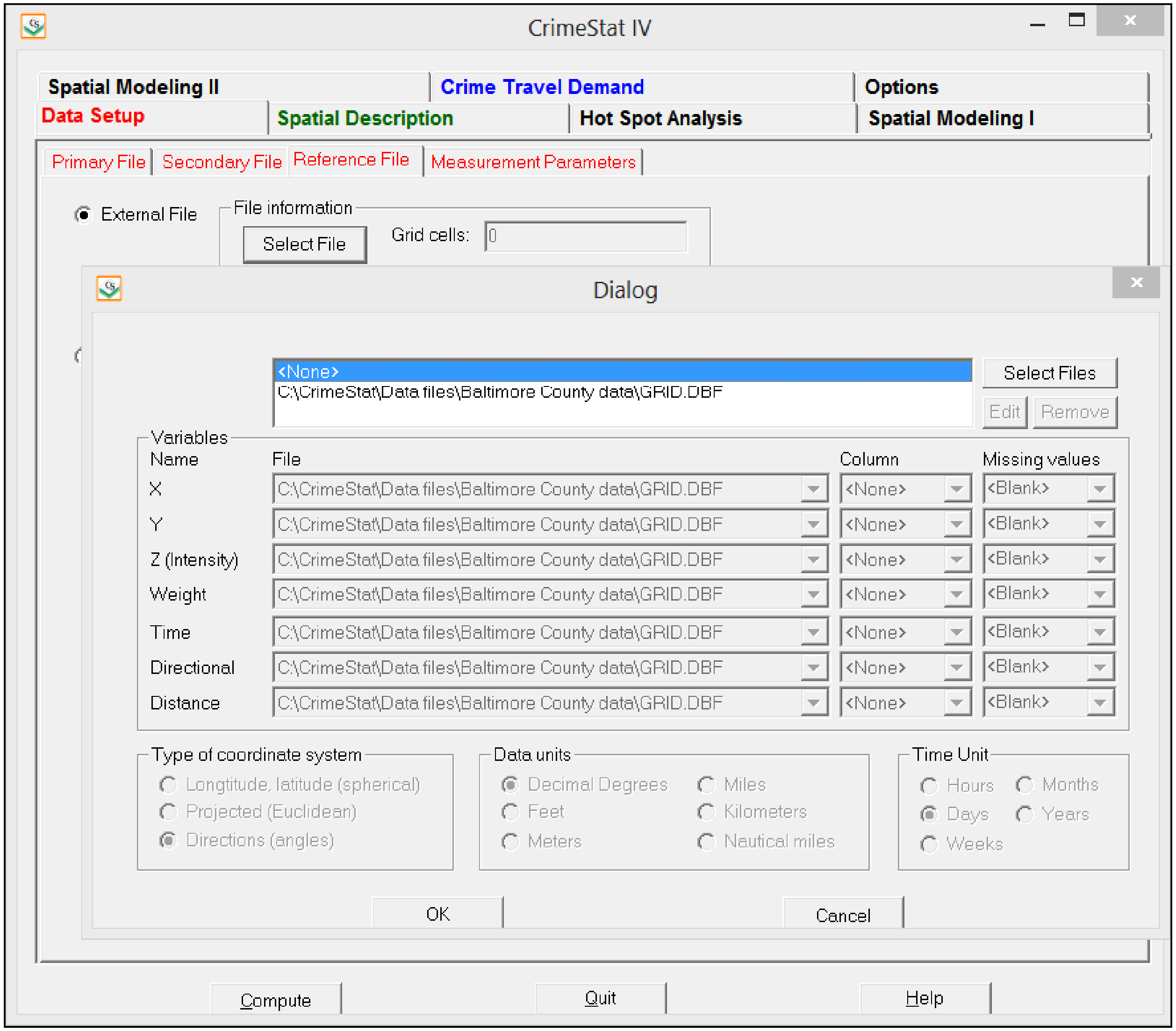
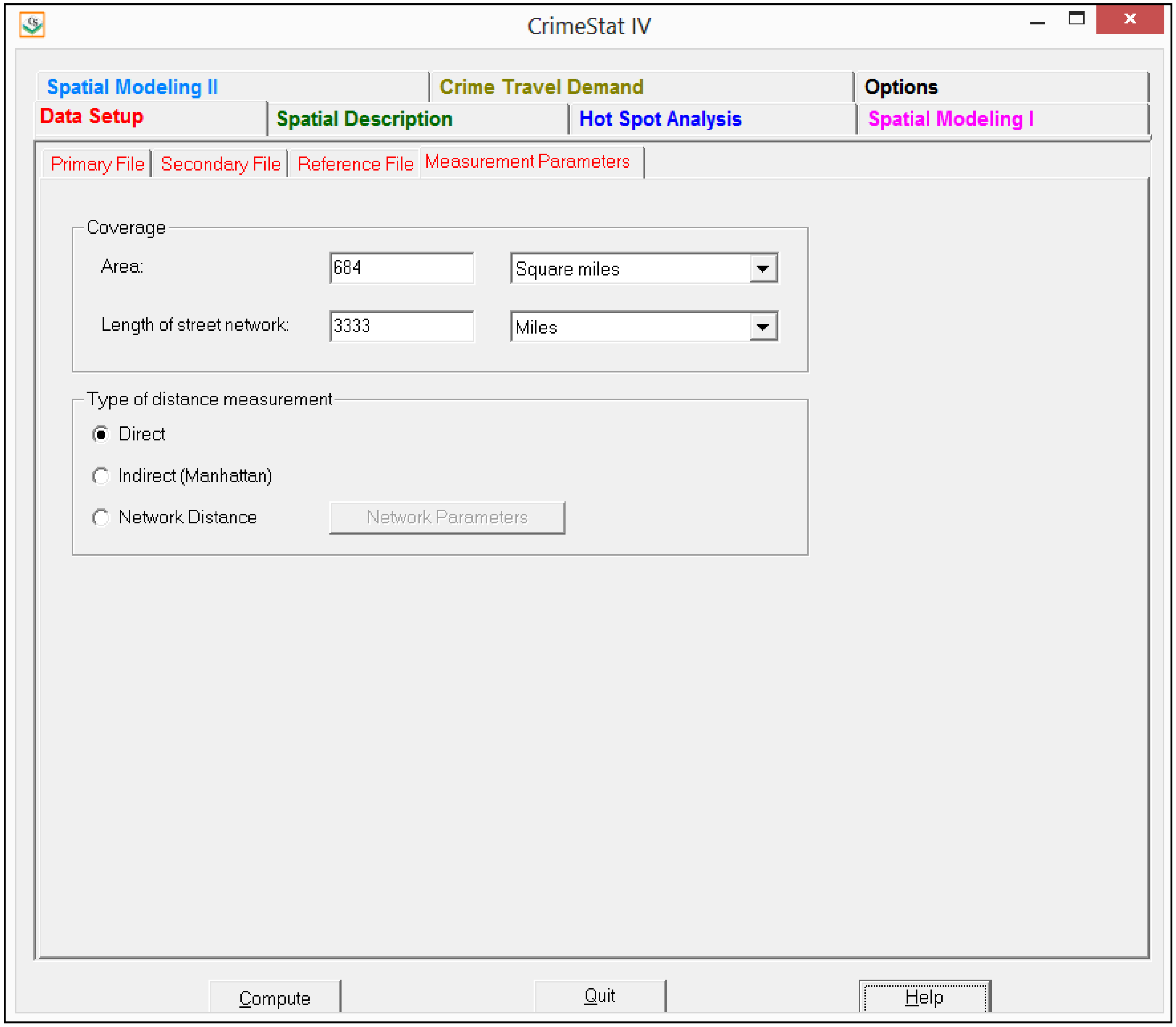


Figure 3.12:
Measurement Parameters Page



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

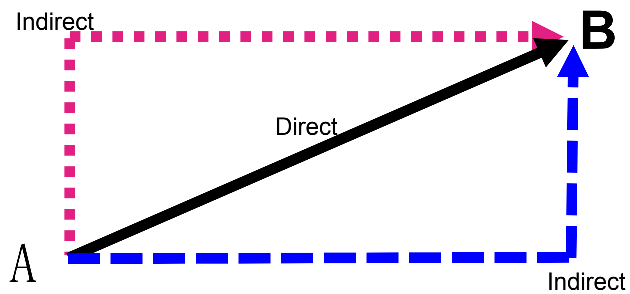
Indirect distance is an approximation to travel on a rectangular road network. This is frequently called *Manhattan* distance, referring to the grid-like structure of Manhattan. Many cities, but certainly not all, lay out their streets in grids. The degree to which this is true varies. Older cities will 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:

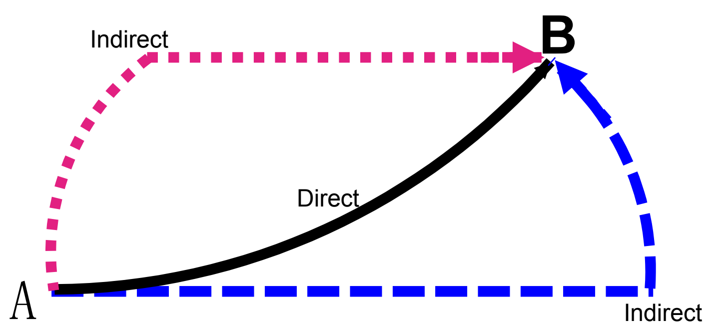
Direct and Indirect Distances

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:

$$d_{AB} = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2} \tag{3.1}$$

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:

$$Radians\ for\ latitude(\varphi) = \frac{2\pi\varphi}{360} \tag{3.2}$$

$$Radians\ for\ longitude(\lambda) = \frac{2\pi\lambda}{360} \tag{3.3}$$

Then, the distance between the two points is determined from:

$$d_{AB} = 2Arcsin\{Sin^2\left[\frac{(\varphi_B - \varphi_A)}{2}\right] + Cos\varphi_A Cos\varphi_B Sin^2[\frac{(\lambda_B - \lambda_A)}{2}]^{\frac{1}{2}}\} \qquad (3.4)$$

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:

$$d_{AB} = |X_A - X_B| + |Y_A - Y_B| \qquad (3.5)$$

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:

$$d_{AB} = \frac{d1_{AB} + d2_{AB}}{2} \qquad (3.6)$$

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 <u>not</u> assign this name to <u>both</u> 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.

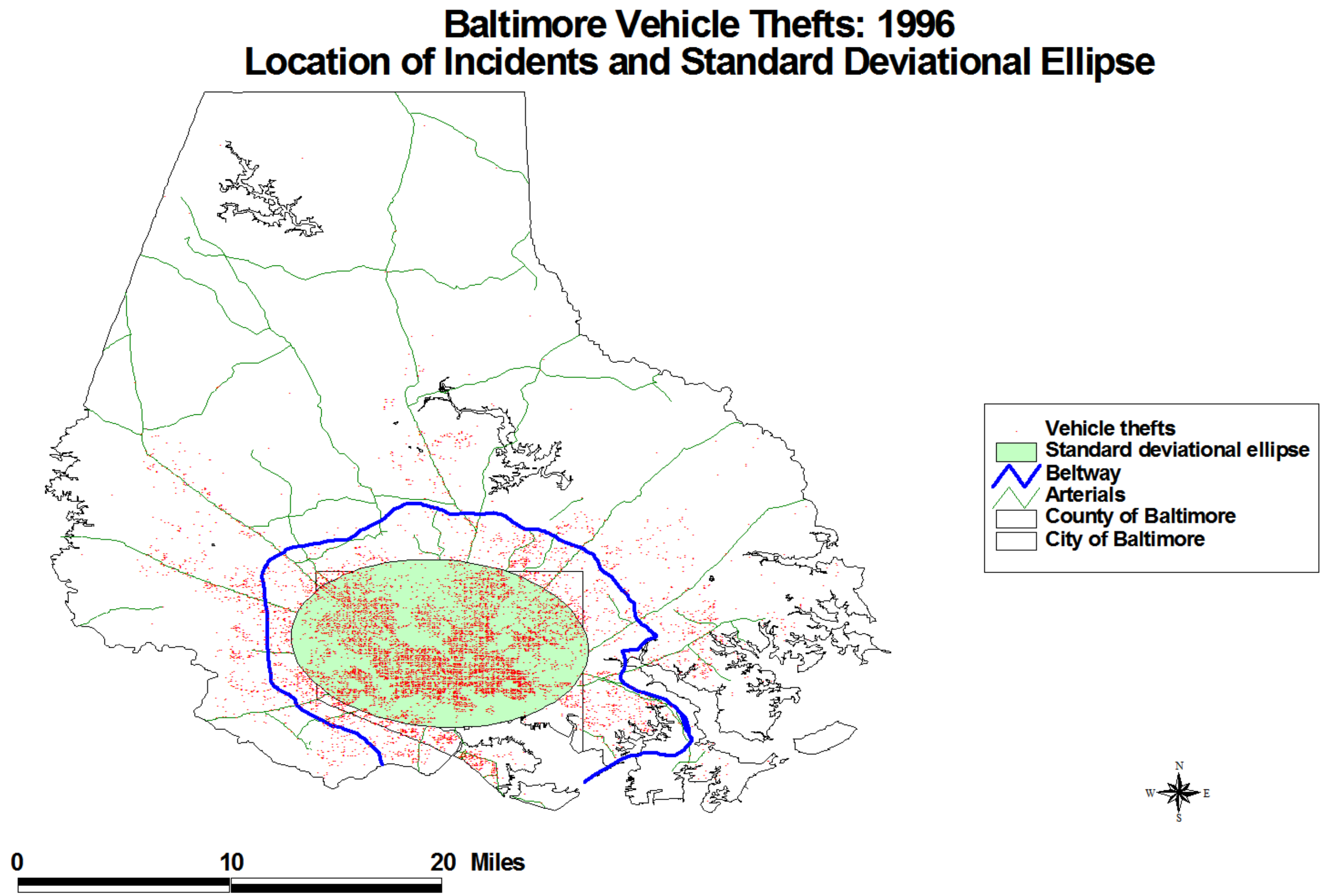
23. For example, Figure 3.14 shows an *ArcGIS®* map of 1996 vehicle thefts in Baltimore City and Baltimore County along with the standard deviational ellipse of the vehicle thefts, calculated with *CrimeStat*. *CrimeStat* outputs the ellipse as a shape file, which is then brought directly into *ArcGIS®*. 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
Vehicle thefts
Standard deviational ellipse
Beltway
Arterials
County of Baltimore
City of Baltimore
0
10
20 Miles
N
W
E
S

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.