CrimeStat Version 3.3 Update Notes:
Part 2: 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

July 2010

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Introduction[1]

The Regression I and Regression II modules are a series of routines for regression modeling and prediction. This update chapter will lay out the basics of regression modeling and prediction and will discuss the *CrimeStat* Regression I and II modules. The routines available in the two modules have also been applied to the Trip Generation model of the Crime Travel Demand module. Users wanting to implement that model should consult the documentation in this update chapter.

We start by briefly discussing the theory and practice of regression modeling with examples. Later, we will discuss the particular routines available 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 (Up. 2.1):

$$y_i = f(x_{1i}, \cdots, x_{Ki}) + \varepsilon_i \qquad \text{(Up. 2.1)}$$

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,

$$y_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_K x_{Ki} + \varepsilon_i \qquad \text{(Up. 2.2)}$$

---

[1] This chapter is a result of the effort of many persons. The maximum likelihood routines were produced by Ian Cahill of Cahill Software in Ottawa, Ontario 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. The programmer for the routines was Ms. Haiyan Teng of Houston, TX who ensured that they worked. 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. 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

$$z_k = \frac{x_k - \bar{x}_k}{std(x_k)} \qquad \text{(Up. 2.3)}$$

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 C 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

$$\sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{k=1}^{K} \beta_k x_{ki} \right) x_{ki} = 0 \qquad \text{(Up. 2.4)}$$

for *k=1* to *K* independent variables or, in matrix notation:

$$\mathbf{X}^T (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = 0 \qquad \text{(Up. 2.5)}$$

$$\mathbf{X}^T \mathbf{X}\boldsymbol{\beta} = \mathbf{X}^T \mathbf{y} \qquad \text{(Up. 2.6)}$$

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

$$\mathbf{b}_{OLS} = (b_0, b_1, \cdots, b_K)^T$$
$$\mathbf{b}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \qquad \text{(Up. 2.7)}$$

An estimate for the error variance follows as

$$s^2_{OLS} = \sum_{i=1}^{N}\left(y_i - b_0 - \sum_{k=1}^{K} b_k x_{ki} \right)^2 /(N-K-1) \qquad \text{(Up. 2.8)}$$

or, in matrix notation,

$$s^2_{OLS} = \mathbf{e}^T\mathbf{e}/(N-K-1) \qquad \text{(Up. 2.9)}$$

Maximum Likelihood Estimation

For the maximum likelihood method, the *likelihood* of a function is the joint probability density of a series of observations (Wikipedia, 2010b; 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:

$$f(x_1, x_2, \cdots, x_N \mid \boldsymbol{\theta}) = f(x_1 \mid \boldsymbol{\theta}) \times f(x_2 \mid \boldsymbol{\theta}) \times \cdots \times f(x_N \mid \boldsymbol{\theta}) \qquad \text{(Up. 2.10)}$$

and is called the ***likelihood*** function:

$$L(\boldsymbol{\theta} \mid x_1, x_2, \cdots, x_N) = f(x_1, x_2, \cdots, x_N \mid \boldsymbol{\theta}) = \prod_{i=1}^{N} f(x_i \mid \boldsymbol{\theta}) \qquad \text{(Up. 2.11)}$$

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,

$$\ln\left\{ \prod_{i=1}^{N} f(x_i \mid \boldsymbol{\theta}) \right\} = \ln\left[f(x_1 \mid \boldsymbol{\theta}) \right] + \ln\left[f(x_2 \mid \boldsymbol{\theta}) \right] + \cdots + \ln\left[f(x_n \mid \boldsymbol{\theta}) \right] \qquad \text{(Up. 2.12)}$$

This is called the ***Log likelihood*** function and is written as:

$$\ln L(\boldsymbol{\theta} \mid x_1, x_2, \cdots, x_N) = \sum_{i=1}^{N} \ln\left[f(x_i \mid \boldsymbol{\theta})\right] \quad \text{(Up. 2.13)}$$

For the OLS model, the log likelihood is:

$$\ln L = -\frac{N}{2}\ln(2\pi) - \frac{N}{2}\ln(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \mathbf{x}_i^T\boldsymbol{\beta})^2 \quad \text{(Up. 2.14)}$$

where N is the sample size and σ^2 is the variance. For the Poisson model, the log likelihood is:

$$\ln L = \sum_{i=1}^{N}\left[-\lambda_i + y_i \ln(\lambda_i) - \ln y_i!\right] \quad \text{(Up. 2.15)}$$

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 functions in Appendix C.

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, 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 C of the *CrimeStat* manual. More detail can be found in Hilbe (2008).

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 likelihoods 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 often 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 8 in the *CrimeStat* manual).

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 may produce distorted results.

Errors are Independent, Constant, and Normally-distributed

Second, the errors in the model, the ε in equation Up. 2.2, 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 of the 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 sometimes 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 importance 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. This is an important task in regression modeling.

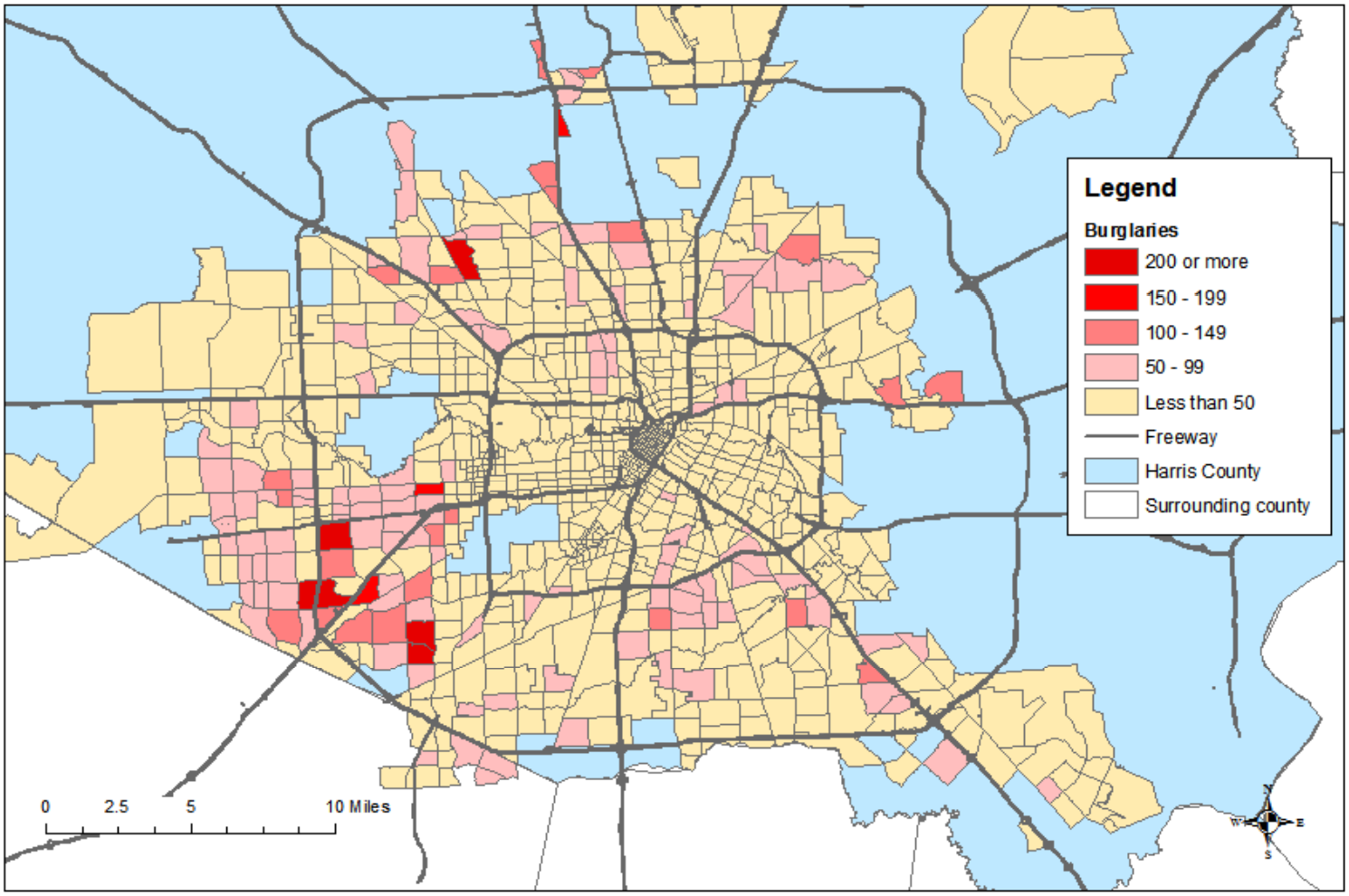
Example of Modeling Burglaries by Zones

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 Up. 2.1 show

Figure Up. 2.1:

Burglaries in the City of Houston

Number in Each Traffic Analysis Zone: 2006



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. The burglaries were then allocated to the 1,179 traffic analysis zones 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 Up. 2.2 show 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 show 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 had the majority of the burglaries, a result that is very typical of crime counts.

Example 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 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 Up. 2.1 present the results of the normal (OLS) model.

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, 1994, 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:

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \qquad \text{(Up. 2.16)}$$

Figure Up. 2.2:

Houston Burglaries in 2006:

Number of Burglaries Per Zone

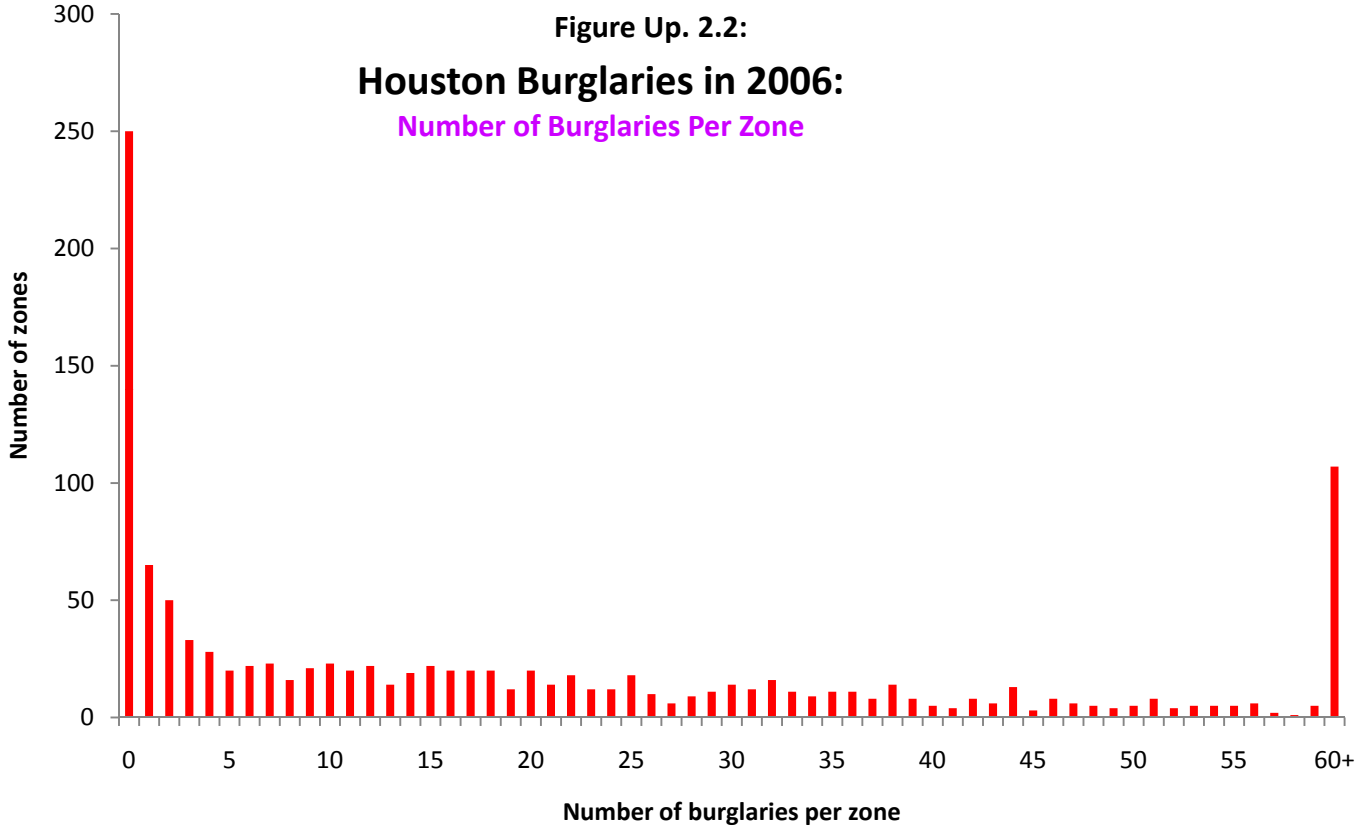


Table Up. 2.1:
Predicting Burglaries in the City of Houston: 2006
Ordinary Least Squares: Full Model
(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1,179
Df:	1,174
Type of regression model:	Ordinary Least Squares
F-test of model:	357.2 p≤.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	t-value	p
INTERCEPT	1	12.9320	1.269	-	10.19	0.001
HOUSEHOLDS	1	0.0256	0.0008	0.923	31.37	0.001
JOBS	1	-0.0002	0.0005	0.903	-0.453	n.s.
MEDIAN HOUSEHOLD INCOME	1	-0.0002	0.00003	0.970	-6.88	0.001

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

$$R_a^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2 / (N - K - 1)}{\sum (y_i - \bar{y})^2 / (N - 1)} \qquad \text{(Up. 2.17)}$$

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 variable 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 parameter included in the table is the *tolerance* of the coefficient. This is a measure of multicollinearity (or one type of overfitting). Basically, it is 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:

$$Tol_i = 1 - R^2_{j \neq i} \qquad \text{(Up. 2.18)}$$

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.

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 to (correlated with) 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. Later in the chapter, we will discuss multicollinearity and the general problem of overfitting in more detail.

Looking specifically at the model in Table Up. 2.1, 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:

$$\begin{aligned}(BURGLARIES)_i = {} & 12.93 + 0.0256(HOUSEHOLDS)_i - 0.0002(JOBS)_i \\ & - 0.0002(MEDIAN\ HOUSEHOLD\ INCOME)_i\end{aligned} \qquad \text{(Up. 2.19)}$$

To illustrate, TAZ 833 had 1762 households in 2006, 2,698 jobs also 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:

$$\begin{aligned}\text{Number of burglaries (TAZ833)} &= 12.93 + 0.0256*1762 - 0.0002*2{,}698 - 0.0002*27{,}500 \\ &= 52.0\end{aligned}$$

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:

$$\text{Residual error}_i = \text{Observed Value}_i - \text{Predicted value}_i \qquad \text{(Up. 2.20)}$$

Table Up. 2.2 give predicted values and residual errors for five of the observations from the Houston burglary data set.

Table Up. 2.2:
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

Analysis of the residual errors is one of the best tools for diagnosing problems with the model. A plot of the residual errors against the predicted values indicates 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 Up. 2.3 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.

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 than 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 Up. 2.4 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 Up. 2.1), typically the zones

Figure Up. 2.3:

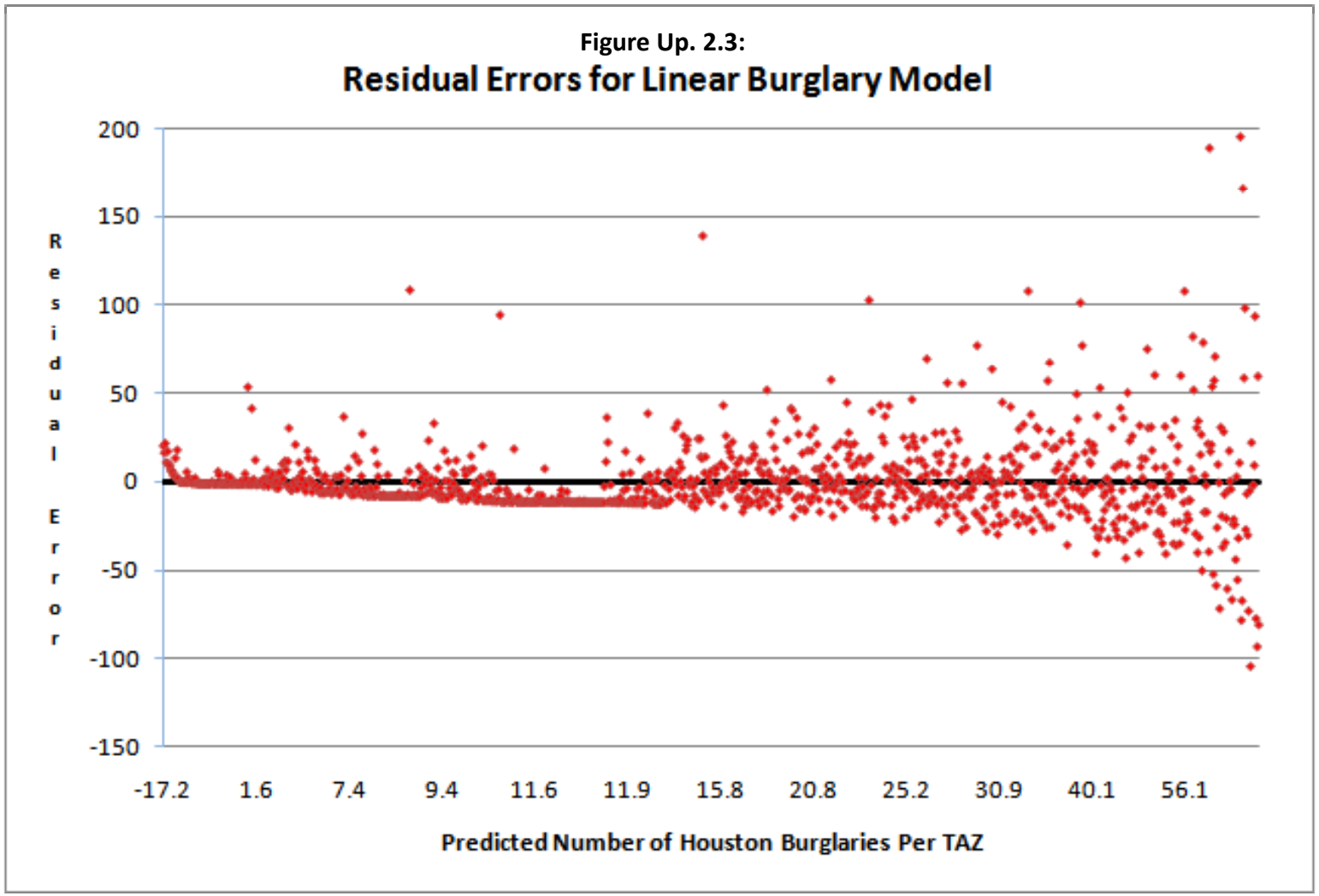
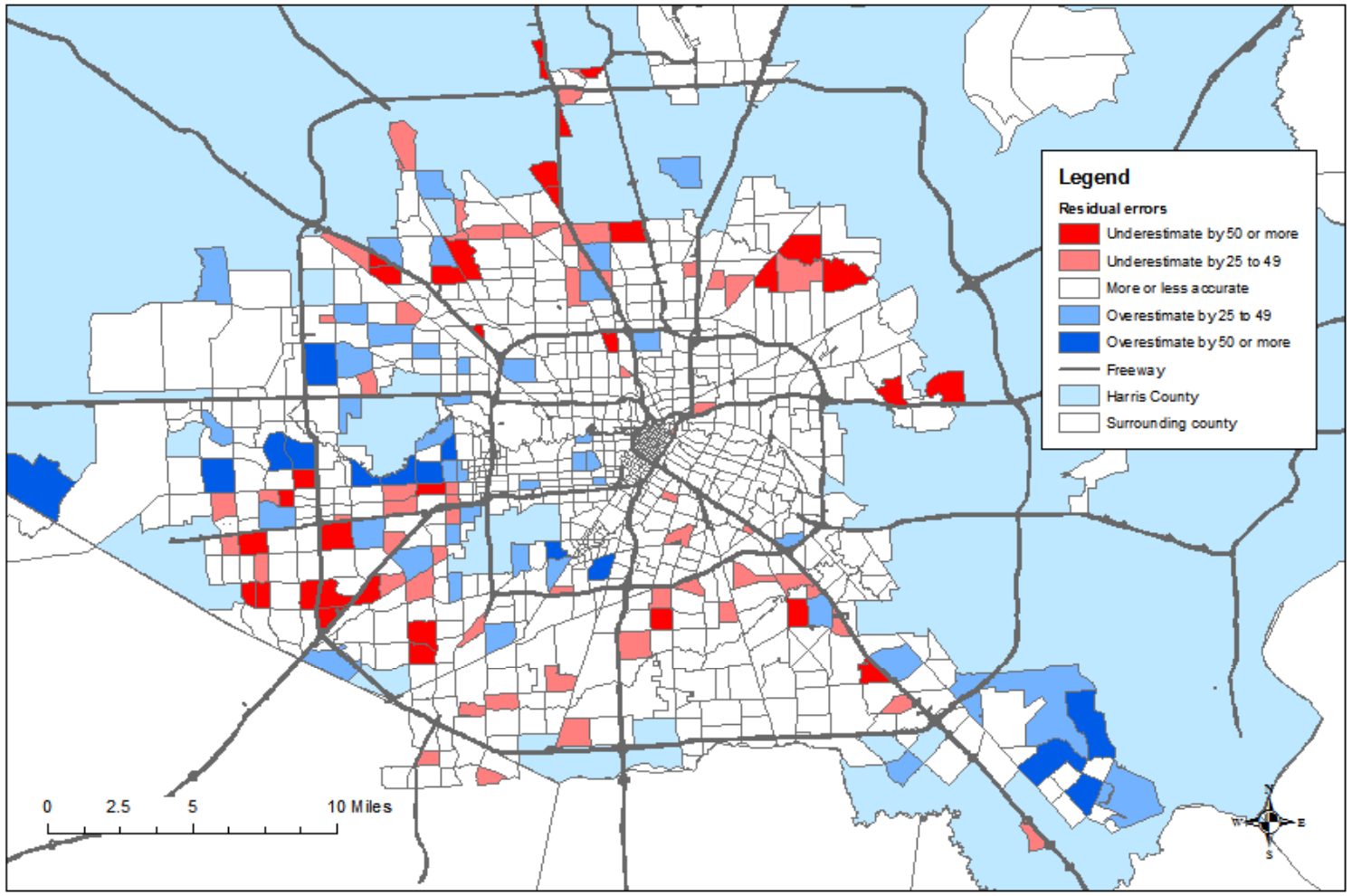

Residual Errors for Linear Burglary Model
Residual Error
200
150
100
50
0
-50
-100
-150
-17.2
1.6
7.4
9.4
11.6
11.9
15.8
20.8
25.2
30.9
40.1
56.1
Predicted Number of Houston Burglaries Per TAZ

Figure Up. 2.4:

Predicting Burglaries in the City of Houston: 2006

Residual Errors from Linear Model



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, 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.

---

[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 it is not intuitive nor easily explained to non-specialists.

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 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 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).

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 a little later in the chapter, 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 Up. 2.3 show the results. Comparing the results with Table Up. 2.1, 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).

Table Up. 2.3:
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≤.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	t-value	p
INTERCEPT	1	12.8099	1.240	-	10.33	0.001
HOUSEHOLDS	1	0.0255	0.0008	0.994	33.44	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.0002	0.00003	0.994	-7.03	0.001

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 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 less variable model.

For the Houston burglary example, the two remaining independent variables in Table Up. 2.3 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, later in the chapter in the discussion of the negative binomial model, 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. That is, one takes:

$$\ln y_i = \log_e (y_i) \qquad \text{(Up. 2.21)}$$

where e is the base of the natural logarithm (2.718…) and regresses the transformed dependent variable against the linear predictors,

$$\ln y_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_K x_{Ki} + \varepsilon_i \qquad \text{(Up. 2.22)}$$

This is equivalent to the equation

$$y_i = e^{\beta_0 + \beta_1 x_{1i} + \cdots + \beta_K x_{Ki} + \varepsilon_i} \qquad \text{(Up. 2.23)}$$

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 values of the dependent variable to a very small number (e.g., 0.001) for cases where the raw dependent variable actually has a value of 0. While this seems like a reasonable solution to the problem, it can lead to strange results. In the burglary data, for example, there were 250 zones (out of 1,179 or 21%) that had zero burglaries!

Example of Transforming Dependent Variable

To illustrate, we transformed the dependent variable in the above example – 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 Up. 2.3 above). Table Up. 2.4 present the results:

Table Up. 2.4:
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≤.0001
R-square:	0.42
Adjusted r-square:	0.42
Mean absolute deviation:	30.7
1st (highest) quartile:	96.9
2nd quartile:	18.2
3rd quartile:	3.3
4th (lowest) quartile:	4.6
Mean squared predictive error:	30,357.4
1st (highest) quartile:	118,774.1
2nd quartile:	2850.2
3rd quartile:	36.7
4th (lowest) quartile:	58.9

Predictor	DF	Coefficient	Stand Error	Tolerance	t-value	p
INTERCEPT	1	1.5674	0.067	-	23.44	0.001
HOUSEHOLDS	1	0.0012	0.00004	0.994	28.84	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.000002	0.000001	0.994	-4.09	0.001

The coefficients are similar in sign. The R^2 value is smaller than the untransformed model (0.42 compared to 0.48). However, the mean squared predictive error is now much higher than the original raw values (30,357.42 compared to 505.14) and the mean absolute deviation is also much higher (30.73 compared to 13.50).[3]

[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.

In other words, transforming the dependent to a natural log has not improved the overall normal model and, in fact, worsened the predictability. The high degree of skewness in the dependent variable was not eliminated by transforming it.

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:

$$z_k = \frac{x_k - \bar{x}_k}{std(x_k)} \qquad \text{(Up. 2.24)}$$

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. Essentially, skewness is a fundamental property of a distribution and the normal model is poorly suited for modeling it.

Count Data Models

In short, a normal linear model is 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 normal 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:

$$\text{Prob}(y_i) = \frac{e^{\lambda}\lambda^{y_i}}{y_i!} \qquad \text{(Up. 2.25)}$$

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 Up.2.5 illustrates the Poisson distribution for different expected means (repeated from chapter 13).

The mean can, in turn, 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:

$$E(y_i \mid \mathbf{x}_i) = \lambda_i = e^{\mathbf{x}_i^T \boldsymbol{\beta}} \qquad \text{(Up. 2.26)}$$

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 Up. 2.26 is sometimes written as

$$\ln(\lambda_i) = \mathbf{x}_i^T \boldsymbol{\beta} = \beta_0 + \sum_{k=1}^{K} \beta_k x_{ki} \qquad \text{(Up. 2.27)}$$

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, especially if the independent variables are categories, rather than continuous (real) variables. However, we will refer to it as a Poisson model. In more familiar notation, this is

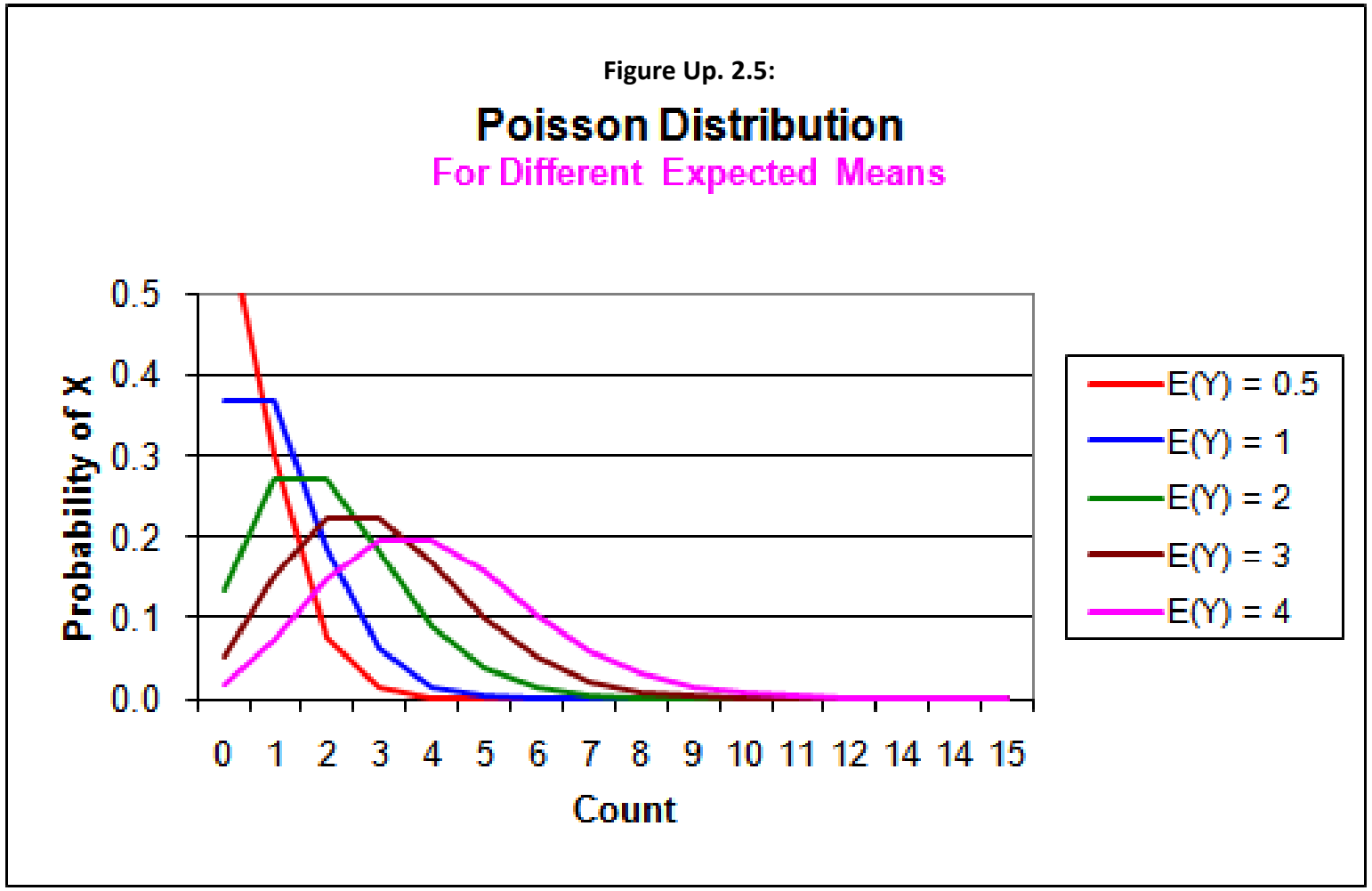
$$\ln(\lambda_i) = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_K x_{Ki} \qquad \text{(Up. 2.28)}$$

That is, the natural log of the mean is a function of *K* independent variables and an intercept.

The data are assumed to reflect the Poisson model. Also, in the Poisson model, 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 very different than a normal model where the residual errors are expected to be constant.

The model is estimated using a maximum likelihood procedure, typically the Newton-Raphson method or, occasionally, using Fisher scores (Wikipedia, 2010a; Cameron & Trivedi, 1998). In Appendix C, Anselin presents a more formal treatment of both the normal and Poisson regression models including the methods by which they are estimated.

Figure Up. 2.5:


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

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 a 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 zones 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.

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, we estimate a Poisson model of the two independent predictors of burglaries (Table Up. 2.5).

Likelihood Statistics

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 it maximizes is a probability and is always less than 1.0, the log likelihood is always negative with a Poisson model.

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:

$$AIC = -2L + 2(K+1) \quad \text{(Up. 2.29)}$$

where L is the log likelihood and K is the number of independent variables. Third, another measure which is very similar is the *Bayes Information Criterion* (or *Schwartz Criterion*), which is defined as:

$$BIC/SC = -2L+[(K+1)\ln(N)] \quad \text{(Up. 2.30)}$$

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 AIC or BIC/SC values are 'best'.

Fourth, a decision about whether the Poisson model is appropriate can be based on the statistic called the deviance which is defined as:

$$Dev = 2(L_F - L_M) = 2\sum_{i=1}^{N}\left[y_i \ln\left(\frac{y_i}{\hat{\lambda}_i}\right) - y_i - \hat{\lambda}_i \right] \qquad \text{(Up. 2.31)}$$

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 overdispersed due to missing variables or non-Poisson form.

Fifth, there is the Pearson chi-square statistic which is defined by

$$Pearson - \chi^2 = \sum_{i=1}^{N}\frac{(y_i - \hat{\lambda}_i)^2}{\hat{\lambda}_i} \qquad \text{(Up. 2.32)}$$

and is approximately chi-square distributed with mean $N-(K+1)$ for a valid Poisson model. Therefore, if the Pearson chi-square statistic divided by degrees of freedom, $Pearson-\chi^2/(N-K-1)$ is significantly larger than 1, overdispersion is also indicated.

Model Error Estimates

Next, there are two statistics that measure how well the model fits the data, or goodness-of-fit. In *CrimeStat*, there are two statistics that measure goodness-of-fit, the Mean Absolute Deviation (MAD) and Mean Squared Predicted Error (MSPE) which were defined above (p. Up. 2.11). Comparing these with the normal model, 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 Up.2.4). Comparing the four quartiles, it can be seen that three of the four quartiles for the normal model have slightly better MAD and MSPE scores than for the Poisson but the differences are not great.

Table Up. 2.5:
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-value of deviance:	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
Over-dispersion tests	
Adjusted deviance:	19.6
Adjusted Pearson Chi-Square:	21.1
Dispersion multiplier:	21.1
Inverse dispersion multiplier:	0.05

Predictor	DF	Coefficient	Stand Error	Tolerance	Z-value	p
INTERCEPT	1	2.8745	0.014	-	212.47	0.001
HOUSEHOLDS	1	0.0006	0.000004	0.994	146.24	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.000009	0.00000	0.994	-28.68	0.001

Over-dispersion Tests

The remaining four summary statistics measure *dispersion*. A more extensive discussion of dispersion is given a little later in the chapter. But, very simply, in the Poisson framework, the variance equals 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 degress of freedom; 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 $Var(y_i) = \lambda_i + \gamma\lambda_i^2$. 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 can be seen in Table Up. 2.5, 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 (a ratio of 5.1:1 compared to 4.8:1 for the normal model).

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 more sound. 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 Up. 2.6 show 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. Most real-world count data are similar to this; the variance will usually be much greater than the mean (Lord et al., 2005). 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).

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; 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:

$$\omega_i = V[y_i \mid \mathbf{x}_i] \qquad \text{(Up. 2.33)}$$

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:

$$\omega_i = \lambda_i + \tau\lambda_i^p \qquad \text{(Up. 2.34)}$$

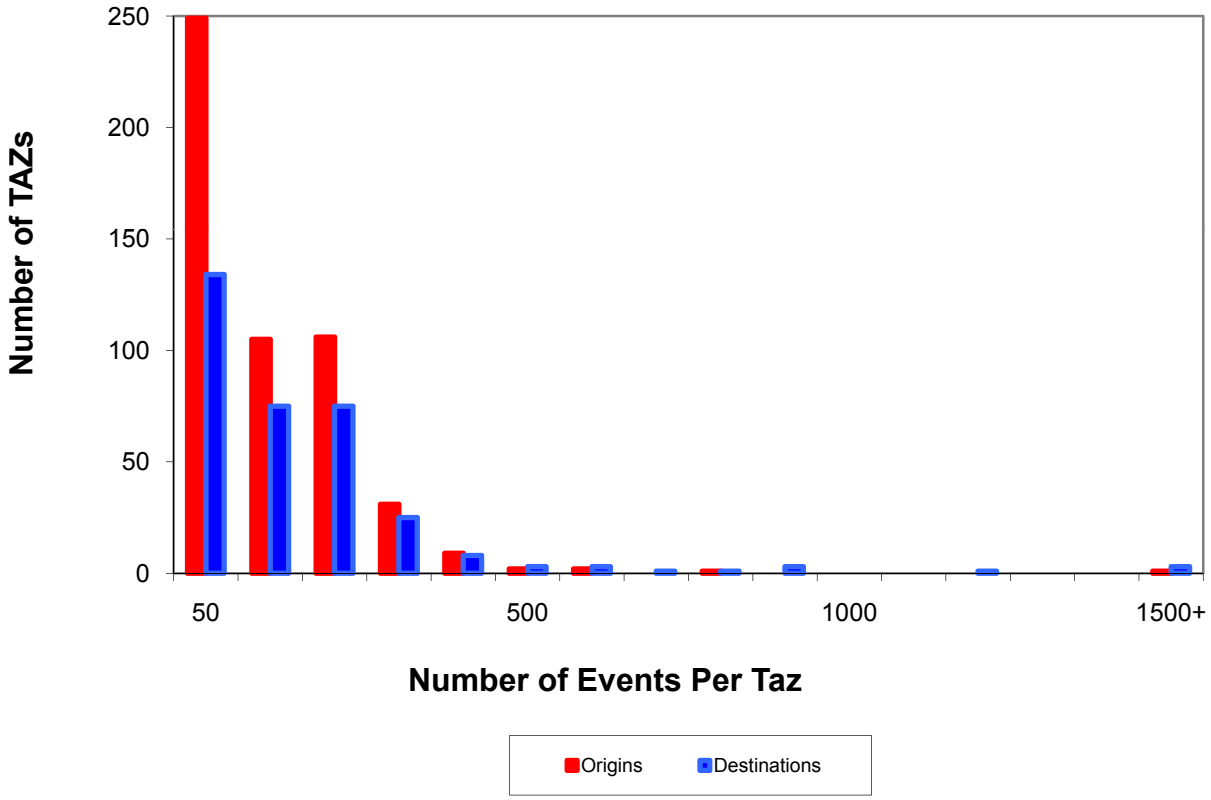
where τ is the *dispersion parameter* and p is a constant (usually 1 or 2). In the case where p is 1, the equation simplifies to:

$$\omega_i = \lambda_i + \tau\lambda_i \qquad \text{(Up. 2.35)}$$

This is the NB1 correction. In the special case where $\tau = 0$, the variance becomes equal to the mean (the Poisson model).

Figure Up. 2.6:

Distribution of Crime Origins and Destinations: Baltimore County, MD: 1993-1997



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:

$$\hat{\tau} = 1/\hat{\psi} = \frac{1}{N-K-1}\sum_{i=1}^{N}\frac{(y_i - \hat{\lambda}_i)^2}{\hat{\lambda}_i} \qquad \text{(Up. 2.36)}$$

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 $\hat{\lambda}_i$ 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 $(y_i - \hat{\lambda}_i)^2$ 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 1.0 indicates over-dispersion while a value less than 1 indicates under-dispersion (which is rare, though possible). A value of 1.0 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*:

$$SE_{adj} = SE \times \sqrt{\hat{\tau}} \qquad \text{(Up. 2.37)}$$

The new standard error is then used in 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 Up. 2.6 show 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 Up. 2.5) 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.

Table Up. 2.6:
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-value of deviance:	0.0001
Pearson Chi-square:	12,402.2
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:	2351.8
2nd quartile:	203.7
3rd quartile:	99.8
4th (lowest) quartile:	206.7
Over-dispersion tests	
Adjusted deviance:	10.5
Adjusted Pearson Chi-Square:	10.6
Dispersion multiplier:	1.0
Inverse dispersion multiplier:	1.0

Predictor	DF	Coefficient	Stand Error	Tolerance	Z-value	p
INTERCEPT	1	2.87452	0.062	-	46.26	0.001
HOUSEHOLDS	1	0.00059	0.00002	0.994	31.84	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.000009	0.000001	0.994	-6.24	0.001

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 with a mean (λ_i) but the dispersion is assumed to follow a Gamma distribution (Lord, 2006; Cameron & Trivedi, 1998, 62-63; Venables and Ripley, 1997, 242-245).

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 a). For our purposes, it is defined as a mixed distribution with a Poisson mean and a one parameter Gamma dispersion function having the form

$$f(y_i / \theta_i) = \frac{e^{-\theta_i}\theta_i^{y_i}}{y_i!} \qquad \text{(Up. 2.38)}$$

where

$$\theta_i = e^{\beta_0 + (\Sigma \beta_i x_i) + \varepsilon_i} \qquad \text{(Up. 2.39)}$$

$$\theta_i = e^{\beta_0 + (\Sigma \beta_i x_i)} e^{\varepsilon_i} \qquad \text{(Up. 2.40)}$$

$$\theta_i = \mu_i \nu_i \qquad \text{(Up. 2.41)}$$

and where θ_i is a function of a one-parameter gamma distribution where the parameter, τ, is greater than 0 (ignoring the subscripts)

$$h(y / \mu, \tau) = \frac{\Gamma(\tau^{-1} + y)}{\Gamma(\tau^{-1})\Gamma(y+1)} \left(\frac{(\tau^{-1})}{\tau^{-1} + \mu}\right)^{\alpha^{-1}} \left(\frac{\mu}{\tau^{-1} + \mu}\right)^{y} \qquad \text{(Up. 2.42)}$$

The model has been applied traditionally to 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, 2010a; 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 such that different observations come 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, they call 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:

$$\omega_i = \mu_i + \mu_i^2 / \psi \qquad \text{(Up. 2.43)}$$

whereupon ψ (Psi) became known as the *inverse dispersion parameter* (McCullagh and 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:

$$\omega_i = \mu_i + \tau \mu_i^2 \qquad \text{(Up. 2.44)}$$

where the variance is now a quadratic function of the Poisson mean (i.e., p is 2 in formula Up. 2.34) and τ is called the *dispersion multiplier*. This is the formulation proposed by Cameron & Trivedi (1998; 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 is then of a Poisson mean but with a 'longer tail' variance function. The dispersion parameter, τ, is now 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:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \qquad \text{(Up. 2.45)}$$

The Poisson mean λ_i is organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i) \qquad \text{(Up. 2.46)}$$

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:

$$D = \sum_{i=1}^{N} \left[y_i \ln\left(\frac{y_i}{\hat{\lambda}_i}\right) - (y_i + \hat{\psi}) \ln\left(\frac{y_i + \hat{\psi}}{\hat{\lambda}_i + \hat{\psi}}\right) \right] \qquad \text{(Up. 2.47)}$$

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 Up. 2.7 present 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 slightly higher than for the Poisson, 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 more sound and does better with highly skewed (over-dispersed) data.

Example 2 of Negative Binomial Regression with Highly Skewed Data

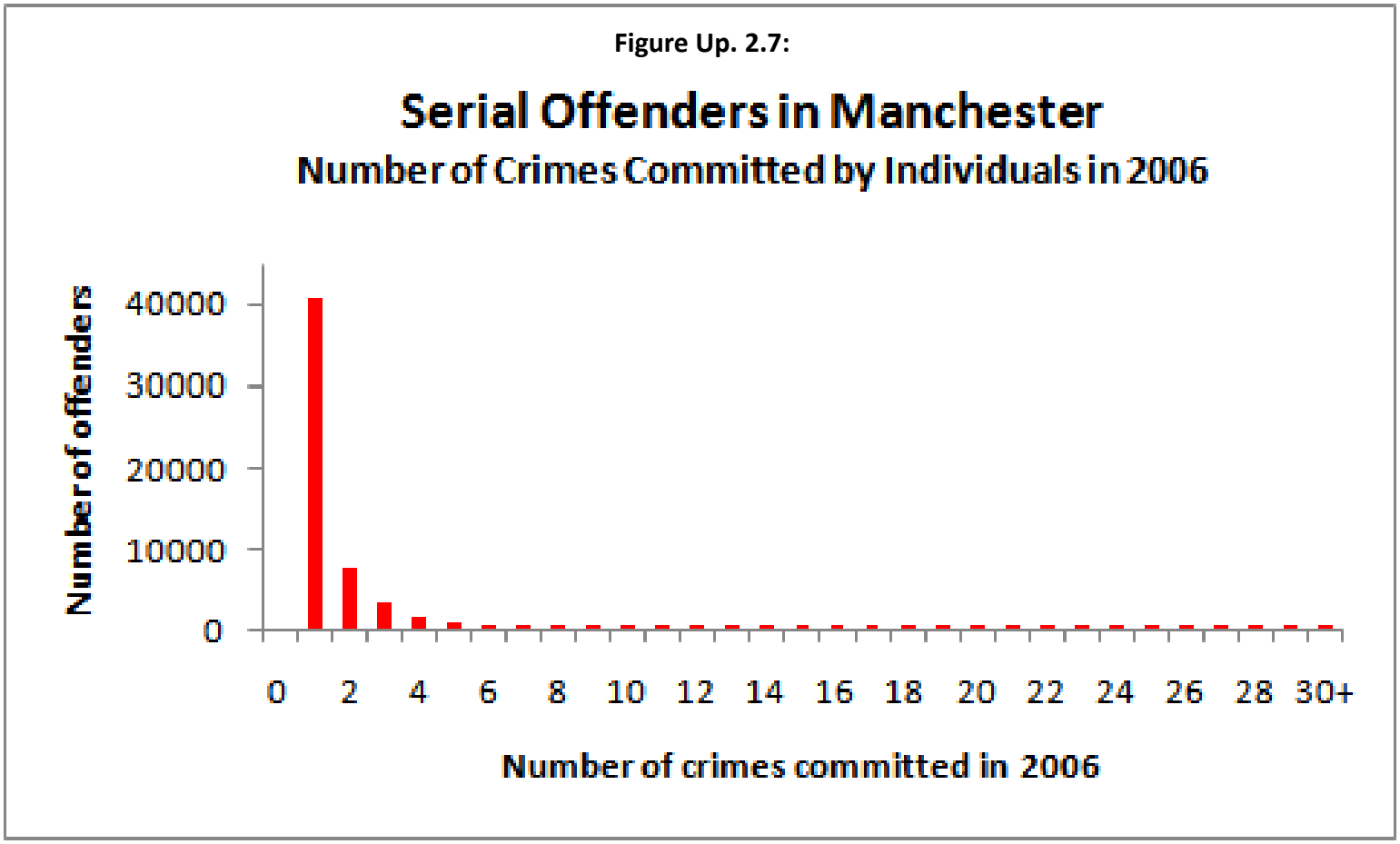
To illustrate further, the negative binomial is very useful when the dependent variable is extremely skewed. Figure Up. 2.7 show the number of crimes committed (and charged for) by individual offenders in Manchester, England in 2006. The X-axis plots the number of crimes committed while the Y-axis plots 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.

Table Up. 2.7:
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-value of deviance:	0.0001
Pearson Chi-square:	1,112.7
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
Over-dispersion tests	
Adjusted deviance:	1.2
Adjusted Pearson Chi-Square:	0.9
Dispersion multiplier:	1.5
Inverse dispersion multiplier:	0.7

Predictor	DF	Coefficient	Stand Error	Tolerance	Z-value	p
INTERCEPT	1	2.3210	0.083	-	27.94	0.001
HOUSEHOLDS	1	0.0012	0.00007	0.994	17.66	0.001
MEDIAN HOUSEHOLD INCOME	1	-0.00001	0.000002	0.994	-5.13	0.001

Figure Up. 2.7:


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

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 Up. 2.8 show the results.

The model was discussed in a recent article (Levine & Lee, 2010). 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 fit quite 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 overdispersion 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 maintain this. 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.

Table Up. 2.8:
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-value of deviance:	0.0001
Pearson Chi-square:	80,950.2
Model error estimates	
Mean absolute deviation:	0.9
1st (highest) quartile:	1.9
2nd quartile:	0.7
3rd quartile:	0.6
4th (lowest) quartile:	0.6
Mean squared predicted error:	3.9
1st (highest) quartile:	13.8
2nd quartile:	0.7
3rd quartile:	0.6
4th (lowest) quartile:	0.6
Over-dispersion tests	
Adjusted deviance:	0.6
Adjusted Pearson Chi-Square:	1.4
Dispersion multiplier:	0.2
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

Alternative Regression Models

Another disadvantage is related to the small sample size and low sample mean bias. It has been shown that the dispersion parameter of NB models can be significantly biased or misestimated when not enough data are available for estimating the model (Lord, 2006).

There are a number of alternative MLE methods for estimating the likely value of a count given a set of independent predictors. 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), and the lognormal function (see NIST 2004 for a list of common non-linear functions).

Limitations of the Maximum Likelihood Approach

The functions considered up to this point are part of the single-parameter exponential family of functions. 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) 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, a phenomenon.

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). The problem is that these corrections only partially adjust the model. It is still possible to overfit a model. 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. The result is that there is a reasonable solution to the problem rather than an exact one.

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 and McGuire, 2002).

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 9 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, and Rubin, 2004; Lee, 2004; Denison, Holmes, Malick and Smith, 2002; Carlin and Louis, 2000; Leonard and 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, Mount 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, 2010c; Cormen, Leiserson, Rivest, & Stein; 2009; So, Ye, & Zhang, 2007; Dijkestra, 1959).

This strategy can be useful if there is a single mountain to climb. 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. Maximum likelihood is similar to this in that it requires a smooth function for which each step upward is assumed to be climbing the mountain. For functions that are smooth, such as the single-parameter exponential family, such an algorithm will work very well.

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).

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 (sampling, if you wish) in order to find a path that really does go up to the peak.

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 – OLS, Poisson, negative binomial, binomial probit, and other, 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 optima) 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 Update chapter 2.1. 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:

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} \qquad \text{(Up. 2.48)}$$

or

$$P(B \mid A) = \frac{P(A \cap B)}{P(A)} \qquad \text{(Up. 2.49)}$$

Bayes Theorem relates the two equivalents of the 'and' condition together.

$$P(B)\times P(A\mid B)=P(A)\times P(B\mid A) \qquad \text{(Up. 2.50)}$$

$$P(A\mid B)=\frac{P(A)\times P(B\mid A)}{P(B)} \qquad \text{(Up. 2.51)}$$

or

$$P(B\mid A)=\frac{P(B)\times P(A\mid B)}{P(A)} \qquad \text{(Up. 2.52)}$$

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:

$$P(\theta\mid X)=\frac{P(X\mid\theta)\times P(\theta)}{P(X)} \qquad \text{(Up. 2.53)}$$

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 experimental or from 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

$$P(\theta\mid X)\propto P(X\mid\theta)\times P(\theta) \qquad \text{(Up. 2.54)}$$

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 systematically. The formula requires that a prior probability 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, and Smith, 2002; Lee, 2004; Gelman, Carlin, Stern, and 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 sufficiently large. An MCMC algorithm, on the other hand, is iterative with the generation of the next sample depending on the value of the current sample. The algorithm requires us to sample until convergence has been obtained. Since the initial values of an MCMC algorithm are usually chosen arbitrarily and samples generated from one iteration to the next are correlated (autocorrelation), 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 Up. 2.54.

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 a multi-dimensional distribution. The parameters are, instead, sampled sequentially from their full conditional distributions, one at a time holding all existing parameters constant (e.g. Gibbs sampling). 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 as a 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 are complex, of course, and involve some detail. The following represents a brief discussion of the steps. In Appendix D, Dominique Lord 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 two types of model:

1. **Poisson-Gamma Model**. This is similar to the negative binomial model discussed above except that it is estimated by MCMC rather than by MLE. Formally, it is defined as:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \qquad \text{repeat (Up. 2.45)}$$

The Poisson mean λ_i is organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i) \qquad \text{repeat (Up. 2.46)}$$

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).

In the Bayesian approach, prior probabilities have to be assigned to all unknown parameters, $\boldsymbol{\beta}$ and ψ. It is usually assumed that the β_k coefficients follow a *multivariate normal* distribution with $k+1$ dimensions:

$$\boldsymbol{\beta} \sim MVN_{k+1}(\mathbf{b}_0, \mathbf{B}_0) \qquad \text{(Up. 2.55)}$$

where MVN_{k+1} indicates a multivariate normal distribution with $k+1$ dimension, and $\mathbf{b}_0$ and $\mathbf{B}_0$ are *hyper-parameters* (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.

2. **Poisson-Gamma-Conditional Autoregressive (CAR) Model**. This is the negative binomial model but with a spatial autocorrelation term. Formally, it is defined as:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \quad \text{(Up. 2.56)}$$

with the mean of Poisson-Gamma-CAR organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i + \phi_i) \quad \text{(Up. 2.57)}$$

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.

Spatial Component

There are two common ways to express the spatial component, either as a CAR or as a Simultaneous Autoregressive (SAR) function (De Smith, Goodchild, & Longley, 2007). The CAR model is expressed as:

$$E(y_i \mid y_{j \neq i}) = \mu_i + \rho \sum_{j \neq i} w_{ij} (y_j - \mu_j) \quad \text{(Up. 2.58)}$$

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. 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:

$$E(y_i \mid y_{j \neq i}) = \mu_i + \rho \sum_{j \neq i} w_{ij} y_j \quad \text{(Up. 2.59)}$$

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 this version of *CrimeStat*, we will only utilize the CAR model. We will add the SAR model to the next version.

Step 2: Setting up a Likelihood Function

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:

$$L = \ln\left(\prod_{i=1}^{n} \frac{e^{-(\lambda_i)}(\lambda_i)^{y_i}}{y_i!}\right) = \sum_{i=1}^{n}\left[\lambda_i + y_i \ln(\lambda_i) - \log\Gamma(y_i + 1)\right] \qquad \text{(Up. 2.60)}$$

with y_i being the observed (actual) value of the dependent variable and λ_i being the posterior mean of each site.

For the Poisson-Gamma-CAR model, the log likelihood function is the same. The only difference is that, for the Poisson-Gamma, the posterior mean is based on

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i) \qquad \text{repeat (Up. 2.46)}$$

while for the Poisson-Gamma-CAR model, the posterior mean is based on:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i + \phi_i) \qquad \text{repeat (Up. 2.57)}$$

Step 3: Defining a Joint Posterior Distribution

In the case of the Poisson-Gamma model, the posterior probability, $p(\boldsymbol{\lambda}, \boldsymbol{\beta}, \psi \mid \mathbf{y})$, of the joint posterior distribution is defined as:

$$\pi(\boldsymbol{\lambda}, \boldsymbol{\beta}, \psi \mid \mathbf{y}) \propto f(\mathbf{y} \mid \boldsymbol{\lambda}) \cdot \pi(\boldsymbol{\lambda} \mid \boldsymbol{\beta}, \psi) \cdot \pi(\beta_1) \cdots \pi(\beta_J) \cdot \pi(\psi) \qquad \text{(Up. 2.61)}$$

and is not in standard form (Park, 2009). Note that this is a general formulation. The parameters of interests are $(\lambda_1, \cdots \lambda_n), (\beta_1, \cdots \beta_J),$ and ψ. Since it is difficult to draw samples of the parameters from the joint posterior distribution, we usually draw samples of 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. As mentioned, the user can substitute more precise values for the expected value of the coefficients (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.[4]

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 Metropolis and his colleagues and improved by Hastings. 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, 2008). The basic algorithm consists of six steps (Train, 2009; Lynch, 2008; 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.
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 or a Poisson-Gamma-CAR function). 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

[4] 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, the conditional probabilities were not easily defined. Consequently, we have 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).

distribution though usually the normal is used. Also, usually the distribution is symmetric though the algorithm can work for non-symmetric proposal distributions, too (see Lynch, 2008, 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:

$$\text{Acceptance probability} = R = \frac{f(\theta_C) * g(\theta_E)}{f(\theta_E) * g(\theta_C)} \qquad \text{(Up. 2.62)}$$

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 (2008) calls it the *importance ratio* since the ratio will be greater than 1 if the candidate value yields a higher density 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 in, for example, 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 draws 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 will not necessarily be accepted if it goes up since it is compared with a random 'step'. Thus, the climber has to explore 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 he/she can utilize that information to ensure that the algorithm starts at a likely place. Lynch (2008) 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 can 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, all the results from the MCMC sample are 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.

Before we discuss some of the subtleties of the method, such as how many samples to draw and how many samples to discard before equilibrium has been established (burn in), let us illustrate this with the example that we have been using in this chapter.

The MCMC algorithm for the Poisson-Gamma (negative binomial) model was run on the Houston burglary dataset. The total number of iterations that were run was 25,000 with the initial 5,000

Table Up. 2.9:
Predicting Burglaries in the City of Houston: 2006
MCMC Poisson-Gamma Model
(N= 1,179 Traffic Analysis Zones)

DepVar:	**2006 BURGLARIES**
N:	1179
Df:	1175
Type of regression model:	Poisson - Gamma
Method of estimation:	MCMC
Number of iterations:	2500 Burn in: 5000
Likelihood statistics	
Log Likelihood:	-4430.8
DIC:	10,105.5
AIC:	8,869.6
BIC/SC:	8,889.9
Deviance:	1,387.5
p-value of deviance:	0.0001
Pearson Chi-Square:	1,106.4
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
Over-dispersion tests	
Adjusted deviance:	1.2
Adjusted Pearson Chi-Square:	0.9
Dispersion multiplier:	1.5
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≤.001

being discarded (the 'burn in' period). In other words, the results are based on the final 20,000 samples. Table Up. 2.9 show the results.

First, there are convergence statistics indicating whether the algorithm converged. They do this by comparing chains of estimated values for parameters, either with themselves or with the complete series. 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 of approximately *m* iterations each where *m* is the square root of *M-L*.

Let:

$$Mean\theta_K = (\sum_i \theta_i) / K \tag{Up. 2.63}$$

$$Mean\theta_M = (\sum_m \theta_m) / m \tag{Up. 2.64}$$

and

$$MCError = \frac{\sqrt{Mean\theta_K - Mean\theta_M}}{m(m-1)} \tag{Up. 2.65}$$

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 Up. 2.9.

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 (Lynch, 2008; Carlin & Louis, 2000). 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

$$G-R = \sqrt{\left(\frac{m+1}{m}\right) * \left(\frac{n-1}{n} + \frac{B}{W}\right) - \left(\frac{n-1}{mn}\right)} \tag{Up. 2.66}$$

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, 2008; 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). In the example above, they 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 (Table Up. 2.7). 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 table also includes a new summary statistic, the Deviance Information Criterion (or DIC). For models estimated with the MCMC, this is generally considered a more reliable indicator than the AIC or BIC/SC criteria. But since this is not calculated for the MLE, we cannot compare them. 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.

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.[5]

Fourth, the over-dispersion tests reveal identical: 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

[5] Frequently, the model error is greater for an MCMC model than an MLE model. Whether this represents true model error or just a coincidence cannot be easily determined at this point.

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.

Why Run an MCMC when MLE is So Easy to Estimate?

What we have seen is that the MCMC negative binomial model produces results that are very similar to that of the MLE negative binomial model. In other words, simulating the distribution of the Poisson-Gamma 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 in calculating time. For example, Table Up. 2.8 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. For the MLE test, the run took 6 seconds. For an MCMC equivalent test, the run took 86 minutes! Clearly, the MCMC algorithm is a lot 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 when all the functions in a mixed function model are part of the exponential family of functions. MLE is particularly well suited for this family. 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 we were to substitute a lognormal term for the Gamma term in the negative binomial (so that the model became a Poisson-lognormal), a different likelihood function would need to be defined. 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).

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. In practice, the MLE is usually the preferred approach for estimating models, unless the

model is too complex to be estimated via a likelihood function or informative priors can be used to refine the estimate of the model.

In future versions of *CrimeStat*, we plan on introducing more complex models. For this version, we introduce the Poisson-Gamma-CAR model which cannot be solved by the MLE approach.

Poisson-Gamma-CAR Model

The Poisson-Gamma-CAR model has three mathematical properties. First, it has a Poisson mean, similar to the Poisson family of models. Second, it has a Gamma dispersion parameter, similar to the negative binomial model. Third, it incorporates an estimate of local spatial autocorrelation in a CAR format (see equation Up. 2.55). As mentioned above, the Poisson-Gamma-CAR function is defined as:

$$y_i \mid \lambda_i \sim Poisson(\lambda_i) \qquad \text{repeat (Up. 2.56)}$$

with the mean of Poisson-Gamma-CAR organized as:

$$\lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i + \phi_i) = \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \phi_i) \cdot \xi_i \qquad \text{repeat (Up. 2.57)}$$

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.

To model the spatial effect, ϕ_i, we assume the following:

$$p(\phi_i \mid \boldsymbol{\Phi}_{-i}) \propto \exp\left(-\frac{w_{i+}}{2\sigma_\phi^2}\left[\phi_i - \rho \sum_{j \neq i} \frac{w_{ij}}{w_{i+}} \phi_j \right]^2 \right) \qquad \text{(Up. 2.67)}$$

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 (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 is Rho (ρ) and might be considered a global component. The second is Tauphi (τ_ϕ) and might be considered a local

component while the third is Alpha (α) and might be considered a neighborhood component since it measures the distance decay. Phi (Φ) is normally distributed and is a function of Rho and Tauphi.

$$\phi_i \mid \mathbf{\Phi}_{-i} \sim N\left(\rho \sum_{j \neq i}^{n} (w_{ij} / w_{i+}) \phi_j , \sigma_\phi^2 / w_{i+} \right) \quad \text{(Up. 2.68)}$$

Tauphi, in turn, is assumed to follow a Gamma distribution

$$\tau_\phi = \sigma_\phi^{-2} \sim Gamma\,(a_\phi, b_\phi) \quad \text{(Up. 2.69)}$$

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

$$\psi \sim Gamma\,(a_\psi, b_\psi) \quad \text{(Up. 2.70)}$$

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.

Finally, 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

$$w_{ij} = e^{-\alpha d_{ij}} \quad \text{(Up. 2.71)}$$

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

$$w_{ij} = K e^{-\alpha d_{ij}} \quad \text{(Up. 2.72)}$$

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

$$c_{ij} = w_{ij} \quad \text{(Up. 2.73)}$$

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.

Example of Poisson-Gamma-CAR Analysis of Houston Burglaries

To illustrate, we run the Houston burglary data set using a negative exponential spatial weights. 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 Table Up. 2.9 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≤.001) indicating that there is substantial spatial autocorrelation in the error term.

Third, we estimated the value of $\boldsymbol{\alpha}$, the distance decay coefficient. In *CrimeStat*, there is a diagnostic utility that will calculate a range of probable values for $\boldsymbol{\alpha}$. 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 alpha value would be -0.637 for distance units in miles.

Fourth, the Poisson-Gamma-CAR model was run on the Houston burglary dataset using the estimated alpha value in mile units (-0.637). Table Up. 2.10 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, 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 the individual coefficients, the intercept and the two independent variables have values very similar to that of MCMC Poisson-Gamma presented in Table Up. 2.9. Note, though, that the coefficient value for the intercept is now smaller. The reason is that the spatial effects, the Φ 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 Up. 2.10:
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	0.95[n.s.]	0.001	0.056	1.023

n.s. Not significant
*** p≤.001

Figure Up. 2.8 show 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 Up. 2.4), 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 Up. 2.11 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 Up. 2.11:
Spatial Autocorrelation in the Residual Errors of the Houston Burglary Model

	Raw Dependent Variable	**Residuals Normal Model**	**Residuals Poisson NB1 Model**	**Residuals MCMC Poisson-Gamma Model**	**Residuals 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.

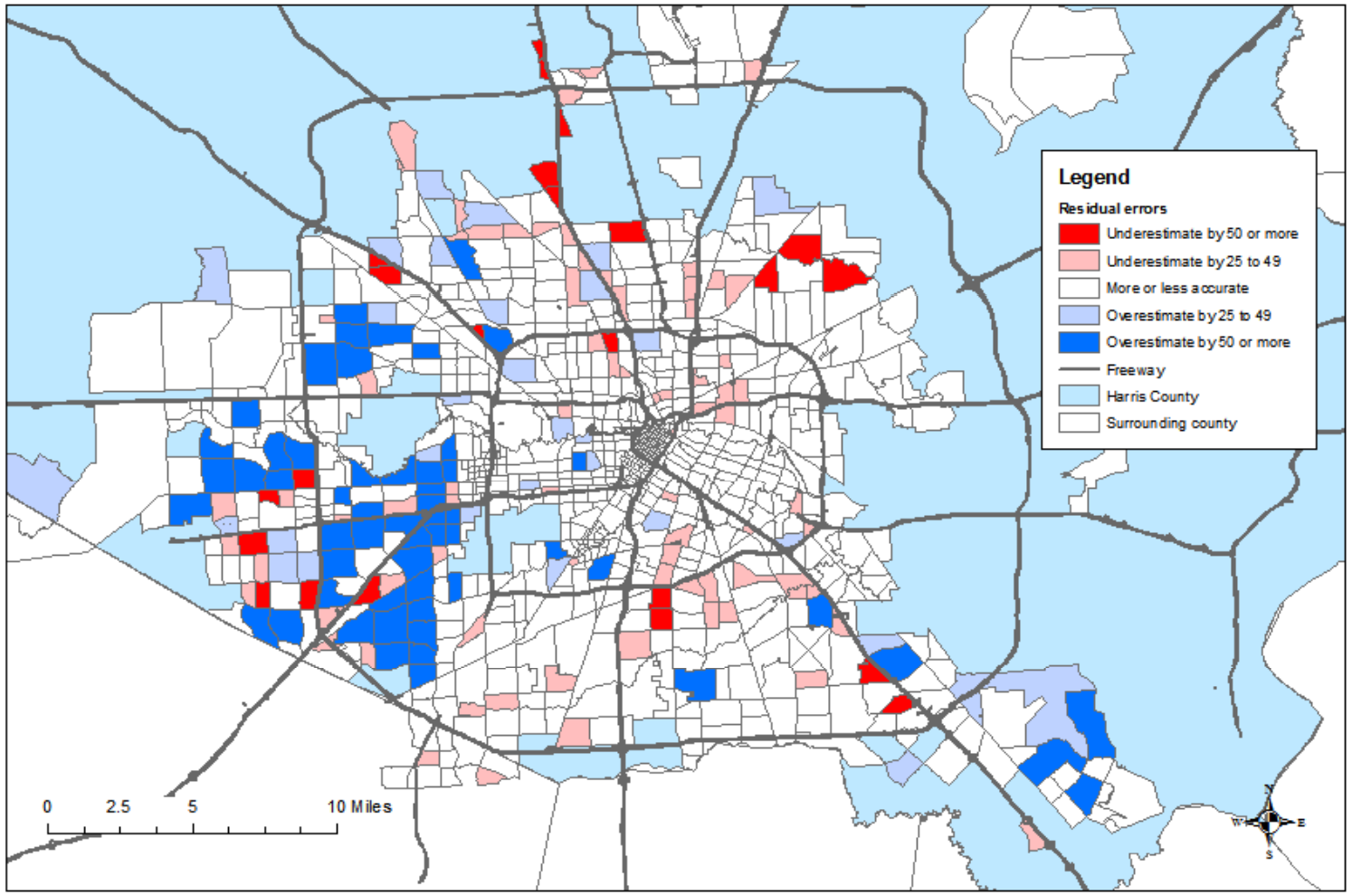
n.s. Not significant
** p≤.01
*** p≤.001
**** p≤.0001

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 is positive spatial autocorrelation

Figure Up. 2.8:

Predicting Burglaries in the City of Houston: 2006

Residual Errors from Poisson-Gamma-CAR Model



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. A routine for the "G" statistic was introduced in *CrimeStat* version 3.2 and the documentation of it can be found in the update chapter for that version.

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 Up. 2.11 indicate 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 of the 'low positive' type. That is, the clustering occurs because zones with low residual errors are predominately near other zones with low residual errors. That is, the models have better predicted the zones with low numbers of burglaries than those with high 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.

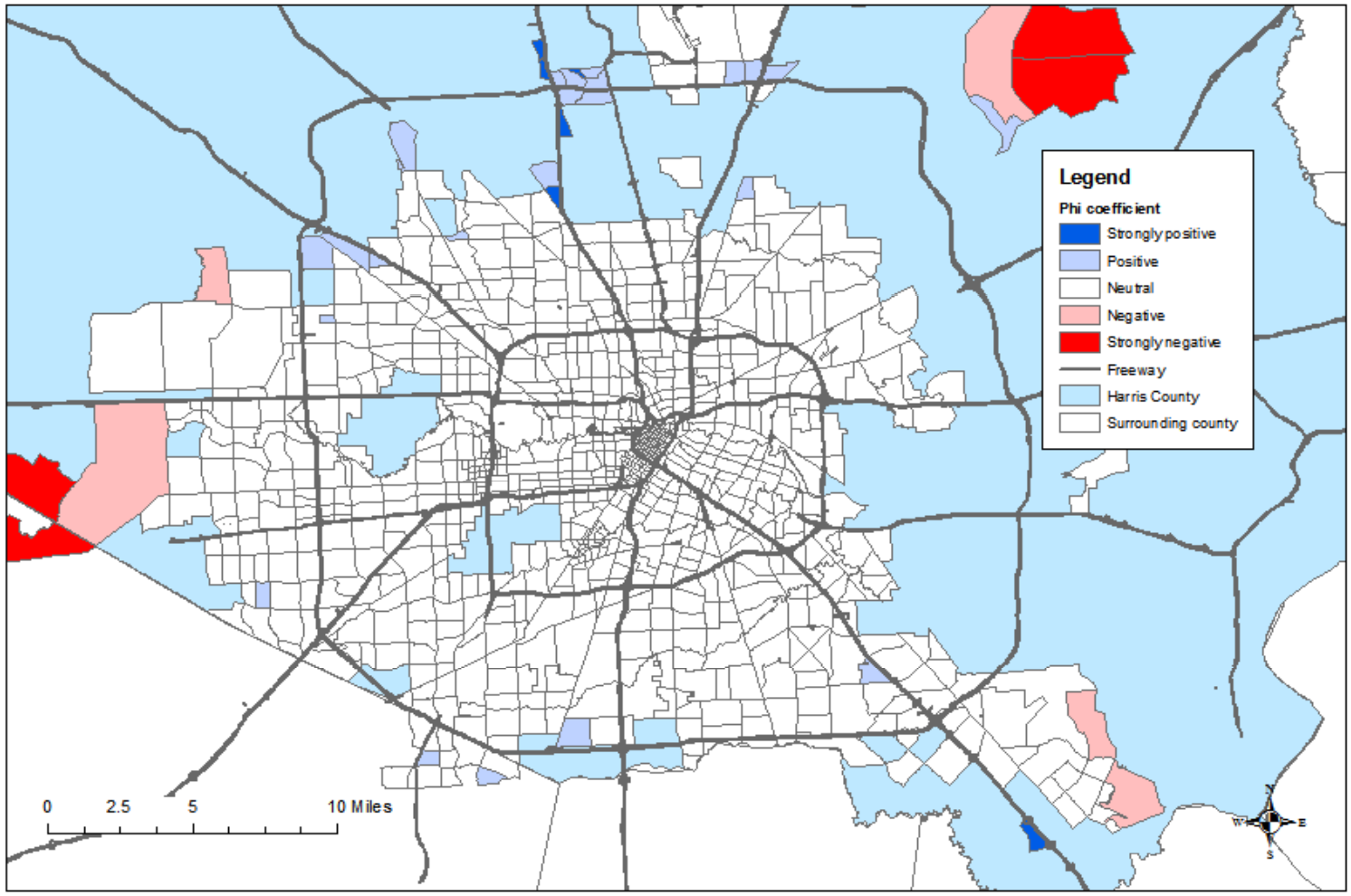
Figure Up. 2.9 show the distribution of the Phi values. By and large, the spatial adjustment is very minor in most parts of Houston with its greatest impact at the edges, where one might expect some spatial autocorrelation due to very low numbers of burglaries and 'edge effects'.

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.

Figure Up. 2.9:

Predicting Burglaries in the City of Houston: 2006

Phi Coefficients from Poisson-Gamma-CAR Model



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.

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, μ_i with a Gamma dispersion:

$$y_i \sim Poisson(\mu_i) \qquad \text{(Up. 2.74)}$$

In turn, the mean of the Poisson is modeled as:

$$\mu_i = \nu_i \lambda_i \qquad \text{(Up. 2.75)}$$

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 (times some multiple 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.

The rate is further structured in the Poisson-Gamma-CAR model as:

$$\mu_i = \nu_i \lambda_i = \nu_i \cdot \exp(\mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i + \phi_i) \quad \text{(Up. 2.76)}$$

where the symbols have the same definitions as in equation Up. 2.57.

With the exposure term, the full model is estimated as the same fashion,

$$y_i \sim Poisson(\nu_i \lambda_i) \quad \text{(Up. 2.77)}$$

$$\lambda_i \sim Gamma(\psi, \psi e^{-\mathbf{x}_i^T \boldsymbol{\beta} - \phi_i}) \quad \text{(Up. 2.78)}$$

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 (2010) 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 an MCMC Poisson-Gamma-CAR model using the number of households as the exposure variable. There was, therefore, only one independent variable, median household income. Table Up. 2.12 show the results.

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 Up. 2.11), the model does not fit the data as well. The log likelihood is lower 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, the adjusted Pearson Chi-square is very high while the adjusted deviance is moderately high. In other words, the exposure variable has not eliminated the dispersion as much as the random effects (non-exposure) model.

Table Up. 2.12:
Predicting Burglaries in the City of Houston: 2006
MCMC Poisson-Gamma-CAR Model with Exposure Variable
(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:	-4,736.6
DIC:	146,129.2
AIC:	9,481.2
BIC/SC:	9,501.5
Deviance:	2,931.1
p-value of deviance:	0.0001
Pearson Chi-square:	34,702.9
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
Inverse dispersion multiplier:	1.7

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 Up. 2.12 (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

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 Up. 2.10. 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. Finally, the average Φ value is positive but not significant, similar to what it was in Table Up. 2.10.

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 0.5th, 2.5th, 97.5th, and 99.5th percentiles. The 2.5th and 97.5th represent 95% credible intervals while the 0.5th and 99.5th 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. 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.

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 Poisson-Gamma-CAR model. 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, for all practical purposes, *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. Alternatively, they can come from other methods that have attempted to analyze the same phenomena. Lynch (2008), 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 an MCMC Poisson-Gamma-CAR model, the estimates from the MLE negative binomial are probably good starting values, as we saw in the Houston burglary example above.

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 Poisson-Gamma-CAR for each of the groups. Table Up. 2.13 show the results of the coefficients with the standard errors in brackets.

Table Up. 2.13:
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)*
Phi (Φ) Coefficient	0.0104 *(0.1117)*	-0.0020 *(0.0676)*	0.0077 *(0.0683)*

Column 1 show the results of running the model on group A. Column 2 show the results of running the model on group B while column 3 show 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 Up. 2.54 does, updating the probability estimate of the data given the likelihood based on the prior probability. In short, using prior estimates combines old information with the new 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 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, 2010). 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 Up. 2.14 present the results.

Table Up. 2.14:
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 INVOLVED 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
DIC:	256,659.6
AIC:	573.4
BIC/SC:	607.6
Deviance:	316.6
p-value of deviance:	0.0001
Pearson Chi-square:	475.6
Model error estimates	
Mean absolute deviation:	0.32
Mean squared predicted error:	0.25
Over-dispersion tests	
Adjusted deviance:	0.60
Adjusted Pearson Chi-Square:	0.91
Dispersion multiplier:	0.15
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
Pct 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≤.01
*** p≤.001

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 pseudo t-test. The results were consistent with expectations, namely zones (TAZs) with greater population, with 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 90,000 iterations after discarding the initial 10,000 'burn in' samples. Table Up. 2.15 show the results. Comparing Table s Up. 2.15 with Up. 2.14, we can see that the overall likelihood statistics was approximately the same as were the over-dispersion 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 iterations 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 are 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 iterations and drop more during the 'burn in' phase. This increases the calculating time, of course, but the results will be more reliable. One can do 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 iterations and 'burn in' samples.

Table Up. 2.15:
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:	**NUMBER OF DRIVERS INVOLVED 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
DIC:	2,915,931.9
AIC:	573.2
BIC/SC:	607.4
Deviance:	317.9
p-value of deviance:	0.0001
Pearson Chi-square:	479.5
Model error estimates	
Mean absolute deviation:	0.32
Mean squared predicted error:	0.25
Over-dispersion tests	
Adjusted deviance:	0.61
Adjusted Pearson Chi-Square:	0.92
Dispersion multiplier:	0.14
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
Pct 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
* $p \leq .05$
** $p \leq .01$
*** $p \leq .001$

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 (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.

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. In a later version of *CrimeStat*, we may allow a graphical time series plot of the parameters.

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 and, thereby, the t-test is 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 Up. 2.12.

Multicollinearity and Overfitting

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 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, with multicollinearity, the overall model will appear 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. A 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 Up. 2.16 illustrate 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.

Table Up. 2.16:
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) **Model 1:**	(2) **Model 2:**	(3) **Model 3:**
Independent variables			
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≤.05
** p≤.01
*** p≤.001

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.

Well, 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. Repeating equation Up. 2.18, tolerance is defined as:

$$Tol_i = 1 - R^2_{j \neq i} \qquad \text{repeat (Up. 2.18)}$$

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. 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. Nevertheless, it is a very good indicator of potential problems. When the tolerance of a variable becomes 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 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. 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 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 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. 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 doesn't 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 model. The condition number of a matrix is an indicator of how amenable it is to digital solution (Wikipedia, 2010d). 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-CAR 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, albeit significance is tested in the usual way. This will be particularly true for complex models, such as the Poisson-Gamma-CAR. An example will illustrate this. With an Intel 2.4 Ghz computer with a duel 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 Up. 2.17 present the results.

As can be seen, the calculation time goes up exponentially with the sample size. Further, with the spatial Poisson-Gamma-CAR model, a limit was reached. Because this routine is calculating the distance between each observation and every other observation as part of calculating the spatial weight coefficients (see equation Up. 2.55), 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 Up. 2.17:
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). 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. 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 user defines certain 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.

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.

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;

$$AdjStd.Err = StdErr_{block} * \sqrt{\frac{\bar{n}}{N}} \qquad \text{(Up. 2.79)}$$

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.

Another statistic that does not scale well with the block sampling method is the Deviance Information Criterion (DIC). Consequently, the DIC is calculated as the average of the block sample rather than being scaled to a full dataset. A note to that effect is printed. In making comparisons between models, one should use the block sample average for the DIC. Other statistics, however, appear to be well estimated by the block sampling method.

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 Up. 2.18 show the results of the traffic dataset. The coefficients are very close within the second decimal place and the adjusted standard errors are within the third decimal place. On the other hand, the block sampling method took 11.2 minutes to run compared to only 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.

Test 2

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, 2010). 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. Not all of the variables are significant, according to the t-test of the full dataset.

Table Up. 2.18:
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
n.s. Not significant
* $p \leq .05$
** $p \leq .01$
*** $p \leq .001$

Table Up. 2.19 show the results of the journey to crime 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.

Table Up. 2.19:
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 run:	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
$\mathbf{X_1}$	0.8871***	0.025	1.0135*	0.430	0.028
$\mathbf{X_2}$	0.3311***	0.018	0.3434	0.294	0.019
$\mathbf{X_3}$	-0.2274***	0.012	-0.2751	0.199	0.013
$\mathbf{X_4}$	-0.2820***	0.014	-0.3137	0.231	0.015
$\mathbf{X_5}$	0.2525***	0.016	0.3099	0.256	0.016
$\mathbf{X_6}$	0.3560***	0.027	0.3783	0.488	0.031
$\mathbf{X_7}$	0.0753***	0.013	0.1092	0.214	0.014
$\mathbf{X_8}$	0.1766***	0.021	-0.0030	0.374	0.024
$\mathbf{X_9}$	0.1880***	0.023	0.1326	0.406	0.026
$\mathbf{X_{10}}$	0.0135n.s.	0.016	-0.0070	0.268	0.017
$\mathbf{X_{11}}$	-0.5697***	0.016	-0.6759 *	0.265	0.017
$\mathbf{X_{12}}$	0.0042n.s.	0.014	0.0521	0.226	0.015
$\mathbf{X_{13}}$	-0.2214***	0.016	-0.2755	0.262	0.017
$\mathbf{X_{14}}$	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≤.05
** p≤.01
*** p≤.001

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 won't be. 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 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 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. They may be significantly different than 0, but probably not very important. 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, also the strongest variables will 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.

The *CrimeStat* Regression Module

We now describe the *CrimeStat* regression module.[6] 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. Figure Up. 2.10 displays the Regression I page.

In the current version, six possible regression models are available with several options for each of these:

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

There are several sections to the page that define these models.

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.

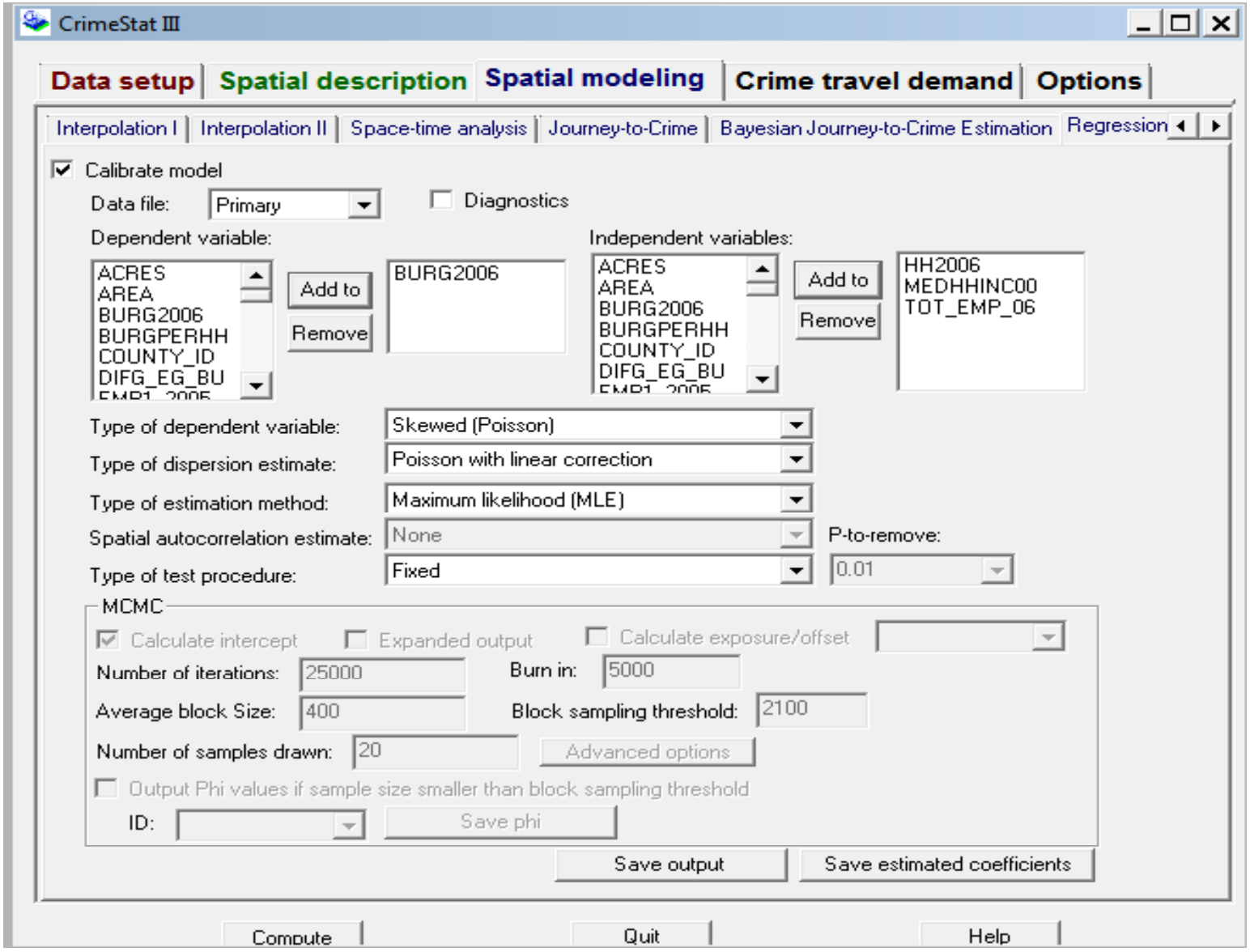
Dependent Variable

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

[6] The code for the Linear, Poisson, and MLE Poisson-Gamma functions came from the *MLE++* package of routines developed by Ian Cahill of Cahill Software, Ottawa, Ontario. We have added additional summary statistics, significance tests, tolerance estimates and stepwise procedures to this package. The code for the MCMC routines was developed by us under instructions from Dr. Shaw-pin Miaou of College Station, TX. The programming was conducted by Ms. Haiyan Teng of Houston, TX. We thank these individuals for their contributions.

Figure Up. 2.10:

Regression I Setup Page



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: Normal (OLS) or Skewed (Poisson). The default is a Poisson. At this point, these are the only choices that are available though we will be adding a binomial and multinomial choice soon.

Type of Dispersion Estimate

The second option that must be defined is the type of dispersion estimate to be used. The choices are Gamma, Poisson, Poisson with linear correction (NB1), and Normal (automatically defined for the OLS model). The default is Gamma. Soon, we will be adding a log linear and possibly another dispersion parameter to the routine.

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 above and in appendices C and D.

Spatial Autocorrelation Estimate

Fourth, if the user accepts an MCMC algorithm, then another choice is whether to run a spatial autocorrelation estimate along with it (a Conditional Autoregressive function, or CAR). This can only be run if the dependent variable is Poisson and the dispersion parameter is Gamma. If the spatial autocorrelation estimate is run, then the model becomes a Poisson-Gamma-CAR while if the spatial autocorrelation estimate is not run the model remains a Poisson-Gamma.

Type of Test Procedure

The fifth, and last option, is whether to run a fixed model or a backward elimination *stepwise* procedure (only with the normal or an MLE Poisson model). With a fixed model, the total model is estimated and the coefficients for each of the variables are estimated at the same time. With the backward elimination stepwise procedure, all the variables in the model initially but are removed one at a time, based on the probability level for remaining in the model.

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 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 to estimate either a Poisson-Gamma or a Poisson-Gamma-CAR model, then several decisions have to be made.

Number of Iterations

Specify 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

Specify 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 MCMC algorithm will be run on all cases unless the number of records exceeds the number specified in the block sampling threshold. The default is 6,000 cases. Note that if you run the MCMC for more cases than this, 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.

Average Block Size

Specify the number of cases to be drawn in each block sample. 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

Specify the number of samples to be drawn. 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 model can be run 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

If the model is a risk or rate model (see equations Up. 2.74 through Up. 2.78), 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 Up. 2.11 show the advanced options dialogue. We would suggest keeping the default values initially until you become very familiar with the routine.

Initial Parameters Values

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 is frequently called a 'non-informative' prior. These are displayed as a blank screen for the Beta box. However, estimates of the beta coefficients can be substituted for the assumed 0 coefficients. To do this, <u>all</u> 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 <u>not</u> include the beta coefficients for the spatial autocorrelation, Φ, 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:

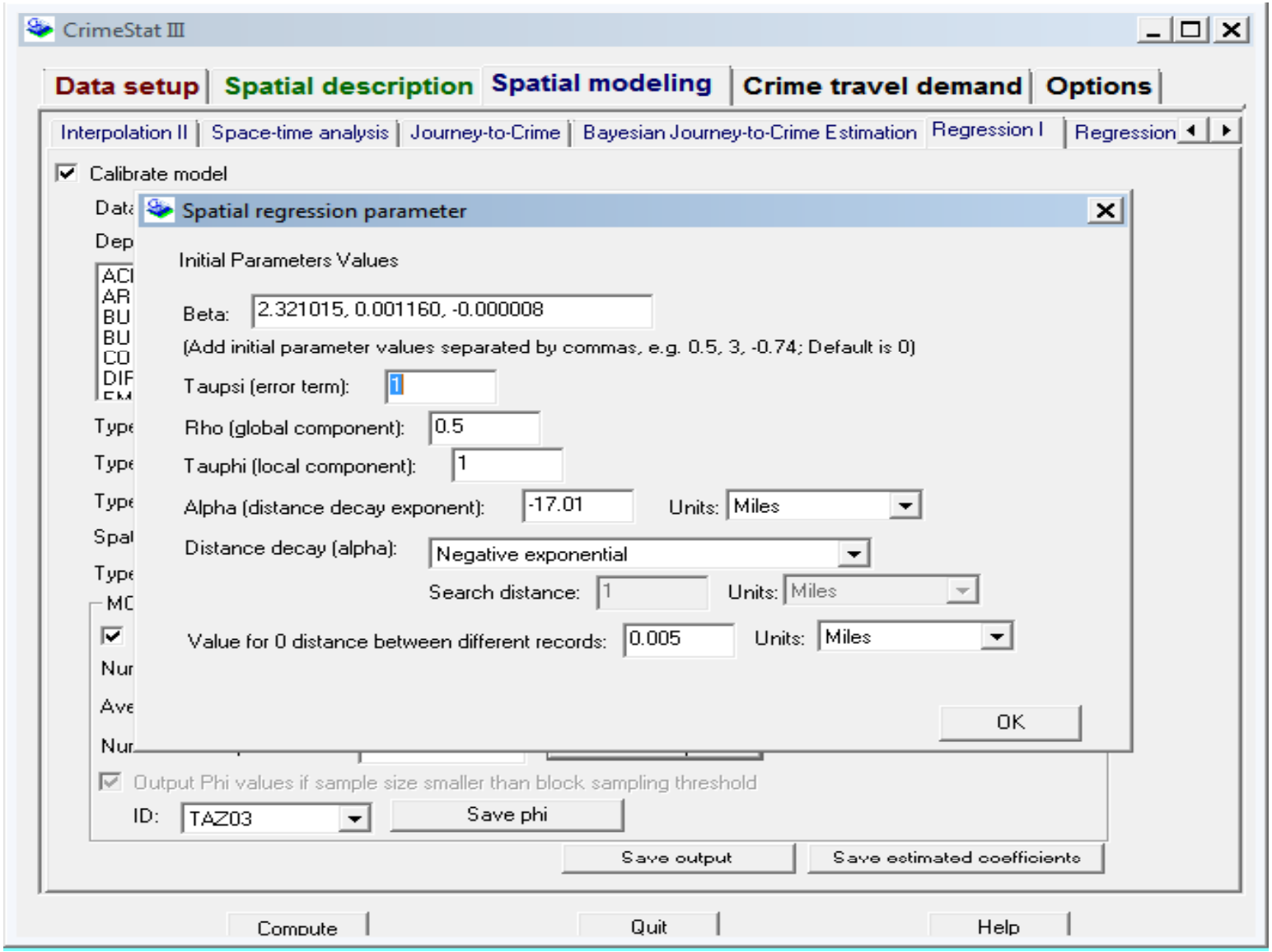
$$Y = e^{4.5 + 0.3\,X1 - 2.1X2 + 3.4X3} \qquad \text{(Up. 2.80)}$$

The researcher wants to repeat this model but with a different data set. The researcher 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 (Up. 2.81)

Figure Up. 2.11:

Advanced Options for MCMC Poisson-Gamma-CAR Model



The routine will then 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 formulas Up. 2.70 – Up. 2.72). 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

$$\text{Weight} = e^{-\alpha * d(ij)} \qquad \text{(Up. 2.82)}$$

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:	$\text{Weight} = e^{-\alpha * d(ij)}$	for $d(ij) \geq 0$, $d(ij) \leq d_p$	(Up. 2.83)
Beyond search distance:	0	for $d(ij) > d_p$	(Up. 2.84)

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 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 the sharp decay, alpha is given by:

$$\alpha = \ln(0.5)/NN(distance) \qquad \text{(Up. 2.85)}$$

where NN(distance) is the nearest neighbor distance.

b. For the shallow distance decay, alpha is given by:

$$\alpha = \ln(0.9)/NN(distance) \quad \text{(Up. 2.86)}$$

where NN(distance) is the nearest neighbor distance.

c. For the intermediate decay, alpha is given by:

$$\alpha = \ln(0.75)/NN(distance) \quad \text{(Up. 2.87)}$$

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 0 Distances 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) 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 dependent variable
2. The number of cases
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)
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. 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 normal 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 (MCMC Poisson-Gamma, Poisson-Gamma-CAR) 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 (Linear, Poisson, Poisson with linear correction, Poisson-Gamma, Poisson-Gamma-CAR)
8. The method of estimation
9. The number of iterations
10. The 'burn in' period
11. The distance decay function used for a Poisson-Gamma-CAR model. This is output with the Poisson-Gamma-CAR model 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. This is 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 the Poisson-Gamma-CAR model 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 the Poisson-Gamma-CAR model 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 the Poisson-Gamma-CAR model 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 Coefficient 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 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.
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, ϕ) 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, ρ); a local component (Tauphi, τ_φ); and a local neighborhood component (Alpha, α, which is defined by the user).
44. 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:

45. 2.5^{th} percentile
46. 5^{th} percentile
47. 10^{th} percentile
48. 25^{th} percentile
49. 50^{th} percentile (median)
50. 75^{TH} percentile
51. 90^{th} percentile
52. 95^{th} percentile
53. 97.5^{th} 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 2.5^{th} and 97.5^{th} percentiles provide approximate 95 percent confidence intervals around the mean coefficient while the 0.5^{th} and 99.5^{th} 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 (Poisson-Gamma-CAR Model Only)

For the Poisson-Gamma-CAR model 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 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 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 Up. 2.3).

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.

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 (see figure Up. 2.10). 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.[7] 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.

[7] 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 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):

$$g = \frac{n}{(n-1)(n-2)} \sum_i [(X_i - \bar{X}) / s]^3 \qquad \text{(Up. 2.88)}$$

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:

$$s = \sqrt{\sum_i \frac{(X_i - \bar{X})^2}{(n-1)}} \qquad \text{(Up. 2.89)}$$

The standard error of skewness (SES) can be approximated by (Tabachnick and Fidell, 1996):

$$SES = \sqrt{\frac{6}{n}} \qquad \text{(Up. 2.90)}$$

An approximate Z-test can be obtained from:

$$Z(g) = \frac{g}{SES} \qquad \text{(Up. 2.91)}$$

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 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. The statistic was discussed extensively in chapter 4. 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 Poisson-Gamma-CAR 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 the Poisson-Gamma-CAR Model

The fourth type of test produced by the diagnostics utility is an estimate of a plausible value for the distance decay function alpha, α, in the Poisson-Gamma-CAR model. 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 gra ph 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.

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 Up. 2.18).

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,

$$LR = 2(L_C - L_B) \qquad \text{(Up. 2.92)}$$

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).

Regression II Module

The Regression II module allows the user to apply a model to another dataset and make a prediction. Figure Up. 2.12 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 two types of models that are fitted – normal and Poisson. For the normal model, the routine fits the equation:

$$Y = \beta_0 + \beta_1 X_1 + \ldots\ldots\ldots + \beta_k X_k \qquad \text{(Up. 2.93)}$$

whereas for the Poisson model, the routine fits the equation:

$$E(Y_i / X_{ki}) = \lambda_i = \mathbf{e}^{\beta 0 + \beta 1 X1 + \ldots\ldots\ldots + \beta k Xk + [\Phi]} \qquad \text{(Up. 2.94)}$$

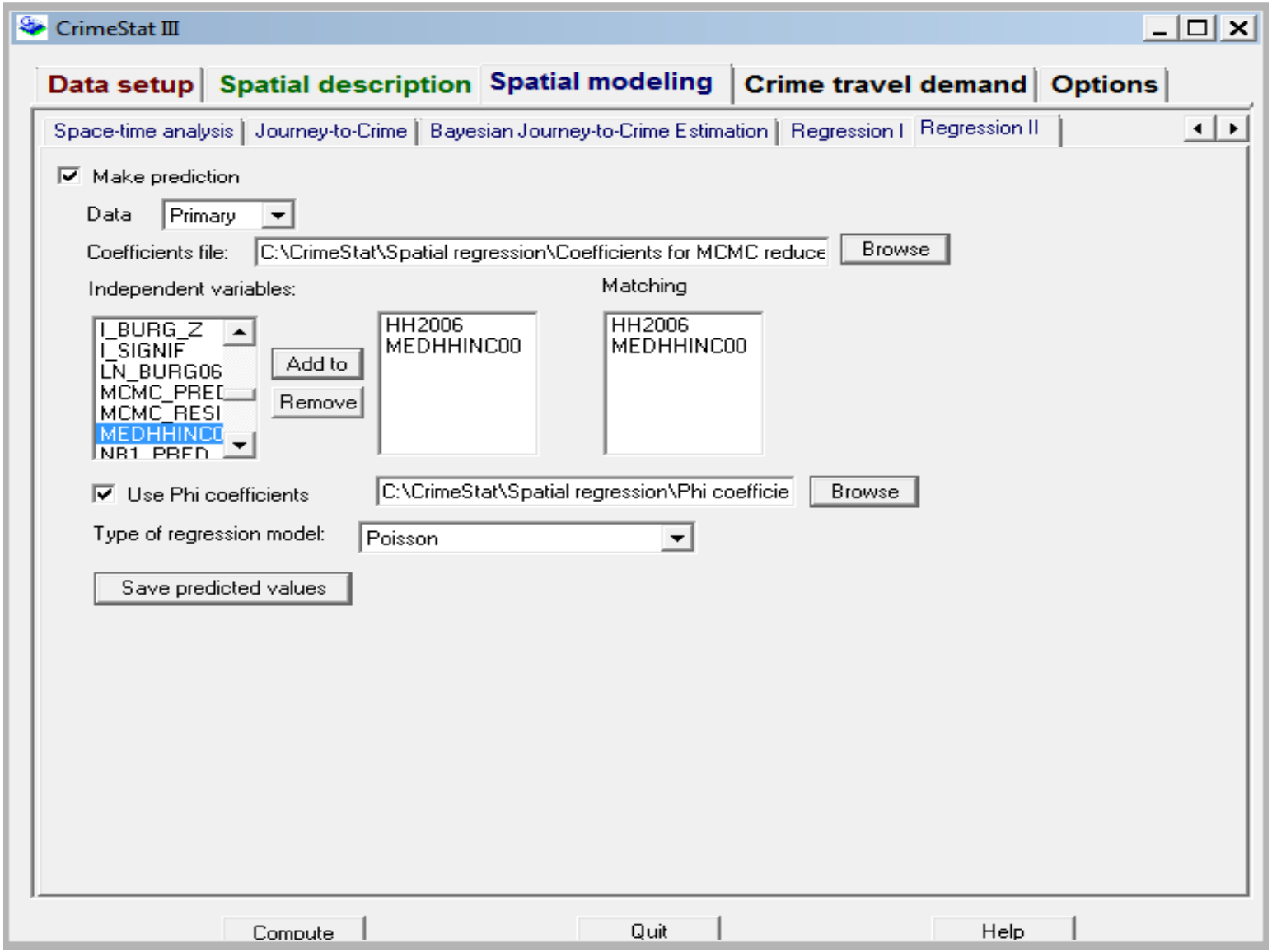
with β_0 being the intercepted (if calculated), $\beta_1 \ldots . \beta_k$ being the saved coefficients and Φ_i is the saved Phi values (if the Poisson-Gamma-CAR model was estimated). Notice that there is no error in each equation. Error was part of the estimation model. What were saved was only the coefficients.

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 dataset based on the order of the coefficients file.

If the model had estimated a general spatial effect from a Poisson-Gamma-CAR model, then the general Φ will have been saved with the coefficient files. If the model had estimated specific spatial effects from a Poisson-Gamma-CAR 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 coefficients file along with the general coefficient file.

Figure Up. 2.12:

Regression II Setup Page



References

Abraham, Bovas & Johannes Ledolter (2006). *Introduction to Regression Modeling*. Thompson Brooks/Cole: Belmont, CA.

Anselin, L. (2002). "Under the hood: Issues in the specification ad interpretation of spatial regression models", *Agricultural Economics*, 17(3), 247-267.

Berk, Kenneth N. (1977). "Tolerance and condition in regression computations", *Journal of the American Statistical Association*, 72 (360), 863-866.

Besag, J., P. Green, D. Higdon, & K. Mengersen (1995). "Bayesian computation and stochastic systems (with discussion)", *Statistical Science*, 10, 3-66.

Bishop, Y. M. M., S. E. Feinberg, & P. W. Holland (1975). *Discrete Multivariate Analysis: Theory and Practice*. MIT Press: Cambridge, MA.

Boswell, M. T. & G.P. Patil, (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.

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 & Pravin K. Trivedi (1998). *Regression Analysis of Count Data*. Cambridge University Press: Cambridge, U.K.

Carlin, Bradley P. & Thomas A. Louis (2000). *Bayes and Empirical Bayes Methods for Data Analysis*, Chapman and Hall/CRC: Boca Raton.

Cressie, Noel (1993). *Statistics for Spatial Data*. John Wiley & Sons: New York.

Clayton, David & John Kaldor (1987). "Empirical Bayes estimates of age-standardized relative risks for use in disease mapping". *Biometrics*, 43, 671-681.

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., C.C. Holmes, B.K. Mallick, & A.F.M. Smith (2002). *Bayesian Methods for Nonlinear Classification and Regression*. John Wiley & Sons, Ltd: Chichester, Sussex.

De Smith, Michael, Michael F. Goodchild, & Paul A. Longley (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.

Draper, Norman & Harry Smith (1981). *Applied Regression Analysis, Second Edition*. John Wiley & Sons: New York.

Findley, David 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.

Fotheringham, A. Stewart, Chris Brunsdon, & Martin Charlton (2002). *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. John Wiley & Sons: New York.

Gelman, A. (1996). "Inference and monitoring convergence". In Gilks, W. R., S. Richardson, & D. J. Spiegelhalter (eds), *Markov Chain Monte Carlo in Pratice*, Chapman and Hall: London.

Gelman, A., J. B. Carlin, H. S. Stern, & D. B. Rubin (2004). *Bayesian Data Analysis* (second edition). Chapman and Hall/CRC: Boca Raton, FL.

Gelman, A. & D. B. Rubin (1992). "Inference from iterative simulation using multiple sequences (with discussion)", *Statistical Science*, 7, 457-511.

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.

Guo, Feng, Xuesong Wang, & Mohamed A. Abdel-Aty (2009). "Modeling signalized intersection safety with corridor-level spatial correlations", *Accident Analysis and Prevention*, In press.

Hall, D. B. (2000). "Zero-inflated Poisson and binomial regression with random effects: a case study". *Biometrics*, 56, 1030-1039.

Hastings, W. K. (1970). "Monte Carlo sampling methods using Markov Chains and their applications", *Biometrika*, 57, 97-109.

H-GAC (2010). Transportation and air quality program, *Houston-Galveston Area Council*. http://www.h-gac.com/taq/.

Hilbe, Joseph M. (2008). *Negative Binomial Regression (with corrections)*. Cambridge University Press: Cambridge.

Husmeier, Dirk & Grainne McGuire (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.

Kanji, Gopal K. (1993). *100 Statistical Tests*. Sage Publications: Thousand Oaks, CA.

Lee, Peter M. (2004). *Bayesian Statistics: An Introduction* (third edition). Holder Arnold: London.

Leonard, Thomas & John S.J. Hsu (1999). *Bayesian Methods: An Analysis for Statisticians and Interdisciplinary Researchers*. Cambridge University Press: Cambridge.

Levine, Ned (2010). "Spatial variation in motor vehicle crashes by gender in the Houston metropolitan area". *Proceedings of the 4th International Women's Issues in Transportation Conference*. In press.

Levine, Ned & Phil Canter (2010). "Linking origins with destinations for DWI motor vehicle crashes: An application of Crime Travel Demand modeling". *Crime Mapping*, Under revision.

Levine, Ned & Patsy Lee (2010). "Gender and journey-to-crime in Manchester, England". *Criminology*, Under revision.

Lord, Dominique (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, Dominique & L.F. Miranda-Moreno (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.

Lynch, Scott M. (2007). *Introduction to Applied Bayesian Statistics and Estimation for Social Scientists*. Springer: New York.

McCullagh, P. & J. A. Nelder (1989). *Generalized Linear Models* (2nd edition). Chapman & Hall/CRC: Boca Raton, FL.

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, Shaw-pin (2006). "Coding instructions for the spatial regression models in CrimeStat". Unpublished manuscript. College Station, TX.

Miaou, Shaw-Pin, Joon Jin Song, & Bani K. Mallick (2003). "Roadway traffic crash mapping: a space-time modeling approach", *Journal of Transportation and Statistics*, 6 (1), 33-57.

Miaou, Shaw-Pin (1996). *Measuring the Goodness-of-Fit of Accident Prediction Models*. FHWA-RD-96-040. Federal Highway Administration, U.S. Department of Transportation: Washington, DC.

StatSoft (2010). "Tolerance", *StatSoft Electronic StatisticsTextbook,* StatSoft:Tulsa, OK. http://www.statsoft.com/textbook/statistics-glossary/t/button/t/

Microsoft (2003). "SKEW - skewness function", *Microsoft Office Excel 2003, Microsoft: Redmond, WA*.

Mitra, S. & S. Washington (2007). "On the nature of over-dispersion in motor vehicle crash prediction models", *Accident Analysis and Prevention*, 39, 459-468.

Myers, R. H. (1990) Classical and Modern Regression with Applications, 2nd edition, Duxbury Press, Belmont, CA.

Nannen, Volker (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.

Ntzourfras, I. (2009). *Bayesian Modeling using WinBugs*. Wiley Series in Computation Statistics, Wiley: New York.

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.

Park, Byung-Jung (2009). "Note on the Bayesian analysis of count data". From Park, Byung-Jung PhD thesis, Texas A & M University: College Station, TX.

Park, E.S., and D. Lord (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 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.

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.

Tabachnick, B. G. & L. S. Fidell (1996). *Using Multivariate Statistics* (3rd ed). Harper Collins: New York.

Train, K. (2009). *Discrete Choice Methods with Simulation* (2nd edition). Cambridge University Press: Cambridge.

Venables, W.N. and B.D. Ripley (1997). *Modern Applied Statistics with S-Plus (second edition)*. Springer-Verlag: New York.

Wikipedia (2010a). "Negative binomial distribution", *Wikipedia*, http://en.wikipedia.org/wiki/Negative_binomial_distribution Accessed February 24, 2010.

Wikipedia (2010b). "Maximum likelihood", *Wikipedia*. http://en.wikipedia.org/wiki/Maximum_likelihood. Accessed March 12, 2010.

Wikipedia (2010c). "Greedy algorithm", *Wikipedia*. http://en.wikipedia.org/wiki/Greedy_algorithm. Accessed March 12, 2010.

Wikipedia (2010d). "Condition number". *Wikipedia. http://en.wikipedia.org/wiki/Condition_number*. Accessed March 19, 2010