Chapter 9:

Hot Spot Analysis of Zones

Ned Levine
Ned Levine & Associates
Houston, TX

Table of Contents

Table of Contents (continued)

Chapter 9:

Hot Spot Analysis of Zones

In this chapter, we will discuss methods for identifying hot spots with zonal data. The user should be thoroughly familiar with the information presented in Chapter 5 on spatial autocorrelation indices because two of the same indices are used for the analysis of local variations in zones.

We are going to look at four techniques for analyzing hot spots with zonal data or with individual level data that have attributes (count or interval variables that measure a characteristic associated with the X and Y coordinates). These are Anselin's Local Moran, the Getis-Ord Local "G", the Zonal Nearest Neighbor Hierarchical Clustering algorithm, and the Risk-adjusted Zonal Nearest Neighbor Hierarchical Clustering algorithm. Figure 9.1 shows the Hot Spot Analysis of Zones page.

Assigning Point Data to Zones

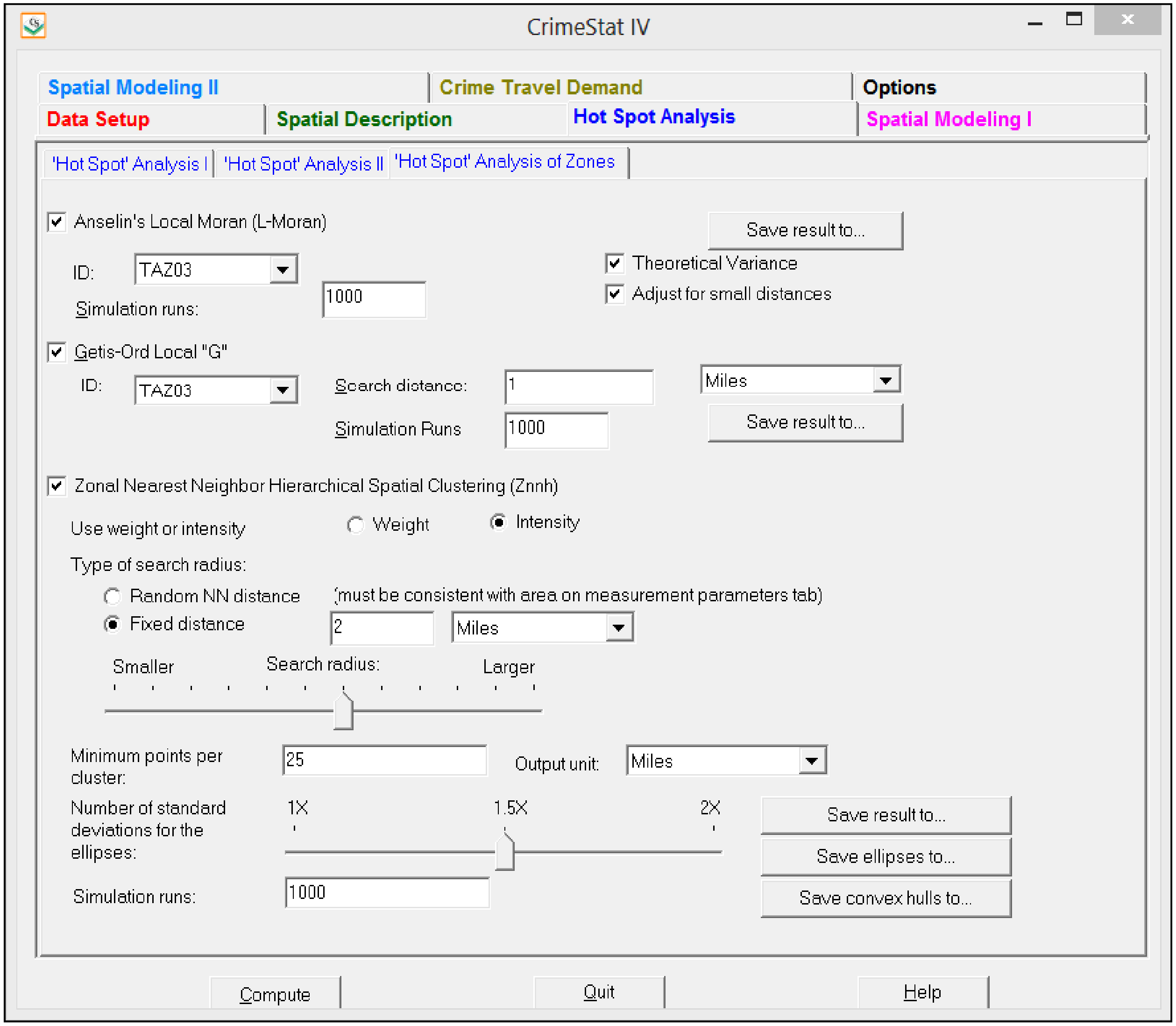
If a user has information on the location of individual events (e.g., robberies), then it is better to utilize that information with the hot spot techniques discussed in Chapters 7 and 8. The individual-level information will contain all the uniqueness of the events.

However, sometimes it is not possible to analyze data at the individual level. The user may need to aggregate individual data points to spatial areas (zones) in order to compare the events to data that are only obtained for zones, such as census data, or to model environmental correlates of the data points or may find that individual data are not available (e.g., when a police department releases information by police beats but not individual streets). Zonal data can include crime counts by zone, socio-economic information (e.g., collected by the census or estimated by a Metropolitan Planning Organization), or some other data that are aggregated to the small areas. In other words, the zone becomes the unit of analysis instead of the individual data points.

Since the zones are not events, they have to be spatially analyzed by assuming that all the data resides at a single point within the zone. This is usually the centroid (the geographical center of the zone) but sometimes the center of minimum distance (the point at which the sum of the distances to all other points is minimized) has been used, too, especially if the zone is very irregularly shaped. However, when individual data points are assigned to zones, information is lost. For example, the distance between zones is a singular value for all the points in those zones whereas there is much greater variability with the distances between individual events. Also,

Figure 9.1:

Hot Spot Analysis of Zones Screen



☑ Anselin's Local Moran (L-Moran)

ID: TAZ03

Simulation runs: 1000

☑ Theoretical Variance

☑ Adjust for small distances

☑ Getis-Ord Local "G"

ID: TAZ03

Search distance: 1 Miles

Simulation Runs 1000

☑ Zonal Nearest Neighbor Hierarchical Spatial Clustering (Znnh)

Use weight or intensity ○ Weight ● Intensity

Type of search radius:

○ Random NN distance (must be consistent with area on measurement parameters tab)

● Fixed distance 2 Miles

Smaller Search radius: Larger

Minimum points per cluster: 25

Output unit: Miles

Number of standard deviations for the ellipses: 1X 1.5X 2X

Simulation runs: 1000

topological information, such as the shape of the zone or the number of other zones that are adjacent, is lost.

For the spatial autocorrelation indices, the interaction between zones is defined by distance. There are advantages and disadvantages. Contiguity (or adjacency) is a property of a zone, not a point. Thus, adjacency defines whether one zone is next to another zone whereas distance is the distance between single points that represent the zones (e.g., centroids). For example, if two zones are 0.25 miles apart, it is not known whether they are adjacent or not. In other words, in adopting a distance-based weight, information about adjacencies is lost. On the other hand, a distance-based weight is standardized. If two zones are adjacent, it is not known how far apart they are separated. Adjacencies can be misleading since they do not indicate the size of the adjacent zones whereas a specified distance is always constant.

The zonal data also must include an *attribute* variable, a variable associated with the zone (e.g., number of robberies; median household income; percentage of households living below poverty level). The attribute can be a *count* or a continuous variable for a distributional property of the zone (e.g., median household income; percentage of households below poverty level) or even a binary variable (e.g.,1 v. 0).[1] The indices discussed in this chapter are applied to the interaction between the attribute variable of the central zone and other zones, weighted by the distance between them.

Individual level data can also have attributes. For example, Levine and Lee (2013) analyzed journey-to-crime distances for offenders in Manchester, England. In this case, the attribute variable was the distance traveled and the statistics discussed in this chapter are appropriate for analyzing that attribute data. Other examples of individual level data with attributes would be the age of the offender, the number of prior convictions, or the number of years of formal education. The key criterion is that the records must have an attribute which is either a count or an interval variable.

Local Indicator of Spatial Association

The basic concept behind a zone-specific measure of spatial autocorrelation is that of a *local indicator of spatial association* (*LISA*) and has been discussed by a number of researchers (Mantel, 1967; Getis, 1991; Anselin, 1995). For example, Anselin (1995) defines this as any statistic that satisfies two requirements:

[1] There is no fundamental difference between a count variable and a continuous interval or ratio variable since a real number can be converted into a count by multiplying by a power of 10 (e.g., $1.23 = 123 \times 10^{-2}$). The statistics discussed in this chapter are applicable to either count or continuous data.

1. The *LISA* for each observation indicates the extent to which there is significant spatial clustering of similar values around that observation; and

2. The sum of the *LISA*s for all observations is proportional to the global indicator of spatial association:

$$L_i = f g(Y_i) \sum_{ji=1}^{K} h(Y_{ji}) \tag{9.1}$$

where L_i is the local indicator of zone i, $g(Y_i)$ is a function of the value of an intensity variable, Y_i, at location i, $h(Y_{ji})$ is a weight function of the values of the intensity variable observed in the neighborhood j_i of i, and f is a scaling constant to ensure that the sum of L_i equals the global spatial autocorrelation index.

The function of the intensity variable can be a raw score, Y_i, a Z-transformation of the intensity variable, such as:

$$Z_i = \frac{(Y_i - \bar{Y})}{s_Y} \tag{9.2}$$

where $\bar{Y}$ is the mean of Y and S_Y is the standard deviation of variable Y, or some other function.

In other words, a *LISA* is an indicator of the extent the value of an observation is affected by its neighboring observations. This requires two conditions. The first is that each observation has a value of an attribute variable that can be assigned to it (i.e., an intensity or weight value) in addition to its X and Y coordinates. For crime incidents, this means data must be aggregated into zones (e.g., number of incidents by census tracts, zip codes, or police reporting districts).

Second, the *neighborhood* has to be defined. This could be either adjacent zones, all other zones negatively weighted by the distance from the observation zone, or all other zones negatively weighted by the distance from the observation zone up to some distance whereupon the weight is zero afterward (a bandwidth). Once these are defined, the *LISA* indicates the value of the observation zone in relation to its neighborhood.

Anselin's Local Moran

Anselin's Local Moran statistic was developed by Luc Anselin and is the oldest LISA statistic (Anselin, 1995). The procedure applies Moran's "I" statistic to individual zones (see Chapter 5), allowing them to be identified as similar or different to their nearby pattern.

The definition of "I_i" is from Getis and Ord (1996):

$$I_i = \frac{(Z_i - \bar{Z})}{S_Z^2} \sum_{j=1}^{N-1} [W_{ij}(Z_j - \bar{Z})] \qquad (9.3)$$

where Z_i is the intensity of observation i, $\bar{Z}$ is the mean intensity over all observations, Z_j is intensity for all other observations, j (where j ≠ i), S_Z^2 is the variance over all observations, and W_{ij} is a distance weight for the interaction between observations i and j. The first term in equation 9.3 refers only to observation i while the second term is the sum of the weighted values for all other observations (but not including i itself).

The expected "I_i" is defined as:

$$E(I_i) = \frac{\sum_{i=1}^{N} w_{ij}}{N-1} \qquad (9.4)$$

where Wij is the distance weight for the interaction between observations i and i. The variances of I_i are somewhat complicated (see endnote *i* for the formulas).

Similarity or Dissimilarity

Since the global Moran's "I" statistic measures similarity in observations over a study area (see Chapter 5), the local Moran "I_i" also indicates the similarity of a zone relative to its neighbors. Thus, in neighborhoods where both the zone and its neighbors have high attribute values, the Local Moran will be positive indicating that the particular zone is similar (i.e., also 'high'). Similarly, in neighborhoods where both the zone and its neighbors have 'low' attribute values, the Local Moran also will be positive indicating that the zone is similar to its neighbors (i.e., also 'low'). When the Local Moran statistic is positive, this is an indicator of *similarity*, not absolute value of the intensity variable.

Conversely, if a zone has a high value of the intensity variable while its neighbors have low values or, alternatively, it has a low value while the neighbors have high values, then the Local Moran statistic will be negative. *Dissimilarity* is an indicator of either a hot spot or a cold spot, in other words zones that are different from their neighborhood. Hot spots would be seen if the number of incidents in a zone is much higher than in the nearby zones. Cold spots would be seen if the number of incidents in a zone is much lower than in the nearby zones.

In other words, the Local Moran statistic indicates whether the zone is similar or dissimilar to its neighbors.

ID Field

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

Distance Weights

The weights, W_{ij}, can be either an indicator of the adjacency of a zone to the observation zone (i.e., ‘1' if adjacent; 0 if not adjacent) or a distance-based weight which decreases with distance between zones i and j. Adjacency indices are useful for defining near neighborhoods; the adjacent zones have full weight while all other zones have no weight. Distance weights, on the other hand, are useful for defining spatial interaction; zones which are farther away can have an influence on an observation zone, although one that is much less. *CrimeStat* uses distance weights, in two forms.

First, there is a traditional distance decay function:

$$W_{ij} = \frac{1}{d_{ij}} \tag{9.5}$$

where d_{ij} is the distance between the observation zone, i, and another zone, j. For example, a zone which is two miles away has half the weight of a zone that is one mile away.

Small distance adjustment

Second, there is an adjustment for small distances. The weight index becomes problematic with small distance between zones since the weight will approach infinity for d_{ij} -> 0. To correct for this, the routine includes an adjustment for small distances so that the maximum weight can be never be greater than 1.0 (see Chapter 5). The adjustment scales distance to one mile, which is a typical distance for crime analysis. When the small distance adjustment is turned on, the minimal distance is scaled automatically to be one mile. The formula used is:

$$W_{ij} = \frac{one\ mile}{one\ mile + d_{ij}} \tag{9.6}$$

in whichever distance units are specified (miles, kilometers, etc).

Output for Each Zone

The output is for each zone includes:

1. The sample size
2. The ID identifier
3. The X coordinate
4. The Y coordinate
5. The "I_i" value
6. The expected "I_i".

If the variance box is checked, the program will also calculate the variance, standard error, and a Z-test of "I_i" for each zone. The default is for the variance not to be calculated.

Simulation of Confidence Intervals for Anselin's Local Moran

There are two ways to estimate confidence intervals for Anselin's Local Moran. First, the routine can calculate the variance and, for each zone, the standardized "I_i" score to produce a Z-test of the significance of the "I_i". Assuming the sample size is greater than 120, 95% percent confidence intervals can be estimated by:

$$95\%\ confidence\ intervals = I_i \pm 1.98SE_i \tag{9.7}$$

and 99% confidence intervals can be estimated by:

$$99\%\ confidence\ intervals = I_i \pm 2.58SE_i \tag{9.8}$$

One problem with this test is that "I_i" may not actually follow a normal standard distribution. That is, if "I_i" is calculated for all zones with random data, the distribution of the statistic may not be (and often will not be) normally distributed. This would be especially true if the variable of interest, Z, is skewed with some zones having very high values while the majority having low values, as is typically true with crime distributions.

Second, the user can estimate confidence intervals (called *credible intervals*) using a Monte Carlo simulation. A *permutation* type simulation is run whereby the locations of the zones are kept and the original values of the intensity variable, Z, are maintained but randomly re-assigned to zones for each simulation run. This will maintain the structure of the attribute "Z" variable but will estimate the value of "I_i" for each under random assignment of this variable.

Note that a simulation may take time to run especially if the data set is large or if a large number of simulation runs are requested.

If a permutation Monte Carlo simulation is run to estimate credible intervals, specify the number of simulations to be run (e.g., 1,000, 5,000, 10000). In addition to the “I_i” for each zone, the expected “I_i” and the variance (if requested), the output includes the results that were obtained by the simulation for:

1. The minimum “I_i” value
2. The maximum “I_i” value
3. The 0.5 percentile of “I_i”
4. The 2.5 percentile of “I_i”
5. The 97.5 percentile of “I_i”
6. The 99.5 percentile of “I_i”

The two percentile pairs (2.5 and 97.5; 0.5 and 99.5) create approximate 95% and 99% credible intervals respectively. The minimum and maximum “I_i” values create an ‘envelope’ around each zone. It is important to run enough simulations to produce reliable estimates.

The tabular results can be printed, saved to a text file or saved as a '.dbf' file with a *LMoran<root name>* prefix with the root name being provided by the user. For the latter, specify a file name in the “Save result to” in the dialogue box. The ‘dbf’ file can then be linked to the input ‘dbf’ file by using the ID field as a matching variable. This would be done if the user wants to map the “I_i” variable, the Z-test, or those zones for which the “I_i” value is either higher than the 97.5 or 99.5 percentiles or lower than the 2.5 or 0.5 percentiles of the simulation results.

Example 1: Local Moran Statistics for Baltimore Auto Thefts

Using data on 14,853 motor vehicle thefts for 1996 in both Baltimore County and Baltimore City, the number of incidents occurring in each of 1,349 census block groups was calculated (Figure 9.2). As seen, the pattern shows a higher concentration towards the center of the metropolitan area, as would be expected, but that the pattern is not completely uniform.

There are many block groups within the City of Baltimore with very low counts of auto thefts and there are block groups within the County with very high counts. Using these data, *CrimeStat* calculated the Local Moran statistic with the variance box checked and the small distance adjustment used. The range of I_i values varied from -37.26 to +180.14 with a mean of 5.20. The standardized Local Moran ‘Z’ varied from -12.71 to 50.12 and with a mean of 1.61. Figure 9.3 maps the distribution. Because a negative I_i value indicates dissimilarity, these values have been drawn in red compared to blue for a positive I_i value. As seen, in both the City of Baltimore and the County of Baltimore, there are block groups with large negative I_i values, indicating that they differ from the surrounding block groups.

Figure 9.2:

1996 Motor Vehicle Thefts

Number of Auto Thefts Per Block Group

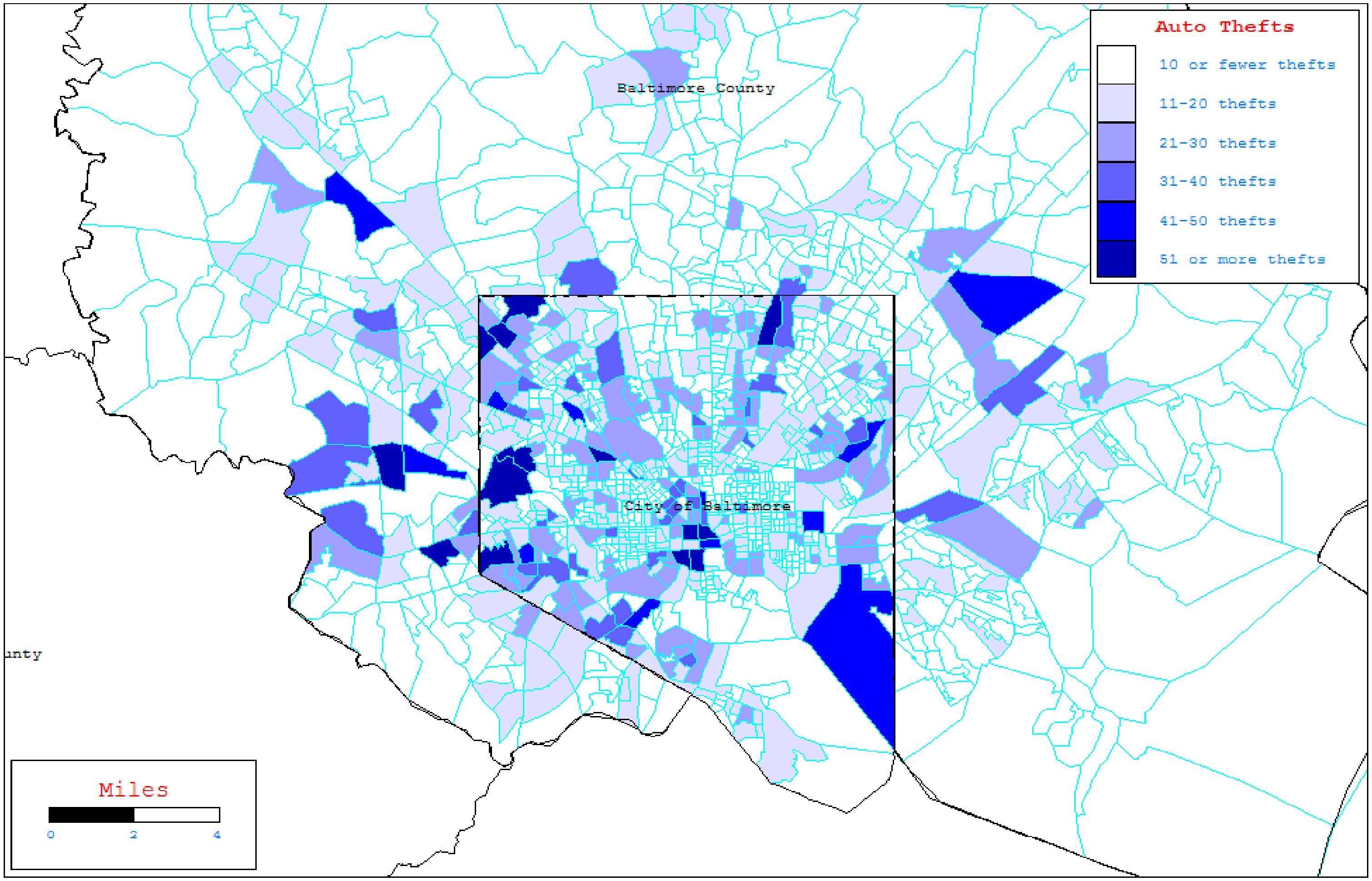
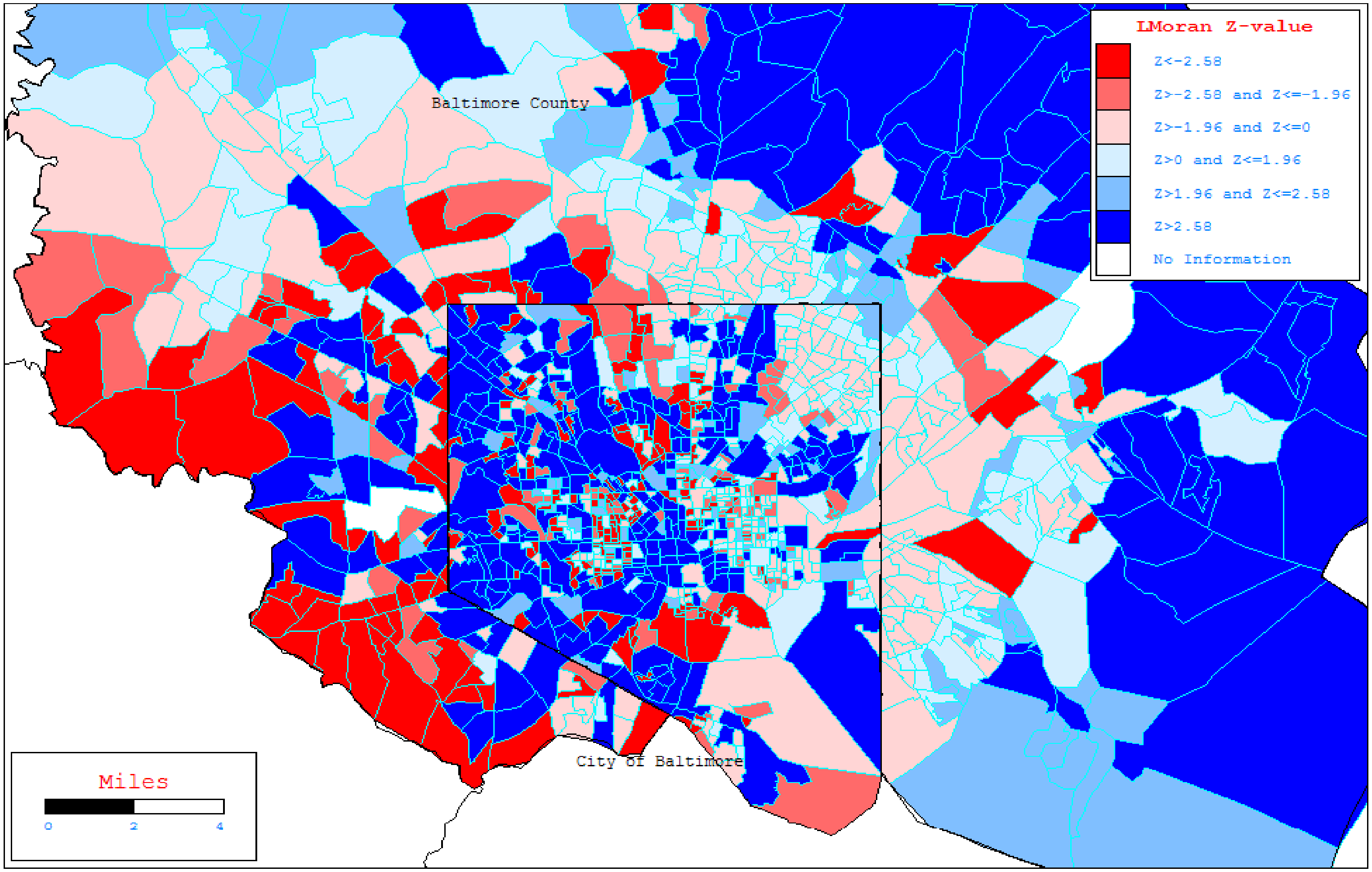


Figure 9.3:

Local Spatial Autocorrelation of 1996 Vehicle Thefts

Local Moran Z-Value of Block Groups



For example, in the central part of Baltimore City, there is a small area of about eight block groups with low numbers of auto thefts, compared to the surrounding block groups. These form a ‘cold spot’. Consequently, they appear in dark tones in Figure 9.3 indicating that they have high I_i values (i.e., negative spatial autocorrelation). Similarly, there are several block groups on the western side of the County which have relatively high numbers of auto thefts compared to the surrounding block groups. They form a hot spot. Consequently, they also appear in dark tones in Figure 9.3 because this indicates positive spatial autocorrelation, having values that are similar to the surrounding blocks. In other words, similarity is shown in blue and dissimilarity in red.

Example 2: Simulated Local Moran Confidence Intervals for Houston Burglaries

To illustrate the simulated confidence intervals, we apply the Local Moran statistic to burglaries in the City of Houston shown in figure 9.4. The data were 26,480 burglaries that occurred in 2006. They were aggregated to 1,179 traffic analysis zones (TAZ). Anselin’s Local Moran statistic was calculated on each of the TAZ’s with 1,000 Monte Carlo simulations being calculated. Figure 9.5 shows a map of the calculated local “I_i” values. It can be seen that there are many more zones of positive spatial autocorrelation where the zones are similar to their neighbors. In most of these cases, the zone has few burglaries whereas it is surrounded by zones that also have few burglaries. A few zones have negative spatial autocorrelation. In most of the cases, the zones have many burglaries and are surrounded by zones with few burglaries.

Confidence intervals were calculated in two ways. First, the theoretical variance was calculated and a Z-test computed. This is done in *CrimeStat* by checking the ‘theoretical variance’ box. The test assumes that “I_i” is normally distributed, which may or may not be a valid assumption. Second, a Monte Carlo simulation was used to estimate the 99% confidence intervals (i.e., outside the 0.5 and 99.5 percentiles).

Table 9.1 shows the results for four records. The four records illustrate different combinations. In the first record (TAZ 522), the “I_i” value is 0.000373, indicating positive spatial autocorrelation (i.e., nearby zones have similar values). Comparing it to the 95% credible intervals, it is larger than the 97.5th percentile. In addition, the Z-test, based on the theoretical variance, is positive. Thus, both the simulated confidence intervals and the theoretical confidence interval indicate that the “I_i” for this zone is significant.

Figure 9.4:

Burglaries in Houston: 2006

Number of Burglaries by Traffic Analysis Zones

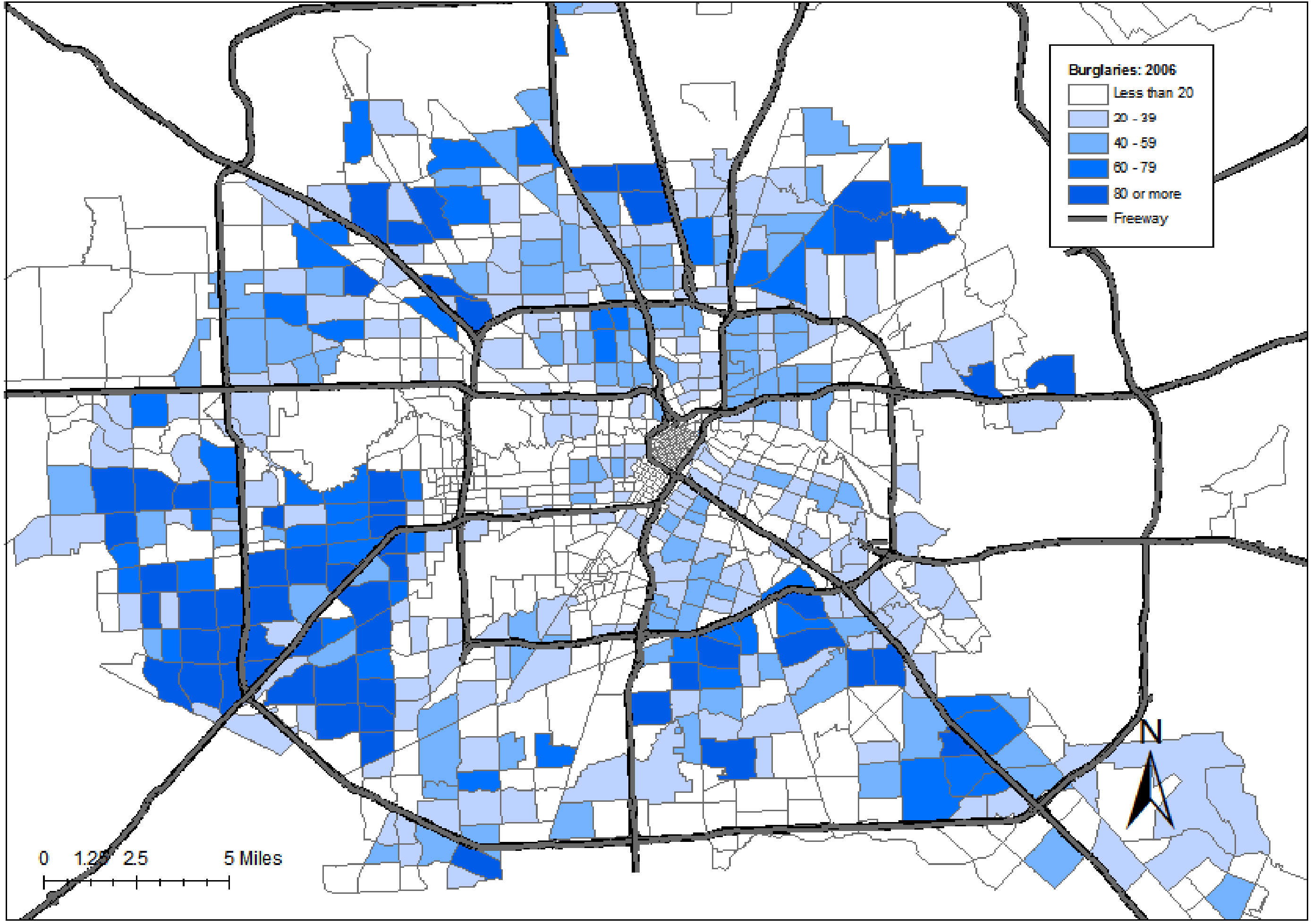


Figure 9.5:

Burglary Hot Spots in Houston: 2006

Local Moran "I" for Traffic Analysis Zones

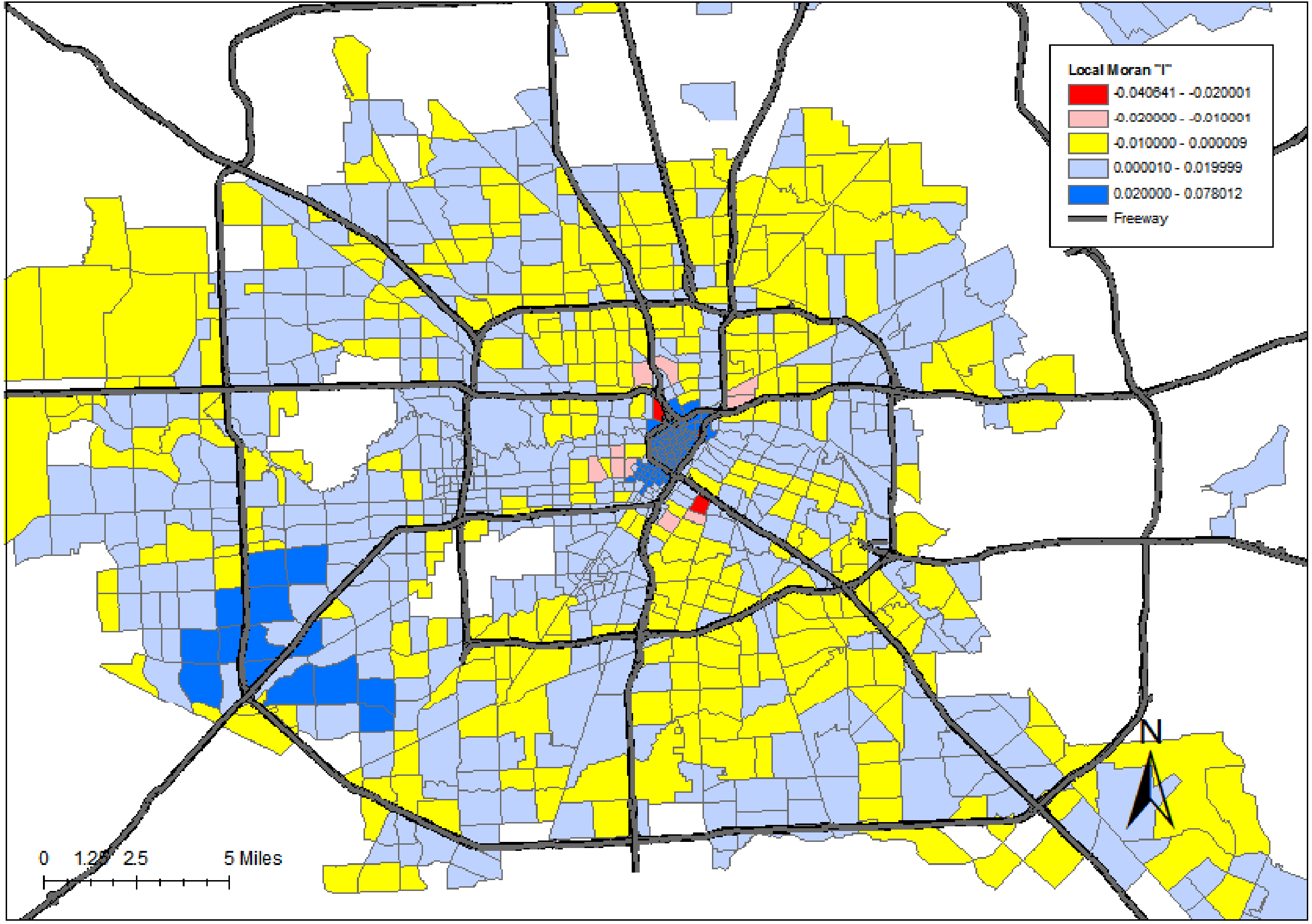


Table 9.1:
Anselin's Local Moran 95% Confidence & Credible Intervals
4 Cases Estimated from Theoretical Variance and from Monte Carlo Simulation

					Simulated		*Theoretical*	
TAZ	**X**	**Y**	**“I_i”**	**Expected**	**0.5 %**	**97.5 %**	**Z-test**	**p**
522	3152030	13941900	**0.000373**	-0.000010	-0.000856	0.000216	2.29	0.05
534	3200630	13955800	**0.000345**	-0.000007	-0.000516	0.000226	1.82	n.s.
182	3126150	13842900	**-0.040641**	-0.000087	-0.014287	0.007292	-9.69	0.0001
384	3156740	13879400	**-0.000886**	-0.000018	-0.001259	0.000593	-2.20	0.05

In the second record (TAZ 534), the “I_i” value is 0.000345, also indicating positive spatial autocorrelation. However, the “I_i” value is greater than the 97.5th percentile, indicating that the simulation suggests the “I_i” is greater than what would be expected by chance. On the other hand, the Z-test, based on the theoretical distribution, is not significant. Thus, there is an inconsistency between simulation test and the Z-test.

In the third record (TAZ 182), there is consistency between the simulated and theoretical significance tests. The “I_i” is negative (-0.040641), indicating negative spatial autocorrelation (i.e., the has different values than nearby zones). The simulation shows that the “I_i” is more negative than the simulated 5th percentile and the Z-test is also significantly negative.

The fourth record (TAZ 384) shows a negative “I_i”, indicating negative spatial autocorrelation (i.e., nearby zones have different values). But there is inconsistency in the test. The simulation shows that this “I_i” falls between the 5th and 97.5th percentiles, indicating non-significance, whereas the Z-test suggests the “I_i” is significant.

In general, simulated confidence intervals will be similar to the theoretical ones. But, there can be discrepancies. The reason is that the sampling distribution of “I_i” may not be (and probably is not) normally distributed. Of the 1,179 traffic analysis zones, 661 showed significant “I_i” values according to the simulated 99% credible intervals (i.e., either equal to or smaller than the 0.5 percentile or equal to or greater than the 99.5 percentile) while 688 of the zones showed significant “I_i” values according to the theoretical Z-test at the 99% level (i.e., having a Z-value equal to or less than -2.58 or equal to or greater than 2.58). It would behoove the user to estimate the number of zones that are significant according to both the simulated and theoretical confidence intervals before making a decision as to which criterion to use.

Therefore, both the simulated confidence interval and the theoretical distribution should be used with caution. The best mapping solution may be to map only those zones that are highly significant with both tests showing substantial significance. Or, alternatively, map only those zones with the highest positive or highest negative “I_i” values.

Uses of Anselin’s Local Moran

Anselin’s Local Moran has a number of uses. First, it can identify zones that are different (dissimilar) from its neighbors. This can be a good first step in finding locations that either have higher crime numbers (a hot spot) or lower crime numbers (a cold spot) than the neighboring areas. This can focus police efforts on identifying the problems that cause the zone to be higher in the case of a hot spot or to identify factors that mitigate crime in the case of a cold spot.

Second, another use of Anselin’s Local Moran statistic is to identify ‘outliers’, zones that are very different from their neighbors. In this case, zones with a high negative I value (e.g., with an “I_i” smaller than two standard deviations below the mean) are indicative of outliers. They either have a high number of incidents whereas their neighbors have a low number or, the opposite, a low number of incidents amidst zones with a high number of incidents. Identifying the outliers can focus on zones that are unique (and which should be studied) or, in multivariate analysis, on zones that need to be statistically treated differently in order to minimize a large modeling error (e.g., creating a dummy variable for the extreme outliers in a regression model).

In short, the Local Moran statistic can be a useful tool for identifying zones that are dissimilar from their neighborhood. To use the Local Moran statistic, however, requires that the data be summarized into zones in order to produce the necessary intensity value. Given that most crime incident databases will list individual events without intensity or weight values assigned, this will entail additional work by a law enforcement agency.

Limitations of Anselin’s Local Moran

There are several limitations to the method. First, because it is an index of similarity, a positive “I_i” value does not necessarily indicate a hot spot. The positive “I_i” value could be due to zones with low values of the intensity variable surrounded by other zones that also have low values. Thus, in terms of using the method to identify hot spots of zones can lead to ambiguous results. It is best seen as a first step in identifying hot spot zones.

Second, there are concerns about the statistical criterion used to identify a zone as being similar or dissimilar to its neighbors. One has to be suspect about a technique that finds significance in more than half the cases. It would probably be more conservative to use 99% confidence intervals for identifying zones that show positive or negative spatial autocorrelation rather than using 95% confidence intervals or, better yet, choosing only those zones that have very negative or very positive “I_i” values. Unfortunately, this characteristic of Anselin’s local Moran is also true of the local Getis-Ord statistic, which is discussed below. The significance tests, whether simulated or theoretical, are not strict enough and, thereby, increase the likelihood of a Type I (false positive) error. A user must be very careful in interpreting “I_i” values for individual zones and would be better served choosing only the very highest or lowest.

For a detailed discussion of problems in conducting tests on local spatial autocorrelation statistics, such as the local Moran or Getis-Ord Local “G” (to be discussedbelow), see Waller and Gottway (2004; p. 238).

Getis-Ord Local “G”

The Getis-Ord Local G statistic applies the Getis-Ord "G" statistic to individual zones to assess whether particular zoness are spatially related to the nearby zones (see Chapter 5). Unlike the global Getis-Ord “G” but like Anselin’s Local Moran, the Getis-Ord Local “G” is applied to each individual zone. The formulation presented here is taken from Wong and Lee (2005). The “G” value is calculated with respect to a specified search distance (defined by the user), namely:

$$G_i(d) = \frac{\sum_i W_{ij}(d) X_j}{\sum_j X_j} \tag{9.9}$$

$$E[G_i] = \frac{W_i}{(N-1)} \tag{9.10}$$

$$Var(G_i) = E(G_i^2) - [E(G_i)]^2 \tag{9.11}$$

$$E[G_i^2] = \frac{1}{(\sum_j X_j)^2}\left[\frac{W_i(n-1-W_i)\sum_j X_j^2}{(N-1)(N-2)}\right] + \frac{W_i(W_i-1)}{(N-1)(N-2)} \tag{9.12}$$

where $\boldsymbol{w_j}$ is the weight of zone “j” from zone “i”, $\boldsymbol{W_i}$ is the sum of weights for zone “i”, and $\boldsymbol{n}$ is the number of cases.

The standard error of G(d) is the square root of the variance of G. Consequently, a Z-test can be constructed by:

$$S.E.[G(d)] = \sqrt{Var[G(d)]} \tag{9.13}$$

$$Z[G(d)] = \frac{G(d) - E[G(d)]}{S.E.[G(d)]} \tag{9.14}$$

A good example of using the Getis-Ord local "G" statistic in crime mapping is found in Chainey and Racliffe (2005, pp. 164-172).

ID Field

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

Search Distance

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

Getis-Ord Local "G" Simulation of Confidence Intervals

Since the Getis-Ord "G" statistic may not be normally distributed, the significance test is frequently inaccurate. Instead, a *permutation* type Monte Carlo simulation can be run whereby the original values of the intensity variable, Z, for the zones are maintained but are randomly re-assigned to zones for each simulation run. This will maintain the distribution of the variable Z but will estimate the value of G for each zone under random assignment of this variable. Specify the number of simulations to be run (e.g., 100, 1000, 10000).

Output for Each Zone

The output is for each zone includes:

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

and if a simulation is run:

10. The 0.5 percentile of "G" for the zone

11. The 2.5 percentile of “G” for the zone
12. The 97.5 percentile of “G” for the zone
13. The 99.5 percentile of “G” for the zone

The two pairs of percentiles (5 and 95; 2.5 and 97.5; 0.5 and 99.5) create approximate 95% and 99% credible intervals respectively around each zone. The minimum and maximum “G” values create an ‘envelope’ around each zone. However, unless a large number of simulations are run, the actual “G” value may fall outside the envelope for any zone. The tabular results can be printed, saved to a text file or saved as a '.dbf' file. For the latter, specify a file name in the “Save result to” in the dialogue box. The file is saved with a *LGetis-Ord<root name>* prefix with the root name being provided by the user.

The ‘dbf’ output file can be linked to the Primary File by using the ID field as a matching variable. This would be done if the user wants to map the “G” variable, the expected “G”, the Z-test, or those zones for which the “G” value is either higher than the 97.5 or 99.5 percentiles or lower than the 2.5 or 0.5 percentiles of the simulation results respectively (95% or 99% confidence intervals).

Example: Testing Houston Burglaries with the Getis-Ord Local “G”

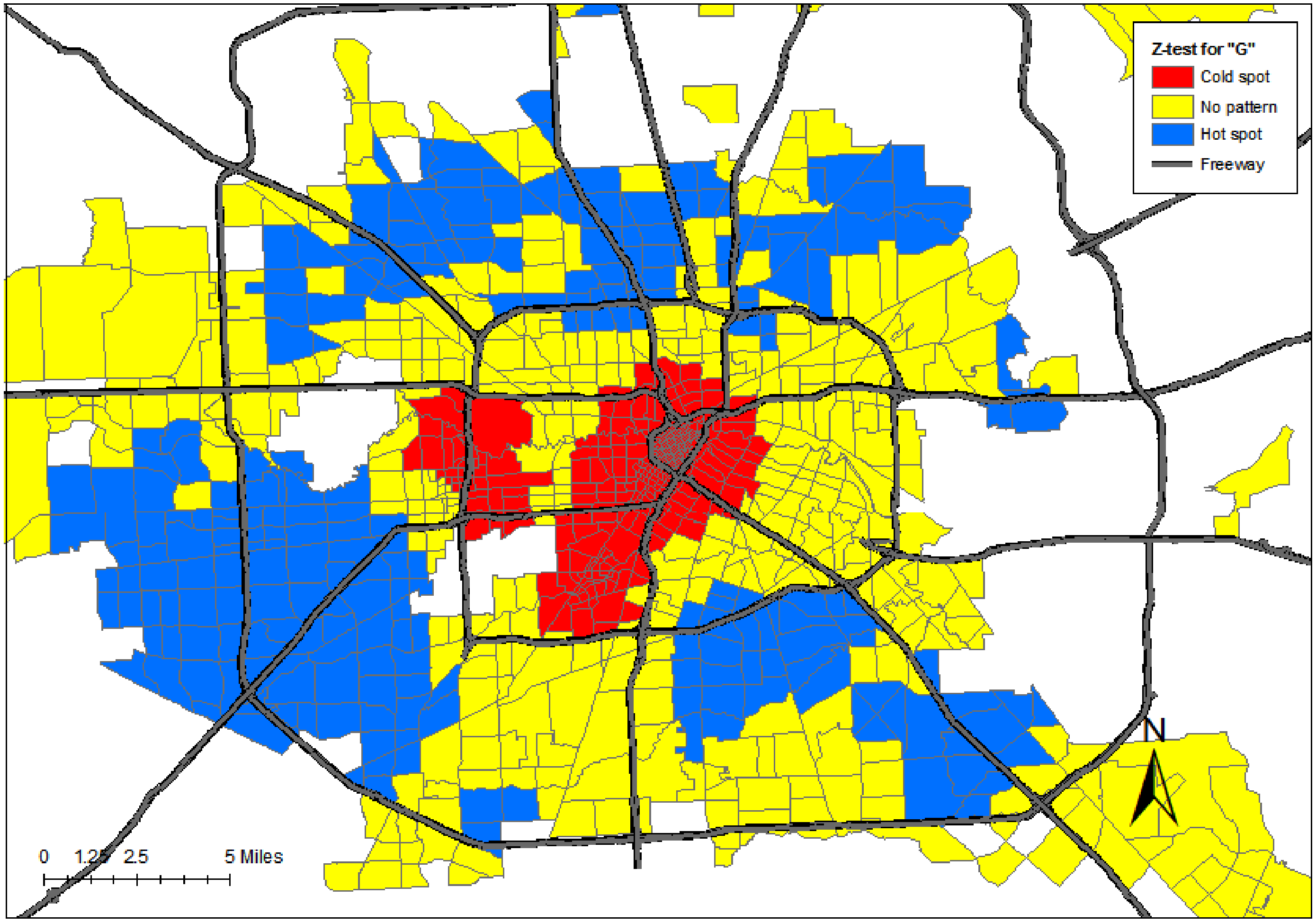
Using the same data set on the Houston burglaries as above, the Getis-Ord Local “G” was run with a search radius of 2 miles. The output file was then linked to the input file using the ID field to allow the mapping of the local “G” values. Figure 9.6 illustrates the Z-test of the Getis-Ord Local “G” for different zones. The map displays the significance of the Z-test (the difference between the “G” and the expected “G” relative to the standard error of “G”). Zones with a Z-test of +1.96 or higher are shown in blue (hot spots). Zones with Z-tests of -1.96 or smaller are shown in red (cold spots) while zones with a Z-test between -1.96 and +1.96 are shown in yellow (no pattern).

As seen, there are some very distinct patterns of zones with high positive spatial autocorrelation and low positive spatial autocorrelation. Examining the original map of burglaries by TAZ (Figure 9.4), it can be seen that where there are many burglaries, the zones tend to show high positive spatial autocorrelation (hot spots) in Figure 9.6. Conversely, where there are few burglaries, the zones show either low positive spatial autocorrelation (‘cold spots’) or, more commonly, no pattern in Figure 9.6. In particular, the greater downtown Houston area, and area southwest of downtown that includes the Texas Medical Center and a commercial area west of downtown around the IH 610 ‘loop’ show areas of significant ‘cold spots’. These are areas dominated by commercial or office buildings and generally have relatively few burglaries.

Figure 9.6:

Burglary Hot Spots in Houston: 2006

Z-test of Getis-Ord "G" with 2 Mile Search Radius by Traffic Analysis Zones



Uses of the Getis-Ord Local “G”

The Getis-Ord Local “G” is very good at identifying hot spots and also good at identifying cold spots. As mentioned, Anselin’s Local Moran can only identify positive or negative spatial autocorrelation, that is, whether the zones are similar or dissimilar. Those zones with positive spatial autocorrelation could occur because zones with high values are nearby other zones with high values or they could occur because zones with low values are nearby other zones with low values. The Getis-Ord Local “G” can distinguish those two types.

Limitations of the Getis-Ord Local “G”

The biggest limitation with the Getis-Ord Local “G”, which also applies to the global Getis-Ord and Getis-Ord Correlogram routines (see Chapter 5), is that it cannot detect negative spatial autocorrelation where a zone is surrounded by neighbors that are different (either having a high value surrounded by zones with low values or having a low value and being surrounded by zones with high values). In actual use, both the Anselin’s Local Moran and the Getis-Ord Local “G” should be used to produce a full interpretation of the rsults.

Another limitation is that the significance tests are too weak, allowing too many zones to show significance. In the data shown in Figure 9.6, 63% of the zones (740) were statistically significant by the Z-test! A simulation of credible intervals also showed a very high proportion having G values greater or less than the 95% credible intervals. Thus, there is a substantial Type I error with this statistic (false positives), a similarity it shares with Anselin’s Local Moran.

Reducing the search radius will reduce the number of zones with significant Z-scores. For example, with a 1 mile search radius, only 44% of the zones were statistically significant by the Z-test. But, given the size of the zones, there is a limit to how small a search radius can be made. With the Houston block groups, for example, the average area of a block group is 0.48 square miles. If a typical block group size is viewed as a square having that area, then each side would be about 0.7 miles in length. Choosing a search radius smaller than 0.7 would end up with many zones not having neighbors selected, especially farther away from the city center where zones are generally much larger in size. This would lead to an unrealistic estimate of the amount of spatial autocorrelation. In other words, there is a trade-off between the precision of the search radius and the accuracy of the “G” estimate. In this case, a search radius of two miles is a realistic search radius for this geographical distribution.

Waller and Gottway (2004, p. 238) point out that there are four problems with the testing of LISA statistics since the measures are interrelated: First, the distributional properties remain largely unknown. Second, multiple tests lead to overly rejecting the null hypothesis, which we have demonstrated above. Third, the LISA's of neighboring zones are often highly correlated due to using the same data and, fourth, many of the tests are based on small samples sizes since the number of events in any one zone may be limited. A random simulation can overcome the first problem by using the empirical distribution as a basis for calculating credible intervals, but it cannot overcome the next three.

In short, a user should be very careful in interpreting zones with significant "G" values and would probably be better served by choosing only those zones with the highest or lowest "G" values.

Zonal Nearest Neighbor Hierarchical Clustering

The zonal nearest neighbor hierarchical spatial clustering routine applies the nearest neighbor hierarchical clustering algorithm (Nnh; see Chapter 7 for the background and details) to zonal data. The point-based Nnh is a constant-distance clustering routine that groups points together on the basis of spatial proximity. A threshold distance is defined and the minimum number of points that are required for each cluster specified. The output can be displayed with ellipses or convex hulls.

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

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

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

For example, if the attribute to be grouped is the number of crimes in a zone, then the routine identifies adjacent zones that have high concentrations of crimes. The user can modify the number of clusters identified and the relative size of them by changing the search radius or the minimum number of attributes that must be grouped together. The results can be output as either standard deviational ellipses or convex hulls.

Weighting Variable

Each zone must be weighted by an attribute variable. This is the weight or intensity variable defined on the Primary File page. The user specifies whether the weight or the intensity variable is to be used for the attribute. The default is Intensity.

Clustering Criteria

Two criteria are used to group zones together.

Criterion 1: Threshold Distance

The first criterion in identifying clusters is whether zones are closer than a specified threshold distance. There are two alternatives in selecting the threshold distance: 1) a fixed distance (the default is 2 miles); or 2) a random nearest neighbor distance.

Fixed distance

Unlike the Nnh routine for clustering points (Chapter 7), the default alternative for selecting a threshold distance in the Znnh is to choose a fixed distance (in miles, nautical miles, feet, kilometers, or meters). The user checks the “Fixed distance” box and selects a threshold distance. The default value is 2 miles but the user can change this.

The main advantage of this method is that, first, the search radius can be specified exactly and, second, unlike points, zones do not overlap and are spatially dispersed. The distance between adjacent zones may be substantial especially for large zones at the periphery of an urban area. Thus, to capture adjacent zones that have high values of the attribute variable requires choosing a search radius that is large.

The main disadvantage of this method is that the choice of a threshold is subjective. There is no reason why any particular search radius should be chosen. Further, the larger the distance that is selected, the greater the likelihood that clusters will be found by chance. This can be tested using a Monte Carlo simulation (see below).

Random nearest neighbor distance

The alternative is to use the expected random nearest neighbor distance for first-order nearest neighbors. The user specifies a *one-tailed* confidence interval around the random expected nearest neighbor distance. The t-value corresponding to this probability level, t, is selected from the Student's t-distribution under the assumption that the degrees of freedom are at least 120.[2] This selection is controlled by a slide bar under the routine (see Figure 9.1). From Chapter 6, the mean random distance is defined as:

$$d_{NN(ran)} = 0.5\sqrt{\frac{A}{N}} \tag{9.15}$$

where A is the area of the region and N is the number of zones and the standard error of the mean random distance is:

$$SE_{d(ran)} \cong \sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \tag{9.16}$$

where A is the area of the region and N is the number of zones. The confidence interval around that distance is defined as:

$$Confidence\ interval = \ d_{NN(ran)} \pm t * SE_{d(ran)} \tag{9.17}$$

where t is the t-value associated with a probability level in the Student's t-distribution.

The approximate lower limit of this confidence interval is:

$$Lower\ limit\ of\ confidence\ interval = \ d_{NN(ran)} - t * SE_{d(ran)}$$

$$\cong 0.5\sqrt{\frac{A}{N}} - t\sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \tag{9.18}$$

[2] This is the next highest degree of freedom in the Student's t-table below infinity.

and the upper limit of this confidence interval is:

$$Upper\ limit\ of\ confidence\ interval = \ d_{NN(ran)} + t * SE_{d(ran)}$$

$$\cong 0.5\sqrt{\frac{A}{N}} + t\sqrt{\frac{(4-\pi)A}{4\pi N^2}} = \frac{0.26136}{\sqrt{\frac{N^2}{A}}} \qquad (9.19)$$

The confidence interval defines a probability for the distance between any *pair* of zones. For example, for a specific *one-tailed* probability, *p*, fewer than *p*% of the zones would have nearest neighbor distances smaller than this selected limit *if* the distribution was spatially random. *If* the data were spatially random and if the mean random distance is selected as the threshold criteria (the default position on the slide bar), approximately 50% of the pairs will be closer than this distance. For randomly distributed data, if a p≤.05 level is taken for t (two steps to the left of the default or the fifth in from the left), then only about 5% of the pairs would be closer than the threshold distance. Similarly, if a p≤.75 level is taken for t (one step to the right of the default or the fifth in from the right), then about 75% of the pairs would be closer than the threshold distance.

Table 9.2:
Approximate Probability Values Associated with Threshold Scale Bar

Position	Scale Bar Probability	Description
1	0.00001	Far left point of slide bar
2	0.0001	Second from left
3	0.001	Third from left
4	0.01	Fourth from left
5	0.05	Fifth from left
6	0.1	Sixth from left
7	0.5	Sixth from right (default value)
8	0.75	Fifth from right
9	0.9	Fourth from righ
10	0.95	Third from righ
11	0.99	Second from righ
12	0.999	Far right point of slide bar

In other words, the threshold distance is a probability level for selecting any *two* zones (a pair) on the basis of a chance distribution. The slide bar has 12 levels and is associated with a probability level for a t-distribution from a sample of 120 or larger. From the left, the p-values are approximately (see Table 9.2 above):

Taking a broader conception of this, if there is a spatially random distribution, then for all distances between pairs of zones, of which there are

$$Combinations = \frac{N(N-1)}{2} \tag{9.18}$$

fewer than *p*% will be shorter than this threshold distance.

Area must be defined correctly

Note that it is *very* important that area be defined correctly for this routine to work. If the user defines the area on the measurement parameters page (see Chapter 3), the Znnh routine uses that value to calculate the threshold distance. If the user does not define the area on the measurement parameters page, the routine calculates the area from the minimum and maximum X/Y values (the bounding rectangle), which will usually be a larger area. In either case, the routine will be able to calculate a threshold distance and run the routine.

However, if the area units are defined incorrectly on the measurement parameters page, then the routine will certainly calculate the threshold distance wrongly. For example, if data are in feet but the area on the measurement parameters page are defined in square miles, most likely the routine will not find any zones that are farther apart the threshold distance since that distance is defined in miles. In other words, it is essential that the area units be consistent with the data for the routine to properly work.

Criterion 2: Zones with the Highest Number of Attributes

The second criterion involves the weighting of each zone. With zonal data, each zone has an attribute value, defined either by the intensity variable or weight variable on the Primary File page. Clusters are defined by those zones that are within the threshold distance but which have the highest combined value of the attribute variable. The algorithm looks for a ‘center’ of three of more zones for which the total value of the attribute variable is highest. Like the Nnh routine, the process is iterative, first finding an approximate center and then re-calculating it with respect to the total value of the attribute variable for those zones within the threshold distance of the center. Eventually, the process stabilizes and the routine quits.

Table 9.3 presents a simple example. Suppose there are two zones (A and B) within the second matrix and each has three other zones closer than the threshold distance (C, D, E for Zone A and F, G, H for Zone B). In this example, Zone A would be chosen as the initial center for the first cluster because the sum of the weights (for itself and for the three other zones that are within the threshold distance) add to 85 whereas the sum of the weights for the other points for Zone B only add to 65 even though Zone B had a higher weight for itself than Zone A.

Table 9.3:

Example of Weighting Pairs of Zones by Attributes

	Zone A		Zone B
Other Zones	**Weighting**	**Other Zones**	**Weighting**
A (itself)	10	B (itself)	20
C	20	F	10
D	30	G	15
E	25	H	20
	---		---
TOTAL:	**85**		**65**

The routine then removes the zones selected for the first cluster (A, C, D, and E). It then attempts to find a second cluster. In this example, there is only one other (B, F, G, and H), which is then removed from the matrix. If there were more zones, the routine would look for additional centers of clusters.

Having completed an initial identification of cluster centers, the routine then calculates the center of minimum distance (CMD) for the selected points and then calculates those zones that are within the threshold distance of the CMD. It repeats the process for a second cluster. After a second round of clustering, the routine repeats the process for a third cluster. The iterations continue until no zones change clusters and the calculated center of minimum distance changes very little.

First-order Clusters

Using these criteria, *CrimeStat* constructs a first-order clustering of the zones. For each first-order cluster, the center of minimum distance is output as the cluster center, which can be saved as a '.dbf' file.

Second and Higher-order Clusters

The first-order clusters are then tested for second-order clustering. The procedure is similar to first-order clustering except that the cluster centers (the center of minimum distance for each) are now treated as 'points' which themselves are clustered (see endnote *ii*). The process is repeated until no further clustering can be conducted. Either all sub-clusters converge into a single cluster, the threshold distance criterion fails, or there are fewer than four seeds in the higher-order cluster.

Note that this process is similar to that of the Nnh routine discussed in Chapter 7 except the selection of clusters is function of the total value of the attribute variable and not just the distance between zones.

Simulating Confidence Intervals

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

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

Type of Graphical Output

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

Ellipse cluster output

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

First and higher-order ellipses will be output as separate objects. The prefix will be 'Znnh1' for the first-order ellipses, 'Znnh2' for the second-order ellipses, and 'Znnh3' for the third-order ellipses. Higher-order ellipses will only index the number.

Output size for ellipses

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

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

Convex hull cluster output

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

The name will be output with a 'CZnnh1' prefix for the first-order clusters, a 'CZnnh2' prefix for the second-order clusters, and a 'Cznnh3' prefix for the third-order clusters. Higher-order clusters will index only the number.

Note that unlike the Nnh clustering algorithm for points, discussed in Chapter 7, the zonal Nnh generally has much larger search areas. Consequently the convex hulls will be much larger than the ellipses, even the 2x ellipse (the opposite is true with the Nnh).

Tabular Output

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

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

and if a simulation is run:

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

Example 1: Simulated Clustering of Zones

To illustrate the Znnh routine, a dispersed cluster structure for an arbitrary variable with five main groupings was created with 1,179 City of Houston Traffic Analysis Zones (TAZ). The five clusters can be labeled as central, southwest, northwest, northeast and southeast. Figure 9.7 illustrates the pattern that was created.

Four separate search areas were selected with a minimum of 25 'events' being required of the attribute variable:

1. 2 miles
2. 5 miles
3. 8 miles
4. 12 miles

Figures 9.8-9.12 illustrate the results of the clustering using these search distances with the standard deviational ellipse. Figure 9.11 also shows the convex hull of the search radius. Notice that a search radius of 2 miles produces small clusters and did not cover the clusters in the northeast, the southeast and most of the southwest. The reason is that TAZs for those areas are quite large with many being larger than 2 miles.

A 5 mile search radius covered the five clusters though the clusters are still small. The 8 mile search radius appeared to fit the data better while the 12 mile search radius produced too large ellipses with one large one for the central area. Note that Figure 9.11 shows the convex hulls of the 8 mile search radius and which covers most of the TAZs of the City of Houston.

Example 2: Clustering of Houston Burglaries by Traffic Analysis Zones

The second example examines burglaries in the City of Houston in 2006. In that year, 24,935 burglaries were recorded. The data from which these came were assigned to blocks. Each of the burglaries was geocoded to the mid-block and then aggregated into 1,179 TAZs. Figure 9.4 above illustrates the pattern of burglaries in Houston.

The Znnh routine was run with four different search radii and with a minimum of 25 burglaries being required for each cluster:

1. 0.5 miles
2. 2 miles
3. 5 miles
4. 8 miles

Figure 9.7:

Test of Znnh Routine

Arbitrary Dispersed Variable

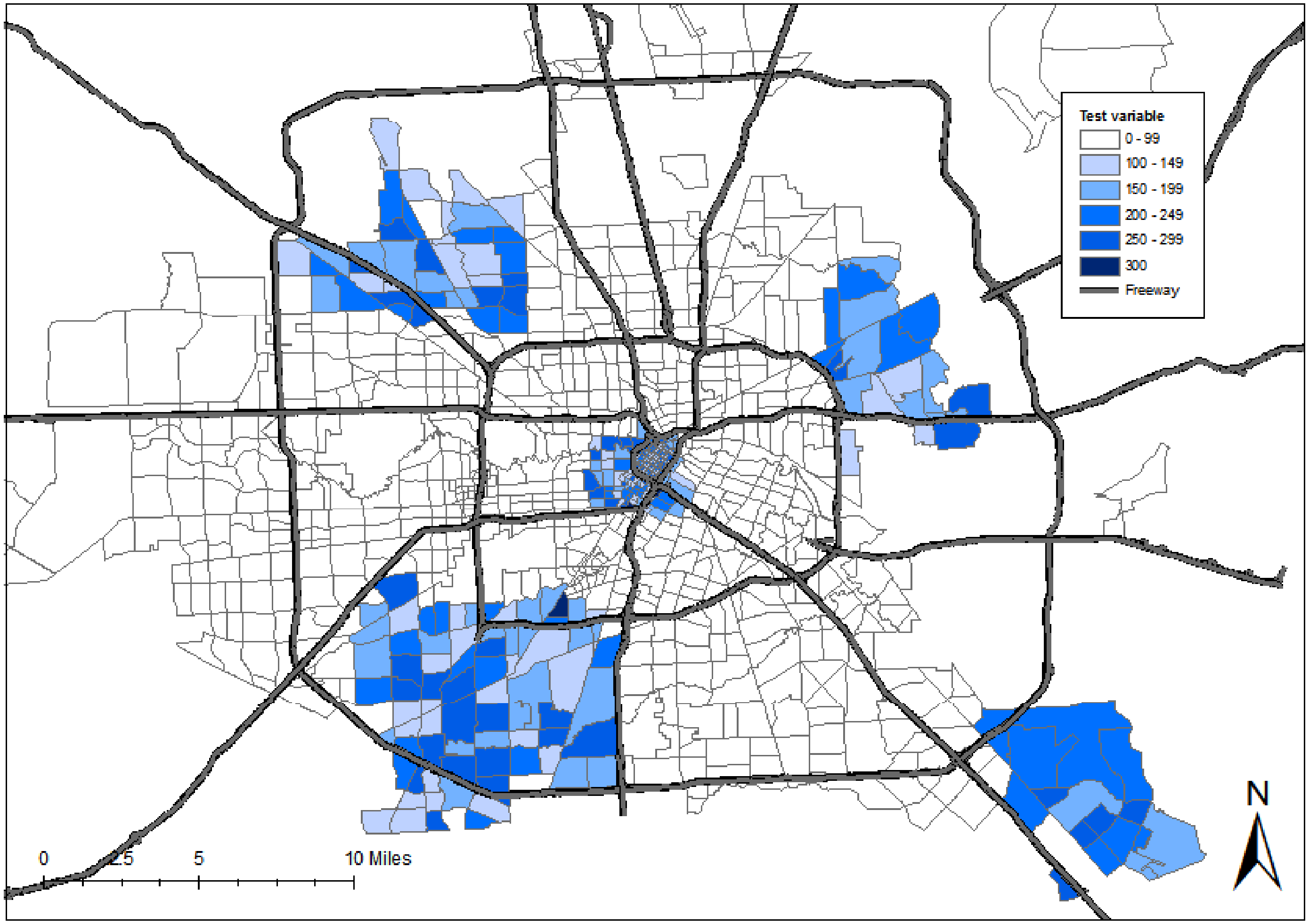


Figure 9.8:

Test of Znnh Routine

Identified Hot Spots with 2 Miles Search Radius and Minimum Number of Events=25

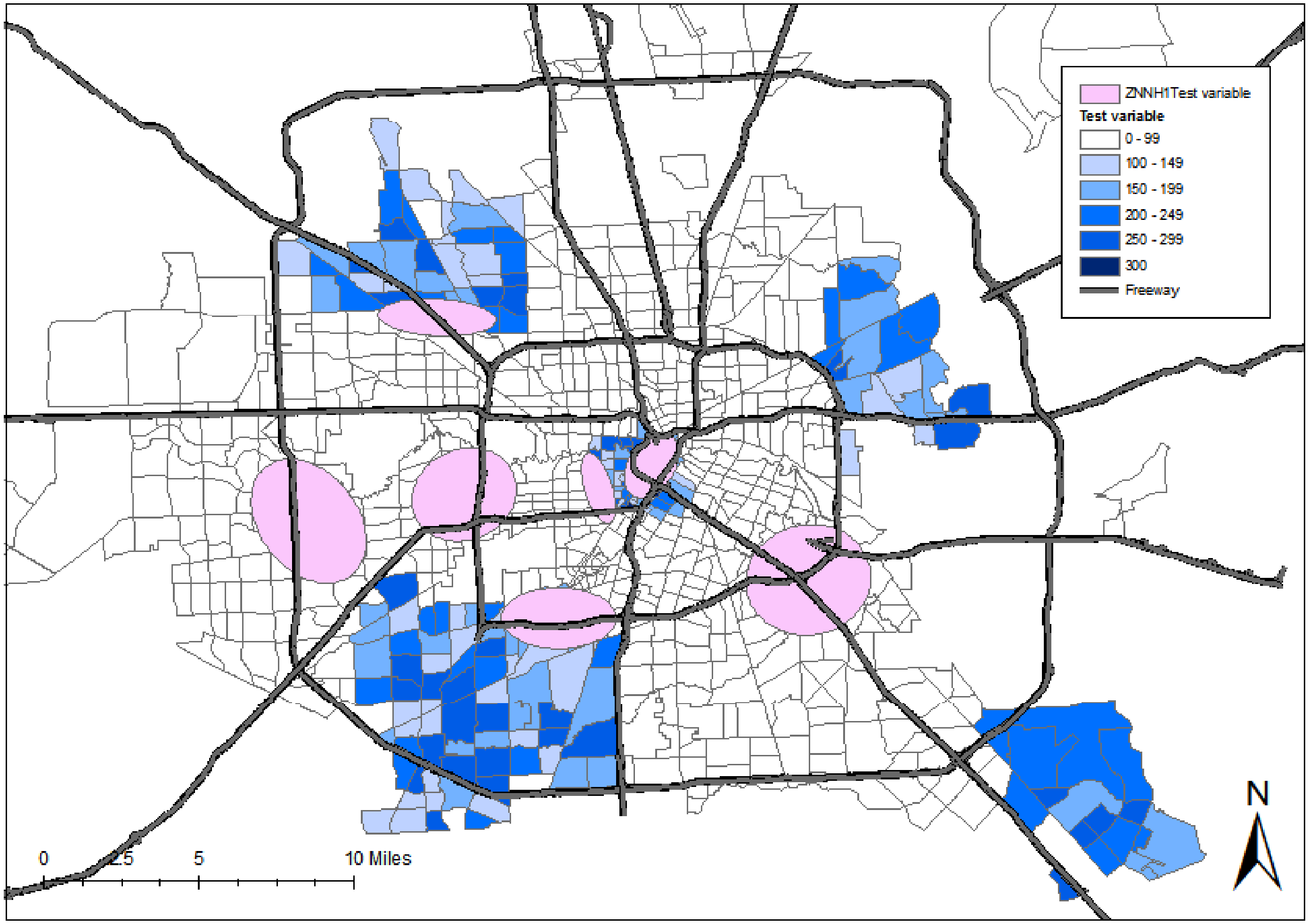


Figure 9.9:

Test of Znnh Routine

Identified Hot Spots with 5 Miles Search Radius and Minimum Number of Events=25

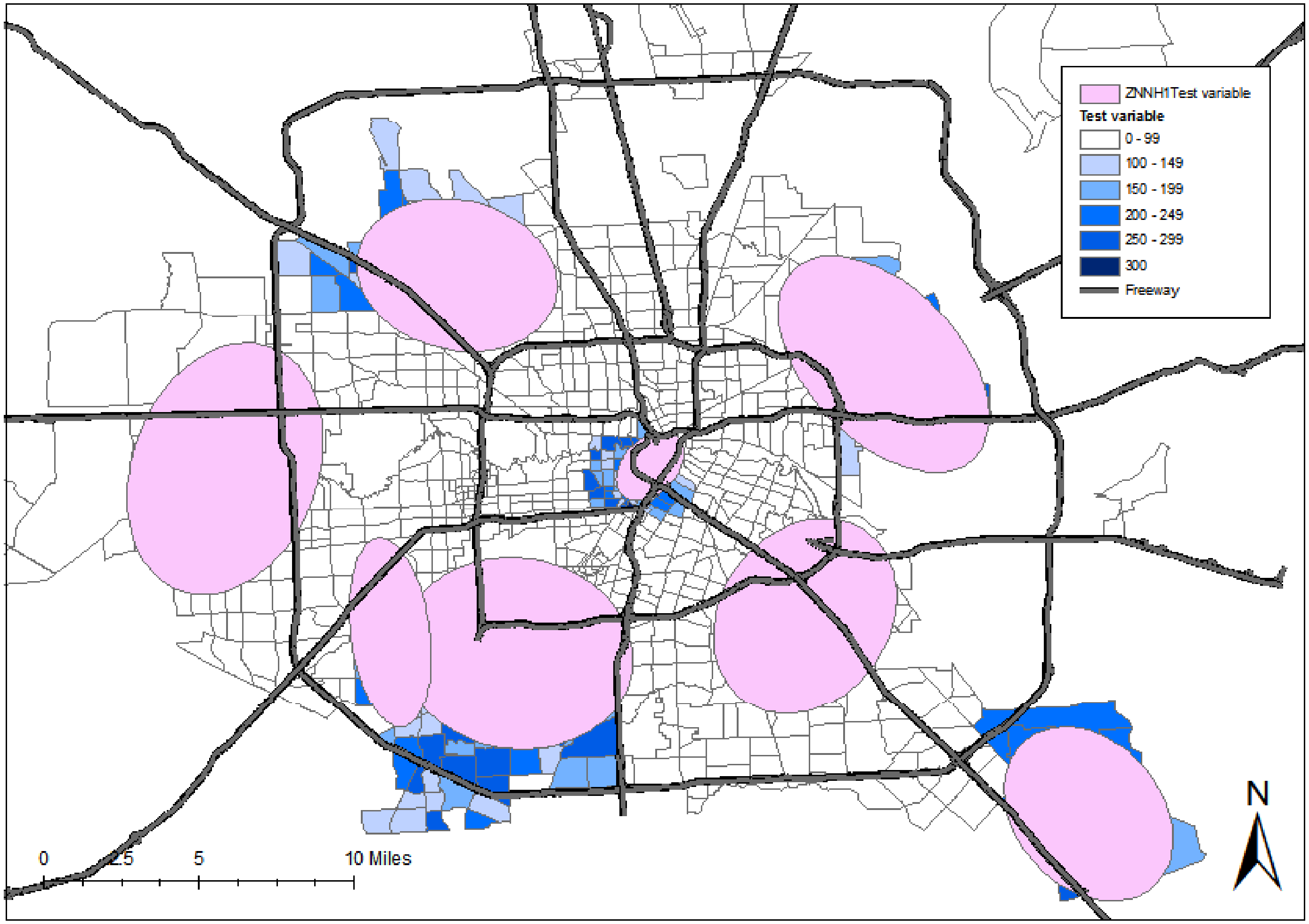


Figure 9.10:

Test of Znnh Routine

Identified Hot Spots with 8 Miles Search Radius and Minimum Number of Events=25

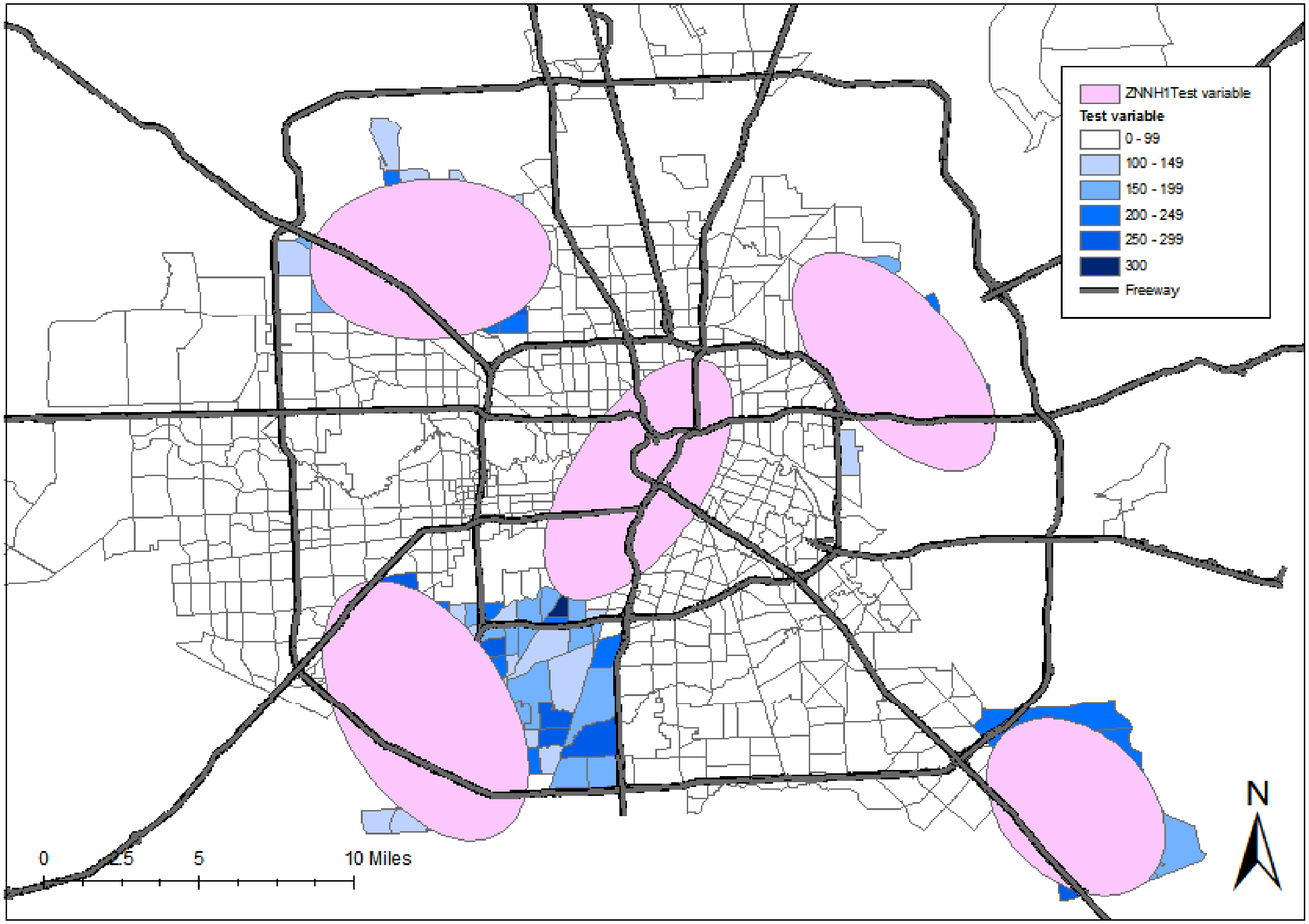


Figure 9.11:

Test of Znnh Routine

Identified Hot Spots with 8 Miles Search Radius and Minimum Number of Events=25

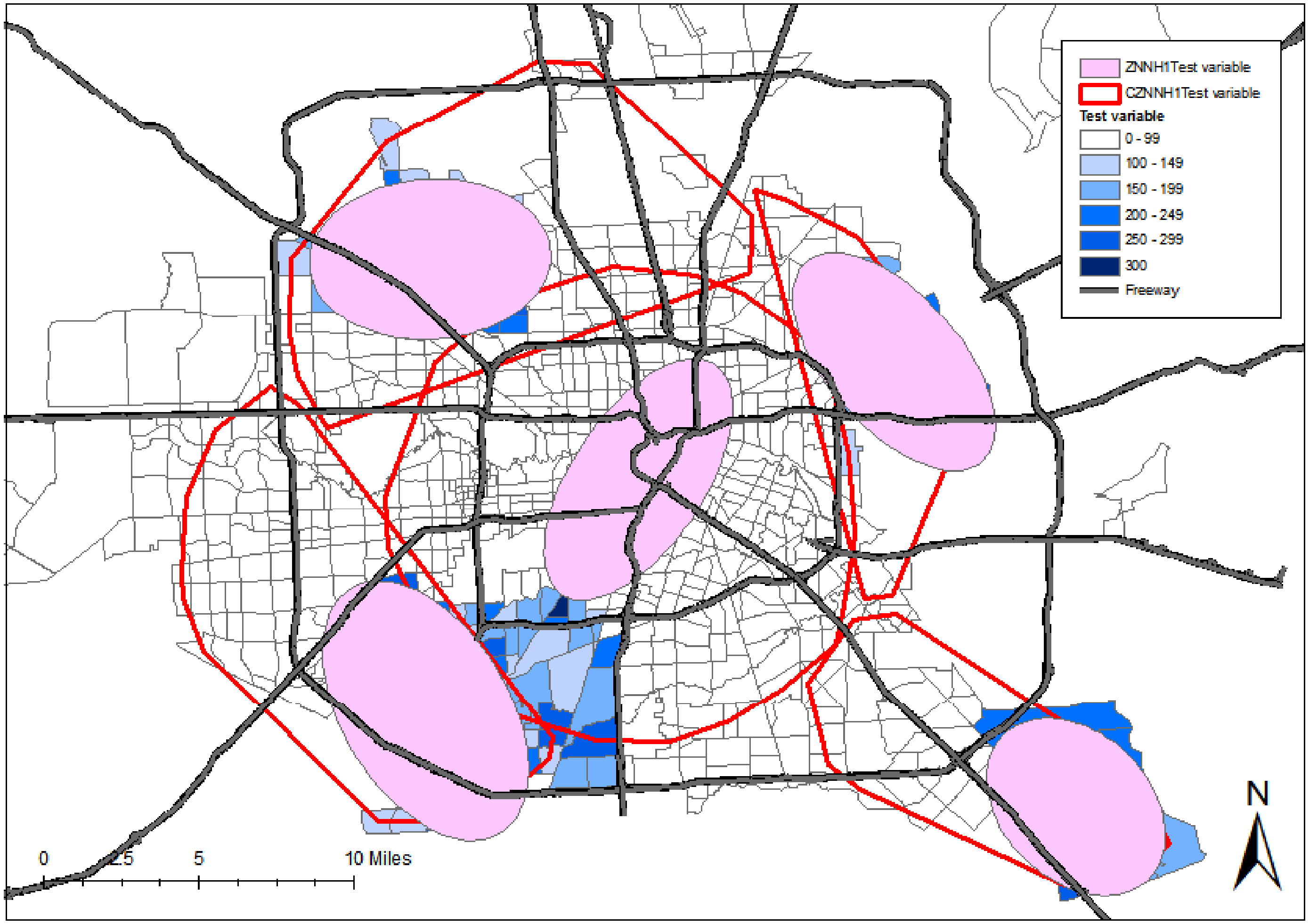
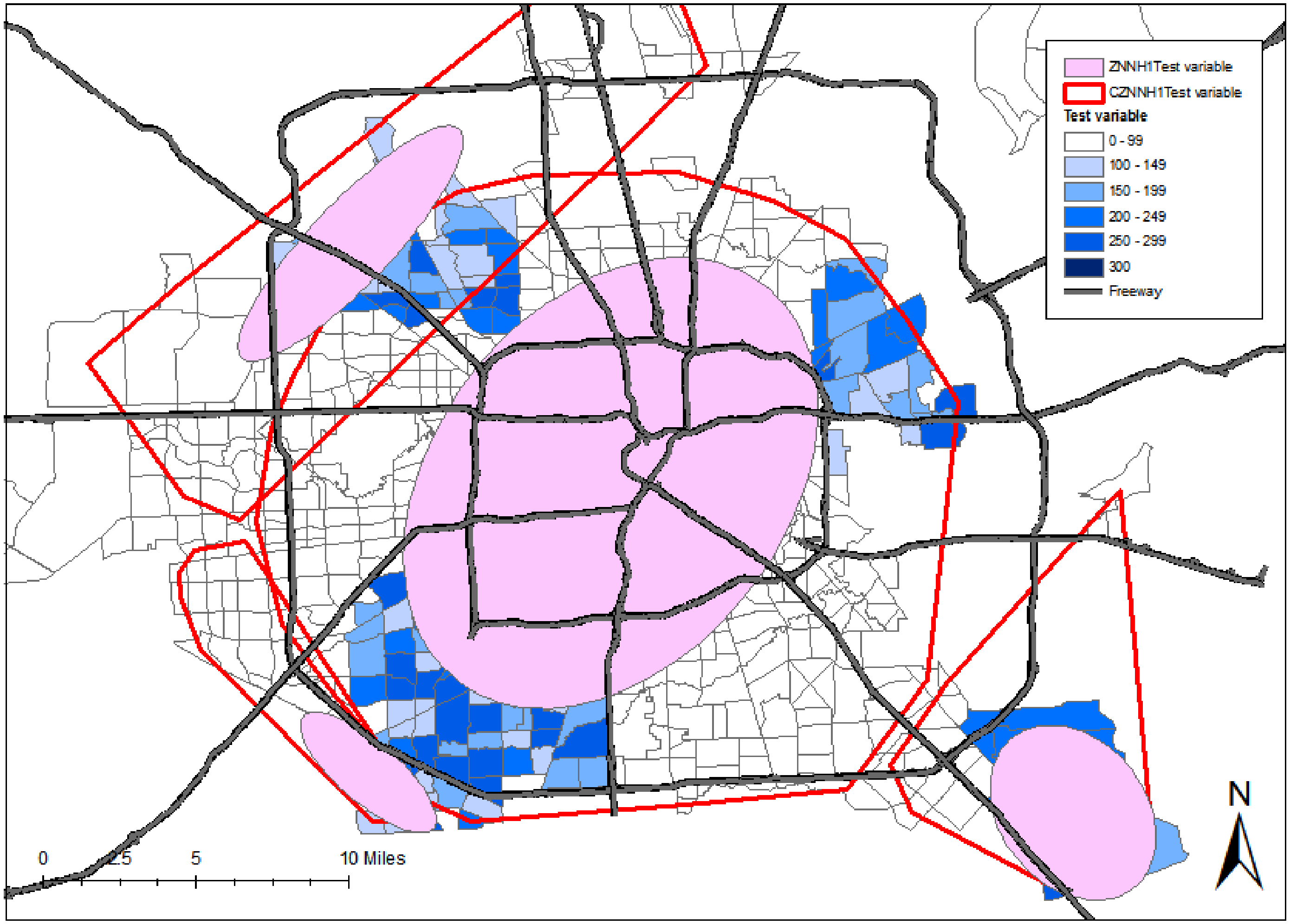


Figure 9.12:

Test of Znnh Routine

Identified Hot Spots with 12 Miles Search Radius and Minimum Number of Events=25



Figures 9-13 shows the results of the 0.5 mile search radius. Four clusters were identified, but they were very small and covered only the downtown Houston area. The reason is that with a half mile radius, only very small TAZ's can be captured within the radius and these are typically in the central downtown area. Further, they do not capture many burglaries, only 139 of the 24,935. However, they do a better job of capturing high density burglary TAZ's, defined as burglaries per square mile (Figure 9.14)

Figure 9.15 through 9.17 show the results of using 2, 5 and 8 mile search radii. The 2 mile search radius produced 10 small clusters; the 5 mile search radius produced 9 medium-sized clusters, and the 8 mile search radius identified 5 moderately large clusters. Clearly the cluster structure produced by the 2 mile search radius was also too small to fit the citywide pattern whereas either the 5 mile search radius or the 8 mile search radius seemed to best fit the overall data. Depending on whether the user wants smaller or larger clusters would determine which of these is selected.

Keep in mind that there is a danger is using large search radii since the likelihood of obtaining clusters by chance increases. To illustrate this, two Monte Carlo simulations of 1000 runs was made with both the 0.5 and the 8 mile search radius. Table 9.4 compares the actual clusters with the simulated clusters.

With the 0.5 mile search radius, no clusters were identified in the Monte Carlo simulation. This indicates that the clusters identified in Figure 9.13 are most likely real. On the other hand, with the 8 mile search radius and randomly distributed data, the expected number of clusters would be expected to vary between 5 and 8 clusters 95% of the time. This is calculated as the credible interval defined by the 2.5th and 97.5th percentiles. Thus, the five clusters obtained by the Znnh are not significantly greater than or smaller than what would be expected by chance. Similarly, the area of the ellipses, the number of attribute points captured and the number of zones are not significantly different than what would be expected by chance.

In short, the distribution that was obtained was not fundamentally different from a chance distribution. This is primarily the result of selecting a very large search radius. A user has to balance the choice between a small search radius which would capture clusters that are statistically much less likely to be due to chance but which cover only a small proportion of the study area with a larger search radius to capture the overall pattern but which increases the likelihood of identifying clusters by chance. In other words, there is a precision versus utility choice with a zonal clustering algorithm such as the Znnh.

Figure 9.13:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 0.5 Mile Search Radius and Minimum Number of Events=25

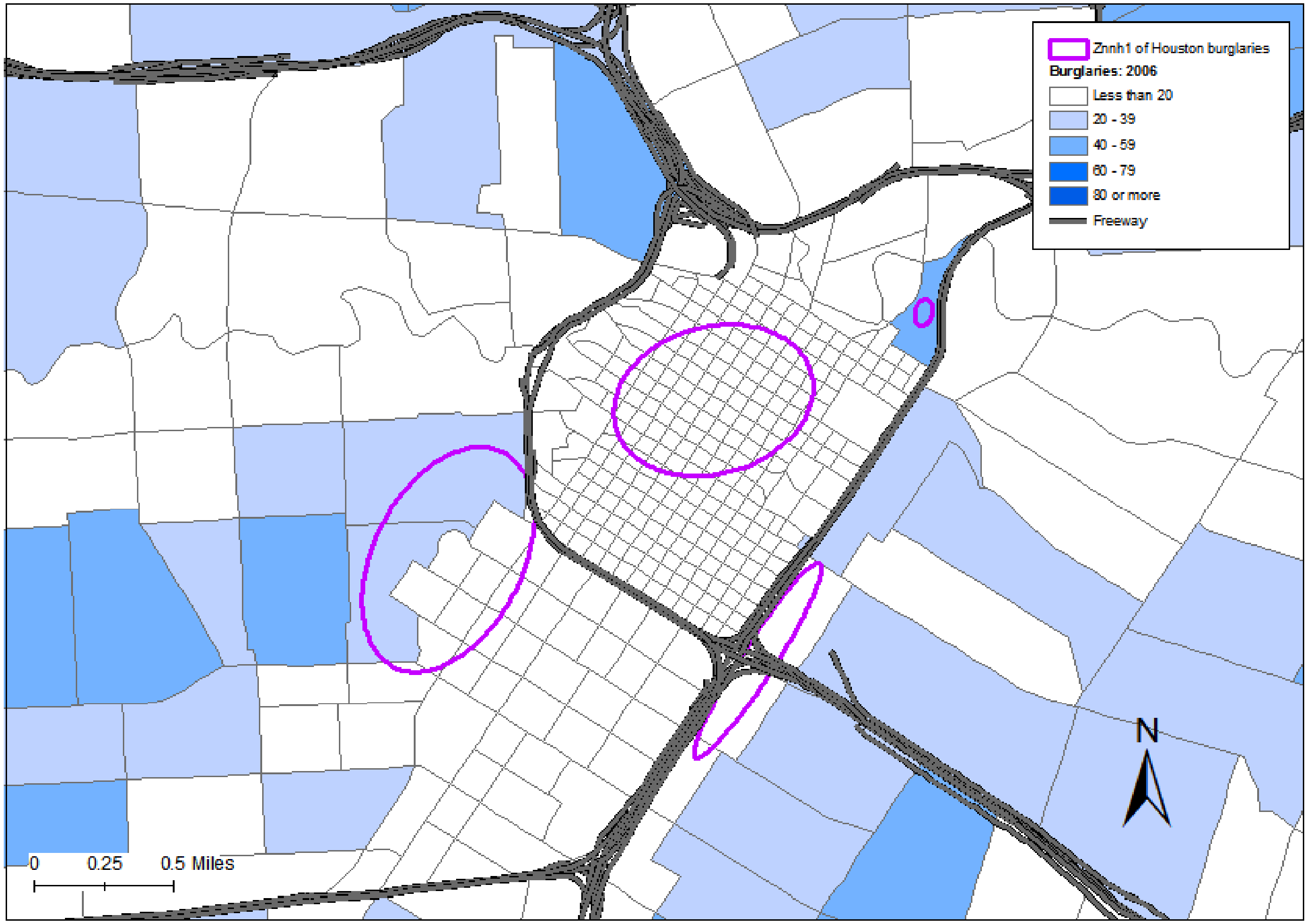


Figure 9.14:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 0.5 Mile Search Radius and Minimum Number of Events=25

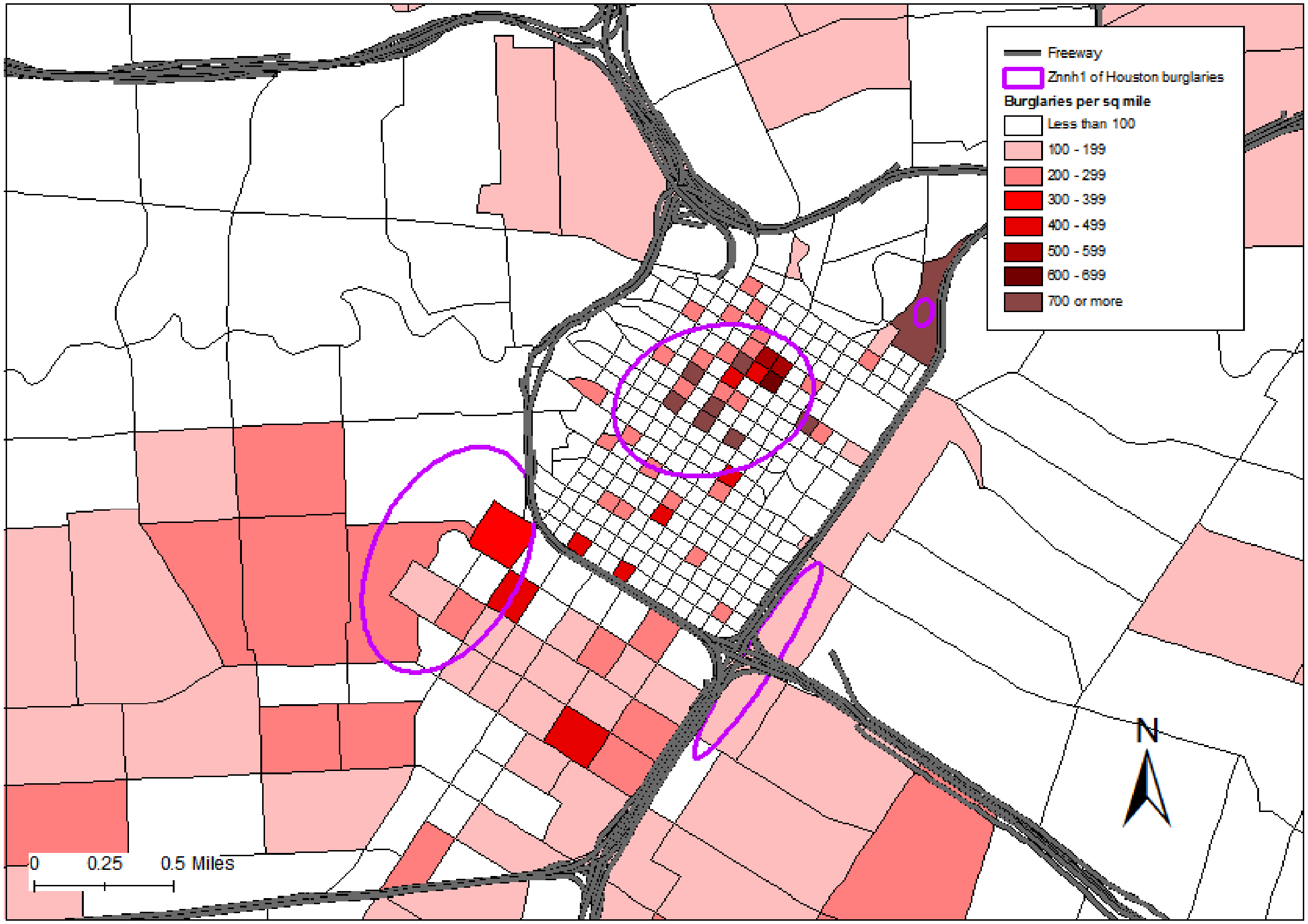


Figure 9.15:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 2 Mile Search Radius and Minimum Number of Events=25

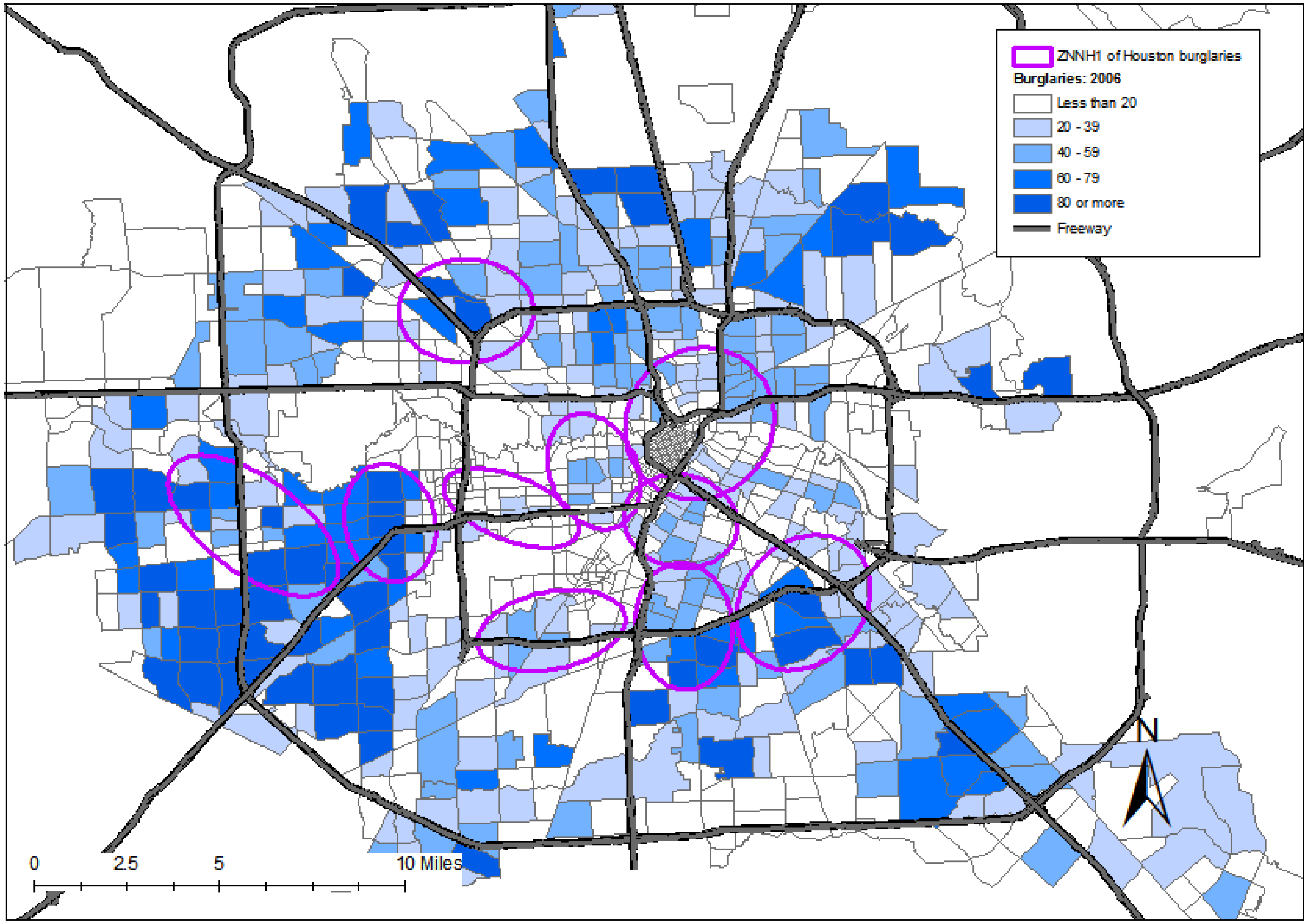


Figure 9.16:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 5 Mile Search Radius and Minimum Number of Events=25

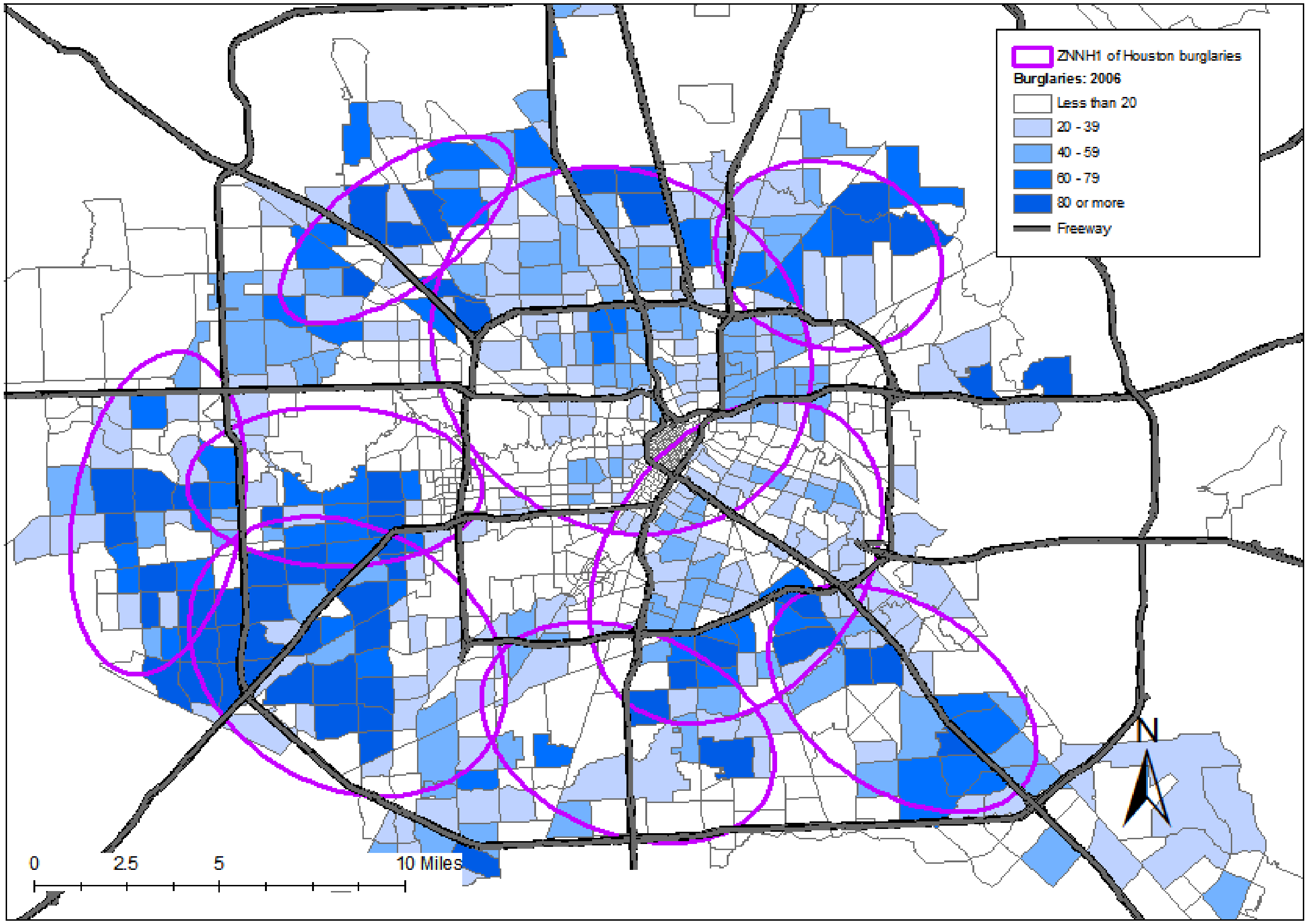


Figure 9.17:

Burglary Hot Spots in Houston: 2006

Identified Hot Spots with 8 Mile Search Radius and Minimum Number of Events=25

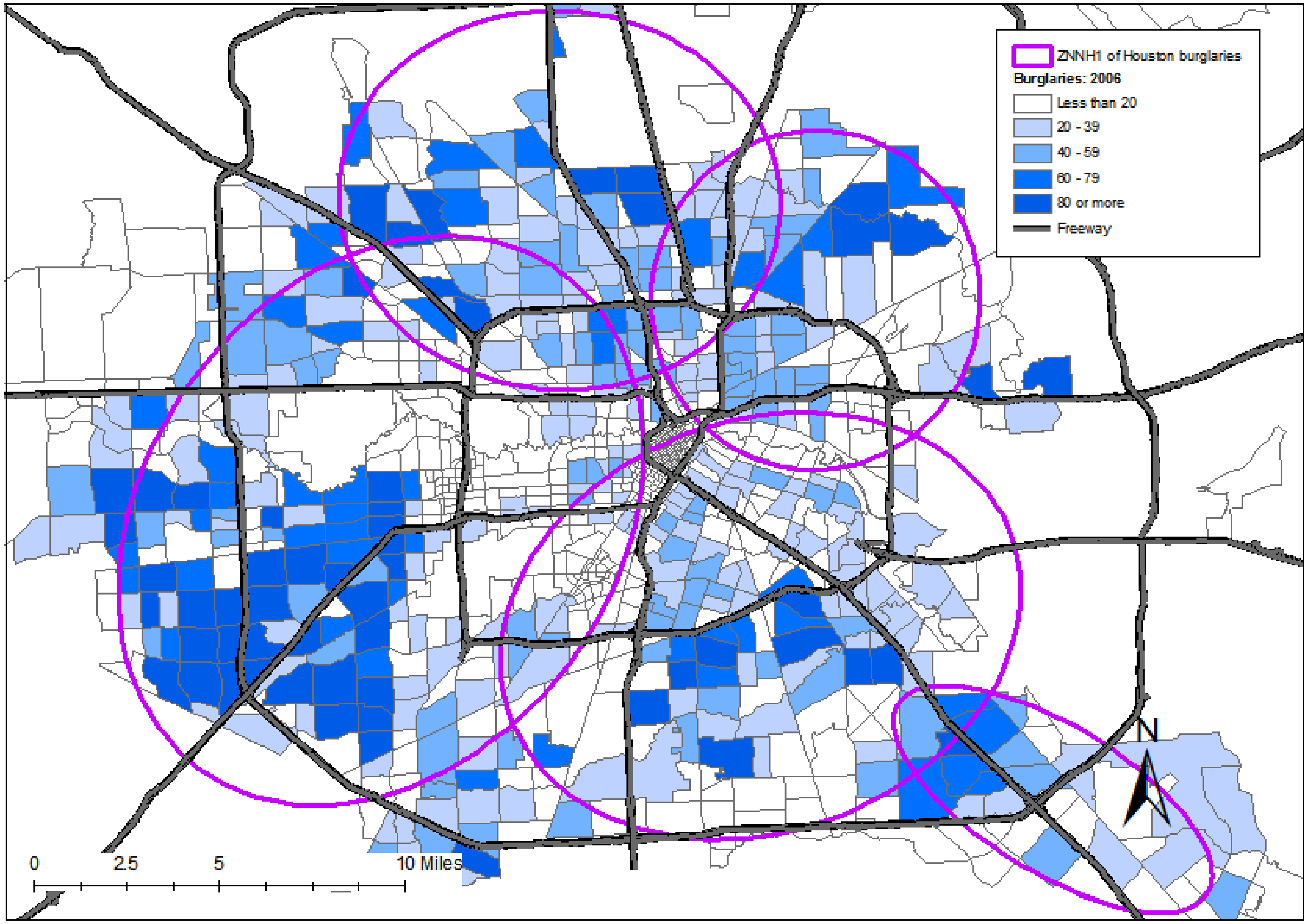


Table 9.4:

Zonal Nearest Neighbor Hierarchical Clustering of Houston Burglaries

(N= 24,935)

0.5 mile search radius, Minimum points per cluster=25

Cluster	Area of Ellipse *(sq mi)*	Number of Points	Number of Zones	Density
1	0.005	56	35	11,633.1
2	0.310	68	155	219.3
3	0.081	29	36	359.4
4	0.374	99	34	264.7

No clusters found in simulation

8 mile search radius, Minimum points per cluster=25

Cluster	Area of Ellipse *(sq mi)*	Number of Points	Number of Zones	Density
1	171.758	12,749	623	74.227
2	99.028	3,253	91	32.849
3	130.048	5,070	288	38.986
4	65.936	2,418	86	36.672
5	31.450	681	26	21.653

Percentile	Clusters	Area of Ellipse *(sq mi)*	Number of Zones	Density
0.5	4	10.34	25	0.707
1.0	5	11.04	25	0.730
2.5	5	12.29	25	0.777
5.0	5	13.74	25	0.862
10.0	5	15.74	26	0.938
90.0	7	239.21	467	2.092
95.0	8	241.59	471	2.183
97.5	8	243.05	477	2.353
99.0	8	244.67	481	2.631
99.5	8	245.06	483	2.798

Uses of Zonal Nearest Neighbor Hierarchical Clustering

This brings up one of the dilemmas in using a zonal clustering technique. On the one hand, since zones do not overlap, the dispersion is much more spread out than with individual events. As seen in Chapter 7, the regular nearest neighbor hierarchical clustering routine (Nnh) produced quite small clusters. With zonal data, however, all the events are assigned to a single point within the zone which either creates a cluster associated with a single or else a dispersion between adjacent zones that have a higher concentration. Since the identification of a single zone is not very useful, the Znnh routine requires a minimum of three adjacent zones to be included in a cluster.

Still, the Znnh can be useful for describing overall cluster patterns in a study area even with the increased uncertainty associated with large search radii. As Figures 9.16 and 9.17 illustrate, meaningful areas of higher concentration can be identified even though the identified clusters cannot be empirically distinguished from a chance distribution. This gives the user flexibility in defining groupings of zones which can then be used for various purposes (e.g., assigning patrols or defining contingency areas).

Limitations of Zonal Nearest Neighbor Hierarchical Clustering

On the other hand, the Znnh routine does have some limitations. The first was shown above, namely that to ensure that clusters are substantially different from that expected by chance, only small search radii can be chosen. However, given that most zones are associated with population density with the smallest zones being in the downtown center but increasing in size with distance from the center, the use of a small search radius becomes less useful.

Second, choosing a larger search radius can produce a cluster structure that appears to fit the data better but cannot be empirically distinguished from a chance distribution. Since there is not a single criterion that can be used to select among these, there is a certain amount of arbitrariness in the selection of a search radius or in the minimum number of events/attribute values specified. A user will have to experiment with different combinations to find the cluster structure that best fits the data. In this sense, the Znnh routine is more similar to the K-means clustering routine discussed in Chapter 8 than the Nnh routine in Chapter 7.

The best solution, of course, is to use the location of individual events and cluster them with either either Nnh, STAC or K-means routines discussed in Chapters 7 and 8. The Znnh routine should only be used if the data are organized by zones and cannot be disaggregated. In this case, the user must be aware of the limitations of the Znnh method and of the trade-off between precision (certainty) and utility.

A third limitation is that the cluster structure will almost certainly be different than had the individual events been clustered using the point-based Nearest Neighbor Hierarchical Clustering routine (Nnh). The requirement that zones do not overlap and that all events are assigned to the centroid of the zone ensures that the Znnh clusters will almost always be larger in size than the point-based Nnh clusters. In short, assigning events to zones and then clustering the zones will produce a larger and less focused cluster structure than the events themselves. The Znnh is only useful when it is not possible to disaggregate events to individual locations.

References

Anselin, L. (1995). Local indicators of spatial association - LISA. *Geographical Analysis*. 27, No. 2 (April), 93-115.

Chainey, S. & Ratcliffe, J. (2005). *GIS and Crime Mapping*, John Wiley & Sons, Inc.:Chichester, Sussex, England.

Getis, A. (1991). Spatial interaction and spatial auto-correlation: a cross-product approach. *Environment and Planning A*, 23, 1269-1277.

Getis, A. & Ord, J. K. (1996). Local spatial statistics: an overview. In Longley, Paul & Batty, Michael (eds), *Spatial Analysis: Modelling in a GIS Environment*. GeoInformation International: Cambridge, England, 261-277.

Mantel, N. (1967). The detection of disease clustering and a generalized regression approach. *Cancer Research*, 27, 209-220.

Waller, L. A. & Gotway, C. A. (2004). *Applied Spatial Statistics for Public Health Data*. John Wiley & Sons: Hoboken, NJ.

Wong, D. W. S. & Lee, J. (2005). *Statistical Analysis of Geographic Information with ArcView GIS and ArcGIS*. J. Wiley & Sons, Inc.: New York.

Endnotes

i. The variance of the Local Moran is defined in three steps:

A. First, define b_2.

$$b_2 = \frac{\sum_{i=1}^{N} \frac{(X_i - \bar{X})^4}{N}}{[\sum_{i=1}^{N} \frac{(X_i - \bar{X})^2}{N}]^2}$$

This is the fourth moment around the mean divided by the squared second moment around the mean.

B. Second, define $2w_{i(kh)}$:

$$2w_{i(kh)} = \sum_{k=1}^{N-1} \sum_{h=1}^{N-1} W_{ik} \, W_{ih}$$

where k ≠ i and h ≠ i. This term is twice the sum of the cross-products of all weights for *i* with themselves, using k and h to avoid the use of identical subscripts. Since each pair of observations, *i* and *j*, has its own specific weight, a cross-product of weights are two weights multiplied by each other (where i ≠ j) and the sum of these cross-products is twice the sum of all possible interactions irrespective of order (i.e., $W_{ij} = W_{ji}$). Because the weight of an observation with itself is zero (i.e., $W_{ii} = 0$), all terms can be included in the summation.

C. Third, define the variance, standard deviation, and an approximate (pseudo) standardized score of I_i:

$$Var(I_i) = \frac{(\sum_{i=1}^{N} \sum_{j=1}^{N} W_{ij}^2)(N - b_2)}{N-1} + \frac{2w_{i(kh)}(2b_2 - N)}{(N-1)(N-2)} + \frac{(\sum_{i=1}^{N} \sum_{j=1}^{N} W_{ij}^2)}{(N-1)^2}$$

$$S(I_i) = \sqrt{Var(I_i)}$$

$$Z(I_i) = \frac{I_i - E(I_i)}{S(I_i)}$$

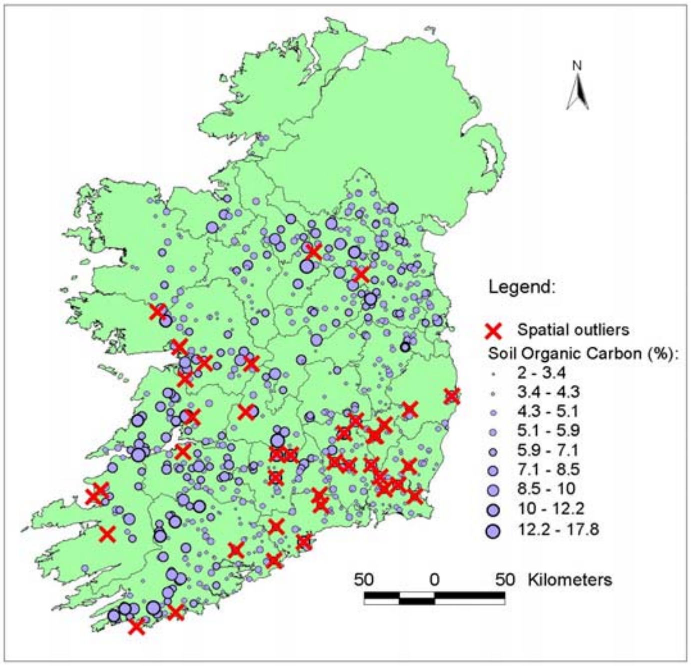
Attachments

Using Local Moran's "I" to Detect Spatial Outliers in Soil Organic Carbon Concentrations in Ireland

Chaosheng Zhang[1] David McGrath[2]
Lecturer in GIS Research Officer
[1] Department of Geography, National University of Ireland, Galway, Ireland
[2] Teagasc, Johnstown Castle Research Centre, Wexford, Ireland

One objective in the study of soil organic carbon concentrations is to produce a reliable spatial distribution map. A geostatistical variogram analysis was applied to study the spatial structure of soils in Ireland for the purpose of carrying out a spatial interpolation with the Kriging method. The variogram looks at similarities in organic carbon concentrations as a function of distance. In the analysis, a relatively poor variogram was observed, and one of the main reasons was the existence of spatial outliers. Spatial outliers make the variogram curve erratic and hard to interpret, and impair the quality of the spatial distribution map.

CrimeStat was used to identify the spatial outliers. The parameter of the standardized Anselin's Local Moran's "I (*z*)" was used. When $z < -1.96$, the sample was defined as a spatial outlier. Out of 678 soil samples, a total of 39 samples were detected as spatial outliers, and excluded in the spatial structure calculation. As a consequence, the variogram curve was significantly improved. This improvement made the final spatial distribution map more reliable and trustable.



Spatial outliers are clearly different from the majority of samples nearby. Compared with the samples nearby, high value spatial outliers are found in the southeastern part, and low value spatial outliers are located in the western and northern parts of the country.