Chapter 7
‘Hot Spot’ Analysis II

This chapter continues the discussion of hot spots. Three additional routines are discussed: ICJIA’s STAC routine (discussed by Richard and Carolyn Block), the K-means routine, and Anselin’s Local Moran. Figure 7.1 displays the ‘Hot Spot’ Analysis II page. The first of these routines, the Spatial and Temporal Analysis of Crime (STAC), was developed by the Illinois Criminal Justice Information Authority and integrated into CrimeStat in version 2. The second routine - K-means, is a partitioning technique. The third technique - Anselin’s Local Moran, is a zonal hot spot method. We’ll start first with STAC, and who better to explain it than the authors of the routine, Richard and Carolyn Block.

Spatial and Temporal Analysis of Crime (STAC)
by
Richard Block
Professor of Sociology
Criminal Justice
Loyola University
Chicago, IL

Carolyn Rebecca Block
Senior Research Analyst
Illinois Criminal Justice Information Authority
Chicago, IL

The amount of information available in an automated pin map can be enormous. When geographic information systems were first introduced into policing, there were few ways to summarize the huge reservoir of mapped information that was suddenly available. In 1989, police departments in Illinois asked the Illinois Criminal Justice Information Authority to develop a technique to identify Hot Spot Areas (the densest clusters of points on a map). The result was STAC, the first crime hot spot program. Through the years, “bells and whistles” have been added to STAC, but the algorithm has remained essentially the same. STAC is a quick, visual, easy-to-use program for identifying Hot Spot Areas.

The STAC Hot Spot Area routine in CrimeStat searches for and identifies the densest clusters of incidents based on the scatter of points on the map. The STAC Hot Spot Area routine creates areal units from point data. It identifies the major concentrations of points for a given distribution. It then represents each dense area by the

STAC is a scan-type clustering algorithm in which a circle is repeatedly laid over a grid and the number of points within the circle are counted (Openshaw, Charlton, Wymer and Craft, 1987; Openshaw, Craft, Charlton, and Birch, 1988; Turnbull, Iwano, Burnett, Howe, and Clark, 1990; Kuldorff, 1995). It, thus, shares with those other scan routines the property of multiple tests, but it differs in that the overlapping clusters are combined into larger cluster until there are no longer any overlapping circles. Thus, STAC clusters can be of differing sizes. The routine, therefore, combines some elements of partitioning clustering (the search circles) with hierarchical clustering (the aggregating of smaller clusters into larger clusters).
Figure 7.1: 'Hot Spot' Analysis II Screen
The STAC Hot Spot Area routine in CrimeStat searches for and identifies the densest clusters of incidents based on the scatter of points on the map. The STAC Hot Spot Area routine creates areal units from point data. It identifies the major concentrations of points for a given distribution. It then represents each dense area by either a standard deviational ellipse or a convex hull, or both (see chapter 4). The boundaries of the ellipses or convex hulls can easily be displayed as mapped layers by standard GIS software.

STAC is not constrained by artificial or political boundaries, such as police beats or census tracts. This is important, because clusters of events and places (such as drug markets, gang territories, high violence taverns, or graffiti) do not necessarily stop at the border of a police beat. Also, shading over an entire area may make it seem that the whole neighborhood is high-crime (or low-crime), even though the area may contain only one or two dense pockets of crime. Therefore, area-shaded maps could be misleading. In contrast, STAC Hot Spot Areas are based on the actual clusters of events or places on the map.

STAC is designed to help the crime analyst summarize a vast amount of geographic information so that practical policy-related issues can be addressed, such as resource allocation, crime analysis, beat definition, tactical and investigation decisions, or development of intervention strategies. An immediate concern of a law enforcement user of automated pin maps is the identification of areas that contain especially dense clusters of events. These pockets of crime demand police attention and could indicate different things for different crimes. For instance, a grouping of Criminal Damage to Property offenses could indicate gang activity. If motor vehicle thefts consistently cluster in one section of town, it could point to the need to change patrol patterns and procedures.

To take an example, Figure 7.2 shows the location of the seven densest Hot Spot Areas of street robbery in 1999 in Chicago. Four of the seven span the boundaries of police districts and two cover only a small part of a larger district. In a shaded area map, these dense clusters of robbery might be not easily identifiable. An area that is really dense might appear to be low-crime because it is divided by an arbitrary boundary. Using a shaded areal map aggregating the data within each district would give a general idea of the distribution of crime over the entire map, but it would not tell exactly where the clusters of crime are located.

For example, figure 7.3 zooms in on Hot Spot Area 4 (the northernmost Hot Spot Area in Figure 7.2). Hot Spot Area 4 covers parts of two districts (shown by a pink boundary line in figure 7.2) There are also four beats (shown by blue boundary lines). The shaded map indicates many incidents in beat 2311, but few in beats 2312, and 2313. The incident distribution indicates that while few incidents occurred overall in 2312 and 2313, most of the incidents that did occur were near to beat 2311. Incidents in beat 2311 mainly occurred on its eastern boundary. Portions of the beat were relatively free from street robbery. The Hot Spot Area identifies this clustering that spans beats and districts. Hot Spot Areas that overlap beat and district boundaries might indicate to patrol officers in these neighboring areas that they should coordinate their efforts in combating crime.
Figure 7.2: STAC Hot Spots for 1999 Street Robberies

1999 Chicago Street Robberies:
STAC 1 Std Deviation Hot Spot Ellipses
Search Radius 750 Meters

Source: Chicago Police Department
Figure 7.3: STAC 1999 Street Robbery Hot Spot Area 4
How STAC Identifies Hot Spot Areas

The following procedures identifies hot spots in STAC. The program implements a search algorithm, looking for Hot Spot Areas.

1. STAC lays out a 20 x 20 grid structure (triangular or rectangular, defined by the user) on the plane defined by the area boundary (defined by the user).

2. STAC places a circle on every node of the grid, with a radius equal to 1.414 (the square root of 2) times the specified search radius. Thus, the circles overlap.

3. STAC counts the number of points falling within each circle, and ranks the circles in descending order.

4. For a maximum of 25 circles, STAC records all circles with at least two data points along with the number of points within each circle. The X and Y coordinates of any node with at least two incidents within the search radius are recorded, along with the number of data points found for each node.

5. These circles are then ranked according to the number of points and the top 25 search areas are selected.

6. If a point belongs to two different circles, the points within the circles are combined. This process is repeated until there are no overlapping circles. This routine avoids the problem of data points belonging to more than one cluster, and the additional problem of different cluster arrangements being possible with the same points. The result is called Hot Clusters.

7. Using the data points in each Hot Cluster, for each cluster the program can calculate the best-fitting standard deviational ellipse or convex hull (see chapter 4). These are called Hot Spot Areas. Because the standard deviational ellipse is a statistical summary of the Hot Cluster points, it may not contain every Hot Cluster point. It also may contain points that are not in the Hot Cluster. On the other hand, the convex hulls will create a polygon around all points in the cluster.

The user can specify different search radii and re-run the routine. Given the same area boundary, different search radii will often produce slightly different numbers of Hot Clusters. A search radius that is either too large or too small may fail to produce any. Experience and experimentation are needed to determine the most useful search radii.
Steps in Using STAC

STAC is available on the Hot Spot Analysis II tab under Spatial Description (see figure 7.1). A brief summary of the steps is as follows:

1. STAC requires a primary file and a reference file (see chapter 3). Optionally, STAC requires the reference file area (on the measurement parameters tab) if simulation runs are requested. Note: while STAC runs quite quickly, it runs more quickly with a Euclidean coordinate system such as UTM or State Plane. For example, an analysis of 13,000 street robberies in Chicago ran in less than two seconds on an 800 mhz PC with projected coordinates (Euclidean), while it took longer with spherical coordinates (latitude/longitude).

2. Define the reference file (see chapter 3). While CrimeStat does not include a data base manager or query system, a user can carry out analysis of different areas of a jurisdiction by using the boundaries of several reference areas. For example, define all of Chicago as a reference area and define each of the twenty-five police districts as additional reference areas. Hot Spot Areas can be identified for the city as a whole and for each district. In other words, the same incident file may be used for analysis of different map areas by using multiple reference files.

3. Define the search radius. Generally, a two-stage analysis is best. Start with a larger search radius and then analyze Hot Spot Areas with a smaller search radius. A search radius of more than one mile may not yield useful results in an area the size of Chicago (320 square miles).

4. Set the output units to miles or kilometers.

5. Specify the file output name for the ellipses or convex hulls.

6. Click on the STAC parameters button.

The object of STAC is to identify hot spots and display them with ellipses or convex hulls. Its key function is visual. Save the ellipses or hulls in the form most appropriate for the system (e.g., ArcView, Atlas, MapInfo). Because the ellipses or convex hulls are generated as polygons, they can be used for selections, queries, or thematic maps in the GIS. In addition to the ellipses and convex hulls, a table is output with all the information on density and location for each ellipse. It can be saved to a 'dbf' file, which can then be read by any spreadsheet program. The ellipses and convex hulls are numbered in the same order as the printed output.
Figure 7.4: STAC Parameters Setup

- Search radius: 250 Meters
- Minimum points per cluster: 5
- Simulation runs: 1000
- Scan type: Rectangular
- Boundary: From data set
- Number of standard deviations for the ellipses: 1X, 1.5X, 2X

Save ellipses to...
Save convex hull to...
Save results to...
Save ellipses to...
Save convex hull to...
Save results to...
**STAC Parameters**

The two most important parameters for running STAC are the boundary of the study area (reference area) and the search radius. A detailed discussion of the parameters follows. Figure 7.4 shows the STAC parameters screen.

**Search Radius**

1. The search radius is the key setting in STAC. In general, the larger the search radius, the more incidents that will be included in each Hot Cluster and the larger the ellipse that will be displayed. Smaller search radii generally result in more ellipses of a smaller size. A good strategy is to initially use a larger radius and then re-analyze areas that are ‘hot’ with a smaller radius. In Chicago, we have found that a 750 meter radius is appropriate for the city as a whole and a 200 meter search radius for one of the 25 districts. It will be necessary to experiment to determine an appropriate search radius.

**Units**

2. Specify the units for the search radius. The default is miles and the default search radius is 0.5 miles. Be careful about using larger search radii. In Chicago, a search radius larger than one mile generates ellipses that are too large to be of any tactical or planning use. Other good choices are 750 meters or 0.25 miles.

**Minimum Points Per Cluster**

3. Specify the minimum number of points to be included in a Hot Cluster. The limit for the minimum points in a Hot Cluster is two. We usually use a minimum of 10.

**Boundary**

4. Select the reference file to be used for the analysis. The user can choose the boundary from the data set (i.e., the minimum and maximum X/Y values) or from the reference boundary. In our opinion, the choice of the reference boundary is best. If the data set is used to define the reference boundary, the smallest rectangle that encompasses all incident will be used.

**Scan Type**

5. Select the scan type for the grid. Choose Rectangular if the analysis area has a mostly grided street pattern. Chose Triangular if the analysis area generally has an irregular street pattern.
Graphical output files

6. Select whether the graphical output will be displayed as standard deviational ellipse or as convex hulls, or both (see chapter 4). For ellipses, select the number of standard deviations for the ellipses. One (1X), 1.5X, and 2X standard deviations can be selected. One standard deviational ellipses should be sufficient for most analysis. While one standard deviational ellipses rarely overlap, 1.5X and 2X two standard deviational ellipses often do. A larger ellipse will include more of the Hot Cluster points; a small ellipse will produce a more focused Hot Cluster identification. The user will have to work out a balance between defining a cluster precisely compared to making it so large as to be unclear where one starts and another ends.

Simulation Runs

7. Specify whether any simulation runs are to be made. To test the significance of STAC clusters, it is necessary to run a Monte Carlo simulation (Dwass, 1957; Barnard, 1963). CrimeStat includes a Monte Carlo simulation routine that produces approximate confidence intervals for the particular STAC model that has been run. The difference between the density of incidents in STAC ellipses in a spatially random data set and the STAC ellipses in the actual data set is a test of the strength of the clustering detected by STAC. Essentially, the Monte Carlo simulation assigns N cases randomly to a rectangle with the same area as the defined study area as specified on the Measurement Parameters tab and evaluates the number of clusters according to the defined parameters (i.e., search radius). It repeats this test K times, where K is defined by the user (e.g., 100, 1,000, 10,000). By running the simulation many times, the user can assess approximate confidence intervals for the particular number of clusters and density of clusters. The default is zero simulation runs because the simulation run option usually increases the calculation time considerably. If a simulation run is selected, the user should identify the area of the study region on the Measurement Parameters tab. It is better to use the jurisdictional area rather than the reference area if the jurisdiction is irregularly shaped.

Output

Ellipses or convex hulls

The ellipses are output with a prefix of “St” before the output file name while the convex hulls are output with a prefix of “Cst” before the output file name. These objects can easily be incorporated into a GIS system. ArcView shape files can be opened as themes. STAC graphic files also can be added as a MapInfo layer using the Universal Translator Tool. MapInfo MiF/Mid files must be imported using the command table—>import. Both MapInfo and ArcView files are polygons and can be used for queries, thematics, and selections.
Table 7.1 shows the printed output. Note that the printed output does not include the file name. Be sure to record the file name and the reference file (if any that is used).

1. The first section of the output documents parameter settings and file size. Sample size indicates the number of points in the file specified in the setup.

2. Measurement Type indicates the type of distance measurement, direct or Indirect (Manhattan).

3. Scan Type indicates a rectangular or triangular grid specified in the setup.

4. Input Unit indicates the units of the coordinates specified in the setup, degrees (if latitude/longitude) or meters or feet (if projected).

5. Output Units indicate the unit of density and length specified in the setup for the output and ellipses. Output Units are generally, miles or kilometers.

6. Search Radius is the units specified in the setup. In Figure 7.2 above, this is meters.

7. Boundary identifies the coordinates of the lower left and upper right corner of the study area.

8. Points inside the boundary count the number of points within the reference file. This may be fewer than the number of points in the total file when a smaller area is being used for analysis (see above).

9. Simulation Runs indicates the number of runs, if any specified in the setup.

10. Finally, STAC printed output provides summary statistics for each Hot Spot Area.

   A. Cluster— an identification number for each ellipse. This corresponds to their order in a table view in ArcView, or the browser in MapInfo.

   B. Mean X and Mean Y - Coordinates of the mean center of the ellipse.

   C. Rotation- the degrees the ellipse is rotated (0 is horizontal; 90 is vertical).

   D. X-axis and Y-axis - the length (in the selected output units) of the x and y axis. In the example, the length of the x axis of ellipse 1 is 1.04768 miles.
Table 7.1
Printed Output for STAC

Spatial and Temporal Analysis of Crime:
---------------------------------------

Sample size ...........: 1181
Measurement type ......: Direct
Scan type....... ......: Rectangular
Input units .... ......: Degrees
Output units ... ......: Miles, Squared Miles, Points per Squared Miles
Standard Deviations ...: 1
Search radius..........: 804.672000
Boundary...............: -76.83302,39.23274 to -76.38390,39.59103
Points inside boundary.: 1179
Simulation runs .......: 1000

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Mean X</th>
<th>Mean Y</th>
<th>Rotation</th>
<th>X-Axis</th>
<th>Y-Axis</th>
<th>Area</th>
<th>Points</th>
<th>Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-76.44915</td>
<td>39.31484</td>
<td>89.41867</td>
<td>1.04768</td>
<td>0.25053</td>
<td>0.82460</td>
<td>106</td>
<td>128.546688</td>
</tr>
<tr>
<td>2</td>
<td>-76.73681</td>
<td>39.28658</td>
<td>69.91502</td>
<td>0.22142</td>
<td>0.88202</td>
<td>0.61354</td>
<td>63</td>
<td>102.682109</td>
</tr>
<tr>
<td>3</td>
<td>-76.57098</td>
<td>39.38499</td>
<td>37.10812</td>
<td>0.34793</td>
<td>0.82213</td>
<td>0.89863</td>
<td>61</td>
<td>67.880882</td>
</tr>
<tr>
<td>4</td>
<td>-76.77129</td>
<td>39.35987</td>
<td>11.26360</td>
<td>0.94336</td>
<td>0.26216</td>
<td>0.77695</td>
<td>61</td>
<td>78.511958</td>
</tr>
<tr>
<td>5</td>
<td>-76.51830</td>
<td>39.26019</td>
<td>8.37773</td>
<td>0.43717</td>
<td>0.25497</td>
<td>0.35017</td>
<td>43</td>
<td>22.796997</td>
</tr>
<tr>
<td>6</td>
<td>-76.60231</td>
<td>39.40086</td>
<td>14.84392</td>
<td>0.17969</td>
<td>0.29466</td>
<td>0.16634</td>
<td>36</td>
<td>16.423811</td>
</tr>
<tr>
<td>7</td>
<td>-76.73087</td>
<td>39.34246</td>
<td>41.07812</td>
<td>0.31007</td>
<td>0.25885</td>
<td>0.25215</td>
<td>35</td>
<td>38.806566</td>
</tr>
<tr>
<td>8</td>
<td>-76.75451</td>
<td>39.31110</td>
<td>74.78196</td>
<td>0.19154</td>
<td>0.19154</td>
<td>0.25215</td>
<td>24</td>
<td>26.326405</td>
</tr>
</tbody>
</table>

Distribution of the number of clusters found in simulation (percentile):

<table>
<thead>
<tr>
<th>Percentile</th>
<th>Clusters</th>
<th>Area</th>
<th>Points</th>
<th>Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>min</td>
<td>12</td>
<td>0.01113</td>
<td>5</td>
<td>4.673554</td>
</tr>
<tr>
<td>0.5</td>
<td>13</td>
<td>0.02389</td>
<td>5</td>
<td>4.924993</td>
</tr>
<tr>
<td>1.0</td>
<td>13</td>
<td>0.03587</td>
<td>5</td>
<td>4.977644</td>
</tr>
<tr>
<td>2.5</td>
<td>14</td>
<td>0.05081</td>
<td>5</td>
<td>5.236646</td>
</tr>
<tr>
<td>5.0</td>
<td>14</td>
<td>0.06177</td>
<td>5</td>
<td>5.505124</td>
</tr>
<tr>
<td>95.0</td>
<td>19</td>
<td>1.24974</td>
<td>14</td>
<td>82.281060</td>
</tr>
<tr>
<td>97.5</td>
<td>19</td>
<td>1.39923</td>
<td>16</td>
<td>101.053102</td>
</tr>
<tr>
<td>99.0</td>
<td>20</td>
<td>1.58861</td>
<td>17</td>
<td>140.078387</td>
</tr>
<tr>
<td>99.5</td>
<td>20</td>
<td>1.67065</td>
<td>19</td>
<td>209.279368</td>
</tr>
<tr>
<td>max</td>
<td>20</td>
<td>2.08665</td>
<td>23</td>
<td>449.401912</td>
</tr>
</tbody>
</table>

E. Area - the area of the ellipse in square units. Ellipses are ordered according to their size. In the example, Ellipse 1 is 0.8246 square miles.

F. Points - the number of points in the Hot Cluster. In the example, there are 61 points in cluster 3.
G. Cluster Density - the number of points per square unit. The largest cluster is not necessarily the densest. In this example, cluster eight is the smallest, but its density is higher than two other clusters.

The best way to print or save CrimeStat printed output is to place the cursor inside the output window and Select all, then copy and paste the selection into a word processing document in landscape mode.

Make sure to adequately annotate the file, especially the type of incidents, the reference boundary, and the name of the output file. This can be very important for future reference.

For Old STAC Users

In general, STAC has retained all the functionality and speed of previous versions. The ellipses will look somewhat different than previous versions, because a more widely accepted method for calculating standard deviational ellipses has been used. STAC for DOS used a 1x standard deviation ellipse. Analysts who want results similar to STAC for DOS should set standard deviations to 1.

The CrimeStat version of STAC has the following improvements over STAC for DOS:

1. STAC no longer requires the use of a special ASCII data file. The data file can be any of those available in CrimeStat.

2. Any projection can be used, including latitude/longitude. Files are not converted into a Euclidean projection.

3. We have not found a limit on the number of points that can be analyzed with the CrimeStat version of STAC. Therefore, a small radius can now be used over large areas.

4. STAC can generate Shape files for ArcView or Mif/Mid files for MapInfo. Both are polygons-not points.

5. It is easier for the user to specify the number of standard deviations for an ellipse (1X, 1.5X, or 2X).

6. Convex hull output has been added.

7. The user can run STAC on a spatially random data set to get an estimate of the degree of clustering detected by STAC in the incident data.
The study area boundary (reference file) can be generated from the data set (we would not suggest doing this since it will be difficult to compare distributions).

**Example 1: A STAC Analysis of 1999 Chicago Street Robberies**

STAC Hot Spot Areas were calculated for all street (or sidewalk or alley) robberies occurring in Chicago in 1999 (n=13,009). There were 13,007 within the search boundary. The search radius was set for 750 meters (approximately ½ mile), and the ellipses were set to one standard deviation. Ten was the minimum number of incidents per cluster.

In figure 7.2 (shown earlier), STAC detected seven ellipses. The areas of the seven ellipses ranged from 5 square kilometers to 0.7 square kilometers, and the number of incidents in an ellipse ranged from 760 to 153. The smallest ellipse (number 7 in figure 7.2) was the densest, 222 robberies per square kilometer. Of the 13,007 incidents, 2,375 were in a cluster. Therefore, 18 percent of all of Chicago’s street robberies in 1999 occurred in 6% of its 233 square mile area.

To map the results, the ellipse boundaries were imported into MapInfo as a mif/mid file and overlaid on a map of police districts. The large blue rectangle in figure 7.2 designates the search boundary (reference file). O’Hare Airport was excluded because exact geo-coding is not possible for the few street robberies that occurred there. At a city-wide scale, the map is interesting, but is mainly useful for confirming what is already known. Ellipse 1, on the west side, has had a high level of violence for many years. Ellipses 2 and 6 are centered on areas where high rise public housing projects are gradually being abandoned. Overall, these ellipses are not very useful for tactical purposes. However, they point out that four Hot Spot Areas cross District boundaries, and that the large number of street robberies in these areas might be lost in separate district reports.

*A Neighborhood STAC Analysis*

The presence of Ellipse 4 (the northernmost ellipse in figure 7.2) might be unexpected to many Chicagoans. The mid-Northside, near the Lake Michigan, is generally considered to be a relatively affluent and safe neighborhood. However, the neighborhood around Ellipse 4 has had a high level of crime for many years. It was an entertainment center in the Roaring Twenties, and several institutions of that era remain. Today it is an area with multiple, often conflicting, uses. A more detailed analysis of the neighborhood with the help of STAC may point to specific areas that need increased patrol or prevention activities.

The second step of STAC analysis was to define a focused search boundary area around Ellipse 4. This was done easily by creating a new map layer in MapInfo and drawing a rectangle around the desired study area. Clicking on the study area gave the required CrimeStat reference boundary maximum and minimum coordinates. Using this more focused boundary, STAC was run a second time with a 200 meter search radius and
Figure 7.5: STAC Hot Spots for Northeast Side Street Robberies
the same file of 13,009 cases. The search boundary (reference file) now contained 442 incidents. STAC detected three ellipses that contained 231 incidents. The STAC ellipses were then imported into MapInfo and mapped (Figure 7.5).

As the area covered by a map grows smaller, detailed information about crime patterns and the community can be added. In this map, the STAC ellipses were overlain with the address locations of incidents (sized according to the number occurring at each location) and streets. Much of the area is relatively crime-free. The most frequent locations for street robbery do not coincide with main streets. Street robbery incidents tend to cluster near rapid transit stations and the blocks immediately surrounding them. For example, Argyle Street, between Broadway and Sheridan, is the site of “New China Town.” It is an an area with a number of street robberies and is a destination area for “Northsiders” who want an inexpensive Chinese or Vietnamese meal.

There is a particularly risky area in the neighborhood of Broadway and Wilson adjacent to Truman Community College. In a previous analysis of the Bronx, Fordham University was shown to be a similar attractor for robbery incidents. Colleges supply good targets for street robbery. Also, authority for security is split between the college and the city police. The area around Broadway and Wilson has been risky for many years. Ninety years ago, it was the northern terminus of rapid transit, and the site of several very inexpensive hotels, two of which still exist. Today the area has several pawn shops and currency exchanges. There is an ATM located in the EL station. The area looks dangerous and dirty. Finally, the area has many blind corners and alleys that could serve as sites for robbery; this is unusual for Chicago. The census block that includes the northwest corner of Broadway and Wilson ranked fifth among Chicago's 21,000 census blocks in number of street robberies in 1999.

Changes need to be made to reduce the risk of street robbery in this area. Mapping identifies a problem with street robberies, but to investigate possible changes it is necessary to go beyond mapping. Aside from changes in patrol practices, what physical changes might aid in crime reduction? The campus has very little parking. The administration assumes that students take public transportation, but many do not. A secure parking garage that could serve both the elevated station and the school could be constructed (vacant land is available). In addition, increased police patrol in the area between the school and the el station could be implemented.

**Advantages of STAC**

STAC has a number of advantages as a clustering algorithm:

1. STAC can analyze a very large number of cases quickly. It is very fast using a Euclidean projection such as UTM or State Plane, and not quite as fast using spherical coordinates (latitude/longitude).

2. The STAC user controls the approximate size of the ellipses (search radius), the minimum number of points per ellipse, and the study area. These
features allow for a broad search for Hot Spot Areas over an entire city and a second search concentrating on a smaller area and deriving focused Hot Spot Areas for local tactical use.

3. STAC and Hierarchical Clustering are complimentary. Hierarchical Clustering first derives small ellipses and then aggregates to larger ones. The recommended STAC procedure is to first derive large scale ellipses and then to analyze these for tactical use.

4. The visual display of STAC ellipses or convex hulls is quite intuitive.

5. Hot spots need not be limited to a single kind of crime, place or even. For example, ellipses of drug crime can be overlain on those for burglary. Some causal factors are also analyzable with STAC ellipses. For example, ellipses of street robbery can be compared to those for liquor licenses.

9. STAC combines features of a hierarchical and partitioning search methods and adapts itself to the size of the clusters.

10. Unlike the Nnh routine, which has a constant threshold (search radius), STAC can create clusters of unequal size because overlapping clusters are combined until there is no overlap.

Limitations of STAC

There are also some limitations to using STAC:

1. The distribution of incidents within clusters is not necessarily uniform. The user should be careful not to assume that it is. A mapped theme of the Mode routine (see chapter 6) according to number of incidents or the single kernel density interpolation (see chapter 8) overlaid with STAC ellipses are good ways to overcome this problem (figure 7.5 above and figure 7.6 below).

2. STAC is based on the distribution of data points. Neither land use nor risk factors is accounted for. It is up to the analyst to identify the characteristics that make a Hot Spot 'hot'.

3. Small changes in the STAC study area boundary can result in quite different depictions of the ellipses. This is true of any clustering routine. Retaining the same reference file over repeated analyses alleviates this problem. The analysis should also be documented for the analysis parameters.

Nevertheless, if used carefully, STAC is a powerful tool for detecting clusters and can allow an analyst to experiment with varying search radii and reference boundaries.

Next, the K-means clustering routine is examined.
Figure 7.6: STAC Robbery Hot Spots and Kernel Density Estimation
**K-Means Partitioning Clustering**

The *K-means* clustering routine (*Kmeans*) is a partitioning procedure where the data are grouped into *K* groups defined by the user. A specified number of seed locations, *K*, are defined by the user (Fisher, 1958; MacQueen, 1967; Aldenderfer and Blashfield, 1984; Systat, 2000). The routine tries to find the best positioning of the *K* centers and then assigns each point to the center that is nearest. Like the Nnh routine, the *Kmeans* assigns points to one, and only one, cluster. However, unlike the nearest neighbor hierarchical (Nnh) procedure, all points are assigned to clusters. Thus, there is no hierarchy in the routine, that is there are no second- and higher-order clusters.

The technique is useful when a user want to control the grouping. For example, if there are 10 precincts in a jurisdiction, an analyst might want to identify the 10 most compact clusters, one for precinct. Alternatively, if a previous analysis has shown there were 24 clusters, then an analyst could check whether the clusters have shifted over time by also asking for 24 clusters. By definition, the technique is somewhat arbitrary since the user defines how many clusters are to be expected. Whether a cluster could be a 'hot spot' or not would depend on the extent to which a user wanted to replicate 'hot spots' or not.

The theory of the K-means procedure is relatively straightforward. The implementation is more complicated. K-means represents an attempt to define an optimal number of *K* locations where the sum of the distance from every point to each of the *K* centers is minimized. It is a variation of the old location theory paradigm of how to locate *K* facilities (e.g., police stations, hospitals, shopping centers) given the distribution of population (Haggett, Cliff, and Frey, 1977). That is, how does one identify *supply* locations in relation to *demand* locations. In theory, solving this question is an empirical solution, what is frequently called *global optimization*. One tries every combination of *K* objects where *K* is a subset of the total population of incidents (or people), *N*, and measures the distance from every incident point to every one of the *K* locations. The particular combination which gives the minimal sum of all distances (or all squared distances) is considered the best solution. In practice, however, solving this is computationally almost impossible, particularly if *N* is large. For example, with 6000 incidents grouped into 20 partitions (clusters), one cannot solve this with any normal computer since there are

\[
\frac{6000!}{20! \cdot 5980!} = 1.456 \times 10^{57}
\]

combinations. No computer can solve that number and few spreadsheets can calculate the factorial of *N* greater than about 127.\(^3\) In other words, it is almost impossible to solve computationally.

Practically, therefore, the different implementations of the K-means routine all make initial guesses about the *K* locations and then optimize the seating of this location in relation to the nearby points. This is called *local optimization*. Unfortunately, each K-means routine has a different way to define the initial locations so that two K-means...
procedures will usually not produce the same results, even if $K$ is identical (Everitt, 1974; Systat, Inc., 1994).

**CrimeStat K-means Routine**

The K-means routine in CrimeStat also makes an initial guess about the $K$ locations and then optimizes the distribution locally. The procedure that is adopted makes initial estimates about location of the $K$ clusters (seeds), assigns all points to its nearest seed location, re-calculates a center for each cluster which becomes a new seed, and then repeats the procedure all over again. The procedure stops when there are very few changes to the cluster composition.

The default K-means clustering routine follows an algorithm for grouping all point locations into one, and only one, of these $K$ groups. There are two general steps: 1) the identification of an initial guess (seed) for the location of the $K$ clusters, and 2) local optimization which assigns each point to the nearest of the $K$ clusters. A grid is overlaid on the data set and the number of points falling within each grid cell is counted. The grid cell with the most points is the initial first cluster. Then, the second initial cluster is the grid cell with the next most points that is separated by at least:

$$\text{Separation} = \frac{A}{t \times 0.5 \times \sqrt{\frac{N}{\text{--------}}}}$$  

(7.1)

where $t$ is the Student's t-value for the .01 significance level (2.358), $A$ is the area of the region, and $N$ is the sample size. A third initial cluster is then selected which is the grid cell with the third most points and is separated from the first two grid cells by at least the separation factor defined above. This process is repeated until all $K$ initial seed locations are chosen.

The algorithm then conducts local optimization. It assigns each point to the nearest of the $K$ seed locations to form an initial cluster. For each of the initial clusters, it calculates the center of minimum distance and then re-assigns all points to the nearest cluster, based on the distance to the center of minimum distance. It repeats this process until no points change clusters. To increase the flexibility of the routine, the grid that is overlaid on the data points is re-sized to accommodate different cluster structures, increasing or decreasing in size to try to find the $K$ clusters. After iterating through different grid sizes, the code makes sure that the final seeds are from the "best" grid or the grid that produces the most clusters. Finally, for each cluster, the routine calculates a standard deviational ellipse and optionally can output the results graphically as either standard deviational ellipses or a convex hulls.
Control over Initial Selection of Clusters

Changing the separation between clusters

One problem with this approach is that in highly concentrated distributions, such as with most crime incidents in a metropolitan area, the separation between clusters may not be sufficiently large to detect clusters farther away from the concentration; the algorithm will tend to sub-divide concentrated groupings of incidents into multiple clusters rather than seek clusters that are less concentrated and, usually, farther away. To increase the flexibility of the routine, CrimeStat allows the user to modify the initial selection of clusters since this has a large effect on the final grouping (Everett, 1974). There are two ways the initial selection of cluster centers can be modified. The user can increase or decrease the separation factor. Formula 7.1 is still used to separate each of the initial clusters, but the user can either select a t-value from 1 to 10 from the drop down menu or write in any number for the separation, including fractions, to increase or decrease the separation between the initial clusters. The default is set at 4.

Figure 7.7 shows a simulation of eight clusters, four of which have higher concentrations than the other two. Two partitions of the data set into eight groups are shown, one using a separation of 4 (dashed green ellipses) and one with a separation of 15 (solid blue ellipses). As seen, the partition with the larger separation captures the eight clusters better. With the smaller separation, the routine will tend to sub-divide more concentrated clusters because that reduces the distance of each point from the cluster center. Depending on the purpose of the partitioning, a greater or lesser separation may be desired.

Selecting the initial seed locations

Alternatively, the initial clusters can be modified to allow the user to define the actual locations for the initial cluster centers. This approach was used by Friedman and Rubin (1967) and Ball and Hall (1970). In CrimeStat, the user-defined locations are entered with the secondary file which lists the location of the initial clusters. The routine reads the secondary file and uses the number of points in the file for K and the X/Y coordinates of each point as the initial seed locations. It then proceeds in the same way with local optimization. When eight points that were approximately in the middle of the eight clusters in figure 7.7 were input as the secondary file, the K-means routine immediately identified the eight clusters (results not shown). Again, depending on the purpose the user can test a particular clustering by requiring the routine to consider that model, at least for the initial seed location. The routine will conduct local optimization for the rest of the clustering, as in the above method.

The K-means output is similar for both routines. It includes the parameters for the standard deviational ellipse of each cluster in the table. In addition, graphically one can output each cluster as a standard deviational ellipse or as a convex hull (see chapter 4). The convex hull draws a polygon around all the points in a cluster. Hence it is a literal
Figure 7.7:  
Separated Data and K-Means Solution  
K=8 Paritions with Two Separations of Initial Seed Locations

Separation=4

Separation=15

Miles

0 5 10

D
description of the extent of the cluster. The ellipse, on the other hand, is an abstraction for
the cluster. Typically, one standard deviation will cover more than 50% of the cases, one
and a half standard deviations will cover more than 90% of the cases, and two standard
deviations will cover more than 99% of the cases, although the exact percentage will
depend on the distribution. In general, use a 1X standard deviational ellipse since 1.5X
and 2X standard deviations can create an exaggerated view of the underlying cluster. The
ellipse, after all, is an abstraction from the points in the cluster which may be arranged in
an irregular manner. On the other hand, for a regional view, a convex hull or a one
standard deviational ellipse may not be very visible. The user has to balance the need to
accurately display the cluster compared to making it easier for a viewer to understand its
location.

**Mean squared error**

In addition, the output for each cluster lists two additional statistics:

\[
\text{Sum of squares of cluster } \mathcal{C} = \sum_{i=1}^{N_C} \left\{ [X_{i \mathcal{C}} - \text{Mean } X_{\mathcal{C}}]^2 + [Y_{i \mathcal{C}} - \text{Mean } Y_{\mathcal{C}}]^2 \right\} \quad (7.2)
\]

\[
\text{Mean squared error of cluster } \mathcal{C} = \frac{\text{SSE}_{\mathcal{C}}}{(N_C - 1)} \quad (7.3)
\]

where \(X_{i \mathcal{C}}\) is the X value of a point that belongs to cluster \(\mathcal{C}\), \(Y_{i \mathcal{C}}\) is the Y value of a point
that belongs to cluster \(\mathcal{C}\), MeanX\(_{\mathcal{C}}\) is the mean X value of cluster \(\mathcal{C}\) (i.e., of only those points
belonging to \(\mathcal{C}\)), MeanY\(_{\mathcal{C}}\) is the mean Y value of cluster \(\mathcal{C}\), and \(N_C\) is the number of points in
cluster \(\mathcal{C}\). There is also a total sum of squares and a total mean square error which is
summed over all clusters

\[
\text{Total Sum of Squares} = \sum_{\mathcal{C}} \text{SSE}_{\mathcal{C}} \quad (7.4)
\]

\[
\text{Total Mean Squared Error} = \sum_{\mathcal{C}} \frac{\text{SSE}_{\mathcal{C}}}{(N-K-1)} \quad (7.5)
\]

where SSE\(_{\mathcal{C}}\) is the sum of squares for cluster \(\mathcal{C}\), N is the total sample size, and K is the
number of clusters. The sum of squares is the squared deviations of each cluster point
from the center of minimum distance while the mean squared error is the average of the
squared deviations for each cluster.

The sum of squares (or sum of squared errors) is frequently used as a criteria for
identifying 'goodness of fit' (Everett, 1974; Aldenderfer and Blashfield, 1984; Gersho and
Gray, 1992). In general, for a given number of clusters, K, those with a smaller sum of
squares and, correspondingly, smaller mean square error are better defined than clusters
with a larger sum of squares and larger mean squared error. Similarly, a K-means
solution that produces a smaller overall sum of squares is a tighter grouping than a grouping that produces a larger overall sum of squares.

But, there can be exceptions. If there are points which are ‘outliers’, that is which don’t obviously fall into one cluster or another, re-assigning them to one or another cluster can distort the sum of squares statistics. Also, in highly concentrated distributions, such as with crime incidents, a smaller sum of squares criteria can be obtained by splitting the concentrations rather than clustering less central and less dense groups of incidents (such as in figure 7.7); the results, while minimizing the sum of squared errors from the cluster centers, will be less desirable because the peripheral clusters are ignored. Thus, these statistics are presented for the user’s information only. In assigning points to clusters, CrimeStat still uses the distance to the nearest seed location, rather than a solution that minimizes the sum of squared distances.

**Visualizing the Cluster**

Finally, the K-means clustering routine (Kmeans) outputs clusters graphically as either ellipses or convex hulls, similar to the other clustering routines. For the ellipses, the user can choose between 1X, 1.5X, and 2X standard deviations to display the ellipses. The graphical ellipses are output with the prefix ‘KM’ before the file name. It should be noted, however, that the ellipses are an abstraction of the cluster. The clusters are not necessarily arranged in ellipses. They are for visualization purposes only. For the convex hull, the routine draws a polygon around the points in each cluster. The graphical convex hulls are output with the prefix “CKM” before the file name.

**K-means Output Files**

The naming system for the K-means outputs is simpler than the Nnh routine since there are no higher-order clusters. Each file is named

- Km<username>
- Ckm<username>

where *username* is the name of the file provided by the user. Within the file, each cluster is named

- KmEll<N><username>
- CkmHull<N><username>

where *N* is the cluster number and *username* is the name of the file provided by the user. For example,

KmEll3robbery

is the third ellipse for the file called ‘robbery’ and
CkmHull12burglary

is the 12th convex hull for the file called 'burglary'.

For the ellipses, a slide-bar allows ellipses to be defined for 1X, 1.5X, and 2X standard deviations and can be output in ArcView '.shp', MapInfo '.mil' or Atlas*GIS '.bna' formats. The convex hulls, on the other hand, draw a polygon around the clustered points.

**Example 2: K-means Clustering of Street Robberies**

In *CrimeStat*, the user specifies the number of groups to sub-divide the data. Using the 1996 robbery incidents for Baltimore County, the data were partitioned into 10 groups with the K-means routine (figure 7.8). As can be seen, the clusters tend to fall along the border with Baltimore City. But there are three more dispersed clusters, one concentrated in the central eastern part of the county and two north of the border with the City. Because these clusters are very large, a finer mesh clustering was conducting by partitioning the data into 31 clusters (figure 7.9). Thirty-five clusters were requested but the routine only found 31 seed location. Consequently, it outputted 31 clusters, which are displayed as ellipses. Though the ellipses are still larger than those produced by the nearest neighbor hierarchical procedure (see figure 6.7 in chapter 6), there is some congruency; clusters identified by the nearest neighbor procedure have corresponding ellipses using the K-means procedure.

Figure 7.10 shows a section of southwest Baltimore County with four full clusters and three partial clusters visible, displayed as convex hulls. Looking at the distribution, several clusters make intuitive sense while a couple of others do not. For example, two clusters highlight a concentration along a major arterial (U.S. Highway 40). Similarly, the cluster in the middle right appears to capture incidents along two arterial roads. However, the other three full clusters do not appear to capture meaningful patterns and appear somewhat arbitrary.

Other uses of the K-means algorithm are possible. One problem that affects most police departments is the need to allocate personnel throughout a city in a balanced and fair way. Too often, some police precincts or districts are overburdened with Calls for Service whereas others have more moderate demand. The issue of re-drawing or re-assigning police boundaries in order to re-establish balance is a continual one for police departments. The K-means algorithm can help in defining this balance, though there are many other factors that will affect particular boundaries. The number of groupings, K, can be chosen based on the number of police districts that exist or that are desired. The locations of division or precinct stations can be entered in a secondary file in order to define the initial 'seed' locations. The K-means routine can then be run to assign all incidents to each of the K groups. The analyst can vary the location of the initial seeds or, even, the number of groups in order to explore different arrangements in space. Once an agreed upon solution is found, it is easy to then re-assign police beats to fit the new arrangement.
Figure 7.8:
Baltimore County Robbery 'Hot Spots'
Using K-Means Routine with K=10 Clusters

Baltimore County

City of Baltimore

Miles

0 2 4
Figure 7.9:
Baltimore County Robbery 'Hot Spots'
Using K-Means Routine with K=31 Clusters
Figure 7.10:
Southwest Baltimore County Robbery 'Hot Spots'
Using K-Means Routine with K=31 Clusters
Advantages and Disadvantages of the K-means Procedure

In short, the K-means procedure will divide the data into the number of groups specified by the user. Whether these groups make any sense or not will depend on how carefully the user has selected clusters. Choosing too many will lead to defining patterns that don't really exist whereas choosing too few will lead to poor differentiation among neighborhoods that are distinctly different.

It is this choice that is both a strength of the technique as well as a weakness. The K-means procedure provides a great deal of control for the user and can be used as an exploratory tool to identify possible 'hot spots'. Whereas the nearest neighbor hierarchical method produces a solution based on geographical proximity with most clusters being very small, the K-means can allow the user to control the size of the clusters. In terms of policing, the K-means is better suited for defining larger geographical areas than the nearest neighbor method, perhaps more appropriate for a patrol area than for a particular 'hot spot'. Again, if carefully used, the K-means gives the user the ability to 'fine tune' a particular model of 'hot spots', adjusting the size of the clusters (vis-a-vis the number of clusters selected) in order to fit a particular pattern which is known.

Yet it is this same flexible characteristic that makes the technique potentially difficult to use and prone to misuse. Since the technique will divide the data set into $K$ groups, there is no assumption that these $K$ groups represent real 'hot spots' or not. A user cannot just arbitrarily put in a number and expect it to produce meaningful results. A more extensive discussion of this issue can be found in Murray and Grubesic (2002). Grubesic and Murray (2001) present some newer approaches in the K-means methodology.

The technique is, therefore, better seen as both an exploratory tool as well as a tool for refining a 'hot spot' search. If the user has a good idea of where there should be 'hot spots', based on community experience and the reports of beat officers, then the technique can be used to see if the incidents actually correspond to the perception. It also can help identify 'hot spots' which have not been perceived or identified by officers. Alternatively, it can identify 'hot spots' that don't really exist and which are merely by-products of the statistical procedure. Experience and sensitivity are needed to know whether an identified 'hot spot' is real or not.

Anselin’s Local Moran Statistic (LMoran)

The last 'hot spot' technique in CrimeStat is a zonal technique called the Anselin's Local Moran statistic and was developed by Luc Anselin (1995). Unlike the nearest neighbor hierarchical and K-means procedures, the local Moran statistic requires data to be aggregated by zones, such as census block groups, zip codes, police reporting areas or other aggregations. The procedure applies Moran's I statistic to individual zones, allowing them to be identified as similar or different to their nearby pattern.
The relationship between land use and the transportation system is an important issue. Many planners recognize that transportation policies, practices and outcomes affect changes in land use, and vice versa, but there is disagreement as to how best to describe this phenomenon. Traditional methods include measures of accessibility via a matrix of zones (tracts, traffic analysis zones, etc.). However, there are limits to the way interaction and accessibility is described with such discrete units.

Through the use of K-Means clustering, an alternate measure of accessibility can be calculated. Rather than relying on census geography, the left map shows ten retail clusters in San Diego County (1995) as calculated by CrimeStat’s K-Means clustering technique (using 1x standard deviational ellipse). The retail hot spots were calculated using a geocoded point file of retail establishments in the county. These clusters are not bound by census geography and allow a more realistic appraisal about the attractiveness of specific regions within the county. An analyst can then determine if residential location within a hot spot has an effect on travel patterns, or if there is a relationship between proximity to a hot spot and travel behavior. While this example illustrates a measure of regional retail attractiveness, the flexibility of CrimeStat allows an analyst to evaluate these relationships on a local level, thus allowing a scope of inquiry from regional to local accessibility (as shown in right map, which uses the same parameters as the left figure, but limiting its sample to retail in a sub-region of San Diego County noted by the arrow).
Hot Spot Verification in Auto Theft Recoveries

Bryan Hill
Glendale Police Department
Glendale, AZ

We use CrimeStat as a verification tool to help isolate clusters of activity when one application or method does not appear to completely identify a problem. The following example utilizes several CrimeStat statistical functions to verify a recovery pattern for auto thefts in the City of Glendale (AZ). The recovery data included recovery locations for the past 6 months in the City of Glendale which were geocoded with a county-wide street centerline file using ArcView.

First, a spatial density “grid” was created using Spatial Analyst with a grid cell size of 300 feet and a search radius of 0.75 miles for the 307 recovery locations. We then created a graduated color legend, using standard deviation as the classification type and the value for the legend being the CrimeStat “Z” field that is calculated.

In the map, the K-means (red ellipses), Nnh (green ellipses) and Spatial Analyst grid (red-yellow grid cells) all showed that the area was a high density or clustering of stolen vehicle recoveries. Although this information was not new, it did help verify our conclusion and aided in organizing a response.
The basic concept 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. 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 *LISAs* for all observations is proportional to the global indicator of spatial association.

\[
L_i = f(Y_i, Y_{ij})
\]  
(7.6)

where \(L_i\) is the local indicator, \(Y_i\) is the value of an intensity variable at location \(i\), and \(Y_{ij}\) are the values observed in the neighborhood \(J_i\) of \(i\).

In other words, a *LISA* is an indicator of the extent to which the value of an observation is similar or different from its neighboring observations. This requires two conditions. First, that each observation has a variable value that can be assigned to it (i.e., an intensity or a weight) in addition to its X and Y coordinates. For crime incidents, this means data that are 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 or all other zones negatively weighted by the distance from the observation zone.

Once these are defined, the *LISA* indicates the value of the observation zone in relation to its neighborhood. Thus, in neighborhoods where there are 'high' intensity values, the *LISA* indicates whether a particular observation is similar (i.e., also 'high') or different (i.e., low) and, conversely, in neighborhoods where there are 'low' intensity values, the *LISA* indicates whether a particular observation is similar (i.e., also 'low') or different (i.e., 'high'). That is, the *LISA* is an indicator of similarity, not absolute value of the intensity variable.

**Formal Definition of Local Moran Statistic**

*The I$_i$ statistic*

Anselin (1995) has applied the concept to a number of spatial autocorrelation statistics. The most commonly used, which is included in *CrimeStat*, is Anselin's Local Moran statistic, \(I_i\), the use of Moran's I statistic as a *LISA*. The definition of \(I_i\) is (from Getis and Ord, 1996):

\[
I_i = \frac{(Z_i - \bar{Z})}{S_Z^2} \sum_{j=1}^{N} W_{ij} \cdot (Z_j - \bar{Z})
\]  
(7.7)
where \( \bar{Z} \) is the mean intensity over all observations, \( Z_i \) is the intensity of observation \( i \), \( Z_j \) is intensity for all other observations, \( j \) (where \( j \neq i \)), \( S_{Z}^2 \) is the variance over all observations, and \( W_{ij} \) is a distance weight for the interaction between observations \( i \) and \( j \). Note, the first term 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).

**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:

\[
W_{ij} = \frac{1}{d_{ij}} \quad (7.8)
\]

where \( d_{ij} \) is the distance between the observation zone, \( i \), and another zone, \( j \). Thus, 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. Depending on the distance scale used (miles, kilometers, meters), the weight index becomes problematic when the distance falls below 1 (i.e., below 1 mile, 1 kilometer); the weight then increases as the distance decreases, going to infinity for \( d_{ij} = 0 \). To correct for this, *CrimeStat* includes an adjustment for small distances so that the maximum weight can be never be greater than 1.0 (see chapter 4). The adjustment scales distances to one mile. When the small distance adjustment is turned on, the minimal distance is scaled automatically to be one mile. The formula used is

\[
W_{ij} = \frac{\text{one mile}}{\text{one mile} + d_{ij}} \quad (7.9)
\]

in whichever units are specified.
Similarity or dissimilarity

An exact test of significance has not been worked out because the distribution of the statistic is not known. The expected value of $I_i$ and the variance of $I_i$ are somewhat complicated (see endnote 7 for the formulas). Instead, high positive or high negative standardized scores of $I_i$, $Z(I_i)$, are taken as indicators of similarity or dissimilarity. A high positive standardized score indicates the spatial clustering of similar values (either high or low) while a high negative standardized score indicates a clustering of dissimilar values (high relative to a neighborhood that is low or, conversely, low relative to a neighborhood that is high). The higher the standardized score, the more the observation is similar (positive) or dissimilar (negative) to its neighbors.

In other words, the Local Moran statistic is a good indicator of either ‘hot spots’ or ‘cold spots’, zones which are different from their neighborhood. ‘Hot spots’ would be seen where the number of incidents in a zone is much higher than in the nearby zones. ‘Cold spots’ would be seen where the number of incidents in a zone is much lower than in the nearby zones. The Local Moran statistic indicates whether the zone is similar or dissimilar to its neighbors. A user must then look at the absolute value of the zone (i.e., the number of incidents in the zone) to see whether it is a ‘hot spot’ or a ‘cold spot’.

For each observation, CrimeStat calculates the Local Moran statistic and the expected value of the Local Moran. If the variance box is checked, the program will also calculate the variance and the standardized Z-value of the Local Moran. The default is for the variance not to be calculated because the calculations are very intense and may take a long time. Therefore, a user should test how long it takes to calculate variances for a small sample on a particular computer before running the variance routine on a large sample.

Example 3: Local Moran Statistics for 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 with a GIS (Figure 7.11). 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 number of auto thefts and there are a number of block groups within the County with a very high number.

Using these data, CrimeStat calculated the Local Moran statistic with the variance box being checked and the small distance adjustment being used. The range of $I_i$ values varied from -37.26 to +180.14 with a mean of 5.20. The pseudo-standardized Local Moran ‘Z’ varied from -12.71 to 50.12 with a mean of 1.61. Figure 7.12 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 their surrounding 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
Figure 7.11:
1996 Motor Vehicle Thefts
Number of Auto Thefts Per Block Group
thefts, compared to the surrounding block groups. These form a 'cold spot'. Consequently, they appear in dark tones in figure 7.12 indicating that they have high I values (i.e., negative 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 7.12 because this indicates negative spatial autocorrelation, having values that are dissimilar to the surrounding blocks.

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 smaller than two standard deviations below the mean, -2) 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 which are unique (and which should be studied) or, in multivariate analysis, on zones which need to be statistically treated different 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 which are dissimilar from their neighborhood. It is the only statistic that is in CrimeStat that demonstrates dissimilarity. The other 'hot spot' tools will only identify areas with high concentrations. 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 intensities, this will entail additional work by a law enforcement agency.

Some Thoughts on the Concept of 'Hot Spots'

Advantages

The seven techniques discussed in this and the last chapter have both advantages and disadvantages. Among the advantages are that they attempt to isolate areas of high concentration (or low concentration in the case of the Local Moran statistic) of incidents and can, therefore, help law enforcement agencies focus their resources on these areas. One of the powerful uses of a 'hot spot' concept is that it is focused. It can provide new information about locations that police officers or community workers may not recognize (Rengert, 1995). Given that most police departments are understaffed, a strategy that prioritizes intervention is very appealing. The 'hot spot' concept is imminently practical.

Another advantage to the identification of 'hot spots' is that the techniques systematically implement an algorithm. In this sense, they minimize bias on the part of officers and analysts since the technique operates somewhat independently of preconceptions. As has been mentioned, however, these techniques are not totally without human judgement since the user must make decisions on the number of 'hot spots' and the size of the search radius, choices that can allow different users to come to different
Figure 7.12:
Local Spatial Autocorrelation of 1996 Vehicle Thefts
Local Moran Z-Value of Block Groups

- Z<-2.58
- Z>-2.58 and Z<=-1.96
- Z>-1.96 and Z<=0
- Z>0 and Z<=1.96
- Z>1.96 and Z<=2.58
- Z>2.58
- No Information
Using Local Moran’s I to Detect Spatial Outliers in Soil Organic Carbon Concentrations in Ireland

Chaosheng Zhang\textsuperscript{1}  
Lecturer in GIS  
\textsuperscript{1}Department of Geography, National University of Ireland, Galway, Ireland

David McGrath\textsuperscript{2}  
Research Officer  
\textsuperscript{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.

\textit{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.
conclusions. There is probably no way to get around subjectivity since law enforcement personnel may not use a result unless it partly confirms what they already know. But, by implementing an algorithm, it forces users to at least go through the steps systematically.

A third advantage is that these techniques are visual, particularly when used with a GIS. The mode and fuzzy mode routines output the results as a dbf file, which can be displayed in a GIS as a proportional circle. The Nnh, Rnnh, Stac, and Kmeans routines can output the results directly as graphical objects, either as standard deviational ellipses or convex hulls; these can be displayed directly in a GIS. The Local Moran technique can be adapted for thematic mapping (as Figure 7.12 demonstrates). Visual information can help crime analysts and officers to understand the distribution of crime in an areas, a necessary step in planning a successful intervention. We should never underestimate the importance of visualization in any analysis.

Limitations

However, there are also some distinct limitations to the concept of a ‘hot spot’, some technical and some theoretical. The choice involved in a user making a decision on how strict or how loose to create clusters allows the potential for subjectivity, as has been mentioned. In this sense, isolating clusters (or ‘hot spots’) can be as much an art as it is a science. There are limits to this, however. As the sample size goes up, there is less difference in the result that can be produced by adjusting the parameters. For example, with 6,000 or more cases, there is very little difference between using the 0.1 significance level in the nearest neighbor clustering routine and the 0.001 significance level. Thus, the subjectivity of the user is more important for smaller samples than larger ones.

A second problem with the ‘hot spot’ concept is that it is usually applied to the volume of incidents and not to the underlying risk. Clusters (or ‘hot spots’) are defined by a high concentration of incidents within a small geographical area, that is on the volume of incidents within an area. This is an implicit density measure - the number of incidents per unit of area (e.g., incidents per square mile). But higher density can also be a function of a higher population at risk.

For some policing policies, this is fine. For example, beat officers will necessarily concentrate on high incident density neighborhoods because so much of their activity revolves around those neighborhoods. From a viewpoint of providing concentrated policing, the density or volume of incidents is a good index for assigning police officers (Sherman and Weisburd, 1995). From the viewpoint of ancillary security services, such as access to emergency medical services, neighborhood watch organizations, or residential burglar alarm retail outlets, areas with higher concentrations of incidents may be a good focal point for organizing these services.

But for other law enforcement policies, a density index is not a good one. From the viewpoint of crime prevention, for example, high incident volume areas are not necessarily unsafe and that effective preventive intervention will not necessarily lead to reduction in crime. It may be far more effective to target high risk areas rather than high volume
areas. In high risk areas, there are special circumstances which expose the population to higher-than-expected levels of crime, perhaps particular concentrations of activities (e.g., drug trading) or particular land uses that encourage crime (e.g., skid row areas) or particular concentrations of criminal activities (e.g., gangs). A prevention strategy will want to focus on those special factors and try to reduce them.

Risk, which is defined as the number of incidents relative to the number of potential victims/targets, is only loosely correlated with the volume of incidents. Yet, 'hot spots' are usually defined by volume, rather than risk. The risk-adjusted hierarchical nearest neighbor clustering routine, discussed in chapter 6, is the only tool among these that identifies risk, rather than volume. It is clear that more tools will be needed to examine hot spot locations that are more at risk.

The final problem with the 'hot spot' concept is more theoretical. Namely, given a concentration of incidents, how do we explain it? To identify a concentration is one thing. To know how to intervene is another. It is imperative that the analyst discover some of the underlying causes that link the events together in a systematic way. Otherwise, all that is left is an empirical description without any concept of the underlying causes. For one thing, the concentration could be random or haphazard; it could have happened one time, but never again. For another, it could be due to the concentration of the population at risk, as discussed above. Finally, the concentration could be circumstantial and not be related to anything inherent about the location.

The point here is that an empirical description of a location where crime incidents are concentrated is only a first step in defining a real 'hot spot'. It is an apparent 'hot spot'. Unless the underlying vector (cause) is discovered, it will be difficult to provide adequate intervention. The causes could be environmental (e.g., concentrations of land uses that attract attackers and victims) or behavioral (e.g., concentrations of gangs). The most one can do is try to increase the concentration of police officers. This is expensive, of course, and can only be done for limited periods. Eventually, if the underlying vector is not dealt with, incidents will continue and will overwhelm the additional police enforcement. In other words, ultimately, reducing crime around a 'hot spot' will need to involve many other policies than simply police enforcement, such as community involvement, gang intervention, land use modification, job creation, the expansion of services, and other community-based interventions. In this sense, the identification of an empirical 'hot spot' is frequently only a window into a much deeper problem that will involve more than targeted enforcement.
Endnotes for Chapter 7

1. STAC is an abbreviation for Spatial and Temporal Analysis of Crime. The temporal section of the program was superceded by several other programs and was not updated for the millennium. Because many law enforcement users refer to STAC ellipses, we have retained that name.

2. The first two digits of a beat number designate the District.

3. The Chicago Police Department made available the incidents in this analysis to Richard Block for the evaluation of the Chicago Alternative Police Strategy (CAPS).

4. In general a designated main surface street occurs every mile on Chicago’s grid, and there are eight blocks to the mile. In this map, Lawrence and Ashland are main Grid streets. In this area, there are also several diagonal main streets that either follow the lake shore or old Indian trails.

5. The total number of ways for selecting K distinct combinations of N incidents, irrespective of order, is (Burt and Barber, 1996, 155):

\[
\frac{N!}{K!(N-K)!}
\]

6. The steps are as follows:

Global Selection of Initial Seed Locations

A. A 100 x 100 grid is overlaid on the point distribution; the dimensions of the grid are defined by the minimum and maximum X and Y coordinates.

B. A separation distance is defined, which is

\[
\text{Separation} = \frac{A}{t \times 0.5 \sqrt{\frac{\text{---}}{N}}}
\]

where \( t \) is the Student’s t-value for the .01 significance level (2.358), \( A \) is the area of the region, and \( N \) is the sample size. The separation distance was calculated to prevent adjacent cells from being selected as seeds.

C. For each grid cell, the number of incidents found are counted and then sorted in descending order.

D. The cell with the highest number of incidents found is the initial seed for cluster 1.
E. The cell with the next highest number of incidents is temporarily selected. If the distance between that cell and the seed 1 location is equal to or greater than the separation distance, this cell becomes initial seed 2.

F. If the distance is less than the separation distance, the cell is dropped and the routine proceeds to the cell with the next highest number of incidents.

G. This procedure is repeated until $K$ initial seeds have been located thereby selecting the remaining cell with the highest number of incidents and calculating its distance to all prior seeds. If the distance is equal to or greater than the separation distance, then the cell is selected as a seed. If the distance is less than the separation distance, then the cell is dropped as a seed candidate. Thus, it is possible that $K$ initial seeds cannot be identified because of the inability to locate $K$ locations greater than the threshold distance. In this case, CrimeStat keeps the number it has located and prints out a message to this effect.

**Local Optimization of Seed Locations**

H. After the $K$ initial seeds have been selected, all points are assigned to the nearest initial seed location. These are the initial cluster groupings.

I. For each initial cluster grouping in turn, the center of minimum distance is calculated. These are the second seed locations.

J. All points are assigned to the nearest second seed location.

K. For each new cluster grouping in turn, the center of minimum distance is calculated. These are third seed locations.

L. Steps J and K are repeated until no more points change cluster groupings. These are the final seed locations and cluster groupings.

7. The formulas are as follows as follows. The expected value of the Local Moran is:

$$E(I_i) = \frac{-\sum_{j=1}^{N} W_{ij}}{N - 1}$$

where $W_{ij}$ is a distance weight for the interaction between observations $i$ and $j$ (either an adjacency index or a weight decreasing with distance). The variance of the Local Moran is defined in three steps:
A. First, define $b_2$.

$$b_2 = \frac{\left( \sum \frac{(X_i - \bar{X})^4}{N} \right)}{\left( \sum \frac{(X_i - \bar{X})^2}{N} \right)^2}$$

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

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

$$2w_{(kh)} = \sum \sum W_{ik} W_{ih} \text{ where } k \neq i \text{ and } h \neq i$$

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 $i \neq j$) 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 $I_i$:

$$\text{Var}(I_i) = \frac{(\sum w_{ij}^2)(n-b_2)}{(n-1)} + \frac{2w_{(kh)}(2b_2 - n)}{(n-1)(n-2)} + \frac{(\sum w_{ij})^2}{(n-1)^2}$$

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

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

8. On one test of 6,051 burglaries with a minimum cluster size requirement of 10 incidents, for example, we obtained 100 first-order clusters, 9 second-order clusters, and no third-order clusters by using a 0.1 significance level for the nearest neighbor hierarchical clustering routine. When the significance level was reduced to 0.001, the number of clusters extracted was 97 first-order clusters, 8 second-order clusters, and no third-order clusters.