Social Network Clustering: An Analysis of Gang Networks

Raymond Ahn
CSULB

Peter Elliott
UCLA

Kyle Luh
HMC

August 5, 2011

Abstract

In Hollenbeck, a gang-dominated region of Los Angeles, gang activity has been monitored by the LAPD. One manner has been in the form of non-criminal stops. We propose a spectral clustering algorithm to predict gang affiliation from the information obtained from these stops. Despite the sparseness of data, the combination of geography and the record of individuals involved in a stop reveals the underlying gang structure and higher order structures (collaboration or rivalry). Having obtained positive results on the real Hollenbeck data, we then develop methods to create gang simulations that will further our understanding of this algorithm and the situation in Hollenbeck.

1 Introduction

Gang violence is a problem that plagues Los Angeles and many other large cities. The social structures in gang territories can transcend individual gangs to include gang cooperation and gang rivalries. There have been efforts to detect these social constructs that rely on criminal stops (see [11], [12], and [13]). We intend to demonstrate that such structure can be extracted from non-criminal stops as well

The rest of the paper follows the following structure: Sections 1.1 and 1.2 introduce Hollenbeck, the area of study, and the data we use to identify social structures in this area. Section 1.3 defines the matrices we use, Section 2 describes the methods we used to identify social structures, Section 3 describes the various methods associated with creating artificial data, and Section 4 describes the results and analysis.

1.1 Hollenbeck

Hollenbeck is one of the policing regions of the Los Angeles Police Department (LAPD). It is located east of Downtown Los Angeles with a population of about 200,000 people in a 15.2 square mile area. According to the LAPD, Hollenbeck is home to 31 known gangs with territories covering almost all of Hollenbeck and is among the top three regions in violent crimes, with homicide rates higher than both Los Angeles and the United States since the early 1990s. LAPD crime data from the years 2004-2006 show that the increase in crime rate was the second highest among all areas the LAPD police [11].

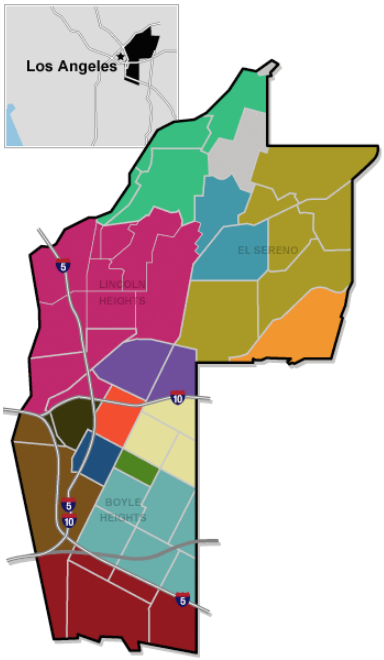


Figure 1: Map of Hollenbeck with gang territories highlighted (courtesy of CNN).

Hollenbeck is bordered by the Los Angeles River to the west, the Pasadena Freeway to the north, Vernon to the south, Pasadena to the northeast, and an unincorporated area of East Los Angeles to the east. According to [11] these natural boundaries serve to limit interactions with other gangs from neighboring areas. Although other gangs in the region might be a cause for concern, there is little interaction between Hollenbeck gangs and those in Pasadena and East Los Angeles. These natural boundaries allow for most of the interactions between gangs to be wholly contained in Hollenbeck, making it a prime location to study. Figure 1 shows the region of Hollenbeck as well as the gang territories.

1.2 Data

The LAPD, with help from UCLA's anthropology department and UCI's criminology department, provided a list of gang members who were involved in strictly non-criminal stops in 2009. This list was obtained by the LAPD patrolling an area and stopping suspected gang members . For each non-criminal incident, we have the following information about each individual: age, sex, ethnicity, gang affiliation, an incident identification number, number of incidents in which the individual as stopped, time, date, and location of each incident, gang territory (including unclaimed territory) in which they were stopped, and whom (if anyone) the individual was stopped with. For individuals involved in multiple incidents, we define each individual's *central location* to be the average location of all incidents. Figure 2 shows the average location of each individual as well as their gang affiliation. As it can be seen in Figure 2 most of the gangs have their members localized in particular areas, while few gangs have members that are spread throughout the area.

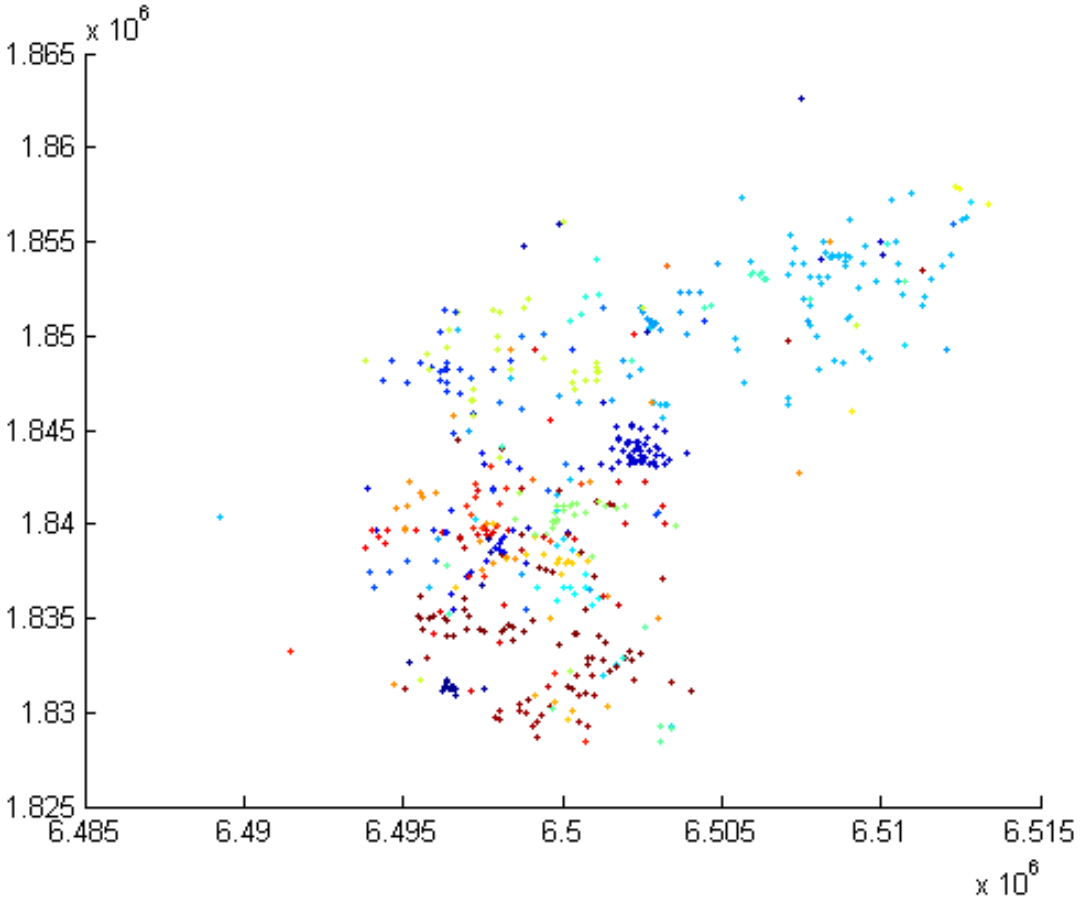


Figure 2: Plot of central location of each of the 748 individuals with points colored by gang affiliation.

To have a ground truth available, we use only some of the information the LAPD has provided. The data contains 1198 individuals, of which only 832 are known Hollenbeck gang members. After removing individuals without an incident identification number or location, we are left with 748 gang members. For the purpose of our project, we will consider only the following information about those 748 individuals: gang affiliation, average location of incident(s), gang territory they were stopped in, and with whom they were stopped with.

1.3 Graph Theory

Graph theory provides a way to study a social network mathematically. By assigning a vertex to each individual and edges as weighted or unweighted connections between two individuals we can use graph theory to define a matrix to represent the social information.

We use different graphs of the form $G(V, E)$ to represent our data. The vertex set $V = \{v_1, v_2, \ldots, v_n\}$ where each v_i represents an individual. By varying the edge set E in each of the different graphs, we can take into account different sets of information. We use matrices to represent these graphs in the following way:

- An unweighted adjacency matrix consisting only of social information where if two individuals v_i and

v_j were stopped together.

$$A_{ij} = \begin{cases} 1 & \text{if } v_i \text{ and } v_j \text{ are connected and } v_i \neq v_j, \\ 0 & \text{otherwise.} \end{cases} \tag{1.1}$$

- A position matrix consisting of the average location of each individual:

$$P_i = \begin{bmatrix} \vdots & \vdots \\ x_i & y_i \\ \vdots & \vdots \end{bmatrix}. \tag{1.2}$$

- A weighted affinity matrix that combines both social and geographical information:

$$W_{ij} = \alpha \exp(\frac{< f_i, f_j >^2}{\sigma_1}) + (1-\alpha)\exp(-\frac{d(v_i, v_j)^2}{\sigma_2}), \tag{1.3}$$

  where f_i is a feature vector for vertex v_i derived from the unweighted adjacency matrix (1.1). f_i is the i-th row of $I + A$, where I is the identity matrix and A is given in (1.1).

  $< f_i, f_j >$ is the dot product between two feature vectors and $d(v_i, v_j)$ is the usual Euclidean distance between the central locations. α is a parameter that weighs the contributions of the social and geographical information to W ($\alpha = 1$ means we are only using social information while $\alpha = 0$ means only geographical information). In order to account for the difference in scale, both feature vectors and distance information are normalized to have unit length.

- A matrix that contains both social and geographical information separately:

$$M = [P|wA] \tag{1.4}$$

  where P is the $n \times 2$ position matrix in (1.2), A is the $n \times n$ adjacency matrix in (1.1), and w is a parameter that weighs A.

Note that for the adjacency matrix (1.1) the diagonal elements, which represent self-connections, are 0, while the weighted affinity matrix (1.3) will have weights on the diagonal. When using both social and geographical information, we will refer to them as "mixed."

2 Methods

We compare four methods for clustering our data set. We give a brief description of each technique and its application. The results of these methods can be seen in Section 4.

2.1 k-means

k-means is one of the simplest unsupervised learning algorithms that attempt to put n data points embedded in $\mathbb{R}^n$ into k clusters. The process is fairly simple, as outlined in the following algorithm [6]:

Algorithm 2.1. k-means clustering algorithm

1. Randomly assign k points to be the initial location of cluster centers (centroids).
2. Assign each point to a cluster based on the nearest centroid.
3. Move each of the k centroids to the center of mass of all points in the corresponding cluster.

4. Repeat steps 2 and 3 until the centroids no longer move.

We applied k-means to the position matrix in (1.2) and the mixed matrix in (1.4). Although k-means is very simple, for the majority of social networks, as well as in our case, k-means by itself will not do a good job (see Section 2.4).

2.2 k-medoids

The difference between k-means and k-medoids is that k-medoids initializes central points (see Algorithm 2.2 step 2) as centroids rather than assigning them randomly. We use the following algorithm proposed by [10].

Algorithm 2.2. k-medoids algorithm

1. Calculate the distance between each pair of points.
2. Calculate $v = \sum_i \frac{d(i,j)}{\sum_l d(i,l)}$ for each point and choose the k smallest values as the initial cluster centroids.
3. Each point is assigned to the nearest cluster centroid.
4. For each cluster, the vertex with the minimum total distance to all other points in the cluster is chosen as the new centroid.
5. Repeat steps 3 and 4 until equilibrium is reached.

We applied k-medoids to the position matrix in (1.2).

2.3 CONCOR

The *convergence of iterated correlations* (CONCOR) is an algorithm that calculates the Pearson product-moment correlation coefficients among rows (or column) of an input matrix by comparing the value of an entry to the mean value of the row (or column) in which it occurs. This results in a new single matrix of calculated correlation coefficients, which represent the structural similarity between vertices. With this new matrix, the process is repeated until all the entries of the matrix are $+1$ or -1, with each number determining a group. The process is then repeated on each subgroup until the number of partitions desired is achieved. See [16] for a more detailed description of the process and the correlation coefficients associated with it.

We applied CONCOR to the affinity matrix in (1.3).

2.4 Spectral Graph Clustering

Spectral clustering is one of the more popular modern clustering algorithms. Rather than describing spectral clustering in great detail, we refer the reader to [15] for a complete analysis of the theory behind spectral clustering. We will be using a modified version of the algorithm proposed by [9]:

Algorithm 2.3. Modified spectral clustering algorithm
Given a set of vertices $V = \{v_1, v_2, \ldots, v_n\}$ that we want to cluster into k groups:

1. Form an affinity matrix $W \in \mathbb{R}^{n \times n}$ as defined in (1.3).
2. Normalize each row of W to have unit length.
3. Find $x_1, x_2, \ldots, x_k$, the m largest eigenvectors of W, and form the matrix $X = [x_1|x_2|\ldots|x_m] \in \mathbb{R}^{n \times m}$.
4. Treat each row of X as a point in $\mathbb{R}^m$ and cluster them into k clusters using k-means (or other methods, see Section 6).

5. Assign the original vertex v_i to cluster j if and only if row i of the matrix X was assigned to cluster j.

The difference between spectral clustering and k-means is that k-means can only create clusters by using linear separation. We illustrate this idea with the example in Figure 3:

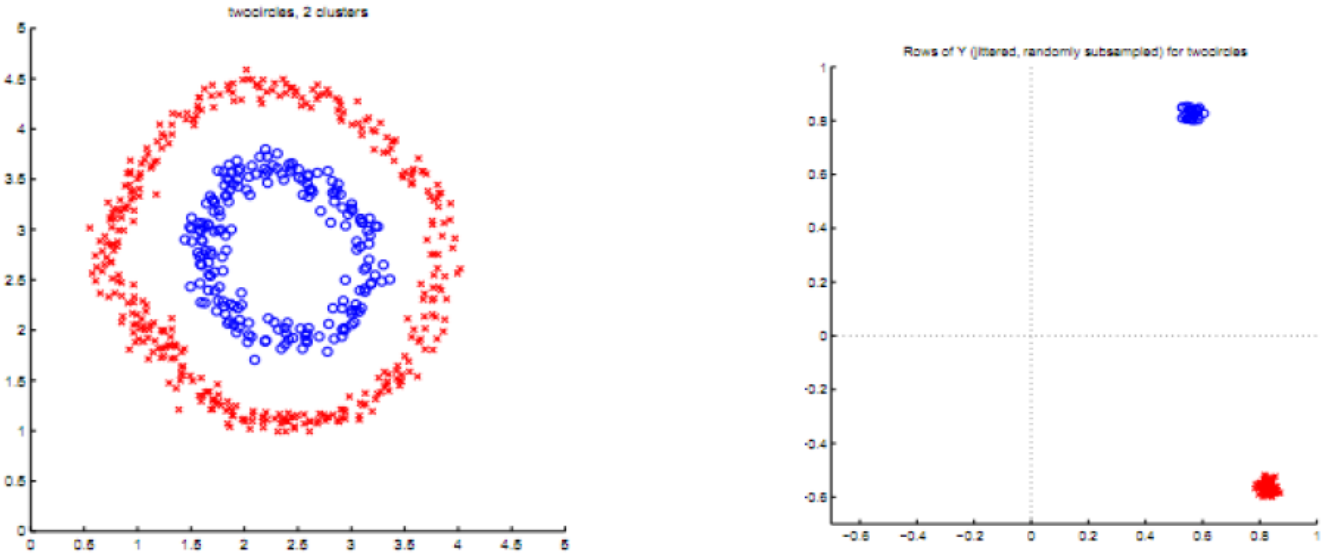


Figure 3: Two concentric circles plot and result of embedding the two eigenvectors into an $\mathbb{R}^2$ eigenspace [9].

If k-means were to be used on this figure, it would cluster the points into two halves since it is impossible to separate the inner circle from the outer circle linearly. By applying spectral clustering and embedding the data points in the span of the two eigenvectors, we see that it creates two linearly separable clusters that k-means can identify. Since gang territories have various shapes and structures, spectral clustering allows us to take those shapes into consideration.

2.4.1 Eigenvalues and eigenvectors

For calculating the eigenvalues and eigenvectors, we use Matlab's *eig* function [8]. The first eigenvalue calculated from the affinity matrix (1.3) is 1 and the corresponding eigenvector a constant vector. A cluster based only on this eigenvector puts all the vertices in one cluster. So we rely on the other non-trivial eigenvectors for information about the clusters. We illustrate the method with the following example in Figure 4:

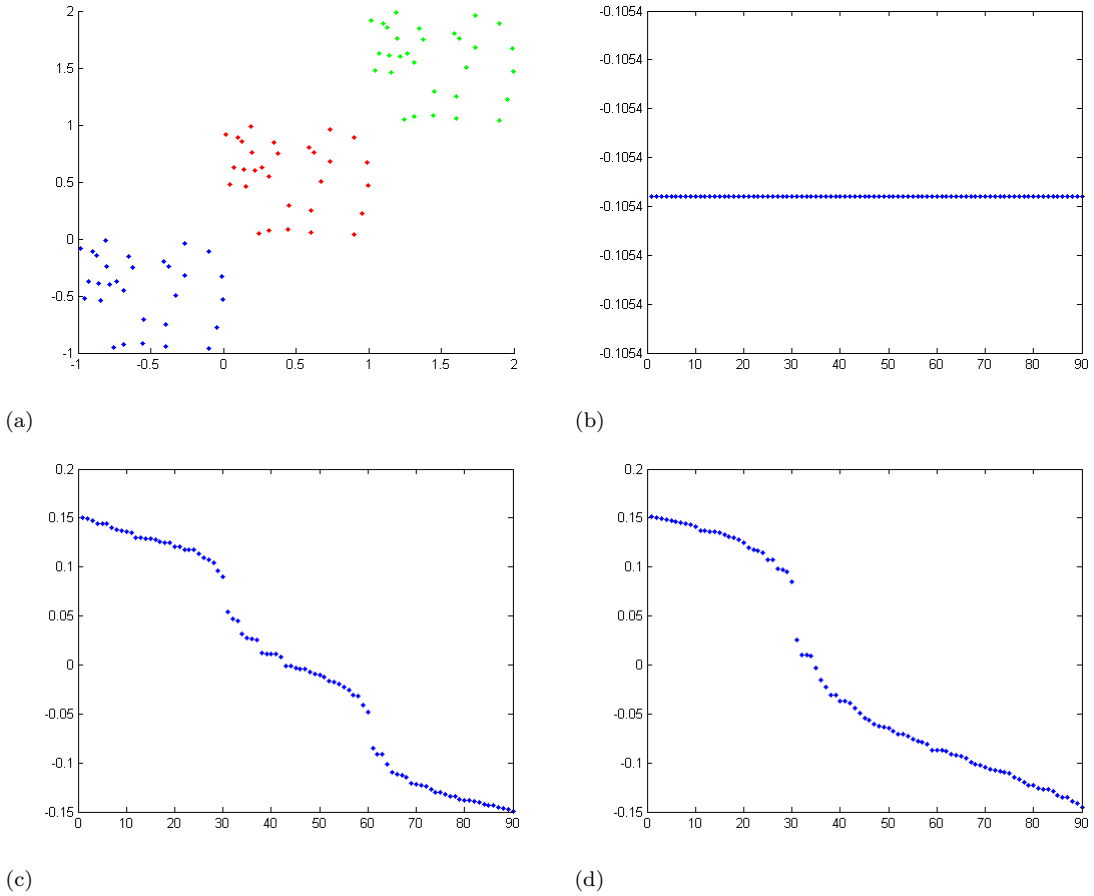
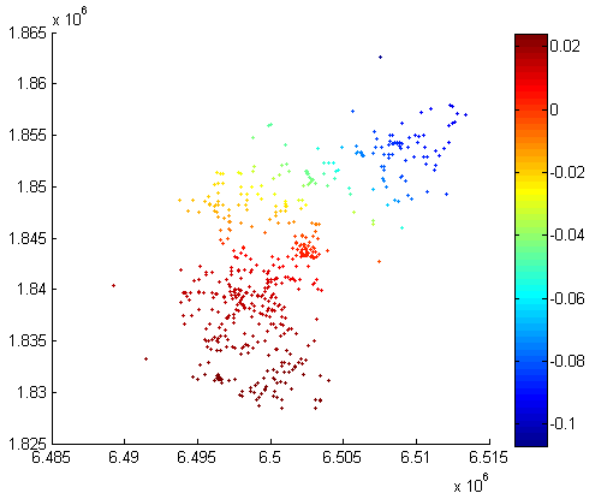


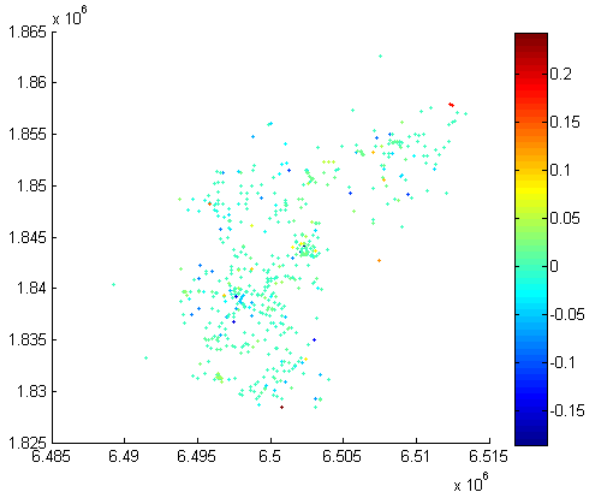
Figure 4: (a) shows a graph of three distinct clusters. (b)-(d) show a plot of the first three eigenvectors.

As stated before, the first eigenvector in Figure 4b is constant and gives no information about the structure of the data set. In Figure 4c the large gaps between the values of the eigenvectors separate the points into the three clusters that are shown in Figure 4a.

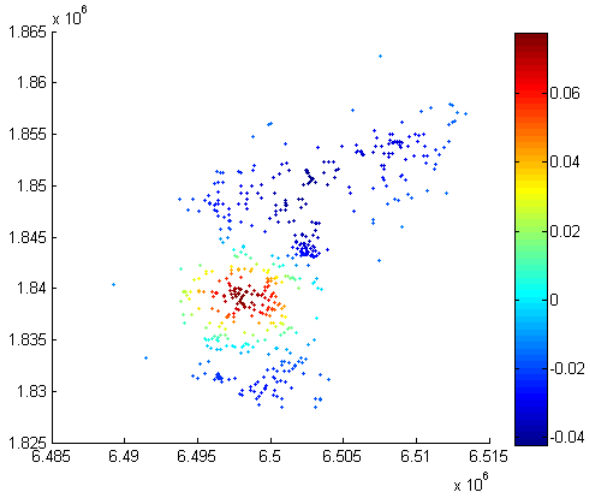
For the Hollenbeck data using either only geographic information ($\alpha = 0$) or social information ($\alpha = 1$) versus using a combination of both ($\alpha \in (0, 1)$), the eigenvectors show different information. To avoid the difficulty of visualizing points in $\mathbb{R}^n$, we apply a color scale to visualize the value of each entry in the eigenvector (see Figure 5) and map it to its corresponding individual. Each individual is then plotted with their average location and color.



(a)



(b)



(c)

Figure 5: Plots of the central location of each individual colored by second eigenvector for three cases. (a) Geography only ($\alpha = 0$), (b) social only ($\alpha = 1$), (c) mixed ($\alpha = 0.5$).

Figure 5 shows the results of color coding each vertex by the value of its corresponding entry in the second eigenvector. By looking at the dark blue or dark red areas we see that Figure 5a shows the results based only on geographical information leading to a north-south gradient and Figure 5b uses only social data leading to very little structure. Figure 5c uses both types of data and shows some of the group structure. The analysis of these results is given in Section 4.2.

Ideally, when groups are infinitely far apart, one would expect k eigenvalues corresponding to k clusters because the rank of the affinity matrix would be k. In this ideal situation, gang members of Hollenbeck do not interact with members outside their own gang and do not step outside the boundaries of their territory. In this case, we expect 31 eigenvalues to represent the 31 gangs. Although we do have an underlying idea as to what the structure of Hollenbeck looks like (the 31 gangs and their territories), we do not know if 31 clusters can correctly capture the social structure of Hollenbeck, and hence we cannot rely on exactly 31 eigenvectors. Instead we use different numbers of eigenvectors to find the best results. Further analysis on the eigenvectors is given in Section 4.2.

3 Artificial Data

As the exact relationship between the known gang affiliations and the "proper" clustering of the Hollenbeck data is an open question, it is important to test the success of clustering algorithms in other ways. One way to develop some ground truth is to generate synthetic data with structural properties that can be controlled. Running the clustering algorithms on that synthetic data can then reveal more about what characteristics of the data have the strongest effect on algorithm performance and how robust the algorithms are in dealing with parameter sensitivity and missing information.

We created an artificial gang database where each data point is a person with three main characteristics: gang affiliation, inter-personal connections, and geographical location. Before creating this data, it is important to consider the distributions of these characteristics in the Hollenbeck data. As the box plot in Figure 6a shows, the majority of the gangs have between 10 and 40 members represented in the data set, with a maximum of 93 and a minimum of 2.

The degree distribution roughly follows a power law, with an estimated exponent of 2.77 and a goodness-of-fit p-value of 0.009 using the methods proposed by [3]. Because the p-value is low, it may be possible to find a better fit for the data. The mean degree is 1.426 and 69.3 percent of connections in the data set are between two people in the same gang. The mean distance from a person to the average location of people in their gang (called the "gang center" in Figure 6c) is 3,312 feet with a standard deviation of 3,297 feet. The mean distance between two gang centers is 9,334 feet with a standard deviation of 5,982 feet. These are the main characteristics the artificial data allows to experiment with.

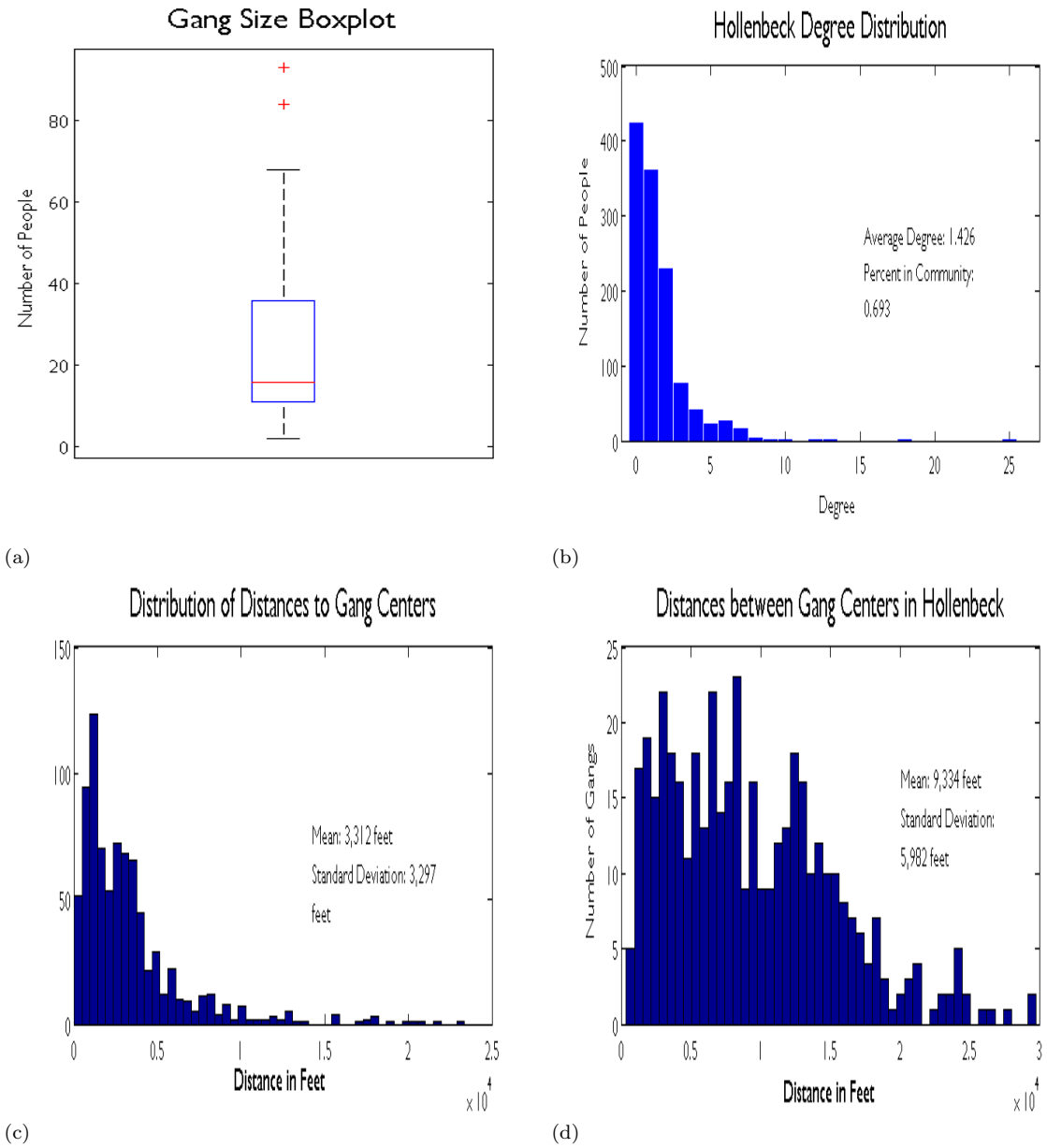


Figure 6: Histograms of various distributions.

We used three different methods to generate synthetic graphs, a power law method, a modification to the Erdős-Rényi model, and geographical thresholding, and we used one model to generate spatial data.

3.1 Power law graph

One common feature of real-world networks is a power law degree distribution [2]. There are several known methods for generating random power law graphs ([1] and [5]). The method we used is most similar to the one proposed by [5]. Using N for the number of nodes, m for the number of communities, k for the desired average degree, μ for the mixing parameter, and γ and β for the exponents of the degree and community

size distributions, respectively, the method is described as follows:

Algorithm 3.1. Power law algorithm

1. Each community size is drawn from a Pareto distribution (a power law probability distribution) with scale 1 and shape β. The community sizes are then multiplied by $\frac{N(\beta-1)}{m\beta}$ and rounded to the nearest integer so that the mean size is approximately $\frac{N}{m}$ and the sum is approximately N. If the sum is greater than N, each community size is reduced by one. If the sum is less than N, a randomly chosen community's size is increased by one. This adjustment continues until the sum of the community sizes is exactly N.

2. Each degree is drawn from a Pareto distribution with scale 1 and shape γ, and we subtract 1 from each degree to account for the possibility of people with no connections. Each degree is then multiplied by $2k/s$, with s the original mean of the distribution, and rounded to the nearest integer so that the mean degree is approximately $2k$. Each degree is then multiplied by $1-\mu$ and μ to get an interior and exterior degree respectively for each person. It is possible that a person will have an interior degree larger than the biggest community size. In that case, the process starts over from scratch at step 1.

3. Each person is randomly assigned to a community. A proposed community is chosen uniformly. If the community already has its allotted number of members or the size of the community is smaller than the interior degree of the person, a new proposed community is drawn.

4. With all communities assigned, connections are made between people. For each person, other people in their community are uniformly chosen to connect with until the interior degree is satisfied. If the chosen person has already satisfied their interior degree, the connection is discarded. This process also stops for each person if there are no new connections after 10,000 iterations. A similar process is used to generate connections between people in different communities.

5. With all the connections created, the actual degree of each individual is calculated. Edges are then randomly discarded until the average degree reaches k.

Though degrees are adjusted after the power law is generated, the degree distribution maintains its power law shape. An example is given in Figure 7 for $\gamma = 3$. The distribution of community sizes is also shown. This method does not create very small communities as seen in the Hollenbeck data. We still think this is accurate because we suspect that the outliers in the Hollenbeck data are a result of sampling methods and lesser gang activity rather than actual anomalously small gangs.

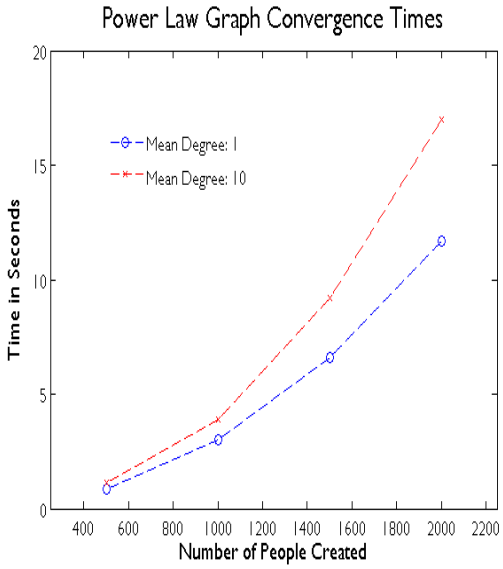
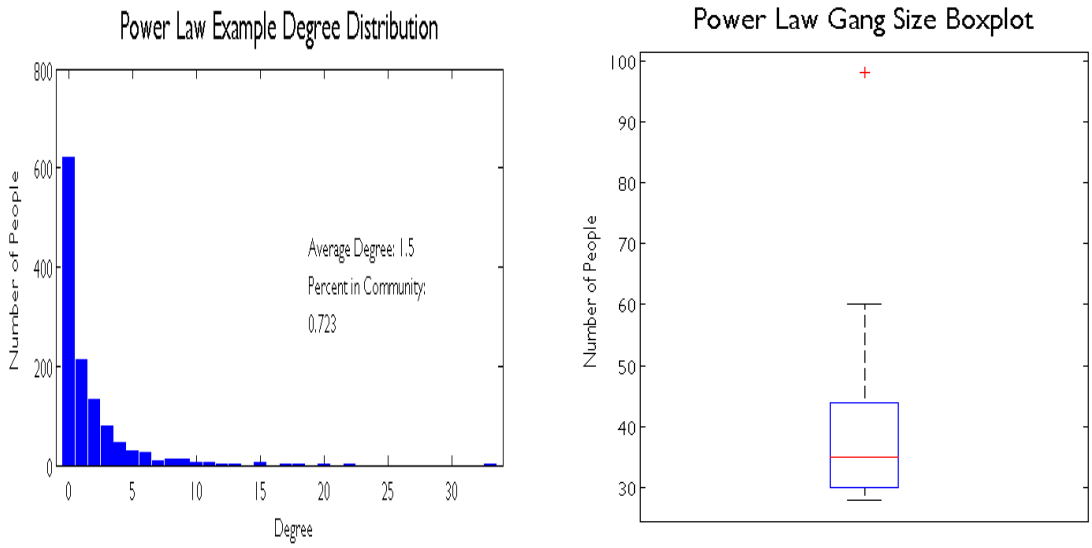


Figure 8: Power law convergence time.



(a) (b)

Figure 7: Power law example degree distribution

Convergence of the algorithm is fast for population sizes comparable to the Hollenbeck data. As can be seen from Figure 8, the average degree chosen also has a small effect on the time. Much of the time to convergence takes place in step 5 of Algorithm 3.1. The algorithm can be tweaked to reduce this, but it would also lead to a greater chance of an average degree of less than the input k. The algorithm also takes significantly longer to converge if a high enough average degree is chosen so that it cannot generate a power law with a maximum interior degree less than the largest community.

3.2 Modified Erdős-Rényi method

One of the most well known random graph models is the Erdős-Rényi model with n vertices and a fixed probability p of having an edge between each vertex [4]. To instead create communities in the graph, we modify the model by using a separate probability for edges within communities and edges outside of communities. As a result, the degree of each vertex follows the sum of two binomial random variables. As the degrees are neither independent nor identically distributed, the expected overall distribution is difficult to calculate, but an example is shown in Figure 9 for interior probability 0.03 and exterior probability 0.0003. Compared to the Hollenbeck and power law graphs, the modified Erdős-Rényi (MER) graphs have many fewer vertices with degree 0, even with a similar average degree.

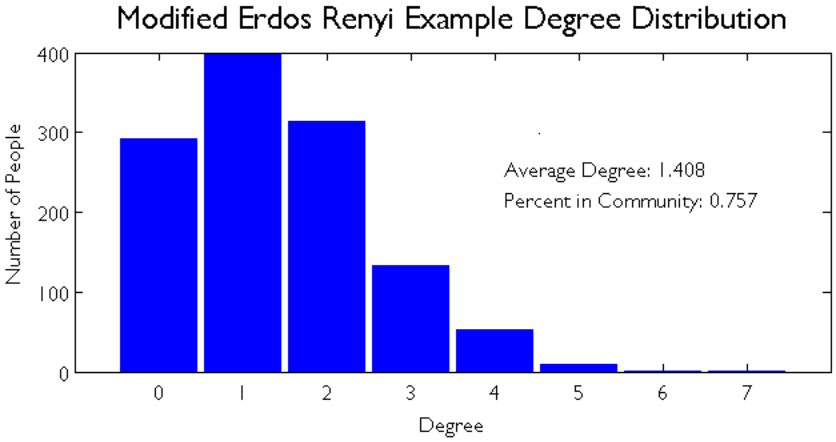


Figure 9: Degree distribution of Modified Erdős-Rényi graph.

Unlike with power law graphs, if a user has a specific average degree and percent within community in mind, the parameter choice for MER graphs is not obvious, but appropriate choices can be estimated using the following equations:

$$r = \frac{\binom{N}{2} - m\binom{N/m}{2}}{(1-\mu)m\binom{N/m}{2}}, \tag{3.1}$$

$$p_1 = \frac{Na}{2(m\binom{N/m}{2} + \frac{1}{r}(\binom{N}{2} - m\binom{N/m}{2}))}, \tag{3.2}$$

$$p_2 = \frac{p_1}{r}, \tag{3.3}$$

where a is the average degree.

Creating MER graphs is even faster than creating power law graphs (compare Figures 8 and 10). The only parameter choice that affects the run-time is the number of people created.

3.3 Geographical Threshold Graph

Unlike the previous two methods, geographical threshold graphs (GTG) [7] use the spatial location of each person when generating the random graph, so we must first use the spatial generator (discussed in 3.4. Once we do have spatial data, the process is as follows:

Algorithm 3.2. Geographical thresholding

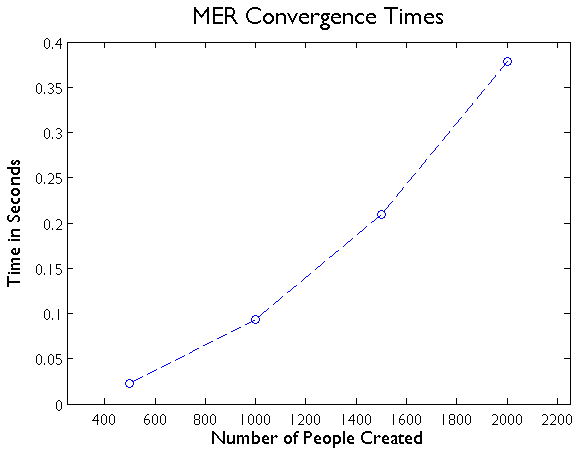


Figure 10: MER convergence time.

1. Each vertex is uniformly assigned a real weight η from 0 to 5.
2. For each pair of vertices x and y, the function $G(x, y)$ is calculated. If $G(x, y)$ is greater than the threshold η, there is an edge between the two vertices.

Many functions can be used in place of G to give different degree distributions and percentages of edges within communities. G returns a degree distribution that follows a power law, as shown in Figure 11.

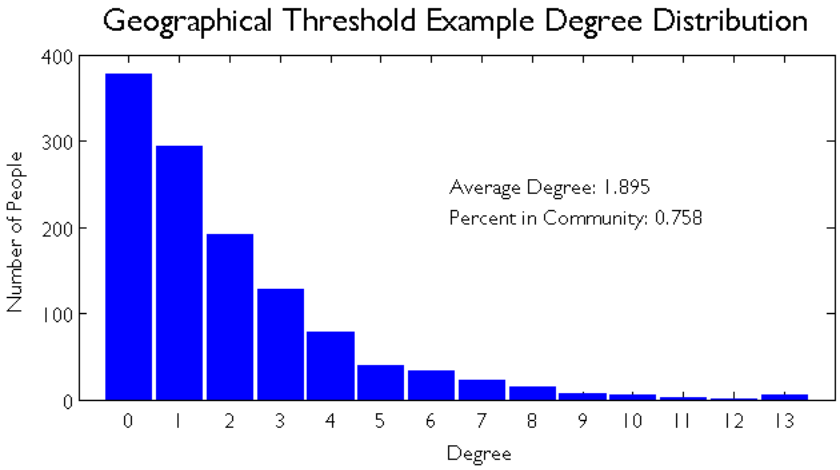


Figure 11: Degree distribution of GTG.

Both the degree distribution and the percentage of edges within communities are heavily dependent on how the spatial coordinates are generated in addition to the choice of parameters for the graph generation. As a result, predicting the average degree based simply on the parameters chosen isn't easy, and it may take several rounds of trial and error to reach a desired result. Unfortunately, the GTG process is easily the slowest, as shown in Figure 12.

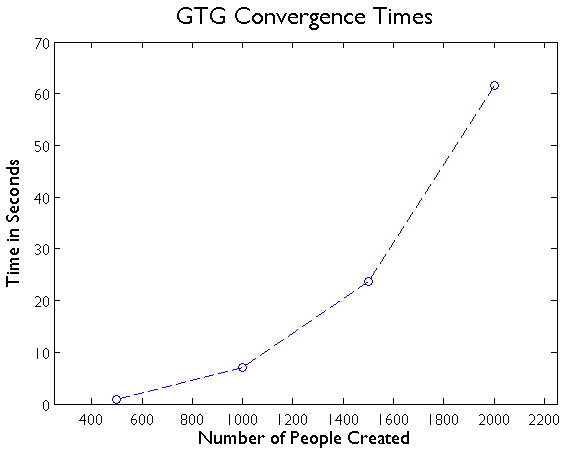
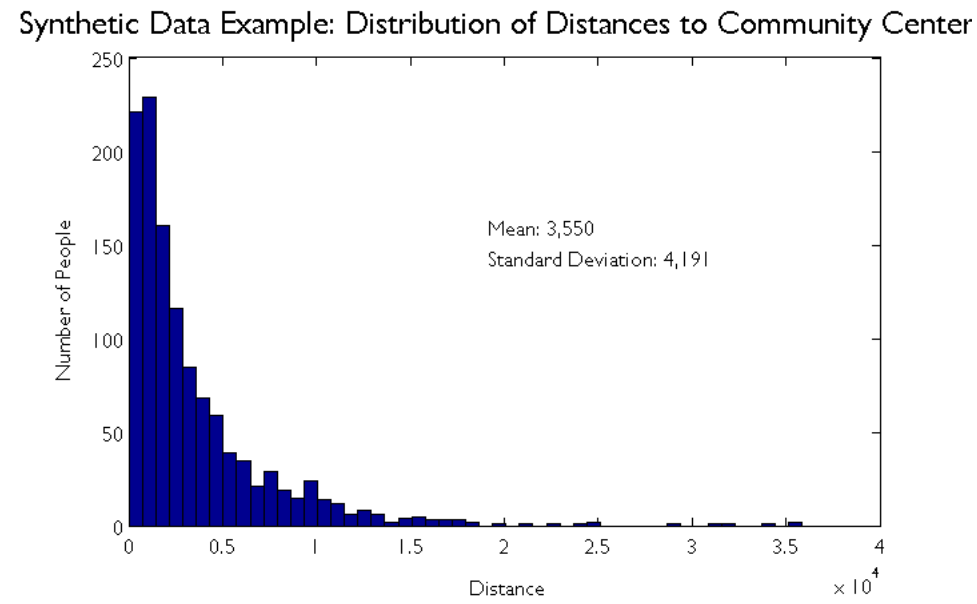


Figure 12: GTG convergence time.

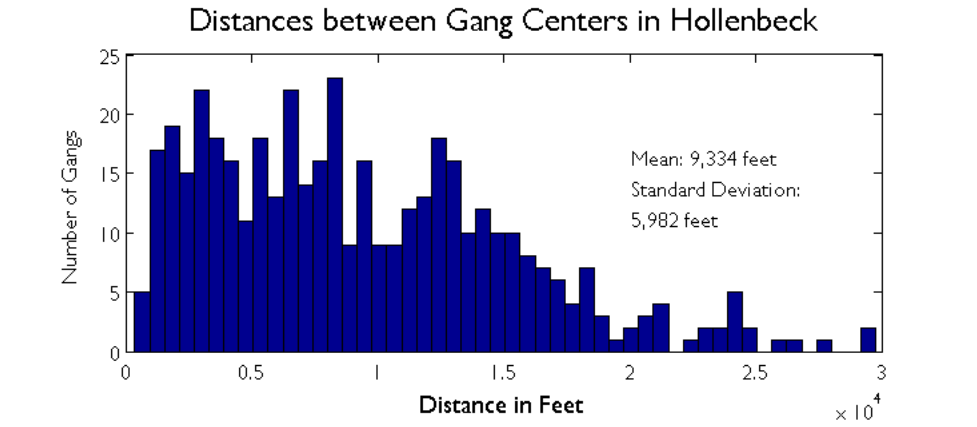
3.4 Spatial Generator

The same spatial generator is used in conjunction with each of the three random graph models. The generator requires a gang affiliation for each person, plus the user can choose the difficulty ("Easy," "Hard," or "Impossible"), which controls the amount of spatial mixing between members of separate gangs.

The first step in the process is to uniformly assign each gang a central location in a rectangle. Then gang members are placed in disks around their gang center. The locations are assigned using polar coordinates, with angle θ distributed uniformly from 0 to 2π and radius r chosen from an exponential distribution. The rate of the exponential distribution is chosen uniformly for each gang, with all members of that gang having the same rate. The difficulty level controls the range of rates for the exponential distributions as well as the size of the rectangle in which the centroids are placed. The "Easy" difficulty setting has higher exponential rates and a bigger rectangle for the gang centers, resulting in more spread out centers but tighter clusters around each center. The "Impossible" difficulty setting gives every gang center the same location. "Hard" is most similar to the Hollenbeck data, with examples shown in Figure 13.



(a)



(b)

Figure 13: Distance distributions for (a) distance to community center and (b) distance between gang centers

The exponential distribution of distances is different from the distribution of gang distances in Hollenbeck. A more detailed model is needed to accurately account for the variance in the shape and spread of gangs in Hollenbeck. Figure 14 shows a generated power law graph with increasing spatial difficulty.

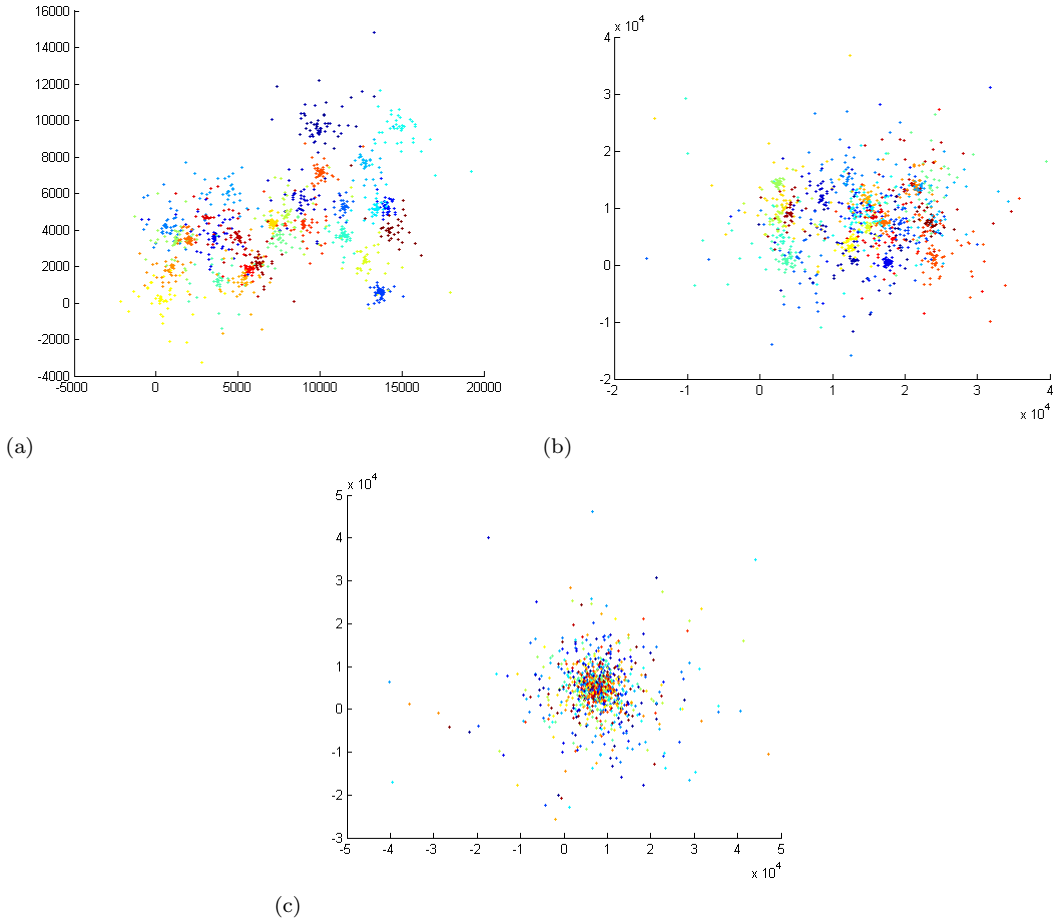


Figure 14: Power law graphs with parameters $k = 1$ and $\mu = 0.5$ with increasing spatial difficulty. (a) easy, (b) hard, (c) impossible.

4 Results

4.1 Measuring quality of results

In order to measure the quality of our results, we use two metrics: purity and adjusted mutual information.

To compute purity, each cluster is assigned to the gang which is present most frequently in the cluster. Purity is defined as:

$$\text{purity}(\Omega, \mathcal{C}) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j|,$$

where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_k\}$ is the set of clusters and $\mathcal{C} = \{c_1, c_2, \ldots, c_j\}$ is the set of gangs. Purity ranges from 0 to 1, with 1 being perfect clustering [6].

Adjusted mutual information is a metric used to compare clusterings. It does so by comparing the expected mutual information to the mutual information as such:

$$AMI(U, V) = \frac{MI(U, V) - E\{MI(T)|a, b\}}{\sqrt{H(U)H(V)} - E\{MI(T)|a, b\}},$$

where U is the true clustering and V is the constructed clustering. MI and $E\{MI\}$ is the mutual information and expected mutual information between U and V, and H is the entropy associated with the clusterings.

The adjusted mutual information gives a value between 0 and 1, with 1 being a perfect clustering. One benefit to using the adjusted mutual information is that it takes into account the variance in sizes of the clusters, since our data consists of gangs with as much as 90 members and as few as 3 members. For more information about the adjusted mutual information and its related equations, we refer the reader to [14].

4.2 Results and analysis

We will now discuss the results from the methods in Section 2 on the Hollenbeck data. All methods were implemented in Matlab [8].

4.2.1 Hollenbeck data

Different methods were applied to different matrices defined in Section 1.3: k-means was done using geographical information and mixed information, (1.2) and (1.4) respectively. (1.2) was used for k-medoids and (1.3) was used for CONCOR. For spectral clustering we used (1.3) with varying parameters.

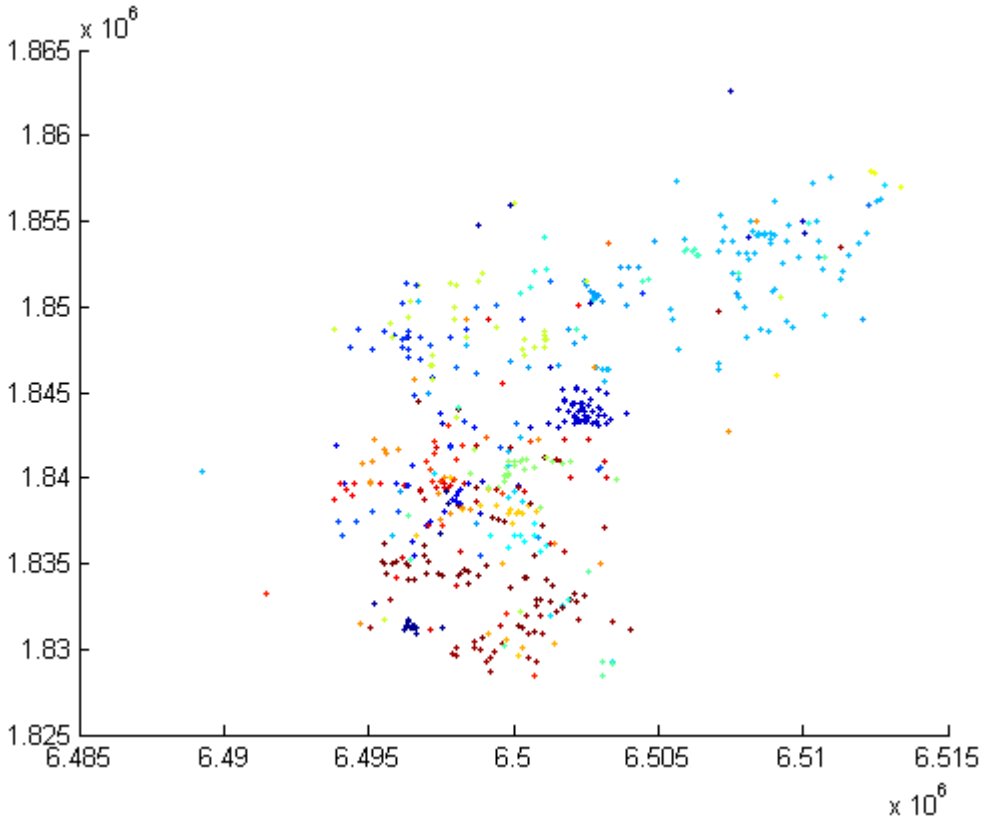


Figure 15: Individuals at their central locations with colors indicating actual gang affiliations

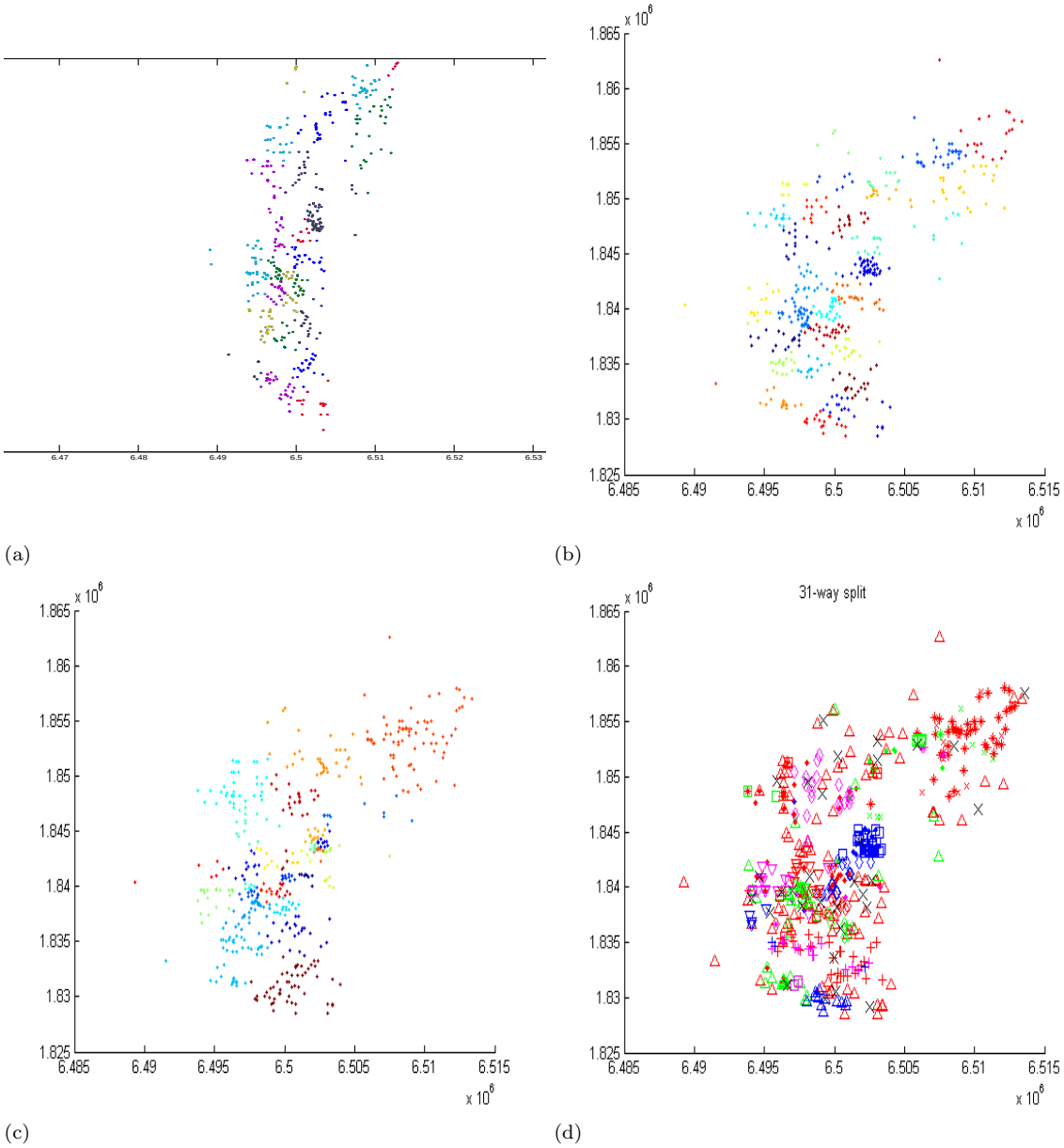
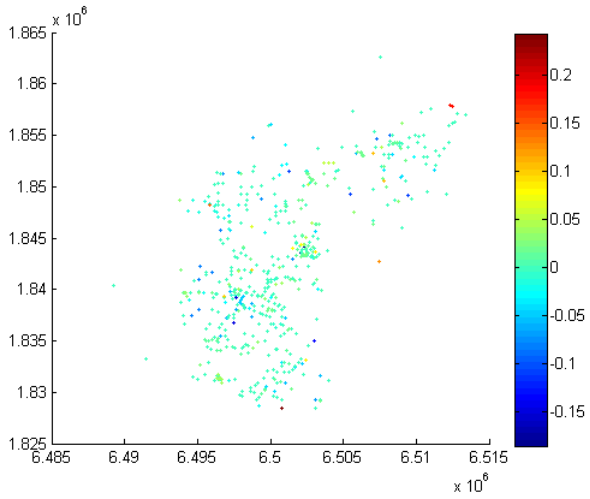
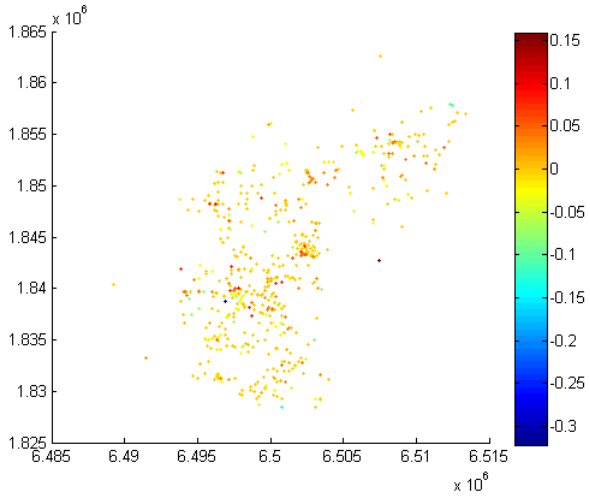


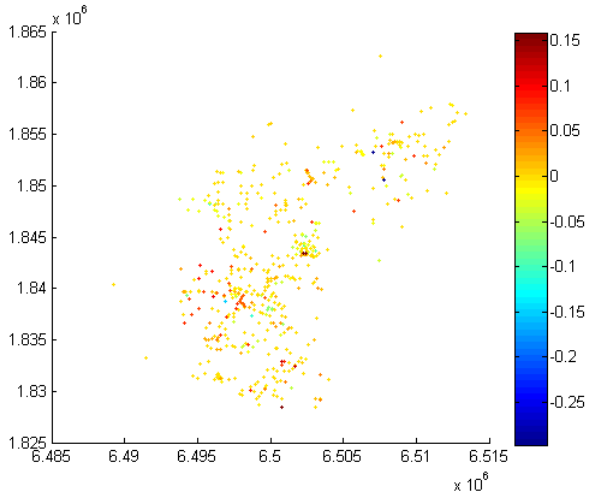
Figure 16: Clustering results. (a) k-means (geography), (b) k-means (mixed), (c) k-medoids (geography), (d) CONCOR (mixed)



(a)

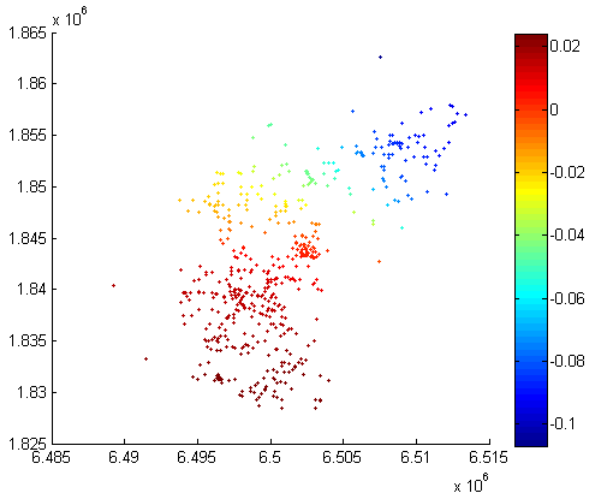


(b)

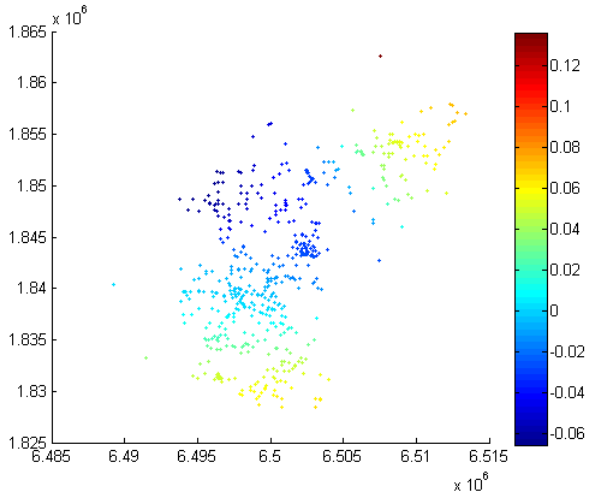


(c)

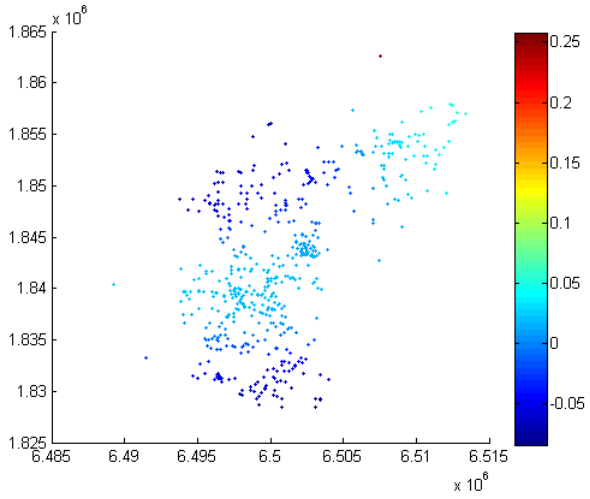
Figure 17: First 3 non-trivial eigenvector plots of (1.3) for social information only ($\alpha = 1$).



(a)

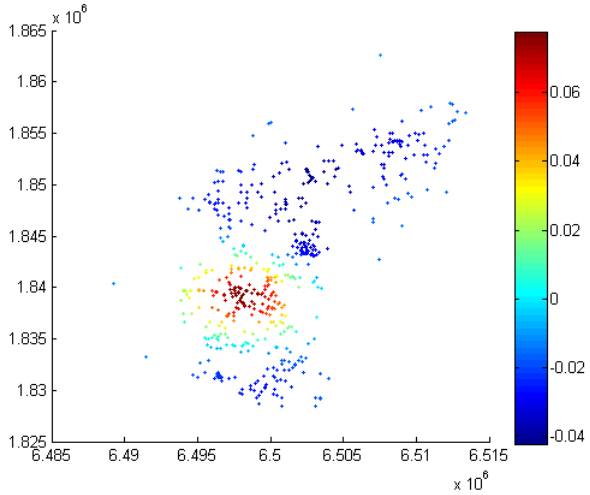


(b)

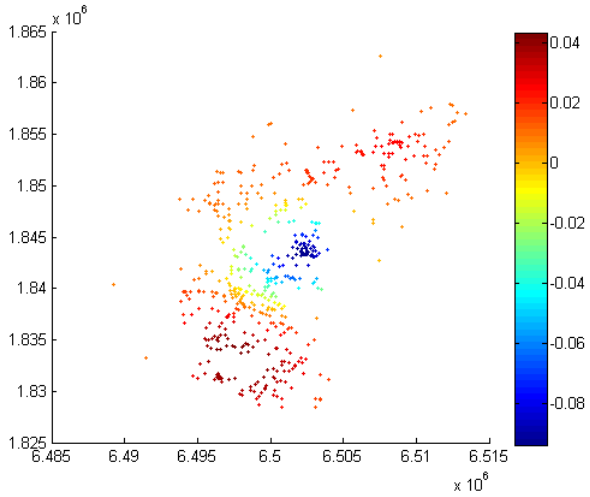


(c)

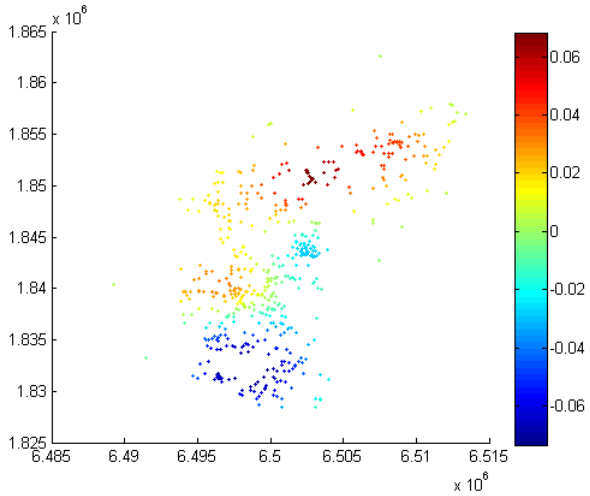
Figure 18: First 3 non-trivial eigenvector plots of (1.3) for geography information only ($\alpha = 0$).



(a)

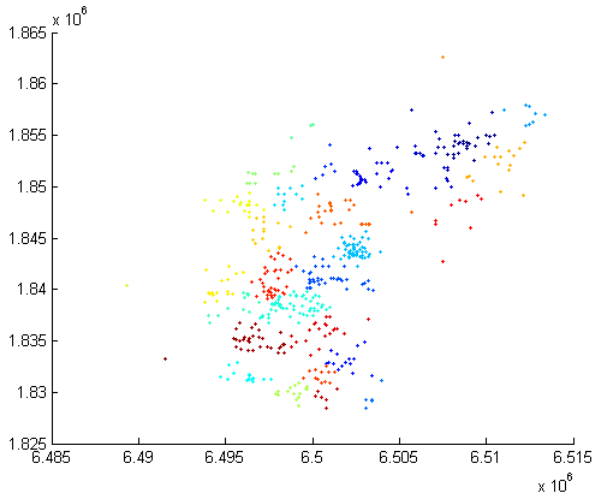


(b)

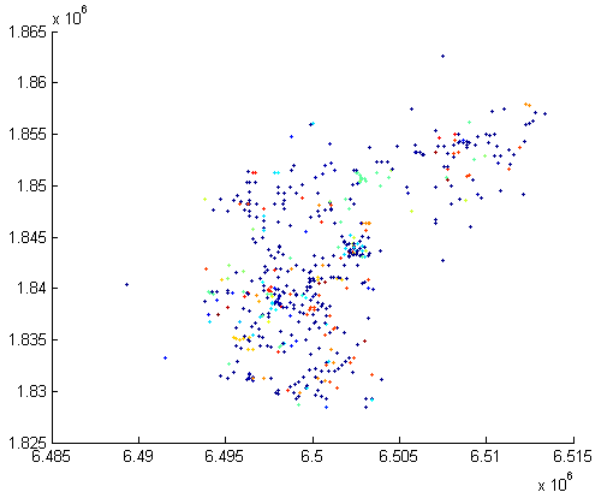


(c)

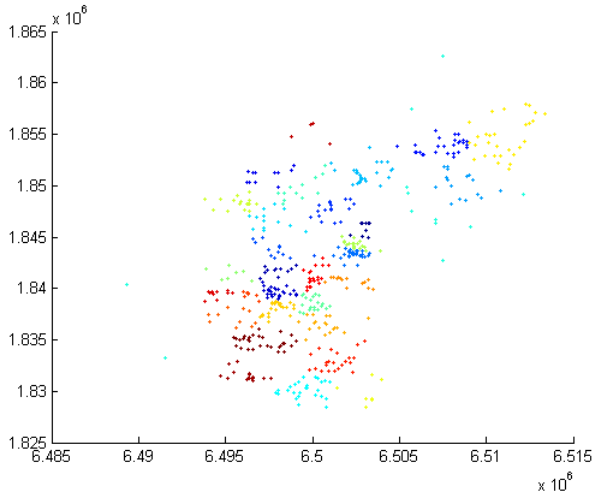
Figure 19: First 3 non-trivial eigenvector plots of (1.3) for mixed information ($\alpha = 0.5$).



(a)



(b)



(c)

Figure 20: Spectral clustering results. (a) geography only ($\alpha = 0$), (b) social only ($\alpha = 1$), (c) mixed ($\alpha = 0.5$).

Method	AMI	Purity
k-means (geography)	0.44	0.53
k-means (mixed)	0.44	0.57
k-medoids (geography)	0.44	0.51
CONCOR (mixed)	0.46	0.51

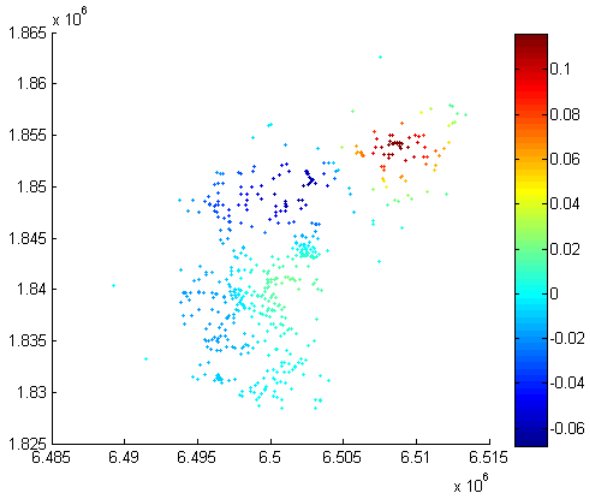
Table 1: Results from Hollenbeck data. K-means was performed using Matlab's *kmeans* function [8]. All values are taken from one run.

α	σ_1	σ_2	AMI	Purity
0.1	10^1	10^5	0.21	0.38
0.5	10^1	10^5	0.18	0.37
0.9	10^1	10^5	0.12	0.39
0.8	10^4	10^7	0.47	0.58
0.4	10^5	10^8	***0.60***	***0.70***
0.1	10^5	10^{10}	0.46	0.56
0.9	10^5	10^{10}	0.44	0.55

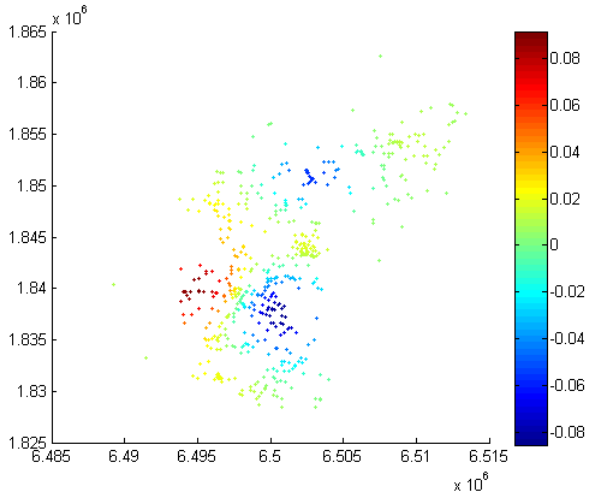
Table 2: Results of spectral clustering using affinity matrix defined (1.3) with various values for α, σ_1, and σ_2.

Table 1 and Figure 16 show the results of various methods using either geographical information only, social information only, or both. Figures 16a, 16c, and 20a show that using only geographical information, the only clustering possible is into set regions that do not represent either social structure or gang territories. Ideally, if individuals would not stray far from their own gang's center, using only geographical information leads to near perfect clustering, but Figure 15 shows that only a few gangs have their members located near their central location. For that reason, clustering by using geographical information only yields low AMI and purity values. Likewise, the clustering using only social information (see Figure 20b) also yields low AMI and purity values. By using both social and geographical information, and with the correct choice of σ_1 and σ_2 we obtained a purity value of 0.7 and an AMI value of 0.6 (see Table 2), which are significantly higher than the results from Table 1.

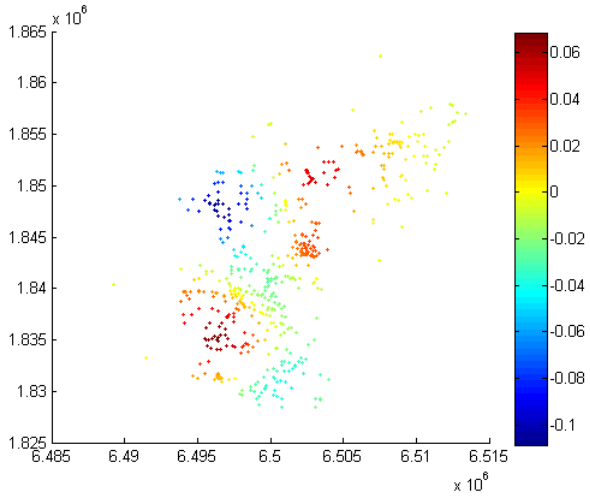
Figures 17, 18, and 20c show how combining both the social and geographical information can yield better results (all three were done using $\sigma_1 = 10^4$ and $\sigma_2 = 10^7$). Figures 17a, 17b, 17c show very little information about the structure of the group when using social information only, while Figures 18a, 18b, 18c show only the gang territory structure and does not identify members who are far away from the central location of the gang they are affiliated with. Only when we combine the information do we get structure that can potentially identify individual gangs. Figures 19a, 19b, 19c show the structure using only the first 3 eigenvectors. Since these results are interesting, we will spend more time analyzing these structures.



(a)



(b)



(c)

Figure 21: Eigenvectors 4 through 6 plots of (1.3) for mixed information with $\alpha = 0.5$.

Figures 19a and 19b clearly show a high concentration of identified gang members in specific locations. Figure 19a identifies an area (in dark red) which contains the gangs *Breed Street*, *Clarence Street*, and *Tiny Boys*. Figure 19b identifies *Big Hazard* (in dark blue). Furthermore, the next few eigenvectors in order of decreasing eigenvalues identify more gangs (see Figures 19c, 21a, 21b, 21c). In order to understand the significance of these identifications, we refer to [12] about the rivalry network of gangs in Hollenbeck. Since the gang territory map (Figure 1) and the rivalry network (Figure 22) are from 2003 while our data is from 2009, it is possible that some gangs no longer exists and that territory is different. We make the assumption that large dominant gangs from 2003 are still in the same position.

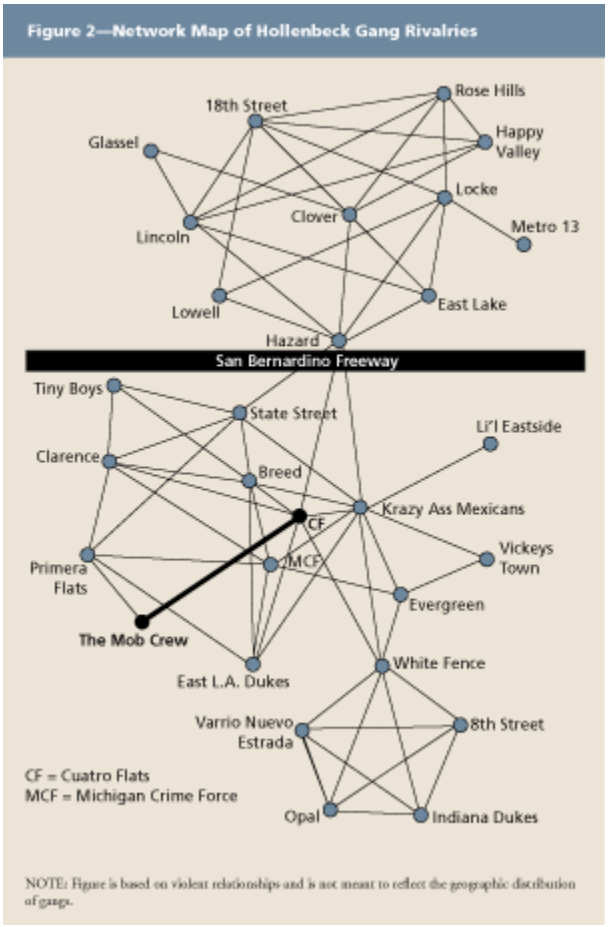


Figure 22: Rivalry network of gangs in Hollenbeck with gangs in their central location [12].

Figure 22 and [12] show that the main reason for a rivalry is proximity to another gang. The gang *Hazard* is positioned in the center of Hollenbeck and has many rivalries with other gangs. Also gangs like *Breed Street* and *Krazy Ass Mexicans* have many rivalries due to their central location in the southern half of Hollenbeck. Since our data consists only of non-criminal stops, we expect that gangs with a high number of rivalries will have more entries corresponding to a stop with someone within their own gang versus someone else. We also expect that these gang members will have central locations closer to each other. Figures 23a and 23b show how we are able to identify the gangs *Big Hazard* and *El Sereno* with only a few individuals incorrectly identified.

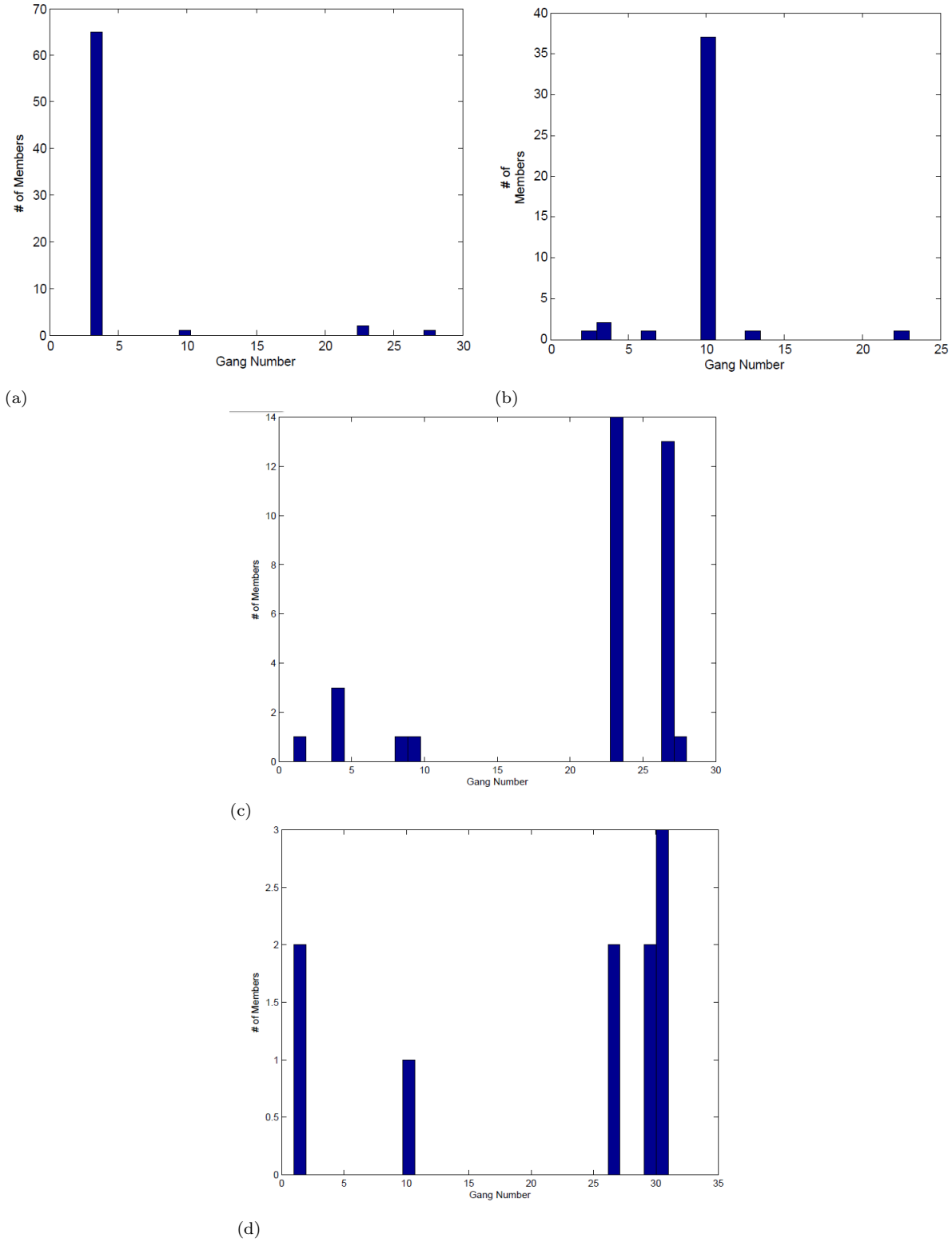


Figure 23: Results of thresholding each eigenvector to identify gangs. (a) identifies gang 3, (b) identifies gang 10, (c) identifies 2 gangs, (d) fails to identify. Eigenvectors were calculated using Algorithm 2.3 and with $\sigma_1 = 100$ and $\sigma_2 = 10^7$.

An interesting result comes from Figure 23c which shows one cluster containing two gangs, namely *Primera Flats* and *The Mob Crew*. Although Figure 22 shows that there is no rivalry between them, we believe that this shows cooperation between these two gangs, but without a current rivalry network, we cannot verify these results. Figure 23d shows that our algorithm performs poorly when detecting gangs of small size. We believe this problem occurs because forcing k-means to make 31 clusters leads to k-means clustering two gangs into one cluster (as in Figure 23c) and clustering outliers into another. This indicates that the social structure of Hollenbeck contains less than 31 clusters. Unfortunately, without more current information, we cannot draw any further conclusions as to exactly how many clusters Hollenbeck contains (see Section 6).

5 Conclusion

The ability of our algorithm to in detecting the gang structure from the provided data is surprising. While geographic data alone was ineffective and social connection data was too sparse, when used in conjunction, the information contained in both sets of data provided insights that neither set provided on its own. We believe that our method may be revealing social structures that transcend gangs, such as cooperations and rivalries. This remains to be verified in the field. Due to the appearance of higher order structures, requiring the algorithm to divide into 31 clusters may mask the dominant social structure. However, at present we have not devised a better way to extract this information from the eigenvectors, besides visually.

The gathering of non-criminal stop data is not unique to Hollenbeck. Our method can be applied in a variety of situations where geography and social contact are recorded and relevant to the social structure. The artificial data we have created will provide a means by which to understand the patterns our algorithm locates. At present, whether or not our artificial data is a reliable test of our algorithm is unclear.

6 Future Work

Much remains to be done in regards to this project and this particular method. It is clear that the eigenvectors are revealing latent social structures. We are still uncertain how best to distill these spectral details and to interpret this structure. Requiring the *a priori* input of the number of clusters may interfere with the detection of higher order clustering (multiple gangs). To address this issue, we believe a thresholding approach on the magnitudes of the eigenvector elements may be more enlightening instead of relying on k-means (see Algorithm 2.3 step 4). We plan to explore this approach in the future. We will also try to improve our results by incorporating information about the gang territory an individual was stopped in into the affinity matrix (1.3) by establishing an affiliation matrix F defined as:

$$F = \begin{array}{c} \\ v_1 \\ v_2 \\ \vdots \\ v_n \end{array} \begin{array}{c} \begin{array}{cccc} \textit{Gang}1 & \textit{Gang}2 & \dots & \textit{Gang}31 \quad \textit{Unclaimed} \end{array} \\ \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix} \end{array}, \tag{6.1}$$

where $F_{ij} = 1$ if v_i was stopped in territory of gang j (or in an unclaimed territory if $j = 32$) and adding it to (1.3) as such:

$$\hat{W}_{ij} = \alpha \exp(\frac{< f_i, f_j >^2}{\sigma_1}) + \beta \exp(-\frac{d(v_i, v_j)^2}{\sigma_2}) + \gamma \exp(\frac{< F_i, F_j >^2}{\sigma_3}), \tag{6.2}$$

where $< F_i, F_j >$ is the dot product between the i-th and j-th row of the affiliation matrix F (6.1) and $\alpha + \beta + \gamma = 1$.

The artificial data we have created needs to be thoroughly tested and compared to real data. Then by varying the structures contained in our artificial data and clustering it with our method (Algorithm 2.3),

we hope to better understand the social structures our algorithm exposes. More actual Hollenbeck data will provide another means of validation of our algorithm.

7 Acknowledgements

We would like to thank Professor Yves Van Gennip for advising us throughout the project. We are also indebted to Professor Blake Hunter for pushing us towards our algorithm. We are grateful to Professors Allon Percus and Jeffrey Brantingham for their helpful advice. We also thank the Los Angeles Police Department, UCLA anthropology department, and UCI criminology department for providing us the data. Finally, we would like to express our appreciation to Professor Andrea Bertozzi and the UCLA REU program for providing this opportunity, and, perhaps more importantly, the funding.

References

[1] W. Aiello, F. Chung, and L. Lu. A random graph model for power law graphs. *Experimental Mathematics*, 10(1):53–66, 2001.

[2] A.L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509, 1999.

[3] A. Clauset, C. Rohilla Shalizi, and M. E. J. Newman. Power-law distributions in empirical data. *ArXiv e-prints*, June 2007.

[4] E.N. Gilbert. Random graphs. *The Annals of Mathematical Statistics*, pages 1141–1144, 1959.

[5] A. Lancichinetti and F. Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4):046110, 2008.

[6] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.

[7] N. Masuda, H. Miwa, and N. Konno. Geographical threshold graphs with small-world and scale-free properties. *Physical Review E*, 71(3):036108, 2005.

[8] Matlab. *MATLAB:2010 version 7.10.0 (R2010a)*. The MathWorks Inc., Natick, Massachusetts, 2010.

[9] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14: Proceeding of the 2001 Conference*, pages 849–856, 2001.

[10] Hae-Sang Park and Chi-Hyuck Jun. A simple and fast algorithm for k-medoids clustering. *Expert Systems with Applications*, 36(2, Part 2):3336 – 3341, 2009.

[11] S.M. Radil, C. Flint, and G.E. Tita. Spatializing social networks: Using social network analysis to investigate geographies of gang rivalry, territoriality, and violence in Los Angeles. *Annals of the Association of American Geographers*, 100(2):307–326, 2010.

[12] G. Tita, J. Riley, G. Ridgeway, and C. Grammich. Unruly turf: The role of interagency collaborations in reducing gun violence. *Rand Review*, 2003.

[13] G.E. Tita and R.T. Greenbaum. Crime, neighborhoods, and units of analysis: putting space in its place. *Putting Crime in its Place*, pages 145–170, 2009.

[14] N.X. Vinh and J. Epps. A novel approach for automatic number of clusters detection in microarray data based on consensus clustering. In *BioInformatics and BioEngineering, 2009. BIBE'09. Ninth IEEE International Conference on*, pages 84–91. IEEE.

[15] Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17:395–416, 2007.

[16] S. Wasserman. *Social network analysis: Methods and applications*. Cambridge university press, 1994.