Journal of Social Structure

JoSS Article: Volume 15

Choosing a Clustering: An A Posteriori Method for Social Networks

Samuel D. Pimentel
Department of Statistics, Wharton School of the University of Pennsylvania, spi@wharton.upenn.edu

Abstract
Selecting an appropriate method of clustering for network data a priori can be a frustrating and confusing process. To address the problem we build on an a posteriori approach developed by Grimmer and King (2011) that compares hundreds of possible clustering methods at once through concise and intuitive visualization. We adapt this general method to the context of social networks, extend it with additional visualization features designed to enhance interpretability, and describe its principled use, outlining steps for selecting a class of methods to compare, interpreting visual output, and making a final selection. The interactive method, implemented in R, is demonstrated using Zachary's karate club, a canonical dataset from the network literature.

Keywords
clustering, networks, social networks, visualization, interactive, community detection, cluster analysis, comparing clusterings

Introduction

In his comment on Handcock, Raftery and Tantrum's 2007 paper, sociologist David Krackhardt characterized cluster analysis of social networks as follows:

> Cleanly identifying clusters of actors in a social system on the basis of their social ties is an age old pursuit of generations … UCINET, the most commonly used package for analysis of network data, has 20 distinct methods for finding clusters or groups, each with a plethora of suboptions and choices of parameter which, depending on the data, may yield wildly differing results. This dizzying array of 'solutions' begs the central question: given the observed data, what is the right number of clusters and what is their composition?[1]

Krackhardt's comment identifies the key problem that network researchers face in choosing an appropriate clustering method for a particular data set: determining what type of structure each method will pick out in a particular data set. The fruitful literature of cluster analysis provides plenty of methods from which to choose but describes and distinguishes them primarily by their under-the-hood operation rather than by their qualitative results. One might hope that understanding the fine details of a clustering algorithm would lead to an understanding of what kind of cluster structures it will recognize. However, clustering algorithms are often based on heuristics and justified by practical performance considerations rather than by formally proven guarantees due to the difficulty of clustering problems (Ovelgönne and Geyer-Schulz, 2012). While an algorithm of this type may have been shown to do well in certain situations, the full range of cases where it performs best and worst may not be known definitively even by its developers. Furthermore, constraints on time and energy make it difficult even to develop a practical working knowledge of each of the many algorithms and its particular parameters and idiosyncrasies.

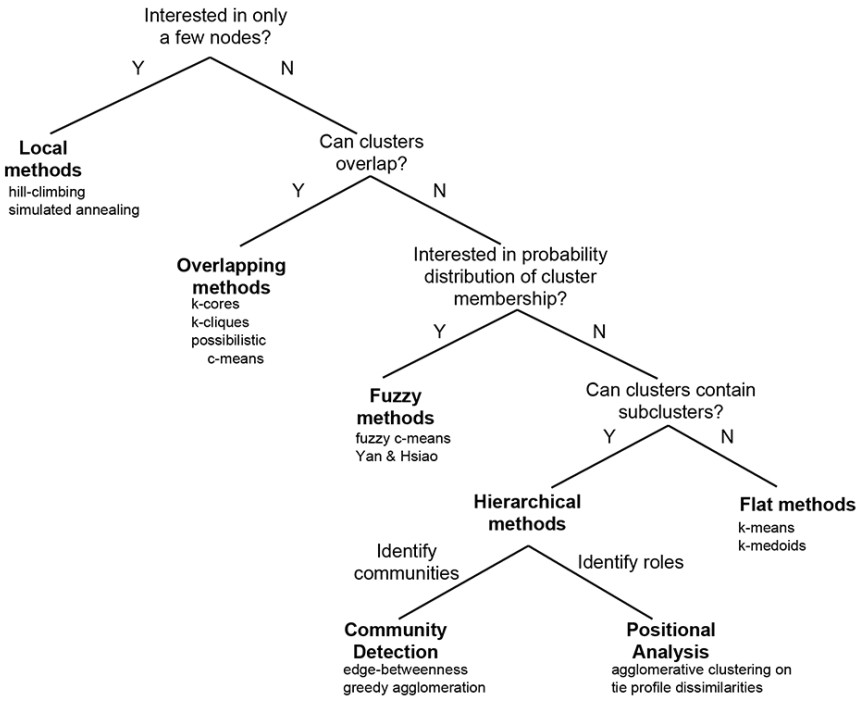

Figure 1: Decision tree showing appropriate a priori criteria in choosing a clustering. Bold-face labels indicate families of clustering algorithm appropriate for the criteria on that part of the tree, and examples of methods in the family are given below in small print. See Schaeffer (2007), Krishnapuram and Keller (1993), Bezdek, Ehrlich, and Full (1984), and Kaufman and Rousseeuw (1987) for more detail on many of the specific examples of methods.

Reviews of clustering from the network literature do provide a helpful general taxonomy for clustering algorithms, which the decision tree shown in Figure 1 attempts to summarize. The series of questions represented highlights key attributes of clustering results that are well-known a priori and allows a researcher who plans to use clustering to eliminate wide class of inappropriate methods immediately. First, a researcher should distinguish between local clustering methods that classify only a few nodes or actors of interest (usually within a graph so large that clustering every point is computationally infeasible) from global clusterings that decompose an entire network into subsections (Schaeffer, 2007). Next, the researcher should define his or her concept of a cluster more clearly by asking several further questions (SAS Institute, 1990): Can a node belong to several clusters or only one? Should cluster membership be viewed as a stochastic process or as a fixed labeling? Can clusters have subclusters or not? Finally, the researcher should make some preliminary decisions about the type of similarity or homogeneity he or she wishes to see in members of an individual cluster. Two (among many) possible clustering rationales are mentioned in the diagram: positional analysis, in which clusters correspond to distinct social positions and co-membership in a cluster indicates a form of structural equivalence (Scott, 2000) or at least a similar tie pattern (Gest, Moody and Rulison, 2007), and community detection, in which clusters represent subgroups of actors who tend to have more ties to one another than to other actors in the network (Fortunato, 2010).

Note that the diagram in Figure 1 could reasonably be expanded at most leaves. For example, input network size and available computational resources could be considered since many clustering algorithms become computationally intractable with sufficiently large networks (Schaeffer, 2007). Some methods are also specific to undirected graphs (Newman, 2006), while others are more appropriate for graphs with valued and/or directed ties, or for networks with attribute data for their actors (Schaeffer, 2007). While not represented on the decision tree, such factors are known a priori and should be allowed to guide the initial choice of clustering method.

However, paring down the range of possible options by such a decision process still leaves a researcher with a large pool of potential clustering algorithms. For example, a researcher with data for a small network of undirected ties who wishes to identify cohesive communities must still decide whether to use fast-greedy agglomeration (Clauset, Newman and Moore, 2004), walktrap analysis (Pons and Latapy, 2005), edge-betweenness (Newman and Girvan, 2004), or one of several others. A user must also select parameters for some of these methods. For most uses of hierarchical clusterings, the researcher must also select a level at which to cut the clustering dendrogram in order to obtain a particular set of labels.

Thus in this situation the researcher still faces a choice among dozens of possible clusterings. In addition, the remaining clustering methods likely share very similar objective functions. Some may indeed provide clusterings much more interesting or preferable than others, but it is extremely difficult to identify these a priori from the definitions of the individual algorithms. Krackhardt's dilemma remains.

A Posteriori Clustering Selection

The difficulty of selecting an optimal clustering method from among many extends beyond networks to a more general class of data. In their 2011 paper Justin Grimmer and Gary King introduce an idea that will be referred to as posteriori clustering selection and present an efficient tool applying it to unstructured text data. A priori clustering selection works on the assumption that the best way to choose an algorithm is according to a decision tree as in Figure 1, with each algorithm corresponding to a particular subclass of problems and/or data types. In contrast a posteriori cluster selection works in theory by "presenting an extremely long list of clusterings (ideally, all of them)" as performed on the data set and "letting the researcher choose the best one for his or her substantive purposes" (Grimmer and King, 2011) Here the choice among a wide range of methods remains, but instead of having to choose between methods based on

algorithm definitions, the researcher is able to determine the best method by examining the resulting clusters in the data according to his or her particular qualities of interest.

In the a posteriori context, researchers still face the challenge of quickly and intelligently navigating the massive collection of completed clusterings in order to find those of interest. Grimmer and King address this issue by presenting a software package that runs "all existing cluster analysis methods" on a dataset and produces a visualization in which each of the possible clusterings is shown as a point in a metric space. The metric space also includes "millions of other clusterings" created by taking weighted combinations of existing methods, and an interactive user interface allows researchers to explore the space and see how cluster identification changes from region to region. By incorporating all possible methods into the space and facilitating efficient perusal of the methods, the package allows researchers to take into account the full range of possibilities for clustering; by leaving the final choice of clustering to the researcher, who manually selects it from the visualization, the method avoids the problems and biases associated with fully-automated computer clustering. Grimmer and King call their approach "general-purpose computer-assisted clustering."

General-purpose computer-assisted clustering seems ideal not only for Grimmer and King's datasets of unstructured text but for most fields that rely on clustering algorithms, especially social network analysis. However, Grimmer and King's work leaves largely open the question of how a researcher is to select a clustering from the visualization space. This is understandable given the great diversity of uses of clustering and possible research goals in conducting it among their audience, but there remains a need for meaningful tools and principles to help researchers make this a posteriori decision given the information provided by the visualization.

We present a general-purpose computer-assisted clustering approach closely modeled on Grimmer and King's work and adapt it to offer a more principled means of a posteriori exploration and clustering selection in the context of social network analysis. The method, implemented in R, performs a variety of hierarchical clusterings under different parameters and allows researchers to incorporate a cluster quality metric through a color-coding of points in the clustering space. In addition, the tool allows researchers to specify a hand-coded clustering for data in which they know or hypothesize a particular clustering structure and will display this clustering as a specially marked point in the cluster-space visualization. These features, along with a solid understanding of the types of structures that may appear in the visualization and of how to interpret them correctly, allow network researchers to efficiently explore a wide variety of possible clusterings and make swift and sound judgments about which particular clustering method should be used for a data set. The following sections describe the steps of the generalized clustering algorithm in greater detail, explain the resulting visualization and its proper interpretation, and demonstrate the method on a dataset from the network literature.

Methods

Grimmer and King's general-purpose clustering framework involves three main steps: first, running many different clustering algorithms with varying parameters on an input data set; second, computing a distance matrix among all the computed clusterings using the Variation of Information (Meila, 2007); and third, plotting the clusterings as points in two dimensions using a Sammon scaling of the distance matrix (Sammon, 1969). These same general steps are applied to social network datasets as detailed below.

Clustering Algorithms

As Grimmer and King point out, any clustering method that creates a valid partition (by assigning each data point to exactly one cluster) can be included in a general-purpose clustering framework. For the purposes of this investigation, however, we focus on methods appropriate for undirected networks with binary ties, including methods oriented towards community detection and positional analysis.

Of the many clustering methods for community detection, the software tool currently incorporates three with easily accessible implementations in the igraph package. The first, due to Clauset, Newman and Moore, is fast-greedy agglomeration (2004). An agglomerative algorithm, it identifies communities by iteratively merging individual actors or clusters into larger clusters, and it decides which groups or individuals to merge by maximizing the increase in modularity score over all possible such merges. The modularity score Q for a clustering represents the degree to which edges within a cluster occur more often than would be expected in a random graph (Newman and Girvan, 2004) While different formulations of Q have been proposed, Clauset, Newman and Moore give the following intuitive formula:

$$Q = \sum_{i=1}^{k} (e_{ii} - a_i^2)$$

where k is the number of clusters, e_{ij} is the fraction of the total number of edges that connect vertices in clusters i and j, and a_i is the fraction of edge endpoints attached to a vertex in cluster i. While the algorithm works greedily and is not guaranteed to find the maximum modularity solution for the data, this is not a reasonable goal anyway; one would need to examine the modularity of each possible clustering, and even for a small network of 20 nodes over 10^{13} clusterings exist.

Another commonly-used method for community detection in undirected binary network settings is Newman's and Girvan's divisive method based on the edge-betweenness measure (2004). Edge betweenness is a count of the number of node pairs for which the edge in question belongs to their shortest connecting path. The algorithm removes the edge or edges with highest betweenness at each step to partition the graph into well-connected communities.

Finally, walktrap clustering performs a series of random walks over a network and generate a dissimilarity matrix based on the likelihood of reaching one node from another in a given random walk (Pons and Latapy, 2005). Then it produces an agglomerative hierarchical clustering by merging the pairs of individuals or clusters that are most similar according to Ward's distance (the mean of the squared dissimilarities between each vertex and its community mean) (Ward, 1963).

The software tool also incorporates another family of agglomerative clustering methods. These methods may be better suited for positional analysis, since they are based on the tie profiles of the network actors. Methods in this group are characterized by two components: a measure of dissimilarity for the nodes of the network based on their respective tie patterns, and an agglomeration rule used to decide how to merge individuals and smaller clusters to form a hierarchical clustering. In fact most clustering methods, including some of the community detection methods detailed above, can be described in these terms. To generate a comprehensive collection of clusterings in this family, one must first identify a list of candidate dissimilarity measures and a list of candidate agglomerative rules. Each unique pairing of one dissimilarity measure with one agglomeration rule then defines a unique clustering method.

The dissimilarity list used by the software tool includes Euclidean distance, Manhattan distance, supremum norm, (transformed) correlation, Jaccard dissimilarity, (transformed) cosine similarity, Hamming distance, and (transformed) gamma correlation. The regular correlation, the gamma correlation, and the cosine similarity are all similarities rather than dissimilarities so we transform them by subtracting values of each from 1. The Euclidean distance, Manhattan distance, and supremum norm are versions of the L^p norm for values of $p = 2$, $p = 1$, and p taken to infinity respectively. The cosine similarity computes the acute angle having a cosine equal to the ratio of the two vectors' dot product divided by the product of their norms. The Jaccard dissimilarity, Hamming distance, and gamma correlation are all used to compare vectors of presence-absence data coordinate-wise. The Hamming distance gives the number of discordant pairings, the Jaccard dissimilarity gives the ratio of discordant pairings to the total number of pairings with at least one nonzero entry, and the gamma correlation gives the signed difference in the number of concordant pairings and the number of discordant pairings scaled by the total number of pairings.

The agglomeration rules used were Ward's distance, single linkage, complete linkage, average linkage, McQuitty's distance, median distance, and centroid distance. Each of these is either a measure of distance between clusters or a measure of quality for a cluster, and agglomeration proceeds by merging the clusters with the shortest distance between them or searching over all possible two-cluster merges to find the one that maximizes the cluster quality measure. Single linkage defines the distance between two clusters as the distance between their closest points, while complete linkage defines inter-cluster distance as the largest such distance. Average linkage is the mean distance between points in the two clusters. The McQuitty distance between two clusterings is defined recursively, where at each step of the algorithm the distance of the most recently merged clusters A and B from any other cluster C is the average between the distances from A to C and from B to C (McQuitty, 1966). Ward distance is the difference between the sum of squared differences from the mean point in a potential merged cluster and the sum of the squared differences from the mean in both of the candidate clusters (Ward, 1963). The centroid distance between two clusters is simply the distance between their mean or "center" points. Finally, median linkage is defined recursively so that at each level the distance of the most recently formed cluster to any other cluster is the average of the median linkages of its two component clusters to that cluster.

Each of the clustering methods detailed above is hierarchical and so produces a series of nested clusterings of different numbers of clusters. To take the various nested clusterings into consideration, the method slices the dendrogram of cluster hierarchy at each level from two clusters up to a large number and represents each of the resulting clusterings separately in the final plot.

Distance Matrix for Clusterings

To embed a large number of clusterings in a geometric space for visual analysis, it is necessary to compute distances or dissimilarities between the clusterings. The Variation of Information (VI) distance metric for comparing clusterings based on the proportion of items in a data set that are sorted into the same group by two different clusterings (Meila, 2007). It is based on the concept of entropy in information theory, which measures the uncertainty of a random variable. In general terms, the VI distance between clusterings A and B is the sum of the uncertainty in assigning a random point to a cluster in A given we know its cluster in B and the uncertainty of assigning the same point to a cluster in B given we know its cluster in A. Equivalent clusterings fully determine each other and obtain a zero value, while highly divergent clusterings give little information about each other and give high values.

Grimmer and King demonstrate that the VI metric is preferable to other cluster dissimilarity indices for general-purpose computer-assisted clustering. For example, VI allows clusterings with different numbers of groups to be compared, and it is also invariant to network size, unlike the commonly used adjusted Rand Index. Many of VI's nice properties have to do with the fact that it is a metric, meaning it obeys certain rules (such as non-negativity and the triangle inequality) that are shared by Euclidean distance. This makes it a natural choice when visualizing cluster dissimilarities as geometric distances.

Scaling and Display

Sammon scaling is a special case of multidimensional scaling (MDS), a dimension-reduction technique used to embed a collection of dissimilarities between (usually high-dimensional) data into a low-dimensional space with as much fidelity as possible (Buja et al., 2008). Its distinction in comparison to standard MDS is its weighting scheme for the dissimilarities: where MDS weights all the dissimilarities equally, Sammon scaling weights the dissimilarities inverse to their size, emphasizing the accurate preservation of small dissimilarities at the expense of larger ones. This makes Sammon scaling a better fit for general-purpose computer-assisted clustering where we are more likely to be interested in comparing small distances (for example, to distinguish which of three similar clustering algorithms comes closest to a particular hand-coded clustering of interest) than large distances (which simply indicate that a pair of clusterings are not particularly similar).

Once the distances between the clusterings have been computed and embedded in a Sammon-scaled plot, an interactive interface allows a user to point and click on clusterings of interest, displaying for each such point a descriptive label that includes the name of the clustering algorithm used and any important distinguishing parameter values. The selected clusterings of interest can be visualized individually in network plots to allow quick comparison and insight.

A Posteriori Selection Criteria

Having created a visualization plot representing the space of possible clusterings for a data set, a researcher usually wishes to identify a particular clustering method to adopt in his or her investigations. Selecting a particular clustering is a very application-specific problem, and there is not an objective "best" solution unless the user is able to define his or her needs precisely and quantitatively. However, several guiding principles and tools may be used to better recognize structure from a visualization plot and draw useful conclusions about the nature of the clustering landscape.

First, a researcher can benefit from recognizing particular structures in the pattern of visualized clusterings. First, points representing clusterings may align themselves in long arcs or "tails" in the space of visualization. Such tails usually represent some kind of axis of variation in the clustering landscape. For example, a tail may appear when a particular parameter is varied in an otherwise identical call to a clustering method. In such a case, the tail can be interpreted as a crude description of the effect of this parameter on differentiating methods. By examining the length and location of these tails, a researcher gains an intuitive sense for the effect of varying a particular clustering parameter and learns which criteria are important.

Another common type of structure visible in visualizations is a high-point-density area or "core." These structures tend to correspond to groups of clusterings that share a general quality of pairwise similarity. By examining a single clustering from a core, a researcher ideally gains a sense for the general qualities of the entire core. This provides an especially useful simplification of the clustering space by binning large number of methods and parameter combinations into a single qualitative group and informing the

researcher that the choice among fellow members is relatively unimportant with respect to his or her data. For many applications, the ideal visualization would be one or more very dense cores surrounded by sparsely-populated regions of degenerate or otherwise peripheral clusterings. Such an image would reduce the choice between hundreds of possible clusterings to one between a few distinct clustering families.

While other visual structures of interest may emerge from applications of the algorithm to various data sets, many will involve variants or combinations of these two types. For example, an elongated core shape may represent an axis of minor variation within an otherwise very similar group of clusterings.

These structures are described in very qualitative terms here, in part because choosing quantitative measures for similarity structures is itself a clustering problem and in our case leads to "infinite regress" (Grimmer and King, 2011). In addition, dimension reduction by scaling means that distances in the 2-dimensional plot (especially large ones) do not correspond exactly with true distances. This demands a measure of skepticism in interpreting plots; for example, researchers should not assume immediately that a tail in one part of the diagram represents a global gradient that applies similarly to points in distant parts of the plot. In addition, researchers working with cores would do well to select several clusterings from each core and compare them as a check for true pairwise similarities within the core and as a view into the nature and possible importance of variations that do exist within the core. Degenerate cases of core-like shapes masking substantial, important differences between clusterings should be identifiable by this approach.

Computing a Shepard plot of the scaled points allows a researcher to visualize the overall fit of the scaling as a representation of the actual distances by taking each pairwise scaled distance between points in the plot and plotting it against the corresponding actual distance (Shepard, 1962). Substantial divergence from the $x = y$ line indicates poor representation of true distances – points above the line are represented as more distant than they are in reality, and points below it are represented as being closer than they are in reality. A degenerate Shepard plot is evidence that the true structure of the clustering landscape is not well-represented in two dimensions by the current clustering algorithm. In this case the researcher may find better insights by examining a smaller set of clusterings and/or by using a different scaling algorithm.

As an aid in examining and interpreting emergent structures in the scatter plot, a researcher may also choose to add metadata to the plotting symbols. An easy adaptation is to use different plotting symbol for different families of clustering method. In addition, one can add a color code to each point according to a cluster quality metric of interest assessed on the clustering at that point. Superimposing colors in this way can add meaningful shape to the clustering landscape, making it easy to quickly identify the regions of the plot with the most desirable methods and narrow in on a final decision. However, cluster quality metrics, like clustering methods, exist in great numbers and particular cluster quality metrics will often be biased towards particular methods (since some methods optimize quality metrics directly). A cautious researcher will therefore use color-coding as a heuristic rather than as a trusted measure of quality and may choose to look at the colored plots for several different quality metrics rather than relying on one alone.

One potentially useful metric is modularity, which is defined above in the definition of the fast-greedy algorithm. Modularity is intuitive as a measure of community structure for graphs and is easy to calculate. In community detection settings a modularity-coded plot can quickly direct researchers to cores and other plot regions where methods perform well in identifying modular structures.

Evaluating the quality of positional clusterings requires a different quality index, since we now wish to group actors based on the similarity of their tie patterns rather than their connectedness to one another. Milligan and Cooper compared a variety of different indices appropriate for this type of clustering (1985). One index that performed highly in their study was a version of Goodman and Kruskal's lambda, which we

will refer to as G2 following Gordon (Goodman and Kruskal, 1954; Gordon, 1999). G2 calculates all possible pairings of individual within-cluster and between-cluster dissimilarities, counts the number of "concordant" such pairings (where the within-cluster dissimilarity is less than the between-cluster dissimilarity) and the number of "discordant" ones (where the between-cluster distance is less than the within-cluster distance). The final score is the difference between the numbers of concordant and discordant pairings, scaled by the total number of pairings. Among its other strengths, G2 can compare clusterings with different numbers of clusters meaningfully. However, G2 requires a dissimilarity matrix to be defined for the data in advance, so when computing it a researcher must choose which dissimilarity to supply. This limits the usefulness of any particular set of G2 values for a visualization plot, since it will be tied to a particular dissimilarity and will tend to favor clusterings based on that similarity. Nevertheless, the method should effectively compare different agglomeration approaches for a given dissimilarity and a researcher can compare G2 results for several dissimilarities to minimize overall interpretive bias. In addition, G2 has a good R implementation in the package fpc (Hennig, 2013).

In other cases, a researcher may be interested primarily in comparing algorithmic clusterings to a particular "ground truth" or otherwise fixed clustering, perhaps an ethnographic hypothesis. Of course the researcher could include the ground truth clustering in the plot, but in addition he or she could color-code individual points with their respective variation-of-information distances to the ground truth clustering. Ideally the configuration of points in the diagram should already provide a good representation of which clustering methods are closest, but the additional use of color codes would preserve information about distance from this particular clustering that would otherwise be lost in the process of multidimensional scaling.

In summary, a posteriori cluster selection proceeds on application-specific terms. However, in most cases the process will involve a search for structures in the plot and a cursory analysis of the nature of these structures, hopefully resulting in a qualitative meta-clustering of some sort. This will be followed by a principled judgment between possible families or parameter combinations, often based on a desirability criterion.

Results

The tool is demonstrated using Wayne Zachary's karate club data (Zachary, 1977). This data set has a "ground truth" clustering based on an actual split in the club that occurred after the data was collected, and we first demonstrate how general-purpose computer-assisted clustering identifies community detection algorithms that choose this clustering or one very similar to it. Then we use the method in a more exploratory context to investigate positional structure in the same data.

Zachary's Karate Club—Community Detection

Each of the 34 actors in this represents a member of a karate club, and ties between actors (all mutual) indicate that the two actors interact in some setting other than official karate club meetings (Zachary, 2007). The data includes labels indicating which of the two factions each club member chose when the club split into two separate communities. In the original paper Zachary used a clustering algorithm to predict the split with a high degree of accuracy (only one actor misclassified). Using general-purpose computer-assisted clustering on this dataset, we sought to quickly find a method to identify the two factions as closely as possible.

Following the decision tree given in the introduction, we determined a priori to investigate global clustering methods that assign discrete (rather than overlapping or probabilistic) labels. Since communities within a karate club might reasonably have interesting subgroups, we chose to use hierarchical methods. Finally, following Zachary, we chose to focus on community detection rather than role structure identification in our approach to this data. As such, only three algorithms oriented specifically towards community detection (edge-betweenness, fast-greedy agglomeration, and walktrap clustering) were included in the array of clusterings performed. In order to examine substructures at several levels, the clustering dendrograms for each method were sliced at several points each to produce different sets of cluster labels. The final clustering space contained 33 unique algorithmic clusterings as well as an additional point representing the clustering implied by the split between the groups. Figure 2 shows a plot of the clusterings produced for Zachary's data, embedded in a 2-dimensional space by Sammon scaling. Different symbols indicate different families of clustering method (here triangles represent edge-betweenness clusterings, X's represent fast-greedy agglomerations, crosses represent walktrap clusterings, and the circle represents the clustering of the actual split).

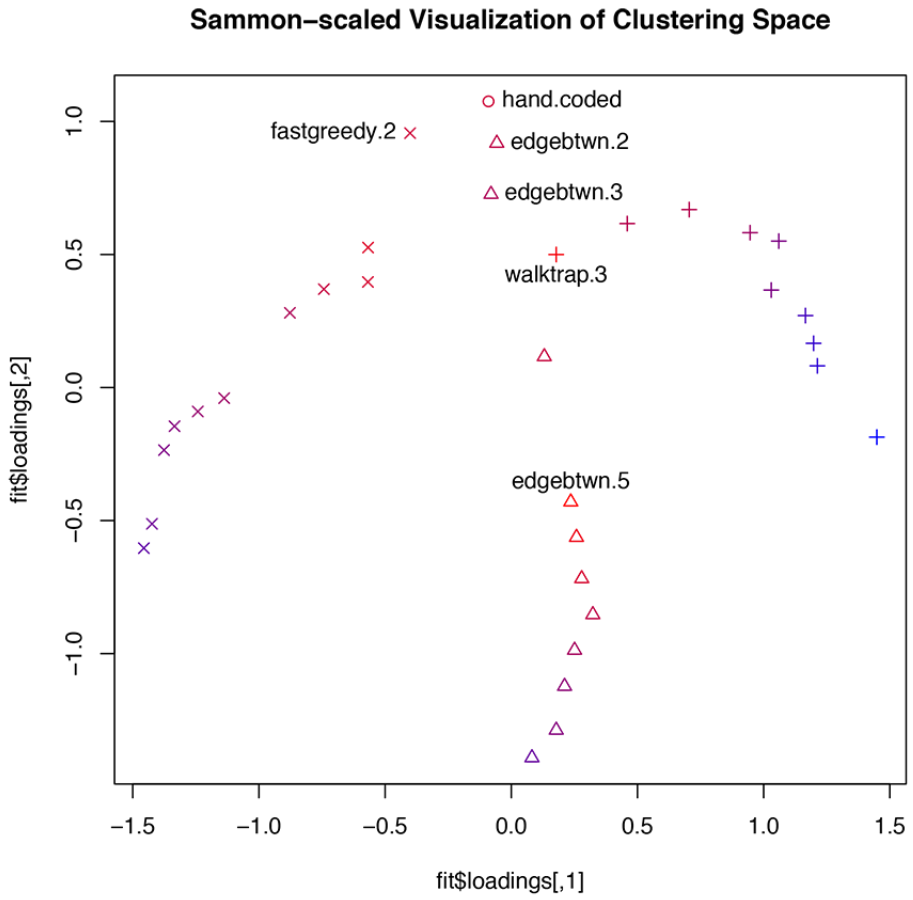

Figure 2: Visualization of possible clusterings for Zachary's network data. The color of a clustering point indicates its modularity score (red = high, blue = low). Triangles represent edge-betweenness clusterings, crosses represent walktrap clusterings, X's represent fast-greedy clusterings, and the circle represents the actual division in the karate club as reported by Zachary. Labels show the algorithm name and the slice point of the dendrogram. When the tool is used in R, a researcher can view such a label for any point in the plot simply by clicking on it.

Because the goal in this problem is to detect cohesive communities, we have used a color-coding that corresponds to a scaled version of the modularity score *Q*. Brighter-red points indicating clusterings of higher modularity (relative among the clusterings in the plot) and blue points indicating clusterings with very poor modularity.

Several interesting features are visible in this plot. First, three distinct tails are apparent, each one corresponding to an individual clustering method. The points in each tail vary according to the level at which the clustering dendrogram has been sliced. In addition, one sees from the points' color codes that the modularity tends to decrease sharply towards the more extreme parts of the tails. This suggests that the network lacks strong group structure at the level of more than seven or eight groups. Clearly the walktrap algorithm performs relatively poorly compared to the other methods for more than two or three clusters, since it has much lower modularity scores (shown by much bluer points) for clusterings with the same number of clusters as given by other algorithms. In any case, the interesting portion of the plot is the top-center area where the tails converge. Here we find the "true" clustering into the two splitting groups (the "hand.coded" label on the plot) and several other clusterings from different algorithms that seem to approximate it closely. This area of the plot also appears to be relatively high in modularity, which confirms that Zachary's original grouping is a good description of group structure in the data. The combination of including this known clustering and overlaying modularity scores organizes the visual space of the plot and helps researchers focus quickly on the most interesting or important clusterings.

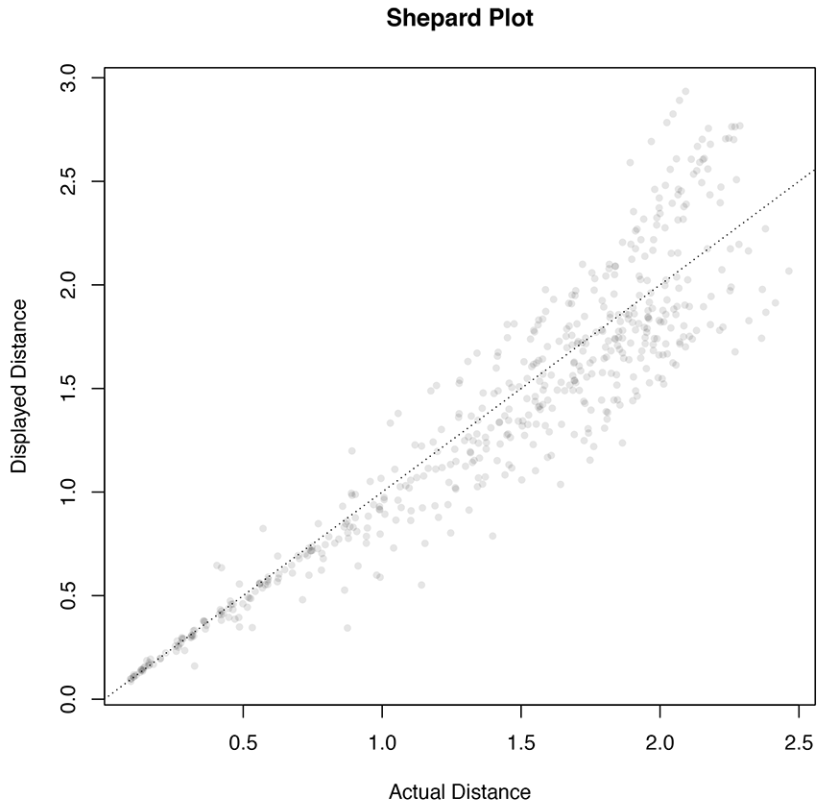

Figure 3: Shepard plot for scaling in visualization of community detection methods for Zachary's data. Each point corresponds to a unique pair chosen from the 34 clusterings in the visualization, and its position shows how closely the displayed distance between the corresponding clusterings adheres to their actual variation-of-information distance. Points lying on the line $y = x$ represent distances that are accurately depicted in the visualization, and the further a point is from this line the more that particular distance is distorted in the display.

Note that the Shepard plot for this layout (Figure 3) indicates that the visualization is a very good representation of the data. The points in the low-distance region adhere almost exactly to the $y = x$ line. In the larger-distance region, many of the points are above the line, suggesting that large distances on the plot are somewhat greater than they are in reality. But since this is true only of distances that are already large, it does not undermine our interpretation of the plot.

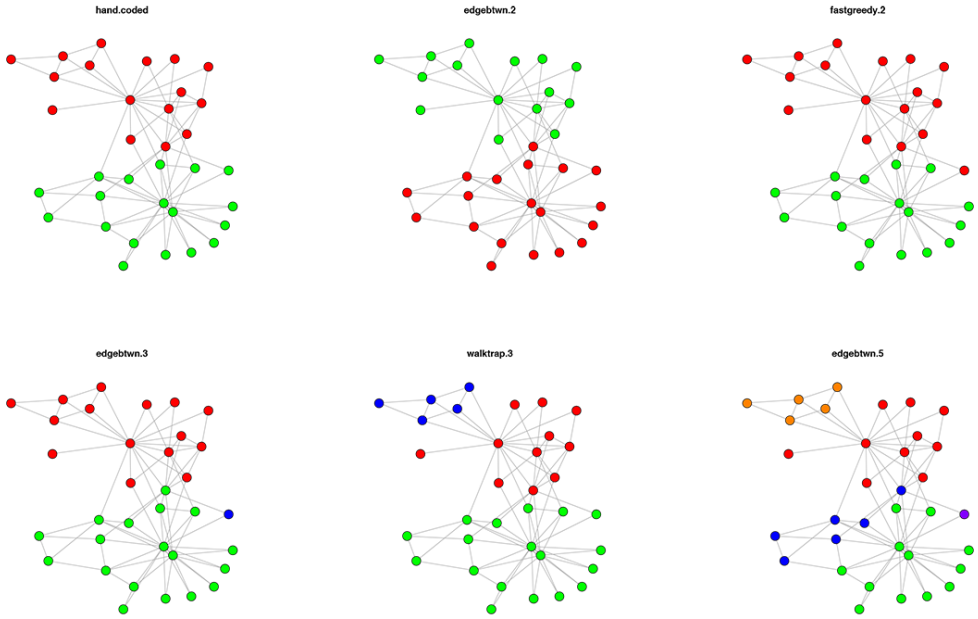

Figure 4: Network visualizations for several clusterings on Zachary's karate club data using the Fruchterman-Reingold layout (Fruchterman and Reingold, 1991). "hand.coded" shows the actual observed split in the club; the other clusterings are each computed by the hierarchical community detection algorithm named in the label and contain the stated number of clusters. In each plot, nodes of the same color are members of the same cluster.

The color-coded scale plot (Figure 2) shows labels for six different clusterings that were selected as "interesting," and Figure 4 shows a network visualization for each. The clustering of the actual split was chosen as a reference, and the four clusterings (fast-greedy agglomeration and edge-betweenness division to two clusters, and edge-betweenness division and walktrap clustering to 3 clusters) that seemed to approximate it most closely were selected. Another clustering more istant from the ground truth version but with a high modularity value (edge-betweenness division to 5 clusters) was also examined.

The clusterings chosen from the region near the hand-coded clustering replicate it very closely. As is clear from the visualization plot, edge-betweenness division to two clusters is the closest match, with only one individual misclassified. In fact this clustering is identical to the one Zachary computes in his work; Zachary explains that the misclassified individual, while socially tied more closely to the student faction, chose to join the other club in order to complete his black belt. So given that this individual's allegiance appears not to be a function of his ties to alters, the walktrap clustering performs as well as any clustering algorithm can be expected to.

The other clustering (edge-betweenness division to five clusters) gives a different picture of the data. Notice that as a hierarchical clustering it is generally consistent with the overall two-faction structure—if we compare it to the two- and three-cluster methods we see that the clusters identified are essentially

subclusters of the same groups—but they focus on identifying smaller substructures. A researcher with a less supervised problem might choose to focus on higher number of clusters. However, edge-betweenness division to two clusters is the best choice for our purpose of predicting a two-way split in the karate club, and the visualization method gives a quick path to that conclusion by showing the edge-betweenness two-cluster solution as the closest point to the ground truth clustering.

Zachary's Karate Club—Positional Analysis

Next we used the same data for a positional analysis. This exercise is necessarily more exploratory than the previous one, since the purpose is now quite different from Zachary's and his labels describing the eventual split can no longer be taken as a ground truth clustering. Computing all the positional clusterings as described in Methods and eliminating duplicate clusterings produced 277 unique clusterings.

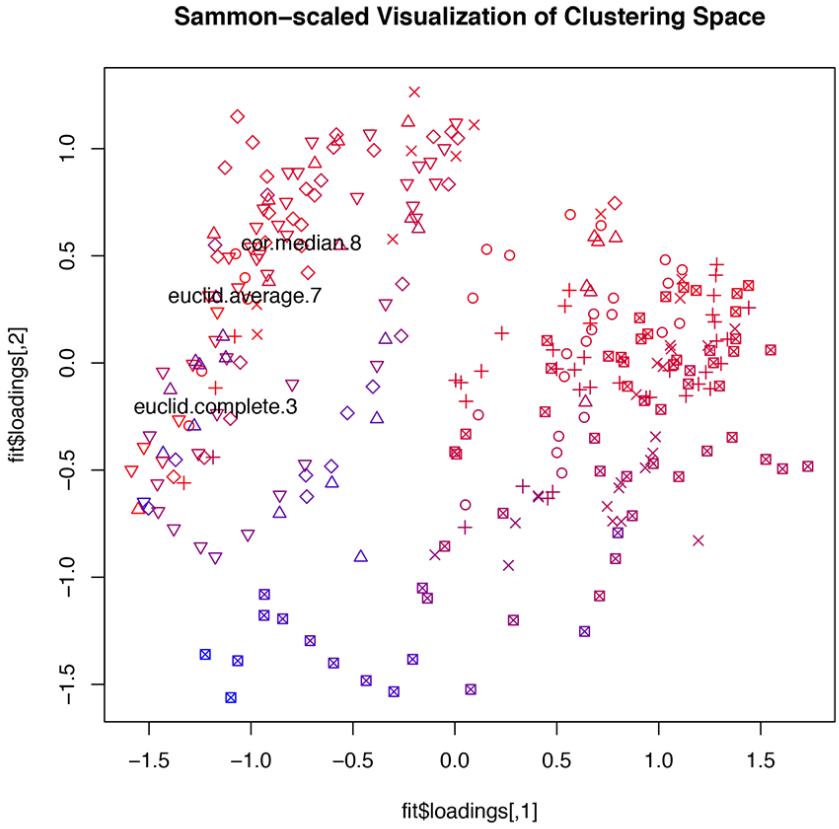

Figure 5: Visualization of positional analysis methods for Zachary's data using color codes given by the G2 score based on Euclidean distance. Red points have a high G2 score (indicating high-quality clusters) and blue points have a low G2 score. Different symbols represent different agglomeration rules: circles represent average linkage, flat-bottomed triangles represent the centroid method, crosses represent complete linkage, X's represent McQuitty's method, diamonds represent median linkage, flat-topped triangles represent single linkage, and boxes with X's inside represent Ward's method. The three labels indicate methods of interest for further investigation — all three are high-quality clusterings that may be good candidates. The labels are located directly below the points they identify.

Figure 5 shows the visualization for positional clusterings of Zachary's dataset. Each plotting symbol corresponds to the agglomeration method used in computing the associated clustering. The color codes correspond to G2 index scores computed using a Euclidean distance matrix. A Shepard plot (Figure 7) was created to judge the quality of fit of the scaling, and the visualization was examined under four other possible color-coding indices to assess the validity of the color index. The Shepard plot shows most points oriented roughly along the $y = x$ line (although the large number of points in the plot means there is some variation around this line). Closer examination shows that most of the strongest-outlying points correspond to distances from clusterings based on the supremum distance, so although we can trust the plot in general we should question the correct distances from these points.

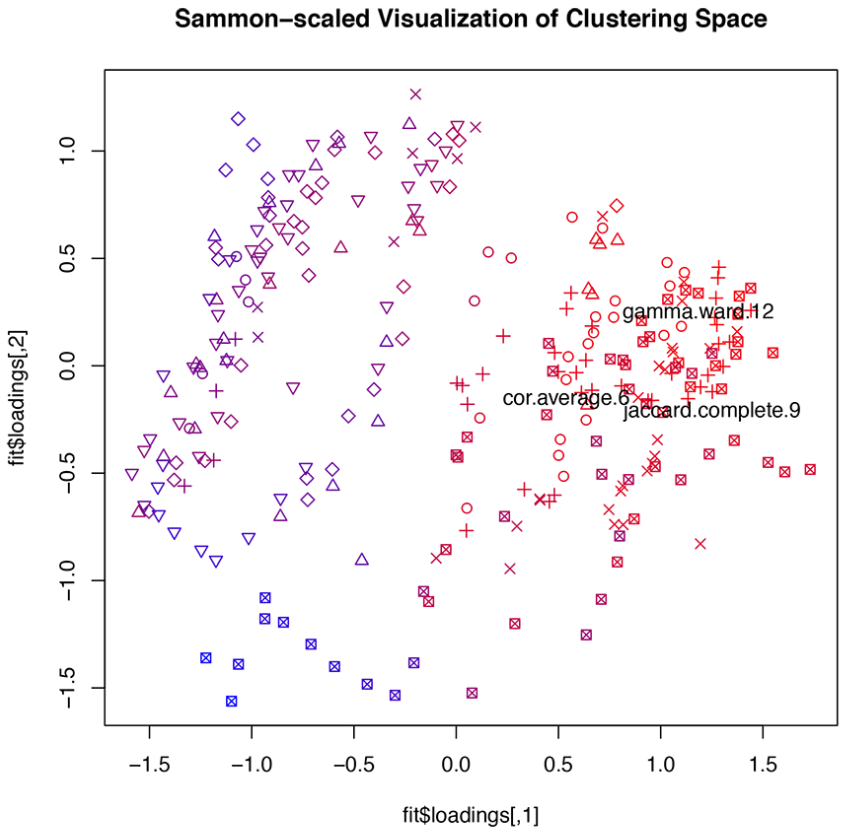

Figure 6: Visualization of positional analysis methods for Zachary's data using color codes given by the G2 score based on correlations. The three labels indicate clusterings judged to be high-quality by this metric. Except for the color codes and the three methods suggested for further study, this figure is identical to Figure ??. Again, the labels are located directly below the points they identify.

The color-code comparison offers more complex feedback about the choice of the Euclidean version of G2. In order to check the validity of these G2 scores as measures of cluster quality, the color-coding for G2 scores was repeated using each of the alternative dissimilarities (Manhattan distance, supremum distance, Hamming distance, gamma correlation, correlation score, Jaccard dissimilarity, and cosine score). Of these different color options, three of them (Manhattan, Hamming, and gamma) were essentially equivalent to the Euclidean-based G2 scheme. The supremum distance produced a degenerate G2 scheme that assigned the same color value to each point (since the supremum distance between any two actors in an undirected

binary matrix is exactly 1, supremum-based measures and clusterings for this dataset tend to be poor). The remaining three dissimilarities produced nearly identical color schemes that differed substantially from the other group. To demonstrate, the data is replotted using the correlation-based G2 score as a color scheme (Figure 6). Looking at both visualizations gives a fuller picture of cluster quality in this space.

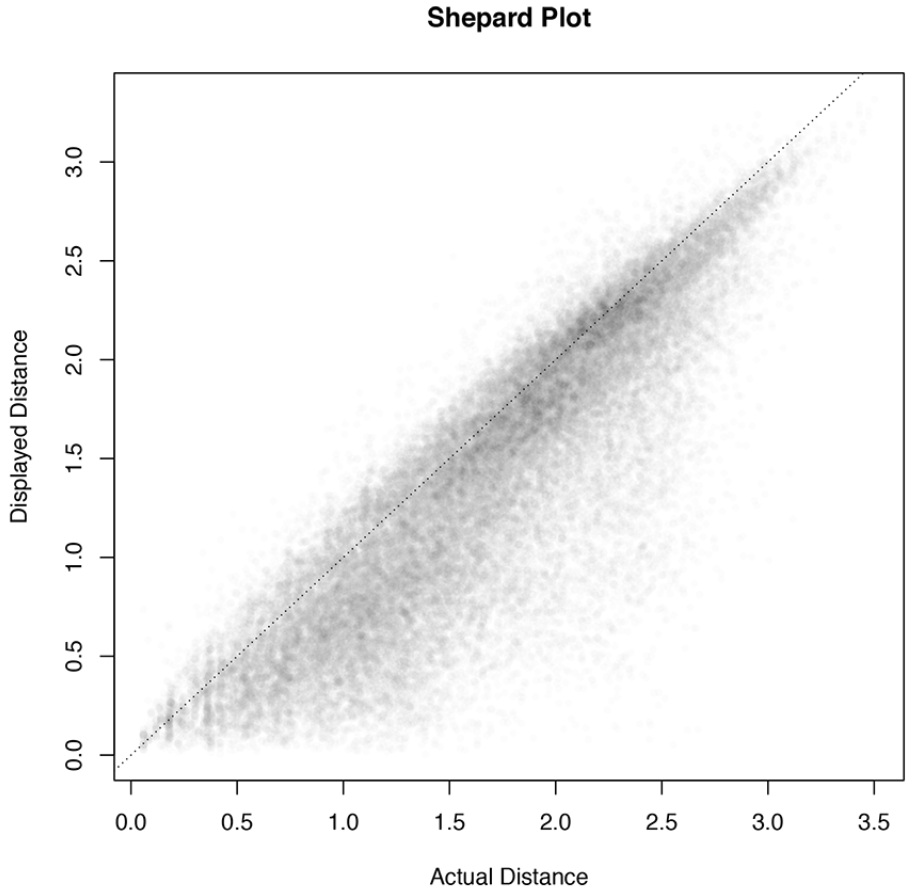

Figure 7: Shepard plot for visualization of positional analysis methods for Zachary's data, indicating the degree of distortion for individual pairwise dissimilarities in the scaled image. Each point corresponds to a unique pair chosen from the 277 clusterings in the visualization, and its position shows how closely the displayed distance between the corresponding clusterings adheres to their actual variation-of-information distance. Points lying on the line $y = x$ represent distances that are accurately depicted in the visualization, and the further a point is from this line the more the distance has been distorted in the scaling.

The resulting plots contain several interesting structures. One is an elongated high-density region in the upper left corner of the plot, and another is a high-density core in the central portion of the plot's right-hand side. The Euclidean-based G2 scores highlights both these regions as high-quality clustering regions, although the correlation-based approach score favors the methods in the right-hand core. Plotting symbols show that the left-hand region consists mostly of methods using a Euclidean distance and using median, single, and centroid agglomeration rules; in contrast, the right-hand consists mostly of clusterings using

Jaccard or correlation-based dissimilarities and agglomeration rules such as Ward distance, McQuitty distance, and complete linkage.

Several thin tails of points appear between the two structures, and another thin tail twists along the bottom of the plot. Most of them tend to have poorer clustering quality (as indicated by blue color codes), and the bottom tail is especially poor according to both color-coding schemes. This tail consists entirely of the poor supremum-based clusterings.

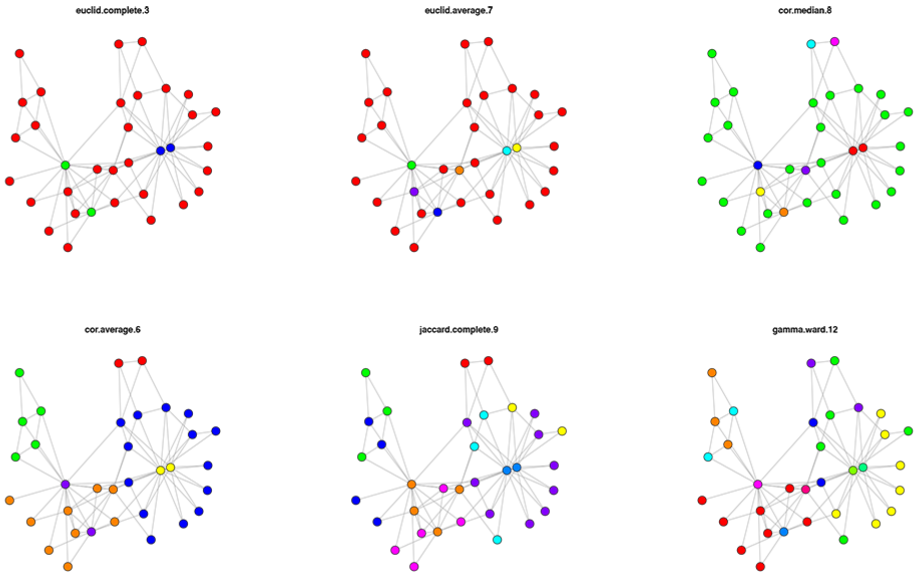

Figure 8: Network visualizations for several positional clusterings of Zachary's karate club data using the Fruchterman-Reingold layout [30]. The colors of the nodes indicate their cluster membership (e.g. all bright red nodes are in the same cluster). The labels for the clusterings contain first a code indicating the dissimilarity used in the algorithm ("euclid" indicates Euclidean distance, "cor" indicates transformed correlation, "gamma" indicates transformed gamma correlation, and "jaccard" indicates Jaccard distance), second a code indicating the agglomeration rule, and finally a number indicating the cluster count and hence the level at which the dendrogram of the hierarchical clustering should be sliced to obtain the clustering. These three pieces of information fully specify the clustering. The three clusterings at the top of the figure were selected from the left-hand portion of the visualization plot and are favored by the Euclidean-based G2 index, while the three on the bottom come from the high-density region on the right-hand side of the visualization and are favored by the correlation-based G2 index.

Three methods from each of the two competing regions of high-quality clustering were chosen and visualized in colored node-and-edge diagrams to compare them qualitatively (Figure 8). The clusterings from the left-hand region (the three in the upper row in Figure 8) tend to distinguish between individual core "leaders" and large periphery of followers. For instance, the positions identified by the euclid.average.7 clustering consist of six clusters with one centrally positioned member each (two of these being the cluster formed by Mr. Hi, the leader of one faction, and the cluster formed by John A., the leader of the other faction) and a seventh cluster composed of all the residual "followers." While the methods from the right-hand region (located in the bottom row of Figure 8) also tend to distinguish these leaders, they also separate the followers from the two factions into separate positional classes. The choice between these two types of clustering is application-dependent. If a researcher is specifically interested in considering members of separate factions to have distinct roles, methods from the core are preferable. However, in general positional analysis does not incorporate constraints of cohesion and since the left-hand structure has generally higher G2 values than the right-hand, we conclude that the best clustering in this case is from the region on the left. There are still many options of clusterings with high G2 values

within this region, but one might decide to look for a small numbers of clusters and on that basis choose the euclid.complete.3 method (shown above in the visualization). Thus a brief consideration of two possible visualizations with different color-codings and a look at a few visualized networks may suffice to fix on a particular dissimilarity (Euclidean distance) and agglomeration rule (complete linkage) of interest for the data.

Discussion

The most evident strength of general-purpose computer-assisted clustering for social networks is the power it gives a researcher to quickly conceptualize the range of possible clusterings available for his or her data. Ultimately the researcher must still make decisions about what type of clustering is best for his or her data and select the algorithm, and using the visualization requires careful thought. Attention to diagnostics such as the Shepard plot is required, and possibly also comparison of multiple versions of the plot incorporating different color-code and plotting-symbol schemes. However, the computer-assisted approach provides a framework for these decision to be made a posteriori and allows the researcher to easily organize and consider concrete information about how a particular method performs on the data rather than having to wade through endless documentation on the parameters and finer points of every possible clustering method.

The community detection example shows how computer-assisted clustering can be used to recreate algorithmically a particular clustering of interest or find methods that separate a particular group hand-coded by a user. On the other extreme, a researcher can use the tool in an exploratory manner to visualize and evaluate a wide range of clusterings, as we do in the positional analysis example.

One possible concern about the use of this method, and of a posteriori approaches in general, is that researchers may simply search through solutions in an unprincipled way, mining the data for a clustering that confirms preconceived ideas about data structure. While the tool may increase a researcher's ability to search out a method that confirms a false hypothesis, it also enhances the researcher's ability to recognize when such a method is inappropriate and to understand and evaluate a wide range of disconfirming clusterings. If a confirming clustering is in a sparse, remote area of the visualization plot and bears a color code indicating poor performance under a cluster quality metric, the researcher should be slow to accept it. On the other hand, a confirming example in a highly populated core indicates a more stable clustering that can be recovered by many methods, and is hence more reasonable. False confirmations can best be avoided when researchers use a priori information to limit the range of methods displayed to those appropriate for the data (for example, using the decision tree in Figure 1).

Grimmer and King suggest that researchers can guard against false confirmations of a hypothesis by taking a random holdout sample of their data before running analysis and re-running candidate clusterings on the holdout data to verify performance. Holdout sampling can be difficult for social network data due to its highly dependent structure, but in certain large datasets one might consider randomly eliminating a few individuals from hypothesized clusters before running a visualization and checking candidate methods against the full data. More theoretical work is needed to understand when such a procedure might be helpful and appropriate.

While researchers may feel overwhelmed by the method's intricate scatter plots populated by hundreds of multicolored points, in principle the method can be focused as narrowly as the researcher desires. For instance, in the positional analysis of Zachary's data, we quickly identified that the supremum distance was a poor dissimilarity to use as a basis for clustering and that reasonably good methods came from two particular regions of the plot populated by largely distinct families of clusterings. A researcher who wished

for a sparser, more immediately interpretable visualization could easily have re-run the analysis for a more refined group of clustering methods (those from one of the two high-quality regions, perhaps).

The method's flexibility makes it relevant in almost any application of clustering. Wherever a clustering algorithm has been applied successfully, general-purpose computer-assisted clustering should be able to match its results by confirming that the algorithm chosen is a good fit and possibly presenting other equally good or better clusterings. As Grimmer and King say of their general-purpose computer-assisted clustering method, "by definition, any one individual method cannot outperform the approach proposed here". In fact, Grimmer and King substantiate this claim through controlled experiments in which data analyses by researchers using the computer-assisted method are compared impartially to analyses done by conventional means and found to be consistently preferable. While these experiments only examined their particular software tool in text analysis applications, there is every reason to expect that similar improvements are gained by using an analogous tool for network data.

Several important obstacles remain in making general-purpose computer-assisted clustering practical for network data in general. First, the current pipeline does not support all classes of clustering methods. The current implementation of the Variation of Information metric works only for disjoint clusterings; to allow fuzzy and overlapping clusterings to be embedded into the visualization plot, a new implementation would need to be chosen. More generally, many clustering methods lack an easily-accessible R implementation and so cannot be smoothly incorporated into the general-purpose clustering algorithm without a large amount of up-front effort. In addition, the current method may not scale well with the size of the network dataset. While 277 clusterings of the 34 members of Zachary's karate club were performed in an infinitesimal interval, computation costs for performing the same clusterings on a network of ten thousand nodes could prove prohibitively expensive. Even if the clusterings could be quickly performed for such a network and interesting methods selected from the Sammon-scaled distance plot, it would be difficult to make a swift visual comparison of the clusterings using the simple network visualization layouts the current method relies upon, and some alternate form of visualization might need to be incorporated. The challenge of handling large networks is particularly pressing because of the increasing prevalence of massive network datasets.

Areas for Innovation

Since many social network datasets have naturally overlapping community structures (Goldberg, Hayvanovych and Magdon-Ismail, 2010), it would be particularly desirable to adapt the current method to handle fuzzy and overlapping clusterings. The difficulty in doing so is choosing a metric to compare these types of clusterings in place of the VI metric for disjoint clusterings. Grimmer and King describe one version of the VI metric that is appropriate for use in visualizing fuzzy clusterings and argue that it is the optimal method under certain assumption, but it is not clear that this is the only possible choice. For example, Cao et al. present a different modification of the VI metric for fuzzy clusterings (2008). For overlapping clusterings, no version of the VI metric seems to exist yet, although recent research has produced other candidate dissimilarity indices. Lancichetti, Fortunato and Kertesz define an index for comparing overlapping clusters based on entropy in (2009). Goldberg, Hayvanovych and Magdon-Ismail present several other indices designed specifically for use with social network data (2010). While not all of these indices are necessarily metrics, some could likely be adapted and implemented in R for use in place of the VI metric, allowing the visualization tool to include a much richer class of clusterings.

The tool could also be improved by adding support for more clustering methods. Many important clustering algorithms used in social network analysis still need to be added, notably methods appropriate for directed and valued ties. In addition, Grimmer and King's original visualization method allowed users

to examine weighted combinations of the computed clusterings by selecting locations between points of interest in the visualization space. Building such support into our tool would allow researchers not only to examine clusterings points actually plotted as points in the distance plot but to interpret any location in the plot as a clustering. Although it would substantially alter the a posteriori interpretation process, such an enhancement could produce valuable candidate clusterings for temperamental clustering problems for which current methods do not perform well.

Developing support for very large networks provides another area for further research. The current approach expects all individual clustering algorithms to be global; in order to provide general-purpose computer-assisted clustering displays for massive networks, one might adapt the method to incorporate the more computationally efficient class of local methods. Even for global clusterings, the current general-purpose clustering method could scale well if slow algorithms could be identified and excluded from consideration. In practice, this could function as an additional branch in the a priori decision tree, separating poor-performance methods from high-efficiency methods based on appropriate metrics of computational tractability. Each such improvement will expand the researcher's power to make wise and well-informed choices among the many options and find the methods most appropriate for his or her data.

Acknowledgments

Dan McFarland and Justin Grimmer provided valuable input on several occasions. The members of the spring 2013 Social Network Analysis workshop at Stanford University gave useful feedback on an early draft, and insightful suggestions from an anonymous reviewer also proved helpful. This work was supported in part by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program.

References

Bezdek, James C., Robert Ehrlich, and William Full (1984). "FCM: The Fuzzy C-Means Algorithm." *Computers & Geosciences* 10, 2-3: 191-203.

Buja, Andreas, Deborah F. Swayne, Michael L. Littman, Nathaniel Dean, Heike Hofmann, and Lisha Chen (2008). "Data Visualization With Multidimensional Scaling." *Journal of Computational and Graphical Statistics*, 17, 2: 333-372.

Carter T. Butts <buttsc@uci.edu> (2010). sna: Tools for Social Network Analysis. R package version 2.2-0. http://CRAN.R-project.org/package=sna

Cao, Tru Hoang, Hai T. Do, Dung T. Hong, and Thanh Tho Quan (2008). "Fuzzy Named Entity-Based Document Clustering." In *Fuzzy Systems, 2008. FUZZ-IEEE 2008.(IEEE World Congress on Computational Intelligence). IEEE International Conference on*, pp. 2028-2034.

Clauset, A., M. E. J. Newman, and C. Moore (2004). "Finding Community Structure In Very Large Networks." *Phys. Rev. E* 70, 066111. arXiv:cond-mat/0408187v2

Dimitriadou, Evgenia, Kurt Hornik, Friedrich Leish, David Meyer, and Andreas Weingessel (2011). e1071: Misc Functions of the Department Of Statistics. TU Wien. R package version 1.6. http://cran.r-project.org/web/packages/e1071

Fortunato, S. (2010). "Community Detection in Graphs." *Physics Reports* 486: 75-174. arXiv:0906.0612v2

Fritsch, Arno (2009). mcclust: Process an MCMC Sample of Clusterings. R package version 1.0. http://CRAN.R-project.org/package=mcclust

Fruchterman, T. M. J., and Reingold, E. M. (1991). "Graph Drawing by Force-directed Placement." *Software: Practice and Experience* 21, 11: 1129-1164.

Gest, Scott D., James Moody, and Kelly L. Rulison (2007). "Density or Distinction? The Roles of Data Structure and Group Detection Methods in Describing Adolescent Peer Groups." *Journal of Social Structure* 8, 1. http://www.cmu.edu/joss/content/articles/volume8/GestMoody/

Goldberg, Mark K., Mykola Hayvanovych, and Malik Magdon-Ismail (2010). "Measuring Similarity Between Sets of Overlapping Clusters." In *Social Computing (SocialCom), 2010 IEEE Second International Conference on*, pp. 303-308.

Goodman, Leo A., and William H. Kruskal (1954). "Measures of Association for Cross Classifications." *Journal of the American Statistical Association* 49, 268: 732-764.

Gordon, A. D. (1999). *Classification*. 2nd ed. (Boca Raton: Chapman & Hall).

Grimmer, J., and G. King (2011). "General Purpose Computer-Assisted Clustering and Conceptualization." *PNAS* 108, 7: 2643-2650.

Handcock, M. S., A. E. Raftery, and J. M. Tantrum (2007). "Model-Based Clustering for Social Networks." *J. R. Statist. Soc. A* 170, Part 2: 301–354. See Discussion on the Paper (Krackhardt), 341.

Hennig, Christian <chrish@stats.ucl.ac.uk>, and Bernhard Hausdorf <Hausdorf@zoologie.uni-hamburg.de> (2010). prabclus: Functions for clustering of presence-absence, abundance and multilocus genetic data. R package version 2.2-2. http://CRAN.R-project.org/package=prabclus

Hennig, Christian (2013). fpc: Flexible procedures for clustering. R package version 2.1-5. http://CRAN.R-project.org/package=fpc

Kaufman, L., and P. J. Rousseeuw (1987). "Clustering by Means of Medoids." In Y. Dodge, ed. *Statistical Data Analysis Based on the L1-Norm* (North Holland: Elsevier), 405-416.

Krishnapuram, Raghu, and James M. Keller (1993). "A Possibilistic Approach to Clustering." *IEEE Transactions on Fuzzy Systems* 1, 2: 98-110.

Lancichinetti, Andrea, Santo Fortunato, and János Kertész (2009). "Detecting the Overlapping and Hierarchical Community Structure in Complex Networks." *New Journal of Physics* 11, 3: 033015.

McFarland, Daniel, Solomon Messing, Mike Nowak, and Sean Westwood (2010). "Social Network Analysis Labs in R." Stanford University.

McQuitty, Louis L. (1966). "Similarity Analysis by Reciprocal Pairs for Discrete and Continuous Data." *Educational and Psychological Measurement* 26: 825-831.

Meila, M. (2007) "Comparing Clusterings: An Information Based Distance." *Journal of Multivariate Analysis* 98, 5: 873-895. http://www.stat.washington.edu/mmp/Papers/compare-jmva-revised.pdf

Milligan, Glenn W., and Martha C. Cooper (1985). "An Examination of Procedures for Determining the Number of Clusters in a Data Set." *Psychometrika* 50, 2: 159-179.

Newman, M., and M. Girvan (2004). "Finding and Evaluating Community Structure in Networks." *Physical Review E* 69, 026113. http://pre.aps.org/abstract/PRE/v69/i2/e026113

Newman, M. E. J. (2006). "Finding Community Structure Using the Eigenvectors of Matrices." *Physical Review E* 74, 036104. arXiv:physics/0605087v3

Ovelgönne, Michael, and Andreas Geyer-Schulz (2012). "An Ensemble Learning Strategy for Graph Clustering." In *Graph Partitioning and Graph Clustering*, 187-206.

Pons, P., and M. Latapy (2005). "Computing Communities in Large Networks Using Random Walks." *Lect. Notes Comput. Sci.* 3733, 284293. http://arxiv.org/pdf/physics/0512106v1.pdf

Sammon, J. W. (1969). "A Non-Linear Mapping for Data Structure Analysis." *IEEE Transactions on Computers* C-18, 5: 401–409.

SAS Institute (1990). "Introduction to Clustering Procedures" and "The CLUSTER Procedure." Chapter 6 in *SAS/STAT User's Guide, Volume 1* (Cary, NC: SAS Institute Inc.), 53-101.

Satu, Elisa Schaeffer (2007). "Graph Clustering." *Computer Science Review* 1, 1: 27-64.

Scott, J. (2000). *Social Network Analysis: A Handbook*. 2nd edition (Thousand Oaks, CA: Sage Publications).

Shepard, R. (1962). "The Analysis of Proximities: Multidimensional Scaling with an Unknown Distance Function, I & II." *Psychometrika* 27: 125-140, 219-246.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. 4th edition (New York: Springer).

Ward Jr., Joe H. (1963). "Hierarchical Grouping to Optimize an Objective Function." *Journal of the American Statistical Association* 38, 301: 236-44.

Wickham, Hadley (2011). "The Split-Apply-Combine Strategy for Data Analysis." *Journal of Statistical Software* 40, 1: 1-29. http://www.jstatsoft.org/v40/i01/

Wild, Fridolin (2011). lsa: Latent Semantic Analysis. R package version 0.63-3. http://CRAN.R-project.org/package=lsa

Zachary, W. W. (1977). "An Information Flow Model for Conflict and Fission in Small Groups." *Journal of Anthropological Research* 33: 452-473.

[1] Handcock, M. S., A. E. Raftery, J. M. Tantrum (2007). "Model-based Clustering for Social Networks." *J. R. Statist. Soc. A* 170, Part 2: 301–354. See Discussion on the Paper (Krackhardt), 341.