Chapter 6

A Graph-Theoretic Approach to Concept Clustering

Chris Esposito

Interobject proximities are an essential input to any of the Pathfinder algorithms that have been developed. However, the role of these proximities or weights after the network has been generated is less clear. There are a variety of viewpoints on this issue, all of which can be cast as different points on a continuum.

At one endpoint is the view that the interobject proximities are still the focus of attention, and that the edges in a PFNET are merely places to hang the most salient of these proximities. Since the general notion of distance is intimately related to the notion of a space, this view of PFNETs may often be adopted when aspects of a spatial model are appropriate and useful. The distance between two nodes is primarily determined by the weights attached to the edges that join them; structural issues, such as the number of edges in the connecting path, are secondary considerations.

At the other endpoint there is the view that the role of the edge weights is largely confined to determining network structure during generation. Such a view treats the PFNET simply as a graph. While it is clearly possible to define the distance between two nodes as the minimum number of edges needed to connect them, this measure is driven entirely by the structure of the network, and there is no guarantee (especially for directed networks) that distance measures so derived will satisfy the axioms for a given type of space, such as a metric space.

To a certain extent, the relative importance assigned to edge weights and network structure depends on the application, or the question to be answered. There are several applications which adopt a view that is in between the two endpoints described above, and use both edge weight and structural information. The work described in Knoebel, Dearholt, and Schvaneveldt (1988) or the database searching work described in Dearholt and Gonzales (1987) are examples of this middle view. This chapter examines how well network solutions fit the original data and compares measures of fit based on graph structure with measures based on edge weights. Also, a study is presented that examines the issue of edge weights versus structure from another perspective, the subjective ratings of clusters produced using edge-weight information (hierarchical clusters) are compared with ratings of clusters produced by some graph-theoretic clustering methods that ignore edge weights.

Parameter Choice and Network Fit

A general question that applies to most data-scaling methods (Pathfinder included) is this: Given a particular dataset and a scaling solution, such as a multidimensional scaling (MDS) layout or a PFNET, how good is the solution? There are several points to consider in answering this question. The first is the simplicity of the solution. If we start with an

$n \times n$ proximity matrix for n objects, then an MDS solution with some small (usually 2 or 3) number of dimensions is preferable to a solution with n–1 dimensions. A PFNET with a significant portion of the original $n^2/2$ edges remaining is, for most exploratory purposes, too dense and complex to be very useful. The experience of looking at large numbers of networks suggests that those where the number of edges is no more than about twice the number of nodes are the easiest to deal with.

However, simplicity is not the only consideration in assessing a solution. We would also like the solution to fit well with the original data and for the essential structure to be revealed with nonessentials carved away. There is a certain tension between these two criteria. The simplest solutions (usually trees in Pathfinder) can be too simple and not represent the latent structure in the original data very well. On the other hand, a trivial way to get a good fit is to essentially reproduce the original *complete network*, deleting few, if any edges. Of course, such a solution is often too complicated to be of much use. The desirable middle ground is a solution that is both simple and fits well with the original data.

Weights vs. Edges

In order to discuss how good a network solution is for a given dataset, we must first examine the factors that determine the simplicity and fit of the network. The most obvious determinants are the network generation parameters q, r, and z, since these completely determine the network for a given dataset. Factors that may be less obvious, but are just as important, are the choice of a correlation measure and the choice of what measurements to take on the network in order to compute the correlation measure. The q values used in the study reported in this chapter were 2 to n–1. The r values were 1 to 8. The z values[1] used were 0.0 to 3.0. The correlation measures used were the Pearson r_p (r_p will be used here instead of r to distinguish the statistic from the Pathfinder r parameter) and Spearman's ρ. The principal difference between these two is that r_p uses interval-scale properties while ρ only makes ordinal assumptions. As it turned out, they produced very similar correlations (as they often do), and so if we perform similar studies in the future, we would probably use only one of them.

Five domains were chosen, and for each domain, a set of representative terms was chosen. The domains were countries (9 terms); pieces of fruit (11 terms); items of clothing (11 terms); cities in New Mexico (20 terms); and a set of words extracted from a machine-readable copy of *Longman's Dictionary of Contemporary English* that were related to the word *bank* (20 terms). For each set of terms, pairwise relatedness ratings were obtained from human subjects to get a proximity matrix. From the individual matrices, an average proximity matrix and a standard deviation matrix were created for each domain. For each domain, three sets of networks were generated. In the first set, q was held to be n–1, z was set to 0.0, and r was varied from 1 to 8 (higher values of r usually produced networks identical to when $r = \infty$). In the second set, r was held at infinity, z was set to 0.0, and q was varied from 2 to n–1. In the third set, r was set to infinity, q was set to n–1, and z was varied from 0.0 to 3.0 in 0.1 increments.

For each network generated, two distance matrices were computed and each was correlated with the original data matrix using the two measures of fit noted above. The first distance matrix was computed by assuming that the distance between two nodes in the network is primarily a function of the weights attached to the edges on the path that joins them. The r value used to compute this distance matrix was always the same as the r value used to generate the network. While it is certainly true that the structure determines what

[1]For a discussion of how the z parameter is used in network generation, see Esposito (Chapter 3, this volume).

paths exist between any pair of nodes, there is no necessary correlation between the distance from one node to another and the number of edges that separate them. Let this set of matrices be called *weight-based*. The second distance matrix was calculated by assuming that the structure of the network is essential. The distance between a pair of nodes is defined to be the minimum number of edges that separate them. Let these matrices be called *edge-based*.

Across the five domains, 353 networks and 706 network distance matrices were generated. The correlations between these network distance matrices and the five averaged proximity matrices provided some interesting results. In 291 of these networks (83%), the edge-based matrices correlated more highly with the raw data than the weight-based ones. As Figures 1 through 3 demonstrate, this was often by a substantial margin.

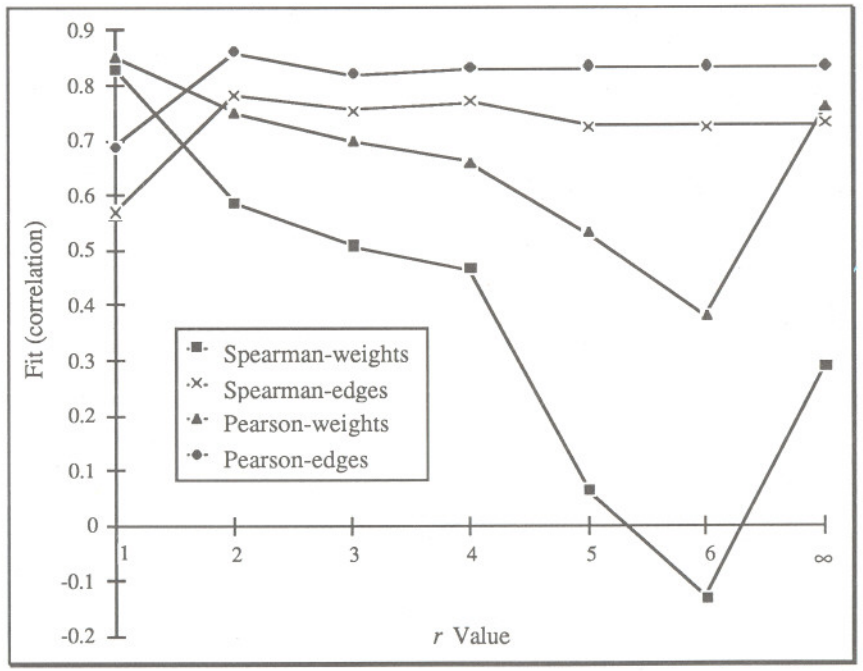

Figure 1. Fit as a function of r value for the Countries dataset, PFNETs($q = n$-1, $z = 0$).

In 62 networks (17%), the weight-based matrices correlated more highly, although the difference between the correlations was often much less. Part of this dominance by the structural view of the networks is explainable if we consider that in 328 of the 353 networks, r was held at infinity. This means that the length of a path is the length of the longest edge on it, so the largest weights occur far more often (and the smallest weights far less often) in the weight-based distance matrices than they did in the original proximity matrices. Since the edge-based approach ignored edge weights, no such skewing was present there. It is therefore not too surprising that the weight-based matrices correlated more poorly with the original data matrices than did the edge-based ones.

It is probably more revealing to look at those 35 networks where r was allowed to vary. In these cases, $q = n-1$ and $z = 0.0$. In 28 (80%) of these networks, the structural (i.e., edge-based) view still correlated more highly with the original data than the weight-based approach. In 7 networks (20%) it does not. Figure 1 shows some representative results. One conclusion that can be drawn from this analysis is that the principal role of the weights is in creating the right structure for the network. After generation, many uses of the network would be much better off concentrating on the graph-theoretic aspects of the network.

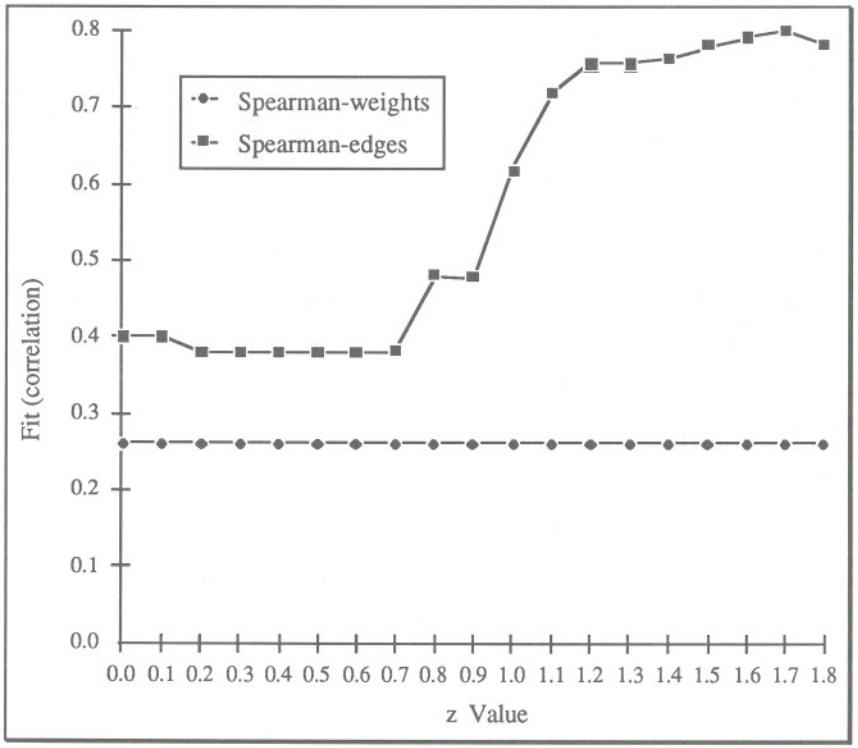

Figure 2. Fit as a function of z value for the Cities dataset, PFNETs(r = ∞, q = n–1).

An Application of Network Structure

One of the results from the last section was that when measuring the fit between a Pathfinder network and the original data, measures of fit based on *graph-theoretic distance* proved generally superior to those measures based on edge weights.

In this section we will examine the issue of edge weights versus structure from a different perspective. A common approach when investigating the structure of a domain is to see how the concepts in that domain are organized into categories or clusters. We describe a study that compares the ratings of clusters produced by using edge-weight information

(hierarchical clusters) with ratings of clusters produced by several graph-theoretic methods that ignore edge weights. A key part of this study is a different approach to clustering validity than is usually adopted, so it is worth discussing some of the validity issues that influenced the design of the study.

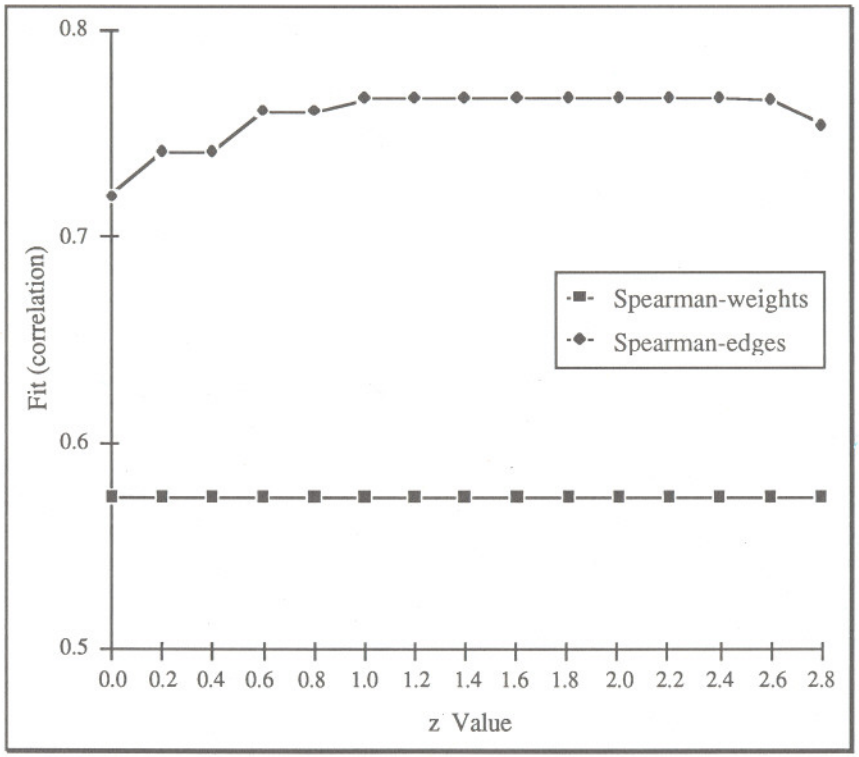

Figure 3. Fit as a function of z value for the Clothing dataset, PFNETs($r = \infty$, $q = n-1$).

General Validation Issues

One of the primary problems in cluster analysis is that the methods for evaluating the validity or usefulness of the generated clusters lag far behind the methods for producing them. The validation process generally proceeds down one of two roads that, unfortunately, rarely intersect. The first road is the researcher's purely subjective evaluation of the produced clusters. In the early stages of exploratory data analysis, it is reasonable for a researcher to use high subjective ratings of clusters in a familiar domain as an indication that the clusters are worth a closer and much more objective look. However, it is embarrassingly easy to unconsciously impose a structure on a two-dimensional plot of a dataset (for example) and then "find" evidence for the "natural" existence of the structure imposed. If these subjective ratings are the only factors considered when assessing clustering validity, there is an unfortunate tendency to treat them as if they are much more conclusive than they really are.

The second road is in many ways the more seductive of the two, in that clustering validity, for which there is a substantial body of literature, is cast in terms of optimizing a criterion function. The assumption implicit in this approach is that any clustering produced by such an optimization will also be optimal, or nearly so. This approach has several problems. First, the criterion is often chosen with little (if any) justification as to why optimizing it leads to better clusters. Second, as Dubes and Jain (1977) point out, many clustering techniques have no problem in producing clusters whether or not they are real. To use some criterion to form the clusters and then also use it to judge their validity effectively sidesteps the entire issue. Some concrete examples from the clustering literature should make the problems often encountered with this second approach a bit clearer.

One class of clustering algorithms (examples are Ball & Hall, 1970; Kennard & Stone, 1969; and MacQueen, 1967) begin with all of the items embedded in some p-dimensional space and then partition this space into some usually prespecified number of clusters. Some sort of global measure of fit (often a squared-error minimization) is used as the criterion function that guides placement of the items into the partitions.

The first problem encountered when using this sort of clustering method is picking the "right" number of clusters or partitions. The usual response is to plot the criterion function value versus the number of clusters, and hope (often in vain) that a sharp and clearly marked *elbow* will appear, thus marking what is presumably the right number of groups. However, as Thorndike (1973) points out, unless one has already demonstrated that optimizing the chosen criterion leads to *better* clusters (i.e., ones with higher subjective ratings), identifying the elbow often gives unsatisfactory results.

The second problem is that choosing a criterion function to optimize generally entails the adoption of some assumptions about the structure of the clusters to be produced. Many of the criteria that have been proposed (Friedman & Rubin, 1967; McRae, 1971) produce spherical clusters. A less restrictive but still common assumption is that all of the clusters have the same shape (Marriot, 1971). The basic problem here is one of imposed structure versus revealed structure: Clusters whose shapes are determined by these assumptions will be produced whether or not they naturally exist, and naturally existing clusters whose shapes are not in accordance with these assumptions will be largely ignored.

Method

The issue at hand is this: We are interested in assessing the value of several graph-theoretic structures as clustering methods by comparing their performance both with a control group of clusters and with a well-accepted hierarchical clustering method. A list of these structures will be given shortly. The domain for this pilot study is the UNIX[2] command set (McDonald & Schvaneveldt, 1988). The set of terms contains 152 UNIX commands. The relatedness data for the generated network were collected in the following way. Each command was put on a card and experienced UNIX users were asked to put related commands in the same pile. There was no constraint on the total number of piles, the size of each pile, or the number of piles a command could appear in (subjects were allowed to make duplicate cards). The proximity between a pair of commands was defined to be the conditional probability of one command in a pair being in a pile given that the other command was already in that pile. For the details on the computation of these conditional probabilities, see McDonald and Schvaneveldt (1988).

This proximity matrix was used in two ways. The first use was as input to a single-link hierarchical cluster analysis. This procedure produced 83 clusters that four subjects were asked to rate. In order to make the later comparisons more meaningful, only the

[2]UNIX is a trademark of AT&T Bell Laboratories.

clusters in the same size range as the graph-theoretic structures (3 to 8 nodes) were chosen, for a total of 36 clusters. For each cluster, the subjects were asked to rate its quality on a scale of 0 to 5, with 0 as "don't know" (usually due to unrecognized terms in the cluster), 1 as "very bad," and 5 as "very good." The subjects were also asked to name the cluster if the rating was ≥ 2. The assumption was that if the rating was either "don't know" or "very bad" there was little point in asking for a name.

The other use for this proximity matrix was as input to Pathfinder. Since no estimates of rating variability could be obtained from these data, the z parameter was set to 0. The q parameter was set to 151 (n–1) and the r-metric to ∞. For each cluster generated from this network, we collected the same information (rating and name) as we did for the hierarchical clustering case, although we used a separate group of eight subjects for this rating process. A list of the cluster structures extracted from this network is given below.

Cliques. A complete graph is one in which every pair of nodes is *adjacent*. A subgraph S is *maximal* with respect to a property P if there is no node v such that $S + v$ also has property P. A clique is a maximal complete subgraph. For the purposes of this study, there were 10 subgraphs that would have been cliques (specifically K_4s) except that a single edge was missing. These were judged to be sufficiently clique-like so that they were included in this category.

Blocks. A *cutpoint* of a graph is a node whose removal increases the number of components, and a bridge is such an edge. A *nonseparable* graph is connected, nontrivial, and has no cutpoints. A *block* of a graph is a maximal nonseparable subgraph (Harary, 1969). Preliminary examination of some PFNETs reveals that many blocks correspond to conceptually coherent categories, such as the set of UNIX mail commands.

Stars. If entities in a basic-level category (Rosch, Mervis, Gray, Johnson, & Boyes-Braem, 1976) are presented to subjects, and relatedness estimates are used as the basis for a PFNET, the resulting network is organized as a *star* (Schvaneveldt, Durso, & Dearholt, 1985) with the category name in the center and exemplars at the tips. The high degree of the center node and the low degree of other nodes adjacent to it provide the basis of a metric for detecting starlike structures. For the purposes of this study, any node with a degree greater than 4 (along with its neighbors) was considered to be a star.

In addition to these structures, a group of random subgraphs was presented for subjects to rate. The presence of random clusters complements the use of subjective ratings as an evaluation measure in that the average rating for this group should provide an idea of what subjective numeric rating corresponds to a verdict of *no clustering*.

There are several ways of constructing random subgraphs. Perhaps the simplest way is to generate random sets of nodes, without concern for whether any pair of nodes or the set of nodes as a whole is connected. However, the size and diversity of the concept set, combined with the relative sparseness of the network, mean that the large majority of random clusters generated without regard to connectedness will be of such low quality that if comparing the average random cluster rating with the average rating for one of the nonrandom types is a test, then it is a test that is very difficult to fail. Passing such a test would then be of little value. A more restrictive criterion that leads to a stricter test is to require that each of the random clusters form a connected subgraph.

The first step in this assessment is to generate a network where reasonable numbers of these nonrandom structures exist for examination. This, in turn, requires that the network be neither too dense nor too sparse. The second step is to look at knowledgeable subjective ratings of these clusters. If these structures have any value as clustering methods then it is reasonable to expect the average subjective rating for each type to exceed the average rating for random connected subgraphs.

The generated network had 184 edges and contained 34 cliques, 23 starlike structures and 45 blocks, so there appeared to be enough of these structures to obtain reasonable averages for each type in this dataset. The number of nodes in these structures ranged from 3 to 8, distributed as follows: 3 nodes (41%), 4 nodes (23%), 5 nodes (17%), 6 nodes (14%), 7 nodes (3%), 8 nodes (1%). In order to be able to make some meaningful comparisons, we took the ratings of 30 randomly chosen connected subgraphs as a baseline for comparison. The size distribution for this set of subgraphs was the same as the above size distribution for the graph-theoretic structures.

Many of the criticisms of other clustering methods discussed earlier stem from two sources. First, these methods are often applied with little regard to the fit between the assumptions made by the scaling model and the properties of the data. Applying a method that makes interval assumptions to ordinal data is a classic example of this. Second, it is often very difficult to assess the validity of a solution. As previously noted, a common problem is to use the same criterion for both cluster generation and cluster evaluation. In this section we deal with both of these criticisms as they might apply to the present work.

With regard to the first criticism, the construction of the networks (with $r = \infty$) depends only on ordinal properties of the data. The graph-theoretic structures depend only on the presence or absence of edges in these networks and so also make only this assumption.

As for the second criticism, the present work clearly separates the generation and evaluation criteria. The generation criterion is that each cluster possess some specified structural properties. The evaluation criterion is that the average subjective rating for that structure type be greater than the average rating for random subgraphs by a statistically significant amount.

Results

As discussed in the previous section, there are five cluster types being examined in this study: random, single-link hierarchical, stars, blocks, and cliques. In addition, each structure had several different sizes. For each subject, we computed an average goodness rating for each type. For each type, we computed an average and standard deviation for each size across subjects and an average across all sizes and subjects. In addition, we combined the ratings for the stars, cliques, and blocks into a single group and computed an average and standard deviation for this group. The mean values are shown in Table 1.

One can frame the analysis in terms of three broad questions. First, which nonrandom cluster types outperform the random clusters? Second, how do the ratings of the graph-theoretic (star, block, and clique) clusters compare with the ratings of the hierarchical clusters? Third, are there substantive rating differences between the graph-theoretic clusters?

In answering these questions, several factors play a role in determining precisely what comparisons will be made and what statistics will be used to evaluate them. The small sample sizes require that the t statistic be used. We can answer the three broad questions given earlier by doing seven comparisons and t tests. Four of these tests are individual comparisons between random cluster ratings and the ratings of each of the nonrandom cluster types. The fifth comparison is between the hierarchical ratings and a combined average rating for the graph-theoretic structures. The sixth comparison is between cliques and blocks. The last comparison is between stars and blocks. Since one group of subjects rated the hierarchical clusters and a separate group rated the graph-theoretic and random clusters, comparisons involving hierarchical cluster ratings used t tests for between-group comparisons with independent sampling. All other comparisons used t tests for correlated samples. For each comparison, the size of the clusters was controlled. For example, cliques were compared only against random clusters of sizes 3 and 4 nodes, since there were no larger cliques.

Table 1. Subjective rating means by cluster type and size.

	Cluster Type				
Size (nodes)	*Random Connected*	*Star*	*Block*	*Clique*	*Single-Link Hierarchical*
3	3.01	-	3.61	4.07	3.96
4	3.38	-	3.63	4.15	4.16
5	2.98	3.59	3.62	-	4.00
6	2.55	3.80	3.07	-	3.83
7	2.91	2.87	3.09	-	2.75
8	2.71	-	3.71	-	4.25
Average	2.92	3.38	3.46	4.12	3.83

For simplicity, let us first deal with the two between-group comparisons, hierarchical ratings versus random ratings, and hierarchical ratings versus average graph-theoretic ratings. Table 2 shows the t value and level of significance for each of the two tests.

The fact that hierarchical clustering significantly outperformed random clustering is not very surprising; the method, when properly used, has proven its value over the years. Given the good showing of the hierarchical clusters, it is also encouraging to note that there is no significant difference between the ratings of the hierarchical clusters and those of the graph-theoretic cluster types, taken altogether. For a more detailed look at each of these cluster types, we must examine the within-subject ratings.

Table 2. Between-group t -test results.

Cluster Type Comparison	*t (10 df)*	*Probability*
Hierarchical vs. Random	3.902	$p < 0.005$
Hierarchical vs. Average Graph-Theoretic	0.869	$p > 0.10$

Table 3 shows the results of the remaining five comparisons. Since there were eight subjects, there are seven degrees of freedom. One bit of notation worth explaining is that the numbers following a cluster type name indicate that only clusters of that size were used in that comparison; for example, "Clique vs. Random (3,4)" means that only random clusters of sizes 3 and 4 were used in that comparison.

Table 3. Within-subject *t* -test results.

Cluster Type Comparison (Number of Nodes)	*t (7 df)*	*Probability*
Clique vs. Random (3,4)	16.12	$p < 0.005$
Block vs. Random	8.68	$p < 0.005$
Star vs. Random (5,6,7)	6.06	$p < 0.005$
Clique vs. Block (3,4)	7.97	$p < 0.005$
Star vs. Block (5,6,7)	2.41	$p < 0.025$

The results of the top three comparisons in the table confirm a conclusion reached from the between-group tests, that the graph-theoretic clusters significantly outperform random clusters. The results of the last two tests suggest that there are significant differences in how well the graph-theoretic clusters perform as clustering methods. The average clique rating is substantially higher than either the block average or the star average, so among this group it emerges as a fairly clear winner. Stars are in second place, outperforming blocks. The dominance of cliques as a clustering method is not unique to this work; Shepard and Arabie (1979) report similar results and characterize a clique as a collection of entities that all share one or more common properties. The fact that some of the structures used in this study were actually *near-cliques* (since one edge was missing) suggests that this criterion is somewhat robust in the face of small violations of the structural requirements.

The cluster types with the two highest averages are cliques and hierarchical clusters. If, as the results just obtained suggest, we accept the proposition that these are generally good clustering methods, it is reasonable to ask what sort of risks are involved in using these methods. More specifically, what is the likelihood that the rating for some particular clique, for example, will be low? One possible way to answer this question is outlined below. For each of the two cluster types, compute an average rating for each cluster across subjects, split the rating scale into equal-sized intervals, and plot the distributions of frequency versus average rating for the hierarchical clusters and the cliques. Figure 4 shows these distribution plots.

As shown in Table 1, the average rating for the random or *no clustering* case was 2.92; hence it is reasonable to assert that any cluster whose rating does not exceed this value is a *bad* cluster. The average ratings for four (11%) out of the 36 hierarchical clusters did not meet this minimal rating; for cliques, only three (8.6%) did not meet or exceed this minimal rating, so the efficiency, or the ratio of good clusters to total clusters, is fairly high for both of these methods.

Conclusions

Earlier we examined an issue relating to network fit: The role of the edge weights after the network has been generated. For any network, it is possible to compute distance between any pair of nodes in two different ways. The first way is the graph-theoretic distance. The second method relies solely on edge weights; this is the distance that is computed by algorithms like Djikstra's Single Source Shortest Path or Floyd's All Pairs Shortest Path (see Aho, Hopcroft, & Ullman, 1974). Across all five domains and a wide range

of *q*, *r*, and *z* values, the distance matrices derived from network structure correlated much more highly with the original data than did distance matrices derived from edge weights. This suggests that the primary role of the edge weights is in creating the right structure for the network.

In this chapter, we also compared several different clustering methods: hierarchical, cliques, stars, and blocks. In addition, all of these were compared against a group of random connected subgraphs. One result is that all four nonrandom methods were significantly better than the random clusters and so have some value as clustering methods, although for the hierarchical clustering method this is not a surprise. Additionally, cliques appear to be the best graph-theoretic clustering method, followed by stars and then blocks.

The work described here represents only a beginning effort. While the results are encouraging, more and larger studies should be done using different domains, additional clustering methods, and more subjects.

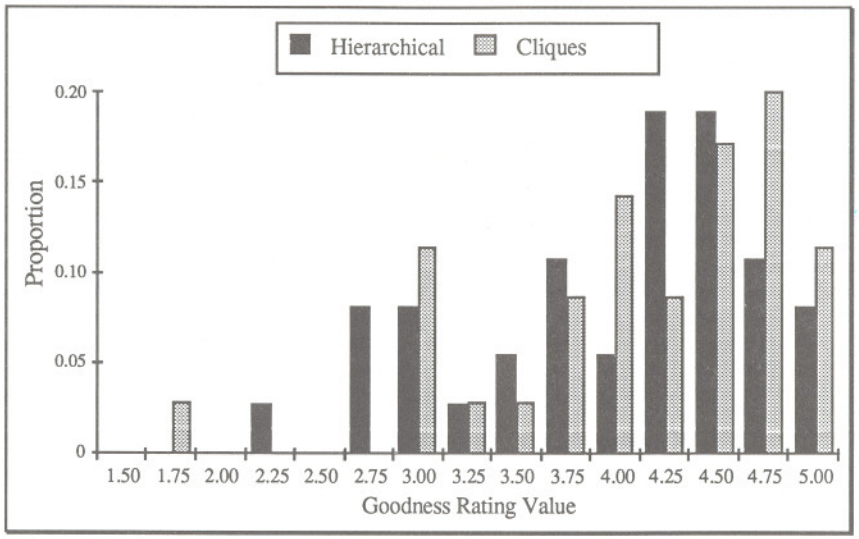

Figure 4. Frequency versus average rating for hierarchical clusters and cliques.