Spring 2003

Machine learning approaches for determining effective seeds for k -means algorithm

Kaveephong Lertwachara
Louisiana Tech University

MACHINE LEARNING APPROACHES FOR DETERMINING EFFECTIVE SEEDS FOR K-MEANS ALGORITHM

by

Kaveephong Lertwachara, B.Eng., B.S., M.B.A.

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Business Administration

COLLEGE OF ADMINISTRATION AND BUSINESS
LOUISIANA TECH UNIVERSITY

May, 2003

UMI Number: 3084539

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

LOUISIANA TECH UNIVERSITY

THE GRADUATE SCHOOL

May 2, 2003
Date

We hereby recommend that the dissertation prepared under our supervision

by Kaveephong Lertwachara

entitled Machine Learning Approaches for Determining Effective Seeds for K-Means Algorithm

be accepted in partial fulfillment of the requirements for the Degree of

Doctor of Business Administration

Supervisor of Dissertation Research

Dwight C. Anderson
Head of Department

Quantitative Analysis
Department

Recommendation concurred in:

Joe M.

Committee

Approved:
Director of Graduate Studies

Shirley P. Reagan
Dean of the College

Approved:
Director of the Graduate School

GS Form 13
(1/00)

ABSTRACT

In this study, I investigate and conduct an experiment on two-stage clustering procedures, hybrid models in simulated environments where conditions such as collinearity problems and cluster structures are controlled, and in real-life problems where conditions are not controlled. The first hybrid model (NK) is an integration between a neural network (NN) and the k-means algorithm (KM) where NN screens seeds and passes them to KM. The second hybrid (GK) uses a genetic algorithm (GA) instead of the neural network. Both NN and GA used in this study are in their simplest-possible forms.

In the simulated data sets, I investigate two properties: clustering performance comparisons and effects of five factors (scale, sample size, density, number of clusters, and number of variables) on the five clustering approaches (KM, NN, NK, GA, GK). Density, number of clusters, and dimension influence the clustering performance of all five approaches. KM, NK, and GK classify well when all clusters contain a similar number of observations, while NK and GK perform better than the KM. NN performs well when one cluster contains more observations than any other cluster. The two hybrid models perform at least as well as KM, although the environments are in favor of the KM. The most crucial information, the true number of clusters, is provided to the KM only. In addition, the cluster structures are simple: the clusters are well separated; the variances and cluster sizes are uniform; the correlation between any pair of variables and collinearity problems are not significant; and the observations are normally distributed.

Real-life problems consist of three problems with a known natural cluster structure and one problem with an unknown natural cluster structure. Overall results indicate that GK performs better than KM, while NK is the worst performing among the five approaches. The two machine learning approaches generate better results than KM in an environment that does not favor KM.

GK has shown to be the best or among the best in a simulated environment and in real-life situations. Furthermore, the GK can detect firms with promising financial prospect such as acquisition targets and firms with "buy" recommendation, better than all other approaches.

APPROVAL FOR SCHOLARLY DISSEMINATION

The author grants to the Prescott Memorial Library of Louisiana Tech University the right to reproduce, by appropriate methods, upon request, any or all portions of this Thesis/Dissertation. It is understood that "proper request" consists of the agreement, on the part of the requesting party, that said reproduction is for his personal use and that subsequent reproduction will not occur without written approval of the author of this Thesis/Dissertation. Further, any portions of the Thesis/Dissertation used in books, papers, and other works must be appropriately referenced to this Thesis/Dissertation.

Finally, the author of this Thesis/Dissertation reserves the right to publish freely, in the literature, at any time, any or all portions of this Thesis/Dissertation.

Author ____________________

Date 5/02/03

DEDICATION

To *The Lertwacharas:*

Yaowanit & Viroj (Mom & Dad),

Kaveepan & Kaveechok (Brothers),

Thammahatai & Thammajade (Motivators), and

Wilawan (Wife)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I am greatly indebted to my family, dissertation committee, teachers, mentors, friends, and colleagues. First, I would like to thank my dissertation committee members: Dr. James J. Cochran, my academic mentor and dissertation committee chairman, for his time, patience, advice, encouragement, and motivation; Dr. Joe M. Pullis and Dr. Otis W. Gilley, who are always available to offer invaluable suggestions and support in many ways. I also would like to thank Dr. Anita Pennathur who suggested the mutual fund classifications and Dr. Zaiyong Tang who introduced the genetic algorithm to me.

Secondly, I wish to express my gratitude to teachers who have educated and encouraged me, especially Dr. Hani I. Mesak for his words of wisdom, encouragement, and inspiration; Dr. Dwight Anderson for his support, and advice; Dr. Gene Johnson; Dr. Thomas L. Means; and Dr. Marc C. Chopin for providing support and opportunities.

I should also thank friends and colleagues, especially the staff in the College of Administration and Business; Ms. Linda Newbold, Ms. Brenda Sanderson, Ms. Sandra K. Nicklas, Ms. Sharon G. Hughes, Ms. Cherry D. Taylor, and Mr. Darrell Eddy, who kindly assisted me throughout my doctoral study.

I cannot conclude this section without expressing my appreciation to my family, especially my first and greatest teachers, my mom and dad. Without all of my teachers, mentors, friends, colleagues, and family, my name would not have been on the front cover of this document.

CHAPTER 1

INTRODUCTION

Since the beginning of human kind, humans have dealt with classifying objects into groups. Humans have to be able to make distinctions between edible and poisonous objects. They may not initially recognize that there are distinct groups of poisonous and edible objects. Nevertheless, once they experience sickness and death, humans learn that there are two groups (poisonous and edible) that comprise the objects. This classification process, in which the number of groups is not known prior to the classification process, is known as 'cluster analysis.' In cluster analysis a large number of objects are classified into a smaller number of meaningful groups based on pre-defined criteria. Therefore, a massive amount of information may be summarized so that it is easily understood and effectively employed. Everitt et al. (2001) state that cluster analysis is a collection of techniques that discover groups in data. These techniques are similar to discrimination methods except that clustering techniques do not generate any discriminating rule and the number of groups is not known prior to the clustering process. Although the techniques originated from biology, practitioners and researchers apply cluster analysis to various applications across disciplines including market segmentation, modeling economic prospects, price discrimination, information retrieval, and disease diagnosis. In market segmentation, for example, marketers can apply an appropriate marketing strategy to each market segment once the market is clustered based on various attributes such as age, gender, disposable income, and geography. The marketer can economically conduct experiments on how the market responds to a new product by taking a random sample from each market cluster in the entire market. Since the size of the sample is smaller than the size of the entire market, the marketer benefits from cluster analysis because of reduced costs and time associated with the market testing. In another application, Sinclair and Cohen (1992) use cluster analysis to uncover five clusters in the softwood industry in North America based on the technology adoption levels. The cluster structure they uncover also explains the softwood producers' profitability, investment intensity, and market share.

This study focuses on the clustering procedure known as the "k-means algorithm." A k-means algorithm begins with a pre-determined starting point, called a 'seed,' for each cluster. Observations are then aggregated into clusters based on their distances from the seeds. The k-means algorithm (KM) recalculates the center of the clusters every time an object is introduced to the group. Observations are then segregated according to their distances from these new centroids. The process is repeated until satisfaction of some decision rule. In addition, KM is designed for non-overlapping clusters (Milligan and Cooper, 1987; and Wedel and Kamakura, 1997). Moreover, MacQueen (1967) and Murty and Krishna (1981) state that KM is efficient in terms of storage requirements and computation time. Applications of KM appear in a wide range of disciplines such as biology, astronomy, image retrieval, data retrieval, economics, management, and market research. Church and Waclawski (1998), for example, investigate the relationship between personality orientation and executive leadership behavior utilizing cluster analysis. They find four distinct groups: innovators, analytical coordinators, implementers, and motivators. Their findings indicate that executives' personalities do relate to their leadership styles. Ng and Huang (1999) use KM to identify new classes of stars. Green, Schaffer, and Patterson (1988) analyze three real-world examples in market segmentation using a modified k-means algorithm.

KM is sensitive to the sequence of the data, specified number of clusters, and initial seeds. If the number of clusters is misspecified and/or the specified seeds are not close to the true cluster means, it is likely that KMs will not effectively and efficiently discover the latent cluster structure. The well-established practice is to use prior knowledge and/or associated theories to estimate the number of clusters and select the initial seeds randomly. Although this approach is expedient, there is evidence that randomly selected seeds are ineffective. Milligan (1980) finds that KM ineffectively identifies the latent cluster structure compared to the other clustering techniques when the seeds are randomly selected. However, his results also demonstrate that once the initial seeds are refined, KM effectively uncovers the latent cluster structure. Thus a random selection of the initial seeds for KM is not recommended (Wedel and Kamakura, 1997), and a need for determining effective initial seeds for KM arises. The process of searching for effective initial seeds for KM involves extensive computations, since computation requirements increase dramatically as the number of clusters, objects, and variables increase.

The recent development of computationally intensive approaches such as machine learning has raised interest in utilizing such approaches to identify the number of clusters and approximate the effective initial seeds for KM. Machine learning approaches adjust parameters computed in previous step until some goodness of fit criterion is met. Machine learning approaches involved in this study include artificial neural network (NN) and genetic algorithm (GA). Backer (1995) defines NNs as “computational models designed to generate performance similar to that of the human brain.” A NN adjusts parameters computed in the previous step based on a learning rule until either an objective function satisfies a pre-determined requirement or a significant improvement in the objective function does not present. On the other hand, GAs are heuristic optimization techniques that imitate genetic production using genetic operators to repeatedly manipulate members in the population, generation after generation, attempting to eventually reach an optimum. Machine learning approaches are nonlinear in nature; therefore, they do not require certain assumptions such as normality and homogeneity of variance, and they are flexible to a variety of forms of objective functions. In addition, Chiou and Lan (2001) state that GAs, in particular, do not require prior assumptions regarding cluster structure. Moreover, the authors also add that the type of variables and the number of variables used in the analysis do not severely affect the accuracy of the GAs but do affect the computation storage and time of GAs. Machine learning approaches have been used in a variety of applications such as analyzing credit card fraud, forecasting machine tool loading, capital markets analysis, crop forecasting, product marketing, and property tax analysis. Considering the weaknesses and requirements of KMs together with the computational ability of the machine learning approaches, it is conceivable that the machine learning approach may contribute a significant improvement to KM in recovering the latent cluster structure.

This study investigates clustering power that machine learning approaches contribute to KMs in both simulated problems and actual financial problems. The simulated problems are generated in accordance with an experimental designed to allow for investigations of the accuracy and the efficiency of the clustering techniques corresponding to different levels of various factors such as the number of attributes, number of groups in the data set, density level, sample sizes, and distances between clusters. The financial problems incorporate acquisition targets and corporate failures predictions, analysts' stock recommendations, mutual funds classification, and latent clusters discovery among dot-com companies. The financial problems are investigated through financial ratios. Machine learning approaches, including NNs and GAs, are employed to determine the number of clusters and their initial seeds to be used as initial values in KM. The clustering performances of KMs with and without assistance from machine learning approaches are compared for the simulated problems and of other benchmarks in financial applications in term of the correct classification rate. In summary, this study attempts to answer the following two research questions.

Research Question #1: Does the k-means algorithm with initial seeds from machine learning approaches outperform the k-means algorithm with random seeds?

Research Question #2: Do the accuracy of the tested clustering approaches differs across levels of the five previously discussed factors?

The academic literature regarding traditional clustering techniques, machine learning approaches, and hybrid models is reviewed in Chapter 2. In Chapter 3 we describe the process by which our simulated problems are generated and discuss the data collection processes for financial problems. The architectures of NNs and GAs are also discussed in this chapter. We report and discuss the results of our analyses on simulated data in Chapter 4. In Chapter 5 we present findings on the real financial problems. Finally, we summarize our results, report limitations of this study, and indicate direction of future research in Chapter 6.

CHAPTER 2

LITERATURE REVIEW

This chapter summarizes and discusses scholarly research in cluster analysis. In addition, this chapter also includes an overview of approaches related to cluster analysis. This chapter is divided into three sections. Section I provides an overview of traditional clustering techniques and summarizes relevant research in the literature. Section II includes an overview of machine learning approaches and a discussion of research in machine learning approaches to cluster analysis. Section III discusses hybrid models in cluster analysis.

2.1 Traditional Clustering Techniques

Cluster analysis (sometimes known as numerical taxonomy, grouping analysis, or unsupervised pattern recognition) is a multivariate procedure that organizes observations into a small number of relatively homogenous and meaningful groups. Generally, there are three types of clustering techniques: overlapping, non-overlapping, and fuzzy models. This study deals exclusively with non-overlapping techniques. These techniques only allow an observation to be in one cluster only. These techniques can be further divided into hierarchical and nonhierarchical methods. The following subsections include overviews of hierarchical and nonhierarchical clustering techniques.

2.1.1 Hierarchical Procedures

Rather than generate a set of clusters directly, these procedures produce a hierarchical tree representing relationships among observations based on a pre-determined measure of their similarity or dissimilarity. Researchers must use judgment in determining the cluster structure. Hierarchical procedures can be further grouped into two classes: agglomerative and disagglomerative. Agglomerative hierarchical procedures start with the maximum possible number of clusters (which is equal to the number of observations). In each proceeding iteration, the number of clusters is reduced by one. This reduction is accomplished by merging the two closest clusters. Obviously, at the last step only one cluster that includes all observations remains. On the other hand, disagglomerative hierarchical techniques (sometimes called divisive procedures) start with one cluster that contains all observations. The most dissimilar observation is then separated from others. This results in repeated formation of singular clusters. Therefore, each final cluster contains only one member. Neither type of hierarchical procedures requires a starting point, but they do require a desired number of clusters; otherwise, a stopping rule must be employed. The stopping rule can be derived from an index that can be classified as internal and external criteria. Internal criteria emerge during the clustering process, while external criteria require some additional information that is not used in the clustering process. The additional information can take a form of a separate data set or a variable that is not involved in the clustering process, or can be prior knowledge of the latent cluster structure (which is not practical for conducting cluster analysis using the real data sets). Milligan (1981) examines cluster recovery measures of 30 internal criteria based on their agreement with four external criteria. Milligan and

Cooper (1985) also investigate 30 internal criteria normally employed to develop stopping rules for the best number of clusters in the data set. The Calinski and Harabasz (1974) index and the Cubic Clustering Criterion (CCC) (Sarle, 1983) are found to be superior to other internal criteria (Milligan and Cooper, 1985). Both indices are automatically given by SAS's FASTCLUS procedure. A larger value of either of these indices indicates better cluster recovery by the clustering techniques. Milligan and Cooper also suggest that these internal criteria are also applicable for nonhierarchical clustering techniques although these internal criteria are examined via hierarchical clustering techniques.

Many studies have investigated cluster recovery achieved by various types of hierarchical techniques. Milligan et al. (1983) investigate the effects of cluster size, dimensionality, and the number of clusters on ability to recover the latent clusters for four hierarchical clustering methods. In addition to these three factors, four types of error perturbations are also included in the simulation. The performances are evaluated on the basis of four external criterion measures, including the Rand's (1971) index, corrected Rand (Childress, 1981), Jaccard statistics (Anderberg, 1973), and Fowlkes and Mallows statistics (Milligan et al., 1983). Their findings indicate that the best number of clusters, provided by the four external criteria, is negatively correlated with the ability to recover the latent cluster structure of the hierarchical clustering techniques given a fixed number of observations in the data set. On the other hand, the recovery ability increases as the number of relevant attributes increase.

Hierarchical clustering techniques effectively classify data regardless of clusters' shapes (Punj and Stewart, 1983). For this reason, applications of the hierarchical clustering procedure generally appear in variety of areas. Klastorin (1982) uncovers five clusters among short-term hospitals using a hierarchical clustering technique. These five clusters differ in terms of location, income of the local population, number of facilities and services, and average cost per case. Kamrani et al. (1993) and Biles et al. (1991) apply hierarchical cluster analysis to a problem of manufacturing design and find a significant contribution of cluster analysis in the efficiency of the new design manufacturing. Sinclair and Cohen (1992) investigate the effect of continuous technology adoption on profitability, investment intensity, and market share in the North American softwood industry. They find five clusters in the data set that suggested relationship between the continuous adoption of new technologies and market share and growth. Hofstede (1998) investigates subcultures in organizations and found professional, administrative, and customer interface subcultures. His results suggest that managers must clarify job classifications correctly in order to make appropriate assignments of personnel to jobs. Harvey (1986), Sackett et al. (1981), and Cornelius et al. (1979) also study job classification using hierarchical clustering procedures.

Despite the usefulness of the hierarchical clustering procedures, drawbacks of these techniques should be noted as well. First, the hierarchical clustering procedures are only applicable to qualitative data. Frequently, one must deal with both qualitative and quantitative data; under such conditions, hierarchical clustering techniques are of limited use. Second, once an observation is classified into a group by a hierarchical clustering procedure, the observation remains in that group throughout the process. If the observation fits better in any other group at any later stage, its membership is not changed. In addition, hierarchical clustering techniques are sensitive to outliers and irrelevant attributes (Punj and Stewart, 1983). Moreover, Murty and Krishna (1981) add that hierarchical clustering techniques involve high storage and computation requirements. As a result, practical application of hierarchical clustering techniques is limited to small sample sets. Furthermore, Milligan (1980) suggests that the hierarchical clustering techniques are sensitive to types of error perturbations in the data set, which include error-free (explain), outliers, distances, random noise dimensions, distance measurements, and standardization. Allowing for these weaknesses, the nonhierarchical clustering procedures are preferred to the hierarchical procedures (Murty and Krishna, 1981; and Punj and Stewart, 1983).

2.1.2 Nonhierarchical Procedures

Nonhierarchical procedures, sometimes referred to as the k-means algorithm (KM) or iterative partitioning methods, are generally preferred to hierarchical clustering techniques when the sample size is large and the data set includes at least one continuous variable (Wedel and Kamakura, 1997). KM begins with a pre-determined starting centroid, or seed for each cluster. Observations are then grouped on the basis of their distances from the seeds. In some nonhierarchical procedures, each observation is placed into the cluster with the nearest centroid and the centroids are recalculated after all observations are assigned to a cluster. In other nonhierarchical procedures, the centroids are recalculated after each observation is assigned to a cluster. In either instance, the clustering procedure is continues (using the new centroids) until some stopping criterion is met.

Wedel and Kamakura (1997) mention five dominant nonhierarchical methods: Forgy's method, Jancey's (1966) method, MacQueen's (1967) method, the convergence method, and the exchange algorithm of Banfield and Bassil (1977). Forgy's method and Jancey's (1966) methods recompute a new set of seeds after all observations are completely assigned. This procedure is repeated until there is no improvement based on an optimization criterion (such as minimizing the sum of squared Euclidean distances between members and their segment mean). MacQueen's (1967) method, the convergence method, and the exchange algorithm of Banfield and Bassil (1977) recalculate the seed every time an observation is merged. Unlike the convergence method and the exchange algorithm of Banfield and Bassil (1977), MacQueen's (1967) method ends after the first round of reallocating all observations; thus, MacQueen's (1967) method consumes the least time relative to the other four dominant nonhierarchical methods (Anderberg, 1973).

Research on KM appears in a wide range of disciplines. Slater and Olson (2001) perform KM on firms' marketing strategies and find four marketing strategies: aggressive marketers, mass marketers, marketing minimizers, and value marketers. They also find that firms perform well if specific marketing strategies are matched with specific business strategies. Barrett and Wilkinson (1985) apply KM to Australian manufacturing firms to eliminate problems in exporting their products and services.

Unfortunately, the nonhierarchical techniques are sensitive to the sequence of the data, specified number of clusters, and initial seeds (Murty and Krishna, 1981; Punj and Stewart, 1983; Milligan and Cooper, 1987; and Wedel and Kamakura, 1997). Furthermore, nonhierarchical procedures require a pre-specified number of clusters and starting points based on the desired number of cluster. These requirements usually cause the two most common problems in classification problems: incorrectly determining the numbers of clusters, and incorrectly assigning observations to clusters. The well-established practice is to initially use a hierarchical clustering technique, prior knowledge and/or associated theory to estimate the number of clusters. The initial seeds are then selected randomly. In addition, the nonhierarchical procedures tend to converge to local optima. Punj and Stewart (1983) summarize cluster analysis in marketing research. According to Punj and Stewart (1983), the purposes of clustering in marketing include market segmentation, buyers' behavior identification, and competitors' recognition. They indicate four issues dealing with using the cluster analysis: data transformation, desired number of clusters, validity, and variable selection. Data transformation does not affect the final outcome of cluster analysis except when a substantial correlation is present in the data set. Punj and Stewart (1983) point out that only when the initial seeds are specified nonrandomly and the number of clusters is correctly specified, KM demonstrated superior performance compared to the hierarchical clustering procedure. Therefore, they recommend a two-stage clustering technique where a hierarchical clustering technique supplies the number of clusters and the initial seeds to a nonhierarchical clustering technique. To verify the stability of the cluster solution, they suggest that it should be applied to a holdout sample for a cross-validation.

Milligan (1980), Hruschka and Natter (1999), Balakrishnan et. al. (1994), Green and Krieger (1995), and Krieger and Green (1996) make extensive performance comparisons between k-means and other clustering algorithms. Milligan (1980), in particular, examines the effect of six types of error perturbation on fifteen clustering techniques including KM. In addition to the six types of error perturbation, the experiment includes three factors: number of clusters, number of attributes, and distribution patterns. Milligan (1980) suggests that the Rand's (1971) index and the point-biserial correlation can be used as external and internal criteria when a comparison involves hierarchical and nonhierarchical clustering procedures because both indices are general and applicable for both hierarchical and nonhierarchical clustering procedures. The results reveal that KM is less successful, relative to hierarchical clustering techniques, in recovering the latent cluster structure and is ranked the worst among all clustering techniques in the framework. However, KM satisfactorily recoveres the latent cluster structure once the number of clusters and initial seeds are specified by a hierarchical clustering technique. Through Monte Carlo simulation, Helsen and Green (1991) and Murty and Krishna (1981) affirm Milligan's (1980) findings that the initial seed selection process does affect the clustering performance of KM.

Consequently, identifying effective initial seeds for KM is of interest to many researchers (Milligan and Cooper, 1987). However, the process is computationally intensive and intractable because of the high number of possible combinations for the initial seeds and its combinatorial character (Pinter and Pesti, 1991).

2.2 Machine Learning Approaches and Clustering

Many machine learning approaches have been applied to clustering problems. Machine learning is a computer system that learns from experiences. The data set passes through the system repeatedly, and the system evaluates its configuration on every repetition until a predetermined criterion is satisfied. Machine learning approaches, which include neural networks (NN) and genetic algorithm (GA), are computationally intensive and are expected to demonstrate promising clustering performances. In addition, NNs are generally less sensitive to dispersion level compared to traditional clustering algorithms

(Chen et al., 1995). The dispersion level measures the within-group variation (the higher dispersion level, the higher within-group variation). However, the results are inconclusive regarding this speculation when each of the machine learning approaches performs cluster analysis individually (Krishnamurthy et al., 1990; Balakrishnan et al., 1994; and Balakrishnan et al., 1996).

The performance of NNs has been shown to degrade as the number of clusters increases (Balakrishnan et al., 1994). Moreover, the NNs require tremendous amounts of computational time (Tam and Kiang, 1992). Balakrishnan et al. (1996) also add that the NNs are sensitive to number of attributes and error levels, where "error" represents data collection and measurement error that may cause a missclassification. The following subsections provide a general background of NNs and GAs and discuss research regarding the use of NNs and GAs in clustering.

2.2.1 Cluster Analysis and NNs

NNs mimic a mechanism of the brain (Hecht-Nielsen, 1990). A NN consists of at least two layers: input and output. Any layer between input and output layers is called a hidden layer. As many hidden layers as desired may be inserted between the input and output layers. Each layer consists of a number of processing units. These processing units are called neurons or nodes and are computing devices. Each neuron in the hidden layer receives inputs from other neurons in the previous layer and sends outputs to neurons in the next layer. Each signal, either input or output, is multiplied by a weight before it is passed on to the next layer. Upon receiving the weighted inputs or signals from neurons in the previous layer, each neuron applies a function (called an activation function) to these signals. The numbers of nodes in each layer need not to be equal. NNs learn a cluster structure from a training data set by adjusting weights for each node in the network to fit the data on a basis of either external or internal measurements. An example of a fully connected (explain) NN is illustrated in Figure 1. All nodes in a layer are connected to all nodes in the previous and following layer.

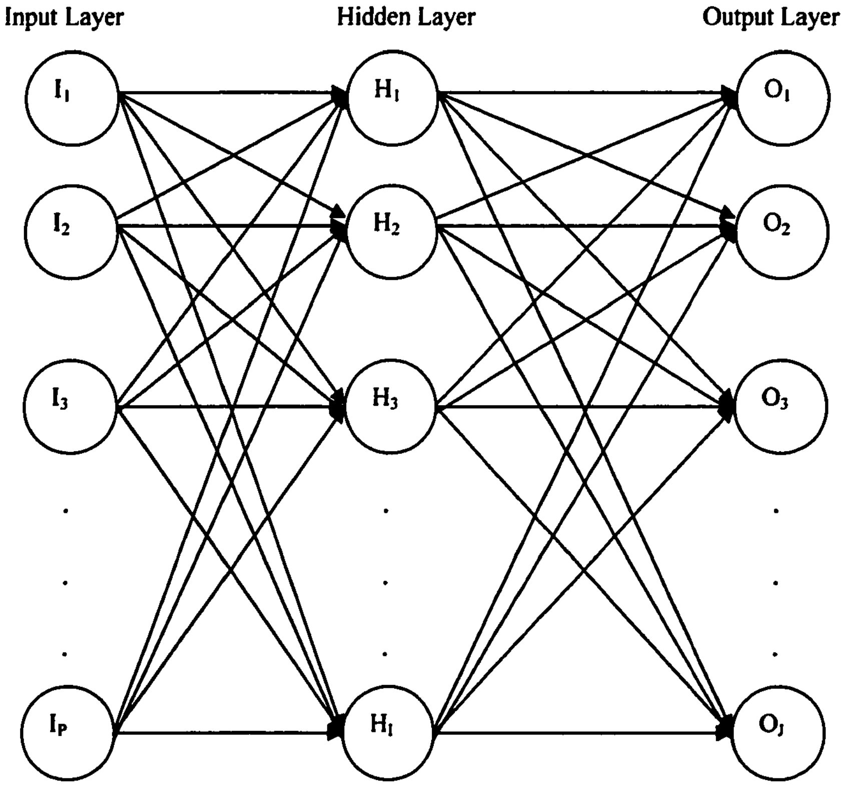

Figure 1: Fully Connected NN

NNs can be categorized on the basis of how they monitor their output and how the data flow through them (Garson, 1998). Based on how the networks monitor their results, NNs can be classified into two groups: supervised and unsupervised. Supervised NNs compare their results with target outputs or actual outcomes. These NNs adjust their weights until a measurement of the differences between the results and the targets falls within a preset tolerance level. Conversely, unsupervised NNs learn from the data set as each observation is fed into the network without comparing their results to a target output.

Based on the direction of the data flow, NNs can be also divided into two groups: feedforward and feedback (Chester, 1993). The data flow through the network once at each round for feedforward NNs. The weights are adjusted as observations pass through using only the information of the current observation. Signals are sent in one direction from input layer to the output layer through the hidden layer, if any exist. The signals do not travel from the later layer back to the earlier layer. On the other hand, the data are circulating in the networks for feedback NNs. The signals can travel back and forth between layers. The weights are adjusted using both the current and previous observations. The number of circulations and how the data circulate depend on the architecture of the networks.

Many NN approaches, including backpropagation (BP) and the self-organizing map (SOM), have been applied to clustering problems. BP can be designed to be either a feedback or a feedforward NN. BP generally uses mean square error and gradient descent to determine the fitness of its predictions and has at least two layers: input and output. The connections are designed based on the objective of researchers. SOM, first introduced in early 1980s by Kohonen (2001), is generally used to (1) classify a data set, (2) establish clusters of different variables, (3) reduce a larger input vector to a smaller number of clusters, and (4) solve routing problems such as the traveling salesman problem (Ritter et al., 1992). In the input layer, the number of neurons is the same as the number of the input variables, and each of these neurons is connected to all neurons in the next layer. In the next, Kohonen, layer, the number of neurons must be at least equal to the number of observations. Each observation belongs to the nearest neuron. Once an observation is assigned to a neuron, the weight or the location of the neuron is recalculated. The recalculation process is referred to as the Kohonen's learning rule.

The use of NNs in cluster analysis has been investigated by a number of researchers. Tam and Kiang (1992) apply NNs to bankruptcy predictions problems and suggest that NNs are superior to linear discriminant analysis because (1) the potentially non-linear function produced by a NN is suitable to a multi-modal data set, (2) NNs are capable of adaptively adjusting the model according to a change in the real-world data, and (3) NNs do not assume any probability distribution and do not require any specific form of input or output. Nonetheless, Tam and Kiang (1992) add that there are some disadvantages in NNs. First, there is no formal procedure in configuring the network. Second, NNs require a tremendous amount of training time. And finally, a symbolic form of the NN is complicated. The authors also suggest combining the NN with other algorithms.

Balakrishnan et al. (1994) apply two types of NNs: Kohonen's learning rule, both with and without conscience, to a simulated data set which is generated following guidelines of Milligan (1980, 1983, and 1985). They define a NN with conscience as a NN that adjusts not only weights of the winning node but also weights of the nodes surround it. Balakrishnan et al. (1994) compare the results to a k-means algorithm's performance. The results of their study indicate that KM generates less misclassification than the two types of NNs. Furthermore, Balakrishnan et al. (1994) add that the performance of NNs worsens as the number of clusters increase.

Krishnamurthy et al. (1990) introduce the frequency-sensitive competitive learning algorithm (FSCL). The FSCL is a modified version of SOM with a penalty applied to the winning node if it wins too often. The weights of the winning node are adjusted to the opposite direction it should be. For example, if the winning node has a value of 5 and the corresponding value is 7, the value of the winning node would change from 5 to 4 or 4.5 rather than 6 or 5.5. The FSCL and SOM perform well with vector quantization (VQ) of speech and images. As described by Kohonen (2001), VQ is a classical signal-approximation method that forms sets of vectors, which are usually called a codebook, to represent the input data vector in the learning phrase. Then the closest vector in the codebook as measured by the Euclidean metric will represent a new input vector. Subsequently, the vector from the codebook will then be transmitted or processed.

NNs have been compared to many traditional clustering algorithms, especially KM. Many researchers have found inconclusive results regarding comparative performance between KM and NNs but a hybrid between KM and NN is usually found superior to either basic approach and is recommended. Balakrishnan et al. (1996) compare performances of FSCL and K-means algorithms using real data and simulated data generated as prescribed in Milligan (1980, and 1985). On the other hand, the real data used in their study incorporate brand choice data in the coffee industry. With simulated data, FSCL's performances are very sensitive to the number of clusters, number of attributes, and error levels, while KM's performances are only sensitive to error levels according to their analysis of variance. With the brand choice data set, the

FSCL provides clusters with similar sizes and high interpretability. However, the FSCL misclassifies members more frequently than KM for the brand choice data. Therefore, they hypothesize that a combined approach might provide a superior cluster solution (in terms of frequency of misclassified observations). The starting seeds for KM are estimated by FCSL. The performances are in between the performances of the FSCL and KM in terms of their interpretabilities and the similarities of cluster sizes. Balakrishnan (1996), therefore, recommends an investigating a hybridization of a NN and a clustering algorithm.

Chen et al. (1995) compare SOM to seven traditional clustering algorithms: single linkage, complete linkage, average linkage, centroid method, Ward's minimum variance, two-stage density linkage, and the Kth-nearest neighbor density linkage. Data sets used in the comparison are randomly generated and varied on four factors: number of clusters, number of variables, relative dispersion within the clusters, and number of observations. The results indicate the SOM is superior to conventional classification algorithms, especially at relatively high levels of dispersion.

Hruschka and Natter (1999) compare a feedforward NN to KM for cluster-based market segmentation. Hruschka and Natter (1999) also analyze the usages of brands of household cleaners in different situations. In their study, a feedforward NN outperforms KM based on the Davies-Bouldin index (Davies and Bouldin, 1979). The NN suggests a two-cluster structure, while KM fails to recover the latent cluster structure on the basis of an external criterion. Hruschka and Natter (1999) also suggest that researchers should consider using feedforward NNs in cluster analysis.

Cinca (1996) complements the SOM and compares it with multivariate statistical models and a multilayer perceptron NN in a framework of financial diagnosis. He reports that the results of the integrated SOM are compromising. In his study, Cinca finds SOMs to be superior to linear discriminant analysis and the multilayer perceptron NN. He also suggests that an integration of NNs with a statistical approach or another machine learning approach would be a very powerful tool.

Results from previous studies are somewhat mixed. In many occasions, NNs are found to be superior to traditional clustering procedures. However, there is evidence that the opposite is true. This discrepancy may be the result of deviation in network architecture. Furthermore, NNs (like other heuristic approaches) are not effective in finding a global optimum (Pinter and Pesti, 1991).

2.2.2 Cluster Analysis and GAs

GAs emulate genetic production in a search for solutions to optimization problems (Holland, 1992). Members of each generation are usually called chromosomes and represent a feasible solution to the problem. Each chromosome consists of basic elements that are referred to as genes. As described by Goldberg (1989) and many others, the processes of GAs are as follows. The initial generation is usually randomly selected. Consequently, members of the initial population are randomly selected to be parents of the next generation with probability of selection based on the member's success in the first generation (a member with a higher evaluating value based on some pre-determined criteria has a higher likelihood of selection). Then the selected chromosomes pass through one or more processes of crossover, mutation, and inversion. Crossover is simply a process of swapping parts of the two selected chromosomes. Figure two illustrates a simple example of a crossover. A crossover point is randomly selected. Then, all genes behind the crossover point of the two selected chromosomes are swapped. Mutation deviates randomly selected genes. Figure three shows how genes are mutated. First, target genes are randomly selected. Then values of the selected genes are changed. Inversion flips the series of genes. Figure four demonstrates a basic inversion operation. First, the GA randomly selects a series of genes. Then the series of selected genes will be reversed. The new chromosomes are substituted for chromosomes with low evaluating values from the previous generation. Thus, the new generation consists both of chromosomes with high evaluating values and new chromosomes. The process is repeated until the improvement in the evaluating value is less than some pre-determined value.

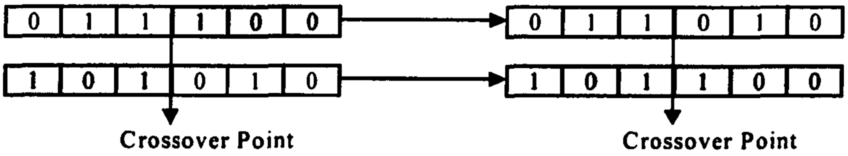

Figure 2: Crossover

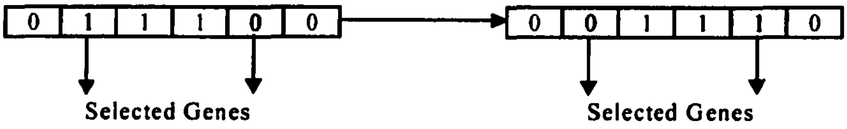

Figure 3: Mutation

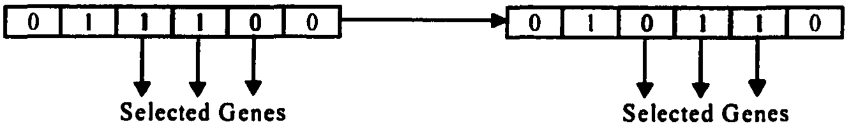

Figure 4: Inversion

GAs are also widely compared to and combined with NNs to solve classification problems (Faulkenauer, 1998). Varetto (1998) performs comparisons between GA and linear discriminant analysis (LDA). The data consists equally of insolvent and solvent firms. LDA yielded a slightly better discriminant rule but consumed more time than GA did. Sexton and Dorsey (2000) configure three GAs and three BP NN models. The six machine learning models are examined in ten different real-world data sets referred to as cancer, card, diabetes, gene, glass, heart, heartc (heart data set without incomplete observations), horse, soybean, and thyroid. Cancer data are originally generated at the University of Wisconsin, Madison by Dr. William H. Wolberg. Card, Diabetes, Gene, Glass, Horse, and Soybean were from the UCI repository of machine learning databases. Heart and Heartc data sets are obtained from four sources: Hungarian Institute of Cardiology, Budapest, Andras Janosi, MD; Univeresity Hospital, Zurich, Switzerland, William Steinbrun, MD; University Hospital, Basel, Switzerland, Matthias Pfisterer MD; and V.A. Medical Center, Long Beach and Cleveland Clinic Foundation, Robert Detrano, MD, Ph.D. A source of the Thyroid data set is not identified. All three GA models outperform each of the NNs except in the Horse data set, where the GA models are ranked first, second, and fourth in terms of an average classification error percentage.

GAs are recognized for their ability to locate a global optimum. GAs have been widely used in applications in many areas such as improving performance of NNs, designing intelligent production lines, identifying images, and predicting stock market movements. Performances of GAs are generally found to be excellent in previous studies. Chiou and Lan (2001) investigate clustering abilities of three configurations of GAs by comparing the clustering performances of the GAs to an agglomerative hierarchical clustering method. The results indicate that the GAs perform better than the hierarchical clustering techniques when the sample size is medium to large. However, GAs included in their study require tremendous storage space relative to the hierarchical clustering technique. Finally, the authors also recommend a hybrid model between a GA and other traditional clustering techniques.

2.3 Hybrid Models

The hybrid systems are expected to eliminate weaknesses and capture strengths of both KM and the machine learning algorithms. Machine learning algorithms are demonstrated to be more accurate than traditional clustering algorithms in many studies (Kattan and Cooper, 1998; Cinca, 1996; and Hruschka and Natter, 1999). In addition, the machine learning approach can identify and ignore influential variables. Moreover, the machine learning approaches do not require assumptions that must be met when using traditional clustering analyses. However, the machine learning approaches consume more time and storage than traditional clustering procedures. Another drawback for NN algorithms is that they are sensitive to cluster sizes in the same data set (Balakrishnan 1996). If all the clusters have approximately equal size, NNs generally perform better than KM. Moreover, results from NNs tend to be unstable and easily converge to local optima if the sequence of the data changes. On the other hand, GAs are insensitive to the data sequence and usually discover a global optimum.

Hybrid models between hierarchical clustering techniques, KM, and machine-learning approaches have also been tested. The integrated approaches are found to be superior to individual procedures (Milligan, 1980; Milligan and Sokol, 1980; Murty and

Krihna, 1981; Punj and Stewart, 1983; Wong and Lane, 1983; Scheibler and Schneider, 1985; Milligan and Cooper, 1987; Lee et al., 1998; Lee et al., 1996; and Markham & Ragsdale, 1995). Murty and Krihna (1981) report satisfactory performances of a hybrid model between the MacQueen's k-means algorithm and a hierarchical clustering technique for concentric and chain-like clusters in terms of accuracy, computation time, and storage requirements. KM performs cluster analysis in the first stage and then passes the seed points to a hierarchical clustering technique. Data used in Murty and Krihna's (1981) study are generated manually in the two-dimensional Euclidean space to form a concentric and a chain-like cluster. The results reaffirm the notion that integrated approaches are superior to individual clustering techniques.

Lee et al. (1998) examine a combined (traditional clustering algorithm and NN) approach. The performance of the integrated procedure is promising in the context of software development cost estimation. Five data sets are randomly generated from the total data sets of software development costs and fed into the clustering algorithm. The clustering analysis identifies a data set that produces the smallest error rate. This information is then passed to the NN in phrase two. Thereafter, five NNs with different configurations are tested. The best configuration is applied to the NN. Finally, a comparison between the NN and the combination of NN and the clustering analysis is examined using four different testing cases. The combined approach adds significant improvement to the NN approach.

Lee et al. (1996) also use hybrid NN models in a framework of bankruptcy predictions. Their study includes SOM, multivariate discriminant analysis (MDA), and induction of decision tree (ID3) (Quinlan, 1986). MDA is similar to LDA except that

MDA is used only when more than one attribute is incorporated in constructing a discriminant rule. ID3's computational time increases only linearly with an increase in the number of observations and attributes. However, the decision tree must be rebuilt entirely upon a new observation available. The hybrid SOM and MDA model outperform other models: MDA, ID3, MDA-assisted NN, ID3-assisted NN, and a hybrid SOM and ID3 models.

Markham & Ragsdale (1995) combine Mahalonobis Distance Measures (MDM) with BP into a supervised multilayer feed-forward NN. Three approaches are compared on two types of data sets: oil quality and bank failure. Similar to the jackknifing procedure, each type of data set is replicated thirty times. The hybrid approach discriminates on the average better than either MDM or NN does individually. The hybrid model produces a smaller average rate of misclassifications than does MDM and NN at a 0.005 significance level.

2.4 Summary

From the literature reviewed in this chapter, one can draw several conclusions. First, hierarchical clustering techniques are only suitable for small data sets with qualitative variables because of their high computation and storage requirements (Murty and Krihna 1981). Second, KM executes cluster analysis better than hierarchical clustering techniques with large sample that include at least one quantitative variable. Third, KM performs cluster analysis poorly if the initial seeds are incorrectly specified; therefore, this condition necessitates effective initial seeds for successful use of KM. Fourth, researchers have investigated the utility of the NNs in classification and cluster analysis problems and find inconclusive results whether or not the NNs cluster data better than traditional clustering techniques. Fifth, studies of application of GAs to cluster analysis problems provide promising results; however, the number of references is limited. And finally, researchers widely agree that the initial seeds developed by other clustering techniques improve clustering performance of the k-means technique.

CHAPTER 3

DATA AND PROCEDURE

Milligan and Cooper (1987) identify three strategies in validation techniques for cluster analysis: mathematical derivation, simulation analysis, and analysis of empirical data sets. They also indicate that the mathematical derivation has often been complicated and provides limited value for applied analyses in the area of cluster analysis. Therefore it is disregarded in this study. Consequently, experiments in this study incorporate two types of analyses: simulation analysis and analysis of empirical data sets. The sections of this chapter proceed as follows. The first section discusses experimental design, data simulation, and clustering techniques used in simulated problems. The second section discusses empirical problems and related data collections, variables, and methodology.

3.1 Simulation Analysis

According to Milligan and Cooper (1987), there are three steps in the simulation analysis: data generation, cluster analysis using clustering techniques of interest, and verification of the cluster results. In this section, data generation, experimental design, and clustering techniques used in this study are discussed.

3.1.1 Data and Experimental Design

Simulated data sets are generated following guidelines in Milligan (1985). Milligan's simulation procedure has been used in several references including Balakrishnan et al. (1994, 1996), Chen et al. (1995), Milligan (1980, 1981a, 1981b, 1985), Milligan & Cooper (1985, 1987), Milligan & Sokol (1980), Milligan et al. (1983). The data are simulated using SAS version 8 because the SAS can simulate data sets, perform the k-means algorithm (KM), neural network (NN) and genetic algorithm (GA). The data are also normally distributed in Euclidean space. Since KM is designed for uncovering non-overlapping cluster structure, clusters must not be overlapping. Thus, cluster seeds are randomly selected except on the first dimension so that the clusters can be controlled to be non-overlapped at least on the first dimension. In order to generate data that possess these characteristics, we follow the simulation process suggested by Khattree and Naik (1999) using the following equation:

$$Y = XG + M$$

where Y is the matrix of the simulated data ranges from 0 to 10.

X is a matrix of random variables that follows the multivariate normal distribution with the means and standard deviation of 0 and 1.

G is a root matrix of a diagonal variance-covariance matrix.

M is a matrix of variable means.

The standard deviations for all variables in matrix G are set to be 1.00 except for first variable, which is a controlled variable. The standard deviation of the first variable equals 0.09; therefore, there are sufficient separations between clusters on this variable (non-overlapping clusters). Although the data are simulated through a diagonal variance-covariance matrix, there is no guarantee that the collinearity problem will not exist because of the randomization in the process. We can only assure that the problem is not substantial. It is worth noting that well-separated clusters and minimal-collinearity data are conditions in favor of KM.

Three basic and two hybrid approaches are tested on five factors: number of clusters, density, dimension, proximity, and sample size with two replications per cell. The numbers of latent clusters are 2, 3, and 7. Three levels of density are 0%, 20%, and 60%, where 0% density represents equal cluster size. A density level of 20% indicates that 20% of all observations are in one cluster and the remaining 80% of all observations are equally assigned to the remaining clusters. In the same manner, 60% density designates 60% of all observations into one cluster and the remaining 40% of all observations into the remaining clusters equally. Effects of dimension are tested on three levels: 3, 5, and 7. Proximity or relative distance between clusters includes three levels: 1, 1.5, and 2 standard deviations from the groups' means. The data are truncated on the first dimension at 1 standard deviation from the mean at the relative distance level of 1. In the similar manner, the data are truncated for the relative distance level of 1.5 and 2 at 1.5 and 2 standard deviation from the cluster means on the first dimension. Two levels of sample size are 210 and 420 observations. Thus, 324 data sets (3 levels of number of clusters, 3 levels of density, 3 level of dimensions, 3 levels of proximity, 2 levels of sample size and 2 replications) are analyzed. In addition to clustering performance comparisons, an analysis of variance (ANOVA) is performed to evaluate the impact of the five factors on each of the clustering approaches.

Table 1: Correlation Matrix

	CORRELATION							TOLERANCE
	X1	X2	X3	X4	X5	X6	X7	
X1	1.00000	-0.0063 0.7598	-0.0140 0.4976	-0.0091 0.6583	-0.0156 0.4509	-0.0034 0.8688	-0.0213 0.3013	0.99895
X2	-0.0063 0.7598	1.00000	-0.0183 0.3694	-0.0075 0.7169	-0.0052 0.7996	-0.0104 0.6145	-0.0332 0.1073	0.99829
X3	-0.0140 0.4976	-0.0183 0.3694	1.00000	0.0138 0.5040	-0.0096 0.6424	0.0067 0.7458	-0.0013 0.9484	0.99912
X4	-0.0091 0.6583	-0.0075 0.7169	0.0138 0.5040	1.00000	0.0023 0.9113	-0.0176 0.3928	0.0106 0.6088	0.99926
X5	-0.0156 0.4509	-0.0052 0.7996	-0.0096 0.6424	0.0023 0.9113	1.00000	0.0150 0.4682	-0.0180 0.3820	0.99907
X6	-0.0034 0.8688	-0.0104 0.6145	0.0067 0.7458	-0.0176 0.3928	0.0150 0.4682	1.00000	-0.0246 0.2340	0.99870
X7	-0.0213 0.3013	-0.0332 0.1073	-0.0013 0.9484	0.0106 0.6088	-0.0180 0.3820	-0.0246 0.2340	1.00000	0.99738

Three data sets are generated for three levels of proximity: 1, 1.5, and 2. The characteristics of the three data sets are similar except for that the values of the first variable vary based on where the data are truncated. For, example, the proximity of 1 indicates that each cluster is truncated at 1 standard deviation from the cluster means. Table 1 illustrates correlation matrix of the seven simulated variables used in this study where numbers on the top line are Pearson correlation coefficients and the numbers on the second line are p-values for each row. Based on the correlation matrix, we do not detect a serious correlation between any pair of variables. Table 1 also provides evidences of acceptable correlations in the data set. The tolerance levels are higher than 0.990 for all variables, which indicates that no variable can be explained by a linear combination of all other variables in the analysis.

3.1.2 Methodology

KM, two machine learning approaches, and two hybrid models are compared on the basis of their ability to maximize the within-group variance and are evaluated using SAS version 8. The FASTCLUS procedure in SAS is used as KM. KM is allowed a maximum number of iterations of five hundred and randomly selects seeds from the data set. Furthermore, KM is given the true number of latent clusters. These configurations should provide an optimal condition for KM. NN is coded by modifying SAS code implemented by Sarle (1994). The GA is implemented using data step and macro commands in SAS. The configurations of NN and GA are discussed in the following sections. Finally, the two hybrid models are combinations of KM and machine learning approaches. Each of the machine learning approaches is used to identify starting seeds and numbers of clusters for KM. The results of the machine learning approaches are used as "fine-tuned" starting points for the k-means. These two hybrid models are referred to as NN-assisted k-means (NK) and genetic-assisted k-means (GK). The comparisons are presented in terms of correctness of cluster recovery and the rank of all methods in each scenario.

NN, a fully connected feedforward neural network as shown in figure 1 with only one node in the output layer, consists of three layers: input, hidden, and output layers. This NN is modified from Sarle's (1994) NN. Sarle (1994) provide a prototype of a simple supervised NN using SAS's PROC NLP while the NN in this study is unsupervised NN. The number of input nodes is exactly the same as the number of attributes. The number of nodes in the hidden layer is equal to the number of desired or expected clusters. Once the initial weights are randomly selected, the observation is fed into the system through the input layer. The hidden layer applies a logistic function (sometimes called "softmax function") to the observation:

$$Z_{ij} = e^{Y}{}_{ij}$$

where Y_{ij} is a function of attributes for the j^{th} observation at the i^{th} hidden node. The output layer then transforms Z_i into a probability using the multinomial logistic function:

$$Prob_{ij} = Z_{ij} / \sum(Z_{ij})$$

where $Prob_i$ represents the probability that the j^{th} observation belongs to group *i*. The output layer applies a competitive rule allowing the competitive node with the highest probability to win and assigns the observation to the winning node (which represents a cluster). The procedure is repeated until there all observations are assigned. As with KM, we allow the NN a maximum of 500 iterations. Bentz and Merunka (2000) configure similar architecture (except that their NN is a generalized form of the multinomial logistic function) that they refer to as a "NN with softmax output." The distinction between the NN1 and the multinomial logistic function is that the multinomial logistic function is a function for classification problems where group memberships are known before a clustering process begins.

Our GA starts with 10 chromosomes in the first generation. Each chromosome, which represents a possible solution for the cluster structure, comprises clusters' means. Therefore, each chromosome consists of *pxk* elements, where *p* is the number of attributes and *k* is number of clusters. Accordingly, the fitness value (the multiplicative inverse of the sum of square error) for each chromosome is calculated and compared. Parents of new chromosomes in the next generation are selected through "the roulette-wheel selection" where opportunity to be selected is determined by the fitness value. A chromosome with a high fitness value is assigned a higher likelihood of being selected as a parent for new members in the next generation. This process is also called "mating." The reproduction process incorporates crossover, mutation, and inversion. A crossover procedure randomly mates two chromosomes, where the probability of being selected for each chromosome is calculated based on its fitness value. Then two crossover points are randomly selected. Next, the two chromosomes are swapped between two crossover points. This generates two new chromosomes that will replace the worst least fit chromosomes from the current generation. Mutation points are randomly selected at a rate of 10% and mutated based on the range of the variable. For example, we have $p \times k$ genes; so we selected 10% of $p \times k$ for mutation. If the selected gene has a value of 2.74 on a variable that ranges from 0 to 10, then the gene takes 10 - 2.74 = 7.26 as a new value. Subsequently, a series of genes are randomly selected and inverted at a 10% rate. The reproduction process is repeated until at least eight chromosomes with the same fitness value are present in the same generation or until the maximum number of iterations is reached.

Unlike KM and NN, the GA is only allowed 50 maximum iterations because it is extremely slow (Chiou and Lan, 2001). The GA can also stop if at least eight out of ten chromosomes indicate approximately similar fitness value. In another word, if seven other chromosomes provide fitness values within 300 units of the best fitness value, the system can stop.

3.2 Analysis of Empirical Data

Our analyses of the empirical data sets also consist of two parts. We first compare cluster structures recovered by the k-means, NN, GA, NK, and GK to the observed cluster structure within finance applications (acquisition target and bankruptcy predictions, mutual fund classifications, and analysts' stock recommendations). We then attempt to discover latent clusters among dot-com companies based on various types of financial information using NK and GK. The following subsections discuss the data gathering process for each financial application. Data used in these financial applications are typically not normally distributed and are likely to contain outliers. Therefore, it is reasonable to speculate that considerable improvement in clustering power would occur when using machine learning approaches. Clustering results are compared to the actual outcome. The results from the hybrid models are expected to be more accurate than the results provided by KM, NN, and GA.

3.2.1 Acquisition Targets and Bankruptcy Predictions

In recent years the economy has fluctuated dramatically. As a result, many companies have been acquired in a bull market and many others fail in a bear market. In explaining these two events, Jain and Kini (1999) point out that a company can reach three different stages for a given period: remaining an independent firm, going out of business, or being acquired. Corporate failures not only cause economical and social losses to the community but also to the management, stockholders, employers, customers, and others (Sung et al., 1999). The prediction of a corporate failure can be an early warning sign to regulators, management, investors, and stakeholders. When such a prediction occurs, corrective actions such as regulations, problem eliminations, immunizations, protections, and improvements can be implemented (Tam and Kiang, 1990). Barnes (1990) suggests that the prediction of corporate failure alone is worthy of research, but that forecasting a merger target is even more desirable. Dietrich and Sorensen (1984) add that a merger decision characterizes a form of investment decision. The net present value of the acquisition should dominate other investment alternatives for the acquiring firm. The increased wealth contributed to shareholders, especially the shareholders of the acquired firm, mainly arises from the synergy of the acquisition (Hanson, 1992). The predictions of the three outcomes are not only helpful for society and stakeholders but also for investors to speculate, analyze, and diversify their portfolios. In addition, a technique that can predict bankrupted firms and merger targets would enable investors to avoid poor investment decisions and to improve their return on investment. These potential benefits motivate the investigation of clusters in the publicly traded companies, where three clusters represent the three potential outcomes (acquired, bankrupted, and independent firms). Predicting the three outcomes requires consideration of two sets of models: merger targets prediction and bankruptcy prediction models studied by various researchers such as Altman (1968), Palepu (1986), Cheng et al. (1989), Tam and Kiang (1992), and Altman et al. (1994).

This study incorporates commonly considered financial ratios, which include sales per total assets, leverage, working capital per share, EBIT, dividend payout, and price-to-book value. The sales, leverage, and market-to-book value tend to undermine the possibility of being acquired. On the other hand, growth rate, liquidity, earnings, and payout ratio appear to enhance the probability of getting acquired (Palepu, 1986; Allen and Cebenoyan, 1991; Ambrose and Megginson, 1992; Thompson, 1997; and Cudd and Duggal, 2000). Kane et al. (1998) indicate also that the possibility of going bankrupt is positively correlated with leverage and payout ratio. Focusing on the effects of the leverage variable, one might realize that the variable positively correlates to both possibilities of being acquired and going bankrupt. Therefore, one might conclude that analyzing the leverage variable alone is not sufficient to reach a conclusion on which stage the company might arrive.

Data for the year 2000 are collected from Compustat's Research Insight. The bankruptcies and acquisitions occurring in 2000, 2001, and 2002 are considered. A random sample is drawn from mining or manufacturing companies (SIC: 1000-3999). I exclude biotech and pharmaceutical companies (SIC: 28XX), electronics and telecom (SIC: 3653-3689), and computer and technology (SIC: 357X) because they exhibit different characteristics (so, that the clustering algorithms do not classify these firms based on the industry). Companies in regulated industries such as financial services and utility providers, which operate under different environment and regulation, are also ignored.

3.2.2 Analysts' Stock Recommendation

Analysts' recommendations have significant impact on individual investors since many investors rely on analysts' recommendations. Givoly and Lakonishok (1984) praise stock recommendation as the most notable output of financial analysts. Stanley, Lewellen, and Schlarbaum (1980), Francis and Soffer (1997), and Gilson (2000) have indicated that individual investors trade stocks according to analysts' recommendations. Barber and Loeffler (1993) and Hirchey, Richardson, and Scholz (2000a, 2000b) also report significant abnormal returns and volumes following "buy" announcements. Hence, analysts' recommendations become interesting subjects to be investigated through empirical evidences.

Although there are substantial bodies of research regarding analysts' recommendations, determinant variables used to derive a recommendation have scarcely been revealed. Previous studies focus mainly on abnormal returns following the announcements of the recommendations. For example, Hirschey et al. (2000) examine the effects of online recommendations on stock price. They find significant increases on "Buy" and decreases on "Sell" recommendations (after an announcement of the recommendations). Barber and Loeffler (1993) provide descriptive characteristics of four portfolios: pros' picks, dartboard stocks, S&P 500, and NYSE firms. These four portfolios are compared in terms of growth, dividend yield, PE ratio, monthly volume, and beta.

Data for the year 2000 are collected from Compustat's Research Insight . Two hundred and ten companies are randomly drawn from firms in the non-regulated industries and the observed recommendations are used as an external criterion. The cluster analysis, in this problem, incorporates five variables: size, dividend yield, PE ratio, monthly volume, and beta.

3.2.3 Mutual Fund Classification

Recent policy changes regarding retirement plans and social security benefits have resulted in increasing popularity of mutual funds (among other investment vehicles) because of their diversification and cost efficiency. Mutual funds are classified on the basis of their stated objectives. These objectives are also related to the fund managers' investment styles and different level of risk. In other words, a fund with a higher aggressiveness investment style is associated with a higher level of risk relative to a fund with a less aggressive style. These funds' objectives must be stated and should accurately reflect investment styles of the funds' managers so that investors can choose funds to invest in based on preferred objectives and risks. A fund misclassification occurs when investment style of the fund's manager is inconsistent with the stated objectives. Kim et al. (2000) point out that misclassifications are sometimes intentional because of the competitiveness within the mutual funds industry. However, the danger of funds misclassification becomes higher as social security benefits diminish and people increasingly invest their savings and retirement funds in these mutual funds.

This part of the analysis examines whether or not mutual funds are misclassified. The fund's stated objective is used as an external criterion by which the clustering results are be evaluated. Brown and Goetzmann (1997) present evidence of misclassification and suggest that past performances and fund characteristics provide an indication of mutual funds' classes. Grinblatt and Titman (1989), DiBartolomeo and Witkowski (1997) and Payne et al. (1999) add that the size of the fund, expense ratio, management fee, and turnover also affect the fund classification. Therefore, variables included in this part of the analysis include percent cash, expense ratio, percent assets in the top 10 holdings, turnover ratio, and manager tenure.

The current data of four hundred and twenty mutual funds are randomly collected from Morningstar's Principia Pro database. Once the classification is identified, effects of misclassifications are also examined.

3.2.4 Risk Classification for Dot-Com Companies

Technology has changed the way business has been conducted over the past several years. Electronic commerce has become an important sector in the business world. Companies in this sector are sometimes called dot-com companies because they receive customers' orders mainly via the Internet or network. Beside analysts' recommendations (which are still ambiguous regarding their reliabilities and risk), classifications for companies in this sector have never been investigated. The companies in this sector (SIC code 737) should be inspected separately because there is no unique characteristic for this sector except that they conduct their business mainly on the Internet. These companies have offered varieties of products and services ranging from booksellers to Internet service providers; hence, heterogeneity of product is high among the dot-com companies. Thus, multi-levels of risk are expected among companies in this sector of the economy. This section attempts to discover a latent cluster structure within this sector. The risk classifications for these companies provide insight concerning their creditworthiness. This information is useful for creditors in making decision regarding the granting of finance services to these companies. Srinivasan and Kim (1987) include current ratio, quick ratio, net worth to total debt, total assets, net income to sales, and net income to total assets when modeling creditworthiness. Variables used by Srinivasan and Kim (1987) are expected to correlate positively with the creditworthiness.

Data for the year 2000 are collected from Compustat's Research Insight. Only the best clustering algorithm and KM are used in the analysis to uncover the latent cluster structure. The CCC and psudo-F indices are criteria used to identify the best cluster structure.

CHAPTER 4

RESULTS ON SIMULATED DATA

In the first part of the analysis on the simulated data, I analyze the percent of correct classification for each of the five clustering approaches. First, I test if the five factors and their first-order interactions affect the accuracy of the five clustering approaches using analysis of variance (ANOVA). Table 2 reports the F-statistics for the main effects and the first-order interaction effects, the average correct classification, and its variation.

4.1 Classification Accuracy and the Five Factors

The k-means algorithm (KM) is the most sensitive but stable approach. Although its accuracy is ranked number three, it varies by all five main effects and by six interaction effects at a significant level of 0.01. However, KM produces the most stable results on the basis of the Root MSE (within-group variation). The neural network (NN) is less sensitive to the main and interaction effects. NN is only sensitive to four main and four interaction effects. The numbers of clusters and dimensions factors, in particular, illustrate similar results to the finding of Balakrishnan et al. (1994, 1996). The two factors deviate the clustering performance of NN. However, NN is ranked second to worst in term of accuracy and the worst in term of variation of the accuracy. It should be noted that NN is not given the true number of latent clusters. NN also takes longer time than KM, which is consistent with Tam and Kiang (1992)'s findings.

Table 2: Main and Interaction effects

	KM	NN	NK	GA	GK
Scale (SC)	5.54 (0.0044)	6.41 (0.0019)	1.41 (0.2460)	2.14 (0.1193)	1.53 (0.2194)
Size (SI)	11.79 (0.0007)	1.76 (0.1863)	13.59 (0.0003)	5.68 (0.0178)	3.29 (0.0707)
Density (DN)	61.84 (<.0001)	151.70 (<.0001)	105.58 (<.0001)	7.47 (0.0007)	9.14 (0.0001)
Cluster (CL)	16002.90 (<.0001)	922.80 (<.0001)	15559.30 (<.0001)	49.36 (<.0001)	5524.30 (<.0001)
Dimension (DI)	16.11 (<.0001)	235.36 (<.0001)	30.07 (<.0001)	29.98 (<.0001)	5.79 (0.0034)
SCxSI	0.56 (0.5724)	0.07 (0.9299)	3.34 (0.0369)	0.69 (0.5012)	0.71 (0.4926)
SCxDN	4.34 (0.0020)	0.23 (0.9232)	0.94 (0.4418)	1.23 (0.2986)	1.12 (0.3461)
SCxCL	4.37 (0.0019)	1.75 (0.1397)	1.28 (0.2785)	1.02 (0.3998)	0.58 (0.6764)
SCxDI	0.51 (0.7288)	9.87 (<.0001)	4.22 (0.0025)	1.91 (0.1088)	0.59 (0.6697)
SIxDN	14.06 (<.0001)	0.34 (0.7137)	5.75 (0.0036)	0.85 (0.4266)	0.84 (0.4335)
SIxCL	8.75 (0.0002)	0.36 (0.7012)	7.71 (0.0005)	0.23 (0.7923)	1.59 (0.2058)
SIxDI	0.19 (0.8258)	0.51 (0.6005)	0.80 (0.4483)	2.24 (0.1084)	1.10 (0.3345)
DNxCL	64.14 (<.0001)	93.07 (<.0001)	47.90 (<.0001)	1.83 (0.1236)	11.93 (<.0001)
DNxDE	1.75 (0.1383)	102.10 (<.0001)	2.19 (0.0703)	0.24 (0.9180)	1.67 (0.1559)
CLxDI	14.23 (<.0001)	14.03 (<.0001)	20.84 (<.0001)	0.47 (0.7573)	2.41 (0.0495)
R-Square	0.991421	0.925849	0.991197	0.436419	0.975365
Mean	0.659125	0.449879	0.660075	0.356342	0.664394
Root MSE	0.027505	0.048341	0.027949	0.065076	0.046724

The hybrid of the neural network and the k-means algorithm (NK) performs well on the basis of correct classification. Its results are also second best based on the Root MSE. However, the NK is sensitive to four main and seven interaction effects. Its F-statistic values are generally between the F-statistics of KM and NN. Even though the number of sources that affect clustering performance of NK equals that of KM, the significance levels are generally lower. The NK appears to preserve the accuracy of KM and retain the insensitivity of NN.

The accuracy of the genetic algorithm (GA) fluctuates across levels of four factors: size, density, cluster, and dimension. Similar to NN, GA performs poorly because the true number of latent clusters is not given and it is also a heuristic approach. GA consumes more time than KM and NN, which is consistent with Chiou and Lan (2001). However, the hybrid of the genetic algorithm and the k-means algorithm (GK) inherits the accuracy of KM and the stability of GA. GK possesses the highest accuracy among the five clustering approaches while it is only sensitive to four factors at a 0.01 significant level.

I then test how the accuracy of the five clustering approaches differ across various levels of the five factors using analysis of variance (ANOVA). Table 3 illustrates the percent correct classification of the five clustering approaches at each level of the five factors. The subscription represents group membership; for example, KM correctly classifies 66.4372% of observations on average when the sample size is 210 and 65.3879% when the sample size is 420. The two percentages are significantly different at a 0.05 level. Therefore, the increase in sample size appears to coincide with a decrease in the accuracy of KM. KM seems to work best when the scale factor is at 1.5, when cluster sizes are similar (density level equals to 0), and at the minimum level of number of clusters. Unexpectedly, the performance of KM decreases as the number of dimensions increases. The results on NN seem to be the opposite of the results on KM. The NN's performance is nonlinearly correlated to the scale since the accuracy is at the lowest point when the scale factor is 1.5. NN seems to work better as the scale deviates from 1.5 while it works best at the lowest level of dimension. Sample size does not affect the NN's performance. NN produces the lowest correct classification when the cluster sizes are approximately equal. The number of clusters appears to be inversely related to the NN's performance. Relationship between the five factors and NK seems to be linear. NK is insensitive to the scale factor and ability to correctly cluster observations is inversely related to sample size, number of clusters, and number of dimensions. NN does not work well when one cluster contains most of the observations (high density). Similar to NK, GA is insensitive to the scale factor and inversely related to sample size, number of clusters, and number of dimensions. However, GA does not perform well when one cluster contains less observation than all other clusters. GK is insensitive to both scale and size factors. It preserves the reaction of the KM to the density, cluster, and dimension factors.

Table 3: Mean Percent of Correct Classifications

VARIABLE	LEVEL	KM	NN	NK	GA	GK
Overall		65.9125	44.9879	66.0075	35.6342	66.4394
Scale	1.0	65.5507	45.7626	65.8152	34.6252	65.9744
	1.5	66.6317	43.6328	65.8313	36.4154	67.0542
	2.0	65.5552	45.5683	66.3761	35.8619	66.2894
Size	210	66.4372	45.3437	66.5799	36.4958	66.9102
	420	65.3879	44.6321	65.4352	34.7725	65.9685
Density	0	68.1524	40.2909	67.7828	36.0646	68.0090
	20	65.5469	43.3019	67.4161	33.7481	65.6592
	60	64.0383	51.3709	62.8237	37.0897	65.6499
Cluster	2	100.0000	58.1969	100.0000	39.3585	100.0000
	3	64.6644	46.6789	65.0974	36.7636	66.1504
	7	33.0731	30.0880	32.9252	30.7804	33.1677
Dimension	3	67.0807	53.0331	67.4387	39.3208	67.6743
	5	65.6524	42.5087	66.0911	35.0400	65.9828
	7	65.0044	39.4219	64.4928	32.5417	65.6610

I next test to determine if the correct classification rate of KM differs from that of hybrid models using the analysis of variance of contrast variables. Table 4 reports results of the ANOVA of contrast variables. NK and KM perform equally well while GK provides a higher correct classification rate than KM at 0.10 significant level within an environment that favors KM. These results (summarized in Tables 4 and 5) lead me to conclude that GK generally outperforms KM with regards to rate of correct classification.

Table 4: Analysis of Variance of Contrast Variables

SOURCE	DF	TYPE III SS	MEAN SQUARE	F VALUE	P > F
NK					
MEAN	1	0.00029241	0.00029241	0.14	0.7079
ERROR	323	0.67136121	0.00207852		
GK					
MEAN	1	0.00899231	0.00899231	2.81	0.0948
ERROR	323	1.03463784	0.00320321		

In conclusion, density level, number of clusters, and number of dimensions greatly and consistently influence the clustering accuracy of all approaches. KM appears to be the most sensitive approach to all five factors, but it performs cluster analysis with the most consistency. If we run KM on the same data set for multiple times, the results would likely to be the same for all repetition (not the case for NN and GA). Although the two machine learning approaches (NN and GA) appear to be insensitive to the testing factors, these two approaches do not achieve comparable rates of correct classification because the simulation process intentionally generates problems that favor KM (so as to enable us to compare the two hybrid approaches to KM under the most rigorous of conditions). However, NK and GK appear to inherit the insensitivity of the machine learning approaches as well as the accuracy and the consistency of KM. Both hybrid models classify observations at least as good as KM even in conditions in favor of KM.

4.2 Ranked Performance

In the first part in this chapter we find that the five factors and their interactions explain much of the differences in the accuracy of the five tested clustering approaches; however, we do not know if the changes in accuracy initiate changes in rank. In another words, we do not have enough information to decide if the five approaches react in a similar manner to the five factors. In the second part of the analysis on the simulated data, I investigate the relationships between the five factors and their interactions on the relative ranks of the five clustering approaches (based on their clustering performances). I, first, rank the five approaches from one to five where one is the approach with the highest correct classification rate and 5 is the approach with the lowest correct classification rate on each of the 324 data sets. I then test to determine if the five factors and their first-order interactions are related to the ranks of the five clustering approaches using analysis of variance (ANOVA).

Table 5 reports the F-statistics for the main effects and the first-order interaction effects, the average rank, and its variation. The rank of KM is related to three main effects: scale, density, and cluster. Its deviation is high according to the root MSE but the mean rank is between the first and the second. The rank of NN is the most sensitive but stable approach. All five main factors are related to the rank of the NN. The NN is ranked between the third and the fourth on average and this ranking does not vary much. The rank of GA also varies by the density, cluster, and dimension factors. GA is ranked between the fourth and the fifth and this rank is very consistent. The ranks of the two hybrid models are similar to the rank of KM and are each related to density and cluster factors.Since the ranks of all five approaches are related to the five factors, I conclude that the rank of the five clustering approaches differ across various levels of the five factors especially the density levels and number of clusters.

Table 5: Main and Interaction effects on Rank

	KM	NN	NK	GA	GK
Scale (SC)	3.19 (0.0426)	4.46 (0.0124)	1.00 (0.3705)	0.22 (0.8046)	0.68 (0.5093)
Size (SI)	0.98 (0.3231)	3.96 (0.0475)	0.03 (0.8711)	0.03 (0.8521)	0.47 (0.4943)
Density (DN)	17.48 (<.0001)	317.52 (<.0001)	62.88 (<.0001)	11.60 (<.0001)	8.24 (0.0003)
Cluster (CL)	136.70 (<.0001)	8.61 (0.0002)	153.26 (<.0001)	238.97 (<.0001)	123.85 (<.0001)
Dimension (DI)	1.17 (0.3118)	37.70 (<.0001)	0.64 (0.5280)	4.03 (0.0188)	4.23 (0.0155)
SCxSI	1.90 (0.1517)	1.08 (0.3398)	0.44 (0.6431)	1.42 (0.2437)	0.33 (0.7195)
SCxDN	6.20 (<.0001)	1.37 (0.2439)	1.57 (0.1823)	0.35 (0.8452)	2.05 (0.0881)
SCxCL	1.00 (0.4097)	0.67 (0.6119)	1.33 (0.2578)	1.99 (0.0956)	0.49 (0.7463)
SCxDI	1.68 (0.1546)	2.79 (0.0267)	2.73 (0.0296)	1.16 (0.3297)	1.32 (0.2633)
SIxDN	2.10 (0.1241)	0.22 (0.8031)	4.01 (0.0193)	1.11 (0.3325)	3.80 (0.0236)
SIxCL	2.29 (0.1035)	0.38 (0.6815)	1.99 (0.1391)	1.99 (0.1381)	1.68 (0.1878)
SIxDI	2.04 (0.1318)	1.25 (0.2887)	2.86 (0.0591)	0.90 (0.4091)	0.90 (0.4070)
DNxCL	19.82 (<.0001)	173.39 (<.0001)	29.39 (<.0001)	8.99 (<.0001)	16.84 (<.0001)
DNxDE	4.12 (0.0029)	32.52 (<.0001)	1.45 (0.2172)	2.71 (0.0304)	0.63 (0.6393)
CLxDI	4.95 (0.0007)	16.94 (<.0001)	0.53 (0.7129)	1.73 (0.1429)	2.43 (0.0481)
R-Square	0.632696	0.854578	0.681060	0.675956	0.575867
Mean	1.845679	3.910494	1.913580	4.382716	1.842593
Root MSE	0.673527	0.474390	0.683927	0.595436	0.73.0674

Table 6: Mean Rank by Factors

VARIABLE	LEVEL	KM	NN	NK	GA	GK
Overall		1.84567	3.91049	1.91358	4.38271	1.42593
Scale	1.0	1.92593	3.83333	1.92593	4.39815	1.82407
	1.5	1.71296	4.01852	1.97222	4.39815	1.79630
	2.0	1.89815	3.87963	1.84259	4.35185	1.90741
Size	210	1.80864	3.96296	1.90741	4.37654	1.81481
	420	1.88272	3.85802	1.91975	4.38889	1.87037
Density	0	1.66667	4.41667	1.69444	4.43519	1.70370
	20	1.71296	4.34259	1.53704	4.54630	1.75000
	60	2.15741	2.97222	2.50926	4.16667	2.07407
Cluster	2	1.00000	4.06481	1.00000	4.93519	1.00000
	3	2.07407	3.82407	2.17593	4.85185	1.98148
	7	2.46296	3.84259	2.56481	3.36111	2.54630
Dimension	3	1.92593	3.61111	1.89815	4.44444	2.00926
	5	1.81481	4.16667	1.87037	4.25000	1.76852
	7	1.79630	3.95370	1.97222	4.45370	1.75000

Table 6 summarizes the average rank of all five approaches in details. Ranks of all approaches are worse when one cluster contains most of the observations (at higher density). The performance of GA improves as the number of clusters increases while all other approaches perform worse under similar conditions. KM, NK, and GK perform well when there are only two clusters. . The results summarized in Table 7 indicate the hybrid approaches perform at least as good as KM on average.

Table 7: Analysis of Variance of Contrast Ranked Variables

SOURCE	DF	TYPE III SS	MEAN SQUARE	F VALUE	P > F
NK					
MEAN	1	1.4938272	1.4938272	0.92	0.3373
ERROR	323	522.5061728	1.6176662		
GK					
MEAN	1	0.0030864	0.0030864	0.00	0.9625
ERROR	323	450.9969136	1.3962753		

In conclusion, the five factors (scale, size, density cluster, and dimension) are related to the five clustering approaches in term of correct classification. However, the accuracy of the five approaches differ similarly across various levels of the five factors. The density level and number of clusters are two major effects most strongly related to the ranks of the five approaches. KM is the most sensitive to the five factors while GA provides the least accurate results but it is the least sensitive to the five factors. The hybrid models inherit the sensitivity of the machine learning approaches and the accuracy and stability of KM. There is no evidence that KM outperforms either of the two hybrid models under conditions favorable to KM. GK is even found to be superior to KM at a 0.10 level of significant on average. Therefore, we expect the hybrid models to dominate KM when applied to the empirical problems in the next chapter (where the conditions do not necessarily favor KM).

CHAPTER 5

EMPIRICAL EVIDENCE

In the previous chapter, I investigate the performance of the five clustering procedures in an environment where clusters are well-separated, variances are approximately equal, observations are normally distributed, correlation between any pair of variables is not substantial, and collinearity problem is at an acceptable level. In addition, the latent number of clusters is only supplied to the k-means algorithm (KM) in all previous analyses. Therefore, the environment created in the simulated problems from the previous chapter favor KM over the other tested clustering approaches. In this chapter I test the five clustering approaches on real-world problems where the test conditions are not controlled, i.e., the clusters may not be well-separated, clusters' variances may not be uniform, observations in each cluster may not be normally distributed, a significant correlation between the dimensions may be present, and collinearity problem may be severe. The real-world problems investigated in this study include both problems with and without natural clusters. Problems with natural clusters include acquisition targets and bankruptcy predictions, analysts' stock recommendation, and mutual funds classification. Finally, I attempt to uncover a latent cluster structure in the dot-com industry since the industry comprises of several of types of companies (and so no natural cluster structure exists).

5.1 Acquisition Targets and Bankruptcy Predictions

Data for this problem are collected from CompactDisclosure and Compustat's Research Insight covering the year of 2000. I first search for the bankrupt and acquired firms in the CompactDisclosure using keywords such as bankrupt, acquired, and merged. Next, I exclude firms with "active" status. I then search for the companies in the same SIC code in the Compustat's Research Insight using the company-name-lookup feature. This data set includes only bankruptcies and acquisitions occurring between the year 2000 and 2002. The random sample contains only mining and manufacturing companies (SIC: 1000-3999). To avoid industry effects, I exclude biotech and pharmaceutical companies (SIC: 28XX), electronics and telecom (SIC: 3653-3689), and computer and technology (SIC: 357X) because companies in these sectors possess characteristics and risk different than the mining and manufacturing companies. I identify 2,577 companies in the Research Insight through the search procedure described earlier. Of these 2,577 companies, twenty-three suffered bankruptcy and eight were acquired. Variables used in this analysis include sales per total assets (X1), financial leverage (X2), working capital per share (X3), EBIT (X4), and dividend payout (X5). I denote independent firms as group 1(G1), acquired firms as group 2 (G2), and bankrupted firms as group 3 (G3).

There are two popular sampling methods found in literature regarding bankruptcy predictions: (a) randomly draw a pre-specified number of firms and (b) match the number of bankrupt and acquired firms by SIC code and total assets. It is obvious that the second sampling method is not probabilistic and does not preserve the true proportions of the three groups. That persuades some researchers to prefer the first approach to the second sampling methods. However, other researchers argue that the prior and posterior probabilities may not be the same, especially in a dynamic setting such as in financial problems. As a result, the proportions of the three groups change over time. Thus, the second sampling method may be as robust as the first method. Since a clear conclusion regarding which sampling method is the best has not been identified, I analyze the problems using both sampling methods.

5.1.1 Random Sampling Method

I first randomly select seventy-four independent companies. Thus, the sample size includes twenty-three bankrupt and eight acquired firms as well as seventy-four independent companies (for a total sample of hundred and five firms). Table 8 reports descriptive statistics for this data set. Note that the variables are not measured on the same scale. The standard deviations also significantly vary from cluster to cluster as indicated by the Hartley's F-Max test (or the Folded Form F test in SAS). For example, the standard deviation of X2 is only 2.939 in the second group and 141.751 in the third group. The clusters are not well-separated on any single dimension; for instance, X1 ranges from 0.000 to 5.054 in the first group while it ranges from 0.008 to 2.433 in the third group, thus these two clusters overlap on this dimension. The descriptive statistics suggest a more complex cluster structure than what was analyzed in the previous chapter.

Table 8: Descriptive Statistics for the Acquisition and Bankruptcy Problem.

Group	VAR	G1	G2	G3	Hartley's F-Max Test	p-value
N		74.000	8.000	23.000		
Max	X1	5.054	4.244	2.433		
	X2	9.666	7.973	35.192		
	X3	23.220	16.012	9.449		
	X4	1822.000	95.058	272.734		
	X5	3815.150	0.000	71.618		
Min	X1	0.000	0.000	0.008		
	X2	-201.035	-1.261	-672.796		
	X3	-16.607	-26.965	-21.949		
	X4	-86.273	-16.222	-177.000		
	X5	-29.774	-3.914	-3.330		
Mean	X1	1.034	1.263	1.040		
	X2	-0.848	2.397	-27.710		
	X3	2.981	1.727	-0.465		
	X4	81.676	12.709	4.869		
	X5	65.143	-0.489	3.521		
Standard	X1	0.837	1.276	0.665	3.682	0.007
Deviation	X2	23.775	2.939	141.751	2326.232	0.000
	X3	4.977	12.651	7.908	6.461	0.000
	X4	244.814	35.557	73.149	11.201	0.001
	X5	446.594	1.384	15.191	104124.658	0.000

Table 9 provides classification results for the five tested clustering procedures. Classification results for the KM, genetic algorithm (GA), and the hybrid between the k-means and genetic algorithm (GK) are similar. The neural network and its hybrid (NN and NK) do not perform as well as the KM, GA, and GK. However, only NN and NK were able to detect the merger targets (group 2). I also test to determine if the relative frequency of the KM differs significantly from that of other clustering approaches. Baesd on McNemar's test, NN's thirty-eight and NK's fifty-eight frequencies in the first group are significantly lower than the KM's seventy-two at 0.01 level.

Table 9: Correct Classifications for the Acquisition and Bankruptcy Problem

Group	Actual	KM	NN	NK	GA	GK
1	74	72	38a	58a	71	72
2	8	0	1	4b	0	0
3	23	3	10a	3	5	3
Overall	105	75	49a	65a	76	75

c significant difference from the k-means' at 0.10 level
b significant difference from the k-means' at 0.05 level
a significant difference from the k-means' at 0.01 level

5.1.2 Matched Sampling Method

Using the matched sampling approach, I match each of twenty-three bankrupt and eight acquired firms with the company in the same industry (four-digit SIC code) with the closest size (total assets). As a result, this data set contains sixty-two companies: twenty-three bankrupt, eight acquired, and thirty-one independent firms. The descriptive statistics for this data set are provided in Table 10. Characteristics of the second and third groups are the same as in Table 8 since the sample of bankrupt and acquired firms is unchanged. Although the descriptive statistics of the first group change, the cluster structure continues to be complex. Variables are still measured on different scales. The standard deviations still vary from cluster to cluster. The clusters are again not separated on any single dimension. Table 11 presents clustering performances of the five clustering approaches. The two hybrids perform as well as the KM while the two machine learning approaches illustrate a higher number of overall correct classifications than that of the individual approaches. The two machine learning approaches (NN and GA) appear to be able to identify bankrupted firms better than the KM. However, the machine learning approaches detects fewer independent firms than the KM.

Table 10: Descriptive Statistics for the Acquisition and Bankruptcy Problem.

Group	VAR	G1	G2	G3	Hartley's F-Max Test	p-value
N		31.000	8.000	23.000		
Max	X1	4.176	4.244	2.433		
	X2	6.687	7.973	35.192		
	X3	29794.000	16.012	9.449		
	X4	338.200	95.058	272.734		
	X5	149.257	0.000	71.618		
Min	X1	0.000	0.000	0.008		
	X2	-201.035	-1.261	-672.796		
	X3	-10.850	-26.965	-21.949		
	X4	-25.970	-16.222	-177.000		
	X5	0.000	-3.914	-3.330		
Mean	X1	1.255	1.263	1.040		
	X2	-4.332	2.397	-27.710		
	X3	966.642	1.727	-0.465		
	X4	32.018	12.709	4.869		
	X5	12.786	-0.489	3.521		
Standard	X1	0.870	1.276	0.665	3.682	0.007
Deviation	X2	36.563	2.939	141.751	2326.232	0.000
	X3	5350.140	12.651	7.908	457716.925	0.000
	X4	77.685	35.557	73.149	4.773	0.013
	X5	32.787	1.384	15.191	561.217	0.000

Table 11: Correct Classification for the Acquisition and Bankruptcy Problem

Group	Actual	KM	NN	NK	GA	GK
1	31	26	16a	26	17a	25
2	8	0	0	1	2	0
3	23	1	16a	1	15a	1
Overall	62	27	32b	28	34a	26

c significant difference from the k-means' at 0.10 level
b significant difference from the k-means' at 0.05 level
a significant difference from the k-means' at 0.01 level

5.2 Analysts' Stock Recommendations

This part of analysis incorporates five variables: five-year average growth (X1), beta (X2), PE ratio (X3), dividend payout (X4), and volume (X5). The data used for this analysis are also comprised of five groups: buy (G1), buy/hold (G2), hold (G3), sell/hold (G4), sell (G5). Data regarding beta, PE ratio, dividend payout, and volume for this analysis are collected from Compustat's Research Insight during the period of 2000. During this period there are 231 companies in the real estate investment trust (REIT) industry (SIC 6798). The average analysts' recommendations and five-year growth rate as of December 2002 are collected from Yahoo!Finance's stock screener website. There are 201 REITs available from the website, 107 of which remain after elimination of the incomplete observations.

5.2.1 Five-Cluster Structure

Table 12 provides descriptive statistics on the data set. No "sell" recommendation exists in this industry, so G5 has zero frequency. Two relatively high-density and two relatively low-density clusters are present in the data set. The variables are also measured on different scales. For example, X1 ranges from -0.445 to 0.651 while X5 ranges from 32.800 to 15,502.400. Mean ranges from 0.052 for X1 in G1 to 3381.793 for X5 in G3. Standard deviations for each dimension vary from group to group; thus, cluster geometric areas are not constant. The clusters are not well-separated on any single dimension; for instance, X1 ranges from 0.026 to 0.122 in the first group while it ranges from -0.445 to 0.651 in the second group, thus the first cluster resides in the second cluster on this dimension. The descriptive statistics for this problem also suggest a more complex cluster structure than what was analyzed in the previous chapter.

Table 12: Descriptive Statistics for the Analysts' Recommendation Problem.

Group	VAR	G1	G2	G3	G4	G5	Hartley's F-Max Test	p-value
N		4.000	35.000	59.000	9.000	0.000		
Max	X1	0.122	0.651	0.391	0.230	0.000		
	X2	0.865	0.865	0.510	0.390	0.000		
	X3	23.922	127.500	94.222	28.560	0.000		
	X4	310.739	1601.000	523.962	293.241	0.000		
	X5	757.400	10493.400	15502.400	2739.800	0.000		
Min	X1	0.026	-0.445	-0.169	0.000	0.000		
	X2	0.080	-0.132	-0.164	-0.075	0.000		
	X3	4.439	5.630	5.429	6.172	0.000		
	X4	47.703	46.448	46.379	53.509	0.000		
	X5	32.800	38.400	219.600	206.300	0.000		
Mean	X1	0.052	0.077	0.070	0.084	0.000		
	X2	0.293	0.205	0.178	0.179	0.000		
	X3	10.312	20.126	17.768	15.248	0.000		
	X4	133.455	201.029	149.402	135.257	0.000		
	X5	289.650	1840.206	3381.793	809.567	0.000		
Std.	X1	0.047	0.151	0.080	0.075	0.000	10.322	0.017
Dev	X2	0.382	0.198	0.153	0.158	0.000	6.234	0.000
	X3	9.229	23.668	12.114	7.243	0.000	10.678	0.000
	X4	123.156	286.609	93.201	65.917	0.000	18.905	0.000
	X5	339.351	1983.974	3678.184	786.307	0.000	117.481	0.000

Classification results for the analysts' recommendation data are presented in Table 13. The two machine learning approaches, NN and GA, are more accurate than the KM in classifying the observations. They both identify the members of the third group (the "hold" recommendation) better than the KM. The GA, in particular, makes 50% correct prediction on the first group (the "buy" group), while the KM fail to detect any members of this group. The KM, NK, and GK illustrate similar accuracy. However, NK is less accurate than KM in classifying the observations, which is consistent with the findings in the acquisition and bankruptcy problem.

Table 13: Correct Classification for the Analysts' Recommendations Problem

Group	Actual	KM	NN	NK	GA	GK
1	4	0	0	0	2	1
2	35	27	13a	15a	7a	27
3	59	15	48a	23	57a	15
4	9	0	0	0	0	0
5	0	0	0	0	0	0
Overall	107	42	61b	38	66a	43

c significant difference from the k-means' at 0.10 level
b significant difference from the k-means' at 0.05 level
a significant difference from the k-means' at 0.01 level

5.2.2 Two-Cluster Structure

Since the first, fourth, and fifth groups are sparse; I collapse the first and second group together and also combine the third, fourth, and fifth groups. This results in the formation of two groups: "buy" (Gl) and "not-buy" (G2). Table 14 reports the descriptive statistics for the new data set for the analysts' recommendation problem. The characteristics of the new data set presented in Table 14 are similar to that of the old data set presented in Table 12. For example, variables are not measured on the same scale, cluster geometric areas are not equal, and the clusters are not well-separated. The cluster structure for the resulting problem is complex

Table 15 reports the correct classification for the five tested clustering methods. NN, GA, and GK generate higher numbers of correct classifications than the KM. However, the KM and NK illustrate similar classification results, which supports the conclusions reached in the analysis of the simulated data set (that the KM and NK are equally accurate and perform equally well with two-cluster structure). Another interesting point that should be addressed is that the GA and GK identify members of the first group ("buy" recommendations) better than the KM. We also see the NN and GA detect more bankrupted and acquired firms than the KM in the acquisition targets and bankruptcy predictions problem. Therefore, I conclude that machine learning approaches (NN and GA) and the two hybrids (NK and GK) are more effective than the KM in detecting members of a minor group.

Table 14: Descriptive Statistics for the Analysts' Recommendation Problem.

Group	VAR	G1	G2	Hartley's F-Max Test	p-value
N		39.000	68.000		
Max	X1	0.651	0.391		
	X2	0.865	0.510		
	X3	127.500	94.222		
	X4	1601.000	523.962		
	X5	10493.400	15502.400		
Min	X1	-0.445	-0.169		
	X2	-0.132	-0.164		
	X3	4.439	5.429		
	X4	46.448	46.379		
	X5	32.800	206.300		
Mean	X1	0.075	0.072		
	X2	0.214	0.178		
	X3	19.120	17.435		
	X4	194.099	147.530		
	X5	1681.174	3041.351		
Standard	X1	0.144	0.079	3.323	0.000
Deviation	X2	0.218	0.152	2.057	0.005
	X3	22.739	11.577	3.858	0.000
	X4	274.093	89.7874	9.319	0.000
	X5	1938.564	3543.538	3.341	0.000

Table 15: Correct Classification for the Analysts' Recommendation Problem.

Group	Actual	KM	NN	NK	GA	GK
1	39	1	0	1	5b	4c
2	68	62	68b	62	65	65
Overall	107	63	68c	63	70c	69c

c significant difference from the k-means' at 0.10 level
b significant difference from the k-means' at 0.05 level
a significant difference from the k-means' at 0.01 level

5.3 Mutual Fund Classification

Originally there are 7,938 domestic stocks funds in Morningstar's Principia Pro. I eliminate funds that do not have one of the following seven objectives: aggressive growth (G1), asset allocation (G2), balanced (G3), equity income (G4), growth (G5), growth and income (G6), and small company (G7) because funds with other objectives engage heavily in either bonds, foreign securities, or utility companies. That might cause the clustering algorithm to classify funds based on types of securities, environment, or restriction rather than objectives. As a result, 6,633 domestic stock funds remain the database. Unfortunately, there are only 4,258 funds remaining with complete information. Out of 4,258 funds, 2,017 funds are no-load funds and 2,241 funds are load funds. I randomly select four hundred and twenty no-load funds and four hundred and twenty load funds from the funds with complete information for the percent cash (x1), expense ratio (x2), percent assets in the top 10 holdings (x3), turnover ratio (x4), and manager tenure (x5). I then analyze both samples separately because load and no-load funds possess different characteristics (shown later in section 5.3.2).

5.3.1 No-Load Funds

As illustrated by the summary statistics in Table 16, the latent cluster structure is very complex because of the following reasons. First, the clusters are not uniform in geometric area based on their standard deviations on the five dimensions. Second, the means and standard deviations on the five dimensions suggest that the clusters are relatively close together, which implies that clusters are not well-separated. In addition, five variables are measured on different scales - X1 ranges from 0 to 69 while X2 extends from 0 to 3.48. X3 and X4 possess higher ranges than X1, X2, and X5.

Table 16: Descriptive Statistics for the Sample of No-Load Funds.

Group	VAR	G1	G2	G3	G4	G5	G6	G7	Hartley's F-Max Test	p-value
N		13.0	20.0	30.0	18.0	200.0	64.0	75.0		
Max	X1	8.0	36.4	51.8	25.5	69.7	20.2	33.4		
	X2	3.5	3.0	3.1	1.6	2.8	2.4	2.5		
	X3	100.0	130.0	100.0	84.0	100.0	89.3	101.5		
	X4	255.0	242.0	227.0	179.0	869.0	218.0	242.0		
	X5	12.0	19.0	10.0	13.0	25.0	23.0	15.0		
Min	X1	0.0	0.0	0.0	0.0	0.0	0.0	0.0		
	X2	0.2	0.1	0.0	0.5	0.0	0.1	0.1		
	X3	17.5	12.5	12.7	15.6	7.1	19.8	2.3		
	X4	6.0	2.0	12.0	8.0	3.0	3.0	3.0		
	X5	2.0	1.0	1.0	1.0	1.0	1.0	1.0		
Mean	X1	4.0	8.1	7.5	5.5	5.2	3.7	5.1		
	X2	1.5	0.9	0.9	0.9	1.2	0.8	1.2		
	X3	44.6	55.1	38.7	29.1	35.4	33.9	25.1		
	X4	126.9	77.8	83.1	65.0	119.5	69.7	94.0		
	X5	4.5	5.2	4.9	5.2	4.3	4.9	4.6		
Std.	X1	2.9	9.3	10.6	7.1	10.2	4.3	6.4	13.360	0.000
Dev	X2	1.0	0.7	0.6	0.3	0.4	0.4	0.5	11.111	0.000
	X3	25.9	37.2	26.6	15.6	15.8	13.0	14.9	8.188	0.000
	X4	77.8	70.7	58.6	43.1	128.9	51.1	57.2	8.944	0.000
	X5	2.6	4.6	2.3	3.4	3.8	3.5	2.7	4.000	0.000

Next I perform cluster analysis using the five clustering approaches on the no-load funds data. Table 17 reports the number of funds correctly classified by approach and by cluster. As illustrated in Table 16 and 18, the numbers of observations are unequal and range from 13 to 200. Assuming that the fund's manager supervises his/her fund consistently with the stated fund's objective, NK provides the least accurate results (correctly classifies 120 out of 420 funds). This is true even though NN generates the most accurate classification (correctly classifies 170 out of 420 funds). The two machine learning approaches (NN and GA) perform better than KM on the basis of the overall correct classification. The number of funds correctly classified by NN (124 out of 200) exceeds the number of funds correctly classified by KM (84 from 200) at a 0.01 level of significance. The rate of correct classification achieved by KM is not significantly better than the corresponding rate achieved by NK, although does KM generate a relatively high number of correct classifications which supports the conclusions reached in the analysis of the simulated data set and in the analysts' recommendation problems (that the KM and NK are equally accurate). Based on the results, the NN, GA, and GK perform cluster analysis significantly better than KM.

Table 17: Correct Classifications for the No-Load Funds Problem

Group	Actual	KM	NN	NK	GA	GK
1	13	0	0	0	0	3c
2	20	2	0	7b	0	3
3	30	2	0	9b	4	1
4	18	2	0	3	4	3
5	200	84	124a	77b	99a	85
6	64	10	32a	6	5c	34a
7	75	22	14c	18	24	17
Overall	420	122	170a	120	136b	146a

c significant difference from the k-means' at 0.10 level
b significant difference from the k-means' at 0.05 level
a significant difference from the k-means' at 0.01 level

5.3.2 Load Funds

Table 18 reports descriptive statistics for the load funds and indicates one high-, two medium-, and four low-density clusters. Overall descriptive statistics reported in this table are similar to the descriptive statistics of the no-load funds reported in Table 16. Cluster structure for load fund is also complex because of reasons similar to those cited in the analysis of the no-load funds. However, some discrepancies occur; for example, the mean of X2 for the second group (G2) changes from 0.948 for no-load funds to 1.604 for load funds. A question, then, arises whether load and no-load funds possess the same characteristics. Thus, I perform the t-test for equality of means and the F-test (Satterthwaite's) for equality of variance on load versus no-load funds in each objective category.

Table 18: Descriptive Statistics for the Sample of Load Funds.

Group	VAR	G1	G2	G3	G4	G5	G6	G7	Hartley's F-Max Test	p-value
N		14.0	21.0	24.0	14.0	202.0	77.0	68.0		
Max	X1	16.9	18.0	9.0	9.6	45.5	31.1	19.4		
	X2	2.6	2.1	2.3	2.3	2.9	2.4	2.7		
	X3	97.1	99.8	100.0	37.3	77.2	99.9	100.0		
	X4	305.0	165.0	334.0	179.0	684.0	270.0	360.0		
	X5	6.0	9.0	6.0	12.0	15.0	12.0	18.0		
Min	X1	0.0	0.0	0.0	0.0	0.0	0.0	0.0		
	X2	0.0	0.5	0.0	0.9	0.6	0.0	0.9		
	X3	13.5	18.8	14.4	16.1	7.8	17.6	7.3		
	X4	27.0	11.0	6.0	15.0	4.0	3.0	5.0		
	X5	1.0	1.0	1.0	1.0	1.0	1.0	1.0		
Mean	X1	5.6	7.2	3.3	3.1	3.5	4.6	4.2		
	X2	1.5	1.6	1.6	1.6	1.8	1.5	1.8		
	X3	37.0	48.0	35.8	25.1	32.6	33.5	23.3		
	X4	137.8	64.1	74.8	90.5	107.5	76.2	120.5		
	X5	3.1	3.6	3.0	3.9	3.5	3.8	3.7		
Std.	X1	5.5	5.7	2.9	3.6	5.6	5.4	5.1	3.863	0.001
Dev	X2	0.7	0.4	0.5	0.5	0.5	0.5	0.4	3.063	0.010
	X3	26.8	30.2	27.1	5.4	12.2	14.1	14.0	31.277	0.000
	X4	104.2	56.1	68.3	48.2	87.7	56.2	81.9	4.673	0.003
	X5	1.7	2.1	1.6	2.7	2.0	2.5	2.4	2.848	0.012

Table 19 reports the t-statistics and the test statistics of the Hartley's F-Max Test by group and by variable. The mean of the second variable and the variance of the first variable seem to be significantly different for all groups while the fifth group is significantly different. Therefore, I conclude that load funds differ from no-load funds and should be analyzed separately (otherwise the clustering algorithm might classify funds based on whether the fund is load or no-load fund).

Table 19: Equality of Means and Variances

	VAR	G1	G2	G3	G4	G5	G6	G7
Mean (L-NL)	X1	-0.92 (0.3696)	-0.38 (0.7090)	2.04 (0.0486)	1.27 (0.2153)	2.09 (0.0384)	-1.06 (0.2952)	-0.92 (0.3764)
	X2	-0.02 (0.9831)	-3.64 (0.0010)	-4.27 (<.0001)	-4.51 (<.0001)	-13.43 (<.0001)	-8.32 (<.0001)	-0.02 (<.0001)
	X3	0.75 (0.4576)	0.67 (0.5085)	0.40 (0.6904)	1.01 (0.3237)	1.99 (0.0479)	0.16 (0.8726)	0.75 (0.4530)
	X4	-0.30 (0.7630)	0.69 (0.4965)	0.48 (0.6322)	-1.58 (0.1253)	1.09 (0.2744)	-0.72 (0.4736)	-0.30 (0.0284)
	X5	1.66 (0.1092)	1.39 (0.1842)	3.33 (0.0011)	1.18 (0.2483)	2.45 (0.0148)	2.12 (0.0422)	1.66 (0.0358)
Variance	X1	3.53 (0.0360)	2.64 (0.0364)	13.50 (<.0001)	3.90 (0.0167)	3.33 (<.0001)	1.53 (0.0826)	1.56 (0.0672)
	X2	1.80 (0.3056)	2.72 (0.0315)	1.17 (0.7061)	2.48 (0.0812)	1.27 (0.0946)	1.40 (0.1730)	1.34 (0.2203)
	X3	1.07 (0.9108)	1.51 (0.3663)	1.04 (0.9142)	8.47 (0.0004)	1.68 (0.0003)	1.16 (0.5366)	1.13 (0.6002)
	X4	1.80 (0.3188)	1.59 (0.3128)	1.36 (0.4330)	1.25 (0.6587)	2.16 (<.0001)	1.21 (0.4386)	2.05 (0.0028)
	X5	2.52 (0.1110)	4.72 (0.0011)	2.01 (0.0896)	1.67 (0.3520)	3.67 (<.0001)	1.96 (0.0052)	1.24 (0.3698)

Table 20: Correct Classifications for the Load Funds Problem

Group	Actual	KM	NN	NK	GA	GK
1	14	2	0	2	0	2
2	21	10	0a	9	6b	4b
3	24	0	0	0	0	0
4	14	0	0	0	0	1
5	202	89	159a	68c	181a	103
6	77	15	24b	15	8a	18
7	68	15	7a	23b	3a	13
Overall	420	131	190a	117c	198a	141c

c significant difference from the k-means' at 0.10 level
b significant difference from the k-means' at 0.05 level
a significant difference from the k-means' at 0.01 level

I then perform cluster analysis using the five clustering approaches on the no-load funds data. Table 20 reports the number of funds correctly classified by approach and by cluster. Results for the load funds data partially support the conclusions reached in analysis of the no-load funds in Table 17. The NN, GA, and GK still outperform the KM at the 0.01 level of significance. The KM still achieves a higher rate of correct classifications than the NK. In conclusion, the results reported in this mutual fund problem support the conclusion drawn from the simulated data. The hybrid models generally perform as well as the KM and occasionally outperform the KM. The machine learning approaches also outperform the KM as expected since the cluster structure is more complex than it is in the simulated problem.

5.4 Overall results on the real-life problems with natural cluster structure

In this section, I examine the overall correct classifications of all five clustering approaches in real-life problems with the natural cluster structure. Table 21 summarizes correct classification rate of the overall results of all clustering approaches. GA consistently outperforms KM in all real problems. Similarly, GK provides better accuracy than KM in the last four problems; 1.61% less accuracy than KM in the matched sample of the acquisition and bankruptcy problems; and similar accuracy in the random sample of the acquisition and bankruptcy problems. KM performs better than NN and NK in the random sample in the acquisition and bankruptcy problem while it achieve the lowest rate of correct classification for the matched sample in the same problem and in the no-load funds data set. KM outperforms the NK in all cases except in the matched sample of the acquisition and bankruptcy problems. However, the differences between performances of

KM and NK are not statistically significant. Thus, we can assume that KM and NK perform equally well. As expected, the two machine learning approaches perform better than KM on average since the conditions are not in favor of KM in the real-life data sets as they are in the simulated data sets. Note that the machine learning approaches also outperform the two hybrids.

Table 21: Relative Frequency of Overall Correct Classifications

Problem	Average Percent Correct	Acquisition & Bankruptcy		Analysts' Stock Recommendations		Mutual Funds Classification	
		Random Sample	Matched Sample	Five-Cluster	Two-Cluster	No Load Funds	Load Funds
Size		105	62	107	107	420	420
# Clusters		3	3	5	2	7	7
KM	45.56%	71.43%	43.55%	39.25%	58.88%	29.05%	31.19%
NN	50.76%	46.67%	51.61%	57.01%	63.55%	40.48%	45.24%
NK	42.98%	61.91%	45.16%	35.51%	58.88%	28.57%	27.86%
GA	55.64%	72.38%	54.84%	61.68%	65.42%	32.38%	47.14%
GK	47.73%	71.43%	41.94%	40.19%	64.49%	34.76%	33.57%

All approaches perform relatively better in the random sample in the acquisition and bankruptcy problem (average correct classification of above 60%) than in the mutual funds classification problem (average correct classification of below 40%). This may have occurred because the number of clusters and the accuracy are inversely correlated as we have seen in the simulated problems. However, there is no evidence in the real-life data set that the two-cluster structure always allows an optimal clustering performance for clustering approaches except for the NN. The correct classification rates of KM, NK, GA, and GK are at peak in the random sample of the acquisition target and bankruptcy predictions problems where a three-cluster structure presents.

Table 22 illustrates the rank performance of the five clustering approaches in each of the real-life problems. The GA is ranked number one in all real-life data sets except for the no-load funds classification problem, while the KM is ranked the fourth in all data sets except for the random sample of the acquisition and bankruptcy predictions problem. NK fails to achieve a better clustering performance than KM. The GK outperforms the KM in all scenarios except for the matched sample in the acquisition and bankruptcy problem (where the KM outperforms the GK by just 1.61% or one company, which is not significant at all level).

Table 22: Rank of the Overall Correct Classifications

Problem	Average Rank	Acquisition & Bankruptcy		Analysts' Stock Recommendations		Mutual Funds Classification	
		Random Sample	Matched Sample	Five-Cluster	Two-Cluster	No Load Funds	Load Funds
Size		105	62	107	107	420	420
# Clusters		3	3	5	2	7	7
KM	3.67	2	4	4	4	4	4
NN	2.50	5	2	2	3	1	2
NK	4.33	4	3	5	4	5	5
GA	1.33	1	1	1	1	3	1
GK	2.83	2	5	3	2	2	3

In conclusion, GK should be preferred since it is ranked in the top three in simulated data sets and in the real-life problems. KM performs very well in simulated data sets where the environments are in favor of KM but its performance is relatively poor in the less favorable environments of our real problems. The machine learning approaches achieve superior clustering results in the real-life problems where the cluster structure is complex but perform relatively poorly in the simulated environments (they are consistently ranked the fourth and fifth). If information regarding complexity of the cluster structure is available, I would employ KM when the cluster structure is simple and GA when the cluster structure is complex. However, such information is generally not available, and under such circumstances I would prefer an approach that performs relatively well in both situations. The GK satisfies the criterion because it provides the best clustering results in the simulated environment and arguably the third best in the real-life problems. Therefore, I conclude that the GK is the best among the five clustering approaches in this study and should be used in the next problem to uncover the cluster structure among the dot-com companies.

5.5 Risk Classification for Dot-Com Companies

Data regarding dot-com companies (SIC: 737X) are collected from Compustat's Research Insight covering the year of 2000. I identify 1037 companies in the database. After elimination of incomplete observations, 892 firms remain in the data set. I randomly select 420 firms of these remaining firms. The seven variables used in this analysis include current ratio (X1), quick ratio (X2), liability per net worth (X3), total assets (X4), net income per sales (X5), net income per total assets (X6), and price to book ratio (X7).

Table 23 summarizes the descriptive statistics for this sample. X1 and X2 show similar means, standard deviations, and ranges. Variables X3, X4, X5, and X7 have much larger ranges and variances. Interestingly, the net income per total assets (X6) has a negative mean and a maximum of 0.52709 while its minimum is -53.3781. This should imply that companies in this industry have earned little profit or actually suffered loses but were still attractive to many investors during this period.

Table 23: Descriptive Statistics for the Sample of Dot-Com Companies

Variable	N	Mean	Standard Deviation	Minimum	Maximum
X1	420	3.54611	3.72262	0.0020	24.47200
X2	420	3.29231	3.65552	0.0010	24.15300
X3	420	104.06020	907.39760	-2878.0000	16425.00000
X4	420	742.20750	5188.00000	0.0260	88349.00000
X5	420	-26.18160	462.21650	-9468.0000	174.78960
X6	420	-1.22365	4.82738	-53.3781	0.52709
X7	420	-17.5408	486.42750	-9878.0000	1100.00000

Next I perform cluster analysis on this sample using the GK since it is chosen to be the best clustering algorithm in this study when complexity of cluster structure is unknown.. The results of the KM and NK are also used for comparisons. The three approaches (KM, NK, and GK) are configured to uncover cluster structure that includes 2 to 10 clusters. I then decide on the number of clusters using the pseudo-F statistics and the cubic clustering criterion (CCC) as internal criteria. Table 24 reports the results on this data set. Two local optima are present in the sample based on the pseudo-F statistics and CCC. One is within the range of 2 to 6 clusters and another occurs between 7 and 10 clusters. If we intend to cluster the observations into no more than six groups, the KM and NK suggest that five is the optimal number of groups as the pseudo-F statistics and CCC are maximized in the range. At the same time, the GK indicates that observations should be classified into only four groups. We also can obtain another solution with the number of clusters above six. Both KM and NK agree that there are ten natural clusters in the data set while the GK argues that the true number of latent clusters is either seven, based on the pseudo-F statistics, or eight based on the CCC. If we have not conducted prior investigation on the three approaches in simulated environments and real-life problems with natural clusters, we might have made a wrong decision by choosing the

10-cluster structure as the optimal solution based on recommendations of KM and NK. Since we have investigated performance of the three clustering approaches earlier and the GK illustrates a superior performance to the KM and NK, I would conclude that the true number of latent clusters in this industry is seven or eight.

Table 24: Internal Criteria: Pseudo-F and Cubic Clustering Criterion.

Number of Clusters	Expected Overall R-Square	Pseudo-F			CCC		
		KM	NK	GK	KM	NK	GK
2	0.45268	364.24	85.263	364.24	0.897	-15.606	0.897
3	0.57221	362.74	287.83	309.65	6.344	0.727	2.446
4	0.63174	369.34	215.26	370.40	13.017	-2.693	13.108
5	0.67208	626.16	307.43	307.37	38.961	12.213	12.207
6	0.70187	552.11	260.86	285.98	36.586	9.426	12.547
7	0.72512	602.71	255.24	667.85	41.977	10.972	45.917
8	0.74395	594.89	227.07	665.88	43.133	9.007	47.387
9	0.75961	608.70	215.49	230.39	44.468	8.939	11.127
10	0.77292	665.93	632.67	170.22	47.016	47.203	2.871

CHAPTER 6

CONCLUSION, LIMITATIONS, AND FUTURE RESEARCH

Cluster analysis has been around for quite sometime whether or not we realize it. In the last several decades, researchers have paid more attention to the cluster analysis since it can be used as a tool to uncover meaningful information from a vast pool of data. Researchers have invested attention and effort developing robust clustering procedures. The k-means algorithm (KM) is one of the popular and robust approaches. It is generally available in many widely used statistical software packages such as SAS and SPSS. It also consumes small amount of computational time dealing with a large data set. Furthermore, KM requires users to specify the number of clusters and initial clusters' means (seeds). Researchers customarily decide on the number of clusters on the basis of either theory or previous experiment and randomly select seeds for KM. However, it has been demonstrated by many researchers that KM does not perform well with the random seeds. Thus, many researchers have suggested a two-stage approach where the seeds are determined in the first stage by some other procedure and KM is performed in the second stage. Promising evidence of the effectiveness of two-stage approaches have been reported in many scholarly research such as Milligan (1980), Helsen and Green (1991), and Murty and Krishna (1981). Nonetheless, the mentioned researchers have employed procedures that require a great deal of computational time and so are unable to handle large data sets efficiently.

With current computing technology, the computational intensive procedures such as machine learning become viable. Two machine learning approaches have demonstrated attractive clustering abilities. The neural network (NN) and the genetic algorithm (GA) are flexible in term of functional forms. They do not require some assumptions that must be met when using linear and parametric procedures. Both NN and GA have been configured to handle tasks in cluster analysis and perform well. Yet they have not been used to pre-screen the seeds for KM.

In this study I investigate and conduct an experiment on two-stage clustering procedures, hybrid models in simulated environments where conditions such as collinearity and cluster structure are controlled. The performance of these procedures is also evaluated on real-life problems where conditions are not controlled. The first hybrid (NK) model integrates a neural network with KM. NN screens seeds and passes them to KM. The second hybrid (GK) is similar but uses a genetic algorithm instead of NN to screen the seeds for KM. Both NN and GA used in this study are of the simplest possible form. NN used in this study is a simple feedforward unsupervised system that consists of three layers. The number of nodes is equal to the number of variables in the first layer and is equal to the number of clusters in the second layer. A single node in the last layer classifies observations into groups. GA used in this study utilizes ten chromosomes. Each chromosome represents a possible solution to the clustering problem where the solution is a set of clusters means. GA utilizes a two-point crossover, ten percent mutation rate, and ten percent inversion rate. The parent chromosomes for the member of the next generation are selected through roulette-wheel selection.

In simulated data sets I investigate two properties: comparative clustering performance and the impact of five factors (scale, sample size, density, number of clusters, and number of variables) on the performance of the five clustering approaches (KM, NN, NK, GA, GK). I find that density, number of clusters, and number of dimensions are related to the clustering performance of all five approaches. The KM, NK, and GK classify well when all clusters contain similar number of observations (equal density) while GK outperform the KM on average. NN performs well when one cluster contains more observations than any other cluster (high density). All five approaches are at their peak performance when there are only two clusters in the data set. The performances degrade as the number of clusters and/or number of variables increases. In the clustering performance comparison, the two hybrid models perform at least as well as the KM even though the simulated environment favors the KM. The most crucial information, the true number of latent clusters, is provided to only the KM. In addition, the clusters structure is simple (the clusters have equal variance, equal number of observations and are well separated). Furthermore, there is relatively low correlation between all pairs of variables. Observations in each cluster are normally distributed. The two machine learning approaches (NN and GA) do not compete well in term of classification accuracy in the simulated problems since they are not given the true number of latent clusters. Thus, they are ranked the fourth and fifth in most scenarios.

The relative performances of the five clustering approaches are evaluated on three real problems with known natural cluster structure and one real problem with unknown natural cluster structure. Overall results indicate that the GK performs better than the KM while the NK is the worst among the five approaches (when natural structure exists). The two machine learning approaches generate relatively superior results in these problems (where an environment does not necessarily favor KM). In the first real problem with known natural cluster structure, the five clustering procedures are used to classify acquisition targets and bankruptcy firms. The GA and GK perform at least as well as the KM while there is no conclusive evidence that the KM outperform the NN and NK. The KM identifies more independent firms but fewer bankrupt and acquired firms than any other approach. In the second problem with known natural cluster structure (analysts' stock recommendations), the two machine learning approaches and the GK consistently outperform the KM. The GA and GK detect more firms with "buy" recommendation than any other approach. In the last problem with a known natural cluster structure (mutual funds' classifications), results are similar to the results reported in the analysts' stock recommendation problem. The KM performs worse than the two machine learning approaches and the GK.

In practice, information regarding cluster structure generally cannot be obtained prior to a cluster analysis. Therefore, we need an algorithm that performs relatively well regardless of environment. The GK has shown to be the best in simulated environment and the third best in real-life situations. Furthermore, the GK can detect firms with promising financial prospect such as acquisition targets and firms with "buy" recommendation than all other approaches. Thus, I would conclude that the GK is the best among the five approaches. I also attempted to uncover a latent cluster structure among dot-com companies using the GK. The GK recommends seven-cluster structure based on the pseudo-F statistics and eight-cluster structure based on CCC while KM and NN fail to recover similar cluster structure.

The results and conclusions reported in this study should be true for the problems only within the boundary of the parameters in the simulated and real problems evaluated in this study. For example, we find that the correct classification of the KM is lower in seven-cluster structure than in two-cluster structure. One should not conclude with certainty that the result of the KM in eight-cluster structure is better than it is in the nine-cluster structure since the eight- and nine-cluster structure are beyond the boundary of the parameters tested in this study. Such an inference is only based on extrapolation of my study results.

Future research may involve three different areas: effects of some error perturbations on these clustering approaches more sophisticated machine learning approaches in cluster analysis and a wider variety of applications. It is possible to investigate effects of some error perturbations on the five clustering approaches using Milligan's (1980) framework as a prototype. One factor that could be incorporated in the simulation process is the shape of the clusters. In this study, all clusters have similar shape, ellipsoidal in all dimensions. In a more advanced study, the shape of one or more clusters could be distorted. Another factor that may be worthy of investigation is misclassification cost. This factor could influence the outcomes dramatically, especially for the machine learning approaches since they provide less consistent results than does KM. In addition, the machine learning approaches tested in this study are in their simplest forms and could possibly be improved in the manner described in previous section. Yet they exhibit promising outcomes in the real problems. Finally, each real problem has a unique set of characteristics. It is relevant to investigate cluster structure of the real-life problems individually.

APPENDIX A

SAS CODE FOR SIMULATION

SAS CODE FOR SIMULATION

```
PROC IML;
      SEED = 0;
      N = 1764;
      SIGMA = {     0.09 0.00 0.00 0.00 0.00 0.00 0.00,
                    0.00 1.00 0.00 0.00 0.00 0.00 0.00,
                    0.00 0.00 1.00 0.00 0.00 0.00 0.00,
                    0.00 0.00 0.00 1.00 0.00 0.00 0.00,
                    0.00 0.00 0.00 0.00 1.00 0.00 0.00,
                    0.00 0.00 0.00 0.00 0.00 1.00 0.00,
                    0.00 0.00 0.00 0.00 0.00 0.00 1.00  };
      mu = {0,0,0,0,0,0,0,0};
      p  = nrow(sigma);
      m = repeat(mu`,n,1);
      g = root(sigma);
      z = normal(repeat(seed,n,p));
      y = z*g + m;
      print 'Multivariate Normal Sample';
      create simdat from Y;
      append from Y;
      close Simdat;


DATA FINA (KEEP=G X1--X7);
      SET SIMDAT;
      NUM=_N_;
      X1=COL2;
      X2=COL3;
      X3=COL4;
      X4=COL5;
      X5=COL6;
      X6=COL7;
      X7=COL8;
            IF X1 < -0.6 THEN X1 = -0.6;
            IF X2 < -2 THEN X2 = -2;
            IF X3 < -2 THEN X3 = -2;
            IF X4 < -2 THEN X4 = -2;
            IF X5 < -2 THEN X5 = -2;
            IF X6 < -2 THEN X6 = -2;
            IF X7 < -2 THEN X7 = -2;
            IF X1 >  0.6 THEN X1 =  0.6;
            IF X2 >  2 THEN X2 =  2;
            IF X3 >  2 THEN X3 =  2;
            IF X4 >  2 THEN X4 =  2;
            IF X5 >  2 THEN X5 =  2;
            IF X6 >  2 THEN X6 =  2;
            IF X7 >  2 THEN X7 =  2;
            X2=(2.5*X2)+5;
            X3=(2.5*X3)+5;
```

```
             X4=(2.5*X4)+5;
             X5=(2.5*X5)+5;
             X6=(2.5*X6)+5;
             X7=(2.5*X7)+5;
             IF NUM <= 252 THEN
                   DO;
                         G=1;
                         X1 = X1+1.25;
                         X2 = X2+0.25;
                         X3 = X3+0.25;
                         X4 = X4+0.25;
                         X5 = X5+0.25;
                         X6 = X6+0.25;
                         X7 = X7+0.25;
                   END;
             IF 252 < NUM <= 504 THEN
                   DO;
                         G=2;
                         X1 = X1+8.75;
                   END;
             IF 504 < NUM <= 756 THEN
                   DO;
                         G=3;
                         X1 = X1+5;
                   END;
             IF 756 < NUM <= 1008 THEN
                   DO;
                         G=4;
                         X1 = X1+7.5;
                   END;
             IF 1008 < NUM <= 1260 THEN
                   DO;
                         G=5;
                         X1 = X1+2.5;
                   END;
             IF 1260 < NUM <= 1512 THEN
                   DO;
                         G=6;
                         X1 = X1+3.75;
                   END;
             IF 1512 < NUM <= 1764 THEN
                   DO;
                         G=7;
                         X1 = X1+6.25;
                   END;
RUN;
```

APPENDIX B

EXAMPLES SAS CODE FOR 5 CLUSTERS WITH 5 VARIABLES

EXAMPLES SAS CODE FOR 5 CLUSTERS WITH 5 VARIABLES

B.1 The K-means Algorithm

```
PROC FASTCLUS DATA=SimDat MEAN=MeanKM OUT=KOUT MAXCLUSTERS=5
MAXITER=500;
    TITLE "K-means";
    VAR X1 X2 X3 X4 X5;
RUN;
```

B.2 The Neural Network

```
PROC NLP data=SimDat random= 50 outest=est out=OutNN1 maxiter= 500;
    array x[5] x1 x2 x3 x4 x5;
    array h[5]; ***hidden neurons;
    array a[5]; ***bias for input;
    array b[5,5]; ***wieghts between input and hidden;
    * c  bias for output in this case it is zero;
    array d[5]; ***wieghts between hidden and output;
    array p[5]; ***probability for group 1 and 2;
    array m[5,5];
    array r[5,5]; ***distance between an observation and its seed;

***** hidden layer -number of neuron in hidden layer equal to number of
clusters;
do ih=1 to 5;
    sum=a[ih];
    do ix=1 to 5;
        sum=sum+x[ix]*b[ix,ih];
    end;

***** logistic function for hidden neurons;
    h[ih]=exp(sum);
end;

***** output;
sum=0;
do ih=1 to 5; ******** ih is cluster number;
    sum=sum+h[ih]*d[ih];
end;

***** logistic function for output;
do ih =1 to 5;
```

```
    p[ih] = h[ih]/sum;
end;

***** Assign group membership;
q=1;
do ih=2 to 5;
  if p[q] < p[ih] then q=ih;
end;

***** residual;

do iq=1 to 5; ******** iq is cluster number;
    do ip=1 to 5; ******** ip is var number;
IF q = iq THEN r[ip,iq]=(x[ip]-m[ip,iq])** 2;
        ELSE IF q<>iq THEN r[ip,iq]=0;
     end;
end;
g=g;
Sumr=0;
do iq=1 to 5;
    do ip=1 to 5;
sumr=sumr+r[ip,iq];
    end;
end;

min sumr;

parms a1-a5 b1-b25 c d1-d5 m1-m25 ;
RUN;
```

B.3 The Genetic Algorithm

```
Title 'Genetic Algorithm';
%LET CHR = 10;
%LET cl = 5;
%LET VAR = 5;
%LET N_M = %eval(&CL*&VAR*&CHR);
%LET NumMu = %eval(&N_M/&CHR);
%LET Iteration = 50;
%LET Accu = 100;
%LET Quit = 9;


*********************Assigning the 1st Generation for GA;
DATA Seeds;
      ARRAY M[&CHR,&CL,&VAR];
      NUM=_N_;
      IF NUM > 1 THEN DELETE;
      DO ic=1 to &CHR;
            DO ig=1 to &CL;
                  DO ix=1 to &VAR;
                        M[ic,ig,ix]=10*RANUNI(0);
                  END;
            END;
      END;
      KEEP M1--M&N_M;
RUN;

%MACRO Upd_Seeds;
DATA FirstGen;
      IF _n_=1 THEN SET Seeds;
      SET SimDat;
      NUM=_n_;
      DROP n;
RUN;
%MEND Upd_Seeds;

%MACRO A_Clus;
DATA NextGen;
      SET FirstGen;
      ARRAY M[&CHR,&CL,&VAR];
      ARRAY X[&VAR];
      ARRAY R[&CHR,&CL];
      ARRAY CLUS[&CHR];
      ARRAY SmallR[&CHR];
      ARRAY TotR[&CHR];
      ARRAY INV[&CHR];
      ARRAY P_INV[&CHR];
      ARRAY TP_INV[&CHR];
* Find the best, worst, second best, and second worst Chromosomes;
      do ic=1 to &CHR; ********* ig is chromosome number;
            do iq=1 to &CL; ******** iq is cluster number;
            R[ic,iq]=0;
            end;
      end;
      do ic=1 to &CHR; ********* ig is chromosome number;
            do iq=1 to &CL; ******** iq is cluster number;
                  do ix=1 to &VAR; ******** iq is cluster number;
                  R[ic,iq]=((x[ix]-m[ic,iq,ix])** 2)+R[ic,iq];
                  end;
            end;
      end;
      do ic=1 to &CHR; ******** iq is chromosome number;
            CLUS[ic]=1;
                  SmallR[ic]=R[ic, 1];
      end;
      do ic=1 to &CHR; ********* ig is chromosome number;
            do iq=2 to &CL; ******** iq is cluster number;
            IF SmallR[ic]>R[ic,iq] THEN CLUS[ic]=iq;
                  IF SmallR[ic]>R[ic,iq] THEN SmallR[ic]=R[ic,iq];
            end;
      end;
      do ic=1 to &CHR; ********* ic is chromosome number;
            TotR[ic]+SmallR[ic];
      end;
RUN;
%MEND A_Clus;
```

```
%MACRO Worst_C;
DATA W_Chroms (DROP=X1--X&VAR);
      SET NextGen;
      ARRAY M[&CHR,&CL,&VAR];
      ARRAY X[&VAR];
      ARRAY R[&CHR,&CL];
      ARRAY CLUS[&CHR];
      ARRAY SmallR[&CHR];
      ARRAY TotR[&CHR];
      ARRAY INV[&CHR];
      ARRAY P_INV[&CHR];
      ARRAY TP_INV[&CHR];
      ARRAY Counter[&CHR];
      IF NUM=107 THEN
            DO;
                  DO ic=1 to &CHR;
                        counter[ic] = 1;
                  END;

                  DO ia=1 To &CHR;
                        DO ib=1 To &CHR;
                              IF ia NE ib Then
                                    Do;
                                          IF abs(TotR[ia]-TotR[ib]) <
&Accu THEN counter[ia] = counter[ia]+ 1;
                                    END;
                        END;
                  End;

                  count=1;
                  Do ic=1 to &CHR-1;
                        IF Counter[ic]>count THEN count=Counter[ic];
                  END;
                        DO ic=1 to &CHR;
                              INV[ic]=1/(TotR[ic]);
                        END;


      T_INV=INV1+INV2+INV3+INV4+INV5+INV6+INV7+INV8+INV9+INV10;

                        DO ic=1 to &CHR;
                              P_INV[ic]=100*INV[ic]/T_INV;
                        END;

                        TP_INV1=P_INV1/100;

                        DO ic=2 to &CHR;
                              TP_INV[ic]=TP_INV[ic-1]+P_INV[ic]/100;
                        END;

                        BEST1=1;

                        DO ic=1 to &CHR;
                              IF P_INV[BEST1]<P_INV[ic] THEN BEST1=ic;
                        END;
```

```
                         IF BEST1 = 1  THEN BEST2=2;  ELSE BEST2=1;

                         DO ic=2 to &CHR;
                               IF ic NE Best1 Then
                               IF P_INV[Best2]<P_INV[ic] THEN
Best2=(ic);
                         END;

                         Worst1=1;
                         DO ic=2 to &CHR;
                               IF P_INV[Worst1]>P_INV[ic] THEN
Worst1=ic;
                         END;

                         IF Worst1 = 1 THEN Worst2=2; ELSE WORST2=1;
                         DO ic=2 to &CHR;
                               IF ic NE Worst1 Then
                               IF P_INV[Worst2]>P_INV[ic] THEN
Worst2=(ic);
                         END;
                   END;
      *     END;
      ELSE DELETE;
RUN;
%MEND Worst_C;

%MACRO Cross0;
DATA W_C_CO;
      SET W_Chroms;
      ARRAY M[&CHR,&CL,&VAR];
      ARRAY TP_INV[&CHR];
      ARRAY INV[&CHR];
      ARRAY INVV[&CHR];
      ARRAY TP2_INV[&CHR];
      ARRAY NewC1[&CL,&VAR];
      ARRAY NewC2[&CL,&VAR];
IF count < &Quit THEN
DO;
*******************Randomly Select 2 Chromosomes;
      RN1=RANUNI(0); *************Random a number between 0 and 1;
      DO ic=1 to &CHR;************Find out the random number fall into
which chromosomes;
            IF TP_INV[&CHR-ic+1] > RN1 THEN RC1=(&CHR-ic+1);
      END;
      DO ic=1 to &CHR; *********Take out the value of the first selected
chromosome;
            IF ic = RC1 THEN INVV[ic]=0;
            ELSE INVV[ic]=INV[ic];
      END;
      **********************ReWeight the Prob;
      T2_INV=INVV1+INVV2+INVV3+INVV4+INVV5+INVV6+INVV7+INVV8+INVV9+INVV
10;
      TP2_INV1=INVV1/T2_INV;
      DO ic=2 to &CHR;
            TP2_INV[ic]=TP2_INV[ic-1]+INVV[ic]/T2_INV;
      END;
```

```
       **********Select the second chromosome;
       RN2=RANUNI(0);
       DO ic=1 to &CHR;
                    IF TP2_INV[&CHR-ic+1] > RN2 THEN RC2=(&CHR-ic+1);
       END;
*********************Randomize CrossOver Point;
       X_Randm=1+(((&VAR)-1)*RANUNI(0));
       X_Rand=ROUND (X_Randm, 1);
       CL_Randm=1+(((&CL)-1)*RANUNI(0));
       CL_Rand=ROUND (CL_Randm, 1);
*************************Start Crossover;
       IF CL_Rand > 1 THEN
       DO;
             DO iq=1 to (CL_Rand-1);
                   DO ix=1 to &VAR;
                         NewC1[iq,ix]=M[RC1,iq,ix];
                         NewC2[iq,ix]=M[RC2,iq,ix];
                   END;
             END;
       END;

       DO ix=1 to X_Rand;
                   NewC1[CL_Rand,ix]=M[RC1,CL_Rand,ix];
                   NewC2[CL_Rand,ix]=M[RC2,CL_Rand,ix];
       END;

       IF X_Rand < (&VAR+1) THEN
       DO;
             DO ix=(X_Rand+1) to &VAR;
                   NewC1[CL_Rand,ix]=M[RC2,CL_Rand,ix];
                   NewC2[CL_Rand,ix]=M[RC1,CL_Rand,ix];
             END;
       END;

       IF CL_Rand < (&CL+1) THEN
       DO;
             DO iq=(CL_Rand+1) to &CL;
                   DO ix=1 to &VAR;
                         NewC1[iq,ix]=M[RC2,iq,ix];
                         NewC2[iq,ix]=M[RC1,iq,ix];
                   END;
             END;
       END;
****************Substitute Two Worsts by Two New Chromosomes;
       DO iq=1 to &CL;
             DO ix=1 to &VAR;
                   M[WORST1,iq,ix]=NewC1[iq,ix];
                   M[WORST2,iq,ix]=NewC2[iq,ix];
             END;
       END;
END;
RUN;
%MEND CrossO;

%MACRO Mutat;
DATA CO_MU;
```

```
      SET W_C_CO;
      ARRAY M[&CHR,&CL,&VAR];
      ARRAY MuC[&NumMu];
      ARRAY MuQ[&NumMu];
      ARRAY MuX[&NumMu];
IF count < &Quit THEN
DO;
***************Start Mutation;
      Do i=1 to &NumMu;
            MuC[i]=1+(((&CHR)-1)*RANUNI(0));
            MuQ[i]=1+(((&CL)-1)*RANUNI(0));
            MuX[i]=1+(((&VAR)-1)*RANUNI(0));
            MC=ROUND (MuC[i], 1);
            MQ=ROUND (MuQ[i], 1);
            MX=ROUND (MuX[i], 1);
            M[MC,MQ,MX]=10-M[MC,MQ,MX];
      END;
END;
RUN;
%MEND Mutat;

%MACRO Invers;
DATA Inversion;
      SET CO_MU;
      ARRAY M[&N_M];
      ARRAY SM[&N_M];
IF count < &Quit THEN
DO;
************************Randomly Select Two Points;
            INVP1=1+(&N_M-1)*RANUNI(0);
            INVP2=1+(&N_M-1)*RANUNI(0);
            INVP1=ROUND (INVP1, 1);
            INVP2=ROUND (INVP2, 1);
***********Figure out Which is Start and Which is End Point;
      DIFF=INVP1-INVP2;
      IF DIFF>0 THEN
            DO;   INVP3=INVP1;
                  INVP1=INVP2;
                  INVP2=INVP3;
            END;
      NumInv=abs(INVP2-INVP1);
      MP_Inv=NumInv/2;
      MP_Inv=Round (Mp_Inv,  1);

***************************Swap Them;
      Do i=1 to MP_Inv-1;
            SM[INVP1+i-1]=M[INVP1+i-1];
            M[INVP1+i-1]=M[INVP2-i+1];
            M[INVP2-i+1]=SM[INVP1+i-1];
      END;
END;
RUN;
%MEND Invers;
%MACRO Replace;
DATA Seeds;
      SET Inversion;
      KEEP M1--M&N_M BEST1;
RUN;
%MEND Replace;
%MACRO GA_Seeds;
DATA SeedGA (KEEP=X1--X&NumMu);
      SET SEEDS;
      ARRAY M[&CHR,&CL,&VAR];
      ARRAY X[&CL,&VAR];
      DO iq=1 to &CL;
            DO ix=1 to &VAR;
                  X[iq,ix]=M[Best1,iq,ix];
            END;
      END;
%MEND GA_Seeds;
%MACRO Ite;
PROC PRINT DATA=Inversion;
      TITLE "This is the &iteration iteration";
      VAR BEST1 BEST2 WORST1 WORST2;
RUN;
%MEND Ite;

%MACRO MyGA;
      %DO i=1 %to &Iteration;
                  %Upd_Seeds
                  %A_Clus
                  %Worst_C
                  %CrossO
                  %Mutat
                  %Invers
                  %Replace;
      %END;
      %Ite
      %GA_Seeds
%MEND MyGA;

%MyGA
RUN;
```

REFERENCES

Agresti, A. 1990. *Categorical Data Analysis*, New York : Wiley.

Allen, L., and A. S. Cebenoyan. 1991. "Bank Acquisitions and Ownership structure: Theory and Evidence," *Journal of Banking and* Finance, vol. 15: 425-448.

Altman, E. I. 1968. "Financial Ratios, Discriminant Analysis and The Prediction of Corporate Bankruptcy," *Journal of Finance*, vol. 23(4): 589-609.

Altman, E. I., G. Marco, and F. Varetto. 1994. "Corporate Distress Diagnosis: Comparisons Using Linear Discriminant Analysis and Neural Networks (The Italian Experience)," *Journal of Banking and Finance*, vol. 18: 505-529.

Ambrose, B. W., and W. L. Megginson. 1992. "The Role of Asset Structure, Ownership Structure and Takeover Defenses in Determining Acquisition Likelihood," *Journal of Financial and Quantitative Analysis*, vol 27(4): 575-589.

Anderberg, M. R. 1973. *Cluster Analysis for Applications*. New York: Academic Press.

Archer, N. P., and S. Wang. 1993. "Application of the Back Propagation Neural Network Algorithm with Monotonicity Constraints for two-group Classification Problems," *Decision Sciences*, vol 24: 60-75.

Backer, E. 1995. *Computer-Assisted Reasoning in Cluster Analysis*. New Jersey: Prentice Hall.

Balakrishnan, P. V., M. C. Cooper, V. S. Jacob, and P. A. Lewis. 1994. "A Study of the Classification Capabilities of Neural Networks Using Unsupervised Learning: A Comparison with K-Means Clustering," *Psychometrika*, vol 59(4): 509-525.

________ . 1996. "Comparative Performance of the FSCL Neural Net and K-Means Algorithm for Market Segmentation," *European journal of operational research*, vol. 93: 346-357.

Banfield, C. F., and L. C. Bassil. 1977. "A Transfer Algorithm for Nonhierarchical Classification," *Applied Statistics*, vol. 26: 206-210.

Barber M. B., and D. Loeffler. 1993. "The "Dartboard" Column: Second-Hand Information and Price Pressure," *Journal of Financial and Quantitative Analysis*, vol. 28(2): 273-284.

Barnes, P. 1990. "The Prediction of Takeover Targets in the U.K. by Means of Multiple Discriminant Analysis," *Journal of Business Finance and Accounting*, vol. 7: 73-84.

Barrett, N. J., and I. F. Wilkinson. 1985. "Export Stimulation: A Segmentation Study of the Exporting Problems of Australian Manufacturing Firms," *European Journal of Marketing*, vol. 19(2): 53-72.

Bentz, Y., and D. Merunka. 2000. "Neural Networks and the Multinomial Logit for Brand Choice Modeling: A Hybrid Approach," *Journal of Forecasting*, vol. 19(3): 177-200.

Biles, W. E., A. S. Elmaghraby, and I. Zahran. 1991. "A Simulation Study of Hierarchical Clustering Techniques for the Design of Cellular Manufacturing Systems," *Computers & Industrial Engineering*, vol. 21(1): 267-271.

Brown, S. J., and W. N. Goetzmann. 1997. "Mutual Fund Styles" *Jounal of Financial Economics*, vol. 43: 373-399.

Calinski, T., and J. Harabasz. 1974. "A Drendrite Method for Cluster Analysis," *Communications in Statistics*, vol. 3: 1-27.

Chen, S. K., P. Mangiameli, and D. West. 1995. "The Comparative Ability of Self-Organizing Neural Networks to Define Cluster Structure," *Omega*, vol. 23(3): 271-279.

Cheng, D., B. E. Gup, and L. D. Wall. 1989. "Financial Determinants of Bank Takeovers," *Journal of Money, Credit, and Banking*, vol. 21(4): 524-536.

Chester, M. 1993. *Neural Networks: A Tutorial.* New Jersey: Prentice Hall.

Childress, M. 1981. "Statistics for Evaluating Classifications: A New View," presented at the Meeting of the Classification Society, Toronto, Canada.

Chiou, Y., and L. W. Lan. 2001. "Theory and Methodology: Genetic Clustering Algorithms," *European Journal of Operational Research*, vol 135: 413-427.

Church A. H., and Waclawski. 1998. "The Relationship Between Individual Personality Orientation and Executive Leadership Behaviour," *Journal of Occupational and Organizational Psychology*, vol. 71: 99- 126.

Cinca, C. S. 1996. "Self Organizing Neural Networks for Financial Diagnosis," *Decision Support Systems*, vol. 17: 227-238.

Cormack, R. M. 1971. "A Review of Classification," *Journal of the Royal Statistical Society (Series A)*, vol. 134: 321-367.

Cornelius, E. T. III, T. J. Carron, and M. N. Collins. 1979. "Job Analysis Models and Job Classification," *Personnel Psychology*, vol. 32(4): 693-693.

Cudd, M., and R. Duggal. 2000. "Industry Distributional Characteristics of Financial Ratios: An Acquisition Theory Application," *Financial Review*, vol. 41: 105-120.

Davies, D. L., and D. W. Bouldin. 1979. "A Cluster Separation Measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1: 224-227.

DiBartolomeo, D., and E. Witkowski. 1997. "Mutual Fund Missclassification: Evidence Based on Style Analysis," *Financial Analysts Journal*, vol. 53(5): 32-43.

Dietrich, J. K., and E. Sorensen. 1984. "An Application of Logit Analysis to Prediction of Merger Targets," *Journal of Business Research*, vol. 12 (1984): 393-402.

Dunn, G., and Brian S. Everitt. 1982. *An introduction to Mathematical Taxonomy.* Cambridge: Cambridge University Press.

Everitt, B. S. 2001. *Cluster Analysis. 4th ed.* London: Arnold Publishers.

Falkenauer, E. 1998. *Genetic Algorithms & Grouping Problems.* New York: Wiley.

Francis, J., & Soffer, L. 1997. "The Relative Informativeness of Analysts' Stock Recommendations and Earnings Forecast Revisions," *Journal of Accounting Research*, vol. 35(2), 193-211.

Garson, D. 1998. *Neural Networks.* London: Sage Publications.

Gen, M., and R. Cheng. 2000. *Genetic Algorithms & Engineering Optimization.* New York: Wiley.

Gilson, S. C. 2000. "Analysts and Information Gaps: Lessons from the UAL Buyout," *Financial Analysts Journal*, vol. 56(6), 82-118.

Givoly, D., & Lakonishok, J. 1984. "The Quality of Analysts' Forecasts of Earnings, *Financial Analysts Journal*, vol. 40(5), 40-47.

Glorfeld, L. W., and B. C. Hardgrave. 1996. "An improved Method for Developing Neural Networks: The case of evaluating Commercial Loan Creditworthiness," *Computers Operations Research*, vol. 23(10): 933-944.

Goldberg, D. E. 1989. *Genetic Algorithms in Search*. New York: Addison-Wesley.

Gordon, M. D. 1991. "User-Based Document Clustering by Redescribing Subject Descriptions With a Genetic Algorithm," *Journal of the American Society for Information Science*, vol. 42 (5): 311-323.

Green, P. E., and A. M. Krieger. 1995. "Alternative Approaches to Cluster-Based Market Segmentation," *Journal of the Marketing Research Society*, vol. 37(3): 221-239.

Green, P. E., C. M. Schaffer, and K. M. Patterson. 1988. "A Reduced-Space Approach to the Clustering of Categorical Data in Market Segmentation," *Journal of the Marketing Research Society*, vol. 30(3): 267-288.

Grinblatt, M., and S. Titman. 1989. "Mutual Fund Performance: An Analysis of Quarterly Portfolio Holdings," *Journal of Business*, vol. 62(3): 393-416.

Gupta, J. N. D. 1999. "Comparing Backpropagation with Genetic Algorithm for Neural Network Training," *Omega*, vol. 27(6): 679-684.

Hanson, R. C. 1992. "Tender Offers and Free Cash Flow: An Empirical Analysis," *The Financial Review*, vol. 27(2): 185-209.

Hartigan, J. A. 1977. "Distribution Problems in Clustering," Edited by Van Ryzin. *Classification and Clustering*, New York: University Press.

Harvey, R. J. 1986. "Quantitative Approaches to Job Classification: A Review and Critique," *Personnel Psychology*, vol. 39(2): 267-289.

Helsen, K., and P. E. Green. 1991. "A Computational Study of Replicated Clustering with an Application to Market Segmentation," *Decision Sciences*, vol. 22(5): 1124-1141.

Hecht-Nielsen, R. 1990. *Neurocomputing : The Technology of Non-Algorithmic Information Processing*, New York: Addison Wesley.

Hirschey, M., V. J. Richardson, and S. Scholz. 2000(a). "Stock-Price Effects of Internet Buy-Sell Recommendtions: The Motley Fool Case," *The Financial Review*, vol. 35: 147-174.

________ . 2000(b). "How "Foolish" are Internet Investors?," *Financial Analysts Journal*, vol. 56(1), 62-69.

Hofstede, G. 1998. "Identifying Organizational Subcultures: An Empirical Approach," *The Journal of Management Studies*, vol. 35 (1): 1-12.

Holland, J. H. 1992. *Adaptation in Natural and Artificial Systems. 1992 ed.* Cambridge: MIT Press.

Holland, J. H. 1992. "Genetic Algorithms," *Scientific American* vol. 267(12): 66-72.

Hopner, F., F. Klowan, R. Kruse, and T. Runkler. 1999. *Fuzzy Cluster Analysis*. New York: Wiley.

Hruschka, H., and M. Natter. 1999. "Comparing Performance of Feedforward Neural Nets and K-Means for Cluster-Based Market Segmentation," *European Journal of Operational Research*, vol. 114: 346-353.

Ignizio, James, and James Soltys. 1996. "Simultaneous Design and Training of Ontogenic Neural Network Classifiers," *Computers Operation Research*, vol. 23(6): 535-546.

Jain, B. A., and O. Kini. 1999. "The Life Cycle of Initial Public Offering Firms." *Journal of Business Finance and Accounting*, vol. 26(9): 1281-1307.

Jancey, R. C. 1966. "Multidimensional Group Analysis," *Australian Journal of Botany*, vol. 14(1): 127-130.

Jardine, N., and R. Sibson. 1971. *Mathematical Taxonomy*. Cambridge: Wiley&Son.

Johnson, R. A., and D. W. Wichern. 1988. *Applied Multivariate Statistical Analyse*. New Jersey: Prentice-Hall.

Kamrani, A. K., H. R. Parsaei, and M. A. Chaudhry. 1993. "A Survey of Design Methods for Manufacturing Cells," *Computers & Industrial Engineering*, vol. 25(1): 487-490.

Kane, G. D. 1998. "Rank Transformations and the Prediction of Corporate Failure," *Contemporary Accounting Research*, vol. 5(2): 145-166.

Kattan, M. W., and R. B. Cooper. 1998. "The Predictive Accuracy of Computer-Based Classification Decision Techniques: A Review and Research Directions," *Omega*, vol. 26(4): 467-482.

Khattree, R., and D. N. Naik. 1999. *Applied Multivariate Statistics with SAS Software*. New York: John Wiley & Sons.

Kim, M., R. Shukla, and M. Tomas. 2000. "Mutual Fund Objective Misclassification," *Journal of Economics and Business*, vol. 52: 309-323.

Klastorin, T. D. 1982. "An Alternative Method for Hospital Partition Determination Using Hierarchical Cluster Analysis," *Operations Research*, vol. 30(6): 1134-1147.

Kohonen, T. 2001. *Self-Organizing Maps.3rd ed.* Berlin:Springer-Verlag.

Krieger, A. M., and P. E. Green. 1996. "Modified Cluster-Based Segments to Enhance Agreement with An Exogenous Response Variable," *Journal of Marketing Research*, vol. 33(3): 351-363.

Krishnamurthy, A. K., S. C. Ahalt, D. E. Melton, and P. Chen. 1990. "Neural Networks for Vector Quantization of Speech and Images," *IEEE journal on selected areas in communications*, vol. 8(8): 1449-1457.

Lee, A., C. H. Cheng, and J. Balakrishnan. 1998. "Software Development Cost Estimation: Integrating Neural Network With Cluster Analysis," *Information & Management*, vol. 34: 1-9.

Lee, K. C., I. Han, and Y. Kwon. 1996. "Hybrid Neural Network Models for Bankruptcy Predictions," *Decision Support Systems*, vol. 18: 63-72.

Lin, Z. C., and C. Ho. 1996. "Application of Fuzzy Set Theory and Backpropagation Neural Networks in Progressive Die Design," *Journal of Manufacturing Systems*, vol. 15(4): 268-282.

MacQueen, J. 1967. "Some Methods for Classification and Analysis of Multivariate Observations," Edited by L. LeCam and J. Neymen *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, 281-997.

Mangiameli, P.,and D. West. 1999. "An Improved Neural Classification Network for the Two-Group Problem," *Computers & Operations Research*, vol. 26: 443-460.

Markham, I. S., and C. T. Ragsdale. 1995. "Combing Neural Networks and Statistical Predictions to Solve the Classification Problem in Discriminant Analysis," *Decision Sciences*, vol. 26(2): 229-242.

Michalewicz, Z. 1994. *Genetic Algorithms + Data Structures = Evolution Programs, 2nd Extended ed.* Berlin: Springer-Verlag.

Milligan, G. W. 1980. "An Examination of the Effect of Six Types of Error Perturbation on Fifteen Clustering Algorithms," *Psychometrika* vol. 45: 325-342.

________. 1981a. "A Monte Carlo Study of Thirty Internal Criterion Measures for Cluster Analysis," *Psychometrika*, vol. 46: 187-199.

________. 1981b. "A Review of Monte Carlo Tests of Cluster Analysis," *Multivariate Behavioral Research*, vol. 16: 379-407.

________. 1985. "An Algorithm for Generating Artificial Test Clusters," *Psychometrika*, vol. 50(1): 123-127.

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

________. 1987. "Methodology Review: Clustering Methods," *Applied Psychological Measurement*, vol. 11(4): 329-354.

Milligan, G. W., and L. M. Sokol. 1980. "A Two-Stage Clustering Algorithm With Robust Recovery Characteristics," *Educational and Psychological Measurement*, vol. 40: 755-759.

Milligan, G. W., S. C. Soon, and L. M. Sokol. 1983. "The Effect of Cluster Size, Dimensionality, and the Number of Clusters on Recovery of True Cluster Structure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5: 40-47.

Murty, M. N., and G. Krishna. 1981. "A Hybrid Clustering Procedure for Concentric and Chain-Like Clusters," *International Journal of Computer & Information Sciences*, vol. 10(6): 397-412.

Nevler, J. 1993. "Hitting the Limits of Neural Networks," *Wall Street & Technology*, vol. 11(7): 34-37.

Ng, M. K. and Z. Huang. 1999. "Data-Mining Massive Time Series Astronomical Data: Challenges, Problems and Solutions," *Information and Software Technology*, vol. 41(9):. 545-556.

O'Donnell, D. 1997. "Teaching Old Models Neural Tricks," *Bank Marketing*, vol. 29(8): 26-29,32.

Payne, T. H., L. Prather, and W. Bertin. 1999. "Value Creation and Determinants of Equity Fund Performance," *Journal of Business Research*, vol. 45: 69-74.

Pauler, G. 1999. "Development of a Neuro-Fuzzy Approach for Treating Multimodal Distribution and Spurious Clusters," *European Journal of Operational Research*, vol. 112(1): 207-220.

Pham, D. T., and D. Karaboga. 2000. *Intelligent Optimization Techniques. 2^{nd} ed.* London: Springer.

Pinter, J., and G. Pesti. 1991. "Set Partition by Globally Optimized Cluster Seed Points," *European Journal of Operational Research*, vol. 51: 127-135.

Punj, G., and D. W. Stewart. 1983. "Cluster Analysis in Marketing Research: Review and Suggestions for Application," *Journal of Marketing Research*, vol. 20(2): 134-148.

Quinlan, J. R. 1983. "Learning Efficient Classification Procedures and Their Application to Chess End Games," Edited by R. S. Michalski et al. *Machine Learning: An Artificial Intelligence Approach*, California: Tioga Pub. Co.

Quinlan, J. R. 1986. "Induction of Decision Trees," *Machine Learning*, vol. 1: 81--106.

Rand, W. M. 1971. "Objective Criteria for the Evaluation of Clustering Methods," *Journal of the American Statistical Association*, vol. 66: 846-850.

Reeves, C. 1995. "A Genetic Algorithm for Flowshop Sequencing," *Computers Operations Research*, vol. 22(1): 5-13.

Ritter, H., T. Martinetz, and K. Schulten. 1992. *Neural Computation and Self-Organizing Maps: An Introduction.* Reading: Addison-Wesley.

Sackett, P. R., E. T. III Cornelius, and T. J. Carron. 1981. "A Comparison of Global Judgement vs. Task Oriented Approaches to Job Classification," *Personnel Psychology*, vol. 34(4): 791-804.

Sarle, W. S. 1983. "Cubic Clustering Criterion," *SAS Technical Report A-108.* Cary: SAS Institute Inc.

Sarle, W. S. 1994. "Neural Network Implementation in SAS® Software," *Proceedings of the Nineteenth Annual SAS Users Group International Conference*, 1-28.

Scheibler, D., and W. Schneider. 1985. "Monte Carlo Tests of the Accuracy of Cluster Analysis Algorithms-A Comparison of Hierarchical and Nonhierarchical Methods," *Multivariate Behavioral Research*, vol. 20: 283-304.

Sexton, R. S., and R. E. Dorsey. 2000. "Reliable Classification using Neural Networks: A genetic algorithm and backpropagation comparison," *Decision Support Systems*, vol. 30(1): 11-22.

Sinclair, Steven A. and D. H. Cohen. 1992. "Adoption of Continuous Processing Technologies: Its Strategic Importance in Standardized Industrial Product-Markets," *Journal of Business Research*, vol. 24(3): 209-224.

Slater, S. F., and E. M. Olson. 2001. "Marketing's Contribution to the Implementation of Business Strategy: An Empirical Analysis," *Strategic Management Journal*, vol. 22(11): 1055-1067.

Sneath, P. H. A. 1968. "Evaluation of Clustering Methods," Edited by A. J. Cole. *Numerical Taxonomy.* London: Academic Press.

Sneath, P. H. A., and R. R. Sokal. 1973. *Numerical Taxonomy.* San Francisco: Freeman.

Sokal, R. R., and P. H. A. Sneath. 1963. *Principles of Numerical Taxonomy.* San Francisco: Freeman.

Solomon, H. 1977. "Data Dependent Clustering Techniques," Edited by J. Van Ryzin. *Classification and Clustering.* New York: University Press.

Soltys, J., J. Ignizio, and P.West. 1998. "Boundary Search Procedure for the design and Training of an Ontogenic Neural Network Classifier," *Computers Operation Research*, vol. 25(1): 19-29.

Somers, M. J. 1999. "Application of Two Neural Network Paradigms to the Study of Voluntary Employee Turnover," *Journal of Applied Psychology*, vol. 84(2): 177-185.

Spangler, W. E., J. H. May, and L. G. Vargas. 1999. "Choosing Data-Mining Methods for Multiple Classification: Representational and Performance Measurement Implications for Decision Support," *Journal of Management Information Systems*, vol. 16(1): 37-62.

Srinivasan, V., and Y. Kim. 1987. "Credit Granting: A Comparative Analysis of Classification Procedures," *Journal of Finance*, vol. 42(3): 665-683.

Stanley, K. L., Lewellen, W. G., & Schlarbaum, G. C. 1980. "Investor Response to Investment Research," *Journal of Portfolio Management*, vol. 6(4), 20-27.

Sung, T. K., N. Chang, and G. Lee. 1999. "Dynamics of Modeling in Data Mining: Interpretive Approach to Bankruptcy Prediction," *Journal of Management Information Systems*, vol. 16(1): 63-85.

Tam, K.Y. 1991. "Neural Network Models and The Prediction of Bank Bankruptcy," *Omega*, vol. 19(5): 429-445.

Tam, K.Y., and M. Y. Kiang. 1990. "Predicting Bank Failures: A Neural Network Approach," *Applied Artificial Intelligence*, vol. 4(4): 265-282.

________. 1992. "Management Applications of Neural Networks: The Case of Bank Failure Predictions," *Management Science*, vol. 38(7): 926-947.

Thompson, S. 1997. "Takeover Activity Among Financial Mutuals: An Analysis of Target Characteristics," *Journal of Banking and Finance*, vol. 21: 37-53.

Trippi, R. R., and E. Turban. 1993. *Neural Networks in Finance and Investing*. Chicago: Probus Publishing Co.

Tryon, R. C., and D. E. Bailey. 1970. *Cluster Analysis*. New York: McGraw-Hill.

Varetto, F. 1998. "Genetic Algorithms Applications in the Analysis of Insolvency Risk," *Journal of Banking & Finance*, vol. 22: 1421-1439.

Wang, S. 1995. "The Unpredictability of Standard Back Propagation Neural Networks in Classification Applications," *Management Science*, vol. 41(3): 555-559.

Wedel, M., and W. Kamakura. 1997. *Market Segmentation : Conceptual & Methodological Foundations*. Boston: Kluwer Academic Publishers.

Wong, M. A., and T. Lane. 1983. "A kth Nearest Neighbor Clustering Procedure." *Journal of the Royal Statistical Society* Series B, v45, 362-368.

Wu, F., and K. K. Yen. 1992. "Applications of Neural Network in Regression Analysis," *Computers and Industrial Engineering*, vol. 23(1): 93-95.