LINEAR GOAL PROGRAMMING IN CLASSIFICATION AND PREFERENCE DECOMPOSITION

by

Kim Fung Bruce Lam
B.B.A. (Hons.), Simon Fraser University, 1984
M.B.A., Simon Fraser University, 1986

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
in the Department
of
ECONOMICS

SIMON FRASER UNIVERSITY
November 1991

APPROVAL

Name: Bruce Lam

Degree: Ph.D. (Economics)

Title of Project: Linear Goal Programming in Classification and Preference Decomposition

Examining Committee:

Chairman: Dr. L. A. Boland

Dr. E. U. Choo
Associate Professor, Business Administration
Senior Supervisor

Dr. A. R. Warburton
Associate Professor, Business Administration

Dr. W. Wedley
Professor, Business Administration
Internal/External Examiner

Dr. A. Stam, Dept. of Mgmt. Sciences
University of Georgia
External Examiner

Date Approved: November 25, 1991

PARTIAL COPYRIGHT LICENSE

Title of Thesis/Project/Extended Essay

Linear Goal Programming in Classification and
Preference Decomposition

Author: ______________________
(signature)

Kim Fung Bruce Lam
(name)

November 25, 1991
(date)

ABSTRACT

Linear programming approaches in multivariate analysis have been studied by many researchers. In this work, linear programming approaches in classification and preference decomposition will be discussed. New and improved models in both classification and preference decomposition will also be introduced.

Classification addresses the problem of assigning objects to appropriate classes. The two main classes of classification approaches are cluster analysis and discriminant analysis. Cluster analysis concerns the 'grouping' of 'similar' objects to initially undefined classes. The main purpose of performing cluster analysis is data simplification. Discriminant analysis concerns the 'separation' of objects from several known populations. The primary purpose of performing discriminant analysis is to assign new objects to the correct population. Many methodş including heuristic approaches and statistical approaches have been proposed to solve the two classes of classification problems.

In cluster analysis, heuristic approaches do not guarantee optimal solutions with respect to any criterion. As a result, various mathematical programming models for cluster analysis have been developed. The advantage of mathematical programming models is that they provide optimal solutions for specific optimization objectives. I develop five new mathematical programming models in cluster analysis. A published data set is used to examine the performance characteristics of the cluster solutions obtained from my models and six other clustering procedures. Six criteria including five different measures of within cluster distances and one measure of between cluster distances are used to evaluate the cluster solutions obtained from the eleven models. All my models provide better cluster solutions than the heuristic approaches.

In discriminant analysis, the objects in the sample data set are from known populations. Popular statistical approaches to discriminant analysis are Fisher's discriminant function and logistic regression. Linear programming approaches in discriminant analysis are known to be robust and have been shown to have competitive performance when compared with the other statistical approaches. In this work, two new linear goal programming models in discriminant analysis are being introduced. The first model incorporates the within group discriminate information which is usually ignored by the other techniques in discriminant analysis. The classification performance of this model and several other popular discriminant techniques are evaluated by both an empirical study, which is based on the actual experience of an MBA admission committee, and a simulation experiment. The second model allows non-monotonic attributes to be included in the estimation process. In order to evaluate the classification performance of the second model and the other approaches in discriminant analysis with non-monotonic attributes, a simulation experiment is being conducted. Classification performance of the models in discriminant analysis are evaluated by the hit-ratio, the number of objects correctly classified. My models perform well in the above experiments.

Preference decomposition is a class of methods used to measure a respondent's multiattribute preference structure. Usually, a set of well selected alternatives is presented to a respondent who then expresses his/her preferences for the alternatives in terms of either nonmetric or metric measurement. Then, based on the overall preferences of the alternatives, preference decomposition is used to estimate the individual's preference function. I introduce a linear goal programming model for preference decomposition with the input preference judgments measured in ratio scale. In order to compare the predictive validity of my model and two other models, LINMAP and Ordinary Least Squares, a simulation experiment is being conducted. Performances of the models are evaluated by both the Pearson correlation coefficients and Spearman rank coefficients between the input preferences and the derived preferences of the alternatives. Both my model and Ordinary Least Squares have higher average Pearson correlation coefficients and average Spearman rank coefficients than LINMAP in this experiment, while the coefficients of my model and Ordinary Least Squares are close to each other.

In summary, I introduce several new linear goal programming models to solve the problems in cluster analysis, discriminant analysis, and preference decomposition. The five new models in cluster analysis provide additional optimization objectives to modelers in cluster analysis. This increase in the flexibility in choosing alternate optimal objectives in cluster analysis will likely motivate more frequent applications of mathematical programming approaches to cluster analysis. My first model in discriminant analysis incorporates the within group discriminate information. Intuitively, this model may provide more accurate estimations of the attribute weights than the other techniques which ignore the within group discriminate information. My second model has the ability to incorporate non-monotonic attributes in discriminant analysis. The existing linear programming approaches in preference decomposition only allow the input preference judgments to be measured in ordinal scale. Theoretically, ratio scale preference judgments contain more information than ordinal scale preference judgments. Consequently, I introduce a linear goal programming model which allows the input preference judgments to be measured in ratio scale.

TABLE OF CONTENTS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The goal programming model was first proposed by Charnes, Cooper, and Ferguson [1955]. Their work has motivated many applications of using linear programming techniques in multivariate analysis including multiple regression, cluster analysis, discriminant analysis, and preference decomposition [e.g., Wagner, 1959; Vinod, 1969; Rao, 1971; Srinivasan and Shocker, 1973; Freed and Glover, 1981a]. In this work, linear goal programming approaches to problems in classification and preference decomposition are being studied. The two main types of classification techniques are cluster analysis and discriminant analysis. The later concerns the 'separation' of objects from several known populations while the former concerns the grouping of 'similar' objects to initially undefined classes. Preference decomposition is a class of methods used to measure an individual's multiattribute preference structure given the individual's relative preferences of some chosen alternatives. New and improved models in both classification and preference decomposition will be introduced.

1.1 CLUSTER ANALYSIS

Cluster analysis has been applied in many fields of scientific inquiry, in business, in industry, and in education. Many heuristic methods have been proposed to solve the clustering problems. Unfortunately, the heuristic methods do not guarantee optimality with respect to any criterion. As a result, various mathematical programming models for cluster analysis have been developed. The objective functions of mathematical programming for cluster analysis are usually defined as measures of either "within cluster distances" or "between cluster distances". With an analog of multicriteria optimization framework, I provide a systematic way of generating a whole array of meaningful criteria for cluster analysis. We also introduce five new mathematical programming models and reformulate two existing models in chapter 2. A published data set is used to examine the performance characteristics of my models and several other popular clustering procedures.

1.2 DISCRIMINANT ANALYSIS

Recently, linear programming approaches to discriminant problems have been widely studied and have been shown to yield satisfactory results. In chapter 3, I introduce two new linear goal programming models in discriminant analysis. The first model works directly with the conditional probabilities of group membership rather than the simple group category to which a member belongs. With the use of conditional probabilities, the strength of group membership is measured and further discrimination within group is possible. Both an empirical data set which is obtained from an M.B.A. admission committee and simulated data are used to test the effectiveness of the above linear goal programming model and the other popular classification methods for discriminant analysis in minimizing misclassification errors. Since the primary purpose of performing discriminant analysis is to assign new objects to the correct population, the performance of the methods are evaluated by the hit ratio, the number of correctly classified objects.

Classification of objects are based on their overall scores computed from the classification function. However, according to the classification function, the higher the attribute score of an object, all other factors being equal, the higher (if this attribute has a positive weight in the classification function) or the lower (if this attribute has a negative weight in the classification function) is its overall score. This implied monotonicity is not reasonable in many situations. The second model I introduce is a linear goal programming model which has the ability to incorporate non-monotonic attributes in discriminant analysis. A simulation experiment is conducted to examine the effectiveness of this goal programming approach to classification problems with and without non-monotonic attributes.

1.3 PREFERENCE DECOMPOSITION

In choosing the best alternative with respect to multiple criteria, it is a common practice to determine the criterion weights which influence the preference of all the alternatives presented. The criteria are aggregated into a single preference function and the alternative with the highest preference function value is identified as the optimal choice. The preference function values can also be used to rank order all the alternatives. However, many decision makers are able to make preference judgments based on the alternative as a whole without any detailed criteria trade-off. To understand the choice behavior of these decision makers, it is pertinent to evaluate how they value the alternatives' criterion levels. Preference decomposition is a class of methods used to decompose the overall preference of alternatives into part-worth evaluations of criterion levels. Given the input judgments of preference among some carefully selected alternatives, the additive preference decomposition estimates a part-worth value for each level of each criterion. The sums of the part worths which correspond as closely as possible to the input preferences are used to determine the preferences among all alternatives. Theoretically, ratio scale preference judgments contain more information than ordinal scale preference judgments. Consequently, in chapter 4, I introduce a linear goal programming model for preference decomposition where the input preference judgments are measured in ratio scale. I conduct a simulation experiment to compare the predictive validity of my model and two other models namely LINMAP and Ordinary Least Squares in estimating the preference function. Performance of the models are evaluated by both the Pearson correlation coefficients and Spearman rank coefficients between the input preferences and the derived preferences of the alternatives. Both my model and Ordinary Least Squares have higher average Pearson correlation coefficients and average Spearman rank coefficients than LINMAP in this experiment, while the coefficients of my model and Ordinary Least Squares are close to each other.

CHAPTER 2

MATHEMATICAL PROGRAMMING APPROACHES TO CLUSTER ANALYSIS

2.1 INTRODUCTION

The problem of cluster analysis is to partition a given set of objects into certain appropriate classes such that it is optimal with respect to a certain chosen criterion function. Many heuristic methods have been proposed to solve the cluster analysis problem. Unfortunately, the heuristic methods do not guarantee optimality with respect to any criterion. As a result, various mathematical programming models for cluster analysis have been developed. In this chapter, I focus the discussion on the mathematical programming approaches. In section 2.4, with an analog of multicriteria optimization framework, I provide a systematic way of generating a whole array of meaningful criteria for cluster analysis. In section 2.5, I introduce five new mathematical programming models, two of which are bicriterion formulations. I also reformulate two existing mathematical programming models which utilize fewer variables than the two existing models. Computational results from a published data set in cluster analysis are presented in section 2.6.

2.2 THE PROBLEM OF GROUPING

Problems of cluster analysis involve grouping a certain number of objects into a certain number of clusters. Given a sample of n objects, the most general problem in cluster analysis is to group the n objects into clusters where the number of clusters is not known. Many heuristic methods have been proposed to solve the cluster analysis problems. The single linkage, complete linkage, group average linkage, and Ward's Method are some popular bottom-up hierarchic methods used in cluster analysis [Blashfield and Aldenderfer, 1978; Gordon, 1981; Johnson and Wichern, 1988]. In section 2.3, hierarchic methods will be discussed. The hill-climbing and the k-means methods [MacQueen, 1967] are two iterative reassigning procedures. Unfortunately, the heuristic approaches do not guarantee global optimality with respect to any criterion. The inadequacy of the heuristic methods has provoked the serious consideration of mathematical programming approaches to cluster analysis [Vinod, 1969; Rao, 1971; Arthanari and Dodge 1981; Aronson and Klein, 1989]. The advantage of mathematical programming models is that they provide optimal solutions to specific optimization objectives. This is most useful when an appropriate criterion for clustering exists. For example, mathematical programming approaches have been successfully applied to many location problems.

In cluster analysis, it is essential to define the distance between each pair of objects in the data set. The distances should reflect the "similarity" between pairs of objects. Usually the objects within clusters are more "similar" than objects in different clusters. Each object is typically described by a vector of attribute values. The "similarity" between objects is measured in terms of some distance metrics defined on the attribute space. Let n be the number of objects in a data set, p be the number of attributes, X be a nxp matrix which represents the attribute values of the n objects in the p dimensional space, and D be the distance matrix derived from X where d_{ij} is defined as the distance between the i-th object and the j-th object. The followings are the distances that are commonly used:

i) City block metric: $d_{ij}= \sum_{k=1}^{p} |x_{ik}-x_{jk}|$

ii) Euclidean metric: $d_{ij}= \left[\sum_{k=1}^{p} |x_{ik}-x_{jk}|^{2} \right]^{1/2}$

iii) Minkowski metric: $d_{ij}= \left[\sum_{k=1}^{p} |x_{ik}-x_{jk}|^{r} \right]^{1/r}$, r is a positive integer.

iv) Weighted Minkowski distance: $d_{ij}= \left[\sum_{k=1}^{p} W_{k}|x_{ik}-x_{jk}|^{r} \right]^{1/r}$, r is a positive integer.

v) Weighted Tchebycheff metric: $d_{ij}= \max_{1\le k\le p} w_{k}|x_{ik}-x_{jk}|$.

2.3 HIERARCHIC METHODS

The agglomerative approach is a popular class of hierarchic methods. Agglomeration starts with n clusters with each cluster contains only one object. Then at each stage, two closest clusters are merged to form one cluster until finally all objects are grouped into a single cluster. Different agglomerative approaches differ by the ways they define the distance between two clusters. The following are some distances between two clusters defined by several common agglomerative hierarchic methods:

Single linkage: It is also called the nearest neighbor method. The distance between two clusters is defined as the distance between their nearest members. At each stage, the two clusters with the smallest distance are merged.

Complete linkage: It is also called the furthest neighbor method. The distance between two clusters is defined as the distance between their furthest members.

Group average linkage: The distance between two clusters is defined as the average of the distances between all pairs of objects in the two clusters.

Wards' Method: It is also called the sum of squares method. In this method two clusters are merged if the amalgamation leads to the minimum increase in the total within cluster sum of squares at each stage.

Hierarchic methods are easy to implement but may not produce optimal solutions.

2.4 SYSTEMATIC WAYS OF GENERATING CRITERIA FOR CLUSTER ANALYSIS

Let m be the number of clusters embedded in the sample. Many methods have been proposed to determine the number of hidden clusters in a data set. Milligan and Cooper [1985] have provided a comparison study of thirty such procedures. However, when m can be prespecified, the remaining task is to identify the clusters according to some specific criteria.

There are many reasonable criteria which can be used in forming clusters. Thus they lead to different objective functions in the mathematical programming approaches. The structure in multicriteria optimization can be used to develop a systematic way of generating meaningful criterion functions for clustering problems. Suppose n objects have been grouped into m clusters. It is logical to evaluate the m clusters as follows:

For each k=1,2,...,m, let f_k be some measure of "deficiency" of the k-th cluster. Typically, f_k measures the "dissimilarity" of all the objects in the k-th cluster. Alternatively, f_k may measure the "closeness" from the k-th cluster to other clusters. It is obvious that we want to "minimize" $F\equiv(f_1,f_2,...,f_m)$, which is a multicriteria optimization problem. A common approach to solve this multicriteria optimization problem is to "combine" $f_1,f_2,...,f_m$ into one single overall objective function. Let g be a real valued function defined on the m-dimensional space of f_k values, k=1,2,...,m, such that $g(f_1, f_{2,} ..., f_m)$ measures the overall "badness" of $f_1, f_2, \ldots, f_m$. Then the problem can be solved by minimizing $g(f_1, f_2, \ldots, f_m)$. There are many acceptable ways to define the cluster "deficiency" $f_1, f_2, \ldots, f_m$ and the overall value function g [Chankong and Haimes 1983]. In this paper, I consider the weighted sum (WS), $g(f_1, f_2, \ldots, f_m) \equiv \sum_{k=1}^{m} w_k f_k$, and the Tchebycheff norm (TF), $g(f_1, f_2, \ldots, f_m) \equiv \text{MAX } \{w_1 f_1, \ldots, w_m f_m\}$, where $w_1, w_2, \ldots, w_m$ are positive numbers which reflect the relative importance of the m clusters respectively. To avoid considering too many different forms of $g(f_1, f_2, \ldots, f_m)$, seven ways of defining $f_1, f_2, \ldots, f_m$ are considered:

a. SDM: Sum of distances from objects to cluster medians,

b. MDM: Maximum distance from objects to cluster medians,

c. SDC: Sum of distances from objects to cluster centers,

d. SWD: Sum of within cluster pairwise distances,

e. MWD: Maximum within cluster pairwise distance,

f. MSW: Maximum sum of within cluster distances from an object,

g. NSB: -(Minimum sum of between cluster distances from an object)

Note that NSB is defined with negation so that f_k is to be minimized. There are 14 (=7x2) different forms for $g(f_1, f_2, \ldots, f_m)$. For convenience, SDM-WS is used to denote $g(f_1, f_2, \ldots, f_m)$ when $f_1, f_2, \ldots, f_m$ are sum of distances from objects to cluster medians (SDM) and g is the weighted sum (WS) function. Theoretically, each form of $g(f_1, f_2, \ldots, f_m)$ can be taken as the objective function to be minimized in a mathematical programming model for cluster analysis. Furthermore, we may combine two or more objective functions together and solve the problem as a new multicriteria problem. Thus many cluster analysis models can be generated. I will introduce two bicriterion models in the next section. Among the 14 basic models, the following have already been studied:

1. Vinod [1969] and Aronson & Klein [1989] use SDM-WS.
2. Rao [1971] uses SWD-WS.
3. Rao [1971] uses MWD-TF.

Discussing all fourteen models will be too lengthy and some of the models may not be as intuitively attractive as the other models.

For example, both MDM and MWD measure "maximum distance" in a cluster. If we use WS for $g(f_1,f_2,\ldots,f_m)$, then in some optimal solutions, some clusters may have large "maximum distances" while other clusters may have small "maximum distances". These results may be difficult to interpret. It is more meaningful to use TF for $g(f_1,f_2,\ldots,f_m)$ such that $g(f_1,f_2,\ldots,f_m)$ can be interpreted as the "overall maximum distance" in all the clusters. Therefore, for illustration purpose, I only look at one model for each of the seven definitions of f_k.

The fundamental question in cluster analysis is what makes a good cluster solution. Evaluation of cluster solutions must, therefore, be based on some meaningful criteria. The different forms for $g(f_1,f_2,\ldots,f_m)$ can be used as criteria in evaluating the cluster solutions. In Section 2.6, we will compare the cluster solutions obtained by different clustering models based on the values of these criteria.

2.5 MATHEMATICAL PROGRAMMING APPROACHES TO CLUSTER ANALYSIS

For all the models discussed in this section, I use the following variables for cluster memberships:

$$z_{ik} = \begin{cases} 1 & \text{if the i-th object belongs to the k-th cluster} \\ 0 & \text{otherwise} \end{cases}$$

for i=1,...,n and k=1,...,m. For models (SDM-WS) and (MDM-TF), the variables used to identify the medians are defined as follows:

$$y_{ik} = \begin{cases} 1 & \text{if the i-th object is the median of the k-th cluster} \\ 0 & \text{otherwise} \end{cases}$$

for i=1,...,n, k=1,...,m. Since the models (SM) [Vinod, 1969] and (SW) [Rao, 1971] use different variables, I reformulate their models in this section.

2.5.1 Cluster-Median Problem

One of the objective functions for the mathematical programming methods in cluster analysis is to minimize the total sum of distances from each object to the "representative" point (cluster median) of the cluster in which the object belongs. If it is restricted that only the n sample points can be chosen as the cluster-medians, then this problem is usually called the cluster-median problem [Vinod 1969]. This objective function shares the same or similar objectives of some location problems [Revelle, Marks and Liebman, 1970; Ghosh and Craig, 1986; Church, Current and Storbeck, 1991]. For example, if we wanted to build three shopping malls to serve twenty suburban areas, we would like to build the shopping malls in certain locations such that the total sum of distances travelled from the twenty areas to their nearest shopping malls is minimized.

Vinod [1969] formulated the cluster-median problem as an integer programming problem (SM) (the formulation is in Appendix 2.1). I reformulate (SM) as follows:

$$\text{(SDM-WS)}\quad \text{MIN} \sum_{i=1}^{n} h_i \qquad (2.1)$$

$$\text{S.T} \sum_{j=1}^{n} d_{ij}y_{jk} - M(1-z_{ik}) - h_i \le 0, \quad \text{for } k=1,\ldots,m \;\; i=1,\ldots,n \qquad (2.2)$$

$$\sum_{k=1}^{m} z_{ik} = 1, \quad \text{for } i=1,\ldots,n \qquad (2.3)$$

$$\sum_{j=1}^{n} y_{jk} = 1, \quad \text{for } k=1,\ldots,m \qquad (2.4)$$

$$h_i \ge 0,\; y_{ik} \text{ zero-one and } z_{ik} \text{ zero-one.}$$

M is a large positive number. When z_{ik} equals to zero, the corresponding constraint in (2.2) becomes redundant. When z_{ik} equals to one, h_i is forced to equal to d_{ij} where point j is the median (y_{jk}=1) of cluster k. Constraints in (2.3) enforce that each object must be belonged to one cluster. Constraints in (2.4) identify exactly m medians.

2.5.2 The Problem of Minimizing the Overall Maximum Distance from an Object to Its Median

Considering the cluster median problem, a different objective is to minimize the overall maximum distance from an object to its median. I formulate this problem as follows:

$$\text{(MDM-TF)} \quad \text{MIN} \quad H \qquad (2.5)$$

$$\text{S.T.} \quad \sum_{j=1}^{n} d_{ij} y_{jk} - M(1-z_{ik}) - H \le 0, \quad \text{for } k=1,\ldots,m \;\; i=1,\ldots,n \qquad (2.6)$$

$$\sum_{k=1}^{m} z_{ik} = 1, \quad \text{for } i=1,\ldots,n \qquad (2.7)$$

$$\sum_{j=1}^{n} y_{jk} = 1, \quad \text{for } k=1,\ldots,m \qquad (2.8)$$

$$H \ge 0, \; y_{ik} \text{ zero-one}, \; z_{ik} \text{ zero-one}.$$

In (2.6), the variable H captures the overall maximum distance from an object to its median. Let the optimal objective value of (MDM-TF) be H^*. Since H^* equals to the overall maximum distance from an object to its median, multiple solutions are likely to exist. Multiple solutions exist when the distances from an object (or several objects) to two or more medians are less than or equal to H^*. If this occurs, then this object (these objects) can be assigned to different clusters without affecting the value of H^*. However, among the multiple solutions, some solutions may be more preferred than the others. In order to choose the "best" clustering solution among the multiple solutions, I formulate the problem as the following bicriterion problem:

$$\text{(MDM-TF-B) MIN} \quad P_1 H + P_2 (\sum_{i=1}^{n} h_i) \tag{2.9}$$

$$\text{S.T.} \quad \sum_{j=1}^{n} d_{ij} y_{jk} - M(1-z_{ik}) - h_i \le 0, \quad \text{for } k=1,\dots,m \quad i=1,\dots,n \tag{2.10}$$

$$H - h_i \ge 0, \qquad \text{for } i=1,\dots,n \tag{2.11}$$

$$\sum_{k=1}^{m} z_{ik} = 1, \qquad \text{for } i=1,\dots,n \tag{2.12}$$

$$\sum_{j=1}^{n} y_{jk} = 1, \qquad \text{for } k=1,\dots,m \tag{2.13}$$

$$H \ge 0, \; h_i \ge 0, \; y_{ik} \text{ zero-one } z_{ik} \text{ zero-one.}$$

In (MDM-TF-B), P_1 is preemptive over (i.e. much larger than) P_2 and H is forced to be equal to the maximum value of h_i in (2.11). The first priority goal of (MDM-TF-B) is the same as the objective in (MDM-TF), while the second priority goal is the same as the objective in (SDM-WS).

2.5.3 The Problem of Minimizing the Sum of Distances from Objects to the 'Centers'

For the cluster median problem, there is no intuitive reason to restrict the medians to be located on the sample points only. A more general problem is to relax this restriction by allowing any point in the p-dimensional attribute space to be the medians. Revelle, Marks and Liebman [1970] classified this type of problem as "Location on a Plane" in location analysis. It allows for more flexibility in the locations of the medians. I formulate this problem as a mixed-integer programming problem (SDC-WS) in the next section.

Consider the cluster-median problem where any point in the p-dimensional attribute space can be selected as a cluster median. Let $(b_{k1},\dots,b_{kp})$ be the k-th median. Using city-block distance to define the distances between the sample points and the medians, this problem can be formulated as follows:

$$\text{(SDC-WS) MIN} \quad \sum_{i=1}^{n} \sum_{j=1}^{p} h_{ij} \tag{2.14}$$

$$\text{S.T.} \quad x_{ij} - b_{kj} - M(1-z_{ik}) - h_{ij} \le 0, \quad \text{for } i=1,\dots,n;\ j=1,\dots,p;\ k=1,\dots,m \tag{2.15}$$

$$b_{kj} - x_{ij} - M(1-z_{ik}) - h_{ij} \le 0, \quad \text{for } i=1,\dots,n;\ j=1,\dots,p;\ k=1,\dots,m \tag{2.16}$$

$$\sum_{k=1}^{m} z_{ik} = 1, \quad \text{for } i=1,\dots,n \tag{2.17}$$

$$b_{kj} \ge 0,\ h_{ij} \ge 0,\ z_{ik} \text{ zero-one.}$$

Constraints (2.15) and (2.16) are linear forms of the following conditions:

$$\left| x_{ij} - b_{kj} \right| - M(1-z_{ik}) - h_{ij} \le 0, \quad \text{for } i=1,\dots,n;\ j=1,\dots,p;\ k=1,\dots,m \tag{2.18}$$

Although (SDC-WS) allows for more flexibility in the location of the medians, with large values of n, m, and p, (SDC-WS) will be a large mixed integer programming problem which is difficult to solve. Hence, (SDC-WS) is more suitable to solve small to moderate size problems. Nevertheless, with small size problems, there is an apparent advantage of (SDC-WS) over (SDM-WS). If the locations of the medians can be chosen only from the sample points, then the smaller the sample size, the less will be the possible locations to locate the medians. As a result, choosing good "representative" medians for the clusters became a more difficult task in (SDM-WS). However, the number of possible locations in (SDC-WS) is still infinite even when the sample size is very small.

2.5.4. The Problem of Minimizing Within Cluster Distances

Another objective is to minimize the total sum of pairwise distances between objects in the same cluster. Rao [1971] developed a zero-one integer programming model (SW) for this problem (the formulation is in the Appendix 2.1). However, (SW) is difficult to solve because it has too many zero-one variables and constraints. Aronson and Klein [1989] provided an improved mixed-integer programming model, (SW1), which contained fewer variables than Rao's model (the formulation is in the Appendix 2.1). They applied their model in the area of computer-assisted process organization for information development.

I reformulate this problem as a mixed-integer programming problem (SWD-WS) which has fewer variables and constraints than both (SW) and (SW1). My model is as follows:

$$\text{(SWD-WS) MIN} \quad \sum_{i=1}^{n} h_i \qquad (2.19)$$

$$\text{S.T.} \quad \sum_{j=1}^{n} d_{ij} z_{jk} - M(1-z_{ik}) - h_i \le 0, \quad \text{for } i=1,\dots,n, \; k=1,\dots,m \qquad (2.20)$$

$$\sum_{k=1}^{m} z_{ik} = 1, \qquad \text{for } i=1,\dots,n \qquad (2.21)$$

$$h_i \ge 0, \; z_{ik} \text{ zero-one.}$$

In (2.20), when z_{ik} equals to one, h_i measures the sum of distances of object i to the other objects in the same cluster. If z_{ik} equals to zero, then the corresponding constraint becomes redundant. The number of variables and number of constraints in (SWD-WS) are both equaled to mn+n, and are much less than (SW) and (SW1).

2.5.5 The Problem of Minimizing the Overall Maximum Within Cluster Distance

Rao [1971] introduced a zero-one integer programming model (MWD-TF) to solve the cluster problem by minimizing the overall maximum pairwise distance within cluster. The formulation is as follows:

$$\text{(MWD-TF) MIN} \quad H \tag{2.22}$$

$$\text{S.T.} \quad d_{ij}z_{ik} + d_{ij}z_{jk} - H \le d_{ij}, \quad \text{for } i=1,\dots,n-1;\ j=i+1,\dots,n;\ k=1,\dots,m \tag{2.23}$$

$$\sum_{k=1}^{m} z_{ik} = 1, \quad \text{for } i=1,\dots,n \tag{2.24}$$

$$H \ge 0, \ z_{ik} \text{ zero-one.}$$

If both z_{ik} and z_{jk} equal to one, then H is forced to be greater than or equal to d_{ij} in (2.23), or H must be equal to the largest pairwise distance in the same cluster. However, similar to the problem of (MDM-TF) in section 2.5.2, multiple solutions may exist. Since if the distances from an object to the objects in more than one cluster are less than or equal to H^*, then this object can be assigned to any of those clusters without affecting the value of H^*. I formulate a new model (MWD-TF-B) which chooses the 'best' clustering solution among the multiple solutions is as follows:

$$\text{(MWD-TF-B) MIN} \quad P_1H + P_2(\sum_{i=1}^{n} h_i) \tag{2.25}$$

$$\text{S.T.} \quad d_{ij}z_{ik} + d_{ij}z_{jk} - H \le d_{ij}, \quad \text{for } i=1,\dots,n-1;\ j=i+1,\dots,n;\ k=1,\dots,m \tag{2.26}$$

$$\sum_{j=1}^{n} d_{ij}z_{jk} - M(1-z_{ik})-h_i \le 0, \quad \text{for } i=1,\dots,n;\ k=1,\dots,m \tag{2.27}$$

$$\sum_{k=1}^{m} z_{ik} = 1, \quad \text{for } i=1,\dots,n \tag{2.28}$$

$$H \ge 0, \ h_i \ge 0, \ z_{ik} \text{ zero-one.}$$

P_1 is preemptive over P_2. (MWD-TF-B) is an improved version of (MWD-TF). The first priority goal of (MWD-TF-B) is to minimize the overall maximum within cluster distance, same as the objective in (MWD-TF), while the second priority goal is to minimize the sum of pairwise distances as in (SWD-WS).

2.5.6 The Problem of Minimizing the Maximum Sum of Distances from One Object to Other Objects Within Cluster

In this section, I define the "dissimilarity" measured in a cluster as the maximum sum of distances from one object to the other objects within the same cluster. The objective function of the corresponding model is to minimize the sum of the "dissimilarity" measured in all the clusters. I formulate this problem as follows:

$$\text{(MSW-WS) MIN} \quad \sum_{k=1}^{m} h_k \qquad (2.29)$$

$$\text{S.T.} \quad \sum_{j=1}^{n} d_{ij}z_{jk} - M(1-z_{ik}) - h_k \le 0, \quad \text{for } i=1,\ldots,n \;\; k=1,\ldots,m \qquad (2.30)$$

$$\sum_{k=1}^{m} z_{ik} = 1, \quad \text{for } i=1,\ldots,n \qquad (2.31)$$

$$h_k \ge 0, \; z_{ik} \text{ zero-one.}$$

In (2.30), h_k is forced to be greater than or equal to the maximum sum of within group distances. Consequently, h_k captures the sum of the within group distances to the worst cluster median. The sum of h_k over all clusters is minimized.

2.5.7 The Problem of Maximizing the Minimum Sum of Distances from One Object to the Other Objects in Different Clusters

Another possible objective in cluster analysis is to maximize the overall minimum sum of distances from one object to all the other objects in different clusters. I formulate this problem as follows:

$$\text{(NSB-TF) MIN} \quad -H \qquad (2.32)$$

$$\text{S.T.} \quad \sum_{j=1}^{n} d_{ij}z_{jk} - M(1-z_{ik}) + H \le \sum_{j=1}^{n} d_{ij}, \quad \text{for } i=1,\ldots,n \;\; k=1,\ldots,m \qquad (2.33)$$

$$\sum_{k=1}^{m} z_{ik} = 1, \quad \text{for } i=1,\ldots,n \qquad (2.34)$$

$$H \ge 0, \; z_{ik} \text{ zero-one.}$$

In (2.33), the value of H is less than or equal to the smallest sum of between cluster distances. If z_{ik} equals to one, then H must be less than or equal to the sum of between cluster distances for object i. Again, if z_{ik} equals to zero, then the corresponding constraint in (2.33) becomes redundant.

2.6 COMPUTATIONAL EXAMPLE

In this section, I apply the above methods and some popular heuristic methods in cluster analysis to a problem, and examine the cluster solutions obtained from the different methods. The data set I use is a distance matrix from Johnson and Wichern [1988, p.546]. This distance matrix describes the distances between twenty-two utilities. I apply six mathematical programming models; (SDM-WS), (MDM-TF-B), (SWD-WS), (MWD-TF-B), (MSW-WS) and (NSB-TF), (the model (SDC-WS) is excluded due to the large problem size), and five heuristic methods namely, between groups average linkage (BGAVG), within group average linkage (WGAVG), single linkage (SINGL), complete linkage (COMPL), and Ward's method (WARD) to the data. I use six forms of $g(f_1, f_2, \ldots, f_m)$ to evaluate these methods. After I obtained the cluster solutions from all the methods, I calculate the six criterion function values for all these cluster solutions and the results are reported in Table 2.1.

Table 2.1: Inefficiency of Cluster Solutions from different Methods

	CRITERIA						
MODELS	SDM-WS	MDM-TF	SWD-WS	MWD-TF	MSW-WS	NSB-TF	AVERAGE
(SDM-WS)	0.000*	0.053*	0.148	0.079*	0.097	0.386	0.127
(MDM-TF-B)	0.015*	0.000*	0.046	0.162	0.038*	0.112*	0.062
(SWD-WS)	0.055	0.061	0.000*	0.162	0.014*	0.089*	0.064
(MWD-TF-B)	0.111	0.095	0.018*	0.000*	0.069	0.139	0.072
(MSW-WS)	0.073	0.167	0.010*	0.162	0.000*	0.148	0.093
(NSB-TF)	0.161	0.350	0.083	0.162	0.065	0.000*	0.137
(BGAVG)	0.051	0.090	0.628	0.144	0.135	0.731	0.297
(WGAVG)	0.051	0.090	0.628	0.144	0.135	0.731	0.297
(SINGL)	0.072	0.090	2.059	0.144	0.175	0.905	0.574
(COMPL)	0.070	0.294	0.075	0.079*	0.091	0.281	0.148
(WARD)	0.008*	0.053*	0.244	0.079*	0.108	0.519	0.169

Note: Asterix represents the best three approaches in each criterion function, and 0.000 represents the best approach.

For each of the criterion functions, it is expected that the best method is the mathematical programming model which gives optimal solution. Note that none of the heuristic methods performs better or as well as any of the mathematical programming approaches. It is interesting to see that for the mathematical programming models, their solutions also perform well when evaluated by the other criterion functions. For example, (MDM-TF-B) is one of the top three methods in four out of the six criterion functions, and each of (SDM-WS) and (SWD-WS) are one of the top three methods in three out of the six criterion functions. Other mathematical programming models also perform well in the other criterion functions. This is reflected by the average values of inefficiency from the eleven models in Table 2.1. The average values of inefficiency of the six mathematical programming models are significantly less than that of the heuristic methods.

2.7 CONCLUSION

The main advantage of the mathematical programming approaches to cluster analysis is the capability of providing an optimal solution with respect to some appropriate objective functions. I provide a systematic way of generating a whole array of meaningful criteria for cluster analysis. In particular, I introduce five new mathematical programming models, including two bicriterion formulations. The ability to generate many meaningful criteria for evaluating cluster solutions increases the power and the flexibility of applying mathematical programming approaches to cluster analysis. The computational results support the use of mathematical programming models in cluster analysis. Additional constraints [Aronson and Klein, 1989] can easily be incorporated into the models.

Appendix 2.1: Formulations of (SM) and (SW)

According to Vinod [1969], the cluster-median problem can be formulated as follows:

Let $x_{ij} = \begin{cases} 1 \text{ if the i-th objects belongs to the j-cluster} \\ 0 \text{ otherwise} \end{cases}$

for i=1,...,n and j=1,...,n.

$$\text{(SM)} \quad \text{MIN} \quad \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij} \tag{2.35}$$

$$\text{S.T.} \quad \sum_{j=1}^{n} x_{ij}=1, \qquad \text{for } i=1,\dots,n \tag{2.36}$$

$$\sum_{j=1}^{n} x_{jj}=m \tag{2.37}$$

$$nx_{jj} \geq \sum_{i=1}^{n} x_{ij}, \qquad \text{for } j=1,\dots,n \tag{2.38}$$

x_{ij} zero-one.

In (SM), the number of clusters defined is n, where n-m clusters are empty. In (2.37), only m of the x_{jj} are equal to 1 (if $x_{jj}=1$, then point j is a median) and the other x_{jj} are equal to zero. The constraints in (2.38) define non-empty clusters.

Rao [1971] discussed an integer programming model, (SW), which has the objective of minimizing the sum of within group pairwise distances. Let

$y^{k}_{ij} = \begin{cases} 1 \quad \text{if both i-th and j-th objects belong to the k-th cluster} \\ 0 \quad \text{otherwise.} \end{cases}$

His formulation is as follows:

$$\text{(SW)} \quad \text{MIN} \quad \sum_{k=1}^{m} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij} y^{k}_{ij} \tag{2.39}$$

$$\text{S.T.} \quad z_{ik} + z_{jk} - y^{k}_{ij} \leq 1, \qquad \text{for } i=1,\dots,n-1;\ j=i+1,\dots,n;\ k=1,\dots,m \tag{2.40}$$

$$\sum_{k=1}^{m} z_{ik} = 1, \qquad \text{for } i=1,\dots,n \tag{2.41}$$

y^{k}_{ij} zero one, z_{ik} zero-one.

Aronson and Klein [1989] modified (SW) by replacing all y^{k}_{ij} , for k=1,...,p in (SW) by y_{ij}. I call their model (SW1).

CHAPTER 3

LINEAR GOAL PROGRAMMING IN DISCRIMINANT ANALYSIS

3.1 INTRODUCTION

In this chapter, two linear goal programming models to discriminant analysis are being introduced. The first model incorporates discriminating information within group in terms of group membership probabilities in a linear goal programming model. The second linear programming model has the ability to capture the non-monotonicity of the attribute scores in discriminant problems. The two models are discussed in section 3.3 and 3.4, respectively. A review of some common techniques in discriminant analysis: Fisher's discriminant function, logistic regression, and linear programming approaches is presented in the next section.

3.2 THE PROBLEM OF SEPARATING GROUPS

Discriminant analysis concerns separating two or more groups of objects in a data set and allocating new objects to previously defined groups. Generally, we start with m random samples from m different populations, of sizes n_1, n_2,..., n_m. Each object is typically described by a vector of attribute values. For a classification problem with two populations π_1 and π_2, classification of objects is based on the measurements on p attributes. The sample data matrices for population 1 and population 2 are $\mathbf{A}_1$ and $\mathbf{A}_2$, respectively.

3.2.1 Fisher's Linear Discriminant Function

For a two-group classification problem, Fisher [1936] attempted to take a linear combination of the p attributes, and choose the coefficients to maximize the ratio of the between group variance to the within group variance. He developed the following sample linear discriminant function:

$$(\bar{a}_1-\bar{a}_2)' S^{-1} \mathbf{a}, \tag{3.1}$$

where $\bar{\mathbf{a}}_1$ and $\bar{\mathbf{a}}_2$ are the sample mean vectors of $\mathbf{A}_1$ and $\mathbf{A}_2$, respectively, S is the pooled sample covariance matrix, and **a** is the vector scores of an object. The classification rule based on the samples becomes the following:

Classify **a** to π_1 if $\quad (\bar{a}_1-\bar{a}_2)' S^{-1} \mathbf{a} \geq 1/2(\bar{a}_1-\bar{a}_2)' S^{-1} (\bar{a}_1-\bar{a}_2)$

Classify **a** to π_2 if $\quad (\bar{a}_1-\bar{a}_2)' S^{-1} \mathbf{a} < 1/2(\bar{a}_1-\bar{a}_2)' S^{-1} (\bar{a}_1-\bar{a}_2)$.

Fisher's linear discriminant function will minimize the probability of misclassification if the following conditions are met:

1. The population distributions are multivariate normal.
2. The covariance matrices of π_1 and π_2 are the same.
3. The means and the common covariance matrix are known.

This has been a very popular approach used in discriminant analysis.

3.2.2 Logistic Regression

The logistic regression follows from the Bayes's Theorem:

$$P(\pi_i|\mathbf{a}) = \frac{P(\mathbf{a}|\pi_i)\ P(\pi_i)}{\sum_{i=1}^{2} P\ (\mathbf{a}|\pi_i)\ P(\pi_i)}, \tag{3.2}$$

where i=1,2 in a two groups discriminant problem. If $P(\mathbf{a}|\pi_i)$ and $P(\pi_i)$ are known, then $P(\pi_i|\mathbf{a})$ can be computed. However, $P(\mathbf{a}|\pi_i)$ and $P(\pi_i)$ generally need to be estimated.

If one assumes the conditional probability density function takes the following form [Day and Kerridge, 1967]:

$$P(a|\pi_i) = r_i.\exp\{-1/2(a-\mu_i)'\Sigma^{-1}(a-\mu_i)\}\theta(a)da, \qquad (3.3)$$

where $\theta(a)$ is a non-negative scalar function of **a** and $\theta(a)$ is integrable, and $r_i>0$ is a normalizing constant, then

$$P(\pi_1|a) = \frac{\exp(\beta_o + \beta'a)}{1 + \exp(\beta_o + \beta'a)}, \qquad (3.4)$$

$$P(\pi_2|a) = \frac{1}{1 + \exp(\beta_o + \beta'a)}, \qquad (3.5)$$

where $\beta_o = \ln(\frac{P(\pi_1)a_1}{P(\pi_2)a_2}) - 1/2(\mu_1'\Sigma^{-1}\mu_1 - \mu_2'\Sigma^{-1}\mu_2), \qquad (3.6)$

$$\beta' = (\mu_1 - \mu_2)'\Sigma^{-1}. \qquad (3.7)$$

There are two approaches to estimate β_o and β'. One is the weighted least-square approach and the other is the maximum likelihood approach [Flath and Leonard, 1979].

3.2.3 Mathematical Programming and Discriminant Analysis

Since Freed and Glover [1981a, 1981b] and Hand [1981] proposed the use of linear programming approaches to solve the discriminant problems, many authors [Bajgier and Hill, 1982; Choo and Wedley, 1985; Freed and Glover, 1986; Stam and Joachimsthaler, 1989; Lee and Ord, 1990] developed different variants of the linear programming models. Furthermore, some authors [Bajgier and Hill, 1982; Choo and Wedley, 1985; Freed and Grover, 1986; Joachimsthaler and Stam, 1988; Stam and Joachimsthaler, 1990; Stam and Jones, 1990; Lee and Ord, 1990] attempted to compare and evaluate the classification performance of the mathematical programming approaches and statistical approaches via empirical or simulated experiments.

The linear programming model introduced by Freed and Glover [1981a] has the objective to minimize the maximum deviation (MMD) for the objects that are misclassified by the classification function. Let **X** be a nxp matrix where x_{ik} denotes the k-th attribute value of the i-th object for i=1,2,...,n, and k=1,2,...,p, where n is the number of objects in the sample, w_k be the estimated weights in the classification function, where k=1,...,p, and G_1 and G_2 be the two sample sets drawn from π_1 and π_2, respectively. The (MMD) model is formulated as follows:

$$\text{(MMD)} \quad \text{MIN} \quad d \tag{3.8}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} w_k x_{ik} + d \geq c, \qquad \text{for } i \in G_1 \tag{3.9}$$

$$\sum_{k=1}^{p} w_k x_{ik} - d \leq c, \qquad \text{for } i \in G_2 \tag{3.10}$$

$$w_k,\ c \text{ unrestricted in sign, } d \geq 0.$$

In (MMD), a normalization constraint is needed to avoid the trivial solution (i.e., the values of all w_k and c equal to zero). Freed and Glover [1986] suggested the use of the following normalization constraint:

$$\sum_{k=1}^{p} w_k + c = r, \tag{3.11}$$

where r is a non-zero number.

Instead of minimizing the maximum deviation, Freed and Glover [1981b] proposed another linear programming model to minimize the sum of individual external deviations (MSD). Let $n = n_1 + n_2$, then

$$\text{(MSD)} \quad \text{MIN} \quad \sum_{i=1}^{n} d_i \tag{3.12}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} w_k x_{ik} + d_i \geq c, \qquad \text{for } i \in G_1 \tag{3.13}$$

$$\sum_{k=1}^{p} w_k x_{ik} - d_i \leq c, \qquad \text{for } i \in G_2 \tag{3.14}$$

$$w_k,\ c \text{ unrestricted in sign, } d_i \geq 0.$$

The objective of (MSD) is to minimize the total group overlap instead of the maximum overlap as in (MMD). A normalization constraint is needed in (MSD) to avoid the trivial solution.

The objective of minimizing the sum of interior distances [Freed and Glover, 1986] can be formulated as the following linear programming problem (MSID):

$$\text{(MSID)} \quad \text{MIN} \quad q_1 d - q_2 \sum_{i=1}^{n} \alpha_i \tag{3.15}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} w_k x_{ik} + d - \alpha_i \geq c, \qquad \text{for } i \in G_1 \tag{3.16}$$

$$\sum_{k=1}^{p} w_k x_{ik} - d + \alpha_i \leq c, \qquad \text{for } i \in G_2 \tag{3.17}$$

$$w_k,\ c \text{ unrestricted in sign, } d \geq 0,\ \alpha_i \geq 0.$$

(MSID) is a bicriterion linear programming model where the two objectives are minimizing the maximum misclassification and maximizing the sum of deviations of the correctly classified objects from the cutoff boundary. The values of q_1 and q_2 reflect the relative importance of the two objectives in (MSID) and must be provided by the decision maker.

Choo and Wedley [1985] discussed the use of an integer programming method to minimize the number of misclassified objects in a classification problem. Their integer programming formulation (MNM) can be stated as follows:

$$\text{(MNM)} \quad \text{MIN} \quad \sum_{i=1}^{n} e_i \tag{3.18}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} w_k x_{ik} + a + Me_i \geq 1, \qquad \text{for } i \in G_1 \tag{3.19}$$

$$\sum_{k=1}^{p} w_k x_{ik} + a - Me_i \leq -1, \qquad \text{for } i \in G_2 \tag{3.20}$$

$$a,\ w_k \text{ unrestricted in sign, } e_i \text{ zero-one.}$$

M is a large positive number, and e_i are zero-one variables for all i. The objective of (MNM) is to minimize the actual number of misclassifications.

Koehler and Erengue [1990], Stam and Joachimsthaler [1990] also studied the problem of finding a linear discriminant function to minimize the number of misclassifications. Their model (MNM2) is as follows:

$$\text{(MNM2)} \quad \text{MIN} \quad \sum_{i=1}^{n} e_i \tag{3.21}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} w_k x_{ik} + Me_i \geq c, \quad \text{for } i \in G_1 \tag{3.22}$$

$$\sum_{k=1}^{p} w_k x_{ik} - Me_i \leq c, \quad \text{for } i \in G_2 \tag{3.23}$$

$$w_k, \ c \text{ unrestricted in sign, } e_i \text{ zero-one.}$$

A normalization constraint is needed to avoid the trivial solution in (MNM2).

Stam and Joachimsthaler [1989] discussed the use of ℓ_p norm in discriminant analysis. Their model can be expressed as the following mathematical programming problem:

$$\text{(LPN)} \quad \text{MIN} \quad [\sum_{i=1}^{n} (d_i)^r \ ]^{(1/r)} \tag{3.24}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} w_k x_{ik} + d_i - \alpha_i \geq c, \quad \text{for } i \in G_1 \tag{3.25}$$

$$\sum_{k=1}^{p} w_k x_{ik} - d_i + \alpha_i \leq c, \quad \text{for } i \in G_2 \tag{3.26}$$

$$w_k \text{ unrestricted in sign, } d_i \geq 0, \ \alpha_i \geq 0, \ r \text{ is a positive integer.}$$

The cutoff value c is chosen arbitrary. They pointed out that (MMD) and (MSD) are the two special cases of the ℓ_p norm discriminant analysis. The (MMD) and (MSD) formulations are the same as (LPN) formulation when $r=\infty$ and $r=1$, respectively.

Another variant of the linear programming models was proposed by Lee and Ord [1990]. They formulated the classification problem similar to a regression problem where the objective function is to minimize the sum of absolute deviations

(LAD). The formulation is as follows:

$$\text{(LAD)} \quad \text{MIN} \quad \sum_{i=1}^{n} (d_i + \alpha_i) \tag{3.27}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} w_k x_{ik} + a + d_i - \alpha_i = 1, \quad \text{for } i \in G_1 \tag{3.28}$$

$$\sum_{k=1}^{p} w_k x_{ik} + a + d_i - \alpha_i = 0, \quad \text{for } i \in G_2 \tag{3.29}$$

$$a,\ w_k \text{ unrestricted in sign, } d_i \geq 0,\ \alpha_i \geq 0.$$

There have been some attempts [Bajgier and Hill, 1982; Freed and Grover, 1986; Joachimsthaler and Stam, 1988; Lee and Ord, 1990] to compare and evaluate the classification performance of the mathematical programming approaches and the classical discriminant techniques using simulated experiments. Some of the results show that linear programming approaches are competitive with the classical discriminant techniques in terms of the classification performance (number of objects correctly classified).

Bajgier and Hill [1982] found that (MSD) and (MMD) outperformed Fisher's linear discriminant function (FLDF) under certain conditions of the data sets in a simulated study. The (MSD) performed better than (FLDF) when the sample sizes of the two groups were unequal. The (MMD) outperformed the other approaches under the condition that group overlap was small, but not performing well when the two groups were close to each other. Freed and Glover [1986, pp.161] reported their finding as "... Tests results showed that among the LP variants tested, the (MSD) formulation generally is the most reliable predictor of group membership". In their simulated experiment, Joachimsthaler and Stam [1988] found that when outliers were present in the data set, (MSD) and logistic discriminant function performed better than (FLDF). Koehler and Erengue [1990] reported that (MNM2) performed better than (FLDF) as the variance heterogeneity of the two populations increased. The above results show that linear programming approaches (especially the MSD) are competitive alternatives to the classical discriminant techniques.

3.3 LINEAR GOAL PROGRAMMING IN ESTIMATION OF CLASSIFICATION PROBABILITIES

The linear programming approaches to classification problems have yielded satisfactory results [Choo and Wedley, 1985; Freed and Glover, 1986; Glover, Keene and Duea, 1988; Joachimsthaler and Stam 1988]. Unlike logistic regression [Cox, 1970; Anderson, 1972], which can incorporate membership probabilities of an object belonging to a group, linear programming models [Choo and Wedley, 1985; Freed and Glover, 1986; Glover, Keene and Duea, 1988; Lee and Ord, 1990] and (FLDF), when applied to estimate the attribute weights of the classification function, do not incorporate discriminating information between members within the same group.

Group membership probabilities, when available, can be and should be used to discriminate between members of the same group. When working with categorical dependent variables, most methods round off the group membership probabilities, and information useful for discriminating between members within the groups is lost. For example, dividing companies into bankrupt and non-bankrupt groups ignores the fact that the non-bankrupt group contains companies of varying qualities. This is a severe disadvantage of (FLDF) and linear programming models where group membership is coded as a dichotomous or categorical variable. Recently, constrained multiple regression models with general ℓ_p-norm are used to estimate membership probabilities [Stam and Ragsdale, 1990].

I provide a continuous goal programming model (GP1) which works directly with the conditional probabilities of group membership rather than the simple group category to which a member belongs. In applying this continuous goal programming model to classification problems, the users have to provide the estimated group membership probabilities of the objects within the sample. With the use of conditional probabilities, the strength of group membership is measured and further discrimination within groups is possible. Actual experience of an MBA admission committee is used to illustrate the implementation of (GP1).

The classification power of (GP1) is compared with four other methods of classification including logistic regression (LG), (MNM) [Choo and Wedley, 1985], multiple regression with minimum sum of absolute deviations (MSAD) [Wagner, 1959], and Fisher's linear discriminant function (FLDF). Moreover, simulated data are used to test the effectiveness of the above approaches in minimizing misclassification errors.

3.3.1 Model Formulation

For the two groups discriminant problem with n objects ($n=n_1+n_2$), let p_i, $i=1,\ldots,n$ be the probability that the i-th object belongs to one of the parent populations. The rows of **X** are arranged in the descending order of the p_i values. Furthermore, let $w_1, w_2, \ldots, w_p$ be the attribute weights to be determined and the weighted sum function $w_1x_{i1}+w_2x_{i2}+\ldots+w_px_{ip}$ is used to distinguish the likelihood of belonging to the population. In the perfect situation, the weighted sums $w_1x_{i1}+w_2x_{i2}+\ldots+w_px_{ip}$ will have the same ordering as the p_i values and thus we would expect the following :

$$\sum_{k=1}^{p} w_k x_{uk} \geq \sum_{k=1}^{p} w_k x_{rk} , \qquad \forall 1 \leq u < r \leq n. \tag{3.30}$$

There are n(n-1)/2 inequalities in (3.30). All these inequalities are used in the ordinal regression [Srinivasan, 1975]. However, when $|p_u-p_r|$ is small, it is not wise to enforce the corresponding ordering in (3.30) due to possible data inaccuracy in p_u and p_r. Thus, unlike the ordinal regression, many pairwise comparisons (u,r) may be excluded from (3.30).

There are many reasonable methods to select the subset of pairwise comparisons to be enforced in (3.30). I suggest one approach as follows:

For any small $\alpha>0$, define D_α by $D_\alpha = \{ (u,r) : 1\le u<r\le n, p_u-p_{r-1}\le\alpha \; \& \; p_u-p_r>\alpha \}$. Essentially, D_α consists of consecutive pairs (u, r) with p_u-p_r slightly greater than α. Let $r_\alpha = \max \{r : (u,r)\in D_\alpha \text{ for } 1\le u\le n\}$. Then none of the pairs in D_α includes any r-th object with $r>r_\alpha$. To avoid this omission, define $U_\alpha = \{(u,r) : r_\alpha+1\le r\le n, \; p_{u+1}-p_r\le\alpha \; \& \; p_u-p_r>\alpha\}$ when $r_\alpha<n$ and let $\Omega_\alpha = D_\alpha \cup U_\alpha$.

The n(n-1)/2 inequalities in (3.30) are replaced by

$$\sum_{k=1}^{p} w_k x_{uk} \ge \sum_{k=1}^{p} w_k x_{rk} , \qquad \forall \; (u,r)\in\Omega_\alpha. \tag{3.31}$$

Since perfect ordering is not always possible, deviational variables d_{ur} are introduced into (3.31) to allow for more flexibility:

$$\sum_{k=1}^{p} w_k x_{uk} + d_{ur} \ge \sum_{k=1}^{p} w_k x_{rk} , \qquad \forall \; (u,r)\in\Omega_\alpha. \tag{3.32}$$

The objective is to minimize the total sum of deviations $\sum_{(u,r)\in\Omega_\alpha} d_{ur}$. Furthermore, it is natural that the difference $\sum_{k=1}^{p} w_k x_{uk} - \sum_{k=1}^{p} w_k x_{rk}$ should be larger for larger values of p_u-p_r. The model formulation for preserving the order of membership probabilities for all the pairs in Ω_α is given below:

$$\text{(GP1)} \quad \text{MIN} \quad \sum_{(u,r)\in\Omega_\alpha} d_{ur} \tag{3.33}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} w_k x_{uk} - \sum_{k=1}^{p} w_k x_{rk} + d_{ur} \ge p_u-p_r, \quad \forall \; (u,r)\in\Omega_\alpha \tag{3.34}$$

$$w_k \text{ unrestricted in sign, } d_{ur}\ge 0, \; \forall (u,r)\in\Omega_\alpha.$$

The attribute weights $w_1, w_2, \ldots, w_p$ obtained from the solution of (GP1) are used to compute the weighted sums $w_1x_{i1} + w_2x_{i2} + \ldots + w_px_{ip}$, i=1,2,...,n, of all the objects. The log odds ratio, $\ln(p_i/(1-p_i))$, i=1,2,...,n, of all the objects, is regressed on the weighted sums to obtain an intercept coefficient b_o and a slope coefficient b_1. For any object i, its membership probability p_i is estimated by

$$\hat{p}_i = \frac{1}{1+\exp[-b_o-b_1(w_1x_{i1}+w_2x_{i2}+\ldots+w_px_{ip})]} . \qquad (3.35)$$

3.3.2 Additional Features

Sometimes, prior information may be available in a classification problem. For example, the members in an MBA admission committee may agree that the higher the GMAT score of an applicant, the higher the probability that this applicant will be accepted. Consequently, the attribute weight for the GMAT score should be positive. Furthermore, the admission committee may also agree that GMAT score is the most important factor in evaluating applicants. Therefore, GMAT score should be weighted more heavily than the other attributes in the classification function. In practice, additional constraints reflecting useful prior information can be added to (GP1) to obtain better and more meaningful attribute weights. These include the positive or negative sign constraints on the weights of certain attributes. Let $\bar{x}_k$ be the average values of the k-th attribute in the development sample, then higher weights can be imposed on more important attributes by adding the following constraint,

$$\bar{x}_u w_u - T(\bar{x}_v w_v) \geq 0 \qquad (3.36)$$

where T can be any desired value which reflects the relative importance of the u-th attribute and the v-th attribute. Extreme values of attribute weights can be avoided by bounding the ratios of the attribute weights. For example, adding the following constraints to (GP1) can avoid extreme values of attribute weights,

$$\bar{x}_k w_k \leq U \qquad (3.37)$$

$$\bar{x}_k w_k \geq L \qquad (3.38)$$

with appropriate values of U and L.

3.3.3 Empirical Evidence

The decision making situation chosen for testing the model (GP1) against four other different methods (LG, MNM, MSAD, and FLDF) is the admission of students to an Executive M.B.A. Program. A committee of four professors was given the task of reviewing all applicant files and making decisions of admission or rejection.

Prior to reviewing the applicant files, the committee convened to discuss evaluation procedures. Although there was much information in the applicant files, the committee agreed that four main attributes were of importance. These were:

(1) managerial experience,

(2) undergraduate preparation,

(3) letters of reference, and

(4) scores on the Graduate Management Admission Test (GMAT).

No prior weightings were established for these attributes. Nevertheless, an evaluation form was prepared which required each committee member to rate each applicant on each attribute. The ratings were semantic indicators as to whether the applicant was poor, fair, good or excellent on the dimensions being considered. As individuals, each committee member read the files and made the ratings. Later, when the members met as a group, these ratings were useful for establishing a consensus. Of 86 applications for the 1987 class, 37 were admitted.

The relevant independent variables used as attributes are presented in Table 3.1. The empirical study is based on 68 cases with complete information. There are 41 cases with probability of admission greater than or equal to 0.50. A list of the 68 cases is given in Appendix 3.1.

The dependent variable (probability of admission) was not actually used in the decision process of the committee. Instead, it was determined afterwards. Each committee member used the Analytic Hierarchy Process, AHP [Saaty, 1977] to determine weights for the four main attributes and the four semantic indicators used to rate each attribute. For example, each member had to think of prototype candidates with poor, fair, good and excellent records on managerial experience. Then with the managerial experience attribute in mind, they undertook AHP paired comparisons between prototype candidates to establish importance weights for the semantic indicators. In a like manner, priority weight z_{1k} was established for each indicator b with respect to each attribute k. Finally, AHP comparisons were carried out to determine weight w_k for each attribute.

Table 3.1: List of Attributes and Labels

Name	Description	Type	Label
JOBL	Job level	C	1=Non-business, 2=Consultant, 3=Low, 4=Middle, 5=Top
YMGT	Years in management	R	
JOBS	Job mobility	I	Number of job title switches in last 5 years
DEG	Highest education	C	1=Non-university, 2=Some university, 3=technical college, 4=CPA/CGA, 5=CA/non-science graduate/non-business graduate, 6=B.Sc., 7=B.Bus., 8=Masters degree, 9=Ph.D.
YEDD	Years out of school	I	Number of years since formal education
LETT	Employer's letter	C	0=No. 1 to 6 indicating strength of reference.
QGMAT	Quantitative score	R	Quantitative GMAT score
TGMAT	Total GMAT score	R	
LETA	Average reference	C	Average strength of references.
PROB	Probability	R	Acceptance probability.

C = Categorical variable, R = Ratio variable, I = Integer variable.

Using the indicator weights as absolute measures, each candidate i has a score $s_i = \sum w_k(\delta_{1ki}z_{1k}+\delta_{2ki}z_{2k}+\delta_{3ki}z_{3k}+\delta_{4ki}z_{4k})$, where $\delta_{bki}=1$ means the candidate i was rated with indicator b with respect to attribute k, and $\delta_{bki}=0$ otherwise. This score s_i can be used to measure the desirability of candidate i. Saaty calls this use of AHP as measurement with absolute values. These scores s_i are converted into probabilities of acceptance by rescaling them between 1, the probability for an imaginary candidate who scores excellent on all indicators, and 0, the probability for another imaginary candidate who scores poor on all dimensions.

Although it is possible to calculate the membership probabilities perceived by each committee member, I have used the average group ratings, importance weights, and indicator values to emulate the group discussion, and consensus process which actually occurred in the committee. Thus, I am using aggregated group scores to generate the committee's probability of acceptance for each candidate. The average attribute weights and indicator values which were used in the study are given in Table 3.2. The resulting acceptance probabilities are in the last column of Appendix 3.1.

Table 3.2: Group Attributes and Indicator Weights

Attribute	Attribute	Group Indicator Weights			
	Weight	Poor	Fair	Good	Excellent
Managerial Experience	.305	.076	.148	.283	.493
Academic Preparation	.280	.056	.138	.323	.483
Letters of Reference	.095	.081	.158	.313	.448
G. M. A. T.	.320	.054	.137	.282	.527

Highest Potential Score = .305(.493)+.280(.483)+.095(.448)+.320(.527)
= .497, which is given a probability of 1.
Lowest Potential Score = .305(.076)+.280(.056)+.095(.081)+.320(.054)
= .064, which is given a probability of 0.

The data of 68 cases are randomly divided into development and validation samples with 34 cases in each sample. For all the methods used, attribute weights are derived from the development sample. Then the remaining holdout sample is used to measure the success of each method. Success is defined herein as the ability of each method to correctly identify the candidates with greater than or equal to 0.50 probability of admission. A listing of the SPSS-X [1988] programme used is given in Appendix 3.2.

All the attribute weights obtained by the various methods are given in Table 3.3. The classification results are given in Table 3.4. (GP1) and (MNM) both have six misclassifications in the validation sample. Both (LG) and (FLDF) have eight misclassifications and (MSAD) has nine misclassifications in the validation sample. (GP1) and (MNM) have the smallest number of misclassifications in the validation sample.

Table 3.3: Attribute Weights of the Different Methods

	ATTRIBUTE WEIGHTS									
	JOBL	YMGT	JOBS	DEG	YEDD	LETT	QGMAT	TGMAT	LETA	CONSTANT
METHOD										
(GP1)	-.357	.116	.130	-.016	.023	-.135	.086	.005	.081	0.0
(RGP1)	-.198	.057	.065	-.042	.009	0	.033	.0032	.082	0.0
(LG)	-.022	.024	.123	.122	.026	.016	.031	.006	-.04	-5.9072
(MNM)	.510	.147	1.122	-.767	.152	-.798	.355	.058	.72	-51.19
(RMNM)	6.793	-1.19	-.147	-4.23	.669	0	0	.202	8.09	-155.20
(MSAD)	-.017	.003	.017	.030	.007	-.021	.003	.0018	.006	-0.8254
(RMSAD)	-.007	.003	.009	.028	.0065	0	.0015	.0018	0	-0.8617
(FLDF)	-.395	-.030	.167	.151	.102	-.314	-.060	.050	.467	-28.888

Note that the attribute weights of DEG and LETT obtained from (GP1) and (MNM) are both negative which seem to contradict apparent intuition. But further analysis shows that applicants with higher degrees may not always be preferred to those with a Bachelor degree in Business (score 7). Applicants with either a Master degree (score 8) or a Ph.D. (score 9) have usually majored in areas other then Business and are not working at high management levels. Furthermore, applicants who are either CPA/CGA (score 4) or CA (score 5) may have a higher chance of being accepted than those with Bachelor degrees in Sciences. Thus, the sign of degree is really indeterminate. However, for LETT it is difficult to interpret why the weight is negative.

Table 3.4: Classification Results of the Different Methods

METHOD	NUMBER OF CASES	"YES" AND PREDICTED "YES"	"NO" AND PREDICTED "NO"	"YES" AND PREDICTED "NO"	"NO" AND PREDICTED "YES"	CORRECT PREDICTION PERCENTAGE
(GP1)	D =34	6	23	2	3	85.3
	V =34	9	19	6	0	82.4
(RGP1)	D =34	5	23	3	3	82.4
	V =34	9	19	6	0	82.4
(LG)	D =34	6	22	2	4	82.4
	V =34	9	17	6	2	76.5
(MNM)	D =34	8	26	0	0	100.00
	V =34	9	19	6	0	82.4
(RMNM)	D =34	8	26	0	0	100.00
	V =34	9	19	6	0	82.4
(MSAD)	D =34	8	23	0	3	91.3
	V =34	8	17	7	2	73.6
(RMSAD)	D =34	8	22	0	4	88.1
	V =34	10	16	5	3	76.5
(FLDF)	D =34	8	24	0	2	94.1
	V =34	10	16	5	3	76.5

D : development sample
V : validation sample

The advantage of the mathematical programming approach is that positive or negative sign constraints can be easily imposed if desired (for example, see Srinivasan, Jain, and Malhotra, 1983). For further analysis, the MBA problem is solved again using (GP1), (MNM), and (MSAD) with positive sign constraints imposed on LETT, QGMAT, TGMAT, and LETA. The "restricted" models (with positive sign constraints) are (RGP1), (RMNM), and (RMSAD), respectively. The classification results of the three "restricted" models are also listed in Table 3.3 and Table 3.4. The validation samples (GP1) and (MNM) have the same correct prediction percentage with or without the sign constraints, while the correct prediction percentage of (MSAD) is improved by 2.9% if the sign constraints are imposed.

3.3.4 Simulation Experiment

Two simulation experiments are conducted to investigate the performance of the different classification techniques in discriminating between groups. For both experiments, the samples are drawn from two multivariate normal populations with three discriminating attributes. Population 1 is distributed as $N(0,I)$ and population 2 is distributed as $N(\nu,\Lambda)$, where ν is a mean vector and Λ is a diagonal matrix. Similar to Bajgier and Hill [1982], the three means in ν are chosen to be equal as are all the diagonal elements in Λ. Three different values, 0.5, 1,and 3 are chosen to represent three different mean vectors, and 1, 4, and 16 are chosen to represent the diagonal elements of three different diagonal matrices. This yields a 3x3 factorial design with nine combinations in each of the two simulation experiments. The relative frequency of population 1 and population 2 are set to be equal.

In the first simulation experiment, the linear sum of the three attribute values of each object is perturbed and then substituted into a logistic equation to compute the probability of group membership. The probability of group membership is obtained from the logistic equation. As a result, the experimental design in the first simulation experiment is in favor of logistic regression. The second simulation experiment tries to eliminate this bias.

In the second simulation experiment the linear sum of the three attribute values of each object is systematically transformed using the following criteria: the highest 50% of the linear sums are transformed into squares of the linear weighted sums, and the lowest 50% of the linear sums are transformed into square roots of the linear sums. After these transformations, the linear sum is perturbed and then substituted into a logistic equation to compute the probability of group membership. In both simulation studies the linear sums are perturbed by adding a random error which is normally distributed with zero mean and variance δ_e^2. Its variance is computed from the following formula [Srinivasan, 1975]:

$$E = \frac{\delta_e^2}{\delta_e^2 + \delta_s^2} = 0.3 \qquad (3.39)$$

where δ_s^2 is the variance of the linear sums in the sample before the error term is introduced. The total sample size is 1030, and consists of 30 cases randomly drawn from the sample as the development sample, and the 1000 cases used as the validation sample. Ten problems are generated for each combination of the ν and Λ values. All five approaches (GP1, LG, MNM, MSAD, and FLDF) are applied to solve the problems. For (MNM), only the LP relaxation is solved. The LP relaxation of (MNM) is similar to (MSD) except for a small difference in the constant terms.

Table 3.5: Average Hits of Different Methods (First Simulation Experiment)

	METHOD	ν=0.5	ν=1	ν=3
Λ=1	(GP1)	970.6	967.0	923.2
	(LG)	974.1	970.0	923.1
	(MNM)	928.2	932.4	890.2
	(MSAD)	956.8	959.9	920.8
	(FLDF)	930.6	926.9	873.1
Λ=4	(GP1)	953.1	946.3	905.7
	(LG)	955.4	949.6	910.5
	(MNM)	917.8	906.3	872.4
	(MSAD)	951.4	946.2	906.3
	(FLDF)	916.9	883.2	872.2
Λ=16	(GP1)	879.1	885.7	842.6
	(LG)	886.0	887.1	846.5
	(MNM)	866.6	861.0	830.3
	(MSAD)	880.8	883.4	838.2
	(FLDF)	842.6	844.2	778.6

The average hit rates (out of 1000) of the validation sample of each combination in the first simulation experiment and in the second simulation experiment are reported in Table 3.5 and Table 3.6, respectively. A summary of the total average hit rates of the nine combinations in each simulation experiment are reported in Table 3.7.

Table 3.6. Average Hits of Different Methods (Second Simulation Experiment)

	METHOD	ν=0.5	ν=1	ν=3
Λ=1	(GP1)	877.2	835.4	632.4
	(LG)	874.7	835.5	628.2
	(MNM)	847.1	803.5	632.7
	(MSAD)	869.3	834.5	639.7
	(FLDF)	860.6	818.7	627.3
Λ=4	(GP1)	694.8	648.0	585.4
	(LG)	690.0	646.5	568.6
	(MNM)	675.9	635.4	573.5
	(MSAD)	675.5	642.0	569.5
	(FLDF)	671.9	633.9	569.4
Λ=16	(GP1)	540.4	540.3	519.6
	(LG)	539.6	534.4	518.2
	(MNN)	544.7	533.9	514.2
	(MSAD)	539.1	530.9	516.6
	(FLDF)	546.9	529.3	516.7

Table 3.7: Total Average Hits in the Two Simulation Experiment

Method	First Simulation Experiment	Second Simulation Experiment
(GP1)	919.25	652.61
(LG)	922.48	648.40
(MNM)	889.47	640.18
(MSAD)	915.98	646.34
(FLDF)	874.26	641.63

In the first simulation experiment, logistic regression has the best performance in term of the average hit rates. Since the original membership probability is computed from a logistic equation, this result is expected. (GP1) has the second highest average hit rate, and (MSAD) has the third highest average hit rate.

In the second simulation experiment, (GP1) has the best performance in terms of the total average hit rates. (LG) has the second highest average hit rate and (MSAD) has the third highest average hit rate. It should be pointed out that the systematic non-linear transformation of the linear sum of the attribute values in the second simulation experiment does not seem to favor any of the five approaches. In summary, (GP1) has performed well in this simulation experiment.

3.4 A LINEAR GOAL PROGRAMMING MODEL FOR CLASSIFICATION WITH NON-MONOTONE ATTRIBUTES

Statistical approaches and linear programming approaches to classification problems presume monotonicity of the attribute scores with respect to the likelihood of belonging to one specific group. This may not be realistic in many applications. In view of this, I propose a general linear programming approach with the ability to capture the non-monotonicity of some attribute scores in classification problems.

3.4.1 The Problem of Non-monotonic Attributes

Objects are classified based on the overall scores computed from the derived classification function. However, according to the classification function, the higher the attribute score of an object, all other factors being equal, the higher (if this attribute has a positive weight in the classification function) or the lower (if this attribute has a negative weight in the classification function) will be its overall score. This implied monotonicity is not reasonable in many situations. For example, if the age and blood pressure of an individual are being used to determine whether to assign an individual to a "beginners" fitness class or to an "advanced" fitness class, then an individual who is either too old or too young and an individual who has either a high blood pressure or a low blood pressure may not be suitable for the "advanced" fitness class. As a result, neither positive weights nor negative weights are suitable for both the age and the blood pressure attributes in the classification function. Similar examples can be found in some medical diagnoses when high attribute scores or low attribute scores may indicate symptoms of certain diseases. One possible approach to overcome this difficulty is to transform the scores of an monotonic attribute as deviations from the "desirable value" for the class, however, sometimes desirable value may not be easy to determine. Moreover, there may exist a range of "desirable values".

In the next section, a linear goal programming model (GP2) which can overcome the difficulty of imposing an implied monotonicity in classification function analysis is developed. Furthermore, a simulation experiment is conducted to examine the effectiveness of this linear goal programming approach to classification problems. The results of the proposed approach are very encouraging.

3.4.2 Model Formulation

As noted earlier, Fisher's linear discriminant function, logistic regression, and linear programming approaches do not handle cases with non-monotonic attribute scores (attribute scores which are not monotonic with respect to the likelihood of belonging to a specific population). In order to overcome this problem, the following approach is suggested. We first consider the case when all the attribute scores are non-monotonic in a two groups discriminant problem. For non-monotonic attributes, their scores are discretized into at least two different levels. Let g_k be the number of levels for the k-th attribute, and

$$\delta^{i}_{k\ell} = \begin{cases} 1\ , & \text{if the level of the k-th attribute of the i-th object is } \ell \\ 0\ , & \text{otherwise} \end{cases}$$

where k=1,...,p, ℓ=1,...,g_k, and i=1,...,n. Let $w_{k\ell}$ be the weight of the ℓ-th level of the k-th attribute in the classification function. The overall score of any object i is equal to $\sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i}$. Equivalently, we can replace each attribute k by g_k dummy variables $\delta_{k\ell}$, ℓ=1,...,g_k in the matrix X. Let c be the cut off value of the overall score between G_1 and G_2. The goal is to determine a set of weights, $w_{k\ell}$ for k=1,...,p, ℓ=1,...,g_k, which satisfies the following conditions:

$$\sum_{k=1}^{p} \sum_{p=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} \geq c, \qquad \text{for } i\in G_1 \qquad (3.40)$$

$$\sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} \leq c, \qquad \text{for } i\in G_2 \qquad (3.41)$$

Since perfect classification results may not always be possible, deviational variables d_i and α_i can be introduced to allow for more flexibility as in standard goal programming models. Hence, (3.40) and (3.41) are replaced by:

$$\sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} + d_i - \alpha_i \geq c, \qquad \text{for } i\in G_1 \qquad (3.42)$$

$$\sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} - d_i + \alpha_i \leq c, \qquad \text{for } i\in G_2 \qquad (3.43)$$

The objective is to minimize the sum of all the d_i values and maximize the sum of all the α_i values. Under perfect classification condition the sum of all the d_i values is equal to zero. Furthermore, if object i is classified correctly, then maximizing α_i will tend to force its overall score as far apart from c as possible [Glover, Keene and Duea, 1988]. Intuitively, this should enhance the classification power. The new goal programming model (GP2) is stated as follows:

$$\text{(GP2)}\quad \text{MIN} \quad P_1\sum_{i=1}^{n} d_i - P_2\sum_{i=1}^{n} \alpha_i \tag{3.44}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} + d_i - \alpha_i \geq c, \qquad \text{for } i\in G_1 \tag{3.45}$$

$$\sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} - d_i + \alpha_i \leq c, \qquad \text{for } i\in G_2 \tag{3.46}$$

$$\sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell} = 1 \tag{3.47}$$

$$w_{k\ell},\ c \text{ unrestricted in sign, } d_i \geq 0,\ \alpha_i \geq 0.$$

Since the primary concern is to make correct classifications, the parameter P_1 should be preemptive over P_2. Equation (3.47) is used to avoid the trivial solution of zero values for all $w_{k\ell}$.

In a classification problem, usually not all attributes are non-monotonic. For monotonic attributes, no discretization is required and only one weight is used for each of these attributes in (GP2). Let I_N be the subset of attributes with non-monotonic scores and I_M be the subset of attributes with monotonic scores. In general, (GP2) can be stated as follows:

$$\text{(GP2)}\quad \text{MIN} \quad P_1\sum_{i=1}^{n} d_i - P_2\sum_{i=1}^{n} \alpha_i \tag{3.48}$$

$$\text{S.T.} \quad \sum_{k\in I_N} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} + \sum_{k\in I_M} w_k x_{ik} + d_i - \alpha_i \geq c, \quad i\in G_1 \tag{3.49}$$

$$\sum_{k\in I_N} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} + \sum_{k\in I_M} w_k x_{ik} - d_i + \alpha_i \leq c, \quad i\in G_2 \tag{3.50}$$

$$\sum_{k\in I_N} \sum_{\ell=1}^{g_k} w_{k\ell} + \sum_{k\in I_M} w_k = 1 \tag{3.51}$$

$$w_{k\ell},\ w_k,\ c \text{ unrestricted in sign, } d_i \geq 0 \text{ and } \alpha_i \geq 0.$$

While $w_{k\ell}$ is the attribute weight of the ℓ-th level of the k-th attribute for $k\in I_N$, and w_k is the weight of the k-th attribute for $k\in I_M$.

The $w_{k\ell}$ obtained from solving (GP2) can be used to compute the overall scores, S_i, of the objects in the sample using,

$$S_i = \sum_{k \in I_N} \sum_{\ell=1}^{g_k} w_{k\ell}\delta^i_{k\ell} + \sum_{k \in I_M} w_k x_{ik}, \qquad i=1,\ldots,n \quad (3.52)$$

For new objects, similar treatments in discretizing the attribute scores are applied. Their overall scores can then be computed as in (3.52). Objects with overall scores greater than c are classified into G_1 and objects with overall scores less than c are classified into G_2.

Similar to the discussions of (GP1) in section 3.3.2, if prior information is available in a classification problem, this information can be incorporated in (GP2). In particular, both the attribute weights, w_k and $w_{k\ell}$, can be restricted to be either positive or negative by imposing positive or negative sign constraints in (GP2).

3.4.4 Simulation Experiment

A simulation experiment is conducted to investigate the performance of linear programming approaches, logistic regression, and Fisher's linear discriminant function, with and without the discretization procedures to classification problems with non-monotonic attributes. Four cases are considered.

Case I: Samples are drawn from two multivariate normal populations with three attributes. Population π_1 is distributed as $N(\upsilon_1, I)$ and population π_2 is distributed as $N(\upsilon_2, I)$, where $\upsilon_1=[1\ 1\ 1]$ and $\upsilon_2=[0\ 0\ 0]$, respectively.

Case II: Population π_1 is distributed as $N(\upsilon_1, I)$ and $\upsilon_1=[1\ 1\ 1]$ (same as in Case I). The sample scores of the three attributes in population π_2 are drawn independently. The sample scores of the first two attributes are drawn from a normal distribution with mean equals to zero and variance equals to one, while the sample scores of the third attribute are drawn from the following distribution [Johnson, 1987]:

$$pN(-1,1) + (1-p)N(3,1). \tag{3.53}$$

With probability p, the process is realized from N(-1,1), and with probability (1-p), the process is realized from N(3,1). With the probability p equals to 0.5, the shape of the distribution is bimodal. In Case II the third attribute score in population π_2 is non-monotonic.

Case III: Population π_1 is the same as in Case I. The sample scores of the three attributes in population π_2 are drawn independently from (3.53). With this set up, all the three attribute scores in population π_2 are non-monotonic.

Case IV: The sample scores of the three attributes in population π_1 are drawn independently from a uniform distribution with mean 7 and variance 8.333. The sample scores of the three attributes in population π_2 are drawn independently from a bimodal uniform distribution (with 0.5 probability of being drawn from a uniform distribution with mean 1.5 and variance 2.083, and 0.5 probability of being drawn from a uniform distribution with mean 12.5 and variance 2.083). All the three attribute scores in population π_2 for Case IV are non-monotonic.

Ten data sets are generated for each Case. Each data set contains 130 cases with 65 cases from each population. Fifteen cases from each population are used as the development sample and the remaining 100 cases as the validation sample.

The classification methods are applied with and without discretization of attribute scores in this simulation experiment. For example, the linear programming approach without discretization is to solve (GP2) when I_N is empty and I_M contains all the attributes. This is not originally designed to solve the problems in Case II, III, and IV. Consequently, I suggest using the new linear goal programming approach (GP2) when the attributes are non-monotonic. Moreover, the procedures to discretize the attribute scores can also be applied to both logistic regression and Fisher's linear discriminant function using dummy variables. Hence, classification methods with discretization of attribute scores are studied. When applying the classification methods with discretization to the development samples, the following four configurations of discretization are used:

(MM,3,3,3): All three attribute scores were discretized into 3 levels,

(MM,C,C,3): The third attribute score was discretized into 3 levels,

(MM,4,4,4): All three attribute scores were discretized into 4 levels,

(MM,C,C,4): The third attribute score was discretized into 4 levels,

where MM represents the methods used; GP for our linear programming model (GP2); LG for logistic regression; FD for Fisher's discriminant function. Classification methods without discretization are represented by (MM,C,C,C). For example, (GP,C,C,C) is the linear programming model without discretization of any attribute score.

(MM,C,C,C) is applied to all the four cases. To reflect a situation where only the third attribute is non-monotonic, (MM,C,C,3) and (MM,C,C,4) are applied to Case II. For Case III and Case IV, since all attributes are designed to be non-monotonic, (MM,3,3,3) and (MM,4,4,4) are applied.

There are many ways to discretize the scores of an attribute into different levels. The following suggestion is just one of the many reasonable approaches. For non-monotonic attributes, if we discretized their scores into three levels, the objects with attribute scores in the middle level are expected to have higher chances of belonging to one group, while the objects with attribute scores in the first level or the third level are expected to have higher chances of belonging to the other group. In this simulation experiment, since the sample size of the two populations in the development sample are equal, I choose to group the top 25% of the highest attribute values into the first level, the next 50% into the second level, and the bottom 25% into the last level. For four levels, the four levels are grouped by the top 25% of the highest attribute values, the next 25%, the next 25%, and the bottom 25%, respectively. With four levels, we expect either level two and three representing one group and level one and four representing the other group, or level one and three representing one group and level two and four representing the other group.

In Case I, both populations are multivariate normally distributed as assumed by discriminant analysis, the correct models to use are the original classification methods without discretization. As a result, only the classification methods without discretization are applied to Case I. The average hit rates (validation samples) of linear programming, logistic regression, and Fisher's discriminant analysis are 77.3%, 77.8%, and 78%, respectively. The average hit rates of the validation samples of all the methods in Case II to Case IV are reported in Table 3.8.

In Case II, only the third attribute score is non-monotonic, and the average hit rates of (LG,C,C,C) and (FD,C,C,C) are 74.6% each, and 73.6% for (GP,C,C,C). When the third attribute score is discretized as in (MM,C,C,3) and (MM,C,C,4), the range of the average hit rates is from 79.0% to 80.1%. These results suggest that if some attribute score is non-monotonic, classification methods with discretization of the non-monotonic attribute level are better models than the original classification methods without discretization.

Table 3.8: Average Hit Rates of all the Methods

Methods	Case II	Case III	Case IV
Without Discretization			
(GP,C,C,C)	73.6	51.5	52.9
(LG,C,C,C)	74.6	47.6	50.2
(FD,C,C,C)	74.6	48.1	50.2
With Discretization			
(GP,3,3,3)	—	82.0	88.8
(LG,3,3,3)	—	80.7	86.2
(FD,3,3,3)	—	80.3	82.3
(GP,C,C,3)	79.1	—	—
(LG,C,C,3)	79.3	—	—
(FD,C,C,3)	79.6	—	—
(GP,4,4,4)	—	80.3	84.4
(LG,4,4,4)	—	79.1	83.8
(FD,4,4,4)	—	79.6	81.1
(GP,C,C,4)	79.0	—	—
(LG,C,C,4)	79.9	—	—
(FD,C,C,4)	80.1	—	—

In both Case III and Case IV, the classification methods without discretization of attribute scores perform very poorly. Since all the three attributes are non-monotonic, this result is expected. In Case III, among the classification methods with discretization of attribute scores, (GP2) has the highest average hit rates in both (MM,3,3,3) and (MM,4,4,4). Furthermore, in Case IV when the samples are drawn from uniform distribution, (GP2) again has the best classification performance.

3.5 CONCLUSION

The goal programming model (GP1) is based on the order preservation of selected pairs of membership probabilities in the development sample. It has worked well in the M.B.A. admission problem and in the simulation experiment. The different possible variations of using (GP1) need to be explored in terms of the quality of solutions and the difficulty in execution. As pointed out by Srinivasan [1975], it is far more efficient to solve the dual of (GP1) with a bounded variable simplex algorithm. Linear programming sensitivity analysis may be used to improve the development sample by identifying and deleting cases with bad membership probabilities. It will be interesting to see how well (GP1) can be used as a general tool of probability estimation in other applications.

The results of the simulation experiment with non-monotonic attributes suggest that when some or all of the attribute scores are non-monotonic in nature, classification approaches with discretization of attribute scores perform much better than the classification approaches without discretization. Although the procedure for discretizing the attribute scores can also be applied to both logistic regression, and discriminant analysis using dummy variables, (GP2) has a better classification performance than these two approaches in the simulation experiment. Moreover, as discussed earlier, another advantage of using (GP2) is that it can incorporate prior information more easily than statistical discriminant analysis and logistic regression.

Appendix 3.1: Listing of 68 Cases

Attribute values are listed in the following order:

JOBL	YMGT	JOBS	DEG	YEDD	LETT	QGMAT	TGMAT	LETA	PROB	
4	5	4	7	17	3	37	600	4	.86	*
5	6	2	4	4	5	38	640	5	.6	
3	8	0	2	2	3	41	650	4	.58	
1	8	0	3	17	5	25	460	1	.19	*
4	10	1	6	10	4	36	560	5	.82	
3	0	3	7	4	3	40	590	4	.49	
3	0	1	5	8	0	38	630	3	.63	*
4	8	2	6	9	4	40	610	5	.91	*
1	16	2	6	16	0	39	614	3	.99	
1	5	0	7	1	4	29	450	3	.26	*
3	3	1	6	6	3	41	530	4	.31	
4	6	1	7	10	3	27	450	4	.52	
4	8	1	5	12	5	35	530	3	.49	*
1	0	0	6	10	4	30	600	3	.49	
3	0	2	3	8	3	32	520	3	.27	*
1	7	3	6	8	4	30	560	1	.49	
2	7	1	6	6	0	23	510	4	.49	
3	0	0	3	22	4	23	460	3	.17	*
3	0	2	5	10	0	26	460	3	.28	
3	12	1	4	2	4	27	480	6	.19	*
3	6	0	3	3	4	15	350	5	.11	
3	2	1	6	12	3	34	470	3	.28	*
3	0	0	7	2	1	29	590	3	.47	
2	2	3	6	14	0	43	690	4	.67	
1	7	1	5	7	4	26	510	2	.36	
6	15	0	5	1	0	32	640	3	.53	*
3	3	2	2	6	3	44	700	5	.61	*
4	7	2	2	6	4	35	580	5	.64	
1	9	1	9	20	4	25	450	1	.49	*
1	0	1	5	10	3	28	510	2	.25	
3	4	5	4	4	4	36	560	4	.49	*
2	3	4	5	6	5	40	610	5	.54	*
3	2	1	3	5	2	34	520	5	.34	*
4	6	1	8	6	5	25	490	4	.47	
1	2	1	3	9	4	21	530	1	.23	
3	0	2	4	7	2	45	650	3	.57	
4	10	3	7	17	4	38	590	5	.63	*
4	10	0	4	4	0	27	490	4	.29	
4	14	1	6	17	2	32	580	5	.35	*
3	0	3	6	6	2	35	500	4	.37	
4	2	1	5	16	2	46	680	4	.55	*
4	7	1	6	11	6	45	640	4	.81	
3	0	0	6	13	2	31	430	5	.27	*
3	1	1	7	5	4	39	560	5	.28	
1	4	3	5	18	3	21	450	1	.4	*
3	0	1	8	1	4	35	570	4	.56	
4	3	3	7	7	5	45	710	6	.91	
3	0	5	6	4	4	27	470	3	.27	*
1	3	2	6	8	4	28	430	1	.12	*
4	2	1	8	6	5	52	780	4	.85	

Listing of 68 Cases (continued)

JOBL	YMGT	JOBS	DEG	YEDD	LETT	QGMAT	TGMAT	LETA	PROB	
1	0	0	5	16	3	29	470	1	.29	*
2	5	1	6	8	0	28	390	4	.1	*
3	3	1	6	14	4	30	480	4	.15	
3	6	1	1	25	3	21	400	5	.11	*
1	14	1	3	34	3	14	370	1	.2	*
4	3	2	6	7	4	32	560	3	.38	*
4	9	2	8	9	4	45	600	5	.76	
1	18	1	5	18	3	26	470	1	.37	*
2	14	2	6	24	0	34	520	5	.49	*
3	0	2	2	8	3	25	450	3	.2	
1	5	1	5	13	3	31	540	1	.43	*
1	0	2	3	8	4	19	410	3	.19	
4	3	2	7	7	2	37	610	4	.32	*
4	9	0	6	13	2	33	600	5	.58	
4	11	0	6	13	4	38	500	4	.42	*
4	5	2	5	7	5	39	620	3	.41	*
1	8	0	6	11	5	36	570	1	.56	
4	5	1	6	9	5	30	600	3	.41	

* : development sample

Appendix 3.2 : Listing of SPSSX Programme

```
$RUN *SPSSX SPRINT=-OUTX
FILE HANDLE OSS / NAME="-Q"
DATA LIST FREE FILE=OSS /
                JOBL,YMGT,JOBS,DEG,YEDD,LETT,
       QGMAT,TGMAT,LETA,PROB
VARIABLE LABELS JOBL   "JOB LEVEL" /
                YMGT   "YEARS IN MANAGEMENT" /
                JOBS   "JOB SWITCHES" /
                DEG    "DEGREE" /
                YEDD   "YEARS SINCE EDUCATION" /
                LETT   "LETTER OF SUPPORT FROM COMPANY" /
                QGMAT  "QUAN GMAT SCORE" /
                TGMAT  "TOTAL GMAT SCORE" /
                LETA   "AVERAGE OF LET1,LET2,LET3" /
                PROB   "ACCEPTANCE PROBABILITY 0,1 "/
COMPUTE         CLASS=(PROB GE 0.50)/
DISCRIMINANT    GROUPS=CLASS(0,1)/
                VARIABLES=JOBL TO LETA/
                ANALYSIS=JOBL TO LETA/
                PRIORS=SIZE
STATISTICS      11 12 13 14
FINISH
```

CHAPTER 4

LINEAR GOAL PROGRAMMING IN PREFERENCE DECOMPOSITION

4.1 INTRODUCTION

Preference decomposition is a class of methods used to estimate the values of the attribute levels (part worths) of the alternatives given an individual's preference judgments of these alternatives. As noted by Green and Srinivasan [1978], the primary objective of performing preference decomposition is to predict the preference orderings of new alternatives. Usually, a set of well selected alternatives is presented to a respondent who then expresses his/her preferences for the alternatives in terms of either nonmetric or metric measurements. For example, in order to study how consumers value the attributes of a certain class of products (alternatives), a consumer (respondent) may be asked to rank order ten different products which are characterized by their differences in the attribute levels. The attributes of the products can be brand name, price level, weight of the product, special feature, quality and so on. Then the additive preference decomposition estimates a part-worth value for each level of each attribute. In nonmetric preference decomposition, the respondent's preference judgments are represented by either ranking of all alternatives or a subset of paired comparisons of the alternatives. Common methods for estimating the parameters in nonmetric preference decomposition are MONANOVA [Kruskal, 1965], PREFMAP [Carroll, 1972], and LINMAP [Srinivasan and Shocker, 1973]. Ratings and constant sum pairwise comparisons [Torgerson, 1958] are methods used to measure intervally-scaled preference judgments. The common metric approach in preference decomposition includes ordinary least squares regression (OLS) with various forms of dummy variables [Johnston, 1972].

I develop a linear goal programming model (GPD) for preference decomposition where the input preference judgments are measured in ratio scale. For each selected pair of alternatives, the respondent is asked to state which alternative is more preferable and by (at least) how many times the chosen alternative is more preferable to the other. Goal programming is used to estimate the part worths of all the attributes by minimizing the badness of fit between the input preferences and the preferences derived from the estimated part worths. Similar to (LINMAP), additional prior subjective constraints, if any, on the part-worths such as range, monotonicity or bounds can be easily enforced in (GPD).

I conduct a simulation study to compare the predictive validity of the two linear programming approaches (GPD) and (LINMAP), and also ordinary least squares in estimating the preference function. The simulated overall preferences are of interval scale. Predictive validity is evaluated by the Pearson correlation coefficient and the Spearman rank coefficient between the input preferences and the derived preferences obtained from each approach. Ordinary least squares and (GPD) both have higher average Pearson correlation coefficients and average Spearman rank coefficients than (LINMAP) in this simulated study.

4.2 REVIEW OF PREFERENCE DECOMPOSITION

Consider the analysis of the preference judgments on n alternatives which are completely described in terms of p attributes. Let X be a nxp matrix where x_{ik} denotes the k-th attribute value of the i-th alternative for i=1,2,...,n, and k=1,2,...,p. Thus the i-th row of X describes the i-th alternative in terms of the p attributes. Usually, it is assumed that for each k, where k=1,2,...,p, the k-th attribute has g_k distinct values (levels). For k=1,2,...,p and ℓ=1,2,...,g_k, let $y_{k\ell}$ denote the ℓ-th value of the k-th attribute.

The part-worth function model [Green and Srinivasan, 1978] posits that for each i=1,2,...,n, the overall preference, S_i, associated to the i-th alternative is given by

$$S_i = \sum_{k=1}^{p} \sum_{\ell=1}^{g_k} f_k(y_{k\ell})\delta^i_{k\ell} \tag{4.1}$$

where $f_k(y_{k\ell})$ denotes the part worth of $y_{k\ell}$ and

$$\delta^i_{k\ell} = \begin{cases} 1, & \text{if the value of the k-th attribute of the i-th alternative is } y_{k\ell} \\ 0, & \text{otherwise.} \end{cases}$$

The dummy variable $\delta^i_{k\ell}$ captures the presence (=1) or absence (=0) of $y_{k\ell}$ in the k-th attribute of the i-th alternative. Equation (4.1) is also used by Wittink and Cattin [1981] to generate data for a Monte Carlo study of alternative estimation methods for preference decomposition. The aim of the preference decomposition is to estimate each of the $g_1+g_2+...+g_p$ part worths $f_k(y_{k\ell})$ for $k=1,2,...,p$ and $\ell=1,2,...,g_k$, from data matrix X and input preference in Ω. Let $w_{\ell k}$ denotes the estimated value for the part worth $f_k(y_{\ell k})$. It follows from (4.1) that S_i is estimated by $\hat{S}_i$ where

$$\hat{S}_i = \sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta^i_{k\ell}. \tag{4.2}$$

Other forms of the preference function are the vector model and the ideal point model. The overall preference, S_i, of the vector model is given by:

$$S_i = \sum_{k=1}^{p} w_k x_{ik} , \tag{4.3}$$

where x_{ik} is the k-th attribute value of the i-th alternative and w_k is the weight of the k-th attribute. For the ideal point model, the weighted Euclidean metric of the i-th alternative from the ideal point is given by $\left[\sum_{k=1}^{p} w_k(x_{ik} - z_k)^2\right]^{1/2}$, where z_k is the k-attribute value of the ideal point (the most preferred point in the p-th dimensional attribute space) of the respondent. The overall preference, S_i, is stated as follows:

$$S_i = \sum_{k=1}^{p} w_k(x_{ik} - z_k)^2. \qquad (4.4)$$

The smaller the value of S, the closer the alternative to the ideal point; and therefore, the more preferred will be the alternative.

Monotonic ANOVA (MONANOVA) was proposed by Kruskal [1965]. MONANOVA attempts to find a set of $w_{k\ell}$ and a set of Z_i (where $Z_i=f(S_i)$ are the monotonically transformed values of S_i) for $i=1,\ldots,n$, $k=1,\ldots,p$ and $\ell=1,\ldots,g_k$, in order to minimize the badness of fit function or stress, S:

$$S = \left[\frac{\sum_{i=1}^{n} [Z_i-\hat{S}_i]^2}{\sum_{i=1}^{n} [\hat{S}_i-\bar{\hat{S}}]^2} \right]^{1/2} \qquad (4.5)$$

where $\hat{S}_i = \sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta^i_{k\ell}$, and $\bar{\hat{S}}$ is the mean of $\hat{S}_i$. The stress function has a lower limit of zero and an upper limit of unity.

PREFMAP [Carroll, 1972] relates preference data to a multidimensional solution. In nonmetric PREFMAP, an individual's overall preference of an alternative, is assumed to be ordinally related to the weighted squared distance of the alternative to the individual's ideal point. In metric PREFMAP, however, overall preference is assumed to be linearly related to the weighted squared distance from the ideal point. Using multiple regression, both the weights and ideal points can be estimated. The multiple regression model is stated as follows:

$$S_i = a - b\sum_{k=1}^{p} w_k z_k^2 - b\sum_{k=1}^{p} w_k x_{ik}^2 + 2b\sum_{k=1}^{p} w_k z_k x_{ik} + \varepsilon_i, \qquad (4.6)$$

where a and b are parameters to be estimated, and ε_i is an error term.

Srinivasan and Shocker [1973] developed a linear programming model (LINMAP) in preference decomposition. Let $\Omega = \{(i,j)\}$ be the set of all paired comparisons obtained from the respondent with the i-th alternative being more preferred to the j-th alternative. Using the part worth function model as the preference function, their model is stated as follows:

$$\text{MIN} \quad \sum_{(i,j)\in\Omega} d_{ij} \tag{4.7}$$

$$\text{S.T.} \quad \sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} - \sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{j} + d_{ij} \geq 0, \ \forall(i,j) \text{ in } \Omega \tag{4.8}$$

$$\sum_{(i,j)\in\Omega} \left[\sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} - \sum_{k=1}^{p} \sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{j} \right] = 1 \tag{4.9}$$

$$w_{k\ell}\geq 0, \ d_{ij}\geq 0.$$

$w_{k\ell}$ is the estimated value of the part worth $f_k(x_{k\ell})$ for $k=1,2,\ldots,p$ and $\ell=1,2,\ldots,g_k$.

(LINMAP) is designed for ordinally-scaled preference data. When ordinally-scaled preference data is collected by paired comparison approach, the respondent is asked to state which alternative is more preferred. However, more information may be contained in ratio-scaled preference data when the respondent is asked to state how many times the preferred alternative is being more preferred to the other alternative. Since (LINMAP) does not incorporate the higher level preference information when estimating the part worth, I develop a linear programming model which utilizes ratio-scaled preferences judgments as the input data.

4.3 MODEL FORMULATION

In this section, I introduce a linear goal programming model for preference decomposition (GPD), with the input preference judgments measured in ratio scale. Let the overall preference associated to the i-th alternative be quantified by S_i,

$i=1,2,\ldots,n$, and for each pair of alternatives i and j, the i-th alternative is preferred to the j-th alternative whenever $S_i>S_j$. Let $\Omega = \{(i,j,t_{ij})\}$ be the set of all paired comparisons obtained from the respondent, with the i-th alternative being t_{ij} times more preferred to the j-th alternative for $t_{ij}\geq 1$. Then for each (i,j,t_{ij}) in Ω, we expect the ratio (S_i/S_j) to be t_{ij} under perfect consistency. That is,

$$S_i - t_{ij}S_j = 0 , \qquad \forall(i,j,t_{ij}) \text{ in } \Omega. \tag{4.10}$$

To be consistent with the input preference structure (4.10) in Ω, it is desirable that the estimated overall preferences $\hat{S}_i$, $i=1,2,\ldots,n$, satisfy

$$\hat{S}_i - t_{ij}\hat{S}_j = 0 , \qquad \forall(i,j,t_{ij}) \text{ in } \Omega. \tag{4.11}$$

Since exact equality in (4.11) is too restrictive, deviational variables d^+_{ij} and d^-_{ij} are introduced to allow for inconsistencies which may exist in the input preference structure in Ω. Thus (4.11) becomes

$$\hat{S}_i - t_{ij}\hat{S}_j + d^-_{ij} - d^+_{ij} = 0 , \qquad \forall(i,j,t_{ij}) \text{ in } \Omega. \tag{4.12}$$

The sum $\sum_{(i,j,t_{ij})\in\Omega} (d^+_{ij} + d^-_{ij})$ of all d^+_{ij} and d^-_{ij} in (4.12) is the badness of fit between S_i and $\hat{S}_i$ with respect to the input preference structure in Ω. Substituting (4.1) into (4.12), we get

$$\sum_{k=1}^{p}\sum_{\ell=1}^{g_k} w_{k\ell}\delta^i_{k\ell} - t_{ij}\left[\sum_{k=1}^{n}\sum_{\ell=1}^{g_k} w_{k\ell}\delta^j_{k\ell}\right] + d^-_{ij} - d^+_{ij} = 0, \tag{4.13}$$

$\forall$ (i,j,t_{ij}) in Ω. Thus the problem of finding the best solution $\{w_{k\ell}\}$ from the data matrix X and the input preference Ω reduces to a goal programming problem of finding $\{w_{k\ell}\}$ which minimizes the badness of fit $\sum_{(i,j,t_{ij})\in\Omega} (d^+_{ij} + d^-_{ij})$ subject to the constraints (4.13), $\forall$ (i,j,t_{ij}) in Ω.

However, it is sometimes difficult for the respondent to express the exact value of t_{ij} directly. It is better to ask the respondent "**at least**" how many times he/she prefers the i-th alternative to the j-th alternative instead of asking him/her the exact number of times he/she prefers the i-th alternative to the j-th alternative. Intuitively this allows more robust input from the respondent and hence more reliable results may be obtained from the goal programming model. Now suppose $\Omega = \{(i,j,t_{ij})\}$ is the set of all paired comparisons obtained from the respondent with the i-th alternative being **at least** t_{ij} times more preferred to the j-th alternative. Then for each (i,j,t_{ij}), the positive value of d_{ij}^- ($d_{ij}^+=0$) implies that $\hat{S}_i<t_{ij}\hat{S}_j$, which conflicts directly with the input preference in Ω. But the positive value of d_{ij}^+ ($d_{ij}^-=0$) implies that $\hat{S}_i>t_{ij}\hat{S}_j$, which is still consistent with the input preference in Ω. Consequently, d_{ij}^- should be minimized before d_{ij}^+ is minimized in the objective function.

I can now define formally the goal programming model for the preference decomposition as follows:

$$\text{(GPD) MIN } P_1\left[\sum_{(i,j,t_{ij})\in\Omega} d_{ij}^-\right] + P_2\left[\sum_{(i,j,t_{ij})\in\Omega} d_{ij}^+\right] \tag{4.14}$$

$$\text{S.T. } \sum_{k=1}^{p}\sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{i} - t_{ij}\left[\sum_{k=1}^{p}\sum_{\ell=1}^{g_k} w_{k\ell}\delta_{k\ell}^{j}\right] + d_{ij}^- - d_{ij}^+ = 0, \quad \forall(i,j,t_{ij})\in\Omega \tag{4.15}$$

$$\sum_{k=1}^{p}\sum_{\ell=1}^{g_k} w_{k\ell} \geq 1 \tag{4.16}$$

$$w_{k\ell}\geq 0,\ d_{ij}^-,\ d_{ij}^+\geq 0,\ \forall(i,j,t_{ij})\in\Omega.$$

P_1 is much greater than P_2 to allow for a larger penalty on the deviational variables d_{ij}^- than d_{ij}^+. Perhaps (GPD) should be solved as a preemptive goal programming problem [Kornbluth, 1973] so that no part of d_{ij}^+ is ever traded off by any part of d_{ij}^-. The primary objective is to minimize $\sum_{(i,j,t_{ij})\in\Omega} d_{ij}^-$ in attempting to achieve $\hat{S}_i \geq t_{ij}\hat{S}_j$. The secondary objective is to minimize $\sum_{(i,j,t_{ij})\in\Omega} d^+_{ij}$ for reaching $\hat{S}_i = t_{ij}\hat{S}_j$.

The inequity $\sum_{k=1}^{p}\sum_{\ell=1}^{k} w_{k\ell} \geq 1$ is the normalization constraint. It can be shown by direct verifications that the relative values of the optimal part worths $w_{k\ell}$ will remain the same if the right hand side of the normalization constraint is changed to any positive constant other than 1. Furthermore, for any nontrivial solution $\{w'_{k\ell}\}$ which is feasible in (GPD) without the normalization constraint, there exists a feasible solution $\{w''_{k\ell}\}$ in (GPD) such that the relative values of $\{w'_{k\ell}\}$ are the same as the corresponding relative values of $\{w''_{k\ell}\}$. Thus, the only effect of the normalization constraint in (GPD) is to avoid the trivial solution (all $w_{k\ell}=0$).

The $w_{k\ell}$ obtained from solving (GPD) can then be used to compute the estimated preference for any alternative by simply summing up $w_{k\ell}$ for all relevant part worths. Thus the rank ordering of all the alternatives can be determined by the magnitudes of their estimated preferences.

4.3.1 Discussion

Several parameter estimation methods in preference decomposition can be used to estimate the part worths. In particular, linear programming techniques for multidimensional analysis of preferences (LINMAP) have higher predictive validity over other methods when there is a dominant attribute [Wittink and Cattin, 1981]. (LINMAP) posits that $S_i - S_j \geq 0$ whenever the i-th alternative is preferred to the j-th alternative. The overall preference, S_i, may also be defined in terms of the distance from the i-th alternative to an ideal alternative. In (GPD), $S_i - S_j \geq 0$ is replaced by $S_i/S_j \geq t_{ij}$ with $t_{ij} \geq 1$. Conceptually, $S_i/S_j \geq t_{ij}$ has more preference information than $S_i - S_j \geq 0$ when $t_{ij} > 1$.

Furthermore, the only feasible solution cut off by the normalization constraint in (GPD) is the trivial solution. There is no unnecessary restrictions introduced by the normalization constraint.

Usually, it is easier for a respondent to identify the preferred alternative as compared to express the magnitude of his/her preference. The cost of having more preference information in $S_i/S_j \geq t_{ij}$ is the difficulty in eliciting t_{ij} from the respondent. One possible method to determine t_{ij} is the constant sum pairwise comparisons by Torgerson [1958]. Another way to assist a respondent in determining t_{ij} in the paired comparison is the nine points intensity scale used in the Analytic Hierarchy Process [Saaty, 1977]. Harker and Vargas [1987] supported the use of the 1 to 9 scale in Saaty' Analytic Hierarchy Process. The ratio scale used in the Analytic Hierarchy Process ranges from 1 to 9 and represents the preference of one alternative to the other such as:

1 - indifference: at least 1 time,
3 - slightly more preferred: at least 3 times,
5 - strongly more preferred: at least 5 times,
7 - demonstratedly more preferred: at least 7 times,
9 - absolutely more preferred: at least 9 times.

Intermediate values of 2,4,6 and 8 are used when a more refined compromise is needed. As long as all the paired comparisons of a respondent are underestimated in the 9 points scale, the constraints $S_i/S_j \geq t_{ij}$ implicitly implied by (4.15) in (GPD) are accurate and thus the corresponding solutions are meaningful.

Theoretically, there are $(g_1)(g_2)\ldots(g_p)$ distinct alternatives from which n of them are selected for preference evaluation, and there are n(n-1)/2 distinct pairwise comparisons. When n is large, it is not feasible to conduct all the paired comparisons. Thus the set Ω used in (GPD) may be replaced by a well selected subset Ω' of Ω. In practice, only the paired comparisons in Ω' would be elicited from the respondent and incorporated into (GPD). Orthogonal array is one of the many methods available for the selection of some representative subset Ω' from Ω [Green, 1974]. Table 4.1 illustrates an orthogonal array of four attributes, each with three levels. It should be pointed out that more reliable solutions would be obtained from (GPD) by increasing the size of the subset Ω'.

Table 4.1: A Symmetrical Orthogonal Array for the 3^4 Factorial Design

	Attributes and Levels			
Alternative	X_1	X_2	X_3	X_4
1	1	1	1	1
2	1	2	2	3
3	1	3	3	2
4	2	1	2	2
5	2	2	3	1
6	2	3	1	3
7	3	1	3	3
8	3	2	1	2
9	3	3	2	1

4.3.2 Computation Example

Consider the preference of alternatives with four attributes, each with three levels of values. By using orthogonal arrays, nine alternatives are selected to be included in the development sample with the remaining 72 alternatives in the validation sample. To test whether (GPD) can perform well and predict preferences from other models, the input preference is generated by the vector model (4.2). The attribute values $y_{k\ell}$ for the nine chosen alternatives X, the weights w_k for the four attributes W and the overall preferences S_i of the alternatives XW are:

$$X = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 3 & 2 & 10 \\ 1 & 5 & 4 & 2 \\ 2 & 1 & 2 & 2 \\ 2 & 3 & 4 & 1 \\ 2 & 5 & 1 & 10 \\ 3 & 1 & 4 & 10 \\ 3 & 3 & 1 & 2 \\ 3 & 5 & 2 & 1 \end{bmatrix}, \quad W = \begin{bmatrix} 6 \\ 3 \\ 4 \\ 3 \end{bmatrix}, \quad XW = \begin{bmatrix} 16 \\ 53 \\ 43 \\ 29 \\ 40 \\ 61 \\ 67 \\ 37 \\ 44 \end{bmatrix}, \quad XW+\epsilon = \begin{bmatrix} 21.1 \\ 56.1 \\ 42.2 \\ 25.9 \\ 32.3 \\ 65.6 \\ 67.8 \\ 35.6 \\ 31.5 \end{bmatrix}$$

A random error value, ϵ_i is added to each of the nine overall preferences in XW. The random error has zero mean and a constant variance. The perturbed values of the nine overall preferences $XW+\epsilon$ are used as input into (GPD). The part-worth values $w_{k\ell}$ obtained from solving (GPD) and the original attribute values $y_{k\ell}$ are given in Table 4.2 below:

Table 4.2: The Part-worth values $w_{k\ell}$ obtained from (GPD) and the original attribute values $y_{k\ell}$

	ATTRIBUTE							
	$y_{k\ell}$				$w_{k\ell}$			
LEVEL	1	2	3	4	1	2	3	4
1	1	1	1	1	0.000	0.000	0.088	0.079
2	2	3	2	2	0.011	0.025	0.065	0.129
3	3	5	4	10	0.041	0.065	0.141	0.356

The estimated part-worth values $w_{k\ell}$ are used to compute the estimated overall preference $\hat{S}_i$ for the nine alternatives in the development sample and the 72 alternatives in the validation sample. The predictive validity is supported by the Pearson correlation coefficient of 0.942 and the Spearman rank coefficient of 0.944 between the original S_i values and the estimated $\hat{S}_i$ values in the validation sample. The Pearson correlation coefficient and the Spearman rank coefficient between the original S_i values and the estimated $\hat{S}_i$ values in the development sample are 0.942 and 0.883 respectively. (GPD) has performed well in this example.

4.4 SIMULATION EXPERIMENT

I design a simulation experiment to evaluate the performance of (GPD), (LINMAP) and Ordinary Least Squares (OLS) approaches in preference decomposition. The data generation procedure is similar to Wittink and Cattin [1981]. The input preference S_i is derived from the part-worth function model, equation (4.1), where all the part worths are drawn from a normal distribution with mean zero and variance one. In this experiment, 30 data sets are used, each data set consisting of 81 alternatives with four attributes, each having three levels of values or part worths. By using orthogonal arrays, nine alternatives are selected to be included in the development sample with the remaining 72 alternatives in the validation sample. A random error value is added to each of the nine overall preferences in the development sample. The random error has zero mean and its variance δ_e^2 is computed from the same formula used by Wittink and Cattin as follows:

$$E = \frac{\delta_e^2}{\delta_e^2 + \delta_s^2} , \qquad (4.17)$$

where δ_e^2 is the error variance and δ_s^2 is the variance in the overall preference S_i before the error term is introduced. I use three different values, 0.1, 0.2, and 0.35 for E in this experiment. The higher the value of E, the larger is the noise term being introduced into the input preference. I randomly generated 10 data sets for each of the E values, and the perturbed values of the nine overall preferences are used as input into (GPD), (LINMAP), and (OLS). The softwares I used for (LINMAP) and (OLS) are Conjoint Linmap [1989], and Conjoint Analyzer [1987], respectively.

The estimated part-worth values $w_{k\ell}$ obtained from the models are used to compute the estimated overall preference $\hat{S}_i$ for the 72 alternatives in the validation sample. The predictive validity is evaluated by the average Pearson correlation coefficient and the average Spearman rank coefficient between the original S_i values and the estimated $\hat{S}_i$ values in the validation samples. The results are reported in Table 4.3.

Table 4.3: Average Pearson correlation coefficient and average Spearman rank coefficient of (GPD), LINMAP and (OLS)

	Pearson correlation coefficient			Spearman rank coefficient		
E	(GPD)	(LINMAP)	(OLS)	(GPD)	(LINMAP)	(OLS)
0.1	0.967	0.913	0.966	0.957	0.912	0.955
0.2	0.927	0.875	0.930	0.927	0.876	0.929
0.35	0.837	0.788	0.838	0.829	0.792	0.827
average	0.910	0.857	0.911	0.904	0.860	0.903

The average Pearson correlation coefficient and the average Spearman rank coefficient of (GPD) and (OLS) are higher than the coefficients of (LINMAP) in all three cases. For the Pearson correlation coefficient, (GPD) is higher than (LINMAP) by 0.0541, 0.0522, and 0.0492 when E equals to 0.1, 0.2, and 0.35, respectively. For the Spearman rank coefficient, (GPD) is higher than (LINMAP) by 0.0451, 0.0511, and 0.0368 when E equals to 0.1, 0.2, and 0.35, respectively. In all three cases, both the average Pearson correlation coefficients and the average Spearman rank coefficients of (GPD) and (OLS) are very close to each other.

4.5 CONCLUSION

A linear goal programming model (GPD) is introduced to estimate the part worths in preference decomposition. The model uses ratio scaled input preference judgments which contain more preference information than ordinal scaled preference judgments commonly used in many preference decomposition models. Since the objective of (GPD) is to directly minimize the badness of fit between the input preferences and the derived preferences, more reliable estimates for the part worths may be expected from the model. In this simulated experiment (GPD) and (OLS) have better performance than (LINMAP) in terms of the Pearson correlation coefficient and the Spearman rank coefficient between the input preference and the derived preference. Additional empirical studies are needed to evaluate the efficiency and the predictive power of (GPD).

CHAPTER 5

CONCLUSION

In this paper, I develop several new linear goal programming models to solve the problems in classification and preference decomposition. In chapter 2, with an analog of multicriteria optimization framework, I provide a systematic way of generating a whole array of meaningful criteria and introduce five new mathematical programming models for cluster analysis. The ability to generate many meaningful criteria for evaluating cluster solutions increases the power and the flexibility of applying mathematical programming approaches to cluster analysis. This will likely motivate more frequent applications of mathematical programming approaches to cluster analysis. The computational results obtained from applying the clustering models to a published data set support the use of mathematical programming models. Future researches applying the above techniques to solve the problems in cluster analysis should be very interesting.

In chapter 3, I introduce two new linear goal programming models for discriminant analysis. The model (GP1) incorporates the within group discriminate information in discriminant analysis. It is based on the order preservation of selected pairs of membership probabilities in the development sample. Intuitively, this model may provide more accurate estimations of the attribute weights than other techniques which ignore the within group discriminate information. It has good performance in both the M.B.A. admission problem and the simulation experiment. The model (GP2) allows non-monotonic attributes to be included in the classification function. Since in some situations, the implied monotonicity of the attribute scores may be violated, thus (GP2) provides more flexibility in applying linear programming techniques to classification problems. The results from the simulation experiment support the use of (GP2) to solve the classification problems when non-monotonic attributes are present.

The existing linear programming approaches in preference decomposition only allow the input preference judgments to be measured in ordinal scale. Theoretically, ratio scale preference judgments contain more information than ordinal scale preference judgments. Therefore, in chapter 4, I introduce a linear goal programming model (GPD) for preference decomposition with the input preference judgments measured in ratio scale. (GPD) performs well in the simulation experiment. Although the performance of (GPD) and (OLS) are close in the simulation experiment, (GPD) has the advantage that additional constraints which reflect useful prior information can easily be added.

REFERENCES

Anderson, J.A. (1972) "Separate Sample Logistic Discrimination", *Biometrika*, 59, 19-35.

Aronson, J.E., and Klein, G. (1989) "A Clustering Algorithm for Computer-Assisted Process Organization," *Decision Sciences* 20, 730-746.

Arthanari, T.S., and Dodge, Y. (1981) *Mathematical Programming in Statistics.* Wiley, NY.

Bajgier, S.M., and Hill, A.V. (1982) "An Experimental Comparison of Statistical and Linear Programming Approaches to the Discriminant Problem", *Decision Sciences* 13, 604-618.

Blashfield, R.K. and Aldenderfer, M.S. (1978) "The Literature on Cluster Analysis," *Multivariate Behavioral Research* 13, 271-295.

Carroll, J.D. (1972) "Individual Differences and Multidimensional Scaling", *Multidimensional Scaling: Theory and Applications in Behavioral Sciences,* Vol.I, R.N. Shepard et al., eds. Seminar Press, New York, 105-155.

Chankong, V. and Haimes, Y.Y. (1983) "Optimization-Based Methods for Multiobjective Decision-Making: An Overview", *Large scale System* 5, 1-33.

Charnes, A., Cooper, W.W., and Ferguson, R.O. (1955) "Optimal Estimation of Executive Compensation by Linear Programming", *Management Sciences* 1, 138-150.

Choo E.U., and Wedley, W.C. (1985), "Optimal Criterion Weights in Repetitive Multicriteria Decision Making", *Journal of Operational Research Society* 36, 983-992.

Church, R., Current, J., and Storbeck, J. (1991) "A Bicriterion Maximal Covering Location Formulation Which Considers the Satisfaction of Uncovered Demand," *Decision Sciences* 22, 38-52.

Conjoint Analyzer (1987), Software, Bretton-Clark, Morristown, NJ 07960.

Conjoint Linmap (1989), Software, Bretton-Clark, Morristown, NJ 07960.

Cox, D.R. (1970) *The Analysis of Binary Data.* London: Methuen.

Day, N.E. and Kerridge, D.F. (1967) "A General Maximum Likelihood Discriminant", *Biometrics* 23, 313-328.

Fisher, R.A. (1936) "The Use of Multiple Measurements in Taxonomic Problems", *Annals of Eugenics* 7, 179-188.

Flath, D. and Leonard, E.W. (1979) "A Comparison of Two Logit Models in the Analysis of Quantitative Marketing Data", *Journal of Marketing Research* 16, 533-538.

Freed, N. and Glover, F. (1981a) "A Linear Programming Approach to the Discriminant Problem", *Decision Sciences* 12, 68-74.

Freed, N. and Glover, F. (1981b) "Simple but Powerful Goal Programming Models for Discriminant Problems", *European Journal of Operational Research* 7, 44-60.

Freed, N., and Glover, F. (1986) "Evaluating Alternative Linear Programming Models to Solve The Two-Group Discriminant Problem", *Decision Sciences* 17, 151-162.

Ghosh, A. and Craig, C.S. (1986) "An Approach to Determining Optimal Locations for New Services," *Journal of Marketing Research* 23, 354-62.

Glover, F., Keene, S., and Duea, B. (1988) "A New Class of Models for the Discriminant Problem", *Decision Sciences* 19, 269-280.

Gordon, A.D. (1981) *Classification*, New York: Wiley.

Green, P.E. (1974) "On the Design of Choice Experiments Involving Multi-factor Alternatives", *Journal of Consumer Research* 1, 661-678.

Green, P.E. and Rao, V.R. (1971) "Conjoint Measurement for Quantifying Judgmental Data", *Journal of Marketing Research* 8, 355-363.

Green, P.E. and Srinivasan, V. (1978) "Conjoint Analysis in Consumer Research: Issues and Outlook", *Journal of Consumer Research,* 5, 103-123.

Hand, D.J. (1981) *Discrimination and Classification*. New York: Wiley.

Harker, P.T. and Vargas, L.G. (1987) "The Theory of Ratio Scale Estimation: Saaty's Analytic Hierarchy Process", *Management Sciences,* 33, 1383-1403.

Joachimsthaler, E.A., and Stam, A. (1988) "Four Approaches to the Classification Problem in Discriminant Analysis: An Experimental Study", *Decision Sciences* 19, 322-333.

Johnson, R.A. and Wichern, D.W. (1988) *Applied Multivariate Statistical Analysis*, 2nd Edition, New Jersey:Prentice Hall.

Johnson M.E. (1987) "*Multivariate Statistical Simulation*", John Wiley and Sons.

Johnston, J. (1972) *Econometrics Methods*, 2nd edition, New York, McGraw-Hill Book Co.

Koehler, G.J. and Erenguc, S.S. (1990) "Minimizing Misclassifications in Linear Discriminant Analysis", *Decision Sciences* 21, 63-85.

Kornbluth J. (1973) "A Survey of Goal Programming", *OMEGA* 1, 193-205.

Kruskal, J.B. (1965) "Analysis of Factorial Experiments by Estimating Monotone Transformations of the Data", *Journal of the Royal Statistical Society,* Series B, 27, 251-263.

Lee, C.K. and Ord, J.K. (1990) "Discriminant Analysis Using Least Absolute Deviations", *Decision Sciences* 21, 86-96.

MacQueen, J.B. (1967) "Some Methods for Classification and Analysis of Multivariate Observation," *Proceeding of 5th Berkeley Symposium on Mathematical Statistics and Probability,* 1, Berkeley, Calif. : University of California Press, 281-97.

Milligan, G.W. and Cooper, M.C. (1985) "An Examination of Procedures for Determining the Number of Clusters in a Data Set," *Psychometrika* 50, No.1, 159-179.

Ragsdale, C., and Stam, A. (1989) "A Robust Nonparametric Procedure to Estimate Response Functions for Binary Choice Models", presented at the CORS/TIMS/ORSA 1989 Conference, Vancouver, Canada.

Rao, M.R. (1971) "Cluster Analysis and Mathematical Programming," *Journal of the American Statistical Associations* 66, 622-626.

Revelle, C., Marks, D. and Liebman, J.C. (1970) "An Analysis of Private and Public Sector Location Models," *Management Science* 16, 692-707.

Saaty, T.L. (1977) "A Scaling Method for Priorities in Hierarchical Structures", *Journal of Mathematical Psychology* 15, 234-281.

SPSS Inc. (1988) *SPSS-X User's Guide*, 3rd ed., SPSS Inc.

Srinivasan, V., Jain, A, and Malhotra, N.K. (1983) " Improving Predictive Power of Conjoint Analysis by Constrained Parameter Estimation", *Journal of Marketing Research* 20, 1983, 433-438.

Srinivasan, V. (1975) "Linear Programming Computational Procedures for Ordinal Regression", *Journal of the Association for Computing Machinery* 23, 475-487.

Srinivasan, V. and Shocker, A.D. (1973) "Linear Programming Techniques for Multidimensional Analysis of Preferences", *Psychometrika* 38, 337-69.

Stam, A. and Joachimsthaler, E.A. (1989) "Solving the Classification problem in discriminant analysis via linear and nonlinear programming methods", *Decision Sciences,* 20, 285-293.

Stam, A. and Joachimsthaler, E.A. (1990) "A Comparison of a Robust Mixed-integer Approach to Existing Methods for Establishing Classification Rules for the Discriminant Problem", *European Journal of Operational Research*, 46, 113-122.

Stam, A. and Ragsdale, C.T. (1990) "A Robust Nonparametric Procedure to Estimate Response Functions for Binary Choice Models", *Operations Research Letters*, 9, 51-58.

Stam, A. and Jones, D.G. (1990) "Classification Performance of Mathematical Programming Techniques in Discriminant Analysis: Results for Small and Medium Sample Sizes", *Managerial and Decision Economics*, 11, 243-253.

Torgerson, W.S. (1958) *Theory and Methods of Scaling*, New York, John Wiley & Sons Inc.

Vinod, H.D. (1969) "Integer Programming and Theory of Grouping," *Journal of the American Statistical Association* 64, 506-519.

Wagner, H.M. (1959) "Linear Programming Techniques for Regression Analysis", *Journal of the American Statistical Association* 54, 206-212.

Wittink, D.R. and Cattin, P. (1981) "Alternative Estimation Methods for Conjoint Analysis: A Monte Carlo Study", *Journal of Marketing Research* 18, 101-106.