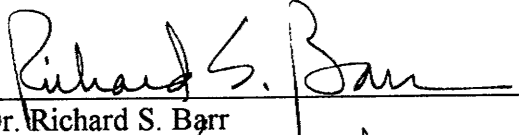
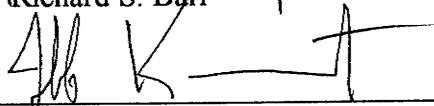
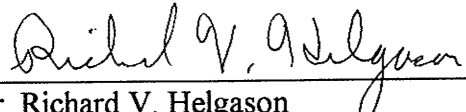


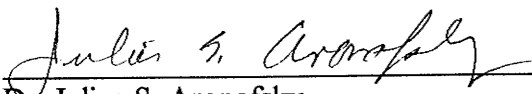
AN EXPERIMENTAL ANALYSIS OF OPTIMIZATION-BASED
DISCRIMINANT MODELS

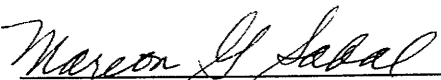
Approved by:


Dr. Richard S. Barr


Dr. Jeffery L. Kennington


Dr. Richard V. Helgason


Dr. Julius S. Aronofsky


Dr. Marion G. Sobol

AN EXPERIMENTAL ANALYSIS OF OPTIMIZATION-BASED
DISCRIMINANT MODELS

A Praxis Presented to the Graduate Faculty of
School of Engineering and Applied Science
Southern Methodist University

in

Partial Fulfillment of the Requirements

for the degree of

Doctor of Engineering

with a

Major in Engineering Management

by

Nandlal M. Singh

(B.Tech., Indian Institute of Technology, 1974)
(M.S.A.E., University of Cincinnati, 1976)
(M.B.A., Southern Methodist University, 1978)
(M.S.O.R., Southern Methodist University, 1979)

December 19, 1992

Singh, Nandlal M.

B.Tech., Indian Institute of Technology, 1974
M.S.A.E., University of Cincinnati, 1976
M.B.A., Southern Methodist University, 1978
M.S.O.R., Southern Methodist University, 1979

An Experimental Analysis of
Optimization-Based
Discriminant Models

Adviser: Associate Professor Richard S. Barr

Doctor of Engineering degree conferred December 19, 1992

Praxis completed November 25, 1992

One of the major purposes of discriminant analysis is to predict or classify entities into one of several mutually exclusive groups, based on information on a set of variables or attributes. This problem is known as the classification problem in discriminant analysis and has been claimed to be the ultimate objective of discriminant analysis.

Discriminant analysis methods have evolved largely from statistical models which rely on strong parametric assumptions, such as multivariate normal populations with same variance-covariance structure. Recently, applications of discriminant analysis have emerged from the fields of mathematical programming and artificial intelligence. Considerable research has been devoted to the development of alternate mathematical programming-based methods that are inherently distribution-free and offer intuitive solutions.

This praxis presents a survey of the existing methods in discriminant analysis and reports the results of an experimental comparison of two linear programming approaches and Fisher's procedure for the discriminant problem. The linear programming approaches are based on a hybrid model formulation proposed by Freed and Glover and a newly introduced method of successive hierarchical improvement by Glover. The models were constructed, validated and analyzed using Fisher's Iris data (classic data used in many discriminant studies), and financial data from 188 *failed* and *successful* U.S. banks.

Testing has revealed that the linear programming approach based on the hybrid model performed better than the traditional statistical analysis, and the successive goal method provided an even stronger solution.

TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	viii
ACKNOWLEDGMENTS	ix
Chapter	
1. SURVEY OF DISCRIMINANT ANALYSIS METHODS	1
1.1. Introduction	
1.2. Praxis' Objectives	
1.3. Notation	
1.4. Background	
1.4.1. Classification Analysis	
1.4.2. Discriminant Analysis	
1.4.3. Classification Errors	
1.5. Types of Linear Discriminant Analysis Methods	
1.5.1. Statistical Methods	
1.5.2. Search Methods	
1.5.3. Mathematical Programming Methods	
2. GLOVER'S NEW DISCRIMINANT ANALYSIS APPROACHES	39
2.1. Introduction	
2.1.1. MP-Based Discriminant Analysis--Problems and Issues	
2.2. Glover's MMD and MSID Models	
2.2.1. Maximize the Minimum Distance (MMD) Models	
2.2.2. Maximize Sum of Internal Deviations (MSID) Models	
2.2.3. Hybrid Models (Glover, 1990)	
2.2.4. Improved Solutions from Hybrid Models	

2.3. Successive Goal Approach	
2.3.1. Procedure for Successive Goal Approach	
3. AN EMPIRICAL COMPARISON OF METHODS	59
3.1. Introduction	
3.2. Selection of a Discriminant Method	
3.3. Methods Selected	
3.4. Experimental Design	
3.5. Experimental Results and Analysis	
4. SUMMARY AND CONCLUSIONS	75
APPENDIX	77
BIBLIOGRAPHY	87

LIST OF ILLUSTRATIONS

Figure	Page
1.1. A Two-Group Classifier	10
3.1. Classification Improvement over Hybrid Model	74
3.2. Successive Improvement of Classification	74

LIST OF TABLES

Table	Page
1.1. Confusion Matrix: Type I & Type II Errors	13
1.2. Cross-Validation Estimators	16
3.1. Variable Definitions	64
3.2. Sample Variable Statistics	65
3.3. 10-Fold Cross-Validation (Iris Data): All Models	70
3.4. Experimental Results for 10-Fold Cross-Validation (Iris Data)	70
3.5. 10-Fold Cross-Validation (Bank Data): Single Stage Hybrid Model	71
3.6. 10-Fold Cross-Validation (Bank Data): Successive Goal Method	71
3.7. Experimental Results for 10-Fold Cross-Validation (Bank Data)	72

ACKNOWLEDGEMENTS

My grateful thanks are due to my advisor, Professor Richard S. Barr, for his incessant guidance, encouragement and patience, and for being available to me at all odd hours—often at very short notice. I am indebted to him for introducing me to Professor Fred Glover and offering me the opportunity to work on Glover's new mathematical programming-based approach to discriminant analysis.

I thank Professor Jeffery L. Kennington, the Chairman of the Department of Computer Science and Engineering at SMU, for the impetus to complete this program. I consider it a privilege to have Professor Julius S. Aronofsky, Professor Richard V. Helgason, and Professor Marion G. Sobol examine me and serve on my advisory committee.

I would like to express my gratitude to all my well-wishers who were cheerleading me from the sidelines, and most of all, my heartfelt thanks to the cheeriest of all the cheerleaders, my wife Sheila, our daughter Natasha, my sister-in-law Asha, and mother-in-law Shant Devi.

Finally, I dedicate this study to the memory of my late father Mr. Manbodh Singh, who, along with my dear mother Mrs. Devraji Devi, has and always will be my constant source of inspiration.

CHAPTER 1

SURVEY OF DISCRIMINANT ANALYSIS METHODS

1.1. Introduction

Consider scenario 1: You are at your friendly bank for a consumer loan. Most probably you will be asked to fill out a loan application form. The information supplied by you and additional information obtained from other sources, such as credit bureaus, will then be used to compute a credit score. If your credit score exceeds a predetermined threshold set by the bank, your application is approved for the requested loan amount; otherwise, the application is denied.

Or consider scenario 2: You are at your family doctor for a full physical checkup. The clinical staff will probably collect samples of some body fluid, record numerous measurements, and then send the data to outside labs for testing. After getting the results from the labs, your family doctor (if you are lucky) or a clinical staff member will inform you of your health status by indicating the presence or absence of any disease based upon a set of predetermined criteria for that disease.

These are two very different environments, but both follow a similar process in unraveling the data supplied by you. These are the applications of a methodology called *discriminant analysis* where a decision maker (your banker or the doctor) classifies you, based on certain characteristics, into one of two or more groups. The problem you posed to your banker or the doctor is known as the classification problem in discriminant analysis.

One of the major purposes of discriminant analysis then is to predict or classify entities into one of several mutually exclusive and collectively exhaustive groups based on information on a set of independent variables or attributes. The focus of discriminant analysis is the determination of the functional forms (discriminant functions) and the estimation of their coefficients including the critical values. The resulting functions are then used to predict group membership of new observations. This is a widely used classification procedure in such fields as social sciences, business, medicine, and life sciences. More recently, it has been gaining widespread acceptance in inductive inference methods in artificial intelligence (AI) approaches to machine learning.

The methods for performing discriminant analysis have evolved largely from statistics, where variables used to characterize the members of the groups are assumed to be parametric (i.e. they follow certain well-defined distributions such as the multivariate normal distribution). Unfortunately, the assumption of normality is violated to a significant extent in many practical applications. Additionally, when the sample sizes are small, the *Central Limit Theorem* does not apply well, likely contributing to poor results, especially when linear classification functions are applied. The issue of normality is particularly relevant in studies employing financial ratios where distributions are flat, skewed, or dominated by outliers. Thus, the violations of the normality assumptions may bias the estimated errors and the tests of significance. Though statisticians have proposed and applied some newer statistical methods that are non-parametric (distribution-free), their applications have often proved to be cumbersome and confusing.

Over the last ten years, considerable research has been devoted to the development of mathematical programming methods for constructing discriminant functions. Mainly emerging from the field of *operations research (OR)*, these methods are inherently distribution-free and intuitive. Not only do these methods provide insight into special

relationships the variables might exhibit, they lend themselves well to computer implementations. Furthermore, the recent advances in computer technology have enabled mathematical programming packages to be extremely "analyst-friendly" and can now be conveniently applied to solve many diverse problems. Simple and direct, these new approaches compete with conventional approaches derived from the principles of classical and Bayesian statistics. More importantly, these methods enable the decision maker to play an active part in the analysis of the solution. This encourages user participation in the selection of appropriate discriminant criteria and allows flexibility in setting relative penalties for misclassification.

OR techniques are not the only alternatives to the classical statistics for classification problems. Recently, the *pattern recognition* (PR) techniques are emerging as important, useful, and rapidly developing alternatives with cross-disciplinary interest and participation. PR techniques are often an important component of intelligent systems in the areas of data pre-processing and decision-making that concern the description or classification of measurements. PR is not comprised of just one approach, but rather is a broad body of often loosely related knowledge and techniques.

1.2. Praxis' Objectives

The objectives of this praxis are:

- (1) to describe classification problems,
- (2) to present a survey of classification algorithms,
- (3) to discuss Glover's new linear programming-based discriminant approaches,
and
- (4) to provide an empirical comparison of selected methods.

In Chapter 1, we describe the classification problems by comparing and contrasting them with several other problems. We also present a detailed survey of popular classification algorithms from the fields of classical statistics, pattern recognition, neural networks, and mathematical programming.

In Chapter 2, we describe in detail the various linear programming formulations proposed by Freed and Glover. In this chapter, we pay special attention to Glover's improved hybrid formulations along with a new approach to improving their LP-based solutions.

In Chapter 3, we discuss the selection of methods and the experimental design, then summarize our findings. We complete the praxis with Chapter 4 presenting the final results and conclusions.

1.3. Notation

This section contains the notational conventions used in this study. The scalars and simple variables are denoted by italicized letters with the uppercase letters reserved for real scalar numbers for easy identification. While matrices are denoted by uppercase letters in boldface type, vectors are denoted by lowercase letters in boldface type. For easy identification and association of data across various disciplines (statistics, operations research, etc.), the matrices and vectors representing data for multiple groups from the same population are subscripted. For example, \mathbf{X}_1 and \mathbf{X}_2 are matrices representing observations from two groups 1 and 2, while \mathbf{h}_1 and \mathbf{h}_2 are vectors representing some other data from the same two groups. This notation should not be confused with any particular column or a row in a matrix or with a particular element in a vector as traditionally used in most textbooks for linear algebra.

The symbol $\mathbf{1}$ denotes a column vector of all ones, while the symbol $\mathbf{0}$ denotes a column vector of all zeros. The product of vectors and matrices is to be performed in a

manner that is conformable for multiplication, that is, a vector is treated as a row when it appears on the left of a matrix and, as a column when it appears on the right. Other mathematical notation employed in this report is standard.

1.4. Background

1.4.1. Classification Analysis

The exercise of comparing and grouping is a fundamental step in the organization of information about one's environment. Even before the use of computers became common, statisticians developed simple methods of objective classification based on standard probability theory. One such theory is that of *classification analysis* which addresses itself to the problem of assigning an object to one of a number of possible groups on the basis of observations made on the object. In classification analysis the existence and structure of the groups themselves are of secondary importance. It is the assignment of new cases that concerns the analyst.

With the advent of computers, there has been an upsurge of interest in automated numerical methods of classification. As the classification problem has proved so important in many different fields of application, it has suffered indirectly from being re-solved many times. Each time a discipline has re-invented the subject of classification, it has introduced its own jargon, its own notation, and its own favorite methods. For example, classification analysis is known as discriminant analysis in statistics, pattern recognition or supervised learning in computer science, and part of decision theory in management science and operations research. The most recent and most important re-use of classification analysis is in the area of expert systems, which seek to capture the reasoning of an expert using artificial intelligence techniques.

Classification analysis addresses itself to the problem of assigning an object to one of a number of possible groups on the basis of observations made on the object. There are, however, two main statistical methods with which classification analysis is often confused. These are *cluster analysis* and *analysis of variance*. Whereas cluster analysis attempts to identify any possible tendency for data to clump together to form groups, classification analysis is only concerned with the problem of classifying new objects into existing groups. And to emphasize this point, classification analysis is not concerned with identifying any possible groupings that might be contained within a mass of data. Analysis of variance (ANOVA) postulates that a particular grouping exists within some data and deals with the statistical proof of that supposition. This is a situation that most often is confused with classification analysis. The reasoning used is that if one can successfully classify new cases to the hypothesized groups with a reasonable accuracy then this is evidence enough that the groups are more than a figment of imagination. While this reasoning is true to a certain extent, there are more efficient and accurate statistical techniques for testing hypotheses about the existence of differences between groups.

It is also relevant to distinguish between *classification* and *dissection*. In dissection, the data set comprises a single group of objects; the aim is to dissect this group into several "sectors" which have certain specified properties. For example, the houses in a town can be regarded as a collection of objects described by two variables specifying their geographical locations. It may be convenient to divide the town into compact postal districts which contain comparable numbers of houses. If the physical distance is regarded as a measure of the dissimilarity between two houses, houses in the same postal district should be fairly similar to one another. However, some of them could also be more similar to houses in other postal districts. In practice, however, this clear division of methods into cluster analysis, ANOVA, and classification analysis is less than clear-cut.

1.4.2. Discriminant Analysis

Discriminant analysis is defined as a *methodology* for classifying or assigning elements (or objects) to predetermined groups. Whereas the problem of classification is to find a way of assigning a new object, discriminant analysis constructs a classification rule—a well-defined procedure that can be described and applied without the need for any additional "subjective" judgments. In other words, classification deals with the assignment of an object to the class to which it is "closest", whereas discrimination analysis deals with the construction of separation rules for distinguishing between categories into which an object may be classified. According to Kendall (1966), a general definition of discrimination is given as follows:

We are given the information that there are k populations $\pi \leftarrow \pi_1, \dots, \pi_k$ ($k \geq 2$) whose parameters are known or to be estimated based on samples drawn from respective populations. Suppose a new observation (vector) is presented to us. We know that it has come from exactly one of those k populations but we do not know the identity of that population. The discrimination problem is to construct a good procedure by which we can assign the new observation to the correct population with a high probability of success.

It is usually quite easy to find rules to discriminate, or separate, the sample cases from each other. It is much harder to develop decision criteria that hold up on new cases. Even with completely "noisy" data and hundreds of sample cases, classes can usually be distinguished with little difficulty. However, these distinctions will usually not hold up on new cases.

The problems associated with applying traditional discriminant models to business data have been widely discussed in the literature. The violation of parametric assumptions such as multivariate normal distributions deserves special attention since real-world data usually do not conform to these basic conditions. In addition, most empirical data include qualitative or dummy variables that cannot be multivariate normal. Another difficulty with applying discriminant models is that an observation is assumed to belong to one of the

given samples or groups. For instance, when bankers construct credit-scoring models to predict loan defaults, it is not clear whether slow-paying accounts should be classified as good or bad. Obviously, the slowness of a customer in repaying the loan is critical to group classification.

1.4.2.1. Dichotomous Discriminant Models

In general, discriminant analysis problems are solved with two-group models for reasons of better understanding. Multiple-group models generally evolve as extensions of two-group models.

Consider the following scenario: A product manager at a food company involved in the launching of new cereals is faced with evaluating a product's likely *success* or *failure* of commercialization. Past research has identified certain market characteristics associated with the successful launch of some popular cereals. The product manager can evaluate the new cereal's characteristics as well as the existing market conditions and arrive at a subjective estimate of its success in the market place. Alternatively, the product manager can quantify the available information from past new cereal introductions, develop a formal model that successfully predicts product success/failure on historical data, and apply the model to the new cereal about to be launched. In this case, the model provides information on the new cereal's performance in the market place. Ultimately, this model can be part of a formal, statistical marketing decision support system to predict success or failure of a new cereal.

When valid and reliable quantitative data or observations of a phenomenon are available, and the outcome variable is binary or dichotomous (e.g. success and failure), the models are called *dichotomous discriminant models*. Traditional methods of discriminant analysis may be used to solve such classification models.

Other well-known instances in which decision makers arrive at probabilistic classification decisions of success or failure include decision models of the bond-rating process, credit-scoring models, models of new business failures, and models for predicting tender offer outcomes.

1.4.2.2. Linear Discriminants

Linear discriminants are the most common form of classifier, and are quite simple in structure. The name linear discriminant simply indicates that a linear combination of the evidence will be used to separate or discriminate among the groups and to select the group assignment for a new case. For a problem involving p variables (attributes), this means, geometrically, that the separating surface between the samples will be a $(p-1)$ -dimensional hyperplane. With three variables, a plane will be sufficient to separate the classes, while with only two variables a line will suffice. It is often helpful to visualize the form of a classifier graphically, which can be done easily in two dimensions. Figure 1.1 depicts the separation of two groups, G_1 and G_2 , with two variables x_1 and x_2 .

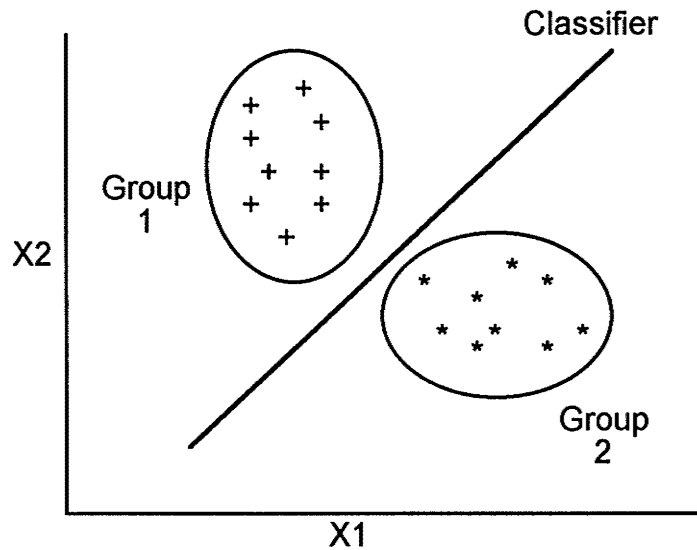


Figure 1.1. A two-group classifier.

In most situations, groups overlap and therefore cannot be completely separated by a plane (or line in two dimensions). This is true for any classifier where the error rate for the best possible classifier is greater than zero. When we are trying to find a solution that minimizes errors on new cases, simple structures such as linear combinations of variables tend to hold up well. Moreover, more than one plane or line can be used to separate two groups, and any separating curve can be approximated by multiple linear segments. Separating surfaces made up of multiple planes or lines are called *piecewise linear discriminants* because each plane or line is used to partially separate one class from another. The general form for any linear classifier function is given in equation (1.1) where (x_1, x_2, \dots, x_p) is the vector of variables, p is the number of variables, and w_i and c are constants that must be estimated.

$$f(x) = (w_1 x_1 + w_2 x_2 + \dots + w_p x_p) - c \quad (1.1)$$

In discriminating several groups from one another, we can define a separate linear discriminator for each group, and then combine them to define the separating surfaces. In the two-group (dichotomous or binary) discrimination problem, a single $(p-1)$ dimensional hyperplane will serve to separate them. For several groups we must either combine discriminants for each pair of classes or modify the decision problem so that it is posed as a sequence of dichotomous decisions of each group versus all the other groups.

A linear discriminant simply implements a weighted sum of the values of the observations. Thus, intuitively, we can think of the linear discriminant as a scoring function that adds to or subtracts from each variable, possibly weighting some variables more than others and yielding a total final score. The group selected, G_k , is the one with the highest score.

1.4.3. Classification Errors

The objective of learning classifications from sample data is to classify and predict successfully on new data. The most commonly used measure of success or failure is a classifier's error rate. Each time a classifier is presented with a case, it makes a decision about the appropriate group for a case. Sometimes it is right, sometimes it is wrong. We use two different terms *true error rate* and *apparent error rate* for quantifying classification errors. The true error rate is statistically defined as the error rate of the classifier for a very large number of new cases that converge in the limit to the actual population distribution. The apparent error rate of a classifier is the error rate of the classifier on the sample cases that were used to design or build the classifier.

$$\text{Apparent error rate} = \frac{\text{Number of errors}}{\text{Number of cases}} \quad (1.2)$$

In general, we attempt to extrapolate performance from a finite sample of cases. The apparent error rate is the obvious starting point in estimating the performance of a classifier on new cases. Unfortunately, in the real world, we usually have relatively modest sample sizes with which to design a classifier and extrapolate its performance on new cases. For most types of classifiers, the apparent error rate is a poor estimator of future performance. Fortunately, effective techniques exist for providing better estimates of the true error rate.

An error is simply a misclassification—the classifier is presented a case, and it classifies the case incorrectly. If all errors are of equal importance, an apparent error rate as calculated in equation (1.2) summarizes the overall performance of a classifier. However, for many applications, distinctions among different types of errors can be very important. In our credit-approval example, the error committed by the banker in approving an applicant as credit-worthy when the person indeed is not credit-worthy will be far more costly than the opposite type of error—of denying an applicant credit when the applicant is in fact credit-worthy.

A *confusion matrix* can be used to lay out the different types of errors. With just two groups, the choices are structured to predict the occurrence or non-occurrence of a single event or hypothesis. In our credit-approval example, when the decision is made about the applicant's credit rating, we find that four possibilities exist.

- (1) The applicant is credit-worthy, and the bank determines that he is credit-worthy; hence, the correct decision has been made.
- (2) The applicant is credit-worthy, but the bank determines otherwise; hence, an error has been made.
- (3) The applicant is not credit-worthy, and the bank determines that he is not credit-worthy; hence, the correct decision has been made.

- (4) The applicant is not credit-worthy, but the bank determines that he is credit-worthy; hence, an error has been made.

In cases (1) and (3), the bank reaches the correct decision; in cases (2) and (4), it makes an error. In such a situation, the two possible errors are frequently called *Type I error* or *Type II error*. Using the language of hypothesis testing, we denote the null hypothesis, H_0 , as: the applicant is credit-worthy. This means that there exists an alternative hypothesis, H_1 , that the applicant is not credit-worthy. If credit is extended when the applicant is not credit-worthy, then we say a Type II error is committed, as in case (4), and if credit is denied to a credit-worthy person, then a Type I error is committed as in case (2). Table 1.1 summarizes the relationship between the actions and the groups.

Table 1.1.--Confusion matrix - Type I & II errors

Action	Applicant is credit-worthy (H_0 is true) Group Positive	Applicant is not credit-worthy (H_0 is false) Group Negative
Accept H_0 (Positive)	Correct decision	Type II error
Reject H_0 (Negative)	Type I error	Correct decision

Any confusion matrix will have k^2 entries, where k is the number of groups. On the diagonal will lie the correct classifications, with the off-diagonal entries containing the various cross-classification errors. An error rate can also be presented as a misclassification cost for better understanding, where a misclassification cost is simply a number that is assigned as a penalty for making a mistake. The total cost of misclassification, thus, is most directly computed as the sum of the costs for each error.

Formally, for any confusion matrix, if E_{ij} is the number of errors entered in the confusion matrix and C_{ij} is the cost for that type misclassification, the total cost of misclassification is given in the following equation

$$Cost = \sum_{i=1}^k \sum_{j=1}^k E_{ij} C_{ij} \quad (1.3)$$

1.4.3.1. Training and Testing Sets

In discriminant analysis, quantitatively, we are given a data set with a set of predictive variables and the cases with their correct classifications. This data is assumed to be a random sample from some large population, and the task is to classify new cases correctly. As we just noted, the performance of a classifier is measured by its error rate. If unlimited cases for training and testing are available, the apparent error rate is the true error rate. This raises the question of how many cases are needed for one to be confident that the apparent error rate is effectively the true error rate? Stated differently, given a random sample drawn from a population, how many cases must be in the sample to guarantee that the error rate on new cases will be approximately the same?

For a real problem, one is given a sample from a single population, and the task is to estimate the true error rate for that population. The technique adopted in such cases is that instead of using all the cases to estimate the true error rate, the cases can be partitioned into two groups, some used for designing the classifier and some for testing the classifier. One group is called the *training set* and the other the *testing set*. The training set is used to design the classifier, and the testing set is used strictly for testing. If we "hold out" the test cases and only look at them after the classifier design is completed, then we can compute the error rate on new cases as described before. The error rate of the classifier on the test cases is called the *test sample error rate*.

1.4.3.2. Cross-Validation

For a single application of the *train-and-test* method, also known as the *holdout* method, a fixed percentage of cases is used for training and the remainder for testing. Such a single random partition can produce misleading error rates for small or moderately-sized samples. However, a series of train-and-test experiments can be conducted by randomly generating multiple train-and-test partitions which is also known as *resampling*. When resampling is performed, a new classifier is learned or constructed from each training sample. The estimated error rate is computed as the average of the error rates for classifiers derived from the random multiple resampling, otherwise known as *cross-validation*.

A special case of resampling or cross-validation is known as *leaving-one-out* where for a sample size n , a classifier is generated using $(n-1)$ cases and tested on the single remaining case. This is repeated n times, each time designing a classifier by leaving-one-out. Thus, each case in the sample is used as a test case and the error rate is the number of errors on the single test cases divided by n . While this is a perfect technique for computing an unbiased error rate, with large sample sizes the technique may be computationally expensive.

As the sample size grows, the cross-validation test partitions can be constructed to hold more than one case while maintaining the accuracy in estimating the error rates. In k -fold cross validation, the cases are randomly divided into k mutually exclusive test partitions of approximately equal size. The cases not found in each test partition are independently used for training, and the resulting classifier is tested on the corresponding test partition. The average error rates over all k partitions is the cross-validated error rate. Table 1.2 summarizes the techniques of error estimation for a sample of n cases.

Table 1.2.--Cross-validation estimators

	Leaving-one-out	k -fold
Training cases	$n-1$	$n-(n/k)$
Testing cases	1	n/k
Iterations	n	k

1.5. Types of Linear Discriminant Analysis Methods

Discriminant analysis methods have evolved largely from statistics with their beginnings in 1935. There are many statistical methods available for inductively determining classification rules. Prominent among them are linear discriminant analysis methods. The field of statistics; however, is no longer the territorial champion of discriminant analysis methods. Recent research in the fields of mathematical programming, artificial intelligence, and machine learning has yielded a plethora of new and advanced approaches.

Whatever may be the field, the basis for formulating classification rules follows a common approach. In two-group linear discriminant analysis, one determines a p -dimensional discriminant non-zero column vector \mathbf{w} and a scalar c such that if $\mathbf{w}\mathbf{x} \leq c$ then observation \mathbf{x} is classified as belonging to group 1. Otherwise \mathbf{x} is classified as belonging to group 2. The (\mathbf{w}, c) pair is estimated from a training set consisting of a sample of observations, each of whose group membership is known. The (\mathbf{w}, c) is called a *discriminant function* such that the hyperplane $\mathbf{w}\mathbf{x} = c$ partitions the p -dimensional Euclidean space R^p into a closed half-space $\mathbf{w}\mathbf{x} \leq c$ and an open half space $\mathbf{w}\mathbf{x} > c$.

A typical application of discriminant analysis is credit scoring. A training set consisting of a history of credit approvals and their subsequent outcomes (payment or default) is used to determine a discriminant function (\mathbf{w}, c) . The information gathered on the new applicant is used to construct a vector of numerical attributes \mathbf{x} . Using the known discriminant function (\mathbf{w}, c) , the applicant is granted credit if $\mathbf{w}\mathbf{x} \leq c$ and denied credit if $\mathbf{w}\mathbf{x} > c$.

Many different methods have been developed to determine (\mathbf{w}, c) from a training set. These methods are predominantly based on statistical, search, and mathematical-programming approaches. While the statistical methods offer probabilistic statements about the results and depend on assumptions that are often inappropriate, the search and the mathematical programming-based methods are non-parametric and have yielded excellent results. For this study, we categorize the methods as follows:

- (1) statistical (parametric) methods,
- (2) search methods, and
- (3) mathematical programming methods.

1.5.1. Statistical Methods

Over the last five decades, numerous methods have been proposed in the statistics literature for discriminant analysis. Some of the prominent and widely used methods are:

- (1) Bayes' Rule,
- (2) Fisher's Linear Discriminant Method (LDM),
- (3) Smith's Quadratic Discriminant Method (QDM), and the
- (4) Logistic Discriminant Method (LGM).

1.5.1.1. Bayes' Rule

This is the simplest of all the classification rules. Based on conditional probabilities, Bayes' rule assigns the object to the group with the highest conditional probability. Let \mathbf{x} be a vector of measured variables and G_k be the k^{th} group, then Bayes' rule is to assign the object to group k where:

$$P(G_k|\mathbf{x}) > P(G_j|\mathbf{x}) \quad \text{for all } j \neq k \quad (1.4)$$

If by any chance there is more than one group with the largest conditional probability, then the tie can be broken by allocating the object at random to one of the tied groups. The success of Bayes' rule is in that all of the information about possible group membership is contained in the set of conditional probabilities. Of course, things are not so simple in practice, especially when it comes to finding the all-important conditional probabilities. Quantities such as $P(G_k|\mathbf{x})$ are very difficult to find by standard methods of estimation; however, this is not the case for quantities such as $P(\mathbf{x}|G_k)$. Fortunately, there is a connection between the two quantities, popularly known as *Bayes' Theorem*. In equation form, Bayes' theorem is stated as follows, where all of the items on the right-hand side of the equation can be found by sampling.

$$P(G_k|\mathbf{x}) = \frac{P(\mathbf{x}|G_k)P(G_k)}{\sum_j P(\mathbf{x}|G_j)P(G_j)} \quad (1.5)$$

Putting Bayes' theorem into Bayes' rule gives:

Assign the new case to group k if

$$\frac{P(\mathbf{x}|G_k)P(G_k)}{\sum_i P(\mathbf{x}|G_i)P(G_i)} > \frac{P(\mathbf{x}|G_j)P(G_j)}{\sum_i P(\mathbf{x}|G_i)P(G_i)} \text{ for all } j \neq k \quad (1.6)$$

or, simplifying the terms, the rule becomes:

Assign the new case to group k if:

$$\frac{P(\mathbf{x}|G_k)}{P(\mathbf{x}|G_j)} > \frac{P(G_j)}{P(G_k)} \text{ for all } j \neq k \quad (1.7)$$

which is the final practical form of Bayes' rule.

The main problem with Bayes' rule is that while it solves our problem completely, it is almost unusable. The reason for this is the sheer volume of data that has to be collected to estimate $P(\mathbf{x}|G_k)$. For example, suppose we are to make measurements on five variables each consisting of ten possible categories. This would require, just for one group, estimating fifty relative frequencies with sample sizes in excess of 500. This sampling would have to be repeated and tabulated for each group.

However impractical it may be, it is not possible to ignore Bayes' rule. A large part of discriminant analysis is concerned with finding practical forms of Bayes' rule that are appropriate under special conditions or finding simple approximations that have acceptably low error rates.

1.5.1.2. Fisher's Linear Discriminant Method (LDM)

Linear discriminant analysis as originally developed by Fisher in 1935, has become the foundation for modern statistical procedures for classifying a subject based on certain attributes into one of two or more mutually exclusive and collectively exhaustive groups. While few researchers or practitioners dispute the general usefulness of this technique, the theoretical base of the method must be examined carefully before each application. The optimality of the LDM can be demonstrated when observations are random samples that have been drawn independently from respective multivariate normal populations.

Let \mathbf{x} be a bivariate vector of k measured variables. Let $P(G_1)$ and $P(G_2)$ be the prior probabilities of the groups G_1 and G_2 . In the case of our cereal example introduced earlier, group G_1 may consist of a set of previously introduced cereals that were successful product innovations and group G_2 may consist of the cereals that were market failures. The p measured variables of the vector \mathbf{x} may represent the information such as the cereal and market characteristics.

Let $f_k(\mathbf{x})$ be the multivariate probability density function of \mathbf{x} from G_k where $k = 1, 2$. Given the assumption that the misclassification costs are equal, the optimal misclassification rule (i.e., Bayes' rule assignment that minimizes the total probability of misclassification) assigns \mathbf{x} to G_1 if:

$$\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{P(G_2)}{P(G_1)} \quad (1.8)$$

and to group G_2 otherwise.

Let us assume that the two populations have mean vectors μ_1 and μ_2 and have the dispersion (variance-covariance) matrix S . Let us assume $P(G_1) = P(G_2) = 1/2$. If μ_1 , μ_2 , and S are replaced by their corresponding maximum likelihood sample estimators \bar{x}_1 , \bar{x}_2 and \bar{S} , then the following rule may be used to make classificatory decisions.

Assign x to group G_1 if

$$x \bar{S}^{-1} (\bar{x}_1 - \bar{x}_2) \geq \frac{1}{2} (\bar{x}_1 - \bar{x}_2)' \bar{S}^{-1} (\bar{x}_1 - \bar{x}_2) \quad (1.9)$$

and to group G_2 otherwise.

This rule is Fisher's well-known linear discrimination function (LDF). We note that the LDF minimizes the total probability of misclassification under the assumption of multivariate, normally distributed, measured variables with known means, and known and equal variance-covariance matrices.

Many researchers have expressed concern about employing Fisher's technique with data that are not normal. When the assumption of multivariate normality is not satisfied, there are two common methods for resolving the problem:

- (1) Improve the distributional properties of the data by various transformations (e.g. log, square root, arcsin),
- (2) Utilize some of the nonparametric techniques such as log-linear, rank discriminant, or mathematical programming.

1.5.1.3. Smith's Quadratic Discriminant Method (QDM)

If the assumption of multivariate normality is satisfied, but the variance-covariance matrices are unequal, then a quadratic form of the LDF minimizes the total probability of misclassification. This rule assigns \mathbf{x} to group G_1 if

$$(\mathbf{x} - \bar{\mathbf{x}}_2)' \bar{\mathbf{S}}_2^{-1} (\mathbf{x} - \bar{\mathbf{x}}_2) - (\mathbf{x} - \bar{\mathbf{x}}_1)' \bar{\mathbf{S}}_1^{-1} (\mathbf{x} - \bar{\mathbf{x}}_1) \geq \ln|\bar{\mathbf{S}}_1| - \ln|\bar{\mathbf{S}}_2| \quad (1.10)$$

and to group G_2 otherwise. $\bar{\mathbf{S}}_1$ and $\bar{\mathbf{S}}_2$ are the unpooled estimates of the population dispersion (variance-covariance) matrices.

1.5.1.4. Logistic Discriminant Method (LGD)

Both the LDM and QDM rules require specification of two probability density functions. The logistic model takes us away from an estimator of $P(\mathbf{x}|G_k)$ in favor of an estimate of the likelihood ratio $P(\mathbf{x}|G_1)/P(\mathbf{x}|G_2)$. Also called a Logit model, LGD is appropriate for any situation when the log of the likelihood ratio can be assumed to be linear. If one is interested in modeling the ratio directly, the following logistic discriminant rule is obtained.

$$\log \left\{ \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \right\} = \beta_0 + \beta' \mathbf{x} \quad (1.11)$$

where β_0 and β are the unknown parameters of the logistic model that must be estimated from the data. According to (1.11) a case is assigned to group G_1 if $\beta_0 + \beta' \mathbf{x} > 0$ and to group G_2 otherwise. The procedure calculates the values of the β coefficients that maximize the likelihood function. One advantage of the logit model is that a unique maximum can always be found.

With this rule, given a vector of measured variables \mathbf{x} , one essentially relates the probability of a case belonging to group G_1 and to the vector \mathbf{x} itself, through the functional form of a logistic cumulative-density function. This approach has many advantages. Because there is no need to specify f , it is generally applicable even if the measured variables are multivariate, non-normally distributed.

1.5.1.5. Problems with Statistical Methods

The problems encountered in statistical methods can be of several different types, depending on the models solved. The methods may produce unreliable results due to a combinations of the following factors:

- (1) the distribution of the variables,
- (2) the group dispersions,
- (3) the interpretation of the significance of individual variables,
- (4) the reduction of dimensionality,
- (5) the definitions of the groups,
- (6) the choice of the appropriate a priori probabilities and/or costs of misclassification, and
- (7) the estimation of classification error rates.

Furthermore, with most of these methods there are no easy ways to measure solution sensitivities or to weight individual observations. These concerns can, however, be easily handled with recent methods using linear programming approaches which will be discussed in detail in Chapter 2.

1.5.2. Search Models in Discriminant Analysis

Attributes or the variables that define the characteristic of an observation play a major role in the analysis and interpretation of the solution. Poorly defined attributes or non-contributing attributes can unnecessarily increase the complexity of the problem. Therefore, reducing the number of attributes used in the construction of discriminant analysis function without sacrificing classification performance is a much desired goal in discriminant analysis. Recently, a number of heuristic methods have been proposed which are non-parametric and provide better control over the attributes. These methods can yield good, and sometimes optimal solutions although they are incapable of proving optimality. Heuristic procedures are based on ideas that trace their origins equally to the fields of operations research and artificial intelligence.

There are five heuristic methods that are considered to be related to the field of artificial intelligence and are used as frameworks for approaching difficult optimization problems. These methods are:

- (1) Neural networks: an associative memory process that has claimed successes in pattern-recognition applications, but has generally shown less than impressive results on optimization problems.
- (2) Genetic algorithms: another memoryless process that relates to phenomena in the biological sciences. These methods have been applied with some success to optimization problems, image registration problems, machine learning, and a large variety of problems in the field of operations research.
- (3) Nearest neighbor search: a completely nonparametric, direct table lookup procedure. This method can produce any arbitrarily complex surface to separate the classes based only on the configuration of the sample points and their metric.

- (4) Simulated annealing: a memoryless process that draws on phenomena in the physical sciences (i.e. metallurgy) and has been shown to be effective for certain applications.
- (5) Tabu search: a high-level method, or meta-strategy, useful for solving optimization problems. Tabu search employs memory functions to carry out its strategic operations and typically out-performs simulated annealing and genetic algorithm approaches.

In the following sections, we discuss the neural networks, the nearest neighbor, and the genetic search approaches in detail.

1.5.2.1. Neural Network Models

Neural networks, which attempt to replicate the learning processes of systems of neurons, are the focus of much of the current classification research. Given a basic structure, the internal parameters of such a model are determined automatically through training with examples from known groups. Although this training requires a great deal of computation, the classification of new observations is rapid and highly accurate in some cases, due to robustness and the model's ability to deal with "noisy" data. Successful applications include credit analysis, character and handwriting recognition, and image processing.

Research in neural network modeling attempts to provide insights into the processes by which intelligent beings perform recognition, memorization, and learning. A neural network consists of a large number of processing units communicating with each other in an asynchronous manner. Because of the embedded parallelism, in both its architecture and communication pattern, the study of a neural network model is usually encompassed in the larger framework of parallel distributed processing models. Used as a modeling tool, neural networks offer a number of distinct features. First, knowledge

stored in a network is distributed as a pattern of connections among its processors. Such a representation can tolerate individual-processor breakdown without impairing the performance of the entire network. Second, models for classification and recognition tasks can be constructed by training a neural network with examples of the target concept, an example-driven learning process. Most important, this learning process can be conducted incrementally with new examples.

A neural network model consists of a number of processing units interconnected with each other in the network. Each unit is a simple computation device that receives input signals from other units, aggregates the signals based on an input function, and generates an output signal based on an input function. The output signal is then routed to other units as directed by the topology of the network.

Interestingly, many neural networks turn out to be variations of (piecewise) linear classifiers. However, unlike the statistical classifiers, most neural network classifiers are nonparametric. The simplest neural network device is the single-output *perceptron*, which we will briefly examine next. More complex neural networks can be described as combinations of many perceptrons in a network.

The simplest perceptron is a device that decides whether an input pattern belongs to one of two classes. Although the perceptron has a representation that is readily implemented in hardware, it is strictly the equivalent of the linear discriminant function as described before.

$$f(x) = (w_1x_1 + w_2x_2 + \dots + w_px_p) - c$$

The weights, w , can assume real values, both positive and negative, so we can rewrite the above equation as :

$$f(\mathbf{x}) = \mathbf{w}\mathbf{x} + \Theta \quad (1.12)$$

where the variables are described as inputs \mathbf{x} , and Θ is the constant. Geometrically, in two dimensions, the constant Θ indicates the intersection of the line with the y-axis. The inputs are multiplied by weights and summed with a constant. The perceptron produces an output indicating membership in group 1 or group 0. When the sum is greater than 0, the output is 1, and the group 1 is selected; otherwise, the output is 0, and the group 0 is selected. The output is produced according to the following equation:

$$\begin{aligned} \text{Output} &= 1 \text{ if } (\mathbf{w}\mathbf{x} + \Theta) > 0 \\ &= 0 \text{ otherwise} \end{aligned} \quad (1.13)$$

Here, the constant Θ is referred to as the threshold or bias. Geometrically, the perceptron selects group 1 for points above the line $\mathbf{w}\mathbf{x} + \Theta = 0$ and group 0 for points below the line in a two dimensional case.

1.5.2.2. Genetic Search Model

In general, a *genetic algorithm* (GA) starts with a number of initial possible solutions. These constitute the *initial population*. A second population is determined from the first population by survival of members and through mating and random mutations. Selection of surviving members is determined randomly according to member

fitness. In optimization problems, fitness is determined from the objective function. The higher the member fitness, the more likely is its selection. Successive populations are determined in a similar manner.

In the formulation of GA problems, the decision variables are represented as a bit stream of length L . The set of allowable bit streams is given by S . A population P is an ordered subset of S containing K elements and P as $P = \{s_1, \dots, s_k\} \subseteq S$. Starting with P , a second population is formed using a sequence of different procedures.

A discriminant model is presented below that minimizes the number of misclassifications and the number of used attributes. Let $nz(\mathbf{w})$ be the number of nonzero components of \mathbf{w} . Also, let $m(\mathbf{w})$ be the minimal number of misclassifications for a given \mathbf{w} over all possible values of c . The discriminant problem can be stated simply as:

$$\text{Minimize } m(\mathbf{w}) + \frac{nz(\mathbf{w})}{(p+1)} \quad (1.14)$$

Here, we encode each \mathbf{w} element by a bit stream of length b . Then the vector \mathbf{w} can be represented by a bit stream of length $L = bp$. Using the property that if \mathbf{w} solves the above problem then $\lambda\mathbf{w}$ also solves the respective problem for any $\lambda > 0$, we can restrict our search to $C = \{\mathbf{w}: -1 \leq \mathbf{w} \leq 1\}$.

The genetic algorithms are computationally efficient and produce results comparable to other heuristics. As with other heuristics, there is no formal termination rule, hence the amount of computational effort expended is determined by the number of populations the user chooses to inspect.

1.5.2.3. Nearest Neighbor Classifier

The nearest neighbor classifier is a member of a family of classifiers called k -nearest neighbor classifier (k - NN). Instead of finding the single nearest neighbor, the k -nearest neighbors are found, where k is some constant. The decision rule is taken to choose the group that appears most frequently among the k -neighbors. An odd number of neighbors are used so that ties will not occur.

The method requires that the distance between a new case and every entry in the sample table be compared. Typically the distance is compared attribute by attribute, and then summed. Computationally, the nearest neighbor method involves no effort in learning from the samples. The trade-off is that the computation for predicting the classification of a new case is relatively large. The new case must be compared with every case in the sample. This division of computation between learning and classification is the reverse from what happens with all other classification methods. For other methods, learning can be quite expensive, but classification and prediction on a new case typically involve a trivial matching step. The process of finding the nearest neighbor can be speeded up dramatically by sorting and storing the samples in a specialized (k - d) tree format. It can reduce the number of samples required from n to only $\log n$. Thus if one considers such presorting to be part of the learning, the nearest neighbor method can be seen to behave computationally like the other methods.

1.5.3. Mathematical Programming Models in Discriminant Analysis

As described earlier, any two-group linear discriminant function including that of Fisher's can be expressed in the form of (\mathbf{w}, c) , where an object, \mathbf{x} , is classified as belonging to group 1 if $\mathbf{w}\mathbf{x} \leq c$ and to group 2 if $\mathbf{w}\mathbf{x} > c$. Points on the hyperplane $\mathbf{w}\mathbf{x} = c$ are grouped into either group 1 or 2, as specified in advance. We can safely assume that the

points on the hyperplane belong to group 1. With this assumption, (\mathbf{w}, c) can properly be thought of as a linear discriminant function.

As we have noted before, in all of the cases the linear discriminant classifier is determined from a set of observations whose group membership is known. This set of data is called the training set. We will use the following additional notation in our discussion on the mathematical programming formulations.

p = number of attributes (variables);

n_j = number of observations in group j , $j = 1, 2$, where $n_j > 0$;

$n = n_1 + n_2$ (the total number of observations in the training set); and

$\mathbf{X}_j = (n_j \times p)$ matrix for group j , $j = 1, 2$.

Mathematical programming approaches to linear discriminant analysis consider the following problem: Determine a scalar c and a non-zero vector $\mathbf{w} \in R^p$ such that the hyperplane $\mathbf{w}\mathbf{x} = c$ partitions the p -dimensional Euclidean space R^p into a closed half-space $\mathbf{w}\mathbf{x} \leq c$ and an open half space $\mathbf{w}\mathbf{x} > c$.

Ideally the separating hyperplane $\mathbf{w}\mathbf{x} = c$ should satisfy $\mathbf{X}_1\mathbf{w} \leq c\mathbf{1}$ for group 1 observations, and $\mathbf{X}_2\mathbf{w} > c\mathbf{1}$ for group 2 observations. Since such an hyperplane usually does not exist, mathematical programming (MP) techniques try to determine a hyperplane that optimizes a certain criterion function. Most mathematical programming formulations are based on a set of well-defined constraints and objective functions. In their general form, mathematical programming methods can be stated as follows.

$$\text{Minimize } f(\mathbf{w}, c) \quad (1.15)$$

$$\text{subject to: } \mathbf{X}_1 \mathbf{w} \leq c \mathbf{1} \quad (1.16)$$

$$\mathbf{X}_2 \mathbf{w} > c \mathbf{1} \quad (1.17)$$

$$\mathbf{w} \neq \mathbf{0} \quad (1.18)$$

Since MP algorithms require the constraint space to be closed, constraint (1.17) is usually replaced by

$$\mathbf{X}_2 \mathbf{w} \geq c \mathbf{1} \quad (1.19)$$

or

$$\mathbf{X}_2 \mathbf{w} \geq (c + \varepsilon) \mathbf{1} \quad (1.20)$$

where ε is an arbitrarily small positive number. We note that while constraint (1.19) is a relaxation of (1.17), constraint (1.20) is a restriction of (1.17). Therefore, if the value of ε is not chosen properly, an optimal solution may be eliminated from consideration. Furthermore, constraint (1.20) may introduce a classification gap, while (1.19) may result in a point being classified as belonging to both groups.

The earliest mathematical programming approach to linear discriminant analysis is due to Mangasarian (1965). Since then two major areas of research have developed in this field. The first area focuses on extensions and improved formulations with the following key objectives:

- (1) to avoid potential misspecifications of the problem,
- (2) to investigate issues such as the stability of the problem solution with respect to transformations and rotations, and
- (3) to analyze the occurrence of the unacceptable and trivial solutions.

The second area investigates the classificatory performance of the various MP-based techniques relative to parametric statistical procedures such as LCF and QCF. We will need the following definitions to understand the various MP formulations:

internal deviation: the deviation from the hyperplane of a point properly classified;

external deviation: the deviation from the hyperplane of a point *improperly* classified;

\mathbf{i}_j : non-negative n_j -vector of internal deviations for group j , $j = 1, 2$;

i : minimum internal deviation;

\mathbf{e}_j : non-negative n_j -vector of external deviations for group j , $j = 1, 2$;

e : maximum external deviation;

\mathbf{d}_j : n_j -vector of deviations (internal and external) for group j , $j = 1, 2$.

Incorporating \mathbf{i}_j and \mathbf{e}_j into constraints (1.16) and (1.19), these constraints are rewritten as

$$\mathbf{X}_1 \mathbf{w} + \mathbf{i}_1 - \mathbf{e}_1 = \mathbf{c} \mathbf{1} \quad (1.21)$$

$$\mathbf{X}_2 \mathbf{w} - \mathbf{i}_2 + \mathbf{e}_2 = \mathbf{c} \mathbf{1} \quad (1.22)$$

Over the past ten years, several MP-based models have been proposed to analyze discriminant analysis problems. Among these, the most commonly used models are:

- (1) MSD: minimize the sum of distances;
- (2) MMD: minimize the maximum distance;
- (3) MIP: minimize the number of misclassifications using mixed integer programming; and
- (4) NLP: minimize a generalized non-linear distance measure.

1.5.3.1. Minimize the Sum of Distances Models (MSD)

In general, these models try to find a hyperplane that minimizes the weighted sum of external deviations. One of the earliest MSD models was given by Hand (1981) as described below:

$$\text{Minimize} \quad z = \mathbf{1e}_1 + \mathbf{1e}_2 \quad (1.23)$$

$$\text{subject to:} \quad \mathbf{X}_1\mathbf{w} + c\mathbf{1} - \mathbf{e}_1 \leq -\mathbf{b}_1 \quad (1.24)$$

$$\mathbf{X}_2\mathbf{w} + c\mathbf{1} + \mathbf{e}_2 \geq -\mathbf{b}_2 \quad (1.25)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (1.26)$$

$$\mathbf{e}_1, \mathbf{e}_2 \geq \mathbf{0} \quad (1.27)$$

$$\mathbf{b}_1, \mathbf{b}_2 \geq \mathbf{0} \quad (1.28)$$

In this model the components of the \mathbf{b}_1 and \mathbf{b}_2 vectors are used for strict separation and are usually set equal to the same value. The objective of MSD is to minimize the sum of external deviations. MSD is a rather stable model in that it rarely gives a trivial solution. The strict separation of the two groups may result in a solution with classification gaps.

1.5.3.2. Minimize the Maximum Deviations Models (MMD)

Introduced by Freed and Glover (1981a, 1981b), these models try to find a hyperplane that minimizes the maximum external deviation. The general model is formulated as:

$$\text{Minimize } d \quad (1.29)$$

$$\text{subject to: } \mathbf{X}_1 \mathbf{w} + d \mathbf{1} \leq c \mathbf{1} \quad (1.30)$$

$$\mathbf{X}_2 \mathbf{w} - d \mathbf{1} \geq c \mathbf{1} \quad (1.31)$$

$$\mathbf{w}, d \text{ unrestricted in sign} \quad (1.32)$$

$$c \text{ is a positive constant} \quad (1.33)$$

As seen from the formulation, c is a pre-chosen boundary value, and the objective of MMD is to maximize the minimum deviation of any group member's score from the break point c . Since d is allowed to take negative values and the objective is to maximize d , MMD can be interpreted as the minimization of the maximum external deviation.

An important property of MMD is that an unbounded solution indicates perfect separation of the two groups. A potential problem with this model is that it can yield a trivial (all zero) solution. However, recent extensions present sufficient conditions to prevent a trivial solution as well as necessary and sufficient conditions for a bounded, nontrivial solution.

1.5.3.3. Mixed-Integer Programming Models (MIP)

In general, mixed-integer programming models try to find a separating hyperplane that minimizes the number of misclassifications. MIP models have usually been solved by a general-purpose commercial code like MPSX, which uses a branch-and-bound procedure. The following mixed integer programming model, which was proposed by Koehler and Erenguc (1990a, 1990b), seeks to minimize the number of misclassified cases.

$$\text{Minimize} \quad z = \mathbf{1}y_1 + \mathbf{1}y_2 \quad (1.34)$$

$$\text{subject to:} \quad \mathbf{X}_1\mathbf{w} \leq c\mathbf{1} + My_1 \quad (1.35)$$

$$\mathbf{X}_2\mathbf{w} \geq c\mathbf{1} - My_2 \quad (1.36)$$

$$\mathbf{w} \neq \mathbf{0} \quad (1.37)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (1.38)$$

$$y_1, y_2 \text{ are zero-one vectors} \quad (1.39)$$

$$M \text{ is a large positive number} \quad (1.40)$$

The model first checks using the MMD formulation to see if groups can be perfectly separated by a hyperplane. If perfect separation is not possible, then the $\mathbf{w} \neq \mathbf{0}$ constraint is replaced by:

$$\mathbf{X}_1 \mathbf{w} \leq c\mathbf{1} \quad (1.41)$$

$$\mathbf{X}_2 \mathbf{w} \geq c\mathbf{1} \quad (1.42)$$

If a set of data cannot be perfectly separated, then the above constraint is redundant. Koehler and Erenguc (1990a) show that when one directly minimizes the number of classifications, the above constraint prevents the trivial solution.

Bajgier and Hill (1982) provide a method that seeks to optimize a weighted sum of the following three objectives:

- (1) minimize the number of misclassified cases,
- (2) minimize the sum of external deviations, and
- (3) maximize the sum of internal deviations.

The following model as formulated by them minimizes the number of misclassifications.

$$\text{Minimize} \quad z = P_1[\mathbf{1y}_1 + \mathbf{1y}_2] + P_2[\mathbf{1e}_1 + \mathbf{1e}_2] - P_3[\mathbf{1i}_1 + \mathbf{1i}_2] \quad (1.43)$$

$$\text{subject to:} \quad \mathbf{X}_1 \mathbf{w} + \mathbf{i}_1 - \mathbf{e}_1 - c\mathbf{1} = \mathbf{0} \quad (1.44)$$

$$\mathbf{X}_2 \mathbf{w} - \mathbf{i}_2 + \mathbf{e}_2 - c\mathbf{1} = \mathbf{0} \quad (1.45)$$

$$\mathbf{e}_1 \leq M\mathbf{y}_1 \quad (1.46)$$

$$\mathbf{e}_2 \leq M\mathbf{y}_2 \quad (1.47)$$

$$\mathbf{i}_1, \mathbf{i}_2, \mathbf{e}_1, \mathbf{e}_2 \geq \mathbf{0} \quad (1.48)$$

$$\mathbf{w} \text{ unrestricted in sign} \quad (1.49)$$

where c is a positive constant; P_1, P_2, P_3 are positive weights; $\mathbf{y}_1, \mathbf{y}_2$ are zero-one vectors; and M is a large positive number. A closer look at this formulation will reveal that by systematically setting P_j to zero, different new objective functions can be developed easily.

1.5.3.4. Nonlinear Programming (NLP) Models

The nonlinear programming models aim at obtaining a separating hyperplane that minimizes a generalized distance measure.

$$\text{Minimize} \quad z = (\mathbf{1e}_1^r + \mathbf{1e}_2^r)^{1/r} \quad (1.50)$$

$$\text{subject to:} \quad \mathbf{X}_1\mathbf{w} + \mathbf{i}_1 - \mathbf{e}_1 \leq c\mathbf{1} \quad (1.51)$$

$$\mathbf{X}_2\mathbf{w} - \mathbf{i}_2 + \mathbf{e}_2 \geq c\mathbf{1} \quad (1.52)$$

$$\mathbf{w} \text{ unrestricted in sign} \quad (1.53)$$

$$\mathbf{i}_1, \mathbf{i}_2, \mathbf{e}_1, \mathbf{e}_2 \geq \mathbf{0} \quad (1.54)$$

where c is a non-zero constant.

For $1 < r < \infty$, the objective function of NLP is strictly convex. Therefore, if NLP has an optimal solution, it will always have a unique optimal solution. As can be seen, for $r = 1$, NLP yields an MSD model; and for $r = \infty$, it yields an MMD model.

These are just a few of the representative formulations of discriminant models emanating from the annals of operations research. As is obvious, the MP-based models are completely devoid of parametric assumptions and offer great flexibility in introducing heuristics. The MP-based solution approaches lend themselves well to experiments, such as selection (addition and deletion) of attributes and construction of different train-and-test cross-validation data sets.

CHAPTER 2

GLOVER'S NEW DISCRIMINANT ANALYSIS APPROACHES

2.1. Introduction

In a series of papers over the last ten years, several authors have presented increasingly sophisticated mathematical programming models to determine linear discriminant classifiers. However, the objectives used to determine the solution are only loose approximations to the objectives that are used to measure their effectiveness. Researchers have prepared a number of different objectives to more closely approximate the real objectives, thus leading to a large number of alternative models.

One of the first LP-based linear discriminant classification models proposed was due to Freed and Glover (1981a, 1981b) to maximize the minimal deviation (MMD) of an observation's score from a critical value. In its simplest form, the MMD model is formulated as below:

$$\text{Maximize } d \quad (2.1)$$

$$\text{subject to: } \mathbf{X}_1 \mathbf{w} \leq (c-d)\mathbf{1} \quad (2.2)$$

$$\mathbf{X}_2 \mathbf{w} \geq (c+d)\mathbf{1} \quad (2.3)$$

$$\mathbf{w}, d \text{ unrestricted in sign} \quad (2.4)$$

where c is a non-zero positive constant.

This initial paper led to a series of papers that presented a progression of alternative LP models. Subsequently, several researchers, including Glover, reported many unresolved and irksome problems with these models. It is no surprise that the bulk of the literature associated with mathematical programming-based methods for determining linear discriminant functions are easily grouped into two categories: (1) those methods that give empirical comparisons of the performance of one or more models versus parametric methods; and (2) those that point out problems in earlier MP-based models.

2.1.1. MP-based Discriminant Analysis--Problems and Issues

In order to better understand the rationale behind the structure of some of the newer discriminant LP models formulated by Freed and Glover, and other authors, it is important to understand some of the common problems and issues that have received the most attention. They are:

- (1) Unacceptable solutions. A system of equations of the form:

$$\mathbf{X}_1 \mathbf{w} \leq \mathbf{c}_1 \quad (2.5)$$

$$\mathbf{X}_2 \mathbf{w} \geq \mathbf{c}_2 \quad (2.6)$$

has a trivial solution of $(\mathbf{w} = \mathbf{0}, \mathbf{c} = \mathbf{0})$. This provides no discrimination, since every observation can be classified as belonging to both groups 1 and 2. Yet most LP formulations tend to favor this type of solution.

(2) Improper solutions. A problem closely related to unacceptable solutions occurs when there is a c and a non-zero w satisfying:

$$X_1 w = c1 \quad (2.7)$$

$$X_2 w = c1 \quad (2.8)$$

Although (w, c) is a "separating" hyperplane, it does not properly separate the two groups since both the groups may lie on the hyperplane. Again, most LP formulations tend to favor this type of solution, if one exists.

(3) Translation invariances. Some LP formulations provide a different w if the training set is translated. Since data translation is inherently arbitrary, many find this property disturbing. In general, one requires that if (w, c) is a solution to:

$$X_1 w \leq c1$$

$$X_2 w \geq c1$$

It should also be a solution to

$$(X_1 + t1)w \leq c'1 \quad (2.9)$$

$$(X_2 + t1)w \geq c'1 \quad (2.10)$$

where $c' = c + tw$ and t is the translation vector for each group. Some formulations, such as MMD, do not have this property. Similarly, if the data are transformed by a non-singular matrix, B , then the new solution should be equal to $B^{-1}w$. Some LP discriminant models do not have this property.

(4) Unbounded solutions. When a strict separation is possible, some models—particularly MMD where d is unrestricted in sign—yield unbounded solutions. This is undesirable from a practitioner's point of view, and is easily prevented by adding constraints such as

$$-M\mathbf{1} \leq \mathbf{w} \leq M\mathbf{1} \quad (2.11)$$

for $M > 0$, or with a normalization constraint (Glover, 1990).

(5) Unbalanced solutions. A good solution should have a balanced number of misclassifications across the two groups. Many discriminant procedures perform better than Fisher's method, but the gain in classification is usually disproportionately distributed across the two groups. When most of the misclassifications will occur in one group—as with some mixed-integer programming models that minimize the number of misclassifications—it is relatively straightforward to add a balancing constraint to these models.

(6) Number of attributes. A solution that uses fewer attributes may be preferred to one using more, even if the latter case produces slightly better classification results. Such considerations for parsimonious solutions are generally absent from LP formulations. In some statistical methods, variables are sequentially added (removed) one at a time to (from) a linear discriminant function based on some statistical measure of their contribution to the discrimination.

(7) Unfaithful methods. If two groups can be strictly separated, one expects the discriminant analysis procedure to determine a strictly separating hyperplane. If a method always finds a strict separator when one exists, then we say the method is faithful. Fisher's method is unfaithful. Some of the linear programming methods are unfaithful as well.

(8) Computational effort. Real-world discriminant problems typically have a large number of observations and a small number of attributes (variables). As such, many of the linear programming formulations typically have a large number of constraints and a small number of variables. The dual has the opposite properties and accordingly might be the preferred problem to solve. When one considers mixed-integer approaches, there are at least n zero-one integer variables, which can be computationally challenging, hence heuristic solution approaches are appealing.

Along with these problems, there are issues related to the construction of the objective function. Most models used to determine a linear discriminant function do so by optimizing some criterion that is a surrogate for minimizing the number of misclassifications. Such an objective is difficult to achieve, especially in the absence of parametric assumptions or other knowledge of the sampling population.

2.2. Glover's MMD and MSID Models

In their original papers, Freed and Glover (1982) presented several alternative formulations of the discriminant problem, based on two criteria for separating observations from a critical value. The first criterion is to maximize the minimum distance (MMD) of an observation's score from the critical value. The second criterion separates the observations by minimizing a measure of external deviations and maximizing a measure of internal deviations (MSID) of the observations from the cutoff, or critical, value. In the sections that follow, we summarize the various MMD and MSID models that have appeared in the literature. The MMD formulation performs best when group overlap is minimal and the variances and covariances across groups are unequal, resulting in 25 to 30 percent lower rates of misclassification than the more commonly applied linear discrimination function. However, when group overlap is substantial (i.e., the groups are close together or intermixed), the MMD model does not perform as well. The results of

the MSID formulation are almost reversed under these conditions. The MSID model performs well when group overlap is maximal, and the performance is poorest when group overlap is minimal. The MSID model also minimizes misclassifications more effectively when the prior probabilities of the groups are unequal.

2.2.1. Maximize the Minimum Distance (MMD) Models

In general, these models try to find a hyperplane that minimizes the maximum exterior deviations. There are several MMD models due to Glover *et al.* These models, defined below as MMD-1 through MMD-4, incorporate different constraint functions to avoid trivial or unbounded solutions.

MMD-1 (Freed and Glover, 1981a):

$$\text{Maximize } d \quad (2.12)$$

$$\text{subject to: } \mathbf{X}_1 \mathbf{w} + d \mathbf{1} \leq c \mathbf{1} \quad (2.13)$$

$$\mathbf{X}_2 \mathbf{w} - d \mathbf{1} \geq c \mathbf{1} \quad (2.14)$$

$$\mathbf{w}, d \text{ unrestricted in sign} \quad (2.15)$$

where c is a pre-chosen boundary value. The objective of MMD-1 is to maximize the minimum deviation, d , of any group member's score from the break point c . Since d is allowed to take negative values and the objective is to maximize d , MMD-1 can be interpreted as minimizing the maximum external deviation, or maximizing the minimum internal deviation.

An important property of MMD-1 is that an unbounded solution indicates perfect separation of the two groups. A potential problem with this model is that it can yield a trivial solution. Unlike MMD-1, the MMD-2 model is concerned only with external deviations, that is, with observations that lie on the wrong side of the separating hyperplane. The objective of MMD-2 is to minimize the maximum external deviation.

MMD-2 (Freed and Glover, 1986b):

$$\text{Minimize } z = e \quad (2.16)$$

$$\text{subject to: } X_1 w - e \mathbf{1} \leq c \mathbf{1} \quad (2.17)$$

$$X_2 w + e \mathbf{1} \geq c \mathbf{1} \quad (2.18)$$

$$e \geq 0 \quad (2.19)$$

$$w \text{ unrestricted in sign} \quad (2.20)$$

where c is a non-zero positive constant. While the MMD-2 model will always produce a bounded solution, it can yield a trivial solution and it is not invariant under data translation.

The MMD-3 model differs from MMD-2 in that c is now a free variable rather than a constant, and a normalization constraint (2.24) eliminates the possibility of a trivial solution. Unfortunately, this constraint can potentially eliminate an optimal solution as well. Also, MMD-3 is not invariant under data translation.

MMD-3 (Freed and Glover, 1986a):

$$\text{Minimize } z = e \quad (2.21)$$

$$\text{subject to: } \mathbf{X}_1 \mathbf{w} - e\mathbf{1} - c\mathbf{1} \leq \mathbf{0} \quad (2.22)$$

$$\mathbf{X}_2 \mathbf{w} + e\mathbf{1} - c\mathbf{1} \geq \mathbf{0} \quad (2.23)$$

$$\mathbf{w}\mathbf{1} + c = S \quad (2.24)$$

$$e \geq 0 \quad (2.25)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (2.26)$$

where S is a non-zero constant. The MMD-4 model as presented below is different from MMD-3 in that the normalization constraint (2.32) has been modified and S becomes a positive constant.

MMD-4 (Freed and Glover, 1986a):

$$\text{Minimize } z = e \quad (2.27)$$

$$\text{subject to: } \mathbf{X}_1 \mathbf{w} - e\mathbf{1} - c\mathbf{1} \leq \mathbf{0} \quad (2.28)$$

$$\mathbf{X}_2 \mathbf{w} + e\mathbf{1} - c\mathbf{1} \geq \mathbf{0} \quad (2.29)$$

$$\mathbf{w} - \mathbf{u}_1 + \mathbf{u}_2 = \mathbf{0} \quad (2.30)$$

$$c - t_1 + t_2 = 0 \quad (2.31)$$

$$\mathbf{u}_1 \mathbf{1} + \mathbf{u}_2 \mathbf{1} + t_1 + t_2 \leq S \quad (2.32)$$

$$\mathbf{u}_1, \mathbf{u}_2 \geq \mathbf{0} \quad (2.33)$$

$$e, t_1, t_2 \geq 0 \quad (2.34)$$

$$w, c \text{ unrestricted in sign} \quad (2.35)$$

where S is a positive constant. Here an upper bound has been placed on the sum of the absolute values of w and c . This has been accomplished by expressing each variable as the difference of two non-negative variables and bounding the sum of the substitute variables. This model has a problem in that it always yields a trivial solution.

MMD-5 (Freed and Glover, 1986b):

$$\text{Minimize} \quad z = e \quad (2.36)$$

$$\text{subject to:} \quad X_1 w - e1 - c1 \leq 0 \quad (2.37)$$

$$X_2 w + e1 - c1 \geq 0 \quad (2.38)$$

$$w1 = S \quad (2.39)$$

$$e \geq 0 \quad (2.40)$$

$$w, c \text{ unrestricted in sign} \quad (2.41)$$

where S is a non-zero constant. This model is invariant under data translation and never yields a trivial solution, but the normalization constraint (2.39), introduced to eliminate the trivial solution, restricts the feasible space considerably.

2.2.2. Maximize Sum of Internal Deviations (MSID) Models

Freed and Glover introduced a series of such models. In general, MSID models identify a hyperplane which combines the goals of (1) maximizing the sum of weighted deviations (internal) from the separating hyperplane, and (2) minimizing the sum of deviations (external) on the wrong side of the hyperplane. In each model, there are constants in the objectives (\mathbf{h}_1 , \mathbf{h}_2 , \mathbf{k}_1 , \mathbf{k}_2 , H , K) that provide subjective weights concerning the importance of the respective deviations. These models are listed below.

MSID-1:

$$\text{Minimize} \quad z = \mathbf{h}_1 \mathbf{e}_1 + \mathbf{h}_2 \mathbf{e}_2 - \mathbf{k}_1 \mathbf{i}_1 - \mathbf{k}_2 \mathbf{i}_2 \quad (2.42)$$

$$\text{subject to:} \quad \mathbf{X}_1 \mathbf{w} + \mathbf{i}_1 - \mathbf{e}_1 - c \mathbf{1} = \mathbf{0} \quad (2.43)$$

$$\mathbf{X}_2 \mathbf{w} - \mathbf{i}_2 + \mathbf{e}_2 - c \mathbf{1} = \mathbf{0} \quad (2.44)$$

$$\mathbf{e}_1, \mathbf{e}_2, \mathbf{i}_1, \mathbf{i}_2 \geq \mathbf{0} \quad (2.45)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (2.46)$$

where $\mathbf{k}_1, \mathbf{k}_2, \mathbf{h}_1, \mathbf{h}_2$ are non-negative weight vectors. The objective in MSID-1 is a combination of maximizing the weighted sum of internal deviations and minimizing the weighted sum of external deviations. By aggregating the values of external deviations and determining e as the maximum external deviation, the MSID-1 reduces to the following model.

MSID-2:

$$\text{Minimize } z = He - k_1 i_1 - k_2 i_2 \quad (2.47)$$

$$\text{subject to: } X_1 w + i_1 - e_1 - c_1 = 0 \quad (2.48)$$

$$X_2 w - i_2 + e_1 - c_1 = 0 \quad (2.49)$$

$$i_1, i_2 \geq 0 \quad (2.50)$$

$$e \geq 0 \quad (2.51)$$

$$w, c \text{ unrestricted in sign} \quad (2.52)$$

where H is a non-negative constant, and k_1, k_2 are non-negative weight vectors. The overall objective of MSID-2 is to minimize the difference between a weighted maximum external deviation and a weighted sum of internal deviations. By aggregating the values of internal deviations and determining i as the maximum internal deviation, the MSID-1 reduces to the following model.

MSID-3:

$$\text{Minimize } z = h_1 e_1 + h_2 e_2 - Ki \quad (2.53)$$

$$\text{subject to: } X_1 w + i_1 - e_1 - c_1 = 0 \quad (2.54)$$

$$X_2 w - i_1 + e_2 - c_1 = 0 \quad (2.55)$$

$$e_1, e_2 \geq 0 \quad (2.56)$$

$$i \geq 0 \quad (2.57)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (2.58)$$

where K is a non-negative constant, and $\mathbf{h}_1, \mathbf{h}_2$ are non-negative weight vectors. The overall objective of MSID-3 is to minimize the difference between a weighted sum of external deviations and a weighted maximum internal deviation. By aggregating the values of internal and external deviations and determining e and i as the maximum external and internal deviations, the MSID-1 reduces to the following model.

MSID-4:

$$\text{Minimize} \quad z = He - Ki \quad (2.59)$$

$$\text{subject to:} \quad \mathbf{X}_1 \mathbf{w} + i \mathbf{1} - e \mathbf{1} - c \mathbf{1} = 0 \quad (2.60)$$

$$\mathbf{X}_2 \mathbf{w} - i \mathbf{1} + e \mathbf{1} - c \mathbf{1} = 0 \quad (2.61)$$

$$e, i \geq 0 \quad (2.62)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (2.63)$$

where H and K are non-negative constants. The overall objective of MSID-4 is to minimize the difference between a weighted maximum external deviation and a weighted maximum internal deviation. To avoid the trivial solutions that were possible with the first four MSID models, Freed and Glover (1986b) suggested appending a suitable normalization constraint to these models. Although such a normalization constraint will certainly rule out the trivial solution, it can also eliminate certain other solutions and possibly an optimal solution as well.

The last MSID model, MSID-5, is proposed by Bajgier and Hill (1982). This model combines two objectives: (1) minimize the sum of external deviations; and (2) maximize the sum of internal deviations.

MSID-5:

$$\text{Minimize } z = H_1 [1\mathbf{e}_1 + 1\mathbf{e}_2] - H_2 [1\mathbf{i}_1 + 1\mathbf{i}_2] \quad (2.64)$$

$$\text{subject to: } \mathbf{X}_1\mathbf{w} - \mathbf{i}_1 + \mathbf{e}_1 - c\mathbf{1} = \mathbf{0} \quad (2.65)$$

$$\mathbf{X}_2\mathbf{w} - \mathbf{i}_2 + \mathbf{e}_2 - c\mathbf{1} = \mathbf{0} \quad (2.66)$$

$$\mathbf{e}_1, \mathbf{e}_2, \mathbf{i}_1, \mathbf{i}_2 \geq \mathbf{0} \quad (2.67)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (2.68)$$

where H_1 and H_2 are non-negative weight scalars. The overall objective of the MSID-5 is a weighted sum of the above two objectives. With $H_2=0$, MSID-5 reduces to an MSD model. MSID-5 is a fairly well behaved model in that it rarely yields a trivial solution. However, it gives an unbounded solution with $H_1 < H_2$. In general, care must be taken in choosing the objective function weights to prevent unbounded solutions.

The straight-forward approaches of MMD and MSID formulations possess inherent limitations in their solutions which are not shared by classical statistical techniques. Researchers have focused attention on these limitations with the discovery of data sets in which the discrimination power of the LP formulation appears to break down. The difficulties identified in the solution features of LP discriminant formulations may be classified under the headings of degeneracy and stability. As a step toward their resolution, several researchers have shown that these difficulties stem from *normalizations*

implicitly used in LP models. One such proposed normalization sets the sum of the scaling variables equal to a constant to overcome the degeneracy and the instability problems. This property causes the essential form of the LP discriminant solution to be invariant relative to data translations.

There remain two limitations of the LP discriminant models that have not properly been addressed in any of the MMD and MSID models. They are: (1) for best results, the LP discriminant model must be solved twice, once for a normalization constant with positive sign and once for a normalization constant with negative sign; (2) the stability of the proposed normalization is incomplete, failing to yield solutions that are invariant relative to rotations of the problem data. Glover (1990) proposed a new hybrid class of models for discrimination problems by using a normalization that is invariant to both translations and rotations yielding, as a consequence, the stability property previously unachieved for LP discriminant models.

2.2.3. Hybrid Models (Glover, 1990)

To overcome the limitations of the various MMD and MSID models, Glover (1990) developed a hybrid model which combines the basic features of previously developed models and newly proposed normalizations into a single comprehensive model. The most notable improvement of such a model over the previous models is its ability to offer superior solutions by eliminating distortions.

Two primary forms of the hybrid discriminant models as proposed by Glover are given below. In these models, $\mathbf{e}_1, \mathbf{e}_2, \mathbf{i}_1,$ and \mathbf{i}_2 are external and internal deviation variables for points in groups 1 and 2. They refer to the magnitude by which the points lie outside or inside their targeted half spaces. The corresponding coefficients $\mathbf{h}_1, \mathbf{h}_2, \mathbf{k}_1,$ and \mathbf{k}_2 in the objective function discourage external deviations and encourage internal deviations. The h_0 and k_0 are constants to weight the *maximum external deviation* (e_0) and the *minimum internal deviation* (i_0), respectively. [The effects of these variables can be segregated by introducing separate constraints of the form $(\mathbf{X}_1\mathbf{w} - i_0 + e_0 \leq c\mathbf{1})$ and $(\mathbf{X}_2\mathbf{w} + i_0 - e_0 \geq c\mathbf{1})$ at the expense of enlarging the model form.]

Hybrid-1:

$$\text{Minimize} \quad z = h_0 e_0 + \mathbf{h}_1 \mathbf{e}_1 + \mathbf{h}_2 \mathbf{e}_2 - k_0 i_0 - \mathbf{k}_1 \mathbf{i}_1 - \mathbf{k}_2 \mathbf{i}_2 \quad (2.69)$$

$$\text{subject to:} \quad \mathbf{X}_1 \mathbf{w} + i_0 \mathbf{1} + \mathbf{i}_1 - e_0 \mathbf{1} - \mathbf{e}_1 - c\mathbf{1} = \mathbf{0} \quad (2.70)$$

$$\mathbf{X}_2 \mathbf{w} - i_0 \mathbf{1} - \mathbf{i}_2 + e_0 \mathbf{1} + \mathbf{e}_2 - c\mathbf{1} = \mathbf{0} \quad (2.71)$$

$$-n_2 \mathbf{1X}_1 \mathbf{w} + n_1 \mathbf{1X}_2 \mathbf{w} = 2n_1 n_2 \quad (2.72)$$

$$\mathbf{e}_1, \mathbf{e}_2, \mathbf{i}_1, \mathbf{i}_2 \geq \mathbf{0} \quad (2.73)$$

$$e_0, i_0 \geq 0 \quad (2.74)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (2.75)$$

Equation (2.72) is a new normalization where n_1 and n_2 are the number of elements in groups 1 and 2, respectively. The right hand side, $2n_1n_2$, is an arbitrary scaling choice for a positive constant. This particular choice tends to yield \mathbf{w} values closer to an average absolute value of 1.

Hybrid-2:

$$\text{Minimize} \quad z = h_0 e_0 + \mathbf{h}_1 \mathbf{e}_1 + \mathbf{h}_2 \mathbf{e}_2 - k_0 i_0 - \mathbf{k}_1 \mathbf{i}_1 - \mathbf{k}_2 \mathbf{i}_2 \quad (2.76)$$

$$\text{subject to:} \quad \mathbf{X}_1 \mathbf{w} + i_0 \mathbf{1} + \mathbf{i}_1 - e_0 \mathbf{1} - \mathbf{e}_1 - c \mathbf{1} = \mathbf{0} \quad (2.77)$$

$$\mathbf{X}_2 \mathbf{w} - i_0 \mathbf{1} - \mathbf{i}_2 + e_0 \mathbf{1} + \mathbf{e}_2 - c \mathbf{1} = \mathbf{0} \quad (2.78)$$

$$2n_1 n_2 (i_0 - e_0) + n_1 (\mathbf{1i}_2 - \mathbf{1e}_2) + n_2 (\mathbf{1i}_1 - \mathbf{1e}_1) = 2n_1 n_2 \quad (2.79)$$

$$e_0, i_0, \mathbf{e}_1, \mathbf{e}_2, \mathbf{i}_1, \mathbf{i}_2 \geq 0 \quad (2.80)$$

$$\mathbf{w}, c \text{ unrestricted in sign} \quad (2.81)$$

The difference between this model and the Hybrid-1 model is in the normalization constraint. The normalization constraint (2.79) is a simplified form and offers easy incorporation into an LP formulation as the coefficients of the variables do not require extensive calculations as in normalization constraint (2.72). The constraint (2.79) is derived by adding n_2 times each equation of (2.77) and subtracting n_1 times each equation of (2.78).

To achieve strict separation of points by the hyperplane, the quantity c can be replaced by $c-\varepsilon$ for group 1 and $c+\varepsilon$ for group 2 where ε is a positive constant. This will ensure that the elements of group 1 and group 2 lie strictly inside their half spaces whose boundary is marked by c .

2.2.4. Improved Solutions from Hybrid Models

The normalizations (2.72) and (2.79) as defined in the above models are equivalent to creating a *meaningful separation* of group 1 and group 2. They also eliminate the null weighting $\mathbf{w} = \mathbf{0}$ as a feasible solution. A hyperplane creates a meaningful separation of group 1 and group 2 if $n_2(d_1 + d_2) + n_1(d_1 + d_2) > 0$ where d_1 and d_2 denote the net internal deviation of a point in their respective groups from the hyperplane generated by the discriminant model. Hence d_1 is positive (or zero) if group 1 lies within its targeted half space and negative otherwise.

The new normalization constraints ensure a feasible dual solution, thereby guaranteeing that the LP discriminant formulation is bounded for optimality. Necessary conditions for bounded optimality are immediately evident from its dual formulation. Also evident are the necessary conditions for certain variables of the LP discriminant formulation to be nonzero at optimality.

The necessary and sufficient conditions for bounded optimality and non-trivial solutions are:

$$\mathbf{h}_j \geq \mathbf{k}_j \quad \text{for } j = 1, 2 \quad (2.82)$$

$$\mathbf{1k}_1 + \mathbf{1k}_2 \leq k_0 \leq \mathbf{1h}_1 + \mathbf{1h}_2 \quad (2.83)$$

$$\text{Min} \left(\frac{h_0}{2}, h_{1\min}, h_{2\min} \right) \geq \text{Max} \left(\frac{k_0}{2}, k_{1\max}, k_{2\max} \right) \quad (2.84)$$

where $h_{1\min}$ and $h_{2\min}$ are the minimum elements of \mathbf{h}_1 and \mathbf{h}_2 , and $k_{1\max}$ and $k_{2\max}$ are the maximum elements of \mathbf{k}_1 and \mathbf{k}_2 respectively. It may be noted that $\mathbf{h}_j > \mathbf{k}_j$ for $j = 1, 2$ implies that at most one of \mathbf{e}_j and \mathbf{i}_j for $j = 1, 2$ will be positive, an outcome that also holds when $\mathbf{h}_j = \mathbf{k}_j$ for $j = 1, 2$ in the case of extreme point solutions.

2.3. Successive Goal Approach

Researchers tend to measure the usefulness of a solution by its classification power on the training sample or on a validation sample. However, the objectives used in mathematical programming models are only loose approximations to the real objective of maximizing classification power. Further, many of the models are patched up models which exhibit undesirable properties such as yielding meaningless answers, unbounded solutions, and unbalanced classifications.

Glover's Hybrid models with their new normalizations presented in the last section have shown a significant improvement in the determination of a stable, balanced, and unbounded solution. A particularly significant improvement can be had in the classification of observations by repeatedly solving the model while manipulating its objective function. Appropriately termed as the *successive goal approach*, Glover's new

method seeks to identify the points from each group as perfectly differentiated by shifting the separating hyperplane alternately in each direction. This is done by increasing and decreasing the constant value c by an amount such that all the points of the target group are strictly on one side of the hyperplane. Upon identifying the shift for a given group, all points of the alternate group which lie strictly beyond the shifted hyperplane boundary become perfectly differentiated. Such perfectly differentiated points can then be segregated from the remaining points before applying the next stage. The number of stages devoted to creating perfect classification before accepting the current hyperplane without shifting is decided by the decision maker in advance.

2.3.1. Procedure for the Successive Goal Approach

For the successive goal method, a classifier can be constructed as follows:

- Step 0: Set up the problem as a two-group classification problem. For the multiple group case, create any subset of groups as group 1 and the remaining subset as group 2.
- Step 1: Formulate the problem as Glover's new Hybrid model.
- Step 2: Assign objective function coefficients h_1 , h_2 , k_1 , and k_2 values such that $h_i > k_i$. For a balanced differentiation at each successive stage, maintain a ratio of approximately 100:1 between h_1 , h_2 and k_1 , k_2 . Set k_0 as $(1k_1 + 1k_2) + 1$ and h_0 as $3 * k_0$. Set ϵ to zero.
- Step 3: If there are no more points to be differentiated from either group or if no additional iterations remain to be performed, then stop and go to Step 8. Otherwise, solve the model using any robust LP package.

- Step 4: Compute the maximum shift of the hyperplane towards each group. The maximum hyperplane shift into group 1 until only group 1 points remain = $\{\max(\text{external deviation for each point in group 2})\}$; likewise, the equivalent shift into group 2 = $\{\max(\text{external deviation for each point in group 1})\}$.
- Step 5: Identify perfectly differentiated points in each group by comparing their internal deviations against that group's maximum shift. If the maximum shift for a group is less than the internal deviation of a point in that group then classify that point as perfectly differentiated.
- Step 6: Save the classifier (w, c) , the points differentiated on each side, and the maximum shift in each direction.
- Step 7: Re-assign new values to the objective function coefficients h_1 , h_2 , k_1 , and k_2 for all the current and the previously differentiated points by reducing them by a factor of 10 at each stage. Reset h_0 and k_0 accordingly. Go to Step 3.
- Step 8: If there are still some points that remain to be differentiated then solve the model one more time. Use the new hyperplane to differentiate the remaining points without shifting.
- Step 9: List the number of points differentiated in each group. Stop.

CHAPTER 3

AN EMPIRICAL COMPARISON OF METHODS

3.1. Introduction

This chapter investigates and compares the performance of one parametric and two non-parametric discriminant analyses which are based on a statistical discriminant analysis method and two relatively new mathematical programming-based methods. In Section 3.2 of this chapter, we review the criteria for selecting classification techniques for discriminant analysis. In Section 3.3, we list the methods selected for our experiment and briefly describe their formulation. In Section 3.4, we discuss the design of our experiment, review the criteria for selecting the data variables, and discuss the classification efficiency and costs of misclassification. We also lay out a step-wise procedure for each method. Finally, Section 3.5 presents the results of our experimentation.

3.2. Selection of a Discrimination Method

In the previous two chapters, we have examined a number of methods that can learn from data and make predictions on new cases. They are the most widely used discrimination methods for various problems of data classification in real-world applications. These methods, regardless of their origin, possess some form of curve-fitting properties where each method makes a certain assumption about an underlying model, and attempts to train itself (or learn) within that framework. For any given model, a good discrimination method must be capable of finding not one but several "good" fits within that model. As is often the case with the real-world discrimination problems, there

is never enough data to carry out a true training of a model. Therefore, due to these typical shortages of data, it becomes imperative that a method use data efficiently.

The key question is: How do we select the best method? In our literature survey of discrimination methods employed in various disciplines, we noticed a certain tendency among researchers to adhere to just one or two familiar methods of discrimination. While familiarity is comforting, it, nonetheless, deprives the researcher of the technological and qualitative advances offered by newer methods. To select the most appropriate method for a given sample, one should use the method that yields the best empirical performance. The standard to be used for deciding which method is best might well be the best accuracy in terms of lowest estimated true error rates. But this criterion undoubtedly requires a large amount of data. If one has lots of time and computational resources, this is a reasonable approach. While several such secondary factors may play a role in deciding which methods to apply, accuracy of performance should always remain the primary criterion in selecting a discrimination method.

As we noted in Chapter 1, small sample sizes do not accurately estimate true error rates. Many real-world sample sizes are in the hundreds, not in the thousands as required for unbiased estimation. Thus, we need to find or develop a procedure that will estimate the true error rate properly. It has been determined that a train-and-test model with a suitable cross-validation can yield reliable error rates. In general, a 10-fold cross validation is adequate and sufficient for obtaining reliable error rate estimators. When the sample size is small (less than 100), and particularly when it is less than 50, then leaving-one-out cross-validation should be used.

3.3. Methods Selected

In our study, we have selected three methods to compare—two non-parametric and one parametric. While the main purpose of the study is to compare Glover's new successive goal approach to a single-linear programming method (both non-parametric methods), we have included Smith's quadratic discriminant method (QDM) for comparison. In the following sub-sections, we describe the exact formulation for each method selected.

3.3.1. Smith's Quadratic Discriminant Analysis Method (QDM)

This is a parametric method which is well suited for data with multivariate normality and unequal variance-covariance matrices. Smith's quadratic discriminant method as discussed in Section 1.5.1.3 minimizes the total probability of misclassification. In a two group problem, this classification rule assigns a point \mathbf{x} to group G_1 if:

$$(\mathbf{x} - \bar{\mathbf{x}}_2)' \bar{\mathbf{S}}_2^{-1} (\mathbf{x} - \bar{\mathbf{x}}_2) - (\mathbf{x} - \bar{\mathbf{x}}_1)' \bar{\mathbf{S}}_1^{-1} (\mathbf{x} - \bar{\mathbf{x}}_1) \geq \ln|\bar{\mathbf{S}}_1| - \ln|\bar{\mathbf{S}}_2|$$

and to group G_2 otherwise, where $\bar{\mathbf{x}}_1$ and $\bar{\mathbf{x}}_2$ are the group mean vectors and $\bar{\mathbf{S}}_1$, and $\bar{\mathbf{S}}_2$ are estimates of the population dispersion (variance-covariance) matrices for groups 1 and 2, respectively.

3.3.2. Glover's Mathematical Programming-Based Hybrid Model

A relatively new member of the LP-based family of models, we have chosen Glover's Hybrid model for its stability and optimality. As discussed before, with a new normalization constraint and a careful selection of the objective function coefficients, the

model guarantees a bounded nontrivial optimal solution. This is a significant improvement for LP formulations that have suffered in the past from trivial and unbounded solutions. The Hybrid-1 model, shown here, was used for the single-LP approach.

$$\text{Minimize} \quad z = h_0 e_0 + h_1 e_1 + h_2 e_2 - k_0 i_0 - k_1 i_1 - k_2 i_2 \quad (3.1)$$

$$\text{subject to:} \quad X_1 w + i_0 \mathbf{1} + i_1 - e_0 \mathbf{1} - e_1 - c \mathbf{1} = \mathbf{0} \quad (3.2)$$

$$X_2 w - i_0 \mathbf{1} - i_2 + e_0 \mathbf{1} + e_2 - c \mathbf{1} = \mathbf{0} \quad (3.3)$$

$$-n_2 \mathbf{1} X_1 w + n_1 \mathbf{1} X_2 w = 2n_1 n_2 \quad (3.4)$$

$$e_1, e_2, i_1, i_2 \geq 0 \quad (3.5)$$

$$e_0, i_0 \geq 0 \quad (3.6)$$

$$w, c \text{ unrestricted in sign} \quad (3.7)$$

3.3.3. Glover's Successive Goal Approach

An extension of the above hybrid model, this approach relies on segregating perfectly differentiated points in each group and attenuating their influence in the objective function. Segregation of points is accomplished by shifting the hyperplane (by increasing and decreasing c) in the direction of each group until some points become perfectly differentiated. The influence of these segregated points is reduced by successively assigning very low weights to their objective function coefficients at each iteration.

3.4. Experimental Design

While Smith's LDM method and Glover's Hybrid LP model are widely discussed in the literature, there is no published information yet on the implementation and performance of Glover's successive goal method. In order to investigate the quality of the solution produced by the successive goal method, we designed our study to include the following two experiments:

- (1) Verify and validate the quality of the successive goal method on a well-known data set drawn from the discriminant analysis literature, and
- (2) Perform the discriminant analysis on a new data set.

3.4.1. Data and Variable Selection

To fulfill our first experiment, we chose Fisher's Iris data, a classic data set, which is a set of measurements that relate to three species of iris. Measurements of length and width for sepal and petal were made on fifty plants from each of iris setosa, iris versicolor and iris virginica for the purpose of constructing a classification rule. It was this data that originally motivated Fisher to derive the linear discriminant function, so it was appropriate that we also select the same for our experiment as well. While, for simplicity and ease of comparison, we included all four variables of Iris data, we restricted our study to only two of the three groups. We performed a 10-fold cross-validation where the first 45 observations from each of the two groups (setosa and versicolor) were used as the training data and the remaining 5 observations constituted test or holdout data. The test was replicated 10 times with random subsampling.

For our second experiment, we selected a data set consisting of financial information on 188 U.S. banks, grouped as failed (88 banks) and successful (100 banks). These values were a randomly selected subset of the 1984-1989 data used in Barr, Sieford, and Siems (1992). The independent variables chosen for this study include eight

variables based on a set of ratios commonly utilized for analysis and research in the banking industry. To make sure that all major areas of traditional bank performance analysis were represented, a management-quality variable, DEA, was added for completeness (see Barr, Sieford, and Siems, 1992). Table 3.1 and Table 3.2 list the selected variables and their statistics employed in the study. We performed a 10-fold cross-validation on this data where the first 79 observations from group 1 (failed banks) and 90 observations from group 2 (successful banks) were used as the training data and the remaining 9 observations from group 1 and 10 observations from group 2 constituted test or holdout data. The test was replicated 10 times with random subsampling.

Table 3.1.--Variable definitions

Variable	Definition
EC-TL	Ratio of Equity Capital to Total Loans
NL-TA	Ratio of Nonperforming Loans to Total Assets
TL-TA	Ratio of Total Loans to Total Assets
OR-TA	Ratio of Other Real Estate Owned to Total Assets
NI-TA	Ratio of Net Income to Total Assets
NE-TA	Ratio of Noninterest Expense to Total Assets
LA-TA	Ratio of (Liquid Assets minus Volatile Liabilities) to Total Assets
BD-TA	Ratio of Bid Time Deposits to Total Assets
DEA	Data Envelopment Analysis Score

Table 3.2.--Sample variable statistics

Variable	Mean	Median	Std. Dev	Minimum	Maximum
	Group 1	Group 1	Group 1	Group 1	Group 1
	Group 2	Group 2	Group 2	Group 2	Group 2
EC-TL	39911	20215	82926	882	509836
	31557	17677	37746	2092	229674
NL-TA	62669	62669	62669	62669	62669
	49981	28549	58567	3614	329978
TL-TA	6462	3875	10373	282	62936
	5572	3165	6300	442	37428
OR-TA	2158	1267	3568	182	20766
	2704	1442	4946	128	42546
NI-TA	43	24	70	2	417
	35	24	73	4	230
NE-TA	28862	18023	41114	484	280365
	33342	18932	40201	1090	203179
LA-TA	4590	4261	2662	954	13978
	3026	2453	2251	1152	13978
BD-TA	2209	2052	915	1292	5435
	2310	2052	743	1441	6435
DEA	3659	3654	261	3054	4344
	3662	3654	106	3376	4124

3.4.2. Classification Efficiency and Costs of Misclassification

As discussed in Chapter 1, we used the average apparent error rate as the measure for classification efficiency. When discussing the classification efficiency of various models, it is important to distinguish between Type I and Type II errors. In our study of the bank data, the null hypothesis, H_0 , was set such that we recorded a Type I error whenever the classifier classified a failed bank as successful, and a Type II error if the classifier classified a successful bank as failed. Of the two errors, the Type I error, in this case, usually is the more serious since investors, depositors, or the Federal Deposit Insurance Corporation could lose a substantial amount of money to one of these

enterprises. Assuming an equal cost of misclassification for committing each type of error, the goal of any objective function here should be to minimize the total cost of misclassification.

3.4.3. Testing of Methods

In this section, we describe a step-wise procedure for carrying out the experiment for each method. The procedures include assumptions, special parameters, and appropriate stopping criteria.

3.4.3.1. Smith's Quadratic Discriminant Method

The following procedure for Smith's QDM method is based on the bank data. However, except for certain parameters and the data differences, the procedure for the Iris data remains the same.

- Step 0: Set up the problem as a two-group (failed banks as group 1, successful banks as group 2) classification problem with nine variables as shown in Table 3.1. Assign a prior probability of 0.47 to the failed group and a prior probability 0.53 to the successful group based on their sample sizes of 88 and 100, respectively.
- Step 1: For a 10-fold cross-validation, construct a set of 10 random subsamples of training and test data, each consisting of 79 and 9 observations for group 1 and 90 and 10 observations for group 2.
- Step 2: For each sample of training data, compute the quadratic discriminant function for each group. We used a PC-based BASIC program from James (1985).
- Step 3: Classify the observation from each test data group using its respective classifier. Record misclassifications as error Type I or Type II.
- Step 4: Compute the average misclassification from all the 10 tests. Stop.

3.4.3.2. Glover's Hybrid Single LP Model

The following procedure for Glover's LP formulation is based on the bank data. However, except for certain parameters and the data differences, the procedure for the Iris data remains the same.

- Step 0: Set up the problem as (3.1) - (3.7) with failed banks as group 1, successful banks as group 2, and the nine variables shown in Table 3.1.
- Step 1: For a 10-fold cross-validation, construct a set of 10 random subsamples of training and test data, each consisting of 79 and 9 observations for group 1, and 90 and 10 observations for group 2.
- Step 2: For each sample of training data, solve the LP model using an appropriate LP package. Note the classifier (w,c) values for each training sample. (We used a PC-based LP package from IBM called OSL running under GAMS).
- Step 3: Classify the observation from each test data using its respective (w,c) classifier. Record misclassifications as error Type I or Type II.
- Step 4: Compute the average misclassification from all the 10 tests. Stop.

3.4.3.3. Glover's Successive Goal Approach

The following procedure for Glover's successive goal approach is based on the bank data. However, except for certain parameters and the data differences, the procedure for the Iris data remains the same.

- Step 0: Set up the problem as (3.1) - (3.7) with failed banks as group 1, successful banks as group 2, and the nine variables shown in Table 3.1. Initialize the objective function coefficients and parameters as necessary. A listing of the model with the bank data is shown in Appendix A. Set the iteration counter $k = 0$.
- Step 1: For a 10-fold cross-validation, construct a set of 10 random subsamples of training and test data, each consisting of 79 and 9 observations for group 1, and 90 and 10 observations for group 2. Designate all 19 of the test data (group 1 and group 2) observations as yet-to-be-classified (YTBC).
- Step 2: Increment the iteration counter k by 1. If the number of iterations exceeds the specified limit, go to Step 6.
- Step 3: For each sample of training data, solve the LP model as described in Chapter 2 using an appropriate LP package. Record the classifier function (\mathbf{w}, c) and its maximum shift in the direction of each group. (We used a PC-based LP package from IBM called OSL running under GAMS).
- Step 4: Compute (\mathbf{w}_k, c_k^1) and (\mathbf{w}_k, c_k^2) as the shifted classifiers for groups 1 and 2.
- Step 5: Classify each YTBC observation from groups 1 and 2 test data by evaluating it on (\mathbf{w}_k, c_k^1) . If a YTBC from group 1 test data classifies as

group 1, record the observation as correctly classified. However, if a YTBC from group 1 test data classifies as group 2 data, record that observation as misclassified generating a Type I error. If neither happens, leave the observation as YTBC. Next, repeat the evaluation on (\mathbf{w}_k, c_k^2) for all remaining YTBC observations. The misclassification here will generate a Type II error. Go to Step 2.

Step 6: Solve the LP problem again for $(\mathbf{w}_{k+1}, c_{k+1})$. Without any shifting, apply this classifier $(\mathbf{w}_{k+1}, c_{k+1})$ on each YTBC. Record any new Type I and Type II errors.

Step 7: Repeat Step 2 through Step 6 for each sample data set.

Step 8 : Compute the average misclassification from all the 10 tests. Stop.

3.5. Experimental Results and Analysis

Table 3.3 shows the results of the 10-fold cross-validation testing on the Iris dataset, which were identical on all methodologies, as summarized in Table 3.4. In all ten instances, all methods were able to perfectly differentiate the 10 test observations, based on a discriminant model developed from the 90 training observations. The data followed the parametric requirements of the quadratic method, and the testing not only verified the appropriateness of the approach in that context, but effectiveness of the LP-based methods as well.

Table 3.3.--10-fold cross-validation (Iris data): all models

Test No.	Train=45 Test=5	Train=45 Test=5	Correct Group 1	Correct Group 2	Type I Error	Type II Error	Total Error
	Group 1	Group 2	Group 1 as Group 1	Group 2 as Group 2	Group 1 as Group 2	Group 2 as Group 1	
1	45	45	5	5	0	0	0
2	45	45	5	5	0	0	0
3	45	45	5	5	0	0	0
4	45	45	5	5	0	0	0
5	45	45	5	5	0	0	0
6	45	45	5	5	0	0	0
7	45	45	5	5	0	0	0
8	45	45	5	5	0	0	0
9	45	45	5	5	0	0	0
10	45	45	5	5	0	0	0
Avg	45	45	5	5	0	0	0%

Table 3.4.--Experimental results for 10-fold cross validation (Iris data)

Method	Type I Error	Type II Error	Misclassification Error (%)	Accuracy (%)
Quadratic Method	0	0	0%	100%
Hybrid Model	0	0	0%	100%
Successive Approach	0	0	0%	100%

Tables 3.5 through 3.7 summarize the experimental results on the bank data. The quadratic method classified all of the test points in all ten holdouts as being in group 2; hence all 88 group 1 observations were misclassified. Table 3.5 gives the 10-fold cross-validation results for the single-LP Hybrid model, which had an accuracy of 82.5% with 17.5% misclassifications. In this testing, all $h_j=200$ and all $k_j=1$, which seemed to give the best discrimination based on preliminary testing, and h_0 and k_0 were computed as described previously to ensure bounded and nontrivial solutions.

Table 3.5.--10-fold cross-validation (bank data): single stage hybrid model

Test No.	Train=79 Test=9	Train=90 Test=10	Correct Group 1	Correct Group 2	Type I Error	Type II Error	Total Error
	Group 1 (Failed)	Group 2 (Success)	Group 1 as Group 1	Group 2 as Group 2	Group 1 as Group 2	Group 2 as Group 1	
1	69	68	7	7	2	3	5
2	70	64	7	8	2	2	4
3	68	67	8	8	1	2	3
4	71	66	7	8	2	2	4
5	70	67	7	8	2	2	4
6	69	64	8	8	1	2	3
7	68	62	9	0	0	0	0
8	66	78	7	8	2	2	4
9	69	66	9	7	0	3	3
10	69	68	7	7	0	3	3
Avg	68.9	67	7.6	7.9	1.2	2.1	17.5%

Table 3.6.--10-fold cross-validation (bank data): successive goal method

Test No.	Train=79 Test=9	Train=90 Test=10	Correct Group 1	Correct Group 2	Type I Error	Type II Error	Total Error
	Group 1 (Failed)	Group 2 (Success)	Group 1 as Group 1	Group 2 as Group 2	Group 1 as Group 2	Group 2 as Group 1	
1	75	86	6	8	3	2	5
2	76	87	7	7	2	3	4
3	76	85	6	8	3	2	3
4	75	87	6	8	3	2	4
5	74	78	7	7	2	3	4
6	75	85	9	10	0	0	3
7	74	85	9	9	0	1	0
8	73	84	7	10	2	0	4
9	69	72	8	7	1	3	3
10	75	84	6	10	1	0	3
Avg	74.2	83.3	7.1	8.4	1.7	1.6	17.5%

Table 3.7.--Experimental results for 10-fold cross validation (bank data)

Method	Type I Error	Type II Error	Misclassification Error (%)	Accuracy (%)
Quadratic Method	88	0	46.5%	53.5%
Hybrid Model	12	21	17.5%	82.5%
Successive Approach	17	16	17.5%	82.5%

Table 3.6 presents the same results for the successive goal method, where six iterations, or hierarchies, were used. Surprisingly, the accuracy rate was identical to the Hybrid model, although the proportions of Type I and Type II errors were different.

The two experiments employing the above three methods encountered their share of idiosyncrasies. Most of the issues appeared to be data related. Here are some noteworthy observations.

- (1) The two groups chosen from the Iris data (setosa and versicolor) are well differentiated and hence didn't produce any misclassification. We used a linear discriminant classifier as well and the results were identical.
- (2) The statistics on the bank data revealed existence of no parametric distribution. This rendered the quadratic discriminant classifier useless on such a data. This fact was obvious from a high degree of misclassification returned by the statistical classifier.
- (3) As expected, the outliers in the sample data appeared to have a significant influence on the performance of each method.

Analysis of the results shows that successive goal approach performed better than the Hybrid model, which in turn performed much better than the classical quadratic discriminant procedure. While the Hybrid model performed better than the previous LP models, the successive goal approach has emerged as the strongest alternative to any of the previous methods. Let us discuss these conclusions in greater detail.

- (1) The quadratic discriminant approach did not provide a good classification as the data set did not possess adequate distribution properties. Lack of knowledge about the groups' prior probabilities forced us to base them on the sample sizes.
- (2) The Hybrid formulation appeared very stable under numerous settings of objective function coefficients. It held up well against unboundedness.
- (3) The successive goal method differentiated most points during the first two shifts after which it tapered off. The incremental gain at succeeding stages was marginal. However, the solution was found to be intuitive and provided information about the structure of the data.
- (4) It is noteworthy that outliers appeared to be an important factor in the application of the successive goal approach. That is, the shifting methodology appears to be adversely affected by extreme points from each group which reach most deeply into the territory of the other. Removing these extreme points suspected to be unrepresentative of the original group appeared to improve the solution significantly.
- (5) The successive goal method clearly improved the classification at each stage of iteration. Figure 3.1 and Figure 3.2 show the incremental improvement over the Hybrid model.

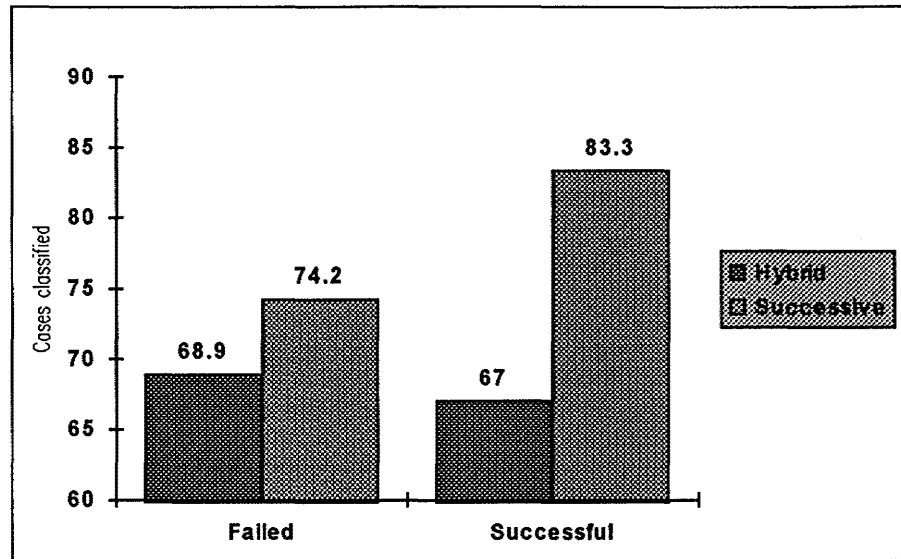


Figure 3.1. Classification improvement over Hybrid model

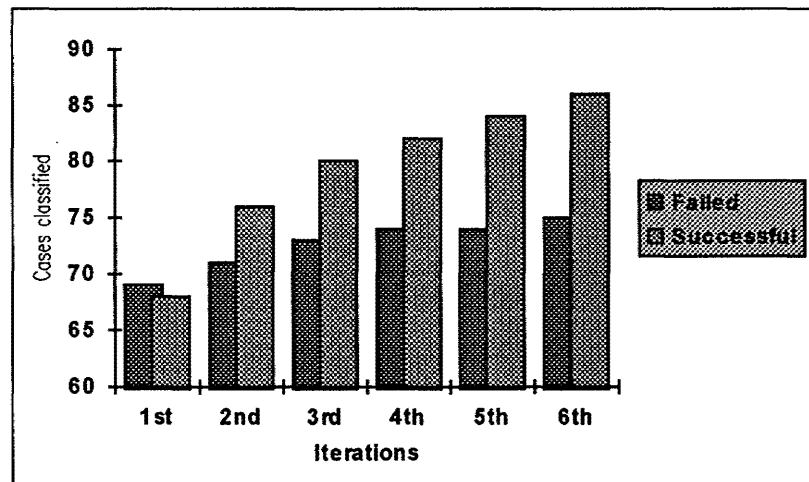


Figure 3.2. Successive improvement of classification

CHAPTER 4

SUMMARY AND CONCLUSIONS

This praxis contributes to the development and implementation of Glover's new LP-based successive goal approach for solving discriminant analysis problems. A GAMS-based LP package from IBM (OSL) was used on a PC to validate the robustness of the two normalization constraints proposed by Glover in his new Hybrid model.

We began with a general description of the classification problems in Chapter 1, where we also reviewed discriminant analysis vis-a-vis classification, ANOVA and cluster analysis. We included in the chapter a detailed survey of discriminant analysis methods and their application in various fields of interest. An in-depth review of discriminant analysis methods based on a mathematical programming approach was presented in Chapter 2 with an emphasis on Freed and Glover's new Hybrid model and a successive goal approach based on the Hybrid model.

In Chapter 3, we presented two non-parametric and one parametric method for use in the design of the experiment, construction of the models, and their solution. The methods selected were Smith's quadratic discriminant method, Glover's Hybrid model, and Glover's new successive goal method. For data, we used Fisher's classic Iris data, and financial data from a study of 188 failed and successful U.S. banks.

The computational results show that the Glover's new normalization constraints improve the quality of the LP solution, especially in the area of non-trivial and unbounded solutions. His newly proposed successive goal approach was equally accurate in classifying new observations; employing the decision-tree technique, this method is intuitive and lends itself well to discrimination problems. The performance of the LP-based approaches to the discriminant problem is poised to quiet the skeptics and critics of their viability.

APPENDIX A

```

$TITLE LP DISCRIMINANT ANALYSIS MODEL -- Bank Data from R. Barr
$OFFUPPER
*Glover's 1989 Discriminant Models
*Output options (see p. 103, 194, 281 in GAMS Reference manual)
OPTION ITERLIM=9000, LIMROW=0, LIMCOL=0, DECIMALS=7, SOLPRINT=OFF;
OPTION RESLIM=2000, OPTCA=0.0, BRATIO=0.25;
SETS
  G1 Group 1 training data /G1R1*G1R79/
  G2 Group 2 training data /G2R1*G2R90/
  T1 Group 1 test data /T1R1*T1R9/
  T2 Group 2 test data /T2R1*T2R10/
  J Attributes /EC-TL, NL-TA, TL-TA, OR-TA, NI-TA, NE-TA, LA-TA, BD-TA, DEA/;
PARAMETER H0 Weight for external deviation constant;
PARAMETER K0 Weight for maximum internal deviation;
PARAMETER H1(G1) Weights for external deviations in Group 1;
PARAMETER H2(G2) Weights for external deviations in Group 2;
PARAMETER K1(G1) Weights for internal deviations in Group 1;
PARAMETER K2(G2) Weights for internal deviations in Group 2;
PARAMETER N1 Observations in Group 1;
PARAMETER N2 Observations in Group 2;
PARAMETER SEP Epsilon for separation amount;
PARAMETER WH H weights;
PARAMETER WK K weights;
PARAMETER RS2 Weight reduction stage 2;
PARAMETER RS3 Weight reduction stage 3;
PARAMETER RS4 Weight reduction stage 4;
PARAMETER RS5 Weight reduction stage 5;
PARAMETER RS6 Weight reduction stage 6;
PARAMETER RAT Weight ratio for each stage;

WH = 200.0; WK = 1.0; RAT = 0.3;
RS2 = 0.10; RS3 = RAT*RS2; RS4 = RAT*RS3; RS5 = RAT*RS4; RS6 = RAT*RS5;
H1(G1) = WH; H2(G2) = WH;
K1(G1) = WK; K2(G2) = WK;
K0 = SUM(G1, K1(G1)) + SUM(G2, K2(G2)) + 1;
H0 = 3*K0;
SEP = 0.0; N1 = CARD(G1); N2 = CARD(G2);

PARAMETER G1MAX Max external for Group 1;
PARAMETER G2MAX Max external for Group 2;
PARAMETER GR1SH1(G1) (BETA1.L - G2Max) Group 1 & Stage 1;
PARAMETER GR2SH1(G2) (BETA2.L - G1Max) Group 2 & Stage 1;
PARAMETER GR1SH2(G1) (BETA1.L - G2Max) Group 1 & Stage 2;
PARAMETER GR2SH2(G2) (BETA2.L - G1Max) Group 2 & Stage 2;
PARAMETER GR1SH3(G1) (BETA1.L - G2Max) Group 1 & Stage 3;
PARAMETER GR2SH3(G2) (BETA2.L - G1Max) Group 2 & Stage 3;
PARAMETER GR1SH4(G1) (BETA1.L - G2Max) Group 1 & Stage 4;
PARAMETER GR2SH4(G2) (BETA2.L - G1Max) Group 2 & Stage 4;
PARAMETER GR1SH5(G1) (BETA1.L - G2Max) Group 1 & Stage 5;
PARAMETER GR2SH5(G2) (BETA2.L - G1Max) Group 2 & Stage 5;
PARAMETER GR1SH6(G1) (BETA1.L - G2Max) Group 1 & Stage 6;
PARAMETER GR2SH6(G2) (BETA2.L - G1Max) Group 2 & Stage 6;

PARAMETER KOUNTG1 G1 points differentiated;
PARAMETER KOUNTG2 G2 points differentiated;
PARAMETER CUMULG1 Cumulative G1 points differentiated;
PARAMETER CUMULG2 Cumulative G2 points differentiated;
PARAMETER FLAGT1(T1) (1SN: S=stage N=1or2);
PARAMETER FLAGT2(T2) (2SN: S=stage N=1or2);

```

PARAMETER A1XB(T1) Computed (AX-B) for Test Group 1 (Yes if < 0);
 PARAMETER A2XB(T2) Computed (AX-B) for Test Group 2 (Yes if > 0);
 PARAMETER SUXEST11 Positive (G1 data in G1);
 PARAMETER SUXEST21 Negative (G2 data in G1);
 PARAMETER SUXEST12 Negative (G1 data in G2);
 PARAMETER SUXEST22 Positive (G2 data in G2);
 PARAMETER TOTALT11 Total Positive (G1 data in G1);
 PARAMETER TOTALT21 Total Negative (G2 data in G1);
 PARAMETER TOTALT12 Total Negative (G1 data in G2);
 PARAMETER TOTALT22 Total Positive (G2 data in G2);

KOUNTG1 = 0; KOUNTG2 = 0; CUMULG1 = 0; CUMULG2 = 0;
 A1XB(T1) = 0; A2XB(T2) = 0;
 GR1SH1(G1) = 0; GR2SH1(G2) = 0; GR1SH2(G1) = 0; GR2SH2(G2) = 0;
 GR1SH3(G1) = 0; GR2SH3(G2) = 0; GR1SH4(G1) = 0; GR2SH4(G2) = 0;
 GR1SH5(G1) = 0; GR2SH5(G2) = 0; GR1SH6(G1) = 0; GR2SH6(G2) = 0;
 FLAGT1(T1) = 0; FLAGT2(T2) = 0;
 SUXEST11 = 0; SUXEST21 = 0; SUXEST12 = 0; SUXEST22 = 0;
 TOTALT11 = 0; TOTALT21 = 0; TOTALT12 = 0; TOTALT22 = 0;

TABLE A1(G1, J) Observations for Group 1

	EC-TL	NL-TA	TL-TA	OR-TA	NI-TA	NE-TA	LA-TA	BD-TA	DEA
G1R1	30511	45703	4712	1434	34	22589	4965	2362	3932
G1R2	22806	41844	5642	2516	34	16067	2453	2052	3654
G1R3	22337	39231	3814	1604	21	22424	2453	2052	3654
G1R4	35549	71685	6648	1538	28	31881	3977	2310	3932
G1R5	29425	47056	4606	1360	35	28490	5948	1858	3199
G1R6	15744	32179	3258	914	18	17961	3096	1631	3336
G1R7	96407	137262	15378	6780	150	52166	8152	2015	3870
G1R8	35274	82076	7620	1610	37	41223	4575	1983	3417
G1R9	39467	65989	6284	2078	51	31792	4921	1784	3798
G1R10	12336	25885	2532	692	14	16262	954	1518	3560
G1R11	19903	43445	4518	1438	28	17879	4630	1292	3398
G1R12	67973	93141	8942	2808	77	27263	3977	2310	3932
G1R13	17327	32235	3222	714	16	16662	2285	1329	3561
G1R14	37240	58934	6416	3226	34	16332	8190	2107	4344
G1R15	16882	29276	2814	740	16	15947	1152	1677	3543
G1R16	4202	6590	806	428	9	4083	1500	2335	3783
G1R17	22291	86824	8008	1728	22	45529	2285	1329	3561
G1R18	432641	535792	55600	20766	345	191727	6305	2924	3839
G1R19	7301	9755	1066	352	6	5328	5273	2003	3472
G1R20	3068	7046	742	182	2	3634	5273	2003	3472
G1R21	16365	21382	2176	822	21	12396	5273	2003	3472
G1R22	38277	48737	4896	1098	18	37355	3980	1708	3779
G1R23	6812	15242	1414	550	10	8086	3980	1708	3779
G1R24	50939	64787	7960	2578	60	37427	8190	2107	4344
G1R25	20692	33655	3124	794	16	20935	4284	1694	3532
G1R26	34852	45676	4808	1358	32	28424	7617	1891	3054
G1R27	18082	35290	3604	1328	15	17448	6305	2924	3839
G1R28	26212	90423	7654	2100	65	50875	3046	1789	3421
G1R29	10687	21842	2142	672	15	9841	7391	2179	3476
G1R30	5618	12721	1592	1010	8	4146	7391	2179	3476
G1R31	53681	73302	7642	2110	28	29119	5273	2003	3472
G1R32	40577	69478	7342	2524	27	35148	2285	1329	3561
G1R33	5402	11178	1138	454	11	6020	2285	1329	3561
G1R34	6598	9135	1016	636	8	1876	4261	4391	3750
G1R35	15226	23652	2390	792	21	10128	954	1518	3560
G1R36	33762	40577	4480	2206	49	12163	12797	3438	3851
G1R37	32530	44458	4752	1600	30	19679	2453	2052	3654
G1R38	34211	47411	4618	1540	41	27207	4921	1784	3798
G1R39	7865	17068	2050	1492	24	7329	2453	2052	3654
G1R40	3586	12060	1016	254	6	4681	3977	2310	3932
G1R41	69172	88715	9318	2514	59	45457	3046	1789	3421
G1R42	15125	23975	2530	790	17	11329	3977	2310	3932
G1R43	11446	43139	3652	890	18	8600	2453	2052	3654
G1R44	19177	45936	3876	822	18	21864	4630	1292	3398
G1R45	9699	16698	1678	384	7	9445	7391	2179	3476
G1R46	38343	53766	5766	1920	44	28576	4965	2362	3932
G1R47	42572	95302	9614	3700	83	40943	1500	2335	3783
G1R48	13492	30735	2882	502	14	13289	5273	2003	3472

G1R49	15011	38962	4246	1762	42	14711	4898	1611	3413
G1R50	21278	34342	3320	1044	25	20688	10877	2863	3755
G1R51	50493	87732	8532	2770	57	39151	4921	1784	3798
G1R52	9670	20814	1778	638	9	13995	7391	2179	3476
G1R53	5827	21348	2058	362	9	10476	2285	1329	3561
G1R54	54418	93586	8834	2558	58	51170	2482	1413	3349
G1R55	43882	75038	7292	2060	50	35365	4284	1694	3532
G1R56	454243	479846	55816	18462	417	209116	8686	3593	3276
G1R57	7858	23369	2138	488	12	10927	3977	2310	3932
G1R58	94541	122249	12766	3490	57	50132	6305	2924	3839
G1R59	47339	87515	9180	3370	93	47673	2319	1462	3249
G1R60	34828	47677	5214	1626	39	22895	4630	1292	3398
G1R61	18438	30411	3562	1852	28	11080	4261	4391	3750
G1R62	3763	8102	864	338	8	5205	3980	1708	3779
G1R63	17163	23761	3040	1078	17	19087	4261	4391	3750
G1R64	5437	10265	1082	262	5	5561	3977	2310	3932
G1R65	5280	29849	2674	526	14	17620	1500	2335	3783
G1R66	36420	215777	18322	5858	131	87408	13978	6435	4010
G1R67	882	2547	282	208	8	484	2453	2052	3654
G1R68	20527	41821	4026	1604	33	22508	5273	2003	3472
G1R69	62856	124105	10656	3466	55	33949	7046	3756	4344
G1R70	14093	18408	2264	430	11	12924	3977	2310	3932
G1R71	6111	11091	1068	276	7	5476	3977	2310	3932
G1R72	16658	41595	3874	836	20	23769	2285	1329	3561
G1R73	11530	20088	2006	718	13	8965	3977	2310	3932
G1R74	10813	34378	3206	836	16	18085	3307	1628	3491
G1R75	29327	60695	6352	1750	29	34236	1500	2335	3783
G1R76	4787	9245	1060	472	11	4031	2482	1413	3349
G1R77	12754	20624	2240	958	17	12264	1546	1673	3317
G1R78	41980	60299	7270	2876	79	14960	4546	2479	3495
G1R79	29397	45452	5318	2432	49	20870	8190	2107	4344;

TABLE A2(G2,J) Observations for Group 2

	EC-TL	NL-TA	TL-TA	OR-TA	NI-TA	NE-TA	LA-TA	BD-TA	DEA
G2R1	6615	16056	2110	1364	21	7333	2453	2052	3654
G2R2	23998	31276	3106	934	11	23317	10558	3422	3703
G2R3	5759	8960	1398	842	12	3353	2453	2052	3654
G2R4	54304	62654	7900	2908	63	42176	2453	2052	3654
G2R5	166243	238004	28808	23904	207	195780	2453	2052	3654
G2R6	11605	15798	1986	1062	29	8787	2453	2052	3654
G2R7	31159	109281	10420	4508	70	68535	2453	2052	3654
G2R8	15624	25033	2702	988	16	15071	1500	2335	3783
G2R9	20723	39960	4126	1892	30	21008	2453	2052	3654
G2R10	18201	45339	4658	1606	18	32340	2453	2052	3654
G2R11	59726	130751	11758	2938	52	96799	2453	2052	3654
G2R12	52617	92225	9110	2684	49	54263	2453	2052	3654
G2R13	32107	43940	4890	1992	46	37509	1530	2284	3556
G2R14	40179	75329	7582	2888	52	56124	2453	2052	3654
G2R15	97876	140222	16078	6884	92	94377	10558	3422	3703
G2R16	9353	34059	3104	1444	15	28928	2453	2052	3654
G2R17	13107	20127	2562	1394	20	18867	2453	2052	3654
G2R18	14434	21106	2482	986	17	11784	2453	2052	3654
G2R19	8398	13858	1572	862	15	10478	1530	2284	3556
G2R20	6210	9543	1266	676	11	7476	13978	6435	4010
G2R21	59152	123864	11398	3400	51	81983	2453	2052	3654
G2R22	103561	220605	21436	6770	112	148553	2453	2052	3654
G2R23	8001	14485	1626	616	15	9667	1530	2284	3556
G2R24	17674	25360	2712	950	19	18426	2453	2052	3654
G2R25	6971	9815	1236	850	14	6777	1500	2335	3783
G2R26	57715	74695	9918	5378	81	34144	2453	2052	3654
G2R27	20284	46825	4456	2590	22	29724	2453	2052	3654
G2R28	3620	4572	578	524	5	3646	5273	2003	3472
G2R29	11588	15808	2492	1670	26	9134	10877	2863	3755
G2R30	15086	35297	7942	2754	60	18460	2453	2052	3654
G2R31	8875	14131	1452	592	12	7716	7391	2179	3476
G2R32	13457	25143	3000	1370	34	19189	1530	2284	3556
G2R33	18642	26671	2822	1264	14	18768	1530	2284	3556
G2R34	14079	35043	3622	1218	20	27950	2453	2052	3654
G2R35	103755	178264	17848	3662	63	108102	2453	2052	3654
G2R36	16602	22685	2990	1432	23	13392	2453	2052	3654

G2R37	2451	3614	616	426	6	1090	4261	4391	3750
G2R38	16664	25400	2994	1662	17	20709	1500	2335	3783
G2R39	4790	7051	1000	550	8	3843	4261	4391	3750
G2R40	10866	15168	1526	742	13	8900	2453	2052	3654
G2R41	14676	40282	3746	1204	14	28529	2453	2052	3654
G2R42	8558	13811	1676	938	13	8841	2453	2052	3654
G2R43	20608	27728	3802	1300	17	43225	2453	2052	3654
G2R44	17994	34273	4136	1702	35	16354	1530	2284	3556
G2R45	11491	14453	1808	554	12	8824	2453	2052	3654
G2R46	23585	33288	3972	1824	28	21354	2453	2052	3654
G2R47	17274	27764	3522	2052	28	19979	2453	2052	3654
G2R48	29210	35483	4336	2020	33	21303	2453	2052	3654
G2R49	29928	36749	4136	2364	32	23225	2453	2052	3654
G2R50	131400	168544	18842	12532	121	126518	2453	2052	3654
G2R51	86001	143188	14510	4396	63	84256	2453	2052	3654
G2R52	53555	67289	8136	4588	53	40005	2453	2052	3654
G2R53	17207	22789	2742	1496	22	14357	2453	2052	3654
G2R54	26790	50915	5504	2624	30	37556	2453	2052	3654
G2R55	32767	57654	5668	1526	28	38267	2453	2052	3654
G2R56	27860	31998	4496	1972	27	22810	2453	2052	3654
G2R57	7548	9496	1218	846	14	6692	2453	2052	3654
G2R58	46297	54372	7996	2874	32	53848	3109	4261	4124
G2R59	39073	53559	6370	2616	48	33908	10558	3422	3703
G2R60	29563	36867	3798	1422	16	28893	2453	2052	3654
G2R61	5307	8096	1082	686	14	4069	2453	2052	3654
G2R62	13907	18536	2500	1734	25	15556	2453	2052	3654
G2R63	27471	29333	3968	2590	31	17202	2453	2052	3654
G2R64	10318	15956	2088	1372	16	6860	2453	2052	3654
G2R65	46337	74262	8404	3118	30	55949	2453	2052	3654
G2R66	19917	32589	3064	2036	30	16419	2453	2052	3654
G2R67	46618	54257	6358	2326	47	27437	1530	2284	3556
G2R68	229674	329978	37428	42546	230	203179	2453	2052	3654
G2R69	25389	30698	4594	2488	44	14014	4261	4391	3750
G2R70	16047	20063	2270	828	18	12357	2924	1441	3376
G2R71	6669	9431	1112	602	10	5153	2453	2052	3654
G2R72	6842	9462	1362	830	14	5834	2453	2052	3654
G2R73	118419	201431	22712	7858	150	119483	1500	2335	3783
G2R74	2092	4594	442	128	4	2912	3980	1708	3779
G2R75	22035	30859	3662	1360	22	9580	4261	4391	3750
G2R76	33610	66234	6760	2168	25	48655	2453	2052	3654
G2R77	21725	24030	3122	1384	21	15489	2453	2052	3654
G2R78	20073	40164	3756	2526	19	29042	2453	2052	3654
G2R79	9815	17844	2186	1004	12	13234	1500	2335	3783
G2R80	23511	29913	3536	1360	26	18996	2453	2052	3654
G2R81	17680	23001	2446	754	12	19352	2453	2052	3654
G2R82	7889	12374	1774	1006	22	10264	1530	2284	3556
G2R83	17915	21590	2640	1258	16	10103	10558	3422	3703
G2R84	16926	24590	2874	768	12	15660	2453	2052	3654
G2R85	80924	204129	19168	3560	54	148425	2453	2052	3654
G2R86	11705	20860	2514	2588	41	18664	2453	2052	3654
G2R87	9296	13546	1636	1018	17	7819	2453	2052	3654
G2R88	6180	7438	994	570	9	3591	2453	2052	3654
G2R89	15551	23622	2760	1680	25	16747	3109	4261	4124
G2R90	9451	14909	1500	528	10	11093	2924	1441	3376;

TABLE TEST1(T1,J) Test data for Group 1

	EC-TL	NL-TA	TL-TA	OR-TA	NI-TA	NE-TA	LA-TA	BD-TA	DEA
T1R1	2082	10092	982	304	6	6299	2453	2052	3654
T1R2	25093	47523	4900	1426	24	27666	2453	2052	3654
T1R3	28368	60999	5302	1206	29	33596	5273	2003	3472
T1R4	44964	153383	15366	7738	139	62842	1500	2335	3783
T1R5	11091	26258	2448	668	13	12609	5273	2003	3472
T1R6	509837	561216	62936	19760	375	280365	13978	6435	4010
T1R7	18398	30780	3370	1198	25	14615	5948	1858	3199
T1R8	5912	12515	1262	734	14	7225	1500	2335	3783
T1R9	29190	42846	4030	1078	24	24807	5273	2003	3472;

TABLE TEST2(T2,J) Test data for Group 2

	EC-TL	NL-TA	TL-TA	OR-TA	NI-TA	NE-TA	LA-TA	BD-TA	DEA
T2R1	16731	25758	3208	1372	26	12994	1500	2335	3783
T2R2	48179	69099	8036	4070	84	30395	1530	2284	3556
T2R3	2612	4631	626	244	5	3335	3980	1708	3779
T2R4	16292	32458	3208	1440	28	21100	1152	1677	3543
T2R5	13091	27596	2938	1388	18	21200	1530	2284	3556
T2R6	54634	75809	7476	2668	45	57780	2453	2052	3654
T2R7	152015	176896	19464	7750	83	132101	2453	2052	3654
T2R8	3265	4474	566	254	6	2638	3307	1628	3491
T2R9	65092	93531	10956	5108	85	37880	2453	2052	3654
T2R10	11015	15494	2308	1362	24	8440	2453	2052	3654

FREE VARIABLES

Z Objective function value
 X(J) Computed Attributes
 B Computed RHS (Ax=B);

POSITIVE VARIABLES

ALPHA0 Maximum external deviation
 ALPHA1(G1) External deviations for Group 1 observations
 ALPHA2(G2) External deviations for Group 2 observations
 BETA0 Minimum internal deviation
 BETA1(G1) Internal deviations for Group 1 observations
 BETA2(G2) Internal deviations for Group 2 observations;

EQUATIONS

OBJECTIVE Weighted sum of deviations
 GROUP1(G1) Goal programming constraints for Group 1 observations
 GROUP2(G2) Goal programming constraints for Group 2 observations
 NORMLN N Normalization constraint
 NORMLNS N* Normalization constraint;

OBJECTIVE.. Z =E= HO*ALPHA0 + SUM(G1, ALPHA1(G1)*H1(G1))
 + SUM(G2, ALPHA2(G2)*H2(G2))
 -KO*BETA0 - SUM(G1, BETA1(G1) *K1(G1))
 - SUM(G2, BETA2(G2) *K2(G2));
 GROUP1(G1).. SUM(J, A1(G1,J)*X(J)) - ALPHA0 - ALPHA1(G1) + BETA0 + BETA1(G1)
 =E= B - SEP;
 GROUP2(G2).. SUM(J, A2(G2,J)*X(J)) + ALPHA0 + ALPHA2(G2) - BETA0 - BETA2(G2)
 =E= B + SEP;
 NORMLN.. -N2*SUM((G1,J), A1(G1,J)*X(J)) + N1*SUM((G2,J), A2(G2,J)*X(J))
 =E= 2*N1*N2;
 NORMLNS.. 2*N1*N2*(BETA0 - ALPHA0)
 + N2 * SUM(G1, BETA1(G1) - ALPHA1(G1))
 + N1 * SUM(G2, BETA2(G2) - ALPHA2(G2)) =E= 2*N1*N2;
 MODEL UNNORM /OBJECTIVE, GROUP1, GROUP2/;
 MODEL NORMV1 /OBJECTIVE, GROUP1, GROUP2, NORMLN/;
 MODEL NORMV2 /OBJECTIVE, GROUP1, GROUP2, NORMLNS/;

DISPLAY " ===== 1st Stage =====";
 SOLVE NORMV2 USING LP MINIMIZING Z;
 G1MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G1, ALPHA1.L(G1));
 G2MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G2, ALPHA2.L(G2));
 GR1SH1(G1) \$ ((BETA1.L(G1) GE G2MAX)) = 101;
 GR2SH1(G2) \$ ((BETA2.L(G2) GE G1MAX)) = 201;
 KOUNTG1 = SUM(G1, 1 \$ (GR1SH1(G1) EQ 101));
 KOUNTG2 = SUM(G2, 1 \$ (GR2SH1(G2) EQ 201));
 CUMULG1 = CUMULG1 + KOUNTG1;
 CUMULG2 = CUMULG2 + KOUNTG2;
 DISPLAY X.L, Z.L, B.L, G1MAX, G2MAX, KOUNTG1, KOUNTG2, CUMULG1, CUMULG2, N1, N2;
 *DISPLAY ALPHA0.L, BETA0.L, ALPHA1.L, BETA1.L, ALPHA2.L, BETA2.L;

```

A1XB(T1) = (SUM(J, TEST1(T1,J)*X.L(J)) - B.L);
A2XB(T2) = (SUM(J, TEST2(T2,J)*X.L(J)) - B.L);
FLAGT1(T1) $ ((A1XB(T1) LE 0) AND (ABS(A1XB(T1)) GE G2MAX)) = 111;
FLAGT1(T1) $ ((A1XB(T1) GT 0) AND (A1XB(T1) GT G1MAX)) = 112;
FLAGT2(T2) $ ((A2XB(T2) LE 0) AND (ABS(A2XB(T2)) GE G2MAX)) = 211;
FLAGT2(T2) $ ((A2XB(T2) GT 0) AND (A2XB(T2) GT G1MAX)) = 212;
SUXEST11 = SUM(T1, 1 $ (FLAGT1(T1) EQ 111));
SUXEST12 = SUM(T1, 1 $ (FLAGT1(T1) EQ 112));
SUXEST21 = SUM(T2, 1 $ (FLAGT2(T2) EQ 211));
SUXEST22 = SUM(T2, 1 $ (FLAGT2(T2) EQ 212));
TOTALT11 = TOTALT11 + SUXEST11;
TOTALT12 = TOTALT12 + SUXEST12;
TOTALT21 = TOTALT21 + SUXEST21;
TOTALT22 = TOTALT22 + SUXEST22;
DISPLAY SUXEST11, SUXEST12, SUXEST21, SUXEST22;
DISPLAY TOTALT11, TOTALT12, TOTALT21, TOTALT22, FLAGT1, FLAGT2;

DISPLAY " ===== 2nd Stage =====";
H1(G1) $ ((GR1SH1(G1) EQ 101)) = WH*RS2;
H2(G2) $ ((GR2SH1(G2) EQ 201)) = WH*RS2;
K1(G1) $ ((GR1SH1(G1) EQ 101)) = WK*RS2;
K2(G2) $ ((GR2SH1(G2) EQ 201)) = WK*RS2;
K0 = SUM(G1, K1(G1)) + SUM(G2, K2(G2)) + 1;
H0 = 3*K0;
*DISPLAY H0, K0, H1, K1, H2, K2;

SOLVE NORMV2 USING LP MINIMIZING Z;
ALPHA1.L(G1) $ ((GR1SH1(G1) EQ 101)) = 0.0 ;
ALPHA2.L(G2) $ ((GR2SH1(G2) EQ 201)) = 0.0 ;
G1MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G1, ALPHA1.L(G1));
G2MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G2, ALPHA2.L(G2));
GR1SH2(G1) $ ((BETA1.L(G1) GE G2MAX) AND (GR1SH1(G1) NE 101)) = 102;
GR2SH2(G2) $ ((BETA2.L(G2) GE G1MAX) AND (GR2SH1(G2) NE 201)) = 202;
KOUNTG1 = SUM(G1, 1 $ (GR1SH2(G1) EQ 102));
KOUNTG2 = SUM(G2, 1 $ (GR2SH2(G2) EQ 202));
CUMULG1 = CUMULG1 + KOUNTG1;
CUMULG2 = CUMULG2 + KOUNTG2;
DISPLAY X.L, Z.L, B.L, G1MAX, G2MAX, KOUNTG1, KOUNTG2, CUMULG1, CUMULG2, N1, N2;
*DISPLAY ALPHA0.L, BETA0.L, ALPHA1.L, BETA1.L, ALPHA2.L, BETA2.L;

A1XB(T1) = (SUM(J, TEST1(T1,J)*X.L(J)) - B.L);
A2XB(T2) = (SUM(J, TEST2(T2,J)*X.L(J)) - B.L);
FLAGT1(T1) $ ((A1XB(T1) LE 0) AND (ABS(A1XB(T1)) GE G2MAX) AND (FLAGT1(T1) EQ 0)) = 121;
FLAGT1(T1) $ ((A1XB(T1) GT 0) AND (A1XB(T1) GT G1MAX) AND (FLAGT1(T1) EQ 0)) = 122;
FLAGT2(T2) $ ((A2XB(T2) LE 0) AND (ABS(A2XB(T2)) GE G2MAX) AND (FLAGT2(T2) EQ 0)) = 221;
FLAGT2(T2) $ ((A2XB(T2) GT 0) AND (A2XB(T2) GT G1MAX) AND (FLAGT2(T2) EQ 0)) = 222;
SUXEST11 = SUM(T1, 1 $ (FLAGT1(T1) EQ 121));
SUXEST12 = SUM(T1, 1 $ (FLAGT1(T1) EQ 122));
SUXEST21 = SUM(T2, 1 $ (FLAGT2(T2) EQ 221));
SUXEST22 = SUM(T2, 1 $ (FLAGT2(T2) EQ 222));
TOTALT11 = TOTALT11 + SUXEST11;
TOTALT12 = TOTALT12 + SUXEST12;
TOTALT21 = TOTALT21 + SUXEST21;
TOTALT22 = TOTALT22 + SUXEST22;
DISPLAY SUXEST11, SUXEST12, SUXEST21, SUXEST22;
DISPLAY TOTALT11, TOTALT12, TOTALT21, TOTALT22, FLAGT1, FLAGT2;

DISPLAY " ===== 3rd Stage =====";
H1(G1) $ ((GR1SH1(G1) EQ 101)) = WH*RS3;
H2(G2) $ ((GR2SH1(G2) EQ 201)) = WH*RS3;
K1(G1) $ ((GR1SH1(G1) EQ 101)) = WK*RS3;
K2(G2) $ ((GR2SH1(G2) EQ 201)) = WK*RS3;

H1(G1) $ ((GR1SH2(G1) EQ 102)) = WH*RS2;
H2(G2) $ ((GR2SH2(G2) EQ 202)) = WH*RS2;
K1(G1) $ ((GR1SH2(G1) EQ 102)) = WK*RS2;
K2(G2) $ ((GR2SH2(G2) EQ 202)) = WK*RS2;
K0 = SUM(G1, K1(G1)) + SUM(G2, K2(G2)) + 1 ;
H0 = 3*K0;
*DISPLAY H0, K0, H1, K1, H2, K2;

```

```

SOLVE NORMV2 USING LP MINIMIZING Z;
ALPHA1.L(G1) $ ((GR1SH2(G1) EQ 102) OR (GR1SH1(G1) EQ 101)) = 0.0;
ALPHA2.L(G2) $ ((GR2SH2(G2) EQ 202) OR (GR2SH1(G2) EQ 201)) = 0.0;
G1MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G1, ALPHA1.L(G1));
G2MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G2, ALPHA2.L(G2));
GR1SH3(G1) $ ((BETA1.L(G1) GE G2MAX) AND ((GR1SH2(G1)+GR1SH1(G1)) EQ 0))=103;
GR2SH3(G2) $ ((BETA2.L(G2) GE G1MAX) AND ((GR2SH2(G2)+GR2SH1(G2)) EQ 0))=203;
KOUNTG1 = SUM(G1, 1 $ (GR1SH3(G1) EQ 103));
KOUNTG2 = SUM(G2, 1 $ (GR2SH3(G2) EQ 203));
CUMULG1 = CUMULG1 + KOUNTG1;
CUMULG2 = CUMULG2 + KOUNTG2;
DISPLAY X.L, Z.L, B.L, G1MAX, G2MAX, KOUNTG1, KOUNTG2, CUMULG1, CUMULG2, N1, N2;
*DISPLAY ALPHA0.L, BETA0.L, ALPHA1.L, BETA1.L, ALPHA2.L, BETA2.L;

A1XB(T1) = (SUM(J, TEST1(T1, J)*X.L(J)) - B.L);
A2XB(T2) = (SUM(J, TEST2(T2, J)*X.L(J)) - B.L);
FLAGT1(T1) $ ((A1XB(T1) LE 0) AND (ABS(A1XB(T1)) GE G2MAX) AND (FLAGT1(T1) EQ 0)) = 131;
FLAGT1(T1) $ ((A1XB(T1) GT 0) AND (A1XB(T1) GT G1MAX) AND (FLAGT1(T1) EQ 0)) = 132;
FLAGT2(T2) $ ((A2XB(T2) LE 0) AND (ABS(A2XB(T2)) GE G2MAX) AND (FLAGT2(T2) EQ 0)) = 231;
FLAGT2(T2) $ ((A2XB(T2) GT 0) AND (A2XB(T2) GT G1MAX) AND (FLAGT2(T2) EQ 0)) = 232;
SUXEST11 = SUM(T1, 1 $ (FLAGT1(T1) EQ 131));
SUXEST12 = SUM(T1, 1 $ (FLAGT1(T1) EQ 132));
SUXEST21 = SUM(T2, 1 $ (FLAGT2(T2) EQ 231));
SUXEST22 = SUM(T2, 1 $ (FLAGT2(T2) EQ 232));
TOTALT11 = TOTALT11 + SUXEST11;
TOTALT12 = TOTALT12 + SUXEST12;
TOTALT21 = TOTALT21 + SUXEST21;
TOTALT22 = TOTALT22 + SUXEST22;
DISPLAY SUXEST11, SUXEST12, SUXEST21, SUXEST22;
DISPLAY TOTALT11, TOTALT12, TOTALT21, TOTALT22, FLAGT1, FLAGT2;

DISPLAY " ===== 4th Stage =====";
H1(G1) $ ((GR1SH1(G1) EQ 101)) = WH*RS4;
H2(G2) $ ((GR2SH1(G2) EQ 201)) = WH*RS4;
K1(G1) $ ((GR1SH1(G1) EQ 101)) = WK*RS4;
K2(G2) $ ((GR2SH1(G2) EQ 201)) = WK*RS4;

H1(G1) $ ((GR1SH2(G1) EQ 102)) = WH*RS3;
H2(G2) $ ((GR2SH2(G2) EQ 202)) = WH*RS3;
K1(G1) $ ((GR1SH2(G1) EQ 102)) = WK*RS3;
K2(G2) $ ((GR2SH2(G2) EQ 202)) = WK*RS3;

H1(G1) $ ((GR1SH2(G1) EQ 103)) = WH*RS2;
H2(G2) $ ((GR2SH2(G2) EQ 203)) = WH*RS2;
K1(G1) $ ((GR1SH2(G1) EQ 103)) = WK*RS2;
K2(G2) $ ((GR2SH2(G2) EQ 203)) = WK*RS2;

K0 = SUM(G1, K1(G1)) + SUM(G2, K2(G2)) + 1 ;
H0 = 3*K0;
*DISPLAY H0, K0, H1, K1, H2, K2;

SOLVE NORMV2 USING LP MINIMIZING Z;
ALPHA1.L(G1) $ ((GR1SH3(G1) EQ 103) OR (GR1SH2(G1) EQ 102)
OR (GR1SH1(G1) EQ 101)) = 0.0;
ALPHA2.L(G2) $ ((GR2SH3(G2) EQ 203) OR (GR2SH2(G2) EQ 202)
OR (GR2SH1(G2) EQ 201)) = 0.0;
G1MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G1, ALPHA1.L(G1));
G2MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G2, ALPHA2.L(G2));
GR1SH4(G1) $ ((BETA1.L(G1) GE G2MAX) AND ((GR1SH3(G1)+GR1SH2(G1)+GR1SH1(G1)) EQ 0))=104;
GR2SH4(G2) $ ((BETA2.L(G2) GE G1MAX) AND ((GR2SH3(G2)+GR2SH2(G2)+GR2SH1(G2)) EQ 0))=204;
KOUNTG1 = SUM(G1, 1 $ (GR1SH4(G1) EQ 104));
KOUNTG2 = SUM(G2, 1 $ (GR2SH4(G2) EQ 204));
CUMULG1 = CUMULG1 + KOUNTG1;
CUMULG2 = CUMULG2 + KOUNTG2;
DISPLAY X.L, Z.L, B.L, G1MAX, G2MAX, KOUNTG1, KOUNTG2, CUMULG1, CUMULG2, N1, N2;
*DISPLAY ALPHA0.L, BETA0.L, ALPHA1.L, BETA1.L, ALPHA2.L, BETA2.L;

```

```

A1XB(T1) = (SUM(J, TEST1(T1,J)*X.L(J)) - B.L);
A2XB(T2) = (SUM(J, TEST2(T2,J)*X.L(J)) - B.L);
FLAGT1(T1) $ ((A1XB(T1) LE 0) AND (ABS(A1XB(T1)) GE G2MAX) AND (FLAGT1(T1) EQ 0)) = 141;
FLAGT1(T1) $ ((A1XB(T1) GT 0) AND (A1XB(T1) GT G1MAX) AND (FLAGT1(T1) EQ 0)) = 142;
FLAGT2(T2) $ ((A2XB(T2) LE 0) AND (ABS(A2XB(T2)) GE G2MAX) AND (FLAGT2(T2) EQ 0)) = 241;
FLAGT2(T2) $ ((A2XB(T2) GT 0) AND (A2XB(T2) GT G1MAX) AND (FLAGT2(T2) EQ 0)) = 242;
SUXEST11 = SUM(T1, 1 $ (FLAGT1(T1) EQ 141));
SUXEST12 = SUM(T1, 1 $ (FLAGT1(T1) EQ 142));
SUXEST21 = SUM(T2, 1 $ (FLAGT2(T2) EQ 241));
SUXEST22 = SUM(T2, 1 $ (FLAGT2(T2) EQ 242));
TOTALT11 = TOTALT11 + SUXEST11;
TOTALT12 = TOTALT12 + SUXEST12;
TOTALT21 = TOTALT21 + SUXEST21;
TOTALT22 = TOTALT22 + SUXEST22;
DISPLAY SUXEST11, SUXEST12, SUXEST21, SUXEST22;
DISPLAY TOTALT11, TOTALT12, TOTALT21, TOTALT22, FLAGT1, FLAGT2;
DISPLAY " ===== 5th Stage =====";
H1(G1) $ ((GR1SH1(G1) EQ 101)) = WH*RS5;
H2(G2) $ ((GR2SH1(G2) EQ 201)) = WH*RS5;
K1(G1) $ ((GR1SH1(G1) EQ 101)) = WK*RS5;
K2(G2) $ ((GR2SH1(G2) EQ 201)) = WK*RS5;

H1(G1) $ ((GR1SH2(G1) EQ 102)) = WH*RS4;
H2(G2) $ ((GR2SH2(G2) EQ 202)) = WH*RS4;
K1(G1) $ ((GR1SH2(G1) EQ 102)) = WK*RS4;
K2(G2) $ ((GR2SH2(G2) EQ 202)) = WK*RS4;

H1(G1) $ ((GR1SH2(G1) EQ 103)) = WH*RS3;
H2(G2) $ ((GR2SH2(G2) EQ 203)) = WH*RS3;
K1(G1) $ ((GR1SH2(G1) EQ 103)) = WK*RS3;
K2(G2) $ ((GR2SH2(G2) EQ 203)) = WK*RS3;

H1(G1) $ ((GR1SH2(G1) EQ 104)) = WH*RS2;
H2(G2) $ ((GR2SH2(G2) EQ 204)) = WH*RS2;
K1(G1) $ ((GR1SH2(G1) EQ 104)) = WK*RS2;
K2(G2) $ ((GR2SH2(G2) EQ 204)) = WK*RS2;

K0 = SUM(G1, K1(G1)) + SUM(G2, K2(G2)) + 1 ;
H0 = 3*K0;
*DISPLAY H0, K0, H1, K1, H2, K2;

SOLVE NORMV2 USING LP MINIMIZING Z;
ALPHA1.L(G1) $ ((GR1SH4(G1) EQ 104) OR (GR1SH3(G1) EQ 103)
OR (GR1SH2(G1) EQ 102) OR (GR1SH1(G1) EQ 101)) = 0.0;
ALPHA2.L(G2) $ ((GR2SH4(G2) EQ 204) OR (GR2SH3(G2) EQ 203)
OR (GR2SH2(G2) EQ 202) OR (GR2SH1(G2) EQ 201)) = 0.0;
G1MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G1, ALPHA1.L(G1));
G2MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G2, ALPHA2.L(G2));
GR1SH5(G1) $ ((BETA1.L(G1) GE G2MAX) AND
((GR1SH4(G1)+GR1SH3(G1)+GR1SH2(G1)+GR1SH1(G1)) EQ 0))=105;
GR2SH5(G2) $ ((BETA2.L(G2) GE G1MAX) AND
((GR2SH4(G2)+GR2SH3(G2)+GR2SH2(G2)+GR2SH1(G2)) EQ 0))=205;
KOUNTG1= SUM(G1, 1 $ (GR1SH5(G1) EQ 105));
KOUNTG2= SUM(G2, 1 $ (GR2SH5(G2) EQ 205));
CUMULG1 = CUMULG1 + KOUNTG1;
CUMULG2 = CUMULG2 + KOUNTG2;
DISPLAY X.L, Z.L, B.L, G1MAX, G2MAX, KOUNTG1, KOUNTG2, CUMULG1, CUMULG2, N1, N2;
*DISPLAY ALPHA0.L, BETA0.L, ALPHA1.L, BETA1.L, ALPHA2.L, BETA2.L;

A1XB(T1) = (SUM(J, TEST1(T1,J)*X.L(J)) - B.L);
A2XB(T2) = (SUM(J, TEST2(T2,J)*X.L(J)) - B.L);
FLAGT1(T1) $ ((A1XB(T1) LE 0) AND (ABS(A1XB(T1)) GE G2MAX) AND (FLAGT1(T1) EQ 0)) = 151;
FLAGT1(T1) $ ((A1XB(T1) GT 0) AND (A1XB(T1) GT G1MAX) AND (FLAGT1(T1) EQ 0)) = 152;
FLAGT2(T2) $ ((A2XB(T2) LE 0) AND (ABS(A2XB(T2)) GE G2MAX) AND (FLAGT2(T2) EQ 0)) = 251;
FLAGT2(T2) $ ((A2XB(T2) GT 0) AND (A2XB(T2) GT G1MAX) AND (FLAGT2(T2) EQ 0)) = 252;
SUXEST11 = SUM(T1, 1 $ (FLAGT1(T1) EQ 151));
SUXEST12 = SUM(T1, 1 $ (FLAGT1(T1) EQ 152));
SUXEST21 = SUM(T2, 1 $ (FLAGT2(T2) EQ 251));
SUXEST22 = SUM(T2, 1 $ (FLAGT2(T2) EQ 252));
TOTALT11 = TOTALT11 + SUXEST11;

```

```

TOTALT12 = TOTALT12 + SUXEST12;
TOTALT21 = TOTALT21 + SUXEST21;
TOTALT22 = TOTALT22 + SUXEST22;
DISPLAY SUXEST11, SUXEST12, SUXEST21, SUXEST22;
DISPLAY TOTALT11, TOTALT12, TOTALT21, TOTALT22, FLAGT1, FLAGT2;

DISPLAY " ===== 6th Stage =====";
H1(G1) $ ((GR1SH1(G1) EQ 101)) = WH*RS6;
H2(G2) $ ((GR2SH1(G2) EQ 201)) = WH*RS6;
K1(G1) $ ((GR1SH1(G1) EQ 101)) = WK*RS6;
K2(G2) $ ((GR2SH1(G2) EQ 201)) = WK*RS6;

H1(G1) $ ((GR1SH2(G1) EQ 102)) = WH*RS5;
H2(G2) $ ((GR2SH2(G2) EQ 202)) = WH*RS5;
K1(G1) $ ((GR1SH2(G1) EQ 102)) = WK*RS5;
K2(G2) $ ((GR2SH2(G2) EQ 202)) = WK*RS5;

H1(G1) $ ((GR1SH2(G1) EQ 103)) = WH*RS4;
H2(G2) $ ((GR2SH2(G2) EQ 203)) = WH*RS4;
K1(G1) $ ((GR1SH2(G1) EQ 103)) = WK*RS4;
K2(G2) $ ((GR2SH2(G2) EQ 203)) = WK*RS4;

H1(G1) $ ((GR1SH2(G1) EQ 104)) = WH*RS3;
H2(G2) $ ((GR2SH2(G2) EQ 204)) = WH*RS3;
K1(G1) $ ((GR1SH2(G1) EQ 104)) = WK*RS3;
K2(G2) $ ((GR2SH2(G2) EQ 204)) = WK*RS3;

H1(G1) $ ((GR1SH2(G1) EQ 105)) = WH*RS2;
H2(G2) $ ((GR2SH2(G2) EQ 205)) = WH*RS2;
K1(G1) $ ((GR1SH2(G1) EQ 105)) = WK*RS2;
K2(G2) $ ((GR2SH2(G2) EQ 205)) = WK*RS2;

K0      = SUM(G1, K1(G1)) + SUM(G2, K2(G2)) + 1 ;
H0      = 3*K0;
*DISPLAY H0, K0, H1, K1, H2, K2;
SOLVE NORMV2 USING LP MINIMIZING Z;
ALPHA1.L(G1) $ ((GR1SH5(G1) EQ 105) OR (GR1SH4(G1) EQ 104) OR (GR1SH3(G1) EQ 103)
OR (GR1SH2(G1) EQ 102) OR (GR1SH1(G1) EQ 101)) = 0.0;
ALPHA2.L(G2) $ ((GR2SH5(G2) EQ 205) OR (GR2SH4(G2) EQ 204) OR (GR2SH3(G2) EQ 203)
OR (GR2SH2(G2) EQ 202) OR (GR2SH1(G2) EQ 201)) = 0.0;
G1MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G1, ALPHA1.L(G1));
G2MAX = SEP + ALPHA0.L - BETA0.L + SMAX(G2, ALPHA2.L(G2));
GR1SH6(G1) $ ((BETA1.L(G1) GE G2MAX) AND
((GR1SH5(G1)+GR1SH4(G1)+GR1SH3(G1)+GR1SH2(G1)+GR1SH1(G1)) EQ 0))=106;
GR2SH6(G2) $ ((BETA2.L(G2) GE G1MAX) AND
((GR2SH5(G2)+GR2SH4(G2)+GR2SH3(G2)+GR2SH2(G2)+GR2SH1(G2)) EQ 0))=206;
KOUNTG1= SUM(G1, 1 $ (GR1SH6(G1) EQ 106));
KOUNTG2= SUM(G2, 1 $ (GR2SH6(G2) EQ 206));
CUMULG1 = CUMULG1 + KOUNTG1;
CUMULG2 = CUMULG2 + KOUNTG2;
DISPLAY X.L, Z.L, B.L, G1MAX, G2MAX, KOUNTG1, KOUNTG2, CUMULG1, CUMULG2, N1, N2;
*DISPLAY ALPHA0.L, BETA0.L, ALPHA1.L, BETA1.L, ALPHA2.L, BETA2.L;

A1XB(T1) = (SUM(J, TEST1(T1,J)*X.L(J)) - B.L);
A2XB(T2) = (SUM(J, TEST2(T2,J)*X.L(J)) - B.L);
FLAGT1(T1) $ ((A1XB(T1) LE 0) AND (ABS(A1XB(T1)) GE G2MAX) AND (FLAGT1(T1) EQ 0)) = 161;
FLAGT1(T1) $ ((A1XB(T1) GT 0) AND (A1XB(T1) GT G1MAX) AND (FLAGT1(T1) EQ 0)) = 162;
FLAGT2(T2) $ ((A2XB(T2) LE 0) AND (ABS(A2XB(T2)) GE G2MAX) AND (FLAGT2(T2) EQ 0)) = 261;
FLAGT2(T2) $ ((A2XB(T2) GT 0) AND (A2XB(T2) GT G1MAX) AND (FLAGT2(T2) EQ 0)) = 262;
SUXEST11 = SUM(T1, 1 $ (FLAGT1(T1) EQ 161));
SUXEST12 = SUM(T1, 1 $ (FLAGT1(T1) EQ 162));
SUXEST21 = SUM(T2, 1 $ (FLAGT2(T2) EQ 261));
SUXEST22 = SUM(T2, 1 $ (FLAGT2(T2) EQ 262));
TOTALT11 = TOTALT11 + SUXEST11;
TOTALT12 = TOTALT12 + SUXEST12;
TOTALT21 = TOTALT21 + SUXEST21;
TOTALT22 = TOTALT22 + SUXEST22;
DISPLAY SUXEST11, SUXEST12, SUXEST21, SUXEST22;
DISPLAY TOTALT11, TOTALT12, TOTALT21, TOTALT22, FLAGT1, FLAGT2;

```

```
DISPLAY " ===== Final processing =====";
FLAGT1(T1) $ ((A1XB(T1) LE 0) AND (FLAGT1(T1) EQ 0)) = 171;
FLAGT1(T1) $ ((A1XB(T1) GT 0) AND (FLAGT1(T1) EQ 0)) = 172;
FLAGT2(T2) $ ((A2XB(T2) LE 0) AND (FLAGT2(T2) EQ 0)) = 271;
FLAGT2(T2) $ ((A2XB(T2) GT 0) AND (FLAGT2(T2) EQ 0)) = 272;
SUXEST11 = SUM(T1, 1 $ (FLAGT1(T1) EQ 171));
SUXEST12 = SUM(T1, 1 $ (FLAGT1(T1) EQ 172));
SUXEST21 = SUM(T2, 1 $ (FLAGT2(T2) EQ 271));
SUXEST22 = SUM(T2, 1 $ (FLAGT2(T2) EQ 272));
TOTALT11 = TOTALT11 + SUXEST11;
TOTALT12 = TOTALT12 + SUXEST12;
TOTALT21 = TOTALT21 + SUXEST21;
TOTALT22 = TOTALT22 + SUXEST22;
DISPLAY CUMULG1, CUMULG2, N1, N2;
DISPLAY SUXEST11, SUXEST12, SUXEST21, SUXEST22;
DISPLAY TOTALT11, TOTALT12, TOTALT21, TOTALT22, FLAGT1, FLAGT2;
DISPLAY CUMULG1, CUMULG2, N1, N2;
*End of model -----
```

BIBLIOGRAPHY

- Bajgier, S. M., and A. V. Hill, 1982, "An Experimental Comparison of Statistical and Linear Programming Approaches to the Discriminant Problem," *Decision Sciences* 13, 604-18.
- Barr, R. S., L. Seiford, and T. Siems, 1992, "An Envelopment Analysis Approach to Measuring Management Quality and Predicting Bank Failure," to appear in the *Annals of Operations Research*.
- Cavalier, T. M., J. P. Ignizio, and A. L. Soyster, 1989, "Discriminant Analysis via Mathematical Programming: on Certain Problems and their Causes," *Computers and Operations Research* 16, 353-62.
- Freed, N., and F. Glover, 1981a, "A Linear Programming Approach to the Discriminant Problem," *Decision Sciences* 12, 68-74.
- Freed, N., and F. Glover, 1981b, "Simple but Powerful Goal Programming Models for Discriminant Problems," *European Journal of Operations Research* 7, 44-60.
- Freed, N., and F. Glover, 1982, "Linear Programming and Statistical Discrimination—the LP Side," *Decision Sciences* 13, 172-175.
- Freed, N., and F. Glover, 1986a, "Evaluating Alternative Linear Programming Models to Solve the Two-Group Discriminant Problem," *Decision Sciences* 17, 151-62.
- Freed, N., and F. Glover, 1986b, "Resolving Certain Difficulties and Improving the Classification Power of LP Discriminant Analysis Formulations," *Decision Sciences* 17, 589-95.
- Glorfeld, L. W., and N. Gaither, 1982, "On Using Linear Programming in Discriminant Problems," *Decision Sciences* 13, 167-171.
- Glover, F., 1990, "Improved Linear Programming Models for Discriminant Analysis," *Decision Sciences* 21, 771-85.
- Glover, F., S. Keene, and B. Duea, 1988, "A New Class of Models for the Discriminant Problem," *Decision Sciences* 19, 269-80.

- Hamburg, Morris, 1977, *Statistical Analysis for Decision Making*, Harcourt Brace Jovanovich, Inc., New York.
- Hand, D.J., 1981, *Discrimination and Classification*, John Wiley & Sons, New York.
- James, Mike, 1985, *Classification Algorithms*, John Wiley & Sons, New York.
- Joachimsthaler, E. A., and A. Stam, 1988, "Four Approaches to the Classification Problem in Discriminant Analysis: An Experimental Study," *Decision Sciences* 19, 322-33.
- Kendall, M. G., 1966, "Discrimination and Classification," In P. R. Krishnaiah (Ed.), *Multivariate Analysis*, Academic Press, New York.
- Kennington, J. L. and R. V. Helgason, 1980, *Algorithms for Network Programming*, John Wiley & Sons, New York.
- Koehler, G. J., 1989a, "Characterization of Unacceptable Solutions in LP Discriminant Analysis," *Decision Sciences* 20, 239-57.
- Koehler, G. J., and S. S. Erenguc, 1990a, "Minimizing Misclassifications in Linear Discriminant Analysis," *Decision Sciences* 21, 63-85.
- Koehler, G. J., and S. S. Erenguc, 1990b, "Survey of Mathematical Programming Models and Experimental Results for Linear Discriminant Analysis," *Managerial and Decision Economics* 11, 215-225.
- Mahmood, M., and E. Lawrence, 1987, "A Performance Analysis of Parametric and Nonparametric Discriminant Approaches to Business Decision Making," *Decision Sciences* 18, 308-26.
- Mangasarian, O. L., 1965, "Linear and Nonlinear Separation of Patterns by Linear Programming," *Operations Research* 13, 444-52.
- Nath, R., and T. Jones, 1988, "A Variable Selection Criterion in the Linear Programming Approaches to Discriminant Analysis," *Decision Sciences* 19, 554-63.
- Weiss, Sholom M., and C. A. Kulikowski, 1991, *Computer Systems That Learn*, Morgan Kaufmann Publishers, Inc., San Mateo, California.