Three Essays in Quantitative Marketing

by

Ka-Kit Tse

A thesis submitted in partial fulfillment of the requirements

for

the degree of Master of Philosophy

Department of Marketing

Faculty of Business Administration

The Chinese University of Hong Kong

May 1997

To my parents, Joey and Phyllis

Acknowledgment

In the two-year graduate study, a number of people have contributed to my growth and the completion of my thesis. I would like to take this chance to thank all of them for their valuable advice and help in this dissertation, especially, Professor Kin-nam Lau, and Professor Pui-lam Leung for their valuable advice and help in this dissertation.

The topics of the studies were stimulated by Professor Kin-nam Lau. Under his guidance, I can do my research in a proper and correct way. And he introduces me different prespectives of marketing and leads me into the interesting side of academic research. I am pleased to have him as my supervisor.

Professor Pui-lam Leung helps me a lot in statistical analysis. He teaches me many advanced statistical skills and brings me different approaches to research problems. This leads me into the interesting side of academic research. I sincerely appreciate the thoughtful comments from him.

I would like to express my appreciation to the marketing department of The Chinese University of Hong Kong, particularly to Professor Leo Sin, Professor Alan Tse, Professor Julie Yu, Professor Edwin Tang, and Professor Powpaka. With their help and coach, my studies can be completed successfully.

Finally, I must thank my friends, Yat-Lung Cheung, Ian Wong, Leona Tam, Alan Lau, Milky Hon, and most importantly, my parents, my sister Joey, and my friend Phyllis. They share my ups and downs in study and give their kindly encouragement which strengthened me and helped me to face challenge and difficulties. I would like to attribute the success of this dissertation to the above-mentioned people.

Table of Contents

List of Tables

Statistical applications in business and social sciences are growing rapidly. The most popular technique to analyze the causation relationship among variables is the regression method. Consider a simple case to research the relationship between a consumer's income (X) and his/her spending on a particular product (Y). A regression model can be formulated as:

$$Y_i = a + bX_i + \varepsilon_i$$

where (a, b) are parameters,

ε_i is the noise and i indexes for the consumer.

In addition to the statistical stationary assumptions on the noise term, the above regression model presupposes the income sensitivity parameter 'b' (i.e., the slope) is constant and identical for all consumers in the sample. One of the major task for marketing researchers is to use the sample information to divide the market into meaningful segments so that a firm can achieve better profits through targeting the customers with highest potential. Therefore, we expect 'b' will be large for some customers, but small for another group of customers. The research issues are:

1. to determine the number of groups in the sample,
2. to identify the membership for each group,
3. to find the income sensitivity for each group.

One adhoc approach to address these issues is:

1. to assume k segments in the sample,
2. to use 'clustering analysis' to divide customers into k segments,
3. to perform regression in each segment.

On the surface, this approach is reasonable. However, it does not deliver to what we want. While our original intention is to use 'b' (defined in a regression model) as the basis for segmentation, the two stage procedure uses the distances among individuals as the segmentation base.

In other words, the cluster analysis in stage 1 ignores the causation information (i.e., Y is a dependent variable and X is an independent variable) in the clustering process. Therefore, the clustering procedure is not performed in a regression context. This implies the clustering and regression results obtained by optimizing two different objective functions in stages 1 and 2 may not be consistent.

A better approach "Clusterwise Regression" is proposed by Spath (1979). His algorithm (called "exchange algorithm") integrates the clustering into a regression framework, so that clustering and regression can be performed simultaneously. However, the exchange algorithm can only be applied to minimize the sum of square error or absolute error. In the past two years, we examined the clusterwise regression model and its extensions from the operation research prespective. We have developed numerous mathematical programming models to incorporate parameter heterogeniety in these traditional multivariate statistical methods. This new development allows marketing researchers to perform market segmentation easily with standard mathematical programming software (e.g., GAMS). We further generalize the concept of clusterwise regression to clusterwise discriminant analysis, clusterwise logit and unidimensional scaling model. (see table in the next page for a comparison)

The thesis consists of three applications as follows:

Essay one: A Mathematical Programming Approach to Clusterwise Regression Model and its Extensions (Chapter 2)

Essay two: A Mathematical Programming Approach to Clusterwise Rank Order Logit Model (Chapter 3)

Essay three: A Mathematical Programming Approach to Metric Unidimensional Scaling (Chapter 4)

In addition to these three completed essays, we have been supported by a university grant to research on the taste test methodology and the perceptual mapping technique. These studies have not been completed and more work needs to be done. We report the experimental design and preliminary results on Chapter 5.

Essay	Model	X	Y	Objective	Existing Approach
one	Clusterwise Regression	Observable	Continuous	simultaneously estimate regression and membership parameters	Spath 1979
one	Clusterwise Discriminant	Observable	Categorical	simultaneously estimate discriminant and membership parameters	/
two	Rank Order Logit	non-Observable	Ranking	simultaneously estimate desirability index and membership parameters	Croon 1989
three	Unidimensional Scaling	non-Observable	Continuous	estimate the coordinates of the subjects	Simantiraki 1996 Pliner 1996 Hubert and Arabie 1986

Chapter 2 Essay one - A Mathematical Programming Approach to Clusterwise Regression Model and its extensions

Abstract

In this essay, a clusterwise regression model is used to perform cluster analysis within a regression framework. While the traditional regression model assumes the regression coefficient (β) to be identical for all subjects in the sample, the clusterwise regression model allows β to vary with subjects of different clusters. Since the cluster membership is unknown, the estimation of the clusterwise regression is a difficult combinatorial optimization problem. A "Generalized Clusterwise Regression Model" which is formulated as a mathematical programming problem is proposed in this research. A nonlinear programming procedure (with linear constraints) is used to solve the combinatorial problem and to estimate the cluster membership and β simultaneously. Moreover, by integrating the cluster analysis with the discriminant analysis, a clusterwise discriminant model is developed to incorporate parameter heterogeneity into the traditional discriminant analysis. The cluster membership and discriminant parameters are estimated simultaneously by another nonlinear programming model.

2.1. Introduction

In the past four decades, mathematical programming (MP) approaches to solve multivariate statistical problems have received considerable attention since their introduction by Charnes, Cooper and Fergusion (1955). The major areas of mathematical programming application in statistics include regression analysis (Arthanaria and Dodge, 1981; Schrage, 1991), discriminant / classification analysis (Freed and Glover, 1981, 1986; Glover, 1990; Hand, 1981; Koehler and Erengus, 1990; Lam, Choo and Moy, 1996; Ragsdale and Stam, 1991; Rubin, 1994; Stam, 1990; Wanarat and Pavur, 1996, and others), and cluster analysis (Aronson and Klien, 1989; Jensen, 1969; Klien, Beck and Konsynski, 1988; Koontz, 1975; Mulvey and Beck, 1984; Mulvey and Crowder, 1979; Rao, 1971; Stanfel, 1981, 1986, and others).

These MP applications are interesting. However, they may not be very helpful to analyze real data. Each existing MP model is designed for solving only one particular statistical problem, but the analysis of real data set always involves simultaneous applications of several related statistical models. As an example, consider a simplified version of the market segmentation problem in business. The manager collects a sample of the sales and income data from a set of customers. If the customers have homogenous income elasticity (i.e., the regression coefficient β), β can simply be estimated by regression of sales on income. In real business, customers are heterogenous and income elasticity will vary with customers of different clusters in the sample. The major tasks for the manager are:

(i) use the income elasticity as the basis to divide customers into mutually exclusive segments,

(ii) estimate the average income elasticity for each segment,

(iii) identify the members of each segment.

If we ignore the income elasticity differences among segments, the income elasticity estimated from the regression of sales on income will certainly be biased and inaccurate. In other words, if we want to model the parameter heterogeniety in the traditional regression, the appropriate statistical analysis will involve the simultaneous applications of the cluster analysis and regressioin model. One straight forward approach is the two stage method. In stage 1, we apply cluster analysis to the data set to divide customers into segments. In stage 2, we perform regression for each segment to estimate the income elasticity. The problem is that the functions optimized in stages 1 and 2 are two different objective functions which are not necessarily related. And the rationale behind the two analysis is different. Cluster analysis is non-criterion based which does not consider the causation relationship between the dependent and independent variables, while regression analysis is criterion based.

A better formulation is to integrate the cluster analysis into the regression framework, so that the income elasticities and segment membership parameters can be estimated simultaneously by optimizing one single objective function. In the literature, this integration is named as the "Clusterwise Regression Model". Spath (1979, 1981, 1982, 1985) proposed the "exchange algorithm" which uses the QR-decomposition technique to minimize the sum of square errors in the integrated model. The exchange algorithm is further generalized by Spath (1986) and Meier (1987) to minimize the sum of absolute errors. These clever algorithms are easy to implement, but their performance is quite sensitive to the initial partition and outliers. DeSarbo, Oliver and Rangaswamy (1989) applied the simulated annealing procedure to search for the global optimum in the clusterwise regression. The result is encouraging, but the computational cost is high. Moreover, the issue of choosing the appropriate cooling parameter has not yet been fully resolved (Bertsimas and Tsitsiklis, 1993).

Another approach is the mixture model (Aitkin and Wilson, 1980). It is a parametric procedure with strong distributional assumptions on the noise term. It does not directly classify subjects into segments; instead, it computes the segment membership probabilities for each subject.

In this paper, we incorporate the parameter heterogenity in traditional regression and discriminant analysis. We formulate the clusterwise regression and its extensions as mathematical programming models. A generalized clusterwise regression model will be proposed, and it will include existing models as special cases. We will show that the estimation of the clusterwise regresion model is equivalent to solving a nonlinear mixed integer programming model (NMIP). To estimate the proposed model, a new alternative is derived and it transforms the original nonlinear integer programming model to a simple nonlinear programming model (NLP) with linear constraints. We further extend the idea of the clusterwise regression model to "Clusterwise Discriminant Model" by integrating the cluster analysis into a discriminant framework. While the dependent variable in a clusterwise regression model must be a continuous variable, the dependent variable in the clusterwise discriminant model is categorical. The estimation of the clusterwise discriminant model is formulated as another nonlinear programming problem. All proposed estimation procedures can be easily implemented with existing mathematical programming software (e.g. GAMS).

The paper is organized as follows. In section 1, we present the nonlinear programming formulation of the generalized clusterwise regression model, and apply the new estimation procedure to a data set of electricity consumption. In section 2, we extend the clusterwise regression model to the clusterwise discriminant model, and propose a new method to estimate the model. Another data set is used for illustration. The last section summarizes the contributions and the conclusions.

2.2. A Mathematical Programming Formulation of the Clusterwise Regression Model

We will propose a "Generalized Clusterwise Regression Model" and formulate it as a mathematical programming problem. The proposed model will include the other clusterwise regression specifications as special cases.

2.2.1. The Generalized Clusterwise Regression Model

To illustrate the generalized clusterwise regression model, consider a sample of n subjects with measurements on two variables Y (a continuous dependent variable, e.g., sales) and X (an independent variable, e.g., income). We are asked to divide the sample into two mutually exclusive segments (1, 2) according to the impact of X on Y.

Let the membership parameters for subject i be ${}_1Z_i$ and ${}_2Z_i$ defined as follows:

$$
\begin{aligned}
{}_1Z_i &= 1 \quad \text{if subject i belongs to segment 1 ,} \qquad (1)\\
&= 0 \quad \text{otherwise .}\\
{}_2Z_i &= 1 \quad \text{if subject i belongs to segment 2 ,}\\
&= 0 \quad \text{otherwise .}
\end{aligned}
$$

Since subject i is a member of either segment 1 or segment 2, ${}_1Z_i + {}_2Z_i = 1$.

Then, the specification of the two segments are:

Segment 1: $Y_i = a_1 + b_1 X_i + {}_1\varepsilon_i$ if ${}_1Z_i = 1$ (2)

Segment 2: $Y_i = a_2 + b_2 X_i + {}_2\varepsilon_i$ if ${}_2Z_i = 1$ (3)

where $a = (a_1, a_2)$, $b = (b_1, b_2)$ are segment parameters and $\varepsilon = ({}_1\varepsilon_i\ , {}_2\varepsilon_i\ )$ are noises.

The membership parameters $Z = ({}_1Z_i, {}_2Z_i)_{i = 1,..,n}$, are unknown. The research objective is to estimate Z and segment parameters (a, b) simultaneously.

The original clusterwise regression model (Spath, 1979) is a nonparametric procedure. However, a more rigorous model can be derived if we impose distributional assumption on the noise term ε. For simplicity, let us assume ε to follow the normal distribution, i.e.,

$$f_{i|1}({}_1\varepsilon_i) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{{}_1\varepsilon_i^2}{2\sigma_1^2}} . \qquad \text{if} \qquad {}_1Z_i = 1 \qquad (4)$$

$$f_{i|2}({}_2\varepsilon_i) = \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{{}_2\varepsilon_i^2}{2\sigma_2^2}} . \qquad \text{if} \qquad {}_2Z_i = 1 \qquad (5)$$

where $f_{i|j}$ is density function of ${}_j\varepsilon_i$

σ_1 is standard deviation of ${}_1\varepsilon_i$

σ_2 is standard deviation of ${}_2\varepsilon_i$

The joint likelihood function of ${}_j\varepsilon_i$ (denoted as L) and the joint log-likelihood function (denoted as LL) are:

$$L = \prod_{i=1}^{n} (f_{i|1})^{{}_1Z_i} (f_{i|2})^{(1-{}_1Z_i)} \ , \tag{6}$$

$$LL = \sum_{i=1}^{n} ({}_1Z_i \log(f_{i|1}) + (1-{}_1Z_i)\log(f_{i|2})) \ . \tag{7}$$

Let $\theta_1 = (a_1, b_1, \sigma_1)$ and $\theta_2 = (a_2, b_2, \sigma_2)$ be segment parameters. We substitute (2) and (3) into (7) which is maximized with respect to θ_1, θ_2 and the membership parameters $({}_1Z_i)_{i = 1,..,n}$, i.e.,

$$\max_{\theta_1,\theta_2,({}_1Z_i)_{i=1,..,n}} LL$$

$$= \sum_{i=1}^{n} \Big(\ {}_1Z_i\big(-\frac{(Y_i - a_1 - a_2X_i)^2}{(2\sigma_1^2)} - \log\sigma_1\big) + (1-{}_1Z_i)\big(-\frac{(Y_i - b_1 - b_2X_i)^2}{(2\sigma_2^2)} - \log\sigma_2\big) \ \Big) \tag{8}$$

subject to

$({}_1Z_i)_{i=1,\ldots,n}$ are binary (i.e. 0 or 1) variables.

Since ${}_1Z_i$ is restricted to be binary, (8) is a nonlinear integer programming model which is very difficult to solve. It will be interesting and helpful if the combinatorial optimization can be replaced by a smooth and continuous optimization. In appendix A, we will show that the binary variable restrictions on $({}_1Z_i)_{i=1,\ldots,n}$ are sufficient but not necessary to derive the estimates on θ_1 and θ_2. Therefore, we can formulate the clusterwise regression as follows:

$$\max_{\theta_1, \theta_2, ({}_1Z_i, {}_2Z_i)_{i=1,\ldots,n}} LL = \sum_{i=1}^{n} {}_1Z_i\left(-\frac{{}_1\varepsilon_i^{\,2}}{(2\sigma_1^{\,2})} - \log\sigma_1\right) + {}_2Z_i\left(-\frac{{}_2\varepsilon_i^{\,2}}{(2\sigma_2^{\,2})} - \log\sigma_2\right) \quad (9)$$

subject to

$$y_i = a_1 + a_2X_i + {}_1\varepsilon_i \qquad i = 1,\ldots, n$$

$$y_i = b_1 + b_2X_i + {}_2\varepsilon_i$$

$${}_1Z_i + {}_2Z_i = 1$$

$${}_1Z_i \geq 0$$

$${}_2Z_i \geq 0 .$$

The differences between (8) and (9) are:

(i) We incorporate an additional constraint ${}_1Z_i + {}_2Z_i = 1$ in (9).

(ii) $({}_1Z_i, {}_2Z_i)$ are not restricted to be binary. They can take any values between zero and one.

The formulation in (9) is classified as the nonlinear programming model with all linear constraints, and we name it as the "Generalized Clusterwise Regression Model" which includes other versions of the existing clusterwise regression as special cases. In appendix A, we will prove:

(i) Even though $({}_1Z_i, {}_2Z_i)$ are not restricted to be binary, the optimization in the nonlinear programming model (i.e. (9)) will automatically drive ${}_1Z_i$ and ${}_2Z_i$ to be either zero or one.

(ii) The nonlinear integer programming model (i.e. (8)) and the nonlinear programming model (i.e. (9)) are structurally equivalent, and they have the same global optimum.

While (8) involves the combinational optimization and its solution heavily relies upon heuristics (e.g. simulated annealing, or branch and bound algorithm), (9) is a continuous constrained optimization easily implemented with any existing mathematical programming software. Similar to any other nonlinear optimization models, solution to (9) may be sensitive to the choice of starting points. The procedure to choose starting points for the proposed model is given in Appendix B.

2.2.2. Clusterwise Regression Model (Spath, 1979)

The conditional density functions of ${}_1\varepsilon_i$ and ${}_2\varepsilon_i$ usually have different variances. If the variances are equal which is assumed in the model proposed by Spath (i.e., $\sigma_1^2 = \sigma_2^2$ in (4) and (5)), maximizing LL in (9) is equivalent to minimizing the total sum of square errors and the formulation becomes:

$$\min \quad \sum_{i=1}^{n} {}_1Z_i\,({}_1\varepsilon_i^{\,2}) + {}_2Z_i\,({}_2\varepsilon_i^{\,2}) \tag{10}$$

subject to

$$y_i = a_1 + a_2 X_i + {}_1\varepsilon_i \qquad\qquad i = 1, \ldots, n$$

$$y_i = b_1 + b_2 X_i + {}_2\varepsilon_i$$

$${}_1Z_i + {}_2Z_i = 1\,, \qquad\qquad {}_1Z_i\,,\ {}_2Z_i \geq 0\,.$$

(10) is a nonlinear programming model which solves the original clusterwise regression model proposed by Spath (1979).

2.2.3. A Nonparametric Clusterwise Regression Model

The generalized clusterwise regression model (9), which relies upon the distributional assumption on the density function of ${}_1\varepsilon_i$ and ${}_2\varepsilon_i$, is a parametric procedure for clustering and estimating segment parameters simultaneously. If the distribution of the noise term is not known, most researchers appeal to the robustness by minimizing the sum of absolute errors, instead of the sum of square errors in (10). This can also be formulated as:

$$\min \quad \sum_{i=1}^{n} {}_1Z_i({}_1\varepsilon_i^{\,+} + {}_1\varepsilon_i^{\,-}) + {}_2Z_i({}_2\varepsilon_i^{\,+} + {}_2\varepsilon_i^{\,-}) \tag{11}$$

subject to

$$Y_i = a_1 + a_2 X_i + {}_1\varepsilon_i^{\,+} - {}_1\varepsilon_i^{\,-} \qquad\qquad i = 1, \ldots, n$$

$$Y_i = b_1 + b_2 X_i + {}_2\varepsilon_i^{\,+} - {}_2\varepsilon_i^{\,-}$$

$${}_1Z_i + {}_2Z_i = 1$$

$${}_1Z_i\,,\ {}_2Z_i\,,\ {}_1\varepsilon_i^{\,+}\,,\ {}_1\varepsilon_i^{\,-}\,,\ {}_2\varepsilon_i^{\,+}\,,\ {}_2\varepsilon_i^{\,-} \ \geq\ 0\,.$$

In the above formulation, we decompose $({}_1\varepsilon_i, {}_2\varepsilon_i)$ into the difference of two positive components $({}_1\varepsilon_i^{+} - {}_1\varepsilon_i^{-}, {}_2\varepsilon_i^{+} - {}_2\varepsilon_i^{-})$. Schrage (1991) showed that minimizing the sum of absolute errors is equivalent to minimizing the sum of these components. In (11), the objective function is quadratic and the constraints are all linear.

In addition to the mathematical programming solutions, the exchange algorithm in Spath (1979) or its generalizations (Spath 1986; Meier, 1987) can also be used to optimize the objective functions in the original clusterwise regression and the nonparametric clusterwise regression (see (10) and (11)). However, the exchange algorithm will fail to optimize the objective function in the generalized clusterwise regression model (see (9)).

2.2.4. A Mixture Approach to Clusterwise Regression Model

The primary objective of the proposed mathematical programming models (i.e., (9), (10), (11)) or the exchange algorithm (Spath, 1979) is to simultaneously estimate the membership parameters Z (i.e., clustering) and segment parameters (i.e., regression). From the estimates on Z, we are able to classify a subject to either segment 1 or segment 2. Therefore, they are referred as the "Classification Approach or Classification Model" (Celeux and Govaert, 1986) to solve the clusterwise regression problem. Another approach is the mixture model (Aitkin and Wilson, 1980) which is a parametric procedure. Instead of estimating the membership parameters Z directly, the mixture model is used to estimate the mixing proportion P defined as:

$$P = \sum_{1}^{n} {}_{1}Z_i / n \tag{12}$$

$$1 - P = \sum_{1}^{n} {}_{2}Z_i / n$$

P can be interpreted as the prior probability of an object to belong to segment 1. The mixture model assumes the noise term (ε) in the regressions (2) and (3) to follow the normal distribution f (see (4) and (5)). Then, the segment and mixing parameters can be estimated by maximizing the joint log-likelihood function of ε, i.e.,

$$\max_{(a_1, b_1, a_2, b_2, P)} \sum_{i=1}^{n} \log(P f_{i|1}({}_{1}\varepsilon_i) + (1 - P) f_{i|2}({}_{2}\varepsilon_i)) \tag{13}$$

where $f_{i|1}$ is the density function of ${}_{j}\varepsilon_i$, and $f_{i|2}$ is the density function of ${}_{j}\varepsilon_i$.

The E-M algorithm (Dempster, Laird and Rubin, 1977) is useful to optimize the likelihood function in (13). The posterior membership probabilities can be calculated from the parameter estimates and used to assign subjects into segments as follows:

$$\text{Prob}({}_1Z_i = 1 \mid (Y_i, X_i)) = \hat{p}\hat{f}_{i|1} / (\hat{p}\hat{f}_{i|1} + (1-\hat{p})\hat{f}_{i|2}) \quad (14)$$

Assign subject i to segment 1 if

$$\text{Prob}({}_1Z_i = 1 \mid (Y_i, X_i)) > 0.5 \quad (15)$$

The objective function in the mixture approach (i.e., (13)) is different from that in the classification approach (i.e., (9)). Therefore, they cannot be compared directly. The assignment rule between these two approaches is also different. The rule is "all or nothing (i.e., either segment 1 or segment 2)" for the classification model, and the rule becomes probabilistic for the mixture model. Moreover, the classification model estimates more paraemeters than the mixture model.

In general, the choice between the classification and the mixture approach depends upon the research objectives and the validity of the model assumptions. The mixture approach is chosen if the objective is to perform inference on segment parameters, and the classification approach is more appropriate if the objective is to maximize the goodness of fit. For a more detailed comparison, see Celeux and Govaert (1986).

2.2.5. An Illustrative Application

To illustrate the empirical differences between the mixture model and the proposed nonlinear programming (NLP) model, we apply both models to the electricity consumption data (McCormick, 1993) which is given in Table 1A. In this data set, the subjects are fifty states. The dependent variable is the per capita electricity consumption by state (Y) and the independent variables include price of electricity (X_1), per capita income (X_2) and price of gas (X_3). We assume two mutually exclusive segments and the i-th state belongs to either <u>one</u> of the following regression models:

Segment I : $Y_i = \alpha_0 + a_1 X_{1i} + b_1 X_{2i} + c_1 X_{3i} + {}_1\varepsilon_i$ if ${}_1Z_i = 1$

Segment II : $Y_i = \beta_0 + a_2 X_{1i} + b_2 X_{2i} + c_2 X_{3i} + {}_2\varepsilon_i$ if ${}_2Z_i = 1$

where

$a_1, a_2 < 0$	(i.e. demand curve is downward sloping) ,	(16)
$b_1, b_2 > 0$	(i.e. electricity is the normal goods) ,	(17)
$c_1, c_2 > 0$	(i.e. electricity and gas are substitutes) .	(18)

The mixture model and the NLP model are compared for the general case (i.e. $\sigma_1 \neq \sigma_2$) and the restricted case ($\sigma_1 = \sigma_2$). The formulation of the mixture model is given in equation (13) with $f_{i|1}({}_1\varepsilon_i)$ and $f_{i|2}({}_2\varepsilon_i)$ defined in equations (4) and (5). The mixture model is estimated through the E-M algorithm. Since the constrained E-M algorithm is not easy to handle, we ignore the restrictions on the parameters ((16) - (18)) in the E-M implementation. We also generate 1000 sets of starting points randomly for the E-M algorithm and report the best E-M results in the tables 1B to 1D. On the other hand, the generalized clusterwise regression (9), clusterwise regression (Spath, 1979) (10), and the nonparametric clusterwise regression model (11) with all parameter restrictions are estimated by the proposed "nonlinear programming procedure". The findings are summarized below:

(i) In table 1B, the conditional density functions of ${}_1\varepsilon_i$ and ${}_2\varepsilon_i$ are assumed to have different variances (i.e. $\sigma_1^2 \neq \sigma_2^2$). The likelihood value of the NLP model is -69.151 which is larger than -87.614 of the likelihood value of the mixture model. Moreover, the likelihood value of the two segment models (either the mixture or NLP approach) is much better than that of the one segment homogenous model. The income parameter (i.e. a_2, b_2) are supposed to be positive for electricity to be normal goods. However, b_2 of the mixture model is negative. Due to the constraints from the parameter restrictions in the NLP model, b_2 is zero.

(ii) In table 1C, $\sigma_1^2 = \sigma_2^2$ are imposed. Therefore, maximizing the likelihood function is equivalent to minimizing the sum of square errors (SSE). SSE is sharply reduced from 1042.875 (one segment homogenous model) to 369.94 (two segment mixture model) and to 310.907 (two segment NLP model). Again, the SSE of the NLP model is significantly lower than that of the mixture model. While the sign of b_2 in the mixture model is wrong, b_2 is zero in the NLP model.

(iii) In table 1D, we consider the nonparametric procedure. The objective is to minimize the sum of absolute errors. Since the mixture model does not apply in a nonparametric case, we only compare the one segment homogenous model with the two segment NLP model. The signs of all segment parameters of the NLP model are correct. The sum of absolute errors is reduced from 181.124 (one segment homogenous model) to 93.393 (two segments NLP model).

(iv) In all NLP models, the size of segment 1 is quite close to the size of segment 2 regardless of the objective function that we optimize (i.e., 24 vs 26 (table 1B), 22 vs 28 (table 1C) and 23 vs 27 (table 1D)). However, the segment size ratio in the mixture model changes dramatically from 44/6 = 7.3 (when $\sigma_1 \neq \sigma_2$) to 32/18 = 1.7 (when $\sigma_1 = \sigma_2$). This suggests the mixture approach to segmentation is quite sensitive to model specification.

2.3. Mathematical Programming Formulation of the Clusterwise Discriminant Analysis

In the original clusterwise regression model, the dependent variable Y must be restricted to be continuous. In many situations, Y is categorical. As an example, consider the data set in Table 2A about the mortality due to simulated side impact car collisions (Hardle and Stoker, 1989). The categorical dependent variable Y = 1 indicates mortality, and Y = 0 means survival. The three independent variables are age (X_1), velocity (X_2) and acceleration (X_3). For convenience, (X_1, X_2, X_3) are standardized to have E(X) = 0 and Var(X) = 1 (i.e., the standardized variable = $(X - \overline{X}) / \sigma_x$ where $\overline{X}$ and σ_x are sample mean and standard deviation). If the subjects are homogenous, one discriminant function is sufficient and it applies to all subjects in the sample. Under this assumption, the discriminant function can be estimated using the MSD (i.e., minimize sum of deviations) formulation of the discriminant analysis (Ragsdale and Stam, 1991) as follows:

$$\min \sum_{i \in \{Y_i = 1\}} d_i + \sum_{i \in \{Y_i = 0\}} d_i \qquad (19)$$

subject to

$$a\, x_{1i} + b\, x_{2i} + c\, x_{3i} + d_i \geq \varepsilon \qquad \text{for all } Y_i = 1,$$

$$a\, x_{1i} + b\, x_{2i} + c\, x_{3i} - d_i \leq -\varepsilon \qquad \text{for all } Y_i = 0,$$

where ε is a very small positive constant (e.g. 0.001), d_i is the deviation (noise term), and (a, b, c) are parameters.

In most real cases, the subjects are heterogenous. Let us simply assume two clusters of subjects in the sample. The relationship between Y and X = (X_1, X_2, X_3) in cluster 1 is different from that in cluster 2. Therefore, we need two discriminant functions for two clusters. Let $({}_1Z_i,\ {}_2Z_i)$ be the cluster membership parameters, i.e.,

$$\begin{aligned} {}_1Z_i &= 1 \quad \text{if subject i belongs to cluster 1,} \\ &= 0 \quad \text{otherwise;} \\ {}_2Z_i &= 1 \quad \text{if subject i belongs to cluster 2,} \\ &= 0 \quad \text{otherwise.} \end{aligned} \tag{20}$$

Let V_i be the discriminant function that applies to members in cluster i, and it is specified below:

$$\begin{aligned} &\text{Cluster 1:} \quad V_i = a_1x_{1i} + b_1x_{2i} + c_1x_{3i} \quad \text{for all } {}_1Z_i = 1, \\ &\text{Cluster 2:} \quad V_i = a_2x_{1i} + b_2x_{2i} + c_2x_{3i} \quad \text{for all } {}_2Z_i = 1, \end{aligned} \tag{21}$$

where $a = (a_1, a_2)$, $b = (b_1, b_2)$ and $c = (c_1, c_2)$ are parameters of the discriminant functions.

The cluster membership parameters $Z = ({}_1Z_i,\ {}_2Z_i)_{i = 1,\dots, n}$ are unknown. The research objective is to estimate Z (i.e., clustering), and (a, b, c) (i.e., the discriminant parameters) simultaneously. Therefore, we need to integrate the cluster analysis into a discriminant framework. We name this integration as the "Clusterwise Discriminant Model". To incorporate the cluster analysis into the MSD discriminant model, we propose a simple nonlinear programming procedure for the simultaneous estimation of Z and (a, b, c) as follows:

$$\min \sum_{i\in\{Y_i=1\}} ({}_1Z_i({}_1d_i) + {}_2Z_i({}_2d_i)) + \sum_{i\in\{Y_i=0\}} ({}_1Z_i({}_1d_i) + {}_2Z_i({}_2d_i)) \qquad (22)$$

subject to

$a_1x_{1i} + b_1x_{2i} + c_1x_{3i} + {}_1d_i \geq \varepsilon$ for all $Y_i = 1$,

$a_2x_{1i} + b_2x_{2i} + c_2x_{3i} + {}_2d_i \geq \varepsilon$

and

$a_1x_{1i} + b_1x_{2i} + c_1x_{3i} - {}_1d_i \leq -\varepsilon$ for all $Y_i = 0$,

$a_2x_{1i} + b_2x_{2i} + c_2x_{3i} - {}_2d_i \leq -\varepsilon$

and

${}_1Z_i + {}_2Z_i = 1$

${}_1Z_i, {}_2Z_i \geq 0.$

In the above formulation, we do not restrict $({}_1Z_i, {}_2Z_i)$ to be binary. The formulation itself will automatically force $({}_1Z_i, {}_2Z_i)$ to be either zero or one, and the proof is similar to the proof for the clusterwise regression model (see appendix A). The first two constraints apply to subject with $Y_i = 1$. If ${}_1d_i < {}_2d_i$, ${}_1Z_i = 1$ and the subject is classified as a member of cluster 1. Similarly, the subject is assigned to cluster 2 if ${}_1d_i > {}_2d_i$ (i.e. ${}_2Z_i = 1$). The third and forth constraints apply to subjects with $Y_i = 0$, and the classification rule will depend upon the values of $({}_1Z_i, {}_2Z_i)$.

We analyze the mortality data set (table 2A) with the traditional discriminant model and the clusterwise discriminant model. The results are summarized in tables 2B and 2C. If we assume the subjects are homogenous and fit the data set with traditional discriminant analysis, we find that the parameter estimates are all positive with (b_1, c_1) to be very small. This implies X_2 and X_3 are not good predictors of Y. The total deviations (i.e., error) is 0.026 which produces 6 misclassifications and 3 ties in table 2C. If the subjects are heterogenous and the data set is fitted with the clusterwise discriminant model, the sign of parameter estimates in cluster 1 is just the opposite to that in cluster 2. Since c_1 and a_2 are the largest in clusters 1 and 2 respectively, the most important predictors are X_3 in cluster 1 and X_1 in cluster 2. 19 observations are assigned to cluster 1 and the remaining 37 observations are classified as members of cluster 2. The total deviation is zero which gives us 100% correct classification. As a conclustion, we get a better picture of the data when we use the clusterwise discriminant model.

2.4. Conclusion

The analysis of real data set always involve simultaneous applications of several related statistical models. The mathematical programming model may not be very helpful if it is designed to solve only one particular statistical problem. Traditional mathematical programming formulation of regression model assumes the regression coefficient (β) to be identical for all subjects in the sample. This homogeneity assumption is not realistic. The clusterwise regression model assumes identical β for members within a cluster, but different β for members of different clusters. In this paper, we focus on the mathematical programming formulations of the clusterwise regression model and its extensions.

The clusterwise regression is basically a tough combinatorial problem. It is difficult to solve and implement when the sample size is large. To reduce the computational complexity, we propose a simple nonlinear programming model with linear constraints to solve the complicated combinatorial problem. The proposed procedure simultaneously estimates the cluster membership parameters and the regression parameters for each cluster. The implementation can be well supported by existing mathematical programming software (e.g. GAMS). It is more general than the exchange algorithm because it is not restricted to minimize the sum of square errors or absolute errors. It can be used to optimize any objective function (e.g. (9)). While the stochastic optimization algorithm (e.g. simulated annealing) is designed to search for the global optimum, the proposed procedure may generate local optimum. However, the computation cost is much lower than that of the simulated annealing.

The mixture approach to estimate the clusterwise regression model is quite different from the proposed procedure. They differ in terms of the objective functions, number of parameters and the assignment rules. Therefore, they cannot be directly compared. The choice between these two approaches depend upon the research objectives and the validity of the model assumptions.

We also extend the idea of the clusterwise regression model to address the heterogeniety issue in the discriminant analysis. A "Clusterwise Discriminant" model is proposed by integrating the cluster analysis into the MSD discriminant framework. A nonlinear programming procedure is formulated to simultaneously estimate the cluster membership parameters and the disciminant parameters for each cluster.

In this research, the clusterwise regression and clusterwise discriminant model is illustrated with the case of two clusters (i.e., two segments). Generalization of the proposed formulation to include any number of clusters is quite straight forward. More applications will be needed to validate the usefulness of these mathematical programming formulations of clusterwise regression and clusterwise discriminant models, and we leave them for future research.

Appendix A: Nonlinear programming formulation of the General Clusterwise Regression model

LL in (9) can be maximized in two sequential stages. In stage 1, we choose $({}_1Z_i,\ {}_2Z_i)$ to maximize subject i's log-likelihood function (i.e. ${}_1Z_i \log(f_{i|1}) + {}_2Z_i \log(f_{i|2})$) conditional on segment parameters a = $(a_1,\ a_2)$ and b = $(b_1,\ b_2)$, σ = (σ_1, σ_2). In stage 2, we choose (a, b, σ) to maximize the conditional log-likelihood function for all subjects, i.e.,

$$\max_{a,b,\sigma,({}_1Z_i,{}_2Z_i)_{i=1,\dots,N}} LL = \max_{a,b,\sigma}\Big(\sum_{i=1}^{n} \max_{({}_1Z_i,{}_2Z_i)} ({}_1Z_i \log(f_{i|1}) + {}_2Z_i \log(f_{i|2}))\ \Big) \qquad \text{(A.1)}$$

Since ${}_1Z_i \log(f_{i|1}) + {}_2Z_i \log(f_{i|2})$ is linear in $({}_1Z_i,\ {}_2Z_i)$ with coefficients $(\log(f_{i|1}),\ \log(f_{i|2}))$ which are fixed for given (a, b, σ) in stage 1, $({}_1Z_i,\ {}_2Z_i)$ will have corner solutions, i.e.,

$$\max_{{}_1Z_i,{}_2Z_i} ({}_1Z_i \log(f_{i|1}) + {}_2Z_i \log(f_{i|2})) = \max\ (\log f_{i|1}, \log f_{i|2}) \qquad \text{(A.2)}$$

and

$$\begin{aligned} {}_1Z_i &= 1 \quad \text{if} \quad f_{i|1} > f_{i|2} && \text{(A.3)} \\ &= 0 \quad \text{otherwise} \\ {}_2Z_i &= 1 \quad \text{if} \quad f_{i|2} > f_{i|1} && \text{(A.4)} \\ &= 0 \quad \text{otherwise} \end{aligned}$$

In the derivation of (A.2) - (A.4), we allow $0 \le ({}_1Z_i,\ {}_2Z_i) \le 1$. In other words, we do not restrict $({}_1Z_i,\ {}_2Z_i)$ to be binary. The optimization with constraints ${}_1Z_i + {}_2Z_i = 1$ and ${}_1Z_i,\ {}_2Z_i \ge 0$ will force them to be either zero or one.

Appendix B: Choice of the Starting Points

The solution to the nonlinear optimization models in (9), (10) and (11) may be sensitive to the starting points. A two-stage procedure using "ZERO" as starting points is proposed below to address this issue. We use the NLP model in (9) as an example for illustration.

A. Stage 1:

We solve the nonlinear programming model (9) with one additional constraint restricting the size of segment 1 (i.e. number of members in segment 1) to be n_1, where n_1 is arbitrarily chosen by the researcher. Since $({}_1Z_i,\ {}_2Z_i)$ will be forced into zero or one, the segment size constraint can be written as:

$$\sum_{i=1}^{n} {}_1Z_i = n_1 . \tag{B.1}$$

In summary, we solve the following model in stage 1:

$$\max \quad LL = \sum_{i=1}^{n} ({}_1Z_i \log(f_{i|1}) + {}_2Z_i \log(f_{i|2})) \tag{B.2}$$

subject to

$$y_i = a_1 + a_2 X_i + {}_1\varepsilon_i \qquad\qquad i = 1, \ldots, n$$

$$y_i = b_1 + b_2 X_i + {}_2\varepsilon_i$$

$${}_1Z_i + {}_2Z_i = 1$$

$${}_1Z_i,\ {}_2Z_i \geq 0$$

$$\sum_{i=1}^{n} {}_1Z_i = n_1 .$$

(B.2) is solved with ZERO starting points (i.e. the default values) for all parameters including $({}_1Z_i,\ {}_2Z_i)$. Let the optimal LL conditional on n_1 be $LL^*(n_1)$.

B. Stage 2:

We vary the size of segment 1 (i.e. n_1) systematically from 1 to n with step 's' to maximize $LL^*(n_1)$, i.e.,

$$\max_{n_1} LL^*(n_1)\ . \tag{B.3}$$

where s is arbitrarily chosen by the researcher.

The log-likelihood function may further improve if we use the solution to (B.3) as the starting points to optimize (9). The above procedure may not guarantee a global optimum, but it is simple and reasonable results are obtained from our experimentation.

2.6. References

Aitkin, M., and Wilson, G. T. (1980), "Mixture models, outliers, and the EM algorithm," *Technometrics*, 22, 325-31.

Aronson, J. E., and Klein, G. (1989), "A clustering algorithm for computer-assisted process organization," *Decision Science*, 20, 730-45.

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

Bertsimas, D., and Tsitsikis, T. (1993), "Simulated Annealing," *Statistical Science*, 8, 10-15.

Celeux, G. and Govaert, G (1993), "Comparison of the mixture and the classification maximum likelihood in cluster analysis," *Journal of Statistical Computation and Simulation*, 47, 127-46.

Charnes, A., Cooper, W. W., and Ferguson, R. O. (1955), "Optimal estimation of executive compensation by linear programming," *Management Science*, 1, 138.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977), "Maximum likelihood from incomplete data via the E-M algorithm," *Journal of the Royal Statistical Society*, B39, 1-38.

DeSarbo, W. S., Oliver, R. L., and Rangaswamy A. (1989), "A Simulated Annealing Methodology for Clusterwise Linear Regression," *Psychometrika*, 54 (4), 707-36.

Freed, N., and Glover, F. (1981a), "Simple but powerful goal programming models for discriminant problems," *European Journal of Operational Research*, 7, 44-46.

Freed, N., and Glover, F. (1981b), "A linear programming approach to the discriminant problem," *Decision Sciences*, 12, 68-74.

Freed, N., and Glover, F. (1986), "Evaluating alternative linear programming models to solve the two-group discriminant problem," *Decision Sciences*, 17, 151-62.

Glover, F. (1990), "Improved linear programming models for discriminant analysis," *Decision Sciences*, 21/4, 771-85.

Hardle W. and Stoker T. M. (1989), "Investigating Smooth Multiple Regression by the method of average derivatives," *Journal of the American Statistical Association*, 84, 986-95.

Hand, D. (1981), *Discrimination and Classification*, John Wiley and sons, New York.

Jensen, R. (1969), "A dynamic programming algorithm for cluster analysis," *Operations Research*, 1034-57.

Klein, G., Beck, P., and Konsynski, B. (1988), "Computer aided process structuring via mixed integer programming," *Decision Sciences*, 19, 750-61.

Koehler, G. J., and Erenguc, S. S. (1990), "Minimizing misclassification in linear discriminant analysis," *Decision Sciences*, 21, 63-85.

Koontz, W. et al. (1975), "A branch and bound clustering algorithm," *IEEE Transactions on Computers*, September, 908-15.

Lam, K. F., Choo, E. U. and Moy, J. W. (1996), "Minimizing deviations from the group mean: a linear programming approach for the two group classification problem," *European Journal of Operational Research*, 88, 358-67.

McCormick, R. E. (1993), *Managerial Economics*, Prentice Hall, 184-85.

Meier, J. (1987), "A fast algorithm for clusterwise linear absolute deviations regression," *OR spektrum*, 9, 187-89.

Mulvey, J. M., and Beck, M. P. (1984), "Solving capacitated clustering problems," *European Journal of Operational Research*, 18, 339-47.

Mulvey, J., and Crowder, H. (1979), "Cluster analysis: An application of Lagrangian relaxation," *Management Science*, 25, 329-40.

Ragsdale, C. T., and Stam, A. (1991), "Mathematical programming formulations for the discriminant problem: An old dog does new tricks," *Decision Sciences*, 22, 296-307.

Rao, M. (1971), "Clustering analysis and mathematical programming," *Journal of the American Statistical Association*, 66, 622-26.

Rubin, P. (1994), "A comment regarding polynomial discriminant functions," *European Journal of Operational Research*, 72, 29-31.

Schrage, L. (1991), *LINDO*, The Scientific Press, South San Francisco, CA, Chap. 14, 243-44.

Spath, H. (1979), "Clusterwise Linear Regression," *Computing*, 22 (4), 367-73.

Spath, H. (1981), "Correction to algorithm 39: clusterwise linear regression," *Computing*, 26, 275.

Spath, H. (1982), "Algorithm 48: a fast algorithm for clusterwise linear regression," *Computing*, 29, 175-81.

Spath, H. (1985), *Cluster Dissection and Analysis*, New York: Wiley.

Spath, H. (1986), "Clusterwise linear least absolute deviations regression," *Computing*, 37, 371-78.

Stam, A. (1990), "Extensions of mathematical programming based classification rules: A multicriteria approach," *European Journal of Operational Research*, 48, 351-61.

Stanfel, L. E. (1981), "A Lagrangian treatment of certain nonlinear clustering problems," *European Journal of Operational Research*, 7(2), 121-32.

Stanfel, L. E. (1986), "A recursive Lagrangian method for clustering problems," *European Journal of Operational Research*, 27, 332-42.

Wanarat, P., and Pavur, R. (1996), "Examining the effect of second order terms in mathematical programming approaches to the classification problem," *European Journal of Operational Research*, 93, 582-601.

Table 1A: Electricity Consumption Data Set

State	Per Capita Electricity Consumption	Price of Electricity	Per Capita Income	Price of Gas	State	Per Capita Electricity Consumption	Price of Electricity	Per Capita Income	Price of Gas
AK	19.11	3.85	10.59	1.05	MT	29.67	1.38	6.13	1.57
AL	31.74	2.80	5.62	1.65	NC	24.98	3.01	5.94	2.60
AR	21.63	3.19	5.54	1.24	ND	16.84	3.11	6.19	1.81
AZ	25.17	3.82	6.51	1.68	NE	19.17	2.80	6.72	1.40
CA	17.43	3.61	7.91	1.93	NH	14.20	4.30	6.54	3.08
CO	16.07	2.88	7.16	1.43	NJ	16.82	5.01	7.99	3.14
CT	16.93	4.23	8.06	3.78	NM	18.40	3.19	5.86	1.57
DE	23.83	4.22	7.70	2.60	NV	28.15	2.98	7.99	1.85
FL	19.74	3.69	6.68	1.62	NY	16.17	4.92	7.54	2.96
GA	23.43	3.24	6.01	1.78	OH	29.12	2.90	7.08	2.12
HI	21.04	4.62	7.68	6.99	OK	21.38	2.88	6.35	1.45
IA	18.73	3.47	6.88	1.61	OR	33.09	1.61	7.01	2.48
ID	37.57	1.65	5.98	2.29	PA	21.22	3.68	7.01	2.34
IL	21.59	3.38	7.77	2.07	RI	13.53	4.78	6.78	3.68
IN	25.78	2.95	6.92	1.78	SC	29.62	2.88	5.63	1.69
KS	20.26	3.26	7.13	1.25	SD	15.04	3.10	5.96	1.53
KY	39.03	2.06	5.95	1.69	TN	41.41	2.09	5.79	1.60
LA	33.34	2.21	5.91	1.46	TX	31.17	2.86	6.80	1.89
MA	14.81	4.90	7.26	3.68	UT	19.67	2.83	5.92	1.31
MD	23.82	3.80	7.57	2.79	VA	23.84	3.61	6.87	2.52
ME	14.48	3.29	5.73	3.98	VT	15.17	3.61	5.82	2.68
MI	19.68	3.64	7.62	2.06	WA	40.34	0.99	7.53	2.37
MN	18.36	3.32	7.13	1.81	WI	18.29	3.26	6.89	2.13
MO	18.20	3.30	6.65	1.81	WV	25.35	2.79	5.99	1.87
MS	23.46	3.20	5.03	1.63	WY	28.32	1.81	7.56	1.28

Source: McCormick, Robert E. (1993), Managerial Economics, Prentice Hall, 184-85.

Table 1B Estimating the general clusterwise regression model when $\sigma_1 \neq \sigma_2$

	one segment homogenous model	two segment mixture model	two segment NLP model[1]
segment 1			
a_0	39.219	35.728	18.576
a_1	-6.926	-6.509	-2.110
a_2	0.483	0.720	0.936
a_3	1.348	1.410	0.0
σ_1	4.567	4.363	1.596
size[2]	50	44	24
segment 2			
b_0		70.041	43.289
b_1		-14.164	-5.780
b_2		-0.679	0.0
b_3		3.097	0.687
σ_2		0.040	3.549
size[2]		6	26
log likelihood value	-100.941	-87.614	-69.151

Notes:

1. NLP is the proposed Nonlinear Programming Model.
2. The size of segment s (s=1, 2) in the mixture model is computed by multipling the mixing parameter with the total sample size.

Table 1C Estimating the parametric clusterwise regression model (Spath) when $\sigma_1 = \sigma_2$

	one segment homogenous model	two segment mixture model	two segment NLP model
segment 1			
a_0	39.219	23.063	20.650
a_1	-6.926	-5.676	-5.149
a_2	0.483	1.893	2.078
a_3	1.348	1.304	0.348
size	50	32	22
segment 2			
b_0		51.372	46.557
b_1		-7.401	-6.840
b_2		-0.445	0.0
b_3		0.855	0.845
size		18	28
sum of square errors	1042.807	369.94	310.907

Table 1D Estimating the nonparametric clusterwise regression model

	one segment homogenous model	two segment NLP model
segment 1		
a_0	38.742	27.757
a_1	-6.636	-6.536
a_2	0.421	1.418
a_3	1.391	1.426
size	50	23
segment 2		
b_0		39.329
b_1		-7.8
b_2		1.529
b_3		0.859
size		27
sum of absolute errors	181.124	93.393

Table 2A Data on mortality due to simulated side impact car collisions

X_1	X_2	X_3	Y	X_1	X_2	X_3	Y
22	50	98	0	30	45	95	0
21	49	160	0	27	46	96	1
40	50	134	1	25	44	106	0
43	50	142	1	53	44	86	1
23	51	118	0	64	45	65	1
58	51	143	1	54	45	103	0
29	51	77	0	41	45	102	1
29	51	184	0	36	45	108	1
47	51	100	1	27	45	140	0
39	51	188	1	45	45	94	1
22	50	162	0	49	40	77	0
52	51	151	1	24	40	101	0
28	50	181	1	65	40	82	1
42	50	158	1	63	51	169	1
59	51	168	1	26	40	82	0
28	41	128	0	60	45	83	1
23	61	268	1	47	45	103	1
38	41	76	0	59	44	104	1
50	61	185	1	26	44	139	0
28	41	58	0	31	45	128	1
40	61	190	1	47	46	138	1
32	50	94	0	41	45	102	0
53	47	131	0	25	44	90	0
44	50	120	1	50	44	88	1
38	51	107	1	53	50	128	1
36	50	97	0	62	50	136	1
33	53	138	1	23	50	108	0
51	41	68	1	27	60	176	1
60	42	78	1	19	60	191	0

Notes:

1. Y = 1 indicates mortality, and Y = 0 means survival.
2. X_1 is the age.
3. X_2 is the velocity.
4. X_3 is the acceleration.

Source: Hardle W. and Stoker T. M. (1989), "Investigating Smooth Multiple Regression by the Method of Average Derivatives," Journal of the America Statistical Association 84, 986 - 995.

Table 2B Comparison of the homogenous discriminant model and the clusterwise discriminant model

Parameters	homogenous discriminant model (i.e., one cluster)	clusterwise discriminant model (two clusters)
cluster 1		
a_1	0.002	-0.008
b_1	0.00059	-0.004
c_1	0.00047	0.01
size	58	19
cluster 2		
a_2		0.007
b_2		0.003
c_2		-0.004
size		37
total error	0.026	0
Percentage of correct classifications	84.482%	100%

Table 2C Classification results of the homogenous discriminant model

	Predicted Group		
Actual Group	1	tie	0
1	27	3	4
0	2	0	22

Chapter 3 Essay two - A Mathematical Programming Approach to Clusterwise Rank Order Logit Model

Abstract

In traditional rank order logit model, individual is assumed to come from a homogenous group. However, it is unlikely to be true in reality. In this essay, we propose a clusterwise rank order logit model which is used to identify members of latent segments and estimate parameters simultaneously. The estimation requires a solution to a tough combinatorial problem. In this research, we propose a new formulation to transform the nonlinear integer programming problem (i.e. a combinatorial problem) to a nonlinear programming procedure. The proposed procedure can be easily implemented with existing mathematical programming software and it can further be extended to achieve fast convergence with ZERO (the default value) as starting points. Empirical results suggest that the proposed procedure is a viable alternative to existing algorithms in estimating the classification model for market segmentation.

3.1. Introduction

Traditional rank order logit model is applied to the rankings observed in a random sample of respondents from a particular population. It is implicitly assumed that all members of the population perceive and evaluate the stimuli in essentially the same way. This strong assumption of complete homogeneity in the population is certainly untenable in marketing. Since market segmentation is a very important topic in current market research, people consistently differ in their stimulus evaluation. If an analysis which does not leave room for these interindividual differences to show up, it is doomed to fail and to misrepresent the data.

In this paper, we propose a classification model which modifies the traditional rank order logit model to incorporate latent structure analysis. The classification model is applied to rank order data reported by Croon (1989). The respondents of the data are sampled from a non-homogeneous population. We assume that the non-homogeneous population can be partitioned in several subpopulations, each of them being homogeneous with respect to the stimulus evaluations. In this way each subpopulation defines a latent class which is characterized by a particular set of stimulus scale values.

However, classifying each individual into a latent class is a tough combinatorial problem which is difficult to solve. In the proposed method, we transform the integer programming model into a nonlinear programming model.

As a result, the estimation can be implemented with standard mathematical programming software (e.g. GAMS). The proposed procedure is more general because it is not restricted to minimize the sum of square errors only. Moreover, the computation cost is lower than that of some popular optimization methods, such as, simulated annealing. Compared with the E-M algorithm for the estimation of the mixture model, the proposed procedure is more flexible which does not require any closed form solution in the estimation process and it can incorporate any constraints on the parameters. Since the estimates of most nonlinear statistical models are quite sensitive to starting points, the proposed procedure is further extended to achieve fast convergence with ZERO (the default value) as starting points for all parameters.

The paper will be organized as follows. In sections 1 and 2, we formulate the estimation of the classification model as a nonlinear programming model and apply the new procedure to rank order data. The last section summarizes the contributions and the conclusions.

3.2. Clusterwise Rank Order Logit Model

In marketing research, researchers sometimes ask individuals to rank a set of product according to their favourability. In our model, we assume, for simplicity, that there are two mutually exclusive segments in the data set. It is easy to extend the model to more segments. Assume that there are p products, and let $(a_i)_{i=1,\dots,p}$ and $(b_i)_{i=1,\dots,p}$ be the average desirability index of goal i in segments 1 and 2 respectively, i.e.,

Segment 1: $(a_1, \dots, a_p)$,

Segment 2: $(b_1, \dots, b_p)$.

$({}_1Z_i, {}_2Z_i)$ are segment membership indicators which is unknown and defined as:

${}_1Z_i$	=	1	if subject i comes from segment 1
	=	0	otherwise.
${}_2Z_i$	=	1	if subject i comes from segment 2
	=	0	otherwise.

Since each subject must be a member of either segment 1 or segment 2, thus ${}_1Z_i$ is a binary variable. Then it is a nonlinear integer programming model which is very difficult to solve. However the binary variable restrictions on $({}_1Z_i)_{i=1,\dots,n}$ are only sufficient but not necessary

We replace the hard combinatorial optimization by a smooth and continuous optimization by adding some constraints:

(i) We incorporate an additional constraint ${}_1Z_i + {}_2Z_i = 1$.

(ii) $({}_1Z_i,\ {}_2Z_i)$ are not restricted to be binary. They can take any values between zero and one.

The optimization with constraints ${}_1Z_i + {}_2Z_i = 1$ and ${}_1Z_i,\ {}_2Z_i \geq 0$ will automatically force them to be either zero or one. The individual desirability index on goal j for subject i (denoted by D_{ij}) is modeled as:

$$D_{ij} = {}_1Z_i \ .\ \mathrm{a_j} + {}_2Z_i \ .\ \mathrm{b_j} + \varepsilon_{ij} \qquad \mathrm{i} = 1,\ldots,\mathrm{n};\ \mathrm{j} = 1,\ldots,\mathrm{p} \tag{1}$$

where ε_{ij} is the noise term with the extreme value distribution.

If subject i prefers goal j to k to l to m, his/her rank order probability (Chapman and Staelin, 1984) conditional on segment s (denoted as ${}_i\pi_{jklm}(s)$) is:

$$\begin{aligned} {}_i\pi_{jklm}(1) &= \Pr(\ D_{ij} > D_{ik} > D_{il} > D_{im}) \\ &= \left(\frac{\mathrm{e}^{\mathrm{a_j}}}{\mathrm{e}^{\mathrm{a_j}} + \mathrm{e}^{\mathrm{a_k}} + \mathrm{e}^{\mathrm{a_l}} + \mathrm{e}^{\mathrm{a_m}}}\right)\left(\frac{\mathrm{e}^{\mathrm{a_k}}}{\mathrm{e}^{\mathrm{a_k}} + \mathrm{e}^{\mathrm{a_l}} + \mathrm{e}^{\mathrm{a_m}}}\right)\left(\frac{\mathrm{e}^{\mathrm{a_l}}}{\mathrm{e}^{\mathrm{a_l}} + \mathrm{e}^{\mathrm{a_m}}}\right) \end{aligned} \tag{2}$$

$${}_i\pi_{jklm}(2) = \left(\frac{\mathrm{e}^{\mathrm{b_j}}}{\mathrm{e}^{\mathrm{b_j}} + \mathrm{e}^{\mathrm{b_k}} + \mathrm{e}^{\mathrm{b_l}} + \mathrm{e}^{\mathrm{b_m}}}\right)\left(\frac{\mathrm{e}^{\mathrm{b_k}}}{\mathrm{e}^{\mathrm{b_k}} + \mathrm{e}^{\mathrm{b_l}} + \mathrm{e}^{\mathrm{b_m}}}\right)\left(\frac{\mathrm{e}^{\mathrm{b_l}}}{\mathrm{e}^{\mathrm{b_l}} + \mathrm{e}^{\mathrm{b_m}}}\right)$$

The individual likelihood function of subject i (i.e. ${}_iL_C$) is:

$${}_iL_C = ({}_i\pi_{jklm}(1))^{{}_1Z_i}({}_i\pi_{jklm}(2))^{{}_2Z_i}$$

or the joint log likelihood function is:

$$LL_C = \sum_{i=1}^{n} {}_1Z_i \log({}_i\pi_{jklm}(1)) + {}_2Z_i \log({}_i\pi_{jklm}(2)) \qquad (3)$$

Then, the MLE of a = (a_1,..., a_p), b = (b_1,..., b_p), and Z = $({}_1Z_i, {}_2Z_i)_{i=1,\dots,n}$ can be obtained by maximizing LL_C, i.e.,

$$\max_{a,b,Z} LL_C \qquad (4)$$

subject to

$${}_1Z_i + {}_2Z_i = 1$$

$${}_1Z_i, {}_2Z_i \geq 0$$

$$\sum a_p = \sum b_p = 0$$

where $\sum a_p = \sum b_p = 0$ is the normalization constraint.

3.3. Numerical Illustration

Consider a rank order data set reported by Croon (1989). In his paper, 2262 subjects were asked to rank order the following four political goals according to their desirability:

1. Maintain order in the nation
2. Give people more say in the decisions of the government
3. Fight rising prices
4. Protect freedom of speech

The survey results are reproduced in table 1A. Croon (1989) proposed a mixture model implemented with E-M algorithm to estimate the desirability index of each goal for each segment. His estimation results can be found in tables 1B and 1C.

To illustrate the application of the classification model to rank order data, we assume two mutually exclusive segments. Let $(a_i)_{i=1,\ldots,4}$ and $(b_i)_{i=1,\ldots,4}$ be the average desirability index of goal i in segments 1 and 2 respectively, i.e.,

Segment 1: (a_1, a_2, a_3, a_4) ,

Segment 2: (b_1, b_2, b_3, b_4) .

$({}_1Z_i,\ {}_2Z_i)$ are segment membership indicators defined in the earlier section. The individual desirability index on goal j for subject i is represented in equation (1). We use the mathematical programming software GAMS (General Algebraic Modelling Systems) to optimize the log likelihood function in equation (4).

We apply the classification model (4) to Croon's rank order data set (see table 1A). The estimation is implemented with the two stage nonlinear programming procedure using ZERO as starting points. We also compare our estimates with the results of Croon's mixture model in tables 1B and 1C. In table 1B, the log likelihood value of two segment to four segment classification model is compared to that of the mixture model. The log likelihood value of the mixture model does not improve too much when the number of segments increases. As for the classification model, the log likelihood value is sharply reduced from -6427.05 (one segment case) to -5250.54 (two segments case), to -4364.6 (three segments case), and to -4020.7 (four segments case). This suggests the classification model fits the data better than the mixture model. Croon only reported the estimates on the 3 segment mixture model. Therefore, the comparison in table 1C is limited to the three segment case only. From table 1C, we find that the order of goals according to the estimated desirability index in all 3 segments of the classification model is the same as that of the mixture model (i.e. $a_1 > a_3 > a_2 > a_4$ in segment 1; $b_3 > b_1 > b_2 > b_4$ in segment 2; $c_2 > c_4 > c_3 > c_1$ in segment 3).

3.4. Conclusion

In this paper, we propose a clusterwise rank order logit model to allow the mixture parameters to vary across different subjects. However, estimating the model implies solving a highly nonlinear integer programming problem. To reduce its computational complexity, we develop a new procedure which transforms the nonlinear integer programming model to the nonlinear programming model. Since solutions to most nonlinear models are sensitive to starting points, the new procedure is further extended to achieve fast convergence with two stage optimization process using ZERO (the default value) as starting points. The two stage procedure is applied to a published rank order data, and empirical results suggest that the nonlinear programming approach is a viable alternative to the existing algorithms in estimating the classification model.

The proposed procedure has certain nice features. It is more general because it is not restricted to minimize the sum of square errors only. While the stochastic optimization algorithm (e.g. simulated annealing) is designed to search for the global optimum, the proposed procedure may generate local optimum. However, the computation cost is much lower than that of the simulated annealing. To estimate the traditional mixture model using the E-M algorithm, we need to optimize the objective function in the M step of the E-M algorithm. If the closed form solution does not exist or some inequality restrictions are imposed on the parameters, the E-M algorithm may be quite complicated to implement. The proposed procedure does not require any closed form expression in the implementation and it is flexible enough to incorporate any equality or inequality constraints. Finally, the E-M algorithm, which requires distribution assumptions, is a parametric method. The proposed procedure easily implemented with existing mathematical programming software can be parametric or nonparametric.

In conclusion, the nonlinear programming approach to estimate the model seems to be quite successful. Extensions of this model (e.g. to determine the number of segments endogenously) are possible, and its applications to more complicated data structure (e.g. MDS data or conjoint data) are numerous. We will leave them for future research.

3.5. References

Andrews, D. F. and Herzberg, A. M. (1985), *Data: A Collection of Problems from many Fields for the Student and Research Worker*, Springer, New York, 194-97.

Chapman, R. G. and R. Staelin (1982), "Exploiting Rank Ordered Choice Set Data within the Stochastic Utility Model," *Journal of Marketing Research*, 19, 288-301.

Croon Marcel A. (1989), "Latent Class Models for the Analysis of Rankings," in G. De Soete, H. Feger and K. C. Kauer (eds), *New Developments in Psychological Choice Modeling*, Elsevier Science Publisher, North-Holland.

Dempster, A. P., N. M. Laird, and Donald B. Rubin (1977), "Maximum Likelihood from Incomplete Data Via the E-M-algorithm," *Journal of the Royal Statistical Society*, 39 (Series B), 1-38.

Dillon, W.R., Kumar, A., and De Borrero, M.S. (1993), "Capturing Individual Differences in Paired Comparisons: An Extended BTL Model Incorporating Descriptor Variables," *Journal of Marketing Research*, 30, 42-51.

Dillion, W.R. and Kumar, A. (1994), "Latent Structure and Other Mixture Models in Marketing: An Integrative Survey and Overview," in R. P. Bagozzi (ed.), *Advanced Methods of Marketing Research*, Oxford: Blackwell.

Gupta, S., and P. K. Chintagunta (1994), "On Using Demographic Variables to Determine Segment Membership in Logit Mixture Models," *Journal of Marketing Research*, 31, 128-36.

Kamakura, Wagner A. and Gary J. Russell (1989), "A Probabilistic Choice Model for Market Segmentation and Elasticity Structure," *Journal of Marketing Research*, 26 (November), 379-90.

Kass, G (1980), "An Exploratory Technique for investigating large quantities of categorical data," *Applied Statistics*, 29 (2), 119-27.

Luce, R. D. (1959), *Individual Choice Behavior*, New York: Wiley.

McCormick, Robert E. (1993), *Managerial Economics*, Prentice Hall, 184-85.

Ramaswamy Venkatram, Rabikar Chatterjee and Steven H. Cohen (1996), "Joint Segmentation on Distinct Interdependent Bases with Categorical Data", *Journal of Marketing Research*, 33 (August), 337-50.

Schrage, L. (1991), *LINDO*, The Scientific Press, South San Francisco, CA, Chap. 14, 243-44.

Sclove, Stanley L. (1977), "Population Mixture Models and Clustering Algorithms," *Communication in Statistics, Theor. Meth.*, A6 (5), 417-34.
Späth, Helmut (1979), "Clusterwise Linear Regression," *Computing*, 22 (4), 367-73.

Wedel, Michel and DeSarbo, W.S. (1994), "A Review of Recent Developments in Latent Class Regression Models," in R. P. Bagozzi (ed.), *Advanced Methods of Marketing Research*, Oxford: Blackwell.

Table 1A Observed frequencies of the 24 rankings in Croon's sample

No.	Ranking	Frequency	No.	Ranking	Frequency
1	1234	137	13	3124	330
2	1243	29	14	3142	294
3	1324	309	15	3214	117
4	1342	255	16	3241	69
5	1423	52	17	3412	70
6	1432	93	18	3421	34
7	2134	48	19	4123	21
8	2143	23	20	4132	30
9	2314	61	21	4213	29
10	2341	55	22	4231	52
11	2413	33	23	4312	35
12	2431	59	24	4321	27

Table 1B Comparison of the log likelihood values between the rank order mixture model and the rank order classification model

Number of segments	Log Likelihood value	
	mixture model	classification model
1	-6427.05	-6427.05
2	-6384.89	-5250.54
3	-6373.05	-4364.60
4	-6367.76	-4020.70

Table 1C Comparison of parameter estimates between the rank order mixture model and the rank order classification model for 3 segments case

Parameters	mixture model			classification model		
Desirability Index for	segment			segment		
	1	2	3	1	2	3
goal 1	1.99	0.59	-0.69	2.462	0.221	-1.784
goal 2	-0.92	-1.07	0.63	-1.016	-1.358	1.030
goal 3	0.06	1.73	-0.01	-0.194	2.764	0.098
goal 4	-1.13	-1.25	0.07	-1.252	-1.627	0.655
mixing parameter	0.33	0.45	0.22			
size of the segment				997	907	358

Chapter 4 Essay three : A Mathematical Programming Approach to Metric Unidimensional Scaling

Abstract

Classical unidimensional scaling provides a difficult combinatorial task. In essay three, a procedure formulated as a nonlinear programming (NLP) model is proposed to solve this problem. The new method can be implemented with standard mathematical programming software. Unlike the traditional procedures that minimize either the sum of squared error (L_2 norm) or the sum of absolute error (L_1 norm), the proposed method can minimize the error of L_p norm for $1 \leq p < \infty$. Extensions of the NLP formulation to address the multidimensional scaling problem are also discussed.

4.1. Introduction

The classical metric unidimensional scaling problem is to place n objects on the real line, so that the interpoint distances best approximate the observed dissimilarity between pairs of objects. Formally, the problem is to minimize the objective function:

$$\sigma_1(x) = \sum_{i<j} (d_{ij} - |x_i - x_j|)^2 \tag{1}$$

over the parameter $x = (x_1, \ldots, x_n)$, where x_i is the coordinate of object i, d_{ij} is the observed dissimilarity between objects i and j. We assume that the dissimilarity matrix (d_{ij}) is symmetric with positive elements for all $i \neq j$ and $d_{ii} = 0$ for all $i = 1, \ldots, n$. It is known that this problem is equivalent to an NP-hard combinatorial problem and can be solved exactly only for fairly small n. Various approaches to this problem can be found in the literature: Defays (1978), Guttman (1968), Hubert and Arabie (1986, 1988), de Leeuw and Heiser (1977, 1980), Oslon (1984), Pliner (1984, 1986), and others.

Recently, Simantiraki (1996) used a mixed integer programming (MIP) model to minimize the sum of absolute error (instead of the sum of squared error):

$$\sigma_2(x) = \sum_{i<j} |d_{ij} - |x_i - x_j|| \quad (2)$$

As the author mentioned in the paper, it took several hours on a Sun Workstation to solve a MIP model of 17 objects. Therefore, the MIP approach is not feasible for large n. Pliner (1996) proposed an iterative smoothing algorithm to minimize the sum of squared error $\sigma_1(x)$ in (1). This algorithm works remarkably well for large n. However, the solution depends on the choice of smoothing parameter, and the method may not be easily extended to minimize the absolute error objective function (i.e., $\sigma_2(x)$ in (2)).

In this paper, a nonlinear programming (NLP) formulation for the unidimensional scaling problem is proposed. While the MIP approach must restrict many decision variables to be binary and the smoothing algorithm is limited to minimizing the sum of squared error, the proposed NLP model is free of such restrictions and limitations. Moreover, the proposed model can be generalized to solve the multidimensional scaling problem. It can be implemented with existing mathematical programming software. The paper is organized as follows. Details of this NLP formulation are given in Section 2. In Section 3, two examples from Robinson (1951) (the Mani and Kabah collection of archaeological deposits) are used to illustrate this method. These examples are also considered by Hubert and Arabie (1986), Pliner (1996), and Simantiraki (1996). The extension of the NLP model to address the multidimensional scaling problem is discussed in Section 4. Finally, the conclusions are summarized in Section 5.

4.2. Nonlinear Programming Formulation

We first consider the problem of minimizing the sum of squared error $\sigma_1(x)$ in (1). if $x_i > x_j$, the error is $[d_{ij} - (x_i - x_j)]^2$. Similarly, the error is $[d_{ij} - (x_j - x_i)]^2$ if $x_i < x_j$. Therefore, minimizing $\sigma_1(x)$ is equivalent to minimizing the sum of the minimum between $[d_{ij} - (x_i - x_j)]^2$ and $[d_{ij} - (x_j - x_i)]^2$, i.e.,

$$\min_x \sum_{i<j} (d_{ij} - |x_i - x_j|)^2$$

$$= \min_x \sum_{i<j} \min\{[d_{ij} - (x_i - x_j)]^2, [d_{ij} - (x_j - x_i)]^2\}. \tag{3}$$

To solve (3), we define two variables (w_{1ij}, w_{2ij}) with values restricted to be binary (i.e., zero or one). Then, the mathematical programming formulation of (3) is:

$$\min \sum_{i<j} w_{1ij}(e^2{}_{1ij}) + w_{2ij}(e^2{}_{2ij}), \tag{4}$$

subject to:

$$d_{ij} = x_i - x_j + e_{1ij},$$
$$d_{ij} = x_j - x_i + e_{2ij},$$

$$w_{1ij} + w_{2ij} = 1, and$$
$$w_{1ij}, w_{2ij} \geq 0.$$

where e_{1ij} is the error if $x_i > x_j$ and e_{2ij} is the error if $x_i < x_j$.

Since (w_{1ij}, w_{2ij}) are restricted to be zero or one, (4) is a nonlinear integer programming model which is equivalent to an NP-hard combinatorial problem. Fortunately, such binary variable restrictions on (w_{1ij} and w_{2ij}) are sufficient but not necessary. Conditional on a given x with e_{1ij} and e_{2ij} fixed, the objective function in (4) is linear in w_{1ij} and w_{2ij}. The conditional minimization will result in $w_{1ij} = 1$ if $|e_{1ij}| < |e_{2ij}|$ (i.e., $x_i > x_j$), and $w_{2ij} = 1$ if $|e_{1ij}| > |e_{2ij}|$ (i.e., $x_i < x_j$). Therefore, we do not restrict (w_{1ij}, w_{2ij}) to be binary, and (4) is simply a nonlinear programming (NLP) model with cubic objective function and all linear constraints. This NLP formulation will automatically force (w_{1ij}, w_{2ij}) to be either zero or one. In other words, we replace a complicated combinatorial model by a continuous nonlinear programming model.

To minimize the sum of absolute error (i.e., $\sigma_2(x)$ in (2)), we express each error e_{ij} as the difference between two nonnegative components, e^+_{ij} and e^-_{ij}. For example, $e_{1ij} = e^+_{1ij} - e^-_{1ij}$, where

$$e^+_{1ij} = \begin{cases} e_{1ij}, \ if\ e_{1ij} \geq 0; \\ 0, \ otherwise, \end{cases}$$

$$e^-_{1ij} = \begin{cases} 0, \quad if\ e_{1ij} \geq 0; \\ -e_{1ij}, otherwise, \end{cases}$$

The above implies $|e_{1ij}| = e^+_{1ij} + e^-_{1ij}$. Furthermore, $|e_{1ij}|$ is minimized when either e^+_{1ij} or e^-_{1ij} is zero. Therefore, the NLP formulation to minimize the sum of absolute error is:

$$\min \sum_{i<j} w_{1ij}(e^{+}_{1ij} + e^{-}_{1ij}) + w_{2ij}(e^{+}_{2ij} + e^{-}_{2ij}), \qquad (5)$$

subject to:

$$d_{ij} = x_i - x_j + e^{+}_{1ij} + e^{-}_{1ij},$$
$$d_{ij} = x_j - x_i + e^{+}_{2ij} + e^{-}_{2ij},$$

$$w_{1ij} + w_{2ij} = 1, and$$
$$e^{+}_{1ij}, e^{-}_{1ij}, e^{+}_{2ij}, e^{-}_{2ij}, w_{1ij}, w_{2ij} \geq 0.$$

In (5), the objective function is quadratic, and all constraints are linear. The variables (w_{1ij}, w_{2ij}) are <u>not</u> restricted to be zero or one. If the objective is to minimize the error of the L_p norm (where $1 \leq p < \infty$), (5) will be changed to

$$\min \sum_{i<j} w_{1ij}(e^{+}_{1ij} + e^{-}_{1ij})^{p} + w_{2ij}(e^{+}_{2ij} + e^{-}_{2ij})^{p} \qquad (6)$$

We implement the NLP model using the GAMS (General Algebraic Modeling) software (Brooke, Knodrick and Meeraus, 1992). GAMS is a very popular mathematical programming software in the operation research discipline. It solves the optimization problem using a reduced-gradient algorithm (Wolfe, 1962) combined with a quasi-Newton algorithm (Davidon, 1959). This generally leads to superlinear convergence.

Due to the multiextremal nature of the objective functions in (1) and (2), the number of local optima increases sharply with the sample size. The solution produced by the gradient method is always sensitive to the choice of starting points. This applies to most of the existing methods for the unidimensional scaling problem. Therefore, the solution is not necessarily a global optimum unless a good starting point is used. The traditional approach for global optimization using the gradient method is to solve the model with numerous sets of random starting points. This may also be computationally expensive, and we suggest the following alternative to choose good starting points for the NLP models in (4) - (6).

[1] first, we impose an additional constraint: $\sum_{i<j} w_{1ij} = k$ into the NLP model, where k = 0, s, 2s, ... , t. (t is the integer part of $(n^2 - n)/(2s)$, n is the sample size and s is the step size). That is, we restrict the number of $x_i > x_j$ equal to k. Conditional on each value of k, we solve the NLP model with zeroes (default values) as starting points for all x. We solve the model t times for t solutions and objective function values.

[2] We then choose the best solution x* which corresponds to the smallest among these t objective function values. This x* will be a good starting point for the original NLP problem.

[3] Therefore, we use this x* as our starting point and solve the original NLP (i.e., with the additional constraint $\sum_{i<j} w_{1ij} = k$ removed) and obtain the final solution.

In summary, we solve the NLP model totally $(n^2 - n)/(2s) + 1$ times. Compared with the "random starting point" approach, the proposed alternative reduces the number of times that the model needs to be solved. However, there is no guarantee that this method will always produce global optimum. We illustrate the whole procedure with two numerical examples in next section, and the result is quite encouraging. The step size s is set to one for more accurate results in the illustration.

4.3. Numerical Examples

To test our method described in Section 2, we consider two examples from Robinson (1951): the 8 x 8 dissimilarity matrix for the Mani collection of archaeological deposits, and a 17 x 17 matrix for the Kabah collection. These examples were also used as a test cases by Hubert and Arabie (1986), Pliner (1996), and Simantiraki (1996).

We first use the sum of squared error as our objective function to compare our NLP solutions with solutions from the dynamic programming (DP) algorithm (Hubert and Arabie 1986) and the smoothing algorithm (Pliner 1996). The results are summarized in Tables 1A and 1B. Then we use the sum of absolute error as the criterion to compare our NLP solution with the MIP solution (Simantiraki 1996). The results are reported in Tables 2A and 2B. In the NLP implementation, we do not incorporate the centering constraints.

In Table 1A (the Mani data set) and Table 1B (the Kabah data set), the optimal objective function values produced by all the three procedures (DP, smoothing algorithm and NLP) are the same, and the solutions (i.e., value of x) are structurally equivalent.

In Table 2A (the Mani data set), the MIP and NLP procedures generate the same optimal objective function value. In Table 2B (the Kabah data set), the NLP outperforms the MIP with smaller objective function value. This suggests the branch and bound algorithm of the MIP model does not produce global optimum in this case.

The NLP method is solved using GAMS on a Pentium PC. It takes less than 30 seconds to run the NLP model for the Kabah data set of 17 objects with starting points chosen by the method by Section 2.

4.4. Possible Extensions

In principle, the NLP model can be generalized to solve a multidimensional scaling problem. Let (x_i, y_i) be the coordinate of object i. The relationship between the observed distance and the coordinates of object i and j is assumed to be:

$$d_{ij} = |x_i - x_j| + |y_i - y_j| + e_{ij} \qquad (7)$$

where e_{ij} is the error term.

Depending upon the signs of x_i - x_j and y_i - y_j, e_{ij} may have four different values, i.e.,

$$e_{ij} = \begin{cases} e_{1ij} = d_{ij} - (x_i - x_j) - (y_i - y_j), & if\, x_i > x_j \text{ and } y_i > y_j; & (8) \\ e_{2ij} = d_{ij} - (x_i - x_j) - (y_j - y_i), & if\, x_i > x_j \text{ and } y_i < y_j; & (9) \\ e_{3ij} = d_{ij} - (x_j - x_i) - (y_i - y_j), & if\, x_i < x_j \text{ and } y_i > y_j; & (10) \\ e_{4ij} = d_{ij} - (x_j - x_i) - (y_j - y_i), & if\, x_i < x_j \text{ and } y_i < y_j; & (11) \end{cases}$$

Let (w_{1ij}, w_{2ij}, w_{3ij}, w_{4ij}) be four binary variables to represent the above four cases (e.g., $w_{1ij} = 1$ if ($x_i > x_j$, $y_i > y_j$), and $w_{1ij} = 0$ otherwise). Then, the mathematical programming formulation to minimize the sum of squared error is:

$$\min_{x_i, y_i} \sum w_{1ij}(e^2{}_{1ij}) + w_{2ij}(e^2{}_{2ij}) + w_{3ij}(e^2{}_{3ij}) + w_{4ij}(e^2{}_{4ij}) \tag{12}$$

subject to

$w_{1ij} + w_{2ij} + w_{3ij} + w_{4ij} = 1$ and equations (8) to (11).

We do not need to restrict there w_{ij}'s to be zero or one, the optimization will automatically drive them to be binary. Therefore, (12) is a nonlinear programming model with all linear constraints. The challenge is to get good starting points leading to a global optimum.

4.5. Conclusion and Extensions

In this paper, we proposed a NLP approach to the metric unidimensional scaling problem. This NLP problem can be solved by standard mathematical programming software. Furthermore, the NLP approach is flexible and can deal with a sum of squared error as well as the absolute error objective function. In principle, the NLP can be generalized to solve multidimensional scaling problem as well.

On the other hand, the proposed NLP model is not free of limitations. There is no guarantee that the NLP procedure will produce a global optimum. It ultimately depends on the quality of the starting points. The difficulty to get good starting points increases, when the sample size is large or we try to generalize the NLP model to deal with the MDS problem. This starting point issue is common for most of the statistical models, and we leave it for future research.

4.6. References

Brooke, A., Kendrick, D. and Meeraus, A. (1992), *GAMS, A user's guide*, Scientific Press.

Davidon, W. C. (1959), "Variable metric methods for minimization," *A.E.C. Res. and Develop. Report ANL-5990*, Argonne National Laboratory, Argonne, Illinois.

Defays, D. (1978), "A Short Note on a Method of Seriation," *British Journal of Mathematical and Statistical Psychology*, 3, 49-53.

De Leeuw, J., and Heiser, W. J. (1977), "Convergence of Correction Matrix Algorithms for Multidimensional Scaling," *Geometric Representations of Relational Data: Readings in Multidimensional scaling, Ed., J. C. Lingoes, Ann Arbor*, MI: Mathesis Press, 735-752.

De Leeuw, J., and Heiser, W. J. (1980), "Multidimensional Scaling with Restrictions on the Configuration of points," *Multivariate Analysis, Ed., P. R. Krishnaiah*, amsterdam: North Holland, Vol. V, 501-522.

Guttman, L. (1968), "A General Nonmetric Technique for Finding the Smallest Coordinate Space for a Configuration of Points," *Psychometrika*, 33, 469-506.

Hubert, L. J. and Arabie, P. (1986), "Unidimensional Scaling and Combinatorial Optimization," *Multidimensional Data Analysis, Ed. J. De Leeuw, W. Heiser, J. Meulman and F. Critchley, Leiden*, The Netherlands: DSWO Press, 181-196.

Hubert, L. J. and Arabie, P. (1988), "Relying on Necessary Conditions for Optimization: Unidimensional Scaling and some Extensions," *Classification and Related Methods for Data Analysis, Ed., H. H. Bock*, Amsterdam: North-Holland, 463-472.

Oslon, A. A. (1984), "One-Dimensional Metric Scaling," *Automation and Remote Control*, 45, 783-788.

Pliner, V. (1984), "A Class of Metric Scaling Models," *Automation and Remote Control*, 45, 789-794.

Pliner, V. (1986), "The Problem of Multidimensional Metric Scaling," *Automation and Remote Control*, 47, 560-567.

Pliner, V. (1996), "Metric Unidimensional Scaling and Global Optimization," *Journal of Classification*, 13, 3-18.

Robinson, W. S. (1951), "A Method for Chronologically Ordering Archaeological deposits," *American Antiquity*, 16, 293-301.

Simantiraki, E. (1996), "Unidimensional Scaling: A Linear Programming Approach Minimizing Absolute Deviations," *Journal of Classification*, 13, 19-25.

Wolfe, P. (1962), "The reduced-gradient method," *Unpublished manuscript*, RAND corporation.

Table 1: Results from minimizing the sum of squared error ($\sigma_1(x)$)

Table 1A Mani collection

x	DP*	Smooth*	NLP*
x_1	-1.51	-1.506	0.000
x_2	-1.10	-1.100	0.406
x_3	-0.67	-0.671	0.835
x_4	0.16	0.157	1.640
x_5	0.34	0.340	1.846
x_6	0.91	0.905	2.411
x_7	0.93	0.927	2.434
x_8	0.95	0.947	2.454
$\sigma_1(x)$	2.765	2.765	2.765

Table 1B Kabah collection

x	DP*	Smooth*	NLP*
x_1	-1.13	-1.129	0.000
x_2	-0.86	-0.856	0.274
x_3	-0.51	-0.514	0.616
x_4	-0.43	-0.433	0.697
x_5	-0.23	-0.226	0.903
x_6	-0.15	-0.146	0.983
x_7	-0.08	-0.081	1.048
x_8	-0.01	-0.011	1.119
x_9	0.01	0.012	1.142
x_{10}	0.10	0.102	1.231
x_{11}	0.17	0.174	1.304
x_{12}	0.29	0.292	1.422
x_{13}	0.43	0.427	1.557
x_{14}	0.48	0.485	1.614
x_{15}	0.55	0.553	1.682
x_{16}	0.65	0.645	1.775
x_{17}	0.71	0.706	1.835
$\sigma_1(x)$	5.475**	5.472	5.472

Notes:

*DP = Hubert and Arabie's Dynamic Programming
Smooth = Pliner's Smoothing Algorithm
NLP = Nonlinear Programming
** The value is different because of round off error.

Table 2: Results from minimizing the sum of absolute error $\sigma_2(x)$

Table 2A Mani collection

x	MIP*	NLP*
x_1	-1.549	0.00
x_2	-1.009	0.54
x_3	-0.749	0.80
x_4	0.061	1.61
x_5	0.341	1.89
x_6	0.951	2.50
x_7	0.961	2.51
x_8	0.991	2.54
$\sigma_2(x)$	5.85	5.85

Table 2B Kabah collection

x	MIP*	NLP*
x_1	-1.03	0.00
x_2	-0.87	0.20
x_3	-0.49	0.54
x_4	-0.48	0.52
x_5	-0.26	0.77
x_6	-0.16	0.99
x_7	-0.13	0.87
x_8	0.00	1.07
x_9	0.04	1.02
x_{10}	0.06	1.12
x_{11}	0.18	1.19
x_{12}	0.25	1.28
x_{13}	0.45	1.45
x_{14}	0.51	1.54
x_{15}	0.60	1.64
x_{16}	0.66	1.68
x_{17}	0.67	1.56
$\sigma_2(x)$	20.26	20.06

Notes

*MIP = Simantiraki's Mixed Integer Programming
NLP = Nonlinear Programming

5.1. Project 1 -- An Integrated Approach to Taste Test Experiment Within the Prospect Theory Framework

5.1.1. Experiment Procedure

The product used in all taste test experiments are Coke and Pepsi. The taste test experiments are conducted in mid-Nov 1996. The details are below:

Part 1:

Each respondent is first given $20 as a start up capital and he/she is asked to enter bets on four different games as follows:

Game	# of Dice	Outcome for a win	Winning Prob	Bet Ratio	Bet Amount
1	1	1, 2, 3	0.50	2.5	
2	1	1, 2, 3, 4	0.67	2.0	
3	1	1, 2, 3, 4, 5	0.83	1.8	
4	2	2 -- 10	0.92	1.4	

The respondent is also told that only one of the above 4 games is real. The real game will be determined by a random draw from the four games. In addition to the given $20, he is allowed to borrow from us up to an amount of $20. Thus the bet can be in the range from $0 to $40, and the payoff to the respondent is:

Payoff = 20 + (odd - 1) x bet   if the respondent wins

Payoff = 20 - bet   if the respondent losses

Part 2:

If the payoff to a respondent in part 1 is less than $30, we will give him extra money to increase his total payoff to $30. We denote the asset value of the respondent by P2.

Then three cups of cola drinks are presented to the respondent. They have no difference in the appearance. A standard triangle test is conducted. After tasting the products, each respondent is asked to choose the odd product and enter a bet for his pick. The bet ratio is fixed at 2 (i.e., payoff = 2*bet - bet). The respondent is allowed to borrow up to a maximum of $20 from us. The bet is between $0 to P2 + $20.

After entering the bet, we will ask two questions:

1. Which one do you prefer? Odd one or Others
2. What is the brand of the odd product? ____________

The respondents are divided into two groups: "reveal outcome" group and "not reveal outcome" group.

For the "reveal outcome" group, the respondent will know their payoff after the outcome is revealed, their asset value can be updated as follows:

Current Asset Value (P3) = P2 + bet if the respondent wins

Current Asset Value (P3) = P2 - bet if the respondent losses

Part 3:

Reveal Outcome Group:

Each resondent is given another $30 free as startup capital. The current asset value (P4) is equal to P3 + 30. Similar to part 2, three cups of cola drinks are given to the respondent. Then the respondent is asked to taste the softdrinks, pick the odd one and enter the bet for his pick. The bet ratio is also fixed at 2. The respondent is allowed to borrow up to a maximum of $20 from us. The bet for the respondent in this group can range from $0 to P4 + $20.

After entering the bet, we will ask the two same questions as in part 2.

Then we will reveal the result, and calculate the final asset of the respondent as follows:

Final Asset Value = P4 + bet	if the respondent wins
Final Asset Value = P4 - bet	if the respondent losses

Not Reveal Outcome Group:

Another $30 is given to the respondent free as the startup capital. Similar to part 2, three cups of cola drinks are given to the respondent. Then the respondent is asked to taste the softdrinks, pick the odd one and enter the bet for his pick. The bet ratio is also fixed at 2. The respondent is allowed to borrow from us up to an amount of $20 plus bet in part 2. We implicitly assume the respondent wins in part 2. Therefore, the maximum bet is equal to P2 + bet (part 2) + $30 (extra gift in part 3) + $20 (max. borrow amount).

5.1.2. Experimental Results

Id	Group	Bet1	Bet2	Bet3	Bet4	Game	Win/ Loss	Bet for Taste Test 1	Odd Product Chose	Preferred Product	Identified Brand	Bet for Taste Test 2	Odd Product Chose	Preferred Product	Identified Brand	Odd Product in Taste Test 1	Label	Odd Product in Taste Test 2	Label
1	R	5	5	5	5	2	W	10	C	1*	C**	10	B	0*	P**	C	A	P	B
2	R	17	18	19	20	2	W	10	A	0	0	20	B	1	0	C	A	P	B
3	R	40	40	40	40	3	L	50	A	1	0	130	C	0	0	C	A	P	B
4	R	20	25	29	30	3	L	30	C	0	P	50	B	1	C	C	C	P	C
5	R	10	15	20	30	3	W	35	C	0	P	50	B	0	0	C	C	P	C
6	R	10	11	20	21	1	L	20	C	0	0	30	B	0	0	C	C	P	C
7	R	20	21	22	23	3	W	10	C	0	0	20	A	0	0	C	C	P	C
8	R	20	21	40	40	3	L	15	B	1	P	40	A	0	P	C	B	P	A
9	R	5	10	15	20	1	W	20	B	0	C	20	A	0	P	C	B	P	A
10	R	5	10	30	35	2	L	10	B	1	C	20	B	1	0	C	B	P	A
11	R	5	10	15	20	2	W	20	B	1	C	20	A	1	C	C	B	P	A
12	R	5	6	10	20	3	W	20	A	0	P	20	B	0	0	C	A	P	A
13	R	0	20	30	40	3	W	64	C	0	P	30	B	0	P	C	A	P	A
14	R	5	10	15	17	3	W	20	A	1	C	102	A	0	0	C	A	P	A
15	R	10	11	12	13	4	W	10	B	1	C	70	C	0	V	C	B	P	C
16	R	10	11	19	20	2	L	20	A	0	C	40	C	0	C	C	B	P	C
17	R	0	20	40	40	3	W	72	C	0	0	50	A	1	0	C	C	P	A
18	R	12	13	18	20	3	W	10	C	0	0	10	A	1	C	C	C	P	A
19	R	10	20	30	40	1	W	55	C	0	0	30	C	1	0	C	B	P	B
20	R	10	15	20	25	3	W	16	A	0	P	50	B	0	0	C	B	P	B
21	R	40	40	40	40	1	W	20	A	0	C	50	A	0	0	C	B	P	B
22	R	10	15	16	20	2	L	20	A	1	C	30	A	0	C	C	B	P	B
23	R	9	10	11	15	1	L	11	C	0	0	30	C	0	0	C	A	P	C
24	R	0	20	40	40	2	L	50	A	0	C	40	C	0	0	C	A	P	C
25	R	40	40	40	40	3	W	42	A	1	C	100	C	0	0	C	A	P	C

26	R	10	20	40	40	1	W	55	A	0	0	120	B	0	0	C	A	P	C
27	R	10	15	20	25	3	W	30	A	1	0	36	C	0	0	C	B	P	C
28	R	18	19	20	25	1	W	50	B	1	C	147	C	0	P	C	B	P	C
29	R	3	4	5	6	1	W	10	B	1	C	25	A	0	P	C	B	P	C
30	R	10	20	30	40	2	W	40	A	0	P	50	C	1	C	C	B	P	C
31	R	20	30	40	40	4	W	56	A	0	P	122	A	1	C	C	A	P	B
32	R	0	10	15	20	1	W	40	A	0	P	70	C	1	C	C	A	P	B
33	R	10	15	20	30	2	L	20	C	1	0	20	A	1	0	C	C	P	A
34	NR	10	15	25	40	4	L	30	B	0	0	30	B	0	0	C	A	P	C
35	NR	8	9	10	13	3	W	20	C	1	DP	50	B	0	C	C	A	P	C
36	NR	10	20	30	35	3	W	44	A	1	C	50	A	1	C	C	A	P	C
37	NR	5	10	15	20	4	W	20	C	0	0	20	A	0	0	C	C	P	B
38	NR	10	15	20	25	3	W	20	C	1	C	20	C	0	SC	C	C	P	B
39	NR	5	8	10	15	1	W	15	C	0	P	20	A	0	P	C	C	P	B
40	NR	5	7	13	15	1	L	20	A	0	P	40	C	1	C	P	A	C	C
41	NR	0	0	0	20	3	W	20	A	0	P	40	C	1	C	P	A	C	C
42	NR	2	3	5	10	2	W	20	B	1	P	80	C	1	P	P	A	C	C
43	NR	5	10	20	30	3	W	26	B	0	P	32	A	1	C	P	B	C	A
44	NR	5	10	15	20	1	L	20	A	1	C	20	B	0	P	P	B	C	A
45	NR	10	20	39	40	1	W	55	C	0	P	0	B	0	C	P	C	C	C
46	NR	10	12	14	16	1	L	30	B	0	V	90	A	1	C	P	B	C	A
47	NR	0	0	20	20	3	L	30	B	0	0	80	B	0	0	P	B	C	A
48	NR	20	25	30	30	3	W	40	A	0	P	80	B	1	C	P	A	C	B
49	NR	20	25	35	40	1	W	70	B	1	0	50	B	1	0	P	A	C	B
50	NR	5	10	20	30	2	L	40	A	0	P	100	B	1	C	P	A	C	B
51	NR	20	21	30	31	1	W	60	C	1	P	50	B	1	P	P	A	C	C
52	NR	22	25	28	30	2	W	60	A	1	P	80	B	0	P	P	C	C	B
53	NR	10	15	20	25	1	L	30	B	1	C	60	A	1	C	P	C	C	B
54	NR	11	13	16	18	1	W	10	B	0	K	20	A	1	C	P	B	C	A

55	NR	20	21	25	30	4	L	10	B	0	P	20	C	1	0	P	B	C	A
56	NR	40	40	40	40	3	W	72	B	0	P	172	A	1	C	P	B	C	A
57	NR	3	7	10	20	2	L	10	B	1	C	20	C	1	C	P	A	C	A
58	NR	10	20	30	40	3	W	15	A	0	C	40	B	0	P	P	A	C	A
59	NR	2	3	5	10	1	W	20	A	0	V	30	B	0	V	P	A	C	A
60	NR	20	30	35	40	1	L	30	B	1	C	30	A	1	C	P	A	C	A
61	NR	0	5	10	15	4	W	30	A	0	P	30	A	1	C	P	A	C	C
62	NR	40	40	40	40	4	W	36	A	0	P	56	C	0	0	P	C	C	B
63	NR	20	40	40	40	2	W	60	C	0	C	50	B	0	C	P	C	C	B
64	NR	10	20	30	40	4	W	30	C	0	P	30	A	0	0	P	C	C	B
65	NR	20	25	30	35	1	L	25	B	0	P	30	B	0	P	P	B	C	A
66	NR	20	25	30	40	1	W	40	C	0	0	90	C	1	0	P	C	C	C
67	NR	20	21	22	23	2	L	50	B	0	0	100	A	1	V	P	B	C	A
68	NR	15	16	17	18	3	W	15	B	0	P	5	A	1	C	P	B	C	A
69	NR	15	18	19	20	4	W	30	A	0	C	30	B	0	P	P	A	C	B
70	NR	5	7	8	12	1	W	20	C	0	DC	30	C	1	C	P	B	C	A
71	NR	10	12	14	16	1	L	20	C	0	0	20	C	1	0	P	A	C	A
72	NR	10	20	21	30	2	W	40	C	0	P	60	A	1	C	P	C	C	B

* 1: odd product; 0: others
**C: Coke; P: Pepsi; 0: Can't tell

5.2. Project 2 -- An Integrated Approach to Multi-Dimensional Scaling Problem

5.2.1. Introduction

A key element of competitive marketing strategy is product positioning. Product positioning has been defined as the act of designing the image of the firm's offering so that target customers understand and appreciate what the product stands for in relation to its competitors. Each brand within a set of competitive offerings is thought of as occupying a certain position in a customer's "perceptual space". Perceptual mapping refers generally to techniques used to represent this product space graphically. One of the most frequently used perceptual mapping techniques is multidimensional scaling (MDS). MDS is a class of multivariate statistical methods developed in the behavioral sciences to describe subjects' perceptions of objects (stimuli) vis-a-vis multidimensional spatial structures (Wedel and DeSarbo, 1996). A variant of traditional MDS is to include the preference or choice data in it. The major types of this kinds of models are called unfolding and vector models. For a comprehensive discussion, please see (Davison, 1983). In this research, we propose a new design to collect data of preferences, product similarity, and purchase intention. A new estimation procedure is developed to estimate product positions, ideal points and purchase intention thresholds, simultaneously.

5.2.2. Experimental Procedure

The experiment was conducted in mid-Nov 1996. The experiment consists of four rounds of comparisons. Totally four pictures of products (A, B, C, D) are used in the experiment. In each round, one product is used as a reference product, then the individual is given two other products and is asked to choose one out of the two to see which one is more similar to the reference. Then the respondent is asked which one of the two is more preferred. After using all of the four products as the reference, i.e, totally 12 similarity comparison questions and 12 preferenc questions, then the respondent is asked about the purchase intention of each of the four products. He can answer yes, no, or undecided.

For example, use brand A as the reference, the individual is firstly given with the products B and C. He is then asked to compare and choose the one which is more similar to A. And he is also asked to pick the one which is more preferred from B and C. The same two questions are repeated for the product pair (B, D) and (C, D). Then another product is used as the reference, say B, Then the respondent is asked to answer the same two questions for the product pair (A, C), (A, D) and (C, D). The process keeps going until all of the four products have been used as the reference. Then the purchase intention questions are asked.

The order of appearance of the product is randomized to avoid time and order effect. A copy of the questionnaire is attached in the appendix.

5.2.3. Questionnaire

Reference Product	Which one is more similar to the reference?		Which one do you prefer?
A	B B C	C D D	
B	A A C	C D D	
C	A A B	B D D	
D	A A B	B C C	

Are you intended to purchase?			
A	Undecided	Yes	No
B	Undecided	Yes	No
C	Undecided	Yes	No
D	Undecided	Yes	No

5.2.4. Experimental Results

Similarity Data

	Reference Product											
	1			2			3			4		
	Product pairs being compared with the reference product											
id	2,3	2,4	3,4	1,3	1,4	3,4	1,2	1,4	2,4	1,2	1,3	2,3
1	2	2	4	1	1	4	2	1	4	2	1	2
2	2	4	4	3	4	3	2	4	2	2	3	2
3	2	2	4	3	1	3	2	1	2	2	3	2
4	2	2	4	1	4	4	1	1	4	2	1	2
5	3	4	4	3	4	3	2	1	2	2	3	3
6	2	2	3	1	1	3	1	1	2	2	1	2
7	3	4	4	3	4	4	2	4	2	2	3	2
8	3	2	3	1	4	3	1	1	4	1	1	3
9	2	4	3	1	1	3	2	1	4	2	1	2
10	2	2	3	1	1	3	1	4	4	2	3	3
11	3	2	3	1	4	4	2	1	4	2	3	2
12	3	4	4	3	4	3	2	4	2	2	3	3
13	3	2	3	3	4	3	1	1	2	1	3	3
14	3	4	4	3	4	4	1	4	4	2	3	2
15	2	2	4	3	4	3	2	4	2	2	3	2

16	2	4	4	1	4	4	2	4	4	2	3	3
17	2	2	4	1	4	4	1	1	2	2	1	2
18	2	4	4	3	4	4	2	4	2	2	3	2
19	2	2	4	3	4	3	2	4	2	2	3	3
20	3	4	4	3	4	4	2	1	4	2	3	2
21	3	2	3	1	4	4	1	1	4	2	3	2
22	3	2	3	3	4	3	2	4	2	2	3	2
23	3	4	4	3	4	3	1	1	2	2	3	3
24	3	2	3	3	1	3	1	1	2	2	3	3
25	2	4	4	1	4	4	2	4	4	1	1	2
26	2	2	3	1	1	3	2	1	2	2	3	3
27	3	2	3	3	1	3	2	1	2	2	3	3
28	2	2	3	1	1	3	2	1	4	2	1	2
29	2	2	4	1	4	4	2	4	4	2	3	2
30	3	2	4	3	1	3	2	4	2	1	1	2
31	2	2	3	3	1	3	1	1	2	1	3	3
32	2	4	4	3	4	3	2	4	2	1	1	2
33	2	4	4	3	4	3	2	4	2	1	1	2
34	3	4	3	3	1	4	1	1	2	2	3	2
35	2	4	4	1	4	4	2	1	2	2	1	2
36	2	4	4	1	4	4	2	4	2	2	1	2
37	2	4	4	3	1	3	2	1	2	2	3	2
38	3	2	4	3	4	3	2	1	2	1	1	2
39	3	2	3	3	4	3	2	1	2	2	3	3
40	3	4	3	3	4	4	2	4	2	2	3	2
41	3	4	3	3	1	3	2	4	2	1	1	2
42	2	2	3	3	1	3	2	1	2	2	3	2
43	3	2	3	3	1	3	1	1	2	2	3	3
44	3	2	3	3	4	3	2	1	2	2	3	2
45	3	4	3	1	1	3	2	4	4	2	3	3
46	2	2	3	3	4	4	1	1	2	2	1	2
47	2	4	4	1	1	4	2	1	2	2	3	2
48	3	2	4	3	4	4	1	1	2	2	3	2
49	3	4	3	1	4	3	1	1	4	1	1	2
50	3	2	3	3	1	4	1	1	2	2	1	2
51	2	2	3	1	4	4	2	4	4	2	3	3
52	3	4	4	1	4	4	1	4	4	1	3	3
53	2	4	4	1	1	4	1	4	4	2	3	2
54	2	4	4	1	4	4	2	4	2	2	1	2
55	3	2	3	1	1	3	1	1	2	1	3	3
56	3	2	3	3	1	3	1	1	2	1	1	2
57	3	4	4	3	4	3	2	4	2	1	3	3
58	3	2	3	1	4	4	1	1	2	2	3	2
59	2	2	4	3	4	3	2	4	2	2	3	3
60	2	2	3	1	4	3	1	1	2	1	3	3
61	3	4	3	3	4	3	1	1	2	2	3	2

62	2	2	3	3	4	3	2	4	2	1	3	3
63	2	4	4	3	4	3	2	4	4	2	3	3
64	2	2	3	1	4	3	2	4	2	2	3	2
65	3	2	4	3	4	3	2	4	4	2	3	3
66	3	4	3	3	4	4	1	4	2	2	3	2
67	2	2	4	1	1	4	1	4	4	2	3	2
68	3	2	4	1	4	4	2	4	4	2	1	2
69	3	4	4	1	4	4	1	1	4	1	3	3
70	3	4	3	3	4	3	2	4	2	2	3	2
71	2	4	4	1	4	4	2	1	2	2	1	2
72	2	2	4	1	1	4	2	1	2	2	1	2
73	2	2	4	1	4	4	2	1	2	2	1	2
74	3	2	3	3	4	4	2	1	4	2	3	3
75	2	2	3	1	1	4	1	1	4	1	3	3
76	3	4	3	3	4	4	2	4	4	2	3	2
77	2	2	3	1	1	4	2	4	4	2	3	3
78	2	2	3	3	4	4	2	4	4	2	1	2
79	2	2	4	1	1	3	2	1	2	1	1	2
80	3	4	3	1	1	4	2	4	2	2	1	2
81	2	2	4	3	4	3	2	4	2	2	3	2
82	2	4	4	1	4	4	2	4	2	2	1	2
83	3	2	3	3	4	3	2	1	2	2	3	2
84	2	2	4	3	1	3	2	4	4	2	3	3
85	2	4	4	3	4	4	2	4	4	2	3	2
86	2	2	4	1	1	3	1	1	2	2	3	3
87	2	4	4	3	4	4	1	4	4	2	3	3
88	2	2	4	1	1	4	2	4	4	2	3	3
89	3	4	4	1	4	4	1	4	4	2	3	2
90	3	2	3	3	4	3	2	4	2	2	3	3
91	2	2	3	3	1	3	2	1	2	2	3	3
92	3	4	4	3	1	3	2	1	2	2	3	2
93	2	2	4	1	1	4	2	4	4	2	3	2
94	2	4	4	3	4	3	1	4	4	2	3	2
95	3	2	3	3	4	3	2	4	2	2	3	3
96	2	2	3	3	4	3	2	4	2	2	3	2
97	2	4	3	3	4	3	2	1	2	2	1	2
98	2	2	4	1	1	3	1	1	4	2	3	2
99	3	4	4	3	4	4	2	4	2	1	3	2
100	3	2	4	1	4	4	1	1	2	1	1	2
101	3	4	3	3	4	3	2	4	4	2	3	3
102	2	4	4	3	4	3	2	4	2	2	1	2
103	3	2	3	3	4	4	2	1	2	2	3	2
104	2	4	4	1	4	3	2	4	2	2	1	2
105	3	2	3	1	1	3	1	1	2	2	1	2
106	3	2	3	1	1	4	2	1	4	2	3	2
107	3	4	4	3	4	4	2	4	4	1	1	3

108	3	4	3	1	4	4	1	4	4	1	3	3

Preference Data

	Products being compared in preference question					
id	1 vs 2	1 vs 3	1 vs 4	2 vs 3	2 vs 4	3 vs 4
1	2	3	4	3	4	3
2	1	1	1	2	4	4
3	1	1	1	2	2	3
4	2	3	1	3	2	3
5	1	1	1	3	2	3
6	1	1	1	2	2	4
7	1	1	1	3	4	4
8	1	1	1	3	2	3
9	1	1	1	2	2	3
10	1	1	1	3	4	3
11	1	1	1	2	2	3
12	1	1	1	3	4	3
13	2	1	1	2	2	3
14	1	3	1	3	2	3
15	1	3	1	3	2	3
16	1	3	1	3	4	3
17	2	3	4	2	2	3
18	1	1	1	2	4	4
19	1	1	1	2	2	4
20	2	1	4	2	4	4
21	1	1	1	2	2	3
22	1	1	1	2	4	4
23	1	1	1	2	4	4
24	2	1	1	2	2	3
25	2	1	1	2	2	4
26	1	1	1	2	2	3
27	1	1	1	2	2	3
28	2	1	1	2	2	3
29	2	1	1	2	2	3
30	2	1	1	2	2	3
31	2	3	1	3	2	3
32	1	1	1	2	4	4
33	1	1	1	2	4	4
34	2	1	1	2	2	3
35	2	3	1	2	2	3
36	2	1	1	2	2	4
37	1	1	1	2	2	3
38	1	1	1	2	2	3
39	1	1	1	3	2	3
40	2	3	4	2	4	4
41	1	3	1	3	2	3

42	2	3	1	2	2	3
43	1	3	1	3	2	3
44	2	1	1	2	2	3
45	2	3	1	3	2	3
46	2	3	4	2	4	4
47	2	3	4	3	2	3
48	1	1	1	3	4	4
49	1	3	4	3	4	3
50	1	1	1	2	2	3
51	1	1	1	2	2	4
52	1	1	1	2	2	3
53	1	1	1	2	4	3
54	1	1	1	2	2	3
55	1	3	4	3	4	3
56	1	1	4	2	4	4
57	1	1	4	3	4	4
58	2	1	4	2	4	4
59	1	1	1	2	2	4
60	1	1	1	3	2	3
61	1	3	4	3	4	4
62	1	1	1	2	2	3
63	1	1	1	3	4	3
64	1	1	1	2	2	4
65	2	3	4	3	4	4
66	1	1	1	2	2	4
67	2	1	1	2	2	4
68	2	1	4	2	2	4
69	1	3	4	3	4	4
70	1	1	1	2	2	3
71	1	1	1	3	2	3
72	2	3	1	2	2	3
73	1	1	1	2	2	4
74	1	1	1	3	2	3
75	1	1	1	2	2	3
76	1	3	1	3	4	3
77	1	1	1	2	2	3
78	2	1	4	2	2	4
79	1	1	1	2	2	3
80	1	1	4	3	4	4
81	1	1	1	3	2	3
82	1	1	4	2	4	4
83	2	1	1	2	2	4
84	2	1	1	2	2	4
85	2	1	4	2	2	4
86	1	1	1	2	2	3
87	2	1	4	2	2	4

88	1	1	1	2	2	4
89	2	1	1	2	2	3
90	1	1	1	2	2	3
91	2	1	1	2	2	3
92	1	1	1	2	4	4
93	1	3	4	3	4	4
94	1	1	1	3	4	4
95	2	3	4	3	4	3
96	1	1	1	2	2	3
97	1	1	1	3	4	3
98	2	3	1	2	2	3
99	1	1	1	3	4	4
100	1	1	4	2	4	3
101	1	1	4	3	4	3
102	1	1	1	2	2	4
103	1	1	1	2	2	3
104	2	1	1	2	2	3
105	1	1	1	3	4	4
106	1	1	1	2	2	4
107	1	1	1	3	4	4
108	2	3	4	2	2	4

Preference Data

	Products being compared in preference question					
id	1 vs 2	1 vs 3	1 vs 4	2 vs 3	2 vs 4	3 vs 4
1	2	3	4	3	4	3
2	1	1	1	2	4	4
3	1	1	1	2	2	3
4	1	3	1	3	2	3
5	1	1	1	3	2	3
6	1	1	1	2	2	3
7	1	1	1	2	4	4
8	1	1	1	3	2	3
9	1	1	1	2	2	3
10	1	1	1	3	4	3
11	1	1	1	2	2	3
12	1	1	1	3	2	3
13	1	1	1	2	2	3
14	1	3	1	3	4	3
15	1	3	1	3	2	3
16	1	3	1	3	2	3
17	2	3	4	2	2	3
18	1	1	1	2	2	4
19	1	1	1	2	2	3
20	2	1	4	2	4	4
21	1	1	1	2	2	3

22	1	1	1	2	4	4
23	1	1	1	2	4	4
24	2	1	1	2	2	3
25	2	1	1	2	2	4
26	1	1	1	2	2	3
27	1	1	1	2	2	3
28	1	1	1	2	2	4
29	2	3	1	2	2	3
30	1	1	1	2	2	4
31	2	3	1	2	2	3
32	1	1	4	2	4	4
33	1	1	1	2	4	4
34	2	1	1	2	2	4
35	2	3	1	2	2	3
36	2	1	1	2	4	4
37	1	3	1	2	2	3
38	2	1	1	2	2	3
39	1	1	1	3	2	3
40	2	1	4	2	4	4
41	1	3	1	2	2	3
42	2	3	1	2	2	3
43	1	1	1	3	2	3
44	2	1	1	2	2	3
45	2	3	4	3	2	3
46	2	3	4	3	4	4
47	2	3	1	3	2	3
48	1	1	1	3	4	4
49	1	3	1	3	4	3
50	1	1	1	2	4	4
51	1	1	1	2	2	4
52	1	1	1	3	2	3
53	1	1	1	3	4	3
54	1	1	1	2	2	3
55	1	3	4	3	4	3
56	1	1	4	2	4	4
57	1	1	4	3	4	4
58	2	1	4	2	4	4
59	1	1	1	2	2	3
60	1	1	1	3	2	3
61	1	3	4	3	4	4
62	1	1	1	2	2	3
63	1	3	4	3	4	3
64	1	1	1	2	2	4
65	2	3	4	3	4	4
66	1	1	1	2	2	4
67	2	1	1	2	2	4

68	2	1	4	2	2	4
69	1	3	4	3	4	4
70	1	1	1	2	2	3
71	1	1	1	3	2	3
72	2	3	1	2	2	3
73	2	1	1	2	2	3
74	1	3	1	3	2	3
75	1	1	1	2	2	3
76	1	3	1	3	4	3
77	1	1	1	2	2	3
78	2	1	4	2	4	4
79	1	1	1	2	2	3
80	1	1	1	2	4	4
81	1	1	1	3	2	3
82	1	1	4	2	4	4
83	2	1	1	2	2	4
84	1	1	1	2	2	4
85	2	1	1	2	2	4
86	1	1	1	2	2	4
87	2	1	4	2	2	4
88	1	1	1	2	2	4
89	2	1	1	2	2	4
90	1	1	1	2	2	3
91	2	3	1	2	2	3
92	1	1	1	2	4	4
93	1	3	4	3	4	4
94	1	1	1	3	4	4
95	2	3	4	3	4	3
96	2	3	1	2	2	3
97	1	1	1	3	2	3
98	2	3	1	2	2	3
99	1	1	1	3	4	4
100	1	3	1	3	2	4
101	1	3	4	3	4	3
102	1	1	1	2	4	4
103	1	1	1	2	2	3
104	1	1	1	2	2	4
105	1	1	1	3	4	4
106	1	1	1	2	2	4
107	1	1	4	3	4	4
108	2	1	4	2	2	4

Purchase Intention Data

	Product			
id	1	2	3	4
1	3*	3	3	3
2	1	1	1	1
3	1	2	3	3
4	3	3	3	3
5	2	2	1	3
6	1	1	3	3
7	1	3	3	2
8	1	2	2	3
9	1	1	1	1
10	3	3	3	3
11	1	1	3	3
12	1	1	1	1
13	3	3	3	3
14	1	3	2	1
15	1	3	3	1
16	3	3	3	3
17	1	2	2	2
18	2	3	3	2
19	3	3	3	3
20	1	1	1	1
21	1	1	3	3
22	3	3	3	3
23	1	1	1	1
24	2	2	1	1
25	3	2	3	2
26	2	2	1	3
27	1	1	2	3
28	2	2	3	1
29	2	2	2	1
30	1	1	1	1
31	3	2	2	3
32	1	1	1	1
33	2	1	3	2
34	3	3	3	3
35	1	1	1	1
36	1	1	1	1
37	1	2	1	1
38	1	1	1	1
39	1	1	1	1
40	1	1	1	1
41	1	1	2	2
42	3	2	1	3
43	2	1	1	3

44	2	2	1	3
45	3	3	3	3
46	1	1	1	2
47	1	2	2	1
48	2	3	1	1
49	2	1	2	1
50	2	2	3	3
51	1	1	3	1
52	1	1	1	1
53	3	3	1	3
54	1	1	1	1
55	1	3	2	2
56	3	3	3	1
57	1	1	1	1
58	1	1	3	2
59	1	2	3	1
60	2	1	2	3
61	3	3	3	3
62	1	1	1	1
63	1	3	1	1
64	2	1	1	1
65	2	1	2	2
66	3	3	3	3
67	2	2	2	2
68	3	1	1	1
69	1	1	1	1
70	1	1	1	1
71	3	3	3	3
72	3	2	2	3
73	1	1	1	1
74	2	1	2	3
75	1	1	1	1
76	1	1	1	1
77	1	3	3	3
78	1	1	1	1
79	3	1	1	1
80	1	3	3	2
81	1	1	1	1
82	1	1	1	1
83	3	3	3	3
84	1	1	1	1
85	1	2	3	1
86	1	1	1	1
87	1	2	3	1
88	2	2	3	1
89	1	1	1	3

90	1	1	1	1
91	1	1	1	1
92	1	3	3	2
93	3	3	2	2
94	1	1	1	1
95	1	1	1	1
96	1	1	1	1
97	1	1	1	1
98	1	1	1	1
99	1	3	1	1
100	1	2	1	2
101	2	3	2	3
102	1	1	1	1
103	2	3	1	3
104	1	2	3	3
105	1	1	1	1
106	2	1	1	1
107	1	1	1	1
108	1	2	3	2

*Note: 1 Buy
2 Not Buy
3 Undecided