MULTI-CLASS CLASSIFICATION METHODS UTILIZING MAHALANOBIS TAGUCHI SYSTEM AND A RE-SAMPLING APPROACH FOR IMBALANCED DATA SETS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF MIDDLE EAST TECHNICAL UNIVERSITY

BY

DİLBER AYHAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

APRIL 2009

Approval of the thesis:

MULTI-CLASS CLASSIFICATION METHODS UTILIZING MAHALANOBIS TAGUCHI SYSTEM AND A RE-SAMPLING APPROACH FOR IMBALANCED DATA SETS

Submitted by **DİLBER AYHAN** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan ÖZGEN
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Nur Evin ÖZDEMİREL
Head of Department, **Industrial Engineering** _____

Prof. Dr. Gülser KÖKSAL
Supervisor, **Industrial Engineering Dept., METU** _____

Examining Committee Members:

Prof. Dr. Nur Evin ÖZDEMİREL
Industrial Engineering Dept., METU _____

Prof. Dr. Gülser KÖKSAL
Industrial Engineering Dept., METU _____

Assoc. Prof. Dr. İnci BATMAZ
Statistics Dept., METU _____

Assoc. Prof. Dr. Murat Caner TESTİK
Industrial Engineering Dept., Hacettepe University _____

Assist. Prof. Dr. Serhan DURAN
Industrial Engineering Dept., METU _____

Date: 30.04.2009

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: DİLBER AYHAN

Signature :

ABSTRACT

MULTI-CLASS CLASSIFICATION METHODS UTILIZING MAHALANOBIS TAGUCHI SYSTEM AND A RE-SAMPLING APPROACH FOR IMBALANCED DATA SETS

AYHAN, Dilber

M.S., Department of Industrial Engineering

Supervisor: Prof. Dr. Gülser KÖKSAL

April 2009, 84 pages

Classification approaches are used in many areas in order to identify or estimate classes, which different observations belong to. The classification approach, Mahalanobis Taguchi System (MTS) is analyzed and further improved for multi-class classification problems under the scope of this thesis study. MTS tries to explore significant variables and classify a new observation based on its Mahalanobis distance (MD). In this study, first, sample size problems, which are encountered mostly in small data sets, and multicollinearity problems, which constitute some limitations of MTS, are analyzed and a re-sampling approach is explored as a solution. Our re-sampling approach, which only works for data sets with two classes, is a combination of over-sampling and under-sampling. Over-sampling is based on SMOTE, which generates the synthetic observations between the nearest neighbors of observations in the minority class. In addition, MTS models are used to test the performance of several re-sampling parameters, for which the most appropriate values are sought specific to each case. In the second part, multi-class classification methods with MTS are developed. An algorithm, namely Feature Weighted Multi-class MTS-I (FWMMTS-I), is inspired by the descent feature weighted MD. It relaxes adding up of the MDs for variables equally. This provides representations of noisy variables with weights close to zero so that they do not mask the other variables. As a second multi-class classification algorithm, the original MTS method is extended to multi-class problems, which is called Multi-class MTS (MMTS). In addition, a comparable approach to that of Su and Hsiao

(2009), which also considers weights of variables, is studied with a modification in MD calculation. It is named as Feature Weighted Multi-class MTS-II (FWMMTS-II). The methods are compared on eight different multi-class data sets using a 5-fold stratified cross validation approach. Results show that FWMMTS-I is as accurate as MMTS, and they are better than FWMMTS-II. Interestingly, the Mahalanobis Distance Classifier (MDC) using all the variables directly in the classification model has performed equally well on the studied data sets.

Keywords**:** Classification, Multi-class Classification, Re-sampling, Mahalanobis Taguchi System (MTS), Feature Weighted Mahalanobis Distance.

ÖZ

MAHALANOBIS TAGUCHI SİSTEMİ İLE ÇOKLU SINIFLANDIRMA YÖNTEMLERİ VE DENGELİ OLMAYAN VERİ SETLERİ İÇİN BİR YENİDEN ÖRNEKLEME YAKLAŞIMI

AYHAN, Dilber

Yüksek Lisans: Endüstri Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Gülser KÖKSAL

Nisan 2009, 84 sayfa

Sınıflandırma yaklaşımları farklı gözlemlere ait sınıfları tahmin etmek ya da belirlemek için birçok alanda kullanılmaktadır. Bu çalışma kapsamında, Mahalanobis Taguchi Sistem (MTS) sınıflandırma yaklaşımı incelenmiş ve bu yöntem çok sınıflı problemler için geliştirilmiştir. MTS, önemli değişkenleri seçerek Mahalanobis uzaklığına (MU) göre yeni bir gözlemi sınıflandırmaya çalışır. Bu çalışmada, ilk olarak, MTS yönteminde çoklu bağlantı problemi ile küçük veri kümelerinde görülen örnek büyüklüğü sorunları incelenmiş ve çözüm olarak bir yeniden örnekleme yöntemi geliştirilmiştir. Geliştirilen örnekleme yöntemi iki sınıflı problemler için çalışmakta olup, veri çoğaltma ve azaltma yöntemlerini içermektedir. Veri çoğaltma yöntemi, az sayılı sınıfın gözlemlerine ait yakın komşuluklarda sentetik gözlemler oluşturan SMOTE yöntemine dayanmaktadır. Örnekleme yönteminde, duruma göre en uygun değerleri değişen birkaç yeniden örnekleme parametresinin başarımını test etmek için MTS kullanılmıştır. İkinci bölümde, MTS ile çok sınıflı problemleri çözen yeni sınıflandırma yöntemleri geliştirilmiştir. Ağırlıklı MU yaklaşımı kullanılarak, Değişken Ağırlıklı Çoklu MTS-I (FWMMTS-I) geliştirilmiştir. Bu yaklaşımda, MU'nun değişkenlere dayalı eşit ağırlıklı toplanması özelliği hafifletilmiştir. Gürültü değişkenlerin sıfıra yakın ağırlıklarla temsil edilmesi sağlanarak MU hesaplarken diğer değişkenleri gizlemesi engellenmiştir. İkinci olarak, iki sınıflı problemleri çözen MTS'nin çok sınıflı probleme uyarlanmasıyla, Çok Sınıflı MTS (MMTS) geliştirilmiştir. Ayrıca, Su ve Hsiao (2009) çalışmasında önerilen, diğer bir değişken ağırlıklı

çoklu sınıflandırma yaklaşımında, MU hesaplaması değişikliği yapılarak, Değişken Ağırlıklı Çoklu MTS-II (FWMMTS-II) yöntemi olarak isimlendirilmiştir. Tüm yöntemler tabakalı çapraz doğrulama yaklaşımı kullanılarak sekiz farklı çok sınıflı veri kümelesinde karşılaştırılmıştır. Sonuçlara göre, FWMMTS-I yöntemi MMTS ile aynı başarımı göstermiş ve bunlar ise FWMMTS-II yönteminden daha iyi başarım göstermiştir. İlginç olarak, sınıflandırma modelinde tüm değişkenleri doğrudan kullanan MU Yaklaşımı (MDC) da, çalışılan veri kümelerinde aynı derecede başarım göstermiştir.

Anahtar Kelimeler**:** Sınıflandırma, Çoklu sınıflandırma, Yeniden Örnekleme, Mahalanobis Taguchi Sistem (MTS), Değişken Ağırlıklı Mahalanobis Uzaklığı.

To my family and my husband

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my supervisor Prof. Dr. Gülser Köksal for her valuable encouragement, support and insight throughout the research. I would also like to express my sincere appreciation to the examining committee members for their valuable comments.

I wish to thank my friend Barış Yenidünya for his support he has provided me throughout the study. I wish also thank Berna Bakır for her encouragements.

I would also thank the public institution, the Scientific and Technological Research Council of Turkey (TÜBİTAK), for the encouragements. I specially express my thanks to all my friends and my colleagues for their kind encouragements.

I am deeply grateful to my family for their love and moral support they gave me throughout my life. I am so lucky that I have their unlimited belief and love.

Most special thanks go to my beloved husband Onur Ayhan for his endless love, patience, moral support and for everything he added to my life.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

Multivariate data analysis refers to any statistical technique, which analyzes data having more than one variable. This essentially models real situations. Multi-class classification problems are a subset of multivariate problems. They are defined as finding a prediction model of the associated class of a new example on observed variables. Multivariate analysis becomes nontrivial when there are lots of variables. In particular, multi-class classification remains as a research area.

Mahalanobis Taguchi System (MTS) is a multivariate classification technique with known labels. It is developed with a combination of MD to construct a multidimensional measurement scale from a set of observations to a reference point with the determination of the important variables (Taguchi, 2001).

In particular, we study the effect of class imbalance problems on MTS. The motivation comes from problems due to limited number of observations. In fact, sample size problems can be classified in three types. In the first type, the fact that the number of observations may be large enough but less than the number of variables poses an obstacle to MTS, since MTS empirically requires that the number of samples is grater than the number of variables and collecting more data may be a solution to it. In the second type, number of observations in a class with respect to the other classes, over-represented or under-represented, may cause problems. In the third type, sample size and imbalanced data problem may happen at the same time. As a result, over-sampling algorithms should consider the number of variables, as well. One of the objectives of this thesis study is to overcome the third type drawback.

In this study, we first develop a re-sampling algorithm working for two-class data sets, which is a combination of over-sampling and under-sampling. Over-sampling is done by SMOTE, which is an over-sampling method by generating the synthetic observations between the nearest neighbors of observations in the minority class (Chawla et al., 2002). MTS models are also used to handle several imbalanced data sets resized with under-sampling or over-sampling based on search space of class ratio, sample size and number of nearest neighbors. For the purpose of generating rules for suggested re-sampling parameters in the search space, a decision tree classifier is applied to the performance measures to relate data set characteristics with the performance of the models.

In the second part of the study, several multi-class classification algorithms are developed. In the literature, we encounter one recent study on multi-class MTS, which belongs to Su and Hsiao (2009). The other objective of this study is developing multi-class classification methods. We first develop an algorithm called Feature Weighted Multi-class MTS-I (FWMMTS-I) for multi-class classification problems. The descent feature weighted concept in the study of Wölfel and Ekenel (2005), which relax equal adding up of the variable distances in MD calculation is used in FWMMTS-I. In addition, we extend the original MTS algorithm to multi-class problems (MMTS method). . Su and Hsiao (2009) use a Gram-Schmidt (GS) algorithm, which is criticized in the literature since GS is found highly sensitive to data ordering since it depends on which variable is first selected in the order. Thus, a modification is made in MD calculation of Su and Hsiao (2009). We name the latter algorithm “Feature Weighted Multi-class MTS-II (FWMMTS-II)”. Finally, performances of these algorithms are compared on eight different multi-class data sets. The results are also compared to those of Mahalanobis Distance Classifier (MDC) and Weighted Mahalanobis Distance Classifier (WMDC), which is developed by using the descent feature weighted MD calculation of Wölfel and Ekenel (2005).

This thesis consists of four more chapters other than this first chapter of introduction. In the second chapter, some background information about the multivariate classification systems, MD and MTS are provided along with a comprehensive literature review on delimitations and some popular applications of MTS. Moreover, some of the recent literature on re-sampling is presented. In the third chapter, a new re-sampling algorithm with MTS is presented and its performance on different data sets is discussed. In the fourth chapter, the multi-class MTS

algorithms are presented and their performances are compared. In the last chapter, conclusions and further studies that can be done in the future are stated.

CHAPTER 2

LITERATURE SURVEY AND BACKGROUND

2.1 METHODOLOGY AND BACKGROUND

2.1.1 Classification Problems and Methods

Unlike the univariate analysis, multivariate data analysis refers to any statistical technique used to study data that contains more than one variable. This essentially models the reality since each situation, product, or decision mostly involves more than a single independent variable. The goal of the multi-class classification problems, which is also a subset of multivariate problems, is to find a mapping, a model or a function to predict the associated class of a new example. They assume the existence of a pre-defined set of classes. It is also known as a supervised learning in order to distinguish it from clustering (or unsupervised learning).

The classification methods have their own advantages and disadvantages. To illustrate, while the discriminant analysis assumes that data comes from multivariate normal distribution, the Logistic Regression (LR) and Multivariate Adaptive Regression Splines (MARS) do not. In recent years, the structural models have also become popular. An Artificial Neural Network (ANN) is a mathematical or a computational model based on biological neural networks with a structure changing with respect to external or internal information. It can model nonlinearities similar to MARS. ANN and MARS are typically more successful in modeling high-dimensional problems. Unlike ANN and MARS, LR provides probabilistic statements. Unfortunately, LR may be difficult to use without data pre-processing since it may provide too large or infinite coefficient estimates. Besides, ANN cannot name the most important variables while LR and MARS can do. MARS automatically produces the results, while the ANN architecture should be determined by the user.

2.1.2 Performance Measures of Classification (Binary and Multi-class)

The performance measures, which are taken from Weiss and Zhang (2003), can be listed as follows:

2.1.2.1 Binary Classification Measures

A coincidence (or confusion) matrix illustrates the accuracy of a solution for a classification problem.

Table 2.1: A Simple Coincidence Matrix

		Predicted class	
		Positive	Negative
Actual class	Positive	a (TP)	b (FN)
	Negative	c (FP)	d (TN)

According to Table 2.1, while TP and TN denote the number of positive and negative observations which are classified correctly, FN and FP denote the number of misclassified positive and negative observations, respectively.

Percentage of Correct Classification Rate (PCC):

Percentage of correct classification rate (PCC) gives the proportion of true results (both TP and TN) in total observations. For a total number of observations, *n*, PCC equals to:

$$\text{PCC} = \frac{(\text{TP} + \text{TN})}{n}$$

Kappa:

Kappa is the proportion of correctly classified observations after the probability of chance agreement has been removed. Kappa is always less than or equal to 1. A value of 1 implies perfect agreement and less than 1 implies less than perfect agreement. It is defined as follows:

$$\text{Kappa} = (\theta_1 - \theta_2)/(1 - \theta_2)$$

$$\theta_1 = (a + d)/n$$

$$\theta_2 = \frac{[(a + b)/2.(a + c)/2] + [(c + d)/2.(b + d)/2]}{n^2}$$

Where *n* is the total number of observations.

Receiver Operating Characteristics (ROC) Curve:

ROC is a two-dimensional graph, in which true positives (TP) rate is plotted on the y-axis, and false positives (FP) rate is plotted on the x-axis. The ideal point on the ROC curve would be [0, 1], which means all positive observations are classified correctly and no negative observations are misclassified as positive.

Area under ROC Curve (AUC):

AUC measures the area under the ROC curve.

Precision:

Precision is an indicator of sharpness in identifying the class of interest. It is simply defined as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall (Sensitivity) and Specificity:

The sensitivity (also called recall rate) measures the proportion of actual positives which are correctly identified. The specificity measures the proportion of actual negatives which are correctly identified. They are closely related to the concepts of type I and type II errors.

$$\text{Sensitivity} = \frac{TP}{TP + FN} \qquad \text{Specificity} = \frac{TN}{TN + FP}$$

Specificity is sometimes confused with the precision. The distinction is critical when the classes are different sizes. A test with very high specificity can have very low precision if there are more true negatives than true positives, and vice versa.

F measure:

In general, there is a tradeoff between the precision and recall, which can be achieved. Thus, the F-measure is a convenient way of looking at the tradeoff between precision and recall in a single measure. In a sense, F-measure measures the balance between precision and recall. The traditional F-measure is:

$$F = \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}}$$

Two other commonly used F-measures are the F_2 measure, which weights recall twice as much as precision; and $F_{0.5}$ measure, which weights precision twice as much as recall.

$$F_\beta = \frac{(1+\beta^2)(precision \times recall)}{(\beta^2 \cdot precision + recall)}$$

Geometric Mean of Sensitivity and Specificity

The geometric mean of specificity and sensitivity (G-mean) gives the importance to the balance measurement of class accuracies. G-mean is:

$$\text{G-mean} = \sqrt{sensitivity \times specificity}$$

Stability:

A classification model is stable when it performs just as well on seen (training) and unseen (test) data sets. The stability can be measured as a number between 0 and 1, where 0 means completely stable and 1 means completely unstable. This measure can be calculated as the arithmetic difference divided by arithmetic sum of the training and test classification rates, CR_{TR} and $CR_{TE,}$ respectively.

$$\text{Stability} = (CR_{TR} - CR_{TE})/(CR_{TR} + CR_{TE})$$

2.1.2.2 Multi-class Classification Measures

Average of Class Accuracies:

Class accuracy defines the number of correct classifications in each class. Su and Hsiao (2009) use "Balanced Class Accuracy (BCA)", as an average of class averages since it computes accuracy independent from the size of each class.

$$BCA = \frac{\sum_{i=1}^{L}\left[(TP_i + TN_i)/n_i\right]}{L}$$

where n_i is the size of class i for L classes.

Percentage of Correct Classification Rate (PCC):

PCC is the ratio of true results (both TP and TN) to total observations, which is N for L classes.

$$PCC = \frac{\sum_{i=1}^{L}(TP_i + TN_i)}{N}$$

Stability can also be a measure of multiclass classification.

2.1.3 Mahalanobis Distance

Mahalanobis distance (MD) was first introduced by Prasanta Chandra Mahalanobis in 1936. Considering the correlations, it is a way of making a group of multivariate variables uniform. Classifiers based on MD are mostly used for statistical purposes. MD is also used for selection of outliers. MD can be perceived to be a constant multiple of Hotelling's T^2 (Hawkins, 2003, Abraham and Variyath, 2003).

The covariance between two variables is simply the average product of the values of two variables X_i, X_j, which are expressed as deviations from their respective means, μ_i and μ_j :

$$\mathrm{Cov}(X_i, X_j) = \mathrm{E}\left[(X_i - \mu_i)(X_j - \mu_j)\right] \tag{2.1}$$

Given the covariance of X_i and X_j, correlation coefficient between variables X_i and X_j is obtained as:

$$r_{ij} = \frac{\mathrm{Cov}(X_i, X_j)}{\sigma_i \sigma_j} \tag{2.2}$$

Where σ_i and σ_j are the standard deviations of the variables X_i and X_j.
MD is a squared distance (also denoted as D^2) , which is obtained by:

$$\mathrm{MD} = D^2 = \frac{\mathbf{z}_i^{T} \mathbf{C}^{-1} \mathbf{z}_i}{k} \tag{2.1}$$

$$\mathbf{z}_i = \left(\frac{x_{i1} - \mu_1}{\sigma_1}, \ldots, \frac{x_{ik} - \mu_k}{\sigma_k}\right) \tag{2.4}$$

where:
$\mathbf{z}_i$: standardized $\mathbf{z}_i$ vector obtained by the standardized values of x_i (i=1,…,k)
k : the number of variables.
T : transpose of the vector
$\mathbf{C}^{-1}$: inverse of the correlation matrix

The mean of MD values is expected to be one since Equation (2.3) uses a normalized vector **z** and a normalizing factor 1/*k*. The main assumption of MD is that the variables are multivariate normal. Based on the central-limit theorem, many sampling distributions can be approximated to normal. In addition, this assumption may be relaxed in situations, where the number of observations becomes larger (Johnson and Wichern, 1998).

In addition, when a mixture of continuous and discrete variables is present, MD could be generalized (Bar-Hen and Daudin, 1995). Bar-Hen and Daudin (1995) (as cited in Leon and Carrière, 2005) apply the Kullback–Leibler divergence to the general location model and derive a distance that specializes to the MD in the absence of nominal variables. Afterwards, the distance is utilized for the mixed continuous and discrete data, which provides to use the qualitative as well as the quantitative data (Bedrick et al., 2000 as cited in De Leon and Carrière, 2005). Finally, Leon and Carrière (2005) derive an MD which can be used with data mixed with nominal, ordinal and continuous variables.

MD differs from the Euclidean distance in addressing correlations. Wölfel and Ekenel (2005) state that MD is a weighted Euclidean distance, where the weights are expressed by the covariance matrix. According to Srinivasaraghavan and Allada (2006), MD is superior to the other statistical approaches in the following ways; it considers the covariance and ranges of acceptability (variance) between variables; it compensates for interactions (covariance) between variables; it lacks dimension. It is an effective method since a lot of observations can be analyzed due to the matrix calculation (Riho et al., 2005).

On the other hand, there are also some limitations of MD. MD assumes equal priorities in variables, as well as equal misclassification costs (Sharma, 1996). Moreover, MD does not consider the specific contribution of a variable. In addition, as a requirement of MD, the number of observations collected in the normal group should be larger than the number of variables (Srinivasaraghavan and Allada, 2006). In fact, generally the number of observations may not be enough, compared to its dimensionality. As a result, the covariance matrix usually cannot be estimated accurately. In addition, to calculate MD from an observation **Y** to **X**, **X** and **Y** must have the same number of columns due to the matrix operation (division), but may have different numbers of rows.

MTS method is based on MD and is explained in the following sections.

2.1.4 Mahalanobis Taguchi System

Mahalanobis Taguchi System (MTS) is a method of classification and selection of the significant variables. MTS, a combination of MD with Taguchi's robust engineering, addresses a scale based on data input characteristics to measure the degree of abnormality. Consequently, an unknown observation is assigned to a class. Cudney et al. (2006) illuminate MTS for the statistical measure of how well an unknown observation matches a known observation. Since MTS is based on MD, its assumptions are similar to the assumptions of MD, which are given in the previous part.

In MTS, every example outside the normal space (that is, abnormal example) is regarded as unique, which does not constitute a separate population. As a result, Taguchi and Jugulum (2000) do not accept MTS as a classification method. Taguchi and Jugulum (2000) also mention the usage of categorical data with MTS. Given *m* as the number of levels for a categorical variable, *(m-1)* columns are allocated for the categorical variable. If an observation has a level of one, then all the allocated variables apart from the first column is assigned to be zero. On the other hand, if the level of the categorical variable is 2, the second column corresponding to the given observation is assigned to 1, while the others are assigned to 0.

Woodall et al. (2003) criticize MTS by stating that the other statistical methods are better designed to account for the sampling variation and the variation between two observations. This lack of attention to variation between the observations is more evident in the MTS clinical trials which results to at least some classification errors. On the other hand, Srinivasaraghavan and Allada (2006) mention that MTS is a very effective technique for detection of complicated causes of failures due to its eligibility for the matrix calculation.

The procedure of MTS is not much complicated. In the first stage, data pre-processing is performed. Data set is separated into a normal ("healthy") group, which shows homogenous characteristics, and an abnormal ("unhealthy") group. For example, for a cancer data, the healthy people constitute a normal group, whereas the people with cancer constitute an abnormal group.

MTS is a method of supervised learning in order to distinguish in which the classes (or labels) are known. Mean and standard deviation of the normal group are used in order to standardize the abnormal and normal groups. Normal group constitutes a Mahalanobis space (MS). MTS model tries to memorize the specifications of the normal group by including the inverse of the correlation matrix of the normal group. Data pre-processing is continued to check for detecting the normality of variables and outliers of MS scale.

An outlier is defined as an observation that lies outside the overall pattern of a distribution (Moore and McCabe, 1999) as shown in Figure 2.1. In some of cases, the selection of an appropriate MS becomes nontrivial, although the detection of outliers can be done by means of several ways. Dot plots can be drawn but when the number of variables is large, they are not very visual. The other way is examining standardized normal observations on a Chi-square plot. MD is also used in outlier detection (Anderson and Schumacker, 2003). For example, it is used as an outlier detection method in the De Groot et al. (2001). A threshold value is utilized in detecting outliers via MD to omit the observations having higher MDs. After detecting the outliers, the MS scale is validated. If the MS scale is not validated, it cannot suitably represent the normal condition in reality and is necessary to be reconstructed with a new MS using the remaining normal observations until a suitable MS obtained (Taguchi and Jugulum, 2002). This is all for data pre-processing.

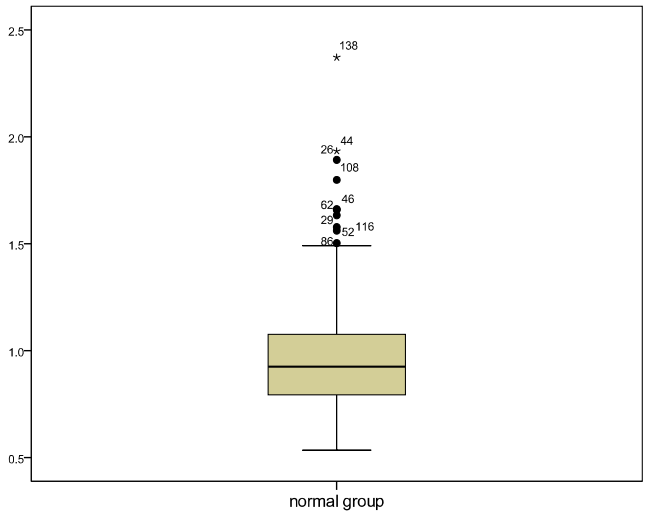

Figure 2.1: Outliers of a Mahalanobis Space (MS) (for the vehicle data)

In the second stage, MD values of abnormal observations are calculated. MD of a standardized $\mathbf{z}_i$ vector is given in Equation (2.3). The MD values of abnormal group are expected to be higher than normal group, as shown in Figure 2.2.

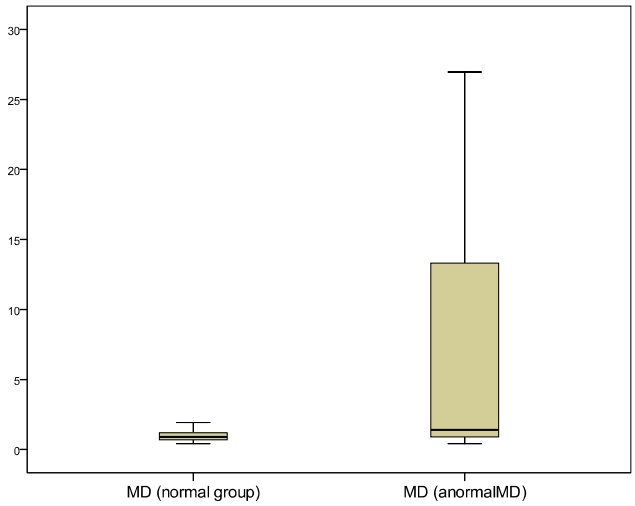

Figure 2.2: MD Values of Normal and Abnormal Group

Then, important variables are selected for the model by using signal-to-noise (S/N) ratios and orthogonal arrays (OAs) based on MD values. The normal or abnormal groups may not be distinguishable due to the improper selection of variables. Each variable in an OA is assigned to one of its column and set with two levels, using and not using the variable. There are three general forms of S/N ratios: **(i)** larger-the-better type, **(ii)** smaller-the-better type and **(iii)** nominal-the-best type. Taguchi and Jugulum (2000) encourage using larger-the-better S/N ratios instead of nominal-the-best type when the true levels of abnormal group are not known.

$$\text{Larger - the - better S/N ratio} = -10 \log \left(\frac{1}{N} \sum_{i=1}^{N} \frac{1}{MD_i^2}\right) \tag{2.5}$$

The steps of the original MTS algorithm:

1. **Data pre-processing steps:**

    a. Collect N observations with two-classes. MTS empirically requires that the number of observations in the normal class is greater than the number of variables, *k*.
    b. Let x_{ij} be the value of i^{th} observation for j^{th} variable (i = 1, . . . , *N*, *j* = 1, . . . , *k*). The vector of variable values is $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ik})^T$.
    c. Calculate the mean of variable values, μ_j, and the standard deviation, σ_j, for each variable x_j in the normal group.
    d. Normalize or standardize observations $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ik})^T$ with μ_j and σ_j as given in Equation (2.4).
    e. Use the square distance of MD given in Equation (2.3):

$$MD = D^2 = \frac{\mathbf{z}_i^T \mathbf{C}^{-1} \mathbf{z}_i}{k}$$

    f. Omit observations having MD values more than $(\overline{X}_{md} + 3s_{Xmd})$ as outliers. Here, $\overline{X}_{md}$ is the average of and s_{Xmd} is the standard deviation of observations in the normal group. MTS requires that the normal class has a high uniformity in terms of variable values. We should keep the requirement in the step (a): N > k.

2. **Model construction steps:**

    a. Using S/N ratios and OA, the important variables are selected. The number of rows in an OA should be at least (*k+2)*, where *k* denotes the number of variables. Furthermore, the number of columns in the OA must be equal to the number of variables. After S/N ratios are calculated, the gain of a variable is the difference of the average S/N ratio between the situations when the variable is used and not used in an OA. In particular, gain indicates the degree of effectiveness in the classification system after the inclusion of the variable. If the gain is positive, the variable is useful to be included in the model. Indeed, the gain of the variable is equivalent to an estimated the main effect of the variable in statistical design of experiments terminology. The presence and the absence of the variables are considered as the levels of an OA in the MTS method. Level-1 in the OA column represents the presence of a variable and Level-2 represents the absence of that variable. S/N ratios are calculated using the levels of OAs based on MD values. Considering the S/N ratios of presence and absence of a variable, if the difference is positive, the variable is included in the model.
    b. Calculate a threshold value, τ of MD for classifying classes. An appropriate threshold, which separates the abnormal observation from the normal group or MS, is found. Consequently, class assignments are done based on the threshold.

3. **Classifying a new data:**

    a. If a new observation has an MD smaller than τ, it is assigned to the normal class. Otherwise, it is in the abnormal class.

The procedure of the original MTS is given in Figure 2.3.

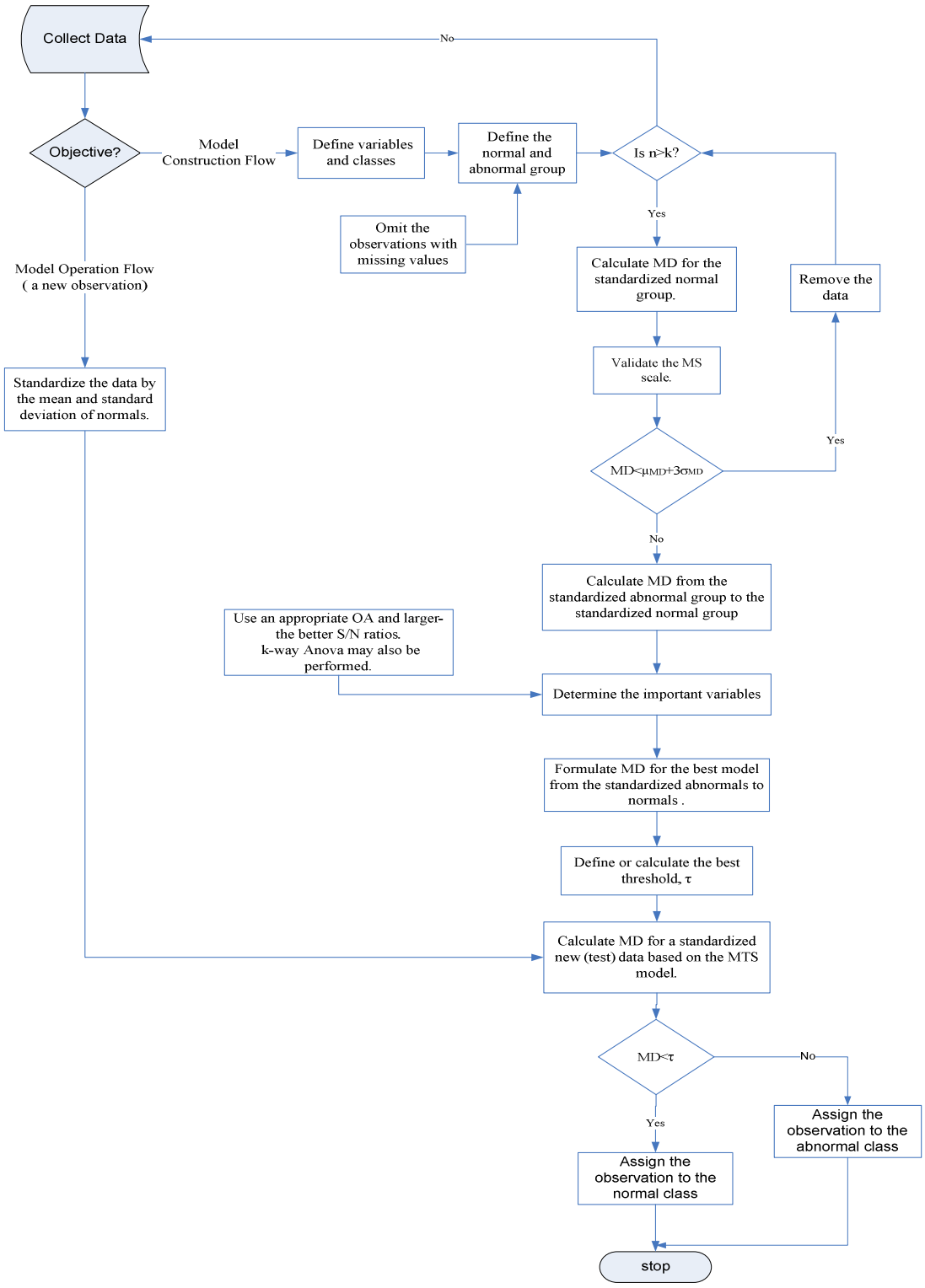

Figure 2.3: The Flowchart of the Original MTS Method

2.1.5 Gram-Schmidt MTS

The difference between Gram-Schmidt MTS (GSMTS) and the original MTS comes from the MD calculation. Gram-Schmidt (GS) process is especially used to obtain better MDs if the observation size is small, and there are multicollinear situations where the correlation matrix is singular. GS process is performed to make variables mutually orthogonal. This process eliminates their relationship (multicollinearity). This makes the covariance matrix almost singular and the inverse matrix invalid. The theoretical background of GSMTS is explained as below.

Given linearly independent standardized vectors of $\mathbf{z}_i$ (i=1,…, k), there exist mutually perpendicular vectors such that:

$$\mathbf{U}_1, \mathbf{U}_2, ..., \mathbf{U}_k$$

$$\mathbf{U}_1 = \mathbf{z}_1 \tag{2.2}$$

Let $\mathbf{z}_1$ is the first standardized variable vector for observations, $\mathbf{z}_2$ is the second and $\mathbf{z}_k$ is the last variable vector. Then, the orthogonal GS vectors of $\mathbf{U}_i$ are:

$$\mathbf{U}_2 = \mathbf{z}_2 - \left(\frac{\mathbf{z}_2' \cdot \mathbf{U}_1}{\mathbf{U}_1' \cdot \mathbf{U}_1}\right) \cdot \mathbf{U}_1$$

$$\mathbf{U}_k = \mathbf{z}_k \quad .... \quad \left(\frac{\mathbf{z}_k' . \mathbf{U}_{k\text{-}1}}{\mathbf{U}_{k\text{-}1}' . \mathbf{U}_{k\text{-}1}}\right) \quad \mathbf{U}_{k\text{-}1} \tag{2.3}$$

For the perpendicular vector for the k^{th} variable, $\mathbf{U}_k$, there must be *(k-1)* GS vector coefficients, denoted as u_i,

$u_1, u_{2,}, \ldots, u_{k\text{-}1}$

Such that;

$$u_{k\text{-}1} = \frac{\mathbf{z}_k'.\mathbf{U}_{k\text{-}1}}{\mathbf{U}_{k\text{-}1}'.\mathbf{U}_{k\text{-}1}} \tag{2.4}$$

Then, the GS vector for the last variable is:

$$\mathbf{U}_k = \mathbf{z}_k - u_1\mathbf{U}_1 - u_2\mathbf{U}_2 - \ldots - u_{k\text{-}1}\mathbf{U}_{k\text{-}1} \tag{2.5}$$

After the calculation of all GS vectors, MD is computed by using the following formula, which is derived from the original MD in Equation (2.3):

$$\mathrm{MD}_j = \frac{1}{k}\left(\frac{\mathbf{u}_{1j}^{\,2}}{s_1^{\,2}} + \frac{\mathbf{u}_{2j}^{\,2}}{s_2^{\,2}} + \ldots + \frac{\mathbf{u}_{kj}^{\,2}}{s_k^{\,2}}\right) \tag{2.6}$$

GSMTS method provides a clear direction of where the improvement efforts should be done. Using this purpose, Srinivasaraghavan and Allada (2006) apply GSMTS in order to evaluate a company's status of lean implementation and success.

Taguchi and Jugulum (2000) employ GSMTS in a medical case study. They prefer not to use principal components (PC), because each PC is a function of the others. However, Hawkins (2003) claims that GS also has the same characteristics since all the GS vectors are a function of the others beforeas it is given in Equation (2.7).

Furthermore, GS is found highly sensitive to data ordering since it depends on which variable is first selected in the order (Woodall et al., 2003). Cudney (2006) finds that GSMTS is not effective because calculations of the S/N ratios are done based on the value of MD, which means the sampling variation is ignored.

In addition, threshold limits of orthogonal vectors corresponding to MD values are given linear; however, it should be in an ellipse form as represented in Figure 2.4.

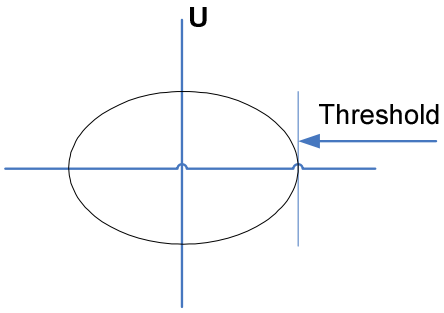

Figure 2.4: Threshold Limits of GS and Ellipse Form of a MS.

There is also modified GS algorithm. Persson (2007) claims that the modified GS (or QR decomposition) is more stable than GS numerically. That is, it is less sensitive to rounding errors. Decomposing a complex $m \times n$ matrix (**X**) (where $m \geq n$) as the product of an $m \times n$ matrix, **Q** with orthogonal columns and an $n \times n$ upper triangular matrix, **R**. In order to find the orthogonal matrix **Q**, there are other methods such as Householder, Givens.

2.1.6 Multi-class Classification with MTS

There are different approaches in classification, especially for binary classification. However, multi-class classification is relatively less investigated (Hsu and Lin, 2002). In the literature, multi-class problems are solved differently, which can be categorized in three types (Su and Hsiao, 2009). The first approach does not need any modification in the two-class classification method to solve multi-class problems. This employs only one run to solve the problem, thus it is simple and time saving. However, this type of algorithm is not encountered much (i.e Mahalanobis Distance Classifier, Decision Tree). The second type is a modification of the original algorithm by considering it as if it is a two-class problem (i.e. some of researches in Support Vector Machines (SVM)). The problem is solved by one model as the first approach. However, in this case the algorithm is exposed to changes (Su and Hsiao, 2009). The last one is

algorithms that decompose the problem into a collection of two-class problems. This has three variations in itself, for each of which Ou et al. (2004) develop some algorithms using the neural networks: **(i)** "one-to-one", which considers each pair of class. It needs *L (L-1)/2* models to solve the problem, where *L* is the number of classes. This approach is used by Friedman (1996) on Support Vector Machine. He also proposes a method for making class assignment after modeling; (**ii**) "one-to-all", such a system that employs *L* number of models. This procedure is repeated for each of the *L* classes, leading to *L* two-way trained classifiers (Ding et al., 2001). Although it considers the data at once, an optimization is required for them since it requires more computational time than binary one. Based on this purpose, Hsu and Lin (2002) study a decomposition implementation for SVM algorithm with this approach. (**iii**) "p-to-all" method, on which Ou et al. (2004) also study. For detailed comparisons, readers may refer to Chin (1998) (as cited in Hsu and Lin, 2004).

MTS is originally designed for binary classification. How it can be effectively extended to multi-class classification is still an ongoing research. In 2009, Su and Hsiao propose a MTS method for multi-class problems, namely Feature Weighted Multi-class MTS (FWMMTS) of Su and Hsiao (2009). GS process is employed for MD calculation.

The steps of multi-class MTS algorithm proposed by Su and Hsiao (2009):

1. For each class, construct the original MTS model as given in Section 2.1.4. Normal group constructs a MS, for which the model is obtained. The other classes are left as abnormal.
2. The GS orthogonal vectors of abnormal and normal groups are computed. By using GS vectors, MD of the abnormal group to the MS is calculated using Equation (2.8).
3. OAs and S/N ratios are used to select the most important variables. Different from the original MTS, weights are calculated based on S/N ratio gains. Gains of the important variables show the degree of their effectiveness in the classification system with respect to its inclusion. The motivation to use the weights for variables comes from equal adding up of the variance normalized squared distance of the variables during the MD calculation.
4. The MD is the sum of products of MD for each variable by its weights.

5. Up to this step, the process is done in the same way for each class. Finally, method is ready to accomplish the classification of the new data to the class of the minimum MD.

Su and Hsiao (2009) compare their method with the other well-known methods using Balance Class Accuracy (BCA) as a performance measure given in Section 2.1.2.2. They find out that their proposed method is as accurate as the Support Vector Machine (SVM).

2.1.7. Applications of MTS

Using MTS, researchers have addressed some of problems such as diagnostic purposes, inspection, fire detection, sensor systems in manufacturing, patient monitoring, forecasting, weather forecasting, credit scoring, and voice recognition.

Some popular applications in the literature of MTS are as follows: Taguchi and Jugulum (2000) utilize MTS to make classification in a medical case study. Additionally, Watabe et al. (2005) detect specific scene within a short time in the digital video storage by means of MTS. Riho et al. (2005) implement MTS for identification of the important parameters in the wafer failure process. Moreover, Aman et al. (2006) try to control the maintenance cost by finding cost prone classes with MTS. In the meantime, forecasting of customer satisfaction ratings is done in a vehicle handling system by Cudney et al. (2006). Das and Datta (2007) ascertain effects of chemical composition of hot rolled steel product whether its quality is “ok” or “diverted”.

2.2 DELIMITATIONS OF MTS

Although MTS has used in various areas, Woodall et al. (2003) do not find MTS easy to implement. The major drawbacks of MTS are summarized as below:

1. Distribution of Variables: MTS is applied on normal distributed data, so has MD. In addition, Woodall et al. (2003) also claim that in the study of Taguchi and Jugulum (2000), the impact of sampling on MTS is unclear.

2. Number of Classes: MTS is proposed for two-class problems. There is one recent study on multi-class problems, proposed by Su and Hsiao in 2009. This study is mentioned in Section 2.1.6. We intend to fill this gap in Chapter 4.
3. Sample Size and Multicollinearity: The number of normal observations should be large enough in order to run the MTS algorithm as an empirical requirement. This delimitation is the motivation of Chapter 3. In addition, variables of the normal group, which constitute a correlation matrix, should not be highly correlated between each other. One solution to this is using statistically independent variables such as GS orthogonal vectors.
4. Selection of Important Variables: OA and S/N ratios are used to find the significant variables. However, MTS solution changes according to allocations of variables in an OA. Thus, the variable selection procedure of the original MTS is problematic.
5. Threshold Determination: Threshold determination is not clear.

Discussions on these drawbacks are explained separately in the following sections.

2.2.1 Distribution of Variables

According to Taguchi and Jugulum (2000), the abnormal group does not constitute a separate population. In addition, Hawkins (2003) underlines that MTS approach may avoid distributional models. As a result, this may warrant the usage of MD as a nonparametric quantity in MTS. This subject is given in detail in Section 2.1.3. However, Woodall et al. (2003) find this unacceptable in the statistical terminology. Because, the main assumption of MD is that the variables are multivariate normal, the MS must be made of a normal group, variables of which follow a normal distribution (Abraham and Variyath, 2003).

The normality of distributions can be evaluated by several ways, such as dot plots for smaller observations, histograms for $n>25$, where n denotes the number of observations, and also a Q-Q plot. As an alternative, symmetry of histograms or nearly straight line of Q-Q plots might indicate the data are normally distributed. When the normality assumption is not satisfied, one alternative is to continue as if it is normally distributed. However, Johnson and Wichern (1998) do not encourage this since it may lead to bad conclusions. The other strategy is making a transformation on data.

A major limitation inherent in MTS is that MD based boundary fails to discriminate data in cases, which mean points of classes stay close to each other (Aman et al., 2006; Abraham et al., 2003). This prevents a clear separation of the two groups, which overlap with each other on a scatter diagram.

2.2.2 Number of Classes

In the literature, there is just one recent study on multi-class MTS, which belongs to Su and Hsiao (2009). This study is mentioned in Section 2.1.6. In this thesis, we also develop new multi-class classification algorithms in Chapter 4.

2.2.3 Sample Size and Multicollinearity

Sample size problems in MTS can be classified in three types. In the first type, the observation size is large enough, but less than the number of variables. As a result, it poses an obstacle to calculate MD. This problem can be solved by increasing the observation size in a homogeneous way. In the second type, data representation with respect to other classes causes problems, which are known as data imbalance problems. This may occur in two ways: an over-represented class, in which the number of observations in one class is much more than the other classes and an under-represented class, in which the number of observations in one class is very less than the other classes. Re-sampling procedures which are over-sampling or under-sampling are generally run for this problem. In the third type, both observation size and data imbalance problems may happen at the same time. As a result, the over-sampling algorithms should also consider the number of variables. This point imposes a new restriction for re-sampling issue in MTS, which we are intended to relax in Chapter 3.

A small observation size may cause multicollinearity problems in MTS. MTS uses the correlation matrix of normal observations. Multicollinearity and singularity are the problems of the correlation matrix that occurs when variables are highly correlated. Correlation matrix is a symmetrical matrix, where each element represents the correlation between two variables. While analyzing the correlation matrices, variables whose correlations greater than 0.9 can be considered multicollinear and those correlation matrices, whose correlations are equal to 1.0, can

be considered singular (Tabachnick and Fidell, 1996). Multicollinearity implies that the variables are highly correlated, on the other hand, singularity indicates that the variables are combinations of each other and redundant. When these problems occur, a solution with MTS cannot be obtained. The main problem due to the singularity and multicollinearity is rank deficiency. It effects the matrix inversion, or division. In fact, when the matrix is singular, because the determinant of the matrix is zero, it prohibits the matrix inversion. When the matrix is multicollinear, the determinant is not exactly zero, but very close to it. As a result, the inverted matrix becomes unstable and fluctuates enormously with only the minor changes in the correlations of variables. An unstable inverted matrix causes unstable multivariate solutions because of the large error terms. Besides, if variables are highly correlated, the marginal contribution of variables cannot be analyzed. This is the case, which interpretations of variables are often not warranted. In order to prevent multicollinearity problems without any structural analysis (i.e. principal components, factor analysis), it is advisable to examine correlations between the variables before analyzing, since the redundant variables inflate the error terms by weakening the analysis as illustrated in Appendix A.1.

Measures, which are widely used to detect multicollinearity in the statistics and in the numerical analysis, are “variance inflation factor (VIF)” and “condition number” (the ratio of the largest eigenvalue to the smallest eigenvalue). In addition to these detection methods, as a very simple way, if the determinant of the correlation matrix is very close to zero, it reflects multicollinearity. Another way to diagnose the multicollinearity is to regress each of the predictors, denoted as $\mathbf{X}_j$ on all the others.

Possible solutions to multicollinearity problems;

1. Correlations between the variables and importance of the variables are analyzed to decide which variables to drop from the model. If all the independent variables are to be kept in the model, then, this avoids making inferences about relationships between response and variables (Mendenhall and Sincich, 2003).
2. The impact of multicollinearity can be reduced by collecting more data or increasing the observation size.
3. The factor analysis and principle components also reduce multicollinearity by centering the variables.

4. The variables can be centered by computing the mean of each independent variable, and then the difference of the observation from its mean is taken. Then, it is divided by the standard deviation of each independent variable.
5. A final approach as a remedy for multicollinearity is to conduct 'ridge regression'. Ridge regression involves transforming all variables in the model and adding a biasing constant to the $\mathbf{X}'\mathbf{X}$ matrix.

In the literature, Taguchi and Jugulum (2000) propose GSMTS as a solution to multicollinear MTS problem. However, there are some critics on it, which are mentioned in Section 2.1.5. However, a modified GS algorithm, or QR decomposition, is considered more stable than GS. It is also used to prevent multicollinearity problems. As a second alternative for MTS, "adjoint matrix" method is proposed by Cudney et al. (2006) in order to calculate MD. This method uses the adjoint of the correlation matrix instead of the matrix division to address the issue of multicollinearity. Its formula is provided in the Appendix A.2. In addition, pseudo-inverse is advised for the cases, where it is not feasible to obtain more data, since then, the data contains a limited amount of information and one must simplify the model accordingly. Inversion of a noninvertible singular matrix (rectangular matrix) can be done with pseudo-inverse (or the Moore-Penrose generalized inverse) given in Appendix A.3. However, it does not provide the accurate solution.

2.2.4 Selection of Important Variables (OAs and S/N ratios)

Variable selection, which eliminates the number of variables, is an active research area in pattern recognition, statistics, and data mining. It can significantly improve the comprehensibility of the resulting models and often build a model that generalizes better in terms of accuracy and simplicity. This process is performed by means of OA and S/N ratios in the original MTS.

OA is a table listing all the combinations of the variables. The presence and the absence of the variables are considered as two levels of the OA in the MTS method. An OA consists of orthogonal vectors. These vectors exhibit the following properties:

(1) They are perpendicular to each other and given a vector $\mathbf{X}$, $\mathbf{X}^T\mathbf{X}$=I where I denotes identity matrix. Thus, $\mathbf{X}^T$ is equal to $\mathbf{X}^{-1}$

(2) Since vectors are mutually perpendicular to each other, they are also statistically independent from each other.
(3) Each vector conveys unique information, which avoids the redundancy.

Although OAs are encouraged by Taguchi et al. (2003) because they can make predictions with a limited number of experiments and combinations of variables, the usage of OAs in variable selection is a bit puzzling. Hawkins (2003) states that when there are suppressor variables, the resolution of an OA becomes very important. Abraham and Variyath (2003) state that when allocations of variables in the OA change, different main effects of variables are obtained. In fact, Woodall et al. (2003) claim that OAs are not suitable in variable selection, since combinations of the variables in an array change the solution. OAs may not provide the exact optimal ordering of variables since they give only fractional factorial design (Woodall et al., 2003). Thus, different statistical procedures are encouraged to find interaction effects.

Selection of variables by statistical tests rather than by OAs is recommended in the literature. Abraham and Variyath (2003) apply the forward selection procedure with S/N ratios and get better results than OAs in terms of low variability and large S/N ratios. Furthermore, a stepwise procedure which is applied in the study of Mason et al. (1997) is encouraged by Hawkins (2003).

Using gains of S/N ratios to detect the significant variables has also some drawbacks on it. In the original MTS, having a gain larger than zero is enough for the variable to be included in the model. However, a variable with a gain value very close to zero is not expected to be significant. Thus, it may not be include in the model. We try to include the significance of variables by using ANOVA, together with S/N ratios while developing the models in Chapters 3 and 4.

2.2.5 Threshold Determination

MD values are evaluated with a threshold, below which an acceptable MD value is required for an observation to classify in the normal group. Taguchi and Jugulum (2003) propose the calculation of quadratic loss function to find a threshold such that the losses due to two values of classification errors are balanced in some sense. However, Abraham et al. (2003) criticize it because there may be some difficulties in cost determination and misclassification. Furthermore,

Woodall et al. (2003) claim that there is no clear explanation about probabilities of misclassifications and the threshold determination in the study of Taguchi and Jugulum (2003).

Su and Hsiao (2007) claim that an appropriate threshold is very remarkable for MTS to carry out the classification process effectively. They show that the selection of threshold also affects the class imbalance sensitivity. As a result, they propose the "probabilistic thresholding method" (PTM) by utilizing the Chebyshev's theorem. The procedure is as follows: they find the percentage of normal group with MD smaller than the minimum MD of abnormal group including a parameter for omitted outliers of normal group. Thus, a parameter, which becomes the upper bound boundary to apply the Chebyshev's theorem, achieves the maximum accuracy.

Yenidünya (2009) also studies the threshold determination in MTS for two-class problems. In the study, several methods such as G-mean, PCC, recall, PTM, F measure, AUC are searched for the best threshold levels. A 3-fold and 3-replicated stratified cross validation (SCV) is used to compare the results of different methods. The results show that G-mean is better in balancing the accuracy of each class, whereas PTM predicts one of the two classes worse. Thus, G-mean is selected as the threshold method in terms of recall and sensitivity. This result is especially useful for imbalanced data sets.

2.3 RE-SAMPLING

A data set is considered imbalanced if classes are not (approximately) equally represented (Chawla et al., 2002). Imbalance problems occur when a classifier tries to detect a rare but an important case, such as fraudulent telephone calls, oil spills in satellite images, failures in a manufacturing process, or rare medical diagnoses (Barandela et al., 2003). In addition, in many real situations, obtaining observations of training set must be limited because of the cost of learning such as obtaining raw data, pre-processing data or storing data (Turney, 2000 as cited in Weiss et al., 2003). In fact, most quality data sets are described as small and imbalanced. By convention, for imbalanced data sets, the classes having more observations are the majority classes and the ones having fewer observations are the minority classes.

Although some practitioners believe that the natural class distributions should be used for modeling, an imbalance situation makes typical classifiers difficult to optimize the overall

accuracy, when they mostly consider the relative distribution of each class. As a result, classifiers tend to ignore small classes while concentrating on classifying large ones accurately. High complexity, imbalance class, and small data set sizes give rise to some very small sub-clusters; consequently, they cannot be classified accurately. In addition, the class imbalance problem causes a classifier to over-fit the data belonging to the class with the greatest number of training observations (Nickerson et al., 2001).

Japkowicz and Stephen (2002) state that the class imbalance problem depends on **i)** the degree of class imbalance; **ii)** the complexity of the concept represented by data; **iii)** the overall size of the training set; and **iv)** the type of the classifier involved.

If a re-sampling approach keeping the existing algorithm unmodified is used, the following alternatives are suggested (Estabrooks et al., 2004); **(i)** over-sampling which consists of copying existing training observations at random and adding them to the training set until a class balance is reached, **(ii)** under-sampling which consists of removing existing observations randomly until a class balance is reached, **(iii)** a combination of over-sampling and under-sampling, which cause both increase and decrease in the data size.

These alternatives have their own advantages and disadvantages. The advantage of over-sampling is that no information from the original training set is lost since all the original data is preserved. However, increasing the size of the training set also increases the training time and the amount of memory required holding the training set, which is a disadvantage.

In addition, some over-sampling methods that duplicate observations of the minority class lead to over-fitting, while under-sampling methods eliminate a large amount of potentially useful information. Previous studies have not reached any conclusive result about which is best in classification performance (Liu et al., 2004). This proves that the choice of the re-sampling method is probably specific to data set and problem (Liu, 2004).

Japkowicz and Stephen (2002) propose a re-sampling algorithm by considering the complexity while generating new observations. The procedure is as follows: given a complexity level, the range of the response is divided into some intervals. The generating points are then randomly selected from intervals.

Weiss et al. (2003) apply a new re-sampling methodology to detect the best class distribution, which also gives the relationship between the class distribution and the classifier performance on seven data sets. They perform the re-sampling with the ratio of class distribution before and after. They gain 10.6 % relative reduction in error rate. Another result of the study indicates that, best class distribution depends on the performance measure. In fact, when AUC is selected as the measure, the best class distribution is found to be near to the balanced class distribution, whereas it is found the original distribution when the accuracy is selected as the measure.

Training set size is also a factor in the classifier's ability to deal with imbalanced data sets (Japkowicz and Stephen, 2002). Similarly, Weiss et al. (2003) search for the best training set size, which gives the best performance. We also study the relation between the training set size and the performance of our re-sampling algorithm in the next chapter.

The class imbalance problem affects the performance of the classifier (Estabrooks et al., 2004). Re-sampling is used for the class imbalance problems in order to increase the classification performance. Estabrooks et al. (2004) search for the rate of re-sampling. They try the under-sampling of majority class, as well as the over-sampling of minority class on the imbalanced training data set. Then, the technique is tested by some learning classifiers on data sets with various degrees of class imbalances.

As a different method, Chawla et al. (2002) attempt to solve the imbalance problem with SMOTE, an over-sampling method by generating the synthetic observations between the nearest neighbors of observations in the minority class. It generates synthetic examples by operating in a "variable space" rather than "data space". For every minority example, its nearest *k* neighbors of the same class are determined, and then some of *k* neighbors are randomly selected depending on the over-sampling rate. After that, new synthetic observations are generated along the line between the minority example and the selected nearest neighbors. Synthetic samples are generated in the following way: Take the difference between the variable vector under consideration and its nearest neighbor. Multiply this difference by a random number between zero and one, and add it to the variable vector under consideration..

Table 2.2: Example of Generation of Synthetic Examples (SMOTE).

Consider (7, 4) is the observation for which k-nearest neighbors are being identified. (6, 5) is one of its k-nearest neighbors.

Let $v_{i,j}$ denotes the j^{th} variable of the i^{th} observation:

$v_{1,1} = 7 \quad v_{2,1} = 6 \quad v_{2,1} - v_{1,1} = -1$

$v_{1,2} = 4 \quad v_{2,2} = 5 \quad v_{2,2} - v_{1,2} = 1$

The new observations will be generated as:

$(v_{3,1}, v_{3,2}) = (7, 4) + \text{rand}(0\text{-}1) * (-1, 1)$

rand (0-1) generates a random number between 0 and 1.

As an advantage of SMOTE, it makes the decision regions larger and less specific (Huang et al. 2005). Furthermore, borderline observations are apt to be misclassified than the ones far from the borderline. Based on this analysis, Huang et al. (2005) develop an algorithm, namely Borderline SMOTE. Different from the other over-sampling methods, they over-sample only the borderline minority observations.

In addition, Kubat and Matwin (1997) employ an under-sampling of the majority class while keeping the original population of the minority class constant.

Re-sampling until the majority and minority classes have equal prior probability may not yield optimal results (Weiss et al., 2003). The amount of over-sampling is generally considered as a parameter of the system (Estabrooks et al., 2004; Weiss et. al. 2003). Furthermore, the best re-sampling rate changes according to the data studied and the re-sampling type (over or under) (Estabrooks et al., 2004; Japkowicz, 2004 as cited in Liu, 2004). This makes difficult to find a rule for re-sampling. Liu (2004) obtains different results as the level of re-sampling changes. In particular, some re-sampling methods might perform better with a higher or lower amount of re-sampling.

Weiss et al. (2003) state that before a re-sampling on a classifier, the sensitivity of the classifier should be checked. This can be checked with measures of sensitivity and specificity, which give opinions about the accuracy on the positive and negatives classes. For the case of MTS, sensitivity to class imbalance problems is just studied by Su and Hsiao (2007). They try to find a new threshold method for MTS, in which the classification performance is not influenced by an imbalanced data. MTS is expected to be sensitive to the number of data of each class. When the number of observations is not enough, there may be multicollinearity problems in MTS as explained in Section 2.2.3. A solution may be increasing the data size. Results of Su and Hsiao (2007) indicate that the selection of threshold eliminates the sensitivity of MTS to imbalanced data.

The classification performance of an imbalanced data set should not be measured with accuracy since this parameter covers the accuracy of majority class excluding the overall accuracy of minority class. Thus, even if the algorithm classifies all the majority observations correctly and misclassifies all the minority observations, the accuracy of the method is still high because there are much more majority observations than minority observations (Huang et al., 2005). Thus, in this case, the classification performance of re-sampling algorithms is usually measured by precision and recall, or F measure that combines both of them (Barandela et al., 2003). Additionally, a ROC curve is mostly preferred due to its independence of the distribution of observations between classes (Kubat and Matwin, 1997). Su and Hsiao (2009) use geometric means of sensitivity and specificity. These measures are formulated in Section 2.1.2. As an alternative, the relative sensitivity (RS), which is the ratio of sensitivity and specificity, is also a measure (Su and Hsiao, 2009).

In the literature, there are re-sampling studies focused on two-class problems. However, multiple class problems are solved by being simplified to two-class problem by using the minority class as a class and the others as a separate class (Weiss, 2004; Su and Hsiao, 2007; Chen et al., 2008).

CHAPTER 3

HANDLING SMALL AND IMBALANCED DATA SETS

In this chapter, the issue of re-sampling is studied in the context of MTS classification on two-class imbalanced data sets. We develop a new re-sampling algorithm, jointly with Berna BAKIR and Barış YENİDÜNYA, to detect the best class distribution and size, which also gives the relationship with the classifier performance on several benchmark data sets.

3.1 THE METHOD

A data set is imbalanced if the number of instances in one class is quite small compared to the other classes. This is the case for many real life problems such as product or process quality improvement, document filtering, gene profiling, and especially in most of real quality problems. For this case, it may be very time and cost expensive to collect data and construct a model on it. Therefore, re-sampling is commonly used as a solution to this problem.

In this part of the study, we aim to develop a re-sampling method for two-class data sets and to relate re-sampling parameters (ratios of classes and data size after re-sampling) to performance measures by considering the initial data size and ratios of classes. Thus, applications are performed on small and imbalanced data sets, in which the classes having more observations are considered majority classes, whereas the ones having fewer observations are the minority classes. This means that the class ratio is a parameter for the degree of the class minority.

Our re-sampling approach, which only works for data sets with two classes, is a combination of over-sampling and under-sampling. In this study, a SMOTE-based re-sampling approach is used for over-sampling. SMOTE (Chawla et al., 2002) is an over-sampling approach, in which the minority class is over-sampled by creating synthetic observations based on *k*-nearest minority

neighbors, where *k* denotes the number of nearest neighbors. Synthetic samples are generated in the following way: The difference between the variable vector under consideration and its nearest neighbor is taken. It is multiplied by a random number between zero and one, and added to the variable vector under consideration. The maximum *k* depends on the initial data size. For example, when there are not many observations in a class, *k* must be small. Thus, we also try to relate the number of the nearest neighbors used in SMOTE with the performance measures. In the literature, SMOTE is used to increase the data size in integer multiples. As a contribution, we oversample the data randomly from the *k*-nearest neighbors as much as required, until the desired class ratio is achieved. In addition, MTS models are used to test the performance of re-sampling for which the most appropriate values are sought for specific to each case.

Steps to develop the re-sampling algorithm:

1. Set the initial parameters such as:

   N_0 : initial number of observations

   r_0 : initial ratio of the minority class

   N_{max} : the maximum number of training observations after re-sampled

   r_{max} : the maximum ratio of the minority class after re-sampled

   k_{max} : the maximum number of neighbors to generate the required data in the minority class (depends on N_0)

   Number of folds, number of replications

   These values are parametric so that they can be changed.
2. After partition data into folds and replications. For each and every fold and replication perform steps 3-10.
3. Check for outliers, check for that the size of the normal group, N (it is required to be larger than the number of variables, m).
4. Calculate the minority class ratio, r.
5. Send the data to three "for" loops:

   1st Loop: Increase the minority class ratio, r, by an increment of 0.1 by under-sampling majority class or over-sampling minority class

   2nd Loop: Increase number of observations, N, by an increment of 50 by under-sampling majority class or over-sampling minority or majority class

3rd Loop: Increase the nearest neighbors (*k*) in SMOTE as explained in Section 2.7, by 1

Throughout these loops, data in minority or majority class can be over-sampled by using SMOTE according to the Table 2.2. In addition, data in majority class can be under-sampled in order to increase the ratio of the minority class.

6. Use re-sampled data in MTS modeling: The steps of the MTS model, which is mixed in the re-sampling algorithm, are explained in Section 2.3.1. It can be summarized as follows: after the data-preprocessing, data is divided into the normal and abnormal groups. Then, MDs of abnormal observations, which give the distance to the normal group, are calculated. Based on the MD values, S/N ratios are calculated with OAs. We normally write $OA_N(s^k)$ to specify such an orthogonal array, which has an array of size N by k, with entries from 0 to s-1. We have used an OA specified as $OA_{200}(2^{100})$ with strength three from the web-site of "A Library of Orthogonal Arrays". In that stage, we use n-way ANOVA analysis along with using variable gains because of the reasons explained in Section 2.2.4. We search for the interval from 0.10 to 0.25 for α-levels because Costanza and Affifi (1979) recommend a significance level of 0.10 and 0.25 as a cutoff value for the variable selection (Sharma, 1996).
7. Calculate the threshold using geometric means of sensitivity and specificity (G-mean) (Yenidünya, 2009): Geometric mean of sensitivity and specificity, G-mean given in Section 2.1.2.1, is chosen for the threshold selection method, since it gives the best class accuracy in both balanced and imbalanced data sets (Yenidünya, 2009).
8. Assign the observation to the abnormal class if its MD value is above of the determined threshold.
9. Find the best N, r and k among all tested values for the highest average and the smallest standard deviation of the G-mean.

Initially with *m* variables, the flowchart of the re-sampling algorithm is given in Figure 3.1.

The re-sampling performances are calculated on the test data, which has the original class distribution, in terms of several performance measures listed in Section 2.1.2.1. G-mean is selected to analyze the results, since it is a combination of sensitivity and specificity. Decision tree classifier is applied to the results to generate rules that relate data set characteristics to the performance of the classifier.

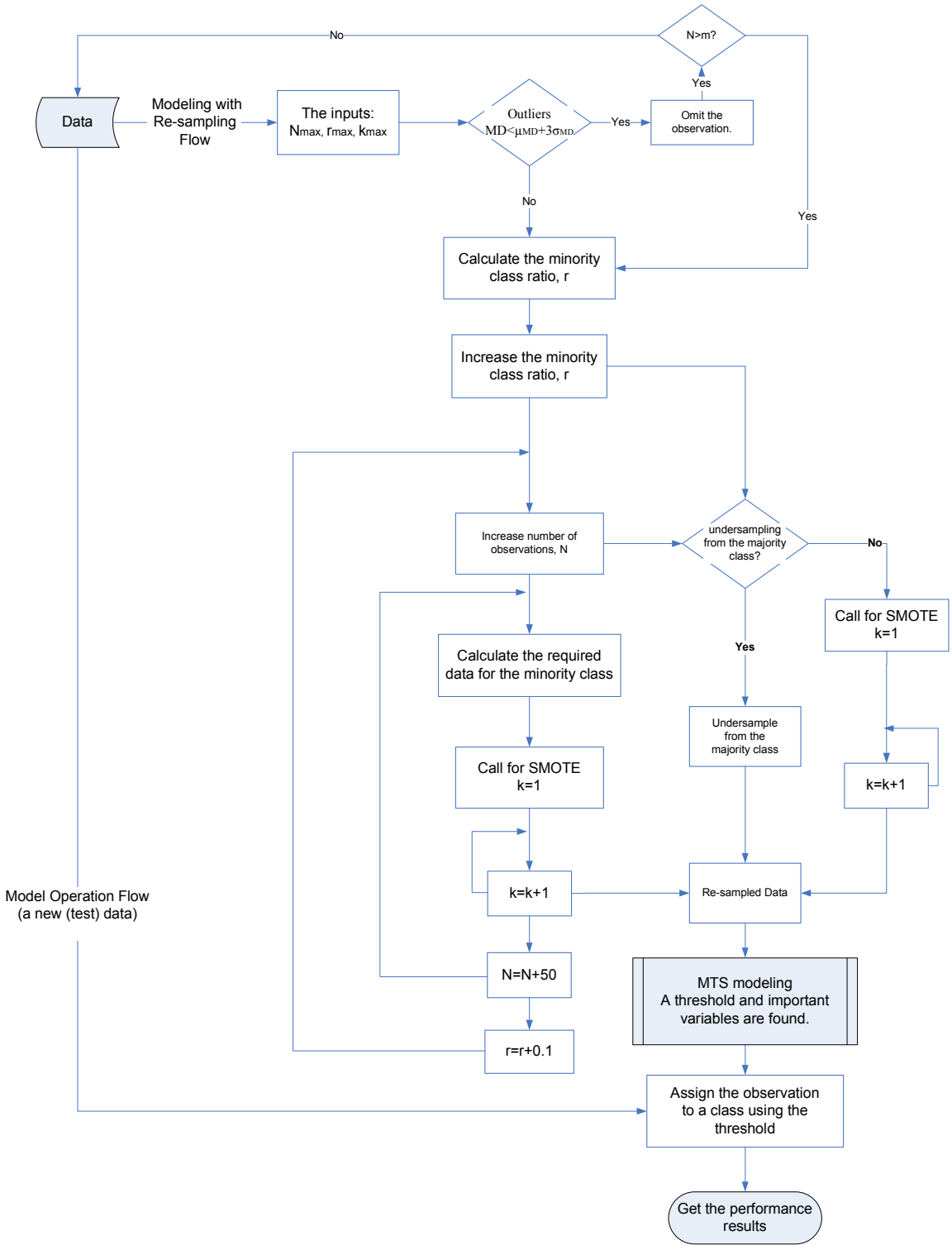

Figure 3.1: The Flowchart of the Re-sampling Algorithm

The model is coded by using *Matlab 7.5*. MD is easily calculated with the "*mahal*" command of *Matlab 7.5*.

3.2 APPLICATIONS AND PERFORMANCE ANALYSIS

Experiments are run to establish the relationship between class distribution, training set size and classifier performance. Data sets are taken from the web-site of the UCI Machine Learning Data Repository. Data sets with non-categorical attributes and high citation rate in scientific articles are tried to be selected.

3.2.1 Applications

The four data sets: blood transfusion, Pima Indian diabetes, magic gamma telescope and Wisconsin breast cancer diagnostic (WBCD) are selected for analysis. The original data set characteristics are summarized in Table 3.1.

Table 3.1: Data Set Information

Data Set Name	**Data Set Number (DS)**	**Data Size**	**Number of Variables**	**Ratio of Minority Class**
Blood Transfusion	1	748	5	0.24
Diabetes	2	768	8	0.35
Telescope	3	19020	10	0.35
WBCD	6	569	30	0.37

The initial parameters of data sets given in Table 3.1 are prepared according to Table 3.2. Hence, the data sets are processed to the initial training data size (N_0): 70, 200 and 500 and initial ratios (r_0): 0.1, 0.2 and 0.3. The re-sampling algorithm is performed on all of the nine combinations of the N_0 and r_0 for each data set according to the re-sampling algorithm given in Figure 3.1.

Table 3.2: Initial Parameters of Data Sets

Overall Size of Training Observations	70 200 500
Class Ratio of Minority Class	0.1 0.2 0.3

To compare the performance measures, which are selected to determine a re-sampling rule, a 3-fold and 3-replicated stratified cross validation is used for each combination. Thus, each class has an equal chance of being one of in the nine folds.

We normally write $OA_N (s^k)$ to specify such an orthogonal array, which has an array of size N by k, with entries from 0 to s-1. We have used an OA specified as $OA_{200}(2^{100})$ with strength three from the web-site of "A Library of Orthogonal Arrays" (http://www.research.att.com/~njas/oadir/).

The parameters of re-sampled data are the class ratio of minority class (r), overall training data size (N), and the number of neighbors (k).

In order to evaluate this sampling algorithm, it is necessary to measure how each of these parameters affects the performance for each combination.

3.2.2. Performance Analysis

Firstly, the results of re-sampling applications on four data sets are plotted to analyze the effect of N, r, k on the average G-mean of nine folds in Figures 3.2-3.7 for the data sets of. They are drawn for two data sets; Diabetes and Telescope, after the parameters of r and N are discretized according to Tables B.1 and B.2.

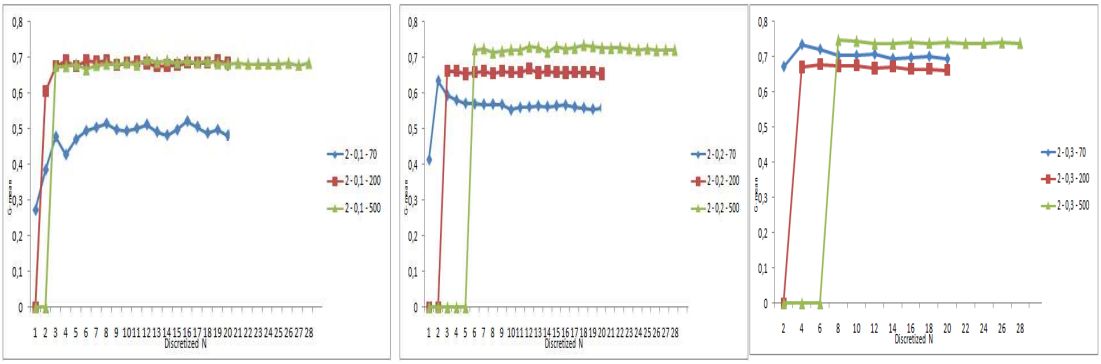

Figure 3.2: Average of G-means versus Discretized N (Diabetes, N_0: 70, 200, and 500)

a) r_0: 0.1 b) r_0: 0.2 c) r_0: 0.3

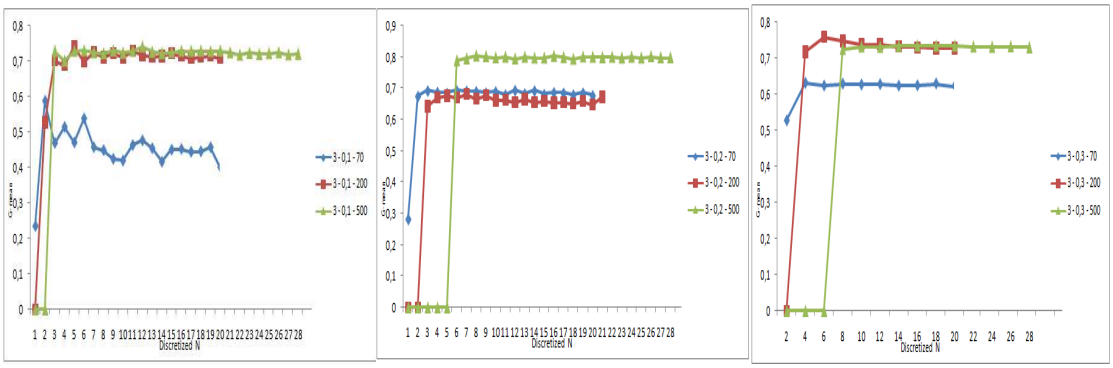

Figure 3.3: Average of G-means versus Discretized N (Telescope, N_0: 70, 200, and 500)

a) r_0: 0.1 b) r_0: 0.2 c) r_0: 0.3

According to Figures 3.2 and 3.3, G-means stay almost the same at a large value as N changes, although there is an increase at first.

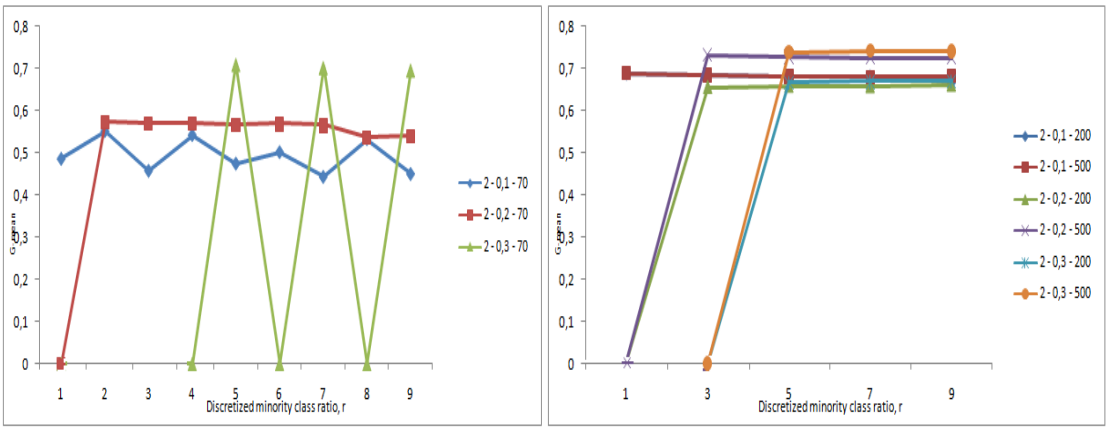

Figure 3.4: Average of G-means versus Discretized Minority Class Ratio, r (Diabetes, r_0: 0.1, 0.2, and 0.3) a) N_0: 70 b) N_0: 200, 500

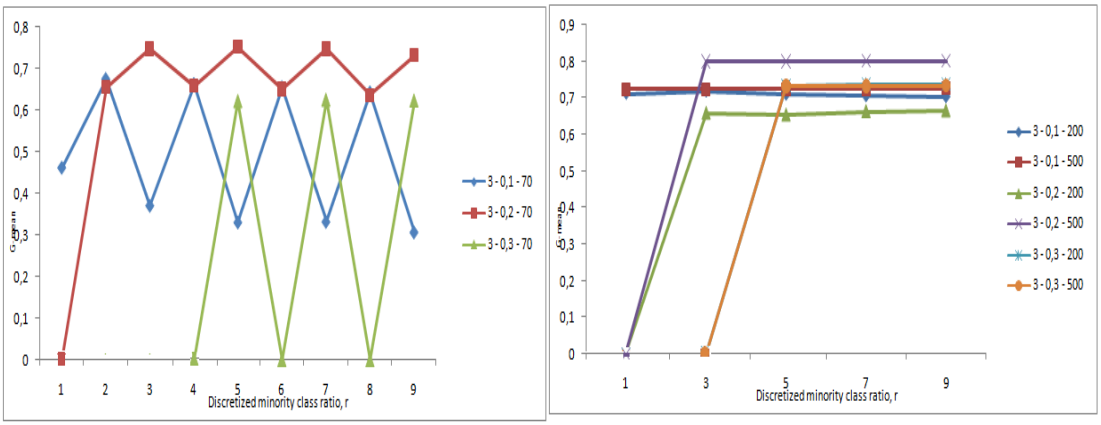

Figure 3.5: Average of G-means versus Discretized Minority Class Ratio, r (Telescope, r_0: 0.1, 0.2, and 0.3) a) N_0: 70 b) N_0: 200, 500

According to Figures 3.4 and 3.5, it is difficult to achieve a concrete assessment. However, it is seen that when the initial data size is larger (N_0: 200, 500), re-sampling increases performance at the first levels of discretized minority class ratios. Then, G-means stay almost constant.

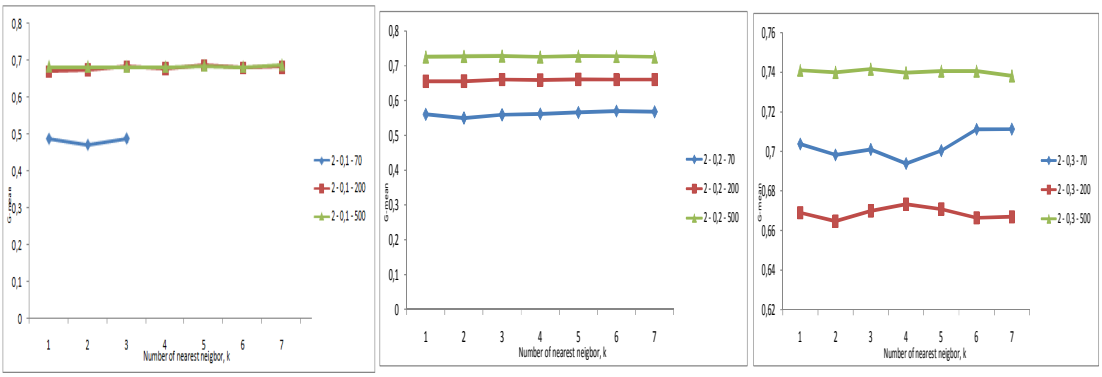

Figure 3.6: Average of G-means versus Number of Nearest Neighbors, *k* (Diabetes, N_0: 70, 200, and 500) r_0: 0.1 b) r_0: 0.2 c) r_0: 0.3

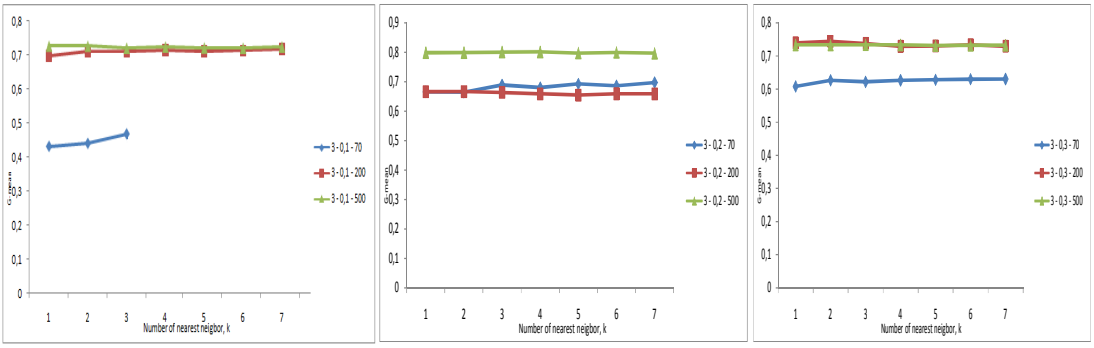

Figure 3.7: Average of G-means versus Number of Nearest Neighbors, *k* (Telescope, N_0: 70, 200, and 500) a) r_0: 0.1 b) r_0: 0.2 c) r_0: 0.3

According to Figures 3.6 and 3.7, it is seen that *k* does not affect G-means, except for the case of N_0: 70, for which *k* should be increased as well.

The overall results of four data sets, given in Table 3.1, are used to achieve a rule for re-sampling with the help of Decision Tree. A decision tree is obtained from *SPSS Clementine 11.1* and given in Figure B.1. According to Figure B.1, it is seen that the results are worse when the original data set is small than larger cases. When N_0 is small, it is good to increase the data set size by over-sampling the data. In addition, the results are better when the over-sampling ratio is very close to the original minority class ratio, when the data is small. When the original data size

is large, the results are better as it is expected to be. In this case, since there are more original data to oversample, better results are seen as the class ratio, r increases.

Based on our re-sampling approach, we can not achieve a rule for the relationship between the initial parameters and re-sampling parameters. Thus, given a case, only re-sampling parameters, which increase the initial performance, can be suggested. For the three data sets in our applications, these are given in Table 3.3. A decision tree based on the suggested parameters of the re-sampling application results is given in Figure B.2. In addition, a plot of parallel coordinates for the values of normalized suggested parameters is given in Figure B.3. It shows the situation of each suggested re-sampling parameter. According to Figures B.2 and B.3, it is seen that, there is no pattern of rule. In fact, the results are very data specific, which makes a rule generation difficult. As a result, for a given data set, suggested re-sampling parameters can be selected after searching for different values of r and N.

Table 3.3: Suggested Re-sampling Parameters for Data Sets; Diabetes, Telescope and WBCD

Data Information Before Re-sampling					**Suggested Parameters (SP) of Re-sampling**					
Data Set (DS)	r_0	N_0	Avr. of G-mean	Std. of G-mean		r	N	k	Avr. of G-mean	Std. of G-mean
2	0.1	70	0.6873	0.0724	SP1	0.1	196	2	0.5771	0.1147
					SP2	0.2	124	1	0.5757	0.0820
					SP3	0.1	296	1	0.5622	0.0841
					SP4	0.1	396	1	0.5597	0.0785
2	0.1	200	0.6873	0.0724	SP5	0.5	76	5	0.7530	0.0740
					SP6	0.1	375	6	0.7295	0.0645
2	0.2	70	0.6521	0.0896	SP7	0.3	32	1	0.6600	0.1157
2	0.2	200	0.6420	0.0526	SP8	0.3	285	1	0.6903	0.0328
3	0.1	70	0.5881	0.2475	SP9	0.2	24	1	0.6765	0.2069
					SP10	0.2	74	3	0.6536	0.1393
3	0.1	200	0.7045	0.1105	SP11	0.1	277	4	0.7615	0.0692
					SP12	0.1	277	2	0.7565	0.0587
					SP13	0.3	193	6	0.7511	0.0822
3	0.2	70	0.6436	0.1089	SP14	0.2	198	5	0.7537	0.0865
3	0.2	200	0.6664	0.0638	SP15	0.4	164	1	0.7011	0.0582
6	0.1	70	0.8749	0.0890	SP16	0.3	444	1	0.9673	0.0302
					SP17	0.5	212	1	0.9583	0.0498
6	0.1	200	0.9300	0.0423	SP18	0.1	425	7	0.9509	0.0402
					SP19	0.1	175	7	0.9462	0.0364
6	0.2	70	0.8018	0.1050	SP20	0.5	310	6	0.9406	0.0535
					SP21	0.3	344	7	0.9404	0.0552
6	0.2	200	0.9602	0.0207	SP22	0.4	316	4	0.9786	0.0266
					SP23	0.4	466	2	0.9783	0.0277
					SP24	0.3	337	7	0.9781	0.0201
					SP25	0.3	437	4	0.9766	0.0264

CHAPTER 4

MULTI-CLASS MAHALANOBIS TAGUCHI SYSTEM METHODS

In this chapter, original methods developed in the thesis for multi-class classification problems based on MTS are presented.

4.1 THE METHODS

The intuition behind the developed methods is explained in this section. Common points for all of the methods are; (**i**) the multi-class classification approach is same in all methods, which it uses "one-to-all" multi-class classification approach given in Section 2.1.6.; (**ii**) data sets are pre-processed in the same way; (**iii**) S/N ratios and OAs are used for variable selection; (**iv**) the class assignment is done in the same way without using a threshold.

As a first common point, the class, for which the one-to-all multi-class classification model is obtained, is selected as the normal group, which also constitutes a Mahalanobis space (MS), while all of the other classes are left as the abnormal one. Thus, giving a multi-class problem with L classes, L two-class problems are obtained by this approach after the original problem is partitioned into a two-class problem.

As a second common point, data pre-processing is performed as the following. First, the variables and classes of the data set are defined and then, the data set is divided into normal and abnormal groups. The normal group represents the selected class for the model, while the abnormal group is composed of the other classes. Observations with missing values are omitted. The data pre-processing is continued with standardization of data with the specifications of the normal group. MS is expected to have homogenous characteristics. As a result, an acceptance

criterion is defined or calculated under which an observation is ensured to be considered in the MS. In fact, we preferred to omit observations having MD values larger than ($\overline{X}_{md} + 3s_{Xmd}$). Here, $\overline{X}_{md}$ is the average and s_{Xmd} is the standard deviation of MD values of the standardized normal group.

As a third common point, in all the methods the S/N ratio corresponding to each run of the OA is computed using the concept of the larger-the-better type as defined in Equation (2.5). OA is a table listing all the combinations of the variables. The size of an OA depends on the number of characteristics and the levels it can take. However, the presence and the absence of the variables are considered as the levels in MTS method. Level-1 in the OA column represents the presence of a characteristic and Level-2 represents the absence of that variable. S/N ratios calculate the gain of a variable when it is included in the model using the levels of OAs.

The last common point is about the way of class assignments. An observation is assigned to a class, which has the minimum MD among the entire MD's calculated from the MTS models of the other classes. This eliminates threshold calculation step of original MTS given in Section 2.1.4. The class assignment is illustrated in Table 4.1.

Table 4.1: An Illustration of the Class Assignment Rule for all of the Methods

Observation	MD for 1st class	MD for 2nd class	MD for 3rd class	Class Assignment
A	2	**1**	4	2nd class
B	**3**	5	6	1st class
C	**4**	**4**	8	1st or 2nd class (randomly)

All of the methods are coded by using *Matlab 7.5*, each of which is explained in the following sections.

4.1.1 Methods Based on the Original MTS

a. Multi-class MTS Method (MMTS)

The first method for multi-class MTS classification is an extension of the original MTS algorithm, which is explained in Section 2.1.4, for multiple class problems.

The algorithm of the multi-class MTS method is given in Figure 4.1, for a given normal group of size n, number of variables k, and number of classes L. According to this algorithm, as a common point of methods, a classification model is developed for each class or MS separately. For this purpose, data is pre-processed for the class under consideration. The size of the normal group (n) should be larger than the number of variables (k). This limitation is considered after data with missing values and outliers are omitted.

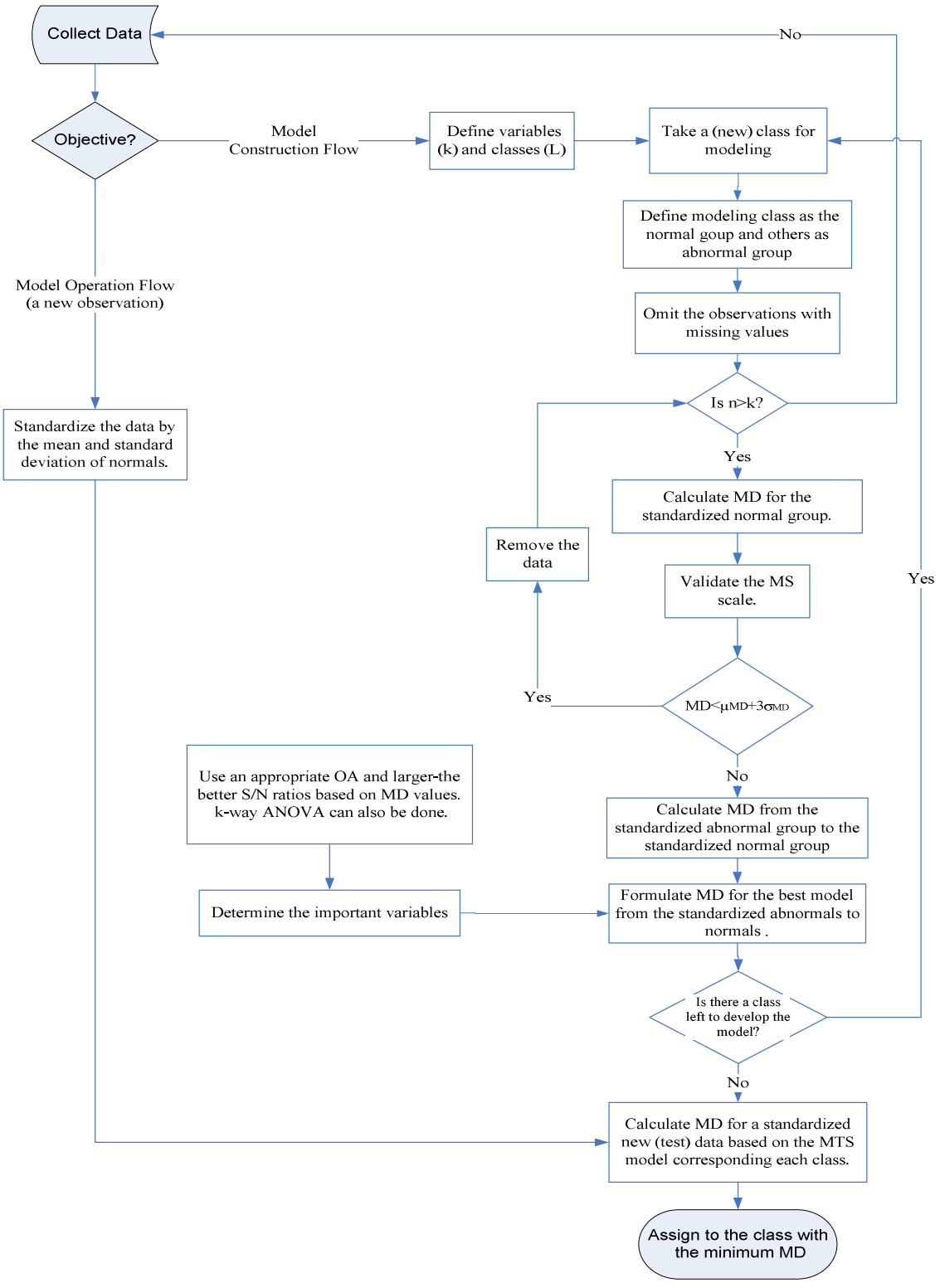

Figure 4.1: The Flowchart of Multi-class MTS (MMTS) Method

After the data pre-processing explained as a common point in the previous Section, MD from the abnormal observations to the normal group are calculated. Then, the calculated MD values are used in detection of important variables, together with OAs and larger-the better S/N ratios, which is given in Equation (2.5). This step is also explained as a common point in Section 4.1. To determine the important variables that should be included in the classification model, we have also used k-way ANOVA, which *k* denotes the number of variables changing for each design combination. For this purpose, the larger-the-better S/N ratios are calculated by using the MDs as explained in the previous Section. Then, for each row in the design matrix or OA matrix, they are analyzed in ANOVA to find whether the variable's S/N ratio makes a significant difference. We search for the interval from 0.10 to 0.25 for α-levels since Costanza and Affifi (1979) recommend as a cutoff value for variable selection (cited as in Sharma, 1996). After developing the MTS model for each MS corresponding to each normal class, the assignment of a new (or test) observation is done according to the rule illustrated in Table 4.1 by using the MTS model.

b. Multi-class MTS Method Based on Gains of Signal-to-Noise Ratios (SNRMMTS)

This method differs from the MMTS method in selection of important variables. Here, only gains of S/N ratios of variables are considered as in the original MTS method.

4.1.2 Feature Weighted Multi-Class MTS-I (FWMMTS-I)

This classification algorithm is developed for multi-class problems with the motivation of "Descent Feature Weighted Mahalanobis Distance" proposed by Wölfel and Ekenel (2005). This study tries to give weights to each variable in the MD calculation. Because the variables with large contributions to the MD can mask all the other variables, the classification considers only these noisy variables. Based on this idea, a new multi-class MTS classification algorithm, namely Feature Weighted Multi-class Mahalanobis Taguchi System (FWMMTS-I), is developed.

In this method, a MS corresponding to each class is constructed and data pre-processing is done in the same way as before. Next, the weighted MD is computed for each given normal group *j* as follows:

1. Adopting the weighted MD formula for our algorithm, let us use *i* for the observation in the standardized abnormal group and *j* for the standardized normal group. $D_{i,j}[c]$ denotes the distance of observation $x_i[c]$ in the abnormal group, to the mean $\mu_j[c]$ of the observations in the normal group *j* corresponding variable *c*.

Let $\sum_j^{-1}[c,c]$ denote the inverse of the diagonal value corresponding to the variable *c* of the correlation matrix. Actually, it has a value of one. Then,

$$\forall c,i,j: \quad D_{i,j}[c] = (x_i[c] - \mu_j[c]) \quad \Sigma_j^{-1}[c,c] \quad (x_i[c] - \mu_j[c]) \quad . \tag{4.1}$$

2. Since the goal is to choose weights such that all of the variables, especially noisy ones, have same influence on the MD value, the variables have to be normalized under the consideration of their average MD. Then, for a total of N observations in the abnormal group, an average MD to the standardized normal group, *j*, is calculated, which is denoted as $\overline{D}[c]$:

$$\forall c: \quad \overline{D}[c] = \frac{1}{N} \sum_{i=1}^{N} D_{i,j}[c] , \tag{4.2a}$$

Afterwards, weights are derived from the average MD corresponding to each variable, under two constraints as below.

$$\forall c: \quad w[c] \geq 0 \tag{4.2b}$$

$$\sum_{c=1}^{k} w[c] = k , \tag{4.2c}$$

where k represents the number of variables.

In order to satisfy the constraints, the distances are inverted and then normalized as in Equation (4.3a) for each variable *c*, by using variables *a*=1,…, *k*.

$$\forall c: \quad w[c] = \mathrm{k} \frac{1/\overline{\mathrm{D}}[c]}{\sum_{a=1}^{\mathrm{k}} 1/\overline{\mathrm{D}}[a]} \quad , \tag{4.3a}$$

where

$$w_{i,j}[c] = w[c] \quad , \forall i, j \, . \tag{4.3b}$$

Weights of variables have to be recomputed for each MS or the standardized normal group under consideration for modeling.

3. Lastly, weighted MD from observation *i* to class *j*, $MD_{i,j}^{weighted}$ is obtained by a sum of products of weight of each variable ($w_{i,j}[c]$) and MD value ($D_{i,j}[c]$):

$$\mathrm{MD}_{i,j}^{weighted} = \sum_{c=1}^{\mathrm{k}} (\mathrm{w}_i[c] . \mathrm{D}_{i,j}[c]) \, . \tag{4.4}$$

The values of $\mathrm{MD}_{i,j}^{weighted}$ are used to calculate the important variables. For this purpose, as a third common point of the methods explained in Section 4.1, S/N ratios together with an OA are utilized. During the variable selection process, k-way ANOVA is used as explained in the previous Section. Consequently, the model based on the normal class is achieved. Up to these steps, the procedure is followed for each of the classes by considering each as a normal group under consideration. Finally, MTS model of each MS is used to calculate MD of a new (or test) observation. We prefer not to calculate the weighted MD in class assignments. In fact, we see that original MD gives a better performance than the weighted MD in class assignments. This means, weights are only considered in the modeling, in which the important variables are found. Finally, the observation is assigned to the class having the minimum MD.

Our approach to calculation of weights in the FWMMTS-I differs from the calculation of weights in Su and Hsiao (2009). While they calculate the weights based on gains of S/N ratios, in our method, weights are calculated based on MD distances corresponding to variables. In

addition, our method utilizes weights in the selection of important variables, whereas Su and Hsiao (2009) utilize weights after the model construction.

4.1.3 Feature Weighted Multi-Class MTS-II (FWMMTS-II)

We use the feature weighted MTS approach of Su and Hsiao (2009), which is explained in Section 2.1.6. We prefer to modify it by using the original MD calculation instead of GS calculation due to drawbacks of GS given in Section 2.1.5 such as GS is found highly sensitive to data ordering since it depends on which variable is first selected in the order (Woodall et al., 2003). We name this algorithm Feature Weighted Multi-class MTS Method-II (FWMMTS-II).

4.2 APPLICATIONS AND PERFORMANCE ANALYSIS

In this section, the multi-class methods explained in Section 4.1 are applied on eight different data sets and compared. As a reminder, MMTS is an extension of the original MTS algorithm to multi-class problems; FWMMTS-I is the adaptation of feature weighted MD, which is proposed by Wölfel and Ekenel (2005), to the multi-class MTS problems; and lastly, FWMMTS-II is the modification of the method of Su and Hsiao (2009) in MD calculation by using the original MD instead of GS.

In order to compare the results with those of Su and Hsiao (2009), we take the Mahalanobis Distance Classifier (MDC) method as a common point. MDC does not search for the important variables. MDC assigns a new observation to the class of the minimum MD. As a consequence, it allows us to see the effect of finding and using important variables on the multi-class classification results, instead of using all of the original variables in the distance calculation.

After using the feature weighted MD in the FWMMTS-I method, a feature weighted modification of MDC, namely Weighted Mahalanobis Distance Classifier (WMDC), is also developed. First, the data is pre-processed as explained in Section 4.1. Next, a class is taken as a reference point, which is used as a normal group, while the other classes left in the abnormal group. WMDC uses the descent feature weighed MD given in Section 4.2.1. Thus, feature weighted MD values for the standardized abnormal observations are calculated by using

Equations from (4.1) to (4.4). Importantly, WMDC does not search for the significant variables similar to MDC. These steps are followed for each of the classes by considering each as a normal group under consideration. Thus, after MD calculation for each normal group, a standardized new or test observation is assigned to a class with minimum MD.

4.2.1 Applications

All of the methods are applied on some benchmark data sets having non-categorical input variables. They are taken from the web-site of the UCI Machine Learning Data Repository. Data sets are selected by considering their citation rate in articles. Also data sets used by Su and Hsiao (2009) are preferred in order to make comparisons. Observations with missing values are omitted during the data pre-processing. After pre-processing, the characteristics of eight different data sets are summarized in Table 4.2.

Table 4.2: Data Set Information

Data Set	**Number of Classes**	**Data Size**	**Number of Variables**	**Ratios of Classes**	**Balanced (B) /Imbalanced(IB)**
wine	3	178	13	33.158, 39.958, 26.884	B
iris	3	150	4	33.378, 33.297, 33.325	B
waveform	3	5000	21	33.14, 32.941, 33.919	B
balance-scale	3	625	4	7.8387, 46.087, 46.074	IB
vehicle	4	846	18	25.052, 25.771, 23.52, 25.657	B
abalone[1]	11	3842	8	3.1, 6.7, 10.4, 14.4, 18.4, 16.8, 12.3, 6.8	IB
yeast[2]	9	1479	6	16.5, 29, 31.305, 2.9748, 2.3692, 3.4498, 11.018, 2.0298, 1.3532	IB
water-treat[3]	4	208	38	43.913, 24.396, 16.552, 15.14	IB

[1] only the classes 5,6,7,8,9,10,11,12,13,14,15 are used.

[2] variable 5 and 6 are omitted because these variables do not change in observations.

[3] only the classes 1, 5, 9, and 11 are used same as Hsiao (2009) and all of the missing observations are deleted.

In order to compare the methods, a 5-fold stratified cross validation (SCV) is used for all data sets. This ensures a given fold contains all of the classes. For SCV, the original data is partitioned into the classes. The classes are then partitioned into 5 sub-samples (folds). Of these sub-samples, while a single sub-sample is retained as the data for testing the model, the remaining sub-samples are used as training data during the model development. This means a total of 45 runs are done for each method.

We normally write $OA_N(s^k)$ to specify such an orthogonal array, which has an array of size N by k, with entries from 0 to s-1. We have used an OA specified as $OA_{200}(2^{100})$ with strength three from the web-site of "A Library of Orthogonal Arrays" (http://www.research.att.com/~njas/oadir/).

As a performance measure, we use BCA (given in Section 2.1.2.2). BCA is useful to consider the correct classification accuracy for each class since in some data sets (such as yeast, abalone, balance-scale and water-treat) there are classes in minority. We also consider the average and standard deviation of BCA values over all five folds. In addition to BCA, average percentage of correct classification (PCC) is also used to see the overall accuracy. Its formula is also given in Section 2.1.2.2.

The average performance results of developed methods are given in Table 4.3. Along with the results of six classification algorithms, the results of MDC in Su and Hsiao (2009) and also results of their proposed FWMMTS (Su and Hsiao, 2009) method are written.

Table 4.3: Application Results of the Methods

Data Set	Measure	MMTS	SNRMMTS	FWMMTS-I	FWMMTS-II	FWMMTS (Su and Hsiao, 2009)	WMDC	MDC	MDC (Su and Hsiao, 2009)
iris	*Avr. BCA*	0.9449	0.9449	0.9584	0.9465	0.9878	0.8557	0.9722	0.9817
	Std. BCA	0.0710	0.0710	0.0360	0.0444	0.0168	0.0761	0.0393	0.0168
	Avr. PCC	0.9627	0.9627	0.9641	0.9552	NS	0.8664	0.9818	NS
	Std. PCC	0.0390	0.0390	0.0291	0.0396	NS	0.0645	0.0257	NS
balance-scale	*Avr. BCA*	0.8093	0.7639	0.8093	0.6589	NS	0.6511	0.8656	NS
	Std. BCA	0.1439	0.2146	0.1439	0.0061	NS	0.0446	0.0488	NS
	Avr. PCC	0.9166	0.6927	0.9166	0.9108	NS	0.5259	0.9091	NS
	Std. PCC	0.0242	0.3155	0.0242	0.0267	NS	0.0771	0.0128	NS
vehicle	*Avr. BCA*	0.8603	0.8603	0.8577	0.3771	0.8352	0.4505	0.8111	0.8454
	Std. BCA	0.0060	0.0060	0.0073	0.0121	0.0307	0.0484	0.0338	0.0313
	Avr. PCC	0.8567	0.8567	0.8541	0.3677	NS	0.4494	0.8055	NS
	Std. PCC	0.0134	0.0134	0.0149	0.0317	NS	0.0499	0.0332	NS
wine	*Avr. BCA*	0.9628	0.9628	0.9686	0.9410	0.9793	0.9031	0.9628	0.9895
	Std. BCA	0.0526	0.0526	0.0460	0.0471	0.0206	0.0606	0.0526	0.0147
	Avr. PCC	0.9758	0.9758	0.9814	0.9458	NS	0.9062	0.9758	NS
	Std. PCC	0.0343	0.0343	0.0198	0.0413	NS	0.0564	0.0343	NS
yeast	*Avr. BCA*	0.4261	0.4012	0.4323	0.4191	0.4793	0.4116	0.3961	0.4829
	Std. BCA	0.0410	0.0396	0.0487	0.0317	0.0309	0.0548	0.0674	0.0498
	Avr. PCC	0.4941	0.4471	0.4921	0.5011	NS	0.4014	0.4210	NS
	Std. PCC	0.0317	0.0616	0.0311	0.0111	NS	0.0302	0.0503	NS
waveform	*Avr. BCA*	0.8511	0.8511	0.8511	0.8108	0.8338	0.8454	0.8529	0.8513
	Std. BCA	0.0124	0.0124	0.0124	0.0113	0.0133	0.0071	0.0166	0.0137
	Avr. PCC	0.8510	0.8510	0.8510	0.8115	NS	0.8456	0.8530	NS
	Std. PCC	0.0125	0.0125	0.0125	0.0147	NS	0.0092	0.0168	NS
water-treat	*Avr. BCA*	0.3143	0.3173	0.3592	0.2138	0.7732	0.6983	0.3416	0.4091
	Std. BCA	0.0767	0.0764	0.1468	0.0933	0.0189	0.0750	0.0942	0.0088
	Avr. PCC	0.3336	0.4717	0.4832	0.2638	NS	0.6024	0.4772	NS
	Std. PCC	0.1425	0.0524	0.0636	0.1567	NS	0.0813	0.0781	NS
abalone	*Avr. BCA*	0.2332	0.2302	0.2332	0.2542	0.2683	0.2484	0.2163	0.2310
	Std. BCA	0.0114	0.0081	0.0114	0.0098	0.0132	0.0103	0.0152	0.0222
	Avr. PCC	0.2092	0.2041	0.2092	0.2389	NS	0.2552	0.1808	NS
	Std. PCC	0.0106	0.0164	0.0106	0.0186	NS	0.0095	0.0107	NS

*NS: Not Studied

Table 4.3 shows that there are some differences between our MDC results and those of Su and Hsiao (2009). Hsiao (2009) explains MDC procedure used in their article in short as following: An individual MS for each class using the training data is constructed, and an unknown example (including the test data) is classified into the class with the minimum MD. We also follow this procedure, but we identifier observations with MD larger than ($\overline{X}_{md} + 3s_{Xmd}$) as outliers. Here, $\overline{X}_{md}$ is the average of and s_{Xmd} is the standard deviation of MD values. However, in Su and Hsiao (2009), there is no information about handling outliers in MDC. Additionally, we delete all the observations having missing values in "water-treat" data set, but there is no comment on this in Su and Hsiao (2009).

It is proven that the results change, when allocations of variables in an OA are changed (Abraham and Variyath, 2003). Because there is no information about the OA used by Su and Hsiao (2009), one has to develop their model by the same OA, which is used in other methods to be compared.

Moreover, we include the BCA and PCC values of each fold as replications in the statistical analysis. Since BCA value of each fold is not given in Su and Hsiao (2009), one has to run their model separately for each fold to collect the replication results. Because of complex calculation procedures of GS algorithm, this study has not been performed in this work. Due to these reasons, we have not included the results of Su and Hsiao (2009) in the statistical analysis performed in the next section.

4.2.2 Performance Analysis

In this part, we statistically compare the six developed algorithms. *Minitab 15* is used for the statistical analysis. A two-way ANOVA with five replications (folds) is performed. BCA and PCC are taken as the "response" in each ANOVA study, separately. The methods are taken as a "factor", while data sets are considered as a "blocking variable". The assumptions of ANOVA, which primarily are constant variance and normality of residuals, are checked. The residual plots for BCA and PCC are given in Appendix C.1. The results of the two-way ANOVA are given in Table 4.4.

Table 4.4: ANOVA for overall BCA results (with each fold)

Source	DF	SS	MS	F	P
Data set (block)	7	15.8384	2.26263	530.03	0.000
Methods (factor)	5	0.3311	0.06622	15.51	0.000
Interaction	35	1.9034	0.05438	12.74	0.000
Error	192	0.8196	0.00427		
Total	239	18.8926			

Table 4.4 shows that, there are significant differences among the six multi-class MTS methods, even if α-level of 0.01 is chosen for the test. When the two-way ANOVA analysis is performed without replications, the result is changed to no difference among the methods, given as in Table 4.5.

Table 4.5: ANOVA for Average BCA results (with averages)

Source	DF	SS	MS	F	P
Data set	7	3.10089	0.442984	37.79	0.000
Methods	5	0.08376	0.016751	1.43	0.238
Error	35	0.41027	0.011722		
Total	47	3.59491			

The difference in p-values between Tables 4.4 and 4.5 is due to inclusion of standard deviations among BCA values in the analysis. When the two-way ANOVA for the PCC values is done, Table 4.6 is obtained.

Table 4.6: ANOVA for overall PCC results (with each fold)

Source	DF	SS	MS	F	P
Data set (block)	7	15.5045	2.21492	512.13	0.000
Methods (factor)	5	0.4188	0.08376	19.37	0.000
Interaction	35	2.0513	0.05861	13.55	0.000
Error	192	0.8304	0.00432		
Total	239	18.8050			

Table 4.6 shows that, there are significant differences among the six multi-class MTS methods, even if α-level of 0.01 is chosen for the test.

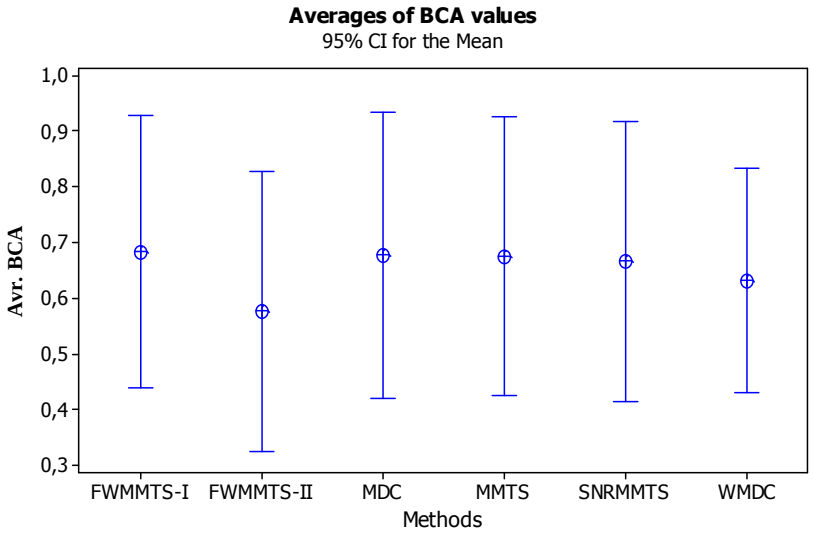

Figure 4.2: 0.95 Confidence Interval of the Mean of the Average BCA Values

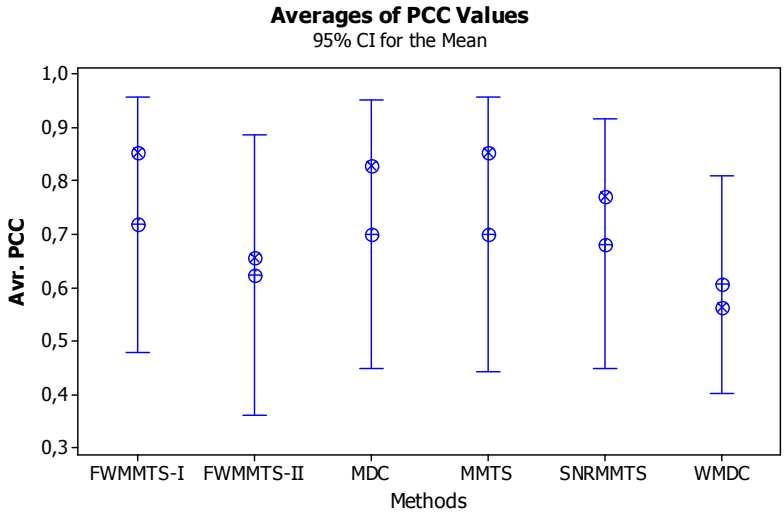

Figure 4.3: 0.95 Confidence Interval of the Mean of the Average PCC Values

Figure 4.2 indicates that FWMMTS-I, MMTS, SNRMMTS and MDC are not significantly different from each other in average BCA performances, and they are better than FWMMTS-II and WMDC. Similarly, Figure 4.3 indicates that FWMMTS-I, MMTS and MDC show very similar performance in average PCC performances, and they might be better than the others.

A detailed multiple comparison analysis can be used to support these observations. The methods are compared in pairs using Bonferroni and Tukey's tests in "ANOVA with General Linear Model (GLM)" tool of *Minitab 15*. In *Minitab* analysis, data sets are added as a random "factor" and methods as a fixed "factor" into the model. Then, the "methods" factor is selected as the comparison term to conduct the comparison tests with a 95 percent confidence interval. The analysis of variance table and multiple comparison results of Bonferroni and Tukey's tests are obtained as shown in Appendix C.2.

The Bonferroni's method can be used both to compare all possible pairs for the specified factors, and to compare each mean to the mean of a control group. Both Bonferroni and Tukey's tests use a family error rate to control Type-I error, whereas a Fisher's Least Significant Difference (LSD) test only uses individual error rate. Since we want to compare all possible pairs of the six methods in terms of BCA and PCC, it is important to consider the family error rate, because chances of making a Type-I error for a series of comparisons will be greater than the error rate

for any one of the individual comparisons alone. The two methods adjust the error rate for individual pair-wise comparisons based on the family error rate chosen and the number of comparisons (Statguide of Minitab 15). These tests boil down to running a bunch of *t* tests and then adjusting the significance level to take the appropriate control of Type I errors. For example, the Bonferroni test uses a straight-forward *t* test but then evaluates that *t* at α-level of 0.05/c, where c is the number of comparisons, 0.05 is the family error rate.

Six methods result in 15 pair-wise comparisons. If α-level of 0.05 is chosen family error rate, the corrected error rate is 0.05/15, which is probability of 0.0033 for significance. Thus, if the adjusted *p-values* for the difference between the mean for any pair is less than 0.0033, this indicates that the difference is significant. According to this probability of significance, *p-values* of multiple comparisons for BCA and PCC are analyzed in Tables 4.7-4.10, in which "<" indicates that the performance of the method in the row list is less than method compared..

Table 4.7: *p-Values* of Bonferroni Multiple Comparison Test for BCA results

Methods	MDC	MMTS	SNRMMTS	WMDC	FWMMTS-I
FWMMTS-II	0.0010 (<)	0.0014 (<)	0.0054	0.3721	0.0003 (<)
MDC		1	1	1	1
MMTS			1	1	1
SNRMMTS				1	1
WMDC					0.5939

Table 4.8: *p-Values* of Tukey's Multiple Comparison Test for BCA results

Methods	MDC	MMTS	SNRMMTS	WMDC	FWMMTS-I
FWMMTS-II	0.0009 (<)	0.0013 <)	0.0048	0.2153	0.0003 (<)
MDC		1	0.9978	0.4621	0.9998
MMTS			0.9992	0.5175	0.9993
SNRMMTS				0.7479	0.9812
WMDC					0.3069

Tables 4.7 and 4.8 indicate that MDC, MMTS and FWMMTS-I methods show better performances than FWMMTS-II in BCA since their pair-wise comparisons give p-values less than the probability of significance, which is 0.0033.

Table 4.9: *p-Values* of Bonferroni Multiple Comparison Test for PCC results

Methods	MDC	MMTS	SNRMMTS	WMDC	FWMMTS-I
FWMMTS-II	0,0418	0,0448	0,3208	1	0,0033 (<)
MDC		1	1	0,0036 (>)	1
MMTS			1	0,0039 (>)	1
SNRMMTS				0,0419	1
WMDC					0,0002 (<)

Table 4.10: *p-Values* of Tukey's Multiple Comparison Test for PCC results

Methods	MDC	MMTS	SNRMMTS	WMDC	FWMMTS-I
FWMMTS-II	0.0329	0.0350	0.1915	0.9811	0.0030 (<)
MDC		1	0.9811	0.0033 (>)	0.9777
MMTS			0.9835	0.0036 (>)	0.9746
SNRMMTS				0.0329	0.7032
WMDC					0.0002 (<)

According to Tables 4.9 and 4.10, MDC, MMTS SNRMMTS and FWMMTS-I show similar performances in terms of PCC, which gives the overall accuracy. Although, FWMMTS-II shows a moderately similar performance with MDC, MMTS, and SNRMMTS, it gives a worse performance than FWMMTS-II, in overall accuracy. In addition, MDC, MMTS and FWMMTS-I show better performances than WMDC in PCC.

The comparison of MDC with other methods indicates that there is no significant difference between multi-class MTS methods MMTS, FWMMTS-I, SNRMMTS and MDC. It is expected that, after finding important variables for multi-class MTS methods, unlike MDC, the models give better results than MDC. Possible reasons of the observation contrary to this expectation are explained below.

The first reason may be the variable selection method. In fact, it is observed that the number of variables after the selection process with OA and S/N ratios does not decrease much for the methods: MMTS, FWMMTS-I, SNRMMTS especially for the studied data sets with more variables such as abalone, vehicle or water. Hence, it is not surprising to observe that MDC is performing equally well with the others MMTS, FWMMTS-I, SNRMMTS. In order to see if there exist advantages of the variable selection procedures of the methods, one should

study more data sets. This may be a reason for being the results very close to MDC. In addition, OA is preferred in the original MTS since it gives acceptable solutions while decreasing number of experiments to calculate S/R ratios (Taguchi et al., 2003). However, our results reveal that finding significant variables with OA does not make a significant improvement in performance results. Certain discussions on OA, (Hawkins, 2003; Woodall et al., 2003; Abraham and Variyath, 2003) are given in detail in Section 2.2.4. In fact, Nagao et al. (1999) encourage increasing the initial number of variables in the training set to make an improvement in the results. These opinions are supporting our results.

The second reason may be the threshold selection method. Being a special characteristic of MTS, finding the best threshold is important for the accuracy of model (Su and Hsiao, 2007). However, in our multi-class MTS methods, a threshold does not exist. Instead, a new observation is assigned to the class of minimum MD. Therefore, our methods show similar performance to that of MDC.

Su and Hsiao (2009) find that their proposed method gives a better performance than MDC. However, when the average BCA results are analyzed, only in two data sets (“water-treat” and “Mfeatures”) significant and in one data set (abalone) moderately difference are seen in total of 12 data sets. We also apply MDC to “water-treat” data set. However, it does not give the same results as in Su and Hsiao (2009). The anticipated reasons are explained before in Section 4.2.1. In fact, there are missing values in this data set, which we have already omitted. However, there is no comment on this in Su and Hsiao (2009). In addition, an OA which is suitable for the “Mfeatures” data set is hard to find since the data set has 649 variables. The BCA performance comparison of MMTS with MDC is not appropriate in that situation, due to these reasons.

The reasons for exclusion of the method proposed by Su and Hsiao (2009) in multiple comparisons are explained in Section 4.2.1. When a simple comparison is of the methods FWMMTS-I, FWMMTS-II, MMTS, SNRMMTS, MDC, WMDC and FWMMTS of Su and Hsiao (2009) done based on the average of the BCA values, the following ANOVA table is obtained by the two-way ANOVA study.

Table 4.11: ANOVA for Average of the BCA Values (Including FWMMTS method of Su and Hsiao (2009))

Source	DF	SS	MS	F	P
Data set	7	2.63253	0.376076	9.06	0.000
Methods	7	0.17778	0.025397	0.61	0.743
Error	49	2.03357	0.041501		
Total	63	4.84388			

The results indicate that there is no evidence for significant difference among the all of the methods. However, this result may be regarded as “false” since Table 4.4 and 4.5 reveal that *p-values* of the methods are changed, when folds are included as replications in the ANOVA study.

Based on these results, we can claim that MMTS and FWMMTS-I and MDC produce similar results for the multi-class classification problems. We also further suggest testing of the proposed methods on more data sets, which have different characteristics of data size, number of classes and variables, to reach stronger conclusion about the superiority of any of the methods.

CHAPTER 5

CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

In this study, we have developed several multi-class classification methods with MTS. We have also studied the sample size and class imbalance problems, which are encountered mostly in small data sets by developing a re-sampling algorithm as a solution.

We have developed the following multi-class classification methods with Mahalanobis Taguchi System (MTS): MMTS, FWMMTS-I, FWMMTS-II, SNRMMTS. MMTS is an expansion of the original MTS algorithm for multiple class problems using ANOVA in variable selection process; SNRMMTS is an expansion of the original MTS algorithm for multiple class problems using only S/N ratios; FWMMTS-I is an adaptation of feature weighted MD, which is proposed by Wölfel and Ekenel (2005), to the multi-class MTS problems; and lastly FWMMTS-II is the modification of the method of Su and Hsiao (2009) in MD calculation by using the original MD instead of GS.

The comparison of MDC, which uses all variables in the distance calculation, with other methods indicates that there is no significant difference between multi-class methods MMTS, FWMMTS-I, SNRMMTS and MDC. One may expect that, after selecting the important variables for multi-class MTS methods, unlike MDC, the models give better results than MDC. In fact, it is observed that the methods select almost the same variables. Hence, it is not surprising to observe that MDC is performing equally well with the others MMTS, FWMMTS-I, SNRMMTS. In order to see if there exist advantages of the variable selection procedures of the methods, one should study more data sets. This may be a reason for being their results very close to MDC. Another reason may be the effect of OA design and usage of OA instead of larger experimental designs. A possible future work is developing better variable selection methods in order to improve the results.

Another reason may be the threshold method. Actually, MTS is different than MDC in case of binary classification; also in the way it assigns new observations to classes. MTS uses a threshold for this purpose. In the literature, it is shown that the threshold method affects the accuracy of the model significantly. However, in our multi-class classification methods, we do not use a threshold. We simply assign a new observation to the class of minimum MD. Therefore, our methods show similar performance to that of MDC.

In addition, we have used the "one-to-all" multi-class approach. Instead, the "one-to-one" approach can be utilized, although it increases the number of models to solve the multi-class problem. We also further suggest testing of the proposed methods on more data sets, which have different characteristics of data size, number of classes and variables, to reach stronger conclusion about the superiority of any of the methods.

Re-sampling, on the other hand, can be performed by over-sampling, which increases the data size; by under-sampling, which decreases the data size by removing existing observations randomly until a class balance is reached; or by a combination of over-sampling and under-sampling. Our re-sampling approach, which only works for data sets with two classes, is a combination of over-sampling and under-sampling. The over-sampling is performed by SMOTE, which generates the synthetic observations between the nearest neighbors of observations in the minority class. In addition, MTS models are used to test the performance of several parameters of re-sampling, for which the most appropriate values are sought for specific to each case.

Based on our re-sampling approach, we can not achieve a rule for the relationship between the initial parameters and re-sampling parameters since the decision tree based applications indicate that the results are changing according to the initial data characteristics. For a given data set, suggested re-sampling parameters can be selected after searching on different values of parameters. For a future work on our re-sampling method, a different classifier (such as Support Vector Machines, Neural Networks), which is sensitive to the imbalanced data, can be tested to demonstrate the effect of re-sampling. In addition, in SMOTE, different selection ways of nearest neighbours can be studied. This re-sampling algorithm can also be extended to multi-class imbalance problems. Lastly, we let a large increase in the number of observations in the training set. It is also possible to modify the algorithm to put a limit on the final sample size, especially for the very small data sets.

REFERENCES

A Library of Orthogonal Arrays, http://www.research.att.com/~njas/oadir/, last visited on March 2009.

Abraham, B. and Variyath, A. M. (2003). Discussion. *Technometrics, 45*(1), 22-24.

Aman, H., Mochiduki, N., and Yamada, H. (2006). A model for detecting cost-prone classes based on mahalanobis-taguchi method. *IEICE Transactions on Information and Systems, E89-D*(4), 1347-1358.

Anderson, C. and Schumacker, R. E. (2003). A comparison of five regression methods with ordinary least squares regression: relative efficiency, bias, and test of the null hypothesis. *Understanding Statistics*. 2 (2), 79-103.

Bakır, B., Köksal, G., Ayhan, D., and Yenidünya, B. (2009). A SMOTE based re-sampling approach optimized for mts classification of imbalanced data. *Workshop on Recent Developments in Applied Probability and Statistics*. Middle East Technical University, Ankara., Turkey.

Barandela, R., Sanchez J.S., Garcia, V., and Rangel, E. (2003) Rapid and Brief Communication Strategies for learning in class imbalance problems. *Pattern Recognition*, 36, 849 – 851.

Bar-Hen, A. and Daudin, J.-J. (1995). Generalization of the Mahalanobis distance in the mixed case. *Journal of Multivariate Analysis*, 53, 332-342.

Bedrick, E., Lapidus, J. and Powell, J.F. (2000). Estimating the Mahalanobis Distance from Mixed Continuous and Discrete Data. *Biometrics*, 56, 394-401.

Chawla, N.V., Bowyer, K.W., Kegelmeyer W.P. and Hall, L.O. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research,* 16, 341-378.

Chen, M. -., Chen, L. -., Hsu, C. -., and Zeng, W. -. (2008). An information granulation based data mining approach for classifying imbalanced data. *Information Sciences, 178*(16), 3214-3227.

Cudney, E. A., Hong, J., Jugulum, R., Taguchi, G., Paryani, K., and Ragsdeli, K. M. (2007). A comparison study of mahalanobis-taguchi system and neural network for multivariate pattern recognition. *Journal of Industrial and Systems Engineering*, 1(2), 139-150.

Cudney, E. A., Paryani, K., and Ragsdell, M. K. (2006). Applying the mahalanobis-taguchi system to vehicle handling. *Concurrent Engineering Research and Applications, 14*(4), 343-354.

Das, P. and Datta, S. (2007). Exploring the effects of chemical composition in hot rolled steel product using mahalanobis distance scale under mahalanobis-taguchi system. *Computational Materials Science, 38*(4), 671-677.

De Groot, P. J., Postma, G. J., Melssen, W. J., Buydens, L. M. C., Deckert, V., and Zenobi, R. (2001). Application of principal component analysis to detect outliers and spectral deviations in near-field surface-enhanced raman spectra. *Analytica Chimica Acta, 446*(1-2), 71-83.

De Leon, A. R., and Carrière, K. C. (2005). A generalized mahalanobis distance for mixed data. *Journal of Multivariate Analysis*, 92(1), 174-185.

De Maesschalck, R., Jouan-Rimbaud, D. , Massart, D.L. Chemom. Int. Lab. Syst. 50 (2000) 1.

Estabrooks, A., Jo T. and Japkowıcz N. (2004). A multiple re-sampling method for learning from imbalanced data sets. *Computational Intelligence*, 20 (1).

Ding, C. H. Q. and Dubchak, I. (2001). Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics, 17*(4), 349-358.

Friedman, J. (1996). Another Approach to Polychotomous Classification. Technical Report. Dept. Statist., Stanford Univ., Stanford, CA.

Hawkins, D. M. (2003). Discussion. *Technometrics, 45*(1), 25-29.

Hsiao, Y.H. (2009). Personal Communication.

Hsu, C-W. and Lin, C-J. (2002). A Comparison of Methods for Multi-class Support Vector Machines. *IEEE Transactions on Neural Networks*, 13(2), 415-425.

Huang, D.S., Zhang, X.-P., Huang, G.-B. (Eds.). (2005). Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. ICIC 2005, Part I, LNCS 3644, pp. 878 – 887, 2005. Springer-Verlag Berlin Heidelberg.

Japkowicz, N. and Stephen S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis*. 6, 429–449.

Johnson, R. A. and Wichern, D. W. (1998). Applied Multivariate Statistical Analysis. 4. edition. Prentice Hall. New Jersey.

Jugulum, R. (2000). New dimensions in multivariate diagnosis to facilitate decision making process. Ph.D. dissertation, Wayne State University, United States- Michigan.

Jugulum, R., Taguchi, G., Taguchi, S., and Wilkins, J. O. (2003). Discussion. *Technometrics*, *45*(1), 16-21.

Kubat, M and Matwin, S., (1997). Addressing the curse of imbalanced training sets: one-sided selection, Proceedings of the 14th International Conference on Machine Learning, Nashville, USA, 179–186.

Liu, A. (2004). The Effect of Over-sampling and Under-sampling on Classifying Imbalanced Text Data sets. MS Dissertation. The University of Texas. Austin.

Mahalanobis P. C. (1936). “On the generalized distance in statistics” *Proceedings of the national institute of science of India*, 2, 49-55.

Mendenhall, W. and Sincich, T. (2003). A second course in statistics regression analysis. 6. edition. Chapter 7. Pearson Prentice Hall. USA.

Moore, D. S. and McCabe, G. P. (1999). Introduction to the Practice of Statistics, 3. edition. New York: W. H. Freeman.

Nagao, M. Yamamoto, M. Suzuki, K. Ohuchi, A. (1999). MTS approach to facial image recognition. *IEEE SMC '99 Conference Proceedings*, *4*, 937-942.

Nickerson, A., Japkowicz, N., and Milios, E. (2001). Using unsupervised learning to guide re-sampling in imbalanced data sets. Proceedings of the Eighth International Workshop on Artificial Intelligence and Statistics, 261-265.

Ou, G., Murphey, Y.L., and Feldkamp L. (2004). Multi-Class Pattern Classification Using Neural Networks. *In ICPR ’04: Proceedings of the Pattern Recognation*, 4, 584-568, IEEE Computer Society.

Persson, P.P. (2007). Lecture 5-Gram-Schmidt Orthogonalization, Lecture Notes of Introduction to Numerical Methods. MIT. http://www-math.mit.edu/~persson/18.335/lec5handout6pp.pdf., Last visited on February 2009.

Riho, T., Suzuki, A., Oro, J., Ohmi, K., and Tanaka, H. (2005). The yield enhancement methodology for invisible defects using the MTS+ method. *IEEE Transactions on Semiconductor Manufacturing*, *18*(4), 561-568.

Sharma, S. (1996). Applied Multivariate Techniques. John Wiley & Sons. Inc., Canada.

Srinivasaraghavan, J., and Allada, V. (2006). Application of mahalanobis distance as a lean assessment metric. *International Journal of Advanced Manufacturing Technology*, *29*(11-12), 1159-1168.

Statguide of Minitab, http://www.minitab.com/products/minitab/documentation.aspx, Last visited on March 2009.

Su, C. T. and Hsiao, Y. H. (2007). An evaluation of the robustness of mts for imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 19 (10), 1321-1332.

Su, C. T. and Hsiao, Y. H. (2009). Multi-class mts for simultaneous variable selection and classification. *IEEE Transactions on Knowledge and Data Engineering*, 21 (2), 192-205.

Tabachnick, B. G., Fidell L.S. (1996). Using multivariate statistics. HarperCollins College Publishers. New York.

Taguchi G., Chowdhury, S., Wu, Y. (2001) .The Mahalanobis-Taguchi system. McGraw-Hill. New York.

Taguchi, G. and Jugulum, J. (2000). New trends in multivariate diagnosis. *The Indian Journal of Statistics, 62*(2), 233-235.

Taguchi, G. and Jugulum, R. (2002). The Mahalanobis-Taguchi Strategy. John Wiley & Sons. Inc. New York.

Watabe, A., Komiya, K., Usuki, J., Suzuki, K., and Ikeda, H. (2005). Effective designation of specific shots on video service system utilizing mahalanobis distance. *IEEE transactions on consumer electronics, 51(1), 152-159*

UCI Machine Learning Data Repository, http://archive.ics.uci.edu/ml/data sets.html, last visited on March 2009.

Weiss, G. M. (2004). Mining with rarity: A unifying framework. *SIGKDD Explorations*, 6(1), 7-19.

Weiss, G. M., Provost F. (2003). Learning When Training Data are Costly: The Effect of Class Distribution on Tree Induction. *Journal of Artificial Intelligence Research*, 19, 315-354.

Weiss, S. M. and Zhang T .(2003). Performance Analysis and Evaluation. The Handbook of Data Mining. Edited By Nong Ye.

Wölfel, M. and Ekenel, H. K. (2005). "Feature Weighted Mahalanobis Distance: Improved Robustness for Gaussian Classifiers", 13th European Signal Processing Conference: EUSIPCO, Antalya, Turkey.

Woodall, W. H., Koudelik, R., Tsui, K. -., Kim, S. B., Stoumbos, Z. G., and Carvounis, C. P. (2003). Response. *Technometrics, 45*(1), 29-30.

Woodall, W. H., Koudelik, R., Tsui, K., Kim, S. B., Stoumbos, Z. G., and Carvounis, C. P. (2003). A review and analysis of the mahalanobis-taguchi system. *Technometrics, 45*(1), 1-15.

Yenidünya, B. (2009). "Robust Design with a Binary Response Using Mahalanobis Taguchi System" M.S Thesis (in preparation), Middle East Technical University, Ankara, Turkey.

APPENDIX A

MATHEMATICAL BACKGROUND

A.1 Inflation of the Error Terms Due To Multicollinearity

Theoretically the problem of multicollinearity can be revealed for a given function Y as follows:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \mu$$

if $X_2 = 2X_1$,

then it becomes;

$$Y = \beta_0 + (\beta_1 + 2\beta_2) X_1 + \mu$$

Thus, only the term of $(\beta_1 + 2\beta_2)$ can be estimated. It is not possible to get separate the estimates of β_1 and β_2.

A square matrix of order n is said to be nonsingular if there exists a matrix **B**, called the multiplicative inverse of **A**, such that $\mathbf{A} \cdot \mathbf{B} = \mathbf{B} \cdot \mathbf{A} = \mathbf{I_n}$ where $\mathbf{I_n}$ is nxn identity matrix, then $\mathbf{B}=\mathbf{A}^{-1}$. Any matrix that does not have an inverse is said to be singular. Recalling from the notes on matrix algebra, the inverse can be found using the determinant of a matrix. In addition, a square matrix **A** is nonsingular if $\mathbf{A} \cdot \mathbf{X} = \mathbf{0}$ implies that **X**=0 or the columns of **A** are linearly independent (Johnson and Wichern, 1998). According to these definitions, every non-square matrix is singular. However, non-square matrices may have right and left inverses $\mathbf{A} \cdot \mathbf{B} = \mathbf{I}$, then we say that B is right inverse of **A** and **A** is left inverse of **B** (Taguchi et al., 2001).

A.2 Adjoint Matrix Approach for MTS

The adjoint $\mathbf{C}_{Adj}$ of a square matrix **C** is formed by taking the transpose of it. MD, which is calculated by adjoint matrix approach, is obtained as:

$$MD_{adj} = D^2 = \frac{\mathbf{z}_i^T \mathbf{C}_{adj}^{-1} \mathbf{z}_i}{k} \quad ,$$

$$\mathbf{z}_i = \left(\frac{x_{i1} - \mu_1}{\sigma_1}, \ldots, \frac{x_{ik} - \mu_k}{\sigma_k}\right) \quad .$$

where:

$\mathbf{z}_i$: standardized $\mathbf{z}_i$ vector obtained by the standardized values of x_i (i=1,…,k)

k : the number of variables.

T : transpose of the vector

The original MD can also be obtained from MD_{adj} :

$$MD = \frac{1}{det\,\mathbf{C}} MD_{adj}$$

A.3 Generalized Inverse Approach

A generalized inverse is also sometimes referred to as the conditional inverse, pseudo inverse, and g-inverse. The importance of the generalized inverse matrix **G** is revealed in the theorem: **G** is a generalized inverse of **A** since **AGA=A** (Moore, 1920 cited by Johnson and Wichern, 1998).

APPENDIX B

RE-SAMPLING

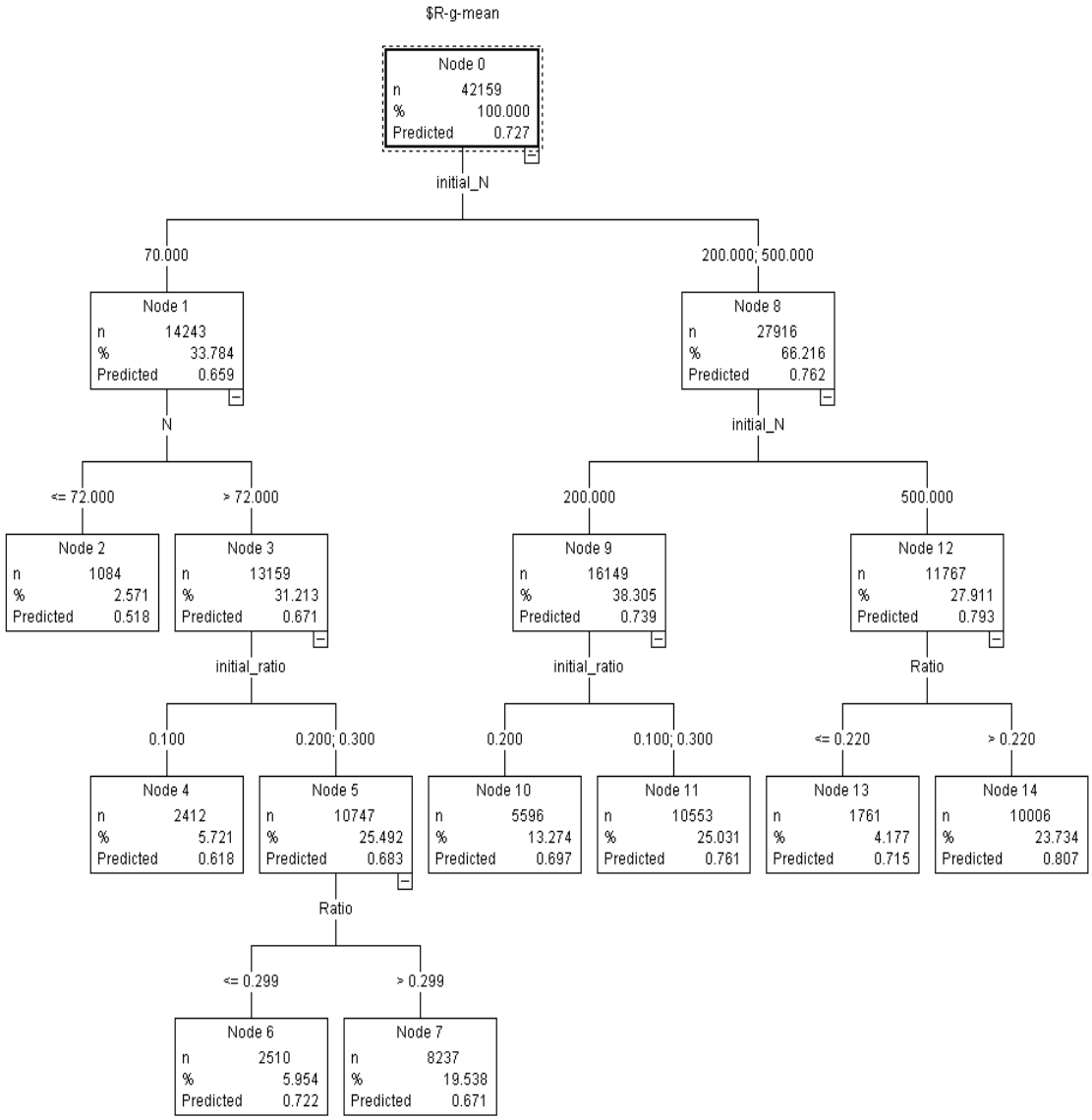

Figure B.1: A Decision Tree Based on Results of the Re-sampling Applications

Table B.1: Discretization Rule of Overall Training Size after Re-sampling

1: <25
2: 25-49
3: 50-74
⋮
27: 650-674
28: 675-700

Table B.2: Discretization Rule of the Minority Class Ratio after Re-sampling

1: 0.091-0.149
2: 0.149-0.199
3: 0.199-0.249
⋮
8: 0.451-0.499
9: 0.499-0.544

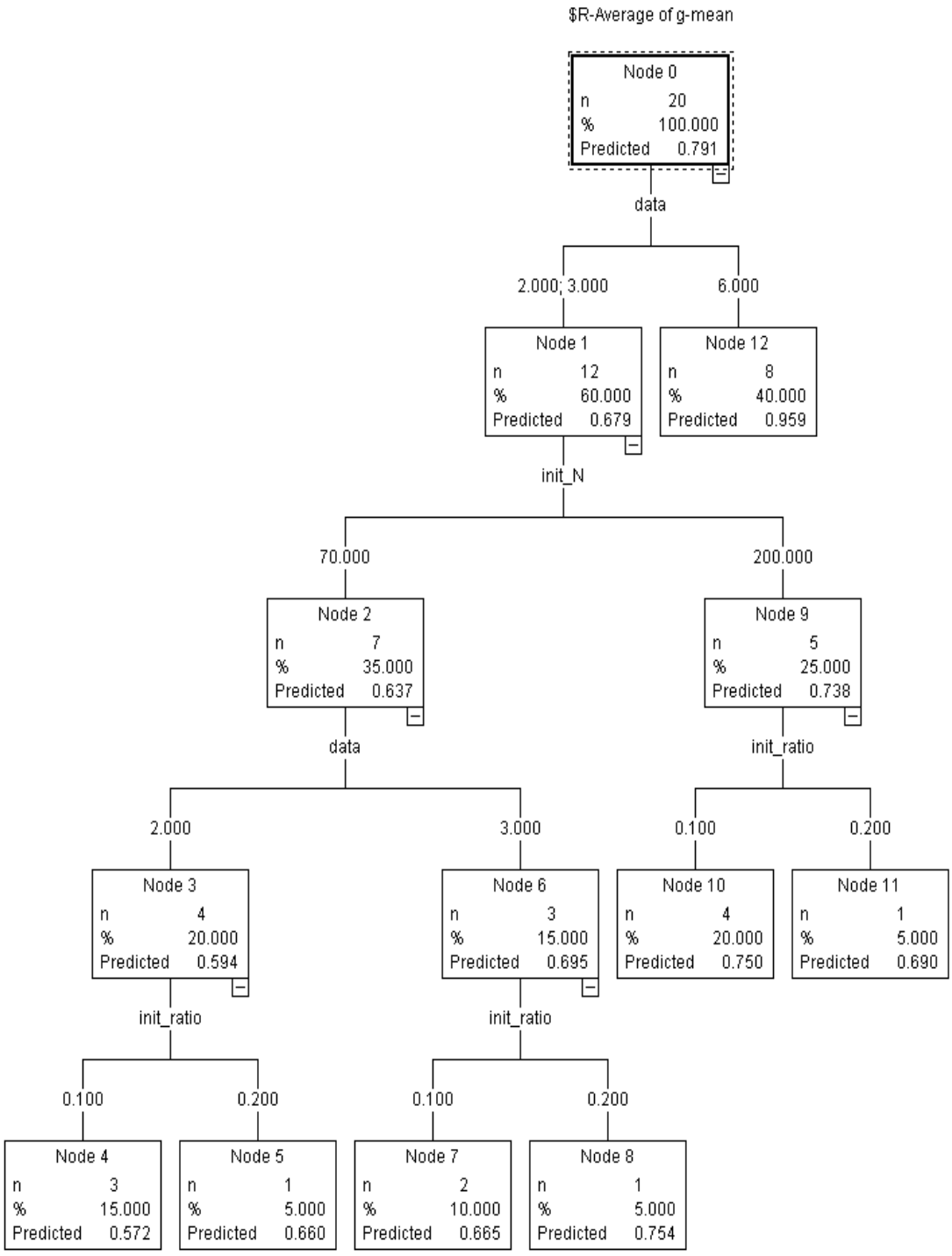

Figure B.2: A Decision Tree Based on Suggested Re-sampling Parameters of the Applications' Results

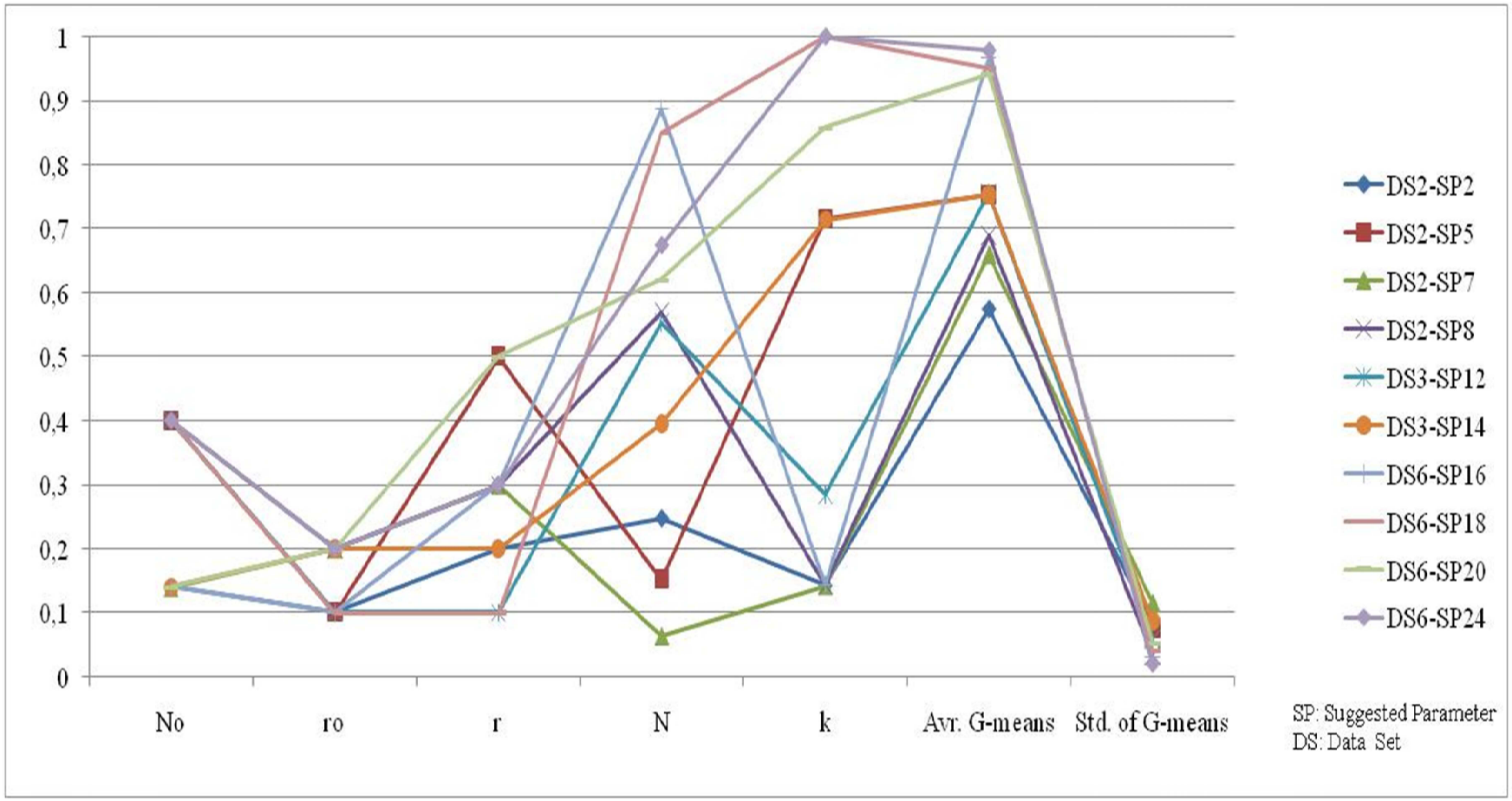

Figure B.3: Parellel Coordinates of the Normalized Suggested Parameters

APPENDIX C

PERFORMANCE ANALYSIS of MULTI-CLASS MTS METHODS

C.1 Residual Plots of ANOVA Study

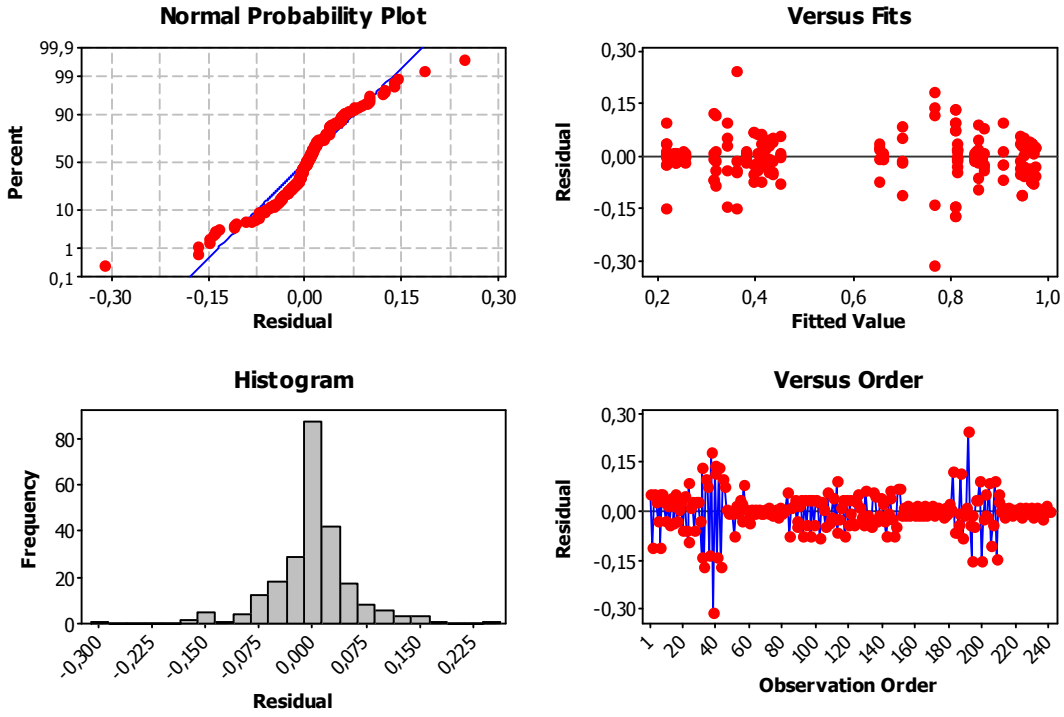

Figure C.1: Residual Plots for BCA

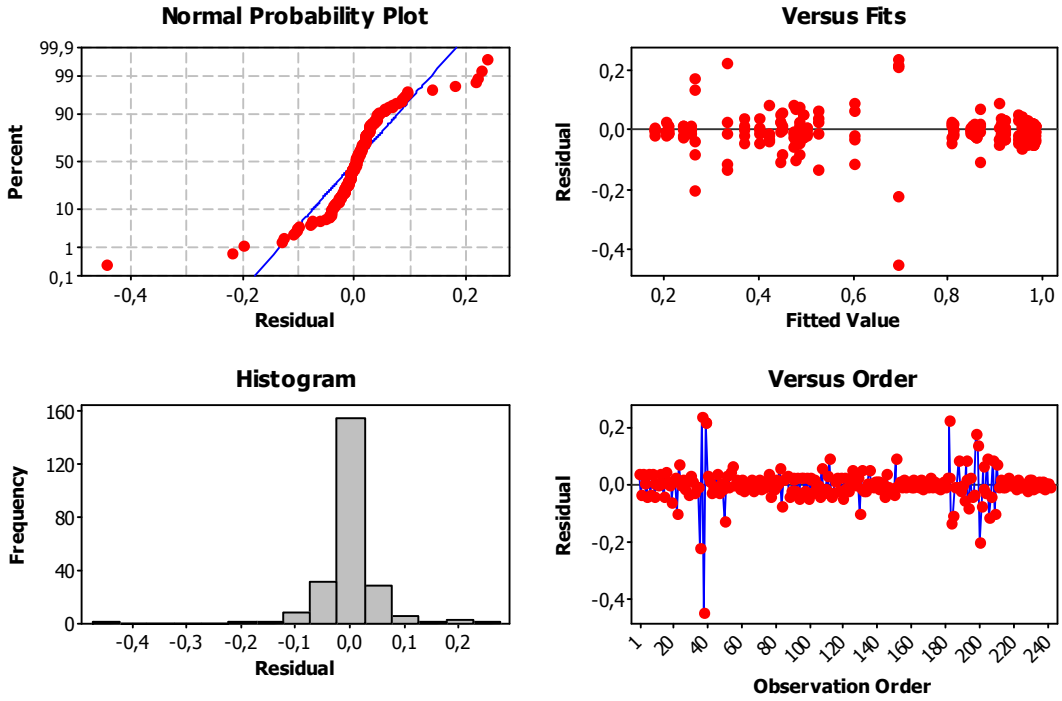

Figure C.2: Residual Plots for PCC

C.2 Multiple Comparisons of the Developed Methods

General Linear Model: BCA versus Data; methods_1

```
Factor     Type    Levels  Values
Data       random       8  abalone; balance-scale; iris; vehicle; water-
                           treat; waveform; wine; yeast
methods_1  fixed        6  FWMMTS-I; FWMMTS-II; MDC; MMTS; SNRMMTS; WMDC


Analysis of Variance for BCA, using Adjusted SS for Tests

Source      DF   Seq SS   Adj SS  Adj MS       F      P
Data         7  15,8384  15,8384  2,2626  188,62  0,000
methods_1    5   0,3311   0,3311  0,0662    5,52  0,000
Error      227   2,7230   2,7230  0,0120
Total      239  18,8926


S = 0,109525   R-Sq = 85,59%   R-Sq(adj) = 84,82%


Bonferroni 95,0% Simultaneous Confidence Intervals
Response Variable BCA
All Pairwise Comparisons among Levels of methods_1
methods_1 = FWMMTS-I  subtracted from:

methods_1    Lower   Center     Upper  --------+---------+---------+-----
---
FWMMTS-II  -0,1787  -0,1060  -0,03337  (------*-------)
MDC        -0,0790  -0,0064   0,06626            (------*-------)
MMTS       -0,0811  -0,0085   0,06418            (------*------)
SNRMMTS    -0,0899  -0,0173   0,05539           (------*-------)
WMDC       -0,1233  -0,0507   0,02196        (------*------)
                                     --------+---------+---------+-----
---
                                          -0,10      0,00      0,10


methods_1 = FWMMTS-II  subtracted from:

methods_1     Lower   Center   Upper  --------+---------+---------+------
--
MDC         0,02698  0,09964  0,1723                       (------*------
)
MMTS        0,02490  0,09755  0,1702                      (-------*------
)
SNRMMTS     0,01611  0,08876  0,1614                      (------*------)
WMDC       -0,01732  0,05533  0,1280                  (-------*------)
                                   --------+---------+---------+------
--
                                        -0,10      0,00      0,10


methods_1 = MDC  subtracted from:

methods_1    Lower    Center    Upper  --------+---------+---------+-----
---
MMTS       -0,0747  -0,00209  0,07057             (------*------)
SNRMMTS    -0,0835  -0,01087  0,06178            (------*------)
WMDC       -0,1170  -0,04430  0,02835        (-------*------)
                                     --------+---------+---------+-----
---
```

```
                                          -0,10      0,00      0,10


methods_1 = MMTS  subtracted from:

methods_1    Lower    Center    Upper  --------+---------+---------+-----
---
SNRMMTS    -0,0814  -0,00879  0,06386            (------*------)
WMDC       -0,1149  -0,04222  0,03044         (------*------)
                                       --------+---------+---------+-----
---
                                          -0,10      0,00      0,10


methods_1 = SNRMMTS  subtracted from:

methods_1    Lower    Center    Upper  --------+---------+---------+-----
---
WMDC       -0,1061  -0,03343  0,03923         (-------*------)
                                       --------+---------+---------+-----
---
                                          -0,10      0,00      0,10


Bonferroni Simultaneous Tests
Response Variable BCA
All Pairwise Comparisons among Levels of methods_1
methods_1 = FWMMTS-I  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
FWMMTS-II     -0,1060     0,02449   -4,329    0,0003
MDC           -0,0064     0,02449   -0,261    1,0000
MMTS          -0,0085     0,02449   -0,346    1,0000
SNRMMTS       -0,0173     0,02449   -0,705    1,0000
WMDC          -0,0507     0,02449   -2,070    0,5939


methods_1 = FWMMTS-II  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
MDC           0,09964     0,02449    4,068    0,0010
MMTS          0,09755     0,02449    3,983    0,0014
SNRMMTS       0,08876     0,02449    3,624    0,0054
WMDC          0,05533     0,02449    2,259    0,3721


methods_1 = MDC  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
MMTS         -0,00209     0,02449   -0,085     1,000
SNRMMTS      -0,01087     0,02449   -0,444     1,000
WMDC         -0,04430     0,02449   -1,809     1,000


methods_1 = MMTS  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
SNRMMTS      -0,00879     0,02449   -0,359     1,000
WMDC         -0,04222     0,02449   -1,724     1,000


methods_1 = SNRMMTS  subtracted from:
```

```
           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
WMDC         -0,03343     0,02449   -1,365     1,000


Tukey 95,0% Simultaneous Confidence Intervals
Response Variable BCA
All Pairwise Comparisons among Levels of methods_1
methods_1 = FWMMTS-I  subtracted from:

methods_1    Lower   Center     Upper  --------+---------+---------+-----
---
FWMMTS-II  -0,1763  -0,1060  -0,03572  (------*------)
MDC        -0,0767  -0,0064   0,06392           (------*------)
MMTS       -0,0788  -0,0085   0,06183           (------*------)
SNRMMTS    -0,0876  -0,0173   0,05304          (------*------)
WMDC       -0,1210  -0,0507   0,01962       (------*------)
                                       --------+---------+---------+-----
---
                                            -0,10      0,00      0,10


methods_1 = FWMMTS-II  subtracted from:

methods_1     Lower   Center   Upper  --------+---------+---------+------
--
MDC         0,02933  0,09964  0,1699                      (------*------
)
MMTS        0,02724  0,09755  0,1679                      (------*------
)
SNRMMTS     0,01845  0,08876  0,1591                     (------*------)
WMDC       -0,01497  0,05533  0,1256                 (------*------)
                                     --------+---------+---------+------
--
                                          -0,10      0,00      0,10


methods_1 = MDC  subtracted from:

methods_1    Lower    Center    Upper  --------+---------+---------+-----
---
MMTS       -0,0724  -0,00209  0,06822            (------*------)
SNRMMTS    -0,0812  -0,01087  0,05943          (------*------)
WMDC       -0,1146  -0,04430  0,02601        (------*------)
                                       --------+---------+---------+-----
---
                                            -0,10      0,00      0,10


methods_1 = MMTS  subtracted from:

methods_1    Lower    Center    Upper  --------+---------+---------+-----
---
SNRMMTS    -0,0791  -0,00879  0,06152           (------*------)
WMDC       -0,1125  -0,04222  0,02809        (------*------)
                                       --------+---------+---------+-----
---
                                            -0,10      0,00      0,10


methods_1 = SNRMMTS  subtracted from:

methods_1    Lower    Center    Upper  --------+---------+---------+-----
---
WMDC       -0,1037  -0,03343  0,03688          (------*------)
```

```
                                        --------+---------+---------+-----
---
                                              -0,10      0,00      0,10


Tukey Simultaneous Tests
Response Variable BCA
All Pairwise Comparisons among Levels of methods_1
methods_1 = FWMMTS-I  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
FWMMTS-II     -0,1060     0,02449   -4,329    0,0003
MDC           -0,0064     0,02449   -0,261    0,9998
MMTS          -0,0085     0,02449   -0,346    0,9993
SNRMMTS       -0,0173     0,02449   -0,705    0,9812
WMDC          -0,0507     0,02449   -2,070    0,3069


methods_1 = FWMMTS-II  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
MDC           0,09964     0,02449    4,068    0,0009
MMTS          0,09755     0,02449    3,983    0,0013
SNRMMTS       0,08876     0,02449    3,624    0,0048
WMDC          0,05533     0,02449    2,259    0,2153


methods_1 = MDC  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
MMTS         -0,00209     0,02449   -0,085    1,0000
SNRMMTS      -0,01087     0,02449   -0,444    0,9978
WMDC         -0,04430     0,02449   -1,809    0,4621


methods_1 = MMTS  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
SNRMMTS      -0,00879     0,02449   -0,359    0,9992
WMDC         -0,04222     0,02449   -1,724    0,5175


methods_1 = SNRMMTS  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
WMDC         -0,03343     0,02449   -1,365    0,7479
```

Figure C.3: General Linear Model: BCA versus methods and data sets

General Linear Model: PCC versus Data; methods_1

```
Factor     Type    Levels  Values
Data       random       8  abalone; balance-scale; iris; vehicle; water-
treat;
                           waveform; wine; yeast
methods_1  fixed        6  FWMMTS-I; FWMMTS-II; MDC; MMTS; SNRMMTS; WMDC


Analysis of Variance for PCC, using Adjusted SS for Tests

Source      DF   Seq SS   Adj SS  Adj MS       F      P
Data         7  15,5045  15,5045  2,2149  174,47  0,000
methods_1    5   0,4188   0,4188  0,0838    6,60  0,000
Error      227   2,8817   2,8817  0,0127
Total      239  18,8050


S = 0,112671   R-Sq = 84,68%   R-Sq(adj) = 83,87%


Bonferroni 95,0% Simultaneous Confidence Intervals
Response Variable PCC
All Pairwise Comparisons among Levels of methods_1
methods_1 = FWMMTS-I  subtracted from:

methods_1    Lower   Center     Upper  ---------+---------+---------+----
---
FWMMTS-II  -0,1694  -0,0946  -0,01989    (-------*------)
MDC        -0,0932  -0,0185   0,05627            (------*-------)
MMTS       -0,0937  -0,0190   0,05573            (------*-------)
SNRMMTS    -0,1110  -0,0363   0,03849          (------*-------)
WMDC       -0,1872  -0,1124  -0,03767  (-------*------)
                               ---------+---------+---------+----
---
                                     -0,10      0,00      0,10


methods_1 = FWMMTS-II  subtracted from:

methods_1     Lower    Center    Upper  ---------+---------+---------+---
----
MDC         0,00142   0,07616  0,15091                     (-------*-----
-)
MMTS        0,00088   0,07562  0,15036                     (-------*-----
-)
SNRMMTS    -0,01636   0,05838  0,13312                   (-------*------)
WMDC       -0,09252  -0,01778  0,05696            (------*-------)
                               ---------+---------+---------+---
----
                                     -0,10      0,00      0,10


methods_1 = MDC  subtracted from:

methods_1    Lower    Center     Upper  ---------+---------+---------+---
----
MMTS       -0,0753  -0,00054   0,07420             (-------*------)
SNRMMTS    -0,0925  -0,01779   0,05695            (------*-------)
WMDC       -0,1687  -0,09394  -0,01920    (-------*------)
                               ---------+---------+---------+---
----
                                     -0,10      0,00      0,10
```

```
methods_1 = MMTS  subtracted from:

methods_1    Lower    Center     Upper  ---------+---------+---------+---
----
SNRMMTS    -0,0920  -0,01725   0,05750           (------*-------)
WMDC       -0,1681  -0,09340  -0,01866    (-------*------)
                                        ---------+---------+---------+---
----
                                             -0,10      0,00      0,10


methods_1 = SNRMMTS  subtracted from:

methods_1    Lower    Center      Upper  ---------+---------+---------+--
-----
WMDC       -0,1509  -0,07616  -0,001418      (------*-------)
                                        ---------+---------+---------+--
-----
                                               -0,10      0,00      0,10


Bonferroni Simultaneous Tests
Response Variable PCC
All Pairwise Comparisons among Levels of methods_1
methods_1 = FWMMTS-I  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
FWMMTS-II     -0,0946     0,02519   -3,756    0,0033
MDC           -0,0185     0,02519   -0,733    1,0000
MMTS          -0,0190     0,02519   -0,754    1,0000
SNRMMTS       -0,0363     0,02519   -1,439    1,0000
WMDC          -0,1124     0,02519   -4,462    0,0002


methods_1 = FWMMTS-II  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
MDC           0,07616     0,02519   3,0231    0,0418
MMTS          0,07562     0,02519   3,0017    0,0448
SNRMMTS       0,05838     0,02519   2,3172    0,3208
WMDC         -0,01778     0,02519  -0,7057    1,0000


methods_1 = MDC  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
MMTS         -0,00054     0,02519   -0,021    1,0000
SNRMMTS      -0,01779     0,02519   -0,706    1,0000
WMDC         -0,09394     0,02519   -3,729    0,0036


methods_1 = MMTS  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
SNRMMTS      -0,01725     0,02519   -0,684    1,0000
WMDC         -0,09340     0,02519   -3,707    0,0039


methods_1 = SNRMMTS  subtracted from:

           Difference       SE of           Adjusted
```

```
methods_1    of Means  Difference  T-Value   P-Value
WMDC         -0,07616     0,02519   -3,023    0,0419


Tukey 95,0% Simultaneous Confidence Intervals
Response Variable PCC
All Pairwise Comparisons among Levels of methods_1
methods_1 = FWMMTS-I  subtracted from:

methods_1    Lower   Center     Upper  --------+---------+---------+-----
---
FWMMTS-II  -0,1670  -0,0946  -0,02230   (-------*------)
MDC        -0,0908  -0,0185   0,05386           (------*------)
MMTS       -0,0913  -0,0190   0,05332           (------*------)
SNRMMTS    -0,1086  -0,0363   0,03608         (------*-------)
WMDC       -0,1847  -0,1124  -0,04008  (------*------)
                                       --------+---------+---------+-----
---
                                            -0,10      0,00      0,10


methods_1 = FWMMTS-II  subtracted from:

methods_1     Lower    Center    Upper  --------+---------+---------+----
----
MDC         0,00384   0,07616  0,14849                    (-------*------
)
MMTS        0,00330   0,07562  0,14795                    (-------*------
)
SNRMMTS    -0,01395   0,05838  0,13071                   (------*------)
WMDC       -0,09011  -0,01778  0,05455           (------*------)
                                       --------+---------+---------+----
----
                                            -0,10      0,00      0,10


methods_1 = MDC  subtracted from:

methods_1    Lower    Center     Upper  --------+---------+---------+----
----
MMTS       -0,0729  -0,00054   0,07179             (------*------)
SNRMMTS    -0,0901  -0,01779   0,05454           (------*------)
WMDC       -0,1663  -0,09394  -0,02162   (-------*------)
                                       --------+---------+---------+----
----
                                            -0,10      0,00      0,10


methods_1 = MMTS  subtracted from:

methods_1    Lower    Center     Upper  --------+---------+---------+----
----
SNRMMTS    -0,0896  -0,01725   0,05508           (------*-------)
WMDC       -0,1657  -0,09340  -0,02108   (-------*------)
                                       --------+---------+---------+----
----
                                            -0,10      0,00      0,10


methods_1 = SNRMMTS  subtracted from:

methods_1    Lower    Center      Upper  --------+---------+---------+---
-----
WMDC       -0,1485  -0,07616  -0,003830     (------*-------)
                                        --------+---------+---------+---
-----
```

```
                                          -0,10      0,00      0,10


Tukey Simultaneous Tests
Response Variable PCC
All Pairwise Comparisons among Levels of methods_1
methods_1 = FWMMTS-I  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
FWMMTS-II     -0,0946     0,02519   -3,756    0,0030
MDC           -0,0185     0,02519   -0,733    0,9777
MMTS          -0,0190     0,02519   -0,754    0,9746
SNRMMTS       -0,0363     0,02519   -1,439    0,7032
WMDC          -0,1124     0,02519   -4,462    0,0002


methods_1 = FWMMTS-II  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
MDC           0,07616     0,02519   3,0231    0,0329
MMTS          0,07562     0,02519   3,0017    0,0350
SNRMMTS       0,05838     0,02519   2,3172    0,1915
WMDC         -0,01778     0,02519  -0,7057    0,9811


methods_1 = MDC  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
MMTS         -0,00054     0,02519   -0,021    1,0000
SNRMMTS      -0,01779     0,02519   -0,706    0,9811
WMDC         -0,09394     0,02519   -3,729    0,0033


methods_1 = MMTS  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
SNRMMTS      -0,01725     0,02519   -0,684    0,9835
WMDC         -0,09340     0,02519   -3,707    0,0036


methods_1 = SNRMMTS  subtracted from:

           Difference       SE of           Adjusted
methods_1    of Means  Difference  T-Value   P-Value
WMDC         -0,07616     0,02519   -3,023    0,0329
```

Figure C.4: General Linear Model: PCC versus methods and data