We propose a new classification method for longitudinal data based on a semi-parametric approach. Our approach builds a classifier by taking advantage of modeling information between response and covariates for each class, and assigns a new subject to the class with the smallest quadratic distance. This enables one to overcome the difficulty in estimating covariance matrices as in linear discriminant analysis while still incorporate correlation into the classifier. Extensive simulation studies and real data applications show that our approach outperforms support vector machine, the logistic regression and linear discriminant analysis for continuous outcomes, and outperforms the naive Bayes classifier, decision tree and logistic regression for discrete responses.

Unlike many other generative approaches, our method is derived with distributional assumptions on the first moment, as compared to the full distribution, and provides a classifier which handles both continuous and discrete responses. Another advantage of our classifier is that it possesses an inferential property under normality since it is built on the quadratic inference function, which is analog of minus twice log-likelihood, and the distance measure for the new classifier also follows a chi-squared distribution if it is assumed that the data follow a multivariate normal distribution. This provides a $p$-value interpretation as to how accurate the classification is for the new subject.
Efficient Classification for Longitudinal Data

by

Xianlong Wang

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented December 11, 2009
Commencement June 2010
Doctor of Philosophy thesis of Xianlong Wang presented on December 11, 2009

APPROVED:

______________________________
Major Professor, representing Statistics

______________________________
Chair of the Department of Statistics

______________________________
Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

______________________________
Xianlong Wang, Author
I am thankful for many people who make my study and thesis possible. I would like to thank all the faculty, staff, and students in the Statistics Department at Oregon State University. It is the systematic curriculum, financial and technical support, and knowledge-sharing environment which made my graduate study enjoyable.

I owe my deepest gratitude to my supervisor Dr. Annie Qu for her supervision, advice, and guidance on my research as well as in preparation of this thesis. She has offered unreserved advice and help to me, from graduate study, choice of thesis topic, research methodology, to thesis revision. I would like to heartily express my thanks to Dr. David Birkes. He has, as always, made available his support and help in my graduate study and on several critical points during my thesis research. Other than knowledge, his pursuit of objectivity and rigor, critical thinking and humbleness are among the scholarly qualities that influence and benefit me the most. I am sincerely grateful for Dr. Robert Smythe. His thoughtful lectures in statistical theory intrigued and strengthened my interest in statistics and well prepared me for my later research. Special thanks to Dr. Jeffery Arthur for taking his time to read this thesis and serving on my graduate committee. Thank you to Dr. Lan Xue for serving on my graduate committee. It is my pleasure to thank Dr. Xiaoli Fern for reviewing the thesis and serving as the graduate council representative.

Lastly, but the most importantly, I heartily thank my family, especially my grandparents and parents for their unequivocal encouragement and endless support in my life. Their love, forever, forms the most effective cushion against any stress and frustration in my study and is indispensable for me to finish my doctoral study.
# TABLE OF CONTENTS

1. INTRODUCTION ................................................................. 1
   1.1. Classification problems ............................................... 1
   1.2. Examples of classifiers ................................................. 2
       1.2.1 Generative and discriminative classifiers ....................... 2
       1.2.2 Bayes optimal classifier ........................................... 3
       1.2.3 Linear discriminant analysis & Fisher's linear discriminant ... 5
       1.2.4 Naive Bayes classifier ............................................. 6
       1.2.5 Support vector machine .............................................. 6
       1.2.6 Decision tree ..................................................... 7
       1.2.7 Logistic regression ............................................... 8
   1.3. Classification in longitudinal studies ............................... 9
       1.3.1 Longitudinal study examples .................................... 9
       1.3.2 Classifiers for longitudinal data ................................ 10
       1.3.3 Features of the new approach .................................. 11
   1.4. Organization of this thesis ......................................... 12

2. METHOD .............................................................................. 13
   2.1. Bias and variance in classifier design ............................... 13
   2.2. The new classifier ..................................................... 14

3. STATISTICAL LEARNING THEORY .......................................... 18
   3.1. Background ............................................................... 18
       3.1.1 Results in learning theory literature ......................... 19
   Finite hypothesis space .................................................. 19
   Infinite hypothesis space ................................................ 20
   Variable hypothesis space ............................................... 21
   3.2. Optimality of the new classifier .................................... 22
TABLE OF CONTENTS (Continued)

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3. Distributional property of $QD$ under normality</td>
<td>28</td>
</tr>
<tr>
<td>4. SIMULATION STUDIES</td>
<td>29</td>
</tr>
<tr>
<td>4.1. Time shift</td>
<td>29</td>
</tr>
<tr>
<td>4.2. Different altitudes</td>
<td>32</td>
</tr>
<tr>
<td>4.3. Binary responses</td>
<td>34</td>
</tr>
<tr>
<td>4.3.1 Simulation study-1</td>
<td>34</td>
</tr>
<tr>
<td>4.3.2 Simulation study-2</td>
<td>36</td>
</tr>
<tr>
<td>5. APPLICATION</td>
<td>38</td>
</tr>
<tr>
<td>5.1. Yeast ($Saccharomyces cerevisiae$) cell cycle data</td>
<td>38</td>
</tr>
<tr>
<td>5.2. Wild-type flies ($Drosophila melanogaster$) temporal data</td>
<td>39</td>
</tr>
<tr>
<td>5.3. Smoking data</td>
<td>40</td>
</tr>
<tr>
<td>5.4. PTSD data</td>
<td>41</td>
</tr>
<tr>
<td>6. SUMMARY AND DISCUSSION</td>
<td>43</td>
</tr>
<tr>
<td>BIBLIOGRAPHY</td>
<td>45</td>
</tr>
<tr>
<td>APPENDICES</td>
<td>51</td>
</tr>
<tr>
<td>A APPENDIX Figures</td>
<td>52</td>
</tr>
<tr>
<td>B APPENDIX Tables</td>
<td>72</td>
</tr>
<tr>
<td>C APPENDIX Proof of Theorem 1</td>
<td>75</td>
</tr>
<tr>
<td>D APPENDIX Proof of Theorem 2</td>
<td>77</td>
</tr>
<tr>
<td>E APPENDIX Classification error of the new classifier</td>
<td>79</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>A.1</td>
<td>Examples of hand-written digits taken from US zip codes</td>
</tr>
<tr>
<td>A.2</td>
<td>An classification example. The straight black line represents a simple model and the red wiggly line represents a complex model.</td>
</tr>
<tr>
<td>A.3</td>
<td>An SVM example with two classes.</td>
</tr>
<tr>
<td>A.4</td>
<td>A decision tree example with two classes.</td>
</tr>
<tr>
<td>A.5</td>
<td>The simulation with 2 hypothetical classes with a time shift.</td>
</tr>
<tr>
<td>A.6</td>
<td>Performance comparison of logistic regression, SVM and the new classifier.</td>
</tr>
<tr>
<td>A.7</td>
<td>Normal approximation to the error rate with respect to time shift and number of repeated measurements.</td>
</tr>
<tr>
<td>A.8</td>
<td>Upper bound of the error rate with respect to time shift and number of repeated measurements.</td>
</tr>
<tr>
<td>A.9</td>
<td>The 2 hypothetical classes when the mean function of class 2 is 2.5 times that of class 1.</td>
</tr>
<tr>
<td>A.10</td>
<td>Performance comparison of the new classifier, logistic regression, SVM and LDA.</td>
</tr>
<tr>
<td>A.11</td>
<td>Normal approximation to the error rate with respect to the amplification factor and number of repeated measurements.</td>
</tr>
<tr>
<td>A.12</td>
<td>Upper bound of the error rate with respect to the amplification factor and number of repeated measurements.</td>
</tr>
<tr>
<td>A.13</td>
<td>Performance comparison of the new classifier, logistic regression, decision tree and the naive Bayes classifier.</td>
</tr>
<tr>
<td>A.14</td>
<td>Normal approximation to the error rate with respect to the coefficients in the linear predictor and number of repeated measurements.</td>
</tr>
<tr>
<td>A.15</td>
<td>Upper bound of the error rate with respect to the coefficients in the linear predictor and number of repeated measurements.</td>
</tr>
<tr>
<td>A.16</td>
<td>Logit of probabilities over time for the two classes.</td>
</tr>
<tr>
<td>A.17</td>
<td>Performance comparison of the new classifier, logistic regression, decision tree and the naive Bayes classifier.</td>
</tr>
</tbody>
</table>
LIST OF FIGURES (Continued)

Figure                                           Page
A.18 Normal approximation to the error rate with respect to the coefficients 65
    in the linear predictor and number of repeated measurements.
A.19 Upper bound of the error rate with respect to the coefficients in the linear predictor and number of repeated measurements. 65
A.20 The 5 classes from the yeast cell cycle data. The thick lines are the fitted means for each class. 66
A.21 The combined 5 classes from the yeast cell cycle data. The thick lines are the fitted means for each class. 67
A.22 The 6 classes from the fruit fly gene expression data. The thick lines are the fitted means for each class. 68
A.23 The combined 6 classes from the fruit fly gene expression data. The thick lines are the fitted means for each class. 69
A.24 The two classes from the smoking data. The logit of high FEV1 probability for each class is displayed. 70
A.25 The 4 classes from the PTSD data. The logit of probability for each class is displayed. 71
LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.1</td>
<td>Simulated generalization error, its standard error and 95% confidence interval for SVM, Logistic regression, LDA, and the new classifier</td>
<td>72</td>
</tr>
<tr>
<td>B.2</td>
<td>Simulated generalization error, its standard error and 95% confidence interval for SVM, Logistic regression, LDA, and the new classifier</td>
<td>72</td>
</tr>
<tr>
<td>B.3</td>
<td>Simulated generalization error, its standard error and 95% confidence interval for Decision tree, Logistic regression, the naive Bayes classifier, and the new classifier</td>
<td>73</td>
</tr>
<tr>
<td>B.4</td>
<td>Simulated generalization error, its standard error and 95% confidence interval for Decision tree, Logistic regression, the naive Bayes classifier, and the new classifier</td>
<td>73</td>
</tr>
<tr>
<td>B.5</td>
<td>Performance of SVM, Logistic regression, LDA, and the new classifier on the yeast cell data</td>
<td>73</td>
</tr>
<tr>
<td>B.6</td>
<td>Performance of SVM, Logistic regression, LDA, and the new classifier on the fruit fly data</td>
<td>74</td>
</tr>
<tr>
<td>B.7</td>
<td>Performance of the new classifier, the naive Bayes classifier, decision tree and logistic regression on the smoking data</td>
<td>74</td>
</tr>
<tr>
<td>B.8</td>
<td>Performance of the new classifier, the naive Bayes classifier, decision tree and logistic regression on the PTSD data</td>
<td>74</td>
</tr>
</tbody>
</table>
EFFICIENT CLASSIFICATION FOR LONGITUDINAL DATA

1. INTRODUCTION

1.1. Classification problems

Pattern recognition has been a both intriguing and important topic, and has been successfully applied in practice for centuries. One fundamental problem in pattern recognition is to find a mapping from one space, conventionally referred as the feature space $X$ to another space $Y$, the target space.

To make the concept tangible, let us consider the following example. Given a sample of handwritten digits, 0 to 9, from many different people, can we design an algorithm so that it can learn from the given sample and recognize handwritten digits in the future? Figure A.1 shows an example of handwritten digits (Bishop, 2006). In this example, a point in the feature space is a vector of measurements on the handwritten digit and the target space is \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}. The answer is positive and algorithms have been designed and successfully applied in various contexts (LeCun et al, 1990; Gader and Khabou, 1996; Kussul et al., 2001). This example describes a typical classification problem where our goal is to predict the class labels, 0 to 9, from new handwritten digits. The given handwritten sample is called a training set, the algorithm employed is called a classifier, and the characteristics of the training examples, either qualitative or quantitative, used to make the prediction are called features.

It is common to refer regression to cases with continuous target values, and classification or clustering to cases with discrete target values as to whether the target space
is known. In essence, these two categories have no difference, although estimation and/or prediction are of interest in most regression problems, while we focus primarily on prediction in classification and clustering. For the later, the target value is aptly called class label since the feature space is divided into different classes according to the distinct discrete target values. Further, classification and clustering are also called supervised learning and unsupervised learning, respectively. While we will use results from clustering analysis in the literature, the main subject in our work will be classification.

1.2. Examples of classifiers

1.2.1 Generative and discriminative classifiers (Bishop, 2006)

Although there are various ways to understand a classifier, whether it is generative or discriminative is an informative aspect. For the purpose of discussion, we assume there are two classes, i.e., $Y = 1$ or $Y = 0$, and the principle can be easily extended to multiclass contexts.

In a classification problem, the features $X \in \mathbb{X}$ and class label $Y \in \mathbb{Y}$ follow a joint probability model $f_{X,Y}(x, y) = f_{X|Y}(x|y)f_Y(y)$. Given features $X = x$, generative classifiers estimate $f_{X|Y}(x|1)f_Y(1)$ and $f_{X|Y}(x|0)f_Y(0)$, and predict the class label to be 1 if $f_{X|Y}(x|1)f_Y(1) > f_{X|Y}(x|0)f_Y(0)$, or 0 otherwise. For example, linear discriminant analysis (LDA, see below) is a generative classifier since it assumes the features follow multivariate normal distribution within each class. Now we rewrite the joint probability model as, $f_{X,Y}(x, y) = f_{Y|X}(y|x)f_X(x)$. Given feature $X = x$, most discriminative classifiers directly approximate the conditional density $f_{Y|X}(y|x)$, and predict the class with larger conditional density. For instance, logistic regression is a discriminative classifier since we directly estimate $f_{Y|X}(y|x)$ through the logit link function.

Generally speaking, generative classifiers converge to the best model for the joint di-
tribution $f(x, y)$ whereas discriminative classifiers converge to the conditional distribution $f(y|x)$. Although a generative classifier generally has smaller variance than a discriminative classifier, the former tends to have more bias in reality due to the fact generative classifiers have more assumptions on the data and violation of these assumptions will result in bias. The comparison and choice between generative and discriminative learning have aroused considerable interest in statistics and machine learning over the last several decades. For an account of relevant literature, the interested reader is referred to Bishop and Lasserre 2007, Bouchard and Triggs 2004, Bouchard 2007, Efron 1975, Greiner and Zhou 2002, Jaakkola and Haussler 1999, Jaakkola et al. 1999, Jebara 2004, Liang and Jordan 2008, McCallum et al. 2006, Ng and Jordan 2002, Salojärvi et al. 2005, Schmah et al. 2009, Wettig et al. 2003, and Xue and Titterington 2008, 2009.

Next we introduce some common classifiers. This list by no means is intended to be inclusive. Rather, we use these classifiers for purpose of illustration, and most of them will be used in our simulations and applications later on.

### 1.2.2 Bayes optimal classifier (Johnson and Wichern, 2007)

The Bayes optimal classifier can be derived from at least two optimality criteria.

The first is to minimize the total probability of misclassification. Using this criterion, we have,

\[
P(\text{Misclassification}) = P(\text{Classify a subject from class 1 to class 2}) \\
\quad \quad \text{or classify a subject from class 2 to class 1}) \\
= P(\text{observation comes from class 1 and is misclassified}) \\
\quad + P(\text{observation comes from class 2 and is misclassified}) \\
= p_1 \int_{R_2} f_1(x)dx + p_2 \int_{R_1} f_2(x)dx \quad (1.1)
\]
where $R_1 = \{ x : \text{A subject with feature } x \text{ will be classified to class 1} \}$ and 
$R_2 = \{ x : \text{A subject with feature } x \text{ will be classified to class 2} \}$. The solution to this 
minimization problem can be shown to be,

$$
R_1 : \frac{f_1(x)}{f_2(x)} > \frac{p_2}{p_1} \\
R_2 : \frac{f_1(x)}{f_2(x)} < \frac{p_2}{p_1}
$$

(1.2)

To understand the second way to derive this classifier, suppose we will allocate a 
new subject $x$ to the class with the larger posterior probability, which, by Bayes’ theorem, 
is given by,

$$
P(Y = 1 | X = x) = \frac{P(Y = 1 \text{ and } X = x)}{P(X = x)} = \frac{P(X = x | Y = 1)P(Y = 1)}{P(X = x | Y = 2)P(Y = 2) + P(X = x | Y = 1)P(Y = 1)}
$$

$$
= \frac{p_1 f_1(x)}{p_1 f_1(x) + p_2 f_2(x)}
$$

(1.3)

$$
P(Y = 2 | X = x) = 1 - P(Y = 1 | X = x)
$$

$$
= \frac{p_2 f_2(x)}{p_1 f_1(x) + p_2 f_2(x)}
$$

Hence,

$$
R_1 : P(Y = 1 | X = x) > P(Y = 2 | X = x) \iff p_1 f_1(x) > p_2 f_2(x) \iff \frac{f_1(x)}{f_2(x)} > \frac{p_2}{p_1}
$$

(1.4)

$$
R_2 : \frac{f_1(x)}{f_2(x)} < \frac{p_2}{p_1}
$$

It is easy to see that when $p_2 = p_1, R_1 = \{ x : f_1(x) > f_2(x) \}$, and the Bayes optimal 
classifier is equivalent to a likelihood ratio test. It is the unique classifier that minimizes the 
misclassification error rate, and hence serves as a gold standard for evaluating classifiers.
In fact, we call a classifier consistent if it is asymptotically equivalent to the Bayes optimal classifier.

For the sake of discussion, we will assume $p_2 = p_1$ for the classifiers to follow, and the feature vector $X = (X_1, \ldots, X_p)$. Further, we have $n_1$ training examples from class 1 including their features $x_{11}, \ldots, x_{1n_1}$ along with class labels $y_{11}, \ldots, y_{1n_1}$, and $n_2$ training examples from class 2 including their features $x_{21}, \ldots, x_{2n_2}$ along with class labels $y_{21}, \ldots, y_{2n_2}$. Furthermore, we let $\bar{x}_1 = \frac{x_{11} + \cdots + x_{1n_1}}{n_1}$, $\bar{x}_2 = \frac{x_{21} + \cdots + x_{2n_2}}{n_2}$, and $S_{\text{pooled}} = \frac{(n_1 - 1)S_1 + (n_2 - 1)S_2}{n_1 + n_2 - 2}$, where $S_1$ and $S_2$ are the sample estimates of the variance-covariance matrices for class 1 and class 2, respectively.

1.2.3 Linear discriminant analysis & Fisher’s linear discriminant (Johnson and Wichern, 2007)

Linear discriminant analysis (LDA) assumes that $X|Y = 1 \sim N(\mu_1, \Sigma)$ and $X|Y = 2 \sim N(\mu_2, \Sigma)$. Applying Bayes’ theorem, we have,

$$R_1 : \frac{f_1(x)}{f_2(x)} > 1 \Leftrightarrow ln f_1(x) - ln f_2(x) > 0$$

$$\Leftrightarrow -\frac{p}{2} ln(\pi) - \frac{1}{2} ln |\Sigma| - \frac{1}{2}(x - \mu_1)'\Sigma^{-1}(x - \mu_1)$$

$$+ \frac{p}{2} ln(\pi) + \frac{1}{2} ln |\Sigma| + \frac{1}{2}(x - \mu_2)'\Sigma^{-1}(x - \mu_2)$$

$$\Leftrightarrow (\mu_1 - \mu_0)'\Sigma^{-1}x > (\mu_1 - \mu_0)'\Sigma^{-1}(\mu_1 + \mu_0)/2 \quad (1.5)$$

In practice, we can directly plug in the sample estimates, $\bar{x}_1$, $\bar{x}_2$, and $S_{\text{pooled}}$ for $\mu_1$, $\mu_2$, and $\Sigma$, respectively.

Although Fisher’s linear discriminant and LDA are usually exchangeable in practice, the former was derived without the distributional assumptions as in the latter. The goal of Fisher’s linear discriminant is to find a linear transformation of the features so that the populations are well separated in the transformed space. Specifically, we want to find
a linear combination $\hat{y} = \hat{a}'x$ so that $\hat{a} = \arg\max_a \frac{(a'\bar{x}_1 - a'\bar{x}_2)^2}{a'S_{\text{pooled}}a}$. That is, for any $a$, we first project $x$ perpendicular to the direction $a$, and Fisher’s linear discriminant finds the direction which separates the projected population means the most relative to within class scatter. It turns out that $\hat{a} = S_{\text{pooled}}^{-1}(\bar{x}_1 - \bar{x}_2)$, and the classification rule coincides with LDA.

1.2.4 Naive Bayes classifier (Mitchell, 1997)

As its name implies, the naive Bayes classifier makes use of Bayes’ rule as in the optimal Bayes classifier. Recall that we classify an observation $x$ according to $P(X = x|Y = y)$. In reality, the conditional probability model is often unknown, and is sometimes difficult to learn from the training examples. A naive approach is to assume the features are conditionally independent. That is, we can assume $P(X = x|Y = y) = P(X_1 = x_1, \ldots, X_p = x_p|Y = y) = P(X_1 = x_1|Y = y) \cdots P(X_p = x_p|Y = y)$. Even though the assumed probability model could be overly simplified, the performance of a naive Bayes classifier is comparable to many classifiers in practice.

1.2.5 Support vector machine (Christiani and Shave-Taylor, 2000)

The support vector machine (SVM) has been a popular machine learning and data mining technique ever since it was proposed. For ease of explanation, let $y = 1$ if an observation is from class 1, and $y = -1$ if from class 2. Further, we assume the two classes are linearly separable, and hence a hyperplane, $a'x + b$, exists such that $y = 1$ if and only if $a'x + b > 0$ and $y = -1$ if and only if $a'x + b < 0$. Since there are many solutions (Figure A.3), we would like to find a separating hyperplane which separates the two classes in some optimal way. A natural optimal criterion is to maximize the shortest distance from any observation to the separating hyperplane, that is, maximize the geometric margin.
Mathematically, this problem is formulated as follows,

\[
\max_{w, b, \gamma} \gamma \\
\text{subject to: } y_{ij} \frac{w' x_{ij} + b}{\|w\|} \geq \gamma \quad i = 1, 2; \quad j = 1, \ldots, n_i
\] (1.6)

It turns out that the optimal solution only depends on a subset of the training examples instead of the whole training set. This subset is referred to as the subset of support vectors, after which the classifier is named. SVM allows flexible hypothesis spaces, and hence can be applied in a wide range of domains. Although the classifier can be learned in polynomial time, immense data sets still pose a challenge in computation.

### 1.2.6 Decision tree (Han and Kamber, 2006)

The decision tree is a classification technique which uses a tree-like model (Figure A.4). A decision tree has 2 types of nodes, leaf nodes that give the final decision or prediction, and decision nodes that give intermediate decisions by splitting features, and, in turn, grow subtrees. Using a decision tree, a prediction can be made by following the tree, starting from the root until a leaf node is reached.

To understand how a decision tree is built, first define the entropy of the training set \( S \),

\[
Ent\sigma(S) = -\hat{p}_1 \log \hat{p}_1 - \hat{p}_2 \log \hat{p}_2
\] (1.7)

where \( \hat{p}_1 \) and \( \hat{p}_2 \) is the proportions of the training set from class 1 and class 2, respectively. For a feature \( X \) which takes values \( x_1, \ldots, x_v \), the associated entropy is given by,

\[
Ent\sigma(S_X) = -\frac{|S_{x_1}|}{|S|} Ent\sigma(S_{x_1}) - \cdots - \frac{|S_{x_v}|}{|S|} Ent\sigma(S_{x_v})
\] (1.8)

where \( S_{x_i} \) denotes the set of training examples whose feature \( X = x_i \).
Note that entropy can be interpreted as a measurement of impurity of a given set of training examples. Using entropy, we now define the information gain with respect to feature $X$ as,

$$ G(S, X) = \text{Entropy}(S) - \text{Entropy}(S_X) $$

At any step in constructing a decision tree, the feature with the largest information gain is selected to split the training examples, until all the features are in the tree or no information gain can be obtained.

The decision tree has gained much popularity in classification mainly due to the following advantages. It has a very flexible hypothesis space, produces a interpretable and easy-to-use classification rule, and provides the relative importance of the features.

Nevertheless, growing, training and pruning a decision tree can be computationally expensive, and overfitting is also a common difficulty in using decision trees. For example, a feature may need to be split many times before the optimal solution is found. Pruning decision trees involves comparing many subtrees, which also increases the computational burden.

1.2.7 Logistic regression (McCullagh and Nelder, 1989)

As a robust statistical model, logistic regression is commonly applied to classification problems. It assumes the following statistical model, $\text{logit}(P(Y = 1|X = x)) = \beta X$. A new observation $x$ is classified to class 1 or 2 according as $\text{logit}(P(Y = 1|X = x)) > 0$ or $\text{logit}(P(Y = 1|X = x)) < 0$.

In general, logistic regression is more robust than LDA since the latter make more assumptions. On the other hand, LDA is more efficient than logistic regression if the model assumptions for LDA hold (Efron, 1975).
1.3. Classification in longitudinal studies

For high-dimensional data such as in time-course yeast cell cycle microarray studies (Spellman et al., 1998; Eisen et al., 1998) and fruit fly data (Arbeitman et al., 2002; Ma et al., 2006), there are thousands of genes involved, and gene expressions are repeatedly measured over many time-points. It would be inefficient and inaccurate to treat the data as if it were from the same homogeneous population group. Statistically it is more sensible to model this type of data as a mixture of several groups. Therefore clustering and classification are commonly used to group observations into classes which have similar biological and statistical meanings and interpretations.

1.3.1 Longitudinal study examples

In a microarray study the genes often regulate various pathways with different biological functions. The main scientific interest typically is to cluster genes into different function groups, and classify a new gene without annotation to a certain functional group with the aid of existing classes.

Continuous responses are of primary interest in most microarray studies. However discrete responses are not uncommon. For example, a study of post-traumatic stress disorder (PTSD) interviewed people who survived residential fires in the Philadelphia area (Keane et al., 1996). Each survivor was interviewed three times, and each time the interviewer recorded whether or not the survivor had symptoms of PTSD, in addition to many other demographic variables. Since many subgroups may exist in the target population, it would be of interest to study the profile of each subgroup. Classification and clustering therefore will be natural and useful tools for this kind of data.

Another example is a study of atopic dermatitis, a skin disease affecting mainly children that runs in phases and is changeable over time. In this kind of study, the
presence of the symptom over time is recorded (Kuss et al. 2006). To improve prediction of risk factors and understand the disease progression, it would be important to classify children at different stages of disease.

1.3.2 Classifiers for longitudinal data

In principle, most traditional classifiers can be used in this context. Among the list are linear discriminant analysis (Fisher, 1936), support vector machines (Boser et al., 1992; Cortes and Vapnik, 1995; Vapnik, 1995; Parka et al., 2008), the k-nearest neighbor rule (Dasarathy, 1991), decision trees (Breiman, 1974), and the naive Bayes classifier (Hastie et al., 2001).

To develop classifiers for temporal gene expression data, the observations can be treated either as multivariate observations or as having a certain functional form. For the multivariate approach, the classical classification method is linear discriminant analysis (LDA). Choi (1972) proposed a classifier based on a mixed model assuming that the repeated measurements are equi-correlated within a class. Bagui and Mehra (1999) developed a multi-stage rank nearest neighbor rule. Brown et al. (2008) applied the support vector machine (SVM). Liang and Kelemen (2005) proposed regularized neural networks to model multiple heterogeneous temporal dynamic patterns of gene expressions. Lee (2004) and Rossi and Villa (2005, 2006) applied the functional SVM to the smoothed data. Leng and Müller (2006) classified time-course gene profiles using logistic regression. Park et al. (2008) showed that the functional support vector machine is effective in discriminating time-course gene expressions with predefined functions. However, these approaches either require the estimation of the covariance matrix, as in the LDA approach, or assume that observations are independent, as in logistic regression, or do not utilize the statistical information associated with the covariates, as in the SVM approaches.
1.3.3 Features of the new approach

We propose a new classification method for time-course gene expression data based on a generative approach using a semiparametric model. We assume that the repeated measurements of the same subject from the same class follow the same functional form. Our approach builds a classifier by modeling each class, defines a "quadratic" distance, and assigns a new subject to the closest class. Our approach is able to overcome the difficulty in estimating covariance matrices while still being able to incorporate correlation into the classifier. The new classifier outperforms traditional classifiers including support vector machines and logistic regression. We will provide asymptotic optimality theory, approximation, and upper bound to the generalization error for our classifier. In addition, our simulation studies and applications to time-course data all indicate that the generalization errors of the new classifier is improved, compared to other classifiers, for small and moderate sample sizes.

Another advantage of our classifier is that it possesses an inferential property under normality. A typical classifier assigns each subject into a class deterministically, that is, the subject either belongs to one class or another. One distinct feature of the new classifier is that in deterministic models it is assumed that the model perfectly fits the data and all measurements are made without error, whereas the new classifier is based on a probabilistic model that is valid for data that is subject to random error. The new classifier is built on the quadratic distance function, which also provides a statistical inference function. It measures the goodness-of-fit of the subject to be assigned for a chosen group. We will show that the quadratic distance function follows a chi-squared distribution given a normal distribution of the responses. The inferential property is able to provide a $p$-value interpretation as to how accurate the classification is for the new subject. In contrast to the deterministic rules of most classifiers, the new classifier is more
sensible for prediction, and provides better statistical interpretation from data.

1.4. Organization of this thesis

The rest of the work is organized as follows. We will derive the methodology in section 2 from quadratic inference functions, and statistical learning theory follows in section 3. In sections 4 and 5, two simulation studies and two real applications will be given, respectively. Section 6 summarizes our findings and discusses extensions and other considerations.
2. METHOD

In this section, we present the new classifier based on minimizing the quadratic
distance function, incorporating correlation structure, and modeling responses and covari-
ates within each class. To motivate our method, we first provide some background on the
challenges in classifier design.

2.1. Bias and variance in classifier design

To shed some light on the major challenges in supervised learning problems, consider
the example illustrated in figure A.2, where + and − represent objects labeled 1 or -1,
respectively. In addition, these objects are characterized by 2 features and consist of a
simple random sample from an unknown population.

If we were to define a boundary for these two classes, the + and the − classes,
we can choose the straight line, a rather simple model, or the wiggly line, a relatively
complex model. Since our goal is to predict future class labels accurately. If we choose
the simple model, the prediction would be less sensitive to a small change on the future
object, whereas the prediction based on the complex model would be more sensitive to
such small changes. This aspect of the classification performance is usually quantified by
the term variance.

On the other hand, the simple model might be further away from the true model
than the complex model on average over repeatedly drawn samples. This is because
the simple model has less flexibility in modeling the data while the complex model can
accommodate more complex structures. This aspect of the classification performance is
quantified by the term bias.
The tradeoff between bias and variance is always a challenge in the design of machine learning algorithms. However, the challenge also offers opportunities for developing new algorithms and improving existing algorithms. Given a classification problem, we would utilize all the available information to balance bias and variance. There are at least two sources of information one can utilize: the observed data information and prior subject matter knowledge. The observed data provide statistical information on the population for fitting models and drawing inferences, while prior subject matter knowledge refines our hypotheses and improves our inference. Every algorithm or model is crafted specifically from observed information and prior assumptions, and works only well when the prior assumptions are correctly specified. We intend to encode the maximum amount of information and utilize valid prior knowledge in order to develop the most powerful algorithm.

The above discussion indicates that bias and variance play a significant role in constructing classifiers. To simplify the problems in classification, let us assume our goal is to minimize the misclassification error (Hastie et al., 2001). As section 1.2.1 implies, a generative classifier is preferred if we are provided sufficient model assumptions. Alternatively, if we can model the data while making as few assumptions as possible, generative classification should be a desirable approach. In the following, we will derive our classifier based on this principle.

### 2.2. The new classifier

For longitudinal data, let $y_i(t)$ be a response variable and $x_i(t)$ be a $p \times 1$ vector of covariates, measured at time $t = t_1, \cdots, t_q$ for subjects $i = 1, \cdots, N$. We assume that the
model satisfies the first moment model assumption

$$\mu_i(t_j) = E\{y_i(t_j)\} = \mu\{x_i(t_j)\}'\beta,$$ (2.1)

where $\mu(\cdot)$ is a known inverse link function and $\beta$ is a $p$-dimensional parameter vector.

The quasi-likelihood equation (Wedderburn, 1974) for longitudinal data is

$$\sum_{i=1}^{N} \hat{\mu}_i'V_i^{-1}(y_i - \mu_i) = 0,$$

where $V_i = \text{Var}(y_i), y_i = (y_i(t_1), \ldots, y_i(t_q))', \mu_i = (\mu_{it_1}, \ldots, \mu_{it_q})'$, and $\hat{\mu}_i = \partial\mu_i/\partial\beta$. In practice, $V_i$ is often unknown, and the empirical estimator of $V_i$ based on sample variance could be unreliable, especially when there is a small number of replications relative to a large number of variance components. Liang and Zeger (1986) introduced generalized estimating equations to substitute $V_i$ by assuming $V_i = A_i^{1/2}RA_i^{1/2}$, where $A_i$ is a diagonal marginal variance matrix and $R$ is a common working correlation matrix, which only involves a small number of nuisance parameters. The advantage of the GEE is that even if the working correlation $R$ is misspecified, the GEE estimator of the regression parameter is still consistent, however, it is not efficient within the same class of estimating functions.

Qu et al. (2000) introduced the quadratic inference function by assuming that the inverse of the working correlation can be approximated by a linear combination of several basis matrices, that is,

$$R^{-1} \approx a_1M_1 + \cdots + a_mM_m$$ (2.2)

where $M_i$’s are symmetric matrices. The advantage of this approach is that it does not require estimation of linear coefficients $a_i$’s that can be viewed as nuisance parameters, since the generalized estimating equation is an approximate linear combination of the
components in the estimating functions:

\[
\bar{g}_N(\beta) = \frac{1}{N} \sum_{i=1}^{N} g_i(\beta) = \frac{1}{N} \left( \sum_{i=1}^{N} (\hat{\mu}_i)' A_i^{-1/2} M_1 A_i^{-1/2} (y_i - \mu_i) \right)
\]

Since the dimension of the above estimating equation is greater than the number of unknown parameters, we cannot set each component in (2.3) to be zero to solve for \( \beta \). Instead we estimate \( \beta \) by setting \( \bar{g}_N \) as close to zero as possible, in the sense of minimizing the quadratic function,

\[
\hat{\beta} = \arg \min_{\beta} \bar{g}_N' \Omega^{-1} \bar{g}_N
\]

where \( \Omega = Var(g_i) \). The covariance \( \Omega \) in (2.4) is often unknown, but can be estimated consistently by \( \bar{W}_N = N^{-1} \sum_{i=1}^{N} g_i g_i' \). The quadratic function,

\[
Q_N(\beta) = N \bar{g}_N' \bar{W}_N^{-1} \bar{g}_N,
\]

is called the quadratic inference function (Qu et al. 2000) since it provides an inference function for the regression parameters.

The new classifier is built on the quadratic distance, aptly named \( QD \). Suppose the new subject is assigned to class \( c \), \( c = 1, \ldots, C \), where \( C \) is the total number of class. The quadratic distance \( QD \) of the repeated measurements \( y \) for the new subject is defined as,

\[
QD_c(y) = g_c' W_c^{-1} g_c,
\]

where \( W_c \) is the estimated variance from the training data in the class \( c \), and \( g_c \) is obtained
as,

\[
g_c = \begin{pmatrix}
(\hat{\mu}_c)' \hat{A}_c^{-1/2} M_1 \hat{A}_c^{-1/2} (y - \hat{\mu}_c) \\
\vdots \\
(\hat{\mu}_c)' \hat{A}_c^{-1/2} M_m \hat{A}_c^{-1/2} (y - \hat{\mu}_c)
\end{pmatrix},
\]

(2.7)

where \(\hat{\mu}_c\) is the estimated mean of \(y\), \(\hat{A}_c\) is the estimated marginal variance for group \(c\), and \(\hat{\mu}_c\) is the estimate of \(\hat{\mu}_c\) as in (2.3), respectively. The following algorithm summarizes the classification rule to assign a new subject.

**Algorithm**

1. For each class, fit a semi-parametric regression model with the quadratic inference functions as in (2.3).

2. Compute \(QD_c(y)\) for each class \(c, c = 1, \cdots, C\).

3. Let \(m\) be the class where \(QD_c(y)\) reaches the minimum, that is, \(m = \arg \min_c QD_c(y)\).

4. Assign the new subject with the response \(y\) to the class \(m\).

For each class, step 1 captures the relevant statistical information by taking advantage of semi-parametric modeling and correlation information in (2.3) within the same class. This follows by computing the quadratic distance of a new subject to each class in step 2. Step 3 and 4 assign the new subject to the class with the nearest quadratic distance.

The new classifier benefits from incorporating information of correlation within subjects in addition to the semi-parametric modeling between the response and covariates. The proper modeling of semi-parametric function for each class is important to draw correct inference for the new subject. The goodness-of-fit test is given in Qu et al., 2000 and Lindsay and Qu, 2003. The basic idea is that if the first moment condition holds, i.e., \(g_N(\beta) = 0\) as in 2.3, \(Q(\hat{\beta})\) follows a \(\chi^2\) distribution with \(mp - p\) degrees of freedom.
3. STATISTICAL LEARNING THEORY

3.1. Background

As we indicated before, classification is a special type of statistical procedure on prediction. In nature, classification involves prediction which steps further than hypothesis testing since the former uses training examples to estimate the distribution and predict future responses whereas the latter makes decision on the distribution family based on the training sample.

The goal of statistical learning theory is to understand the statistical properties of learning algorithms, so that we can find better algorithms to improve them. Statistical learning theory also provides a framework to understand learning problems and algorithms, and guidance to generalizing the classification rule. A vast majority of the results in statistical learning theory provide upper/lower bounds and the estimates of the generalization error. These generalization error bound and estimates are not necessarily tight enough for practical usage, but offer knowledge of learning algorithms, and provide guidance for designing new algorithms and improving existing ones.


We summarize some learning results including error bounds in particular in the following section.
3.1.1 Results in learning theory literature

For illustration purpose, suppose there are \( m \) training examples, \( \{(x_1, y_1), \ldots, (x_m, y_m)\} \), from the population which consists of two classes. The class label is given according to \( y = f(x) \), where \( f \) is conventionally referred as the target concept, and our goal is to estimate or approximate \( f \) as best as possible from a hypothesis space \( H \). For the \( m \) training examples, \( h \in H \) is called a consistent hypothesis if \( h \) predicts all the class labels correctly for the training sample. The generalization error of \( h \) is defined as \( \varepsilon(h) = P(h(x) \neq f(x)) = E[I\{h(x) \neq f(x)\}] \), where \( I \) is a indicate function. Note that a consistent hypothesis does not have to exist in \( H \) for every possible training data set. An algorithm is called a consistent learner if it outputs consistent hypotheses.

**Finite hypothesis space**

**Consistent learning** If we have a finite hypothesis space, we have the following result for a consistent learner due to Blumer et al. (1987). Given training sample size \( m \), for any \( \delta \), with probability at least \( 1 - \delta \), the consistent learner produces a consistent hypothesis with generalization error

\[
\varepsilon \leq \frac{1}{m} \left( \ln |H| + \ln \frac{1}{\delta} \right),
\]

(3.1)

where \( |H| \) stands for the cardinality of \( H \). This bound is commonly referred to as the Blumer bound.

**Inconsistent learning** The above bound assumes the existence of consistent hypotheses in \( H \). However this assumption is not always valid. We have the following more general result if the existence of consistent hypotheses assumption is not valid.
For any $\delta \in [0, 1]$, and any $h \in H$, with probability at least $1 - \delta$,
\begin{equation}
\varepsilon(h) < \varepsilon_T(h) + \sqrt{\frac{\ln|H| + \ln(1/\delta)}{2m}}
\end{equation}
where $\varepsilon_T(h)$ is the training error.

In other words, the true error rate of any classifier is different from the training error by at most $\sqrt{\frac{\ln|H| + \ln(1/\delta)}{2m}}$.

**Infinite hypothesis space**

When infinite hypothesis space is of our interest, the error bound is closely related to the complexity of the hypothesis space, measured by its VC-dimension. Before we give the definition of VC-dimension is given, let us first introduce the notion of shattering. Given a set of $m$ points in the input space, we say hypothesis space $H$ shatters the set if, for any labeling over the set, there exists a consistent hypothesis in $H$.

**Example 1.** Let the sample space be $\mathbb{R}$, $H = \{ f : f(x) = I_{(a,b)}(x), a, b \in \mathbb{R} \}$. It is easy to check that $H$ shatters $\{x_1, x_2\}, x_1, x_2 \in \mathbb{R}$, i.e., $H$ shatters any 2 input points since any labeling on the 2 points can be produced by a hypothesis in $H$. But $H$ cannot shatter $\{x_1, x_2, x_3\}, x_1, x_2, x_3 \in \mathbb{R}$, i.e., $H$ does not shatter any 3 input points. For example, if we let $(x_1, x_2, x_3) = (-2, -1.5, 0)$ and $(y_1, y_2, y_3) = (1, 0, 1)$, there exists no hypothesis in $H$ which produces the class labels based on $(x_1, x_2, x_3)$.

VC-dimension was introduced by Vapnik and Chervonenkis (1971), and the reader is referred to Shelah (1972) and Sauer (1972) for further details and properties on this concept. The VC-dimension of a hypothesis space $H$ is defined to be the largest $m$ such that $H$ can shatter the $m$ points. In the above example, the VC-dimension of $H$ is 2 according to the definition.

**Consistent learning** For consistent learning in an infinite hypothesis space, the
following result gives the upper bounds on the error rate.

If hypothesis space $H$ has VC-dimension $d$, and a learning algorithm outputs a consistent hypothesis $h$ based on $m$ training examples, where

$$m \geq \frac{4}{\varepsilon} \left( d \ln \frac{12}{\varepsilon} + \ln \frac{2}{\delta} \right)$$

(3.3)

for any $\delta \in [0, 1]$. Then with probability at least $1 - \delta$, the generalization error $\varepsilon(h) \leq \varepsilon$.

**Inconsistent learning** For infinite hypotheses space $H$, if a learning algorithm outputs a hypothesis $h$ based on $m$ training examples with training error $\alpha$. Then for any $\delta \in [0, 1]$, with probability at least $1 - \delta$, the generalization error satisfies,

$$\varepsilon(h) \leq \alpha + \sqrt{\frac{d(\ln \frac{2m}{d} + 1) + \ln \frac{4}{\delta}}{m}}$$

(3.4)

This error bound consists of two parts. The first part is the training error $\alpha$, and the second part addresses the complexity of the hypothesis space. For a fixed hypothesis space, the error bound in (3.4) implies that we need to focus on minimizing the training error in searching for a candidate hypothesis.

**Variable hypothesis space**

In practice, the hypothesis space is not necessarily fixed. Instead, there could be a set of hypothesis space choices to avoid over-fitting or under-fitting algorithm. The error bound in (3.4) indicates that we should minimize the training error and reduce the complexity of the hypothesis space to minimize the upper bound. Since the minimal training error obtained from a simpler hypothesis space is usually bigger than from a more complex hypothesis space, there is a tradeoff between minimizing the training error and simplifying the hypothesis space.
One of our strategies is to utilize prior information in order to refine the hypothesis space. In our approach, we intend to optimize the performance of the classifier by incorporating the correlated nature from the longitudinal data and the model assumptions related to the first two moments of the distribution.

3.2. Optimality of the new classifier

In this section, the statistical learning theory for our new classifier will be developed. Specifically, we will investigate the optimality of the new classifier in Theorem 1, upper bound and approximation to the generalization error in Lemma 1 and 2, respectively, and the asymptotic distribution property of the quadratic distance $QD$ under normality in Theorem 2.

We reformulate $g_c$ in (2.7) as,

$$g_c = \tilde{T}'_c (y - \hat{\mu}_c), \quad (3.5)$$

where $\tilde{T}'_c \to_p T'_c$ with

$$T'_c = \begin{pmatrix} (\hat{\mu}_c)' A_c^{-1/2} M_1 A_c^{-1/2} \\ \vdots \\ (\hat{\mu}_c)' A_c^{-1/2} M_m A_c^{-1/2} \end{pmatrix} \quad (3.6)$$

Note that the above convergence holds since GEE and QIF estimators are consistent regardless whether the correct working correlation matrix $R$ is assumed or not.

Theorem 1 presents the optimality of the new classifier. We provide the proof in the Appendix.

**Theorem 1.** Under the first moment assumption in (2.1) and regularity conditions,
the new classifier is asymptotically optimal based on $T_{c_i}y_i$ in misclassification error rate if

$$ T_{c_i}y_i \overset{d}{=} \mu_i + \Sigma_i^{\frac{1}{2}}u \text{ and } |\Sigma_i| = |\Sigma_j| \quad i, j = 1, \ldots, C $$

where $u$ is a random vector with probability density function $f_0(u'u)$ such that $f_0(\cdot)$ is a strictly decreasing density function in $[0, \infty)$.

**Remark**: the requirement, $|\Sigma_i| = |\Sigma_j|$, in Theorem 1 might not be necessary here since $\log|\Sigma_i|$ tends to be closer in the logarithm scale, even if $\Sigma_i$’s are different (Velilla and Hernández, 2005).

Note that the condition on monotone $f$ in Theorem 1 is satisfied in normal case. For example, if $X = (X_1, \ldots, X_p) \sim N(\mu, \Sigma)$, then $X$ can be generated as, $X = \mu + \Sigma^{\frac{1}{2}}U$, where $U \sim N(0, I)$. The probability density function of $U$ is given by

$$ f(u_1, \ldots, u_p) = \frac{1}{(2\pi)^{p/2}}\exp\left(-\frac{1}{2}u'u\right) $$

$$ = \frac{1}{(2\pi)^{p/2}}\exp(-u'u) $$

$$ = f_0(u'u) $$

where $f_0(t) = \frac{1}{(2\pi)^{p/2}}\exp(-t)$, and $f_0$ is a strictly decreasing function over $[0, \infty)$.

In Corollary 1, we provide the asymptotic optimality under the inverse-location regression model which was initially introduced for developing dimension reduction in regression problems (Cook and Yin, 2001). The inverse-location assumption is a special case in (3.7).

**Corollary 1.** Under the first moment assumption (2.1) and regularity conditions, the new classifier based on $T_{c_i}y_i$ is asymptotically optimal with minimum misclassification
error rate if \( T'_c y_i \) follows the inverse location regression model,

\[
T'_c y_i = \mu_i + \Sigma^\frac{1}{2} u \quad i = 1, \ldots, C
\]  \hspace{1cm} (3.8)

The inverse-location regression model can be viewed as the following in the classification framework. Suppose \( X \) is the random vector of covariates or features, \( Y \) is the response variable which represents the class label. Then the inverse location model is satisfied if \( X - \mu e_Y \perp Y \), where \( \mu = (\mu_1, \cdots, \mu_C) \) is the \( p \times C \) matrix of conditional means with \( \mu_c = E(X|Y = c) \) and \( e_Y \) is the \( C \times 1 \) indicator vector, with each \( i \)th component indicating whether the class label of \( Y \) is in \( i \)th class or not. In Theorem 1, if \( T'_c y_i = \mu_i + \Sigma^\frac{1}{2} u \), i.e., \( T'_c y_i - \mu_i = \Sigma^\frac{1}{2} u \), then \( T'_c y_i - \mu_i \perp Z \) where \( Z \) is the class label. One of the major properties of an inverse location regression model is that the distribution of \( X|Y \) depends on \( Y \) only through \( E(X|Y) \).

The implication of the inversion-location regression model (3.8) is that differentiating classes based on the information of conditional mean from each class is a sensible approach. This is reflected in our classifier since \( QD \) is computed from the estimated conditional mean of each classes as in (3.5). On the other hand, given information in the model assumption (2.1), our classifier not only takes advantage of the conditional mean of each class, but also reduces the dimensionality of the problem since the dimension of \( T'_c y \) is almost always less than that of \( y \).

The normal distribution, t-distribution, Laplace (double exponential) distribution, and generalized hyperbolic secant distributions (Harkness, 1968) all satisfy the inverse-location regression model assumptions. Note some symmetric distributions such as Cauchy distribution do not satisfy the assumptions. This is because Cauchy distribution does not have well defined mean and variance, and thus cannot be categorized with the same mechanism as specified in Theorem 1.
In practice, the statistical distribution of the outcome is often unknown. When the elliptical symmetry assumption is in doubt, it is helpful to perform hypothesis testing. For relevant testing procedures, we refer the reader, but are not limited to, Sakhanenko (2002), Huffer and Park (2007), Beran (1979), and Manzotti et al. (2002).

For most small or moderate sample sizes, we assume a relatively simple structure on the mean in (2.1). On the other hand, if we have large samples, we could impose more complex structures on the mean function. Under such circumstances, it turns out that the new classifier is equivalent to LDA. We summarize this fact in the following corollary.

**Corollary 2.** Suppose the assumptions in Theorem 1 hold, then the new method is asymptotically equivalent to LDA if \( \text{range}(T_c) = \mathbb{R}^q \), where \( q \) is the number of repeated measurements.

The corollary implies that, asymptotically, LDA is a special case of our method. When the repeated measurements from each class follow multivariate normal distribution with a common variance matrix, LDA produces optimal solutions for the classification problem. However, LDA requires calculating the covariance matrix, which could be problematic if the sample size is small. The new classifier can overcome the difficulty in calculating the covariance matrix for small samples, but is also able to model each class to maximize the discriminant power. Furthermore, the new method produces the same classification boundary regardless of the error structure, whereas LDA yields linear classification boundary if the classes have a common error structure and nonlinear boundary if classes have different error structures.

In theorem 1, we have shown that the new classifier is asymptotically optimal for elliptically symmetric distributions. In the following two lemmas, we investigate the theoretical properties of the upper bound for the generalization error. This can provide some guidance to assess error rates in practice.

**Lemma 1.** Under the first moment assumption (2.1), for a two-class classification
A problem with equal probability prior, the misclassification error rate for the new classifier is bounded above as,

\[
P(\text{misclassify a subject}) \leq \frac{1}{2} \left( 1 + \frac{(\mu_2 - \mu_1)'A_2(\mu_2 - \mu_1)}{\sqrt{2\text{tr}((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4(\mu_2 - \mu_1)'A_2\Sigma_1A_2(\mu_2 - \mu_1)}} \right)^2 + \frac{1}{2} \left( 1 + \frac{(\mu_1 - \mu_2)'A_1(\mu_1 - \mu_2)}{\sqrt{2\text{tr}((A_2 - A_1)\Sigma_2(A_2 - A_1)\Sigma_2) + 4(\mu_1 - \mu_2)'A_1\Sigma_2A_2(\mu_2 - \mu_1)}} \right)^2\quad (3.9)
\]

The above formula can be further simplified under certain circumstances. For example, if the repeated measurements on subject \(i\) from class \(c\), \(y_i = (y_{i1}, \ldots, y_{in})\), follow a multivariate norm distribution \(N(\mu_c, \Sigma)\), and \(\mu_c = X\beta_c\), then from (3.6), \(T_1 = T_2\), which implies \(W_1 = T_1'\Sigma T_1 = T_2'\Sigma T_2 = W_2\), and hence \(A_1 = T_1W_1^{-1}T_1 = T_2W_2^{-1}T_2 = A_2\). Let \(T_1 = T_2 = T\) and \(A_1 = A_2 = A\). Consequently, the upper bound in Lemma 1 now becomes,

\[
\frac{1}{2} \left( 1 + \frac{1}{\sqrt{4(\mu_2' - \mu_1')(A_2\Sigma A_2)'(\mu_2 - \mu_1)}} \right)^2 + \frac{1}{2} \left( 1 + \frac{1}{\sqrt{4(\mu_1' - \mu_2')(A_1\Sigma A_1)'(\mu_1 - \mu_2)}} \right)^2
\]

further, if \(A\Sigma A = A\), i.e., \(\Sigma\) is the generalized inverse matrix of \(A\), the upper bound is further simplified as,

\[
P(\text{misclassify a subject}) \leq \frac{1}{2} + \frac{1}{4} + \frac{1}{2} + \frac{1}{4} = \frac{4 + (\mu_2 - \mu_1)'A(\mu_2 - \mu_1)}{4} \quad (4)
\]

Lemma 1 indicates that if \((\mu_2 - \mu_1)'A(\mu_2 - \mu_1) > 4\), then the misclassification error
rate is strictly less than 0.5. Even though the upper bound itself seems to be loose, the learner is guaranteed to be better than random guessing. In machine learning field, such a classifier often serves as a base learner in boosting (Hastie et al., 2001) to produce a stronger classifier.

**Lemma 2.** Under the first moment assumption (2.1), for a two-class classification problem with equal probability prior, the normal approximation to the misclassification error rate for the new classifier is given by,

\[
P(\text{misclassify a subject}) \\
\approx 1 - \frac{1}{2} \Phi \left( \frac{(\mu_2 - \mu_1)'A_2(\mu_2 - \mu_1) - tr((A_1 - A_2)\Sigma_1)}{\sqrt{2tr((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4(\mu_2' - \mu_1')A_2\Sigma_1A_2(\mu_2 - \mu_1)}} \right) \\
- \frac{1}{2} \Phi \left( \frac{(\mu_1 - \mu_2)'A_1(\mu_1 - \mu_2) - tr((A_2 - A_1)\Sigma_2)}{\sqrt{2tr((A_2 - A_1)\Sigma_2(A_2 - A_1)\Sigma_2) + 4(\mu_1' - \mu_2')A_1\Sigma_2A_1(\mu_1 - \mu_2)}} \right)
\]

(3.10)

Similar to Lemma 1, if \( T_1 = T_2 \), the approximation can be further simplified as follows,

\[
P(\text{misclassify a subject}) \\
\approx 1 - \frac{1}{2} \Phi \left( \frac{(\mu_2 - \mu_1)'A_2(\mu_2 - \mu_1)}{\sqrt{4(\mu_2' - \mu_1')A_2\Sigma_1A_2(\mu_2 - \mu_1)}} \right) - \frac{1}{2} \Phi \left( \frac{(\mu_1 - \mu_2)'A_1(\mu_1 - \mu_2)}{\sqrt{4(\mu_1' - \mu_2')A_1\Sigma_2A_1(\mu_1 - \mu_2)}} \right)
\]

Moreover, if \( A\Sigma A = A \), where \( \Sigma \) is the generalized inverse matrix of \( A \), the approx-
imation can be further simplified as,

\[ P(\text{misclassify a subject}) \approx 1 - \Phi \left( \frac{\sqrt{(\mu_2 - \mu_1)\'A(\mu_2 - \mu_1)}}{2} \right). \]

### 3.3. Distributional property of \( QD \) under normality

Another advantage of the proposed approach is the inferential property for the new classifier under normality. This is summarized in Theorem 2.

**Theorem 2.** Suppose \( y_1, \cdots, y_N, y_{N+1} \) are independent random vectors from the same class such that \( y_i \sim N(\mu, V) \), and \( x_i(t) \) is a \( p \times 1 \) vector of covariates, \( i = 1, \cdots, N, N + 1 \). If (2.2) holds, then \( QD(y_{N+1}) \sim \chi^2_{mp} \), where \( QD(y_{N+1}) \) is computed according to (2.6) and \( y_{N+1} \) is a new subject.

In addition to using \( QD \) for classifying subjects, this inferential property also provides a confidence measure as to whether the new subject belongs to a given class. To see this, let \( p_c = Pr(T > QD_c(y)) \), where \( T \sim \chi^2_q \), and \( pv_c \equiv \frac{p_c}{\sum_{i=1}^{C} p_i} \). Then \( pv_c \) provides a \( p \)-value interpretation as to how confident and accurate the classification is for the new subject \( y_{N+1} \). That is, the larger \( pv_c \), the more confidence we are in classifying the new subject into class \( c \).
4. SIMULATION STUDIES

In this section, we will show the performance of the new classifier through simulation studies. The accuracy of upper bound (3.9) in Lemma 1 using normal approximation, and (3.10) in Lemma 2 will also be assessed.

We will first use two simulation studies to demonstrate the performance of our method for continuous responses. We compare the performance of our method with that of support vector machine, logistic regression, and linear discriminant analysis.

Error rate from Leave-One-Out cross validation will be used to evaluate their performance. We use R version 2.5.0 to implement our method. Specifically, we use the built-in function, ”svm” in package ”e1071” for SVM with the choices of linear and radial basis kernels, and ”vglm” in package ”vgam” which implements the multinomial logistic regression.

4.1. Time shift

For time course gene expression data, two functional groups may follow a similar pattern except one has a delay in time relative to another. To investigate this setup, we design the following simulation.

Let $\mathbf{Y} = (Y_{t_1}, \cdots, Y_{t_{50}})$ be repeated measurements on a subject $t$ from class 1, and

$$E(Y_{t_i}) = 3t_i + 4 \sin(t_i) - 2 \cos(t_i) + 2 \sin(2t_i) \quad i = 1, \cdots, 50. \quad (4.1)$$
We assume the correlation structure of the repeated measurements is $AR - 1$, that is

$$\text{Cov}(\mathbf{Y}) = \sigma \begin{bmatrix} 1 & \rho & \ldots & \rho^{49} \\ \rho & 1 & \ldots & \rho^{48} \\ \vdots & \vdots & \ddots & \vdots \\ \rho^{49} & \rho^{48} & \ldots & 1 \end{bmatrix}$$

(4.2)

where $\sigma = 15$, and $\rho = 0.85$.

For class 2, let $\mathbf{Y} = (Y_{t1}, \ldots, Y_{t50})$ be repeated measurements of a subject with the same covariance matrix as class 1, except that

$$E(Y_{ti}) = 3(t_i - 0.5) + 4 \sin(t_i - 0.5) - 2 \cos(t_i - 0.5) + 2 \sin(2(t_i - 0.5)), i = 1, \ldots, 50. \quad (4.3)$$

Clearly, the mean function of class 2 simply shifts to the right from class 1 by 0.5 in time.

We generate data with small sample sizes for class 1 and class 2 subjects where $n_1 = 25$ and $n_2 = 25$, respectively. To illustrate the discriminant power between two classes, we generate a sequence of random vectors with the same covariance structure as above, and the mean function of $\mathbf{Y}(s) = (Y(s)_{t1}, Y(s)_{t2}, \ldots, Y(s)_{t50})$ is,

$$E(Y(s)_{ti}) = 3(t_i - s) + 4 \sin(t_i - s) - 2 \cos(t_i - s) + 2 \sin(2(t_i - s)), i = 1, \ldots, 50, \quad (4.4)$$

where $s \in [0, 0.5]$. For two extreme values of $s$, $\mathbf{Y}(s)$ belongs to class 1 when $s = 0$, and belongs to class 2 when $s = 0.5$. For each $s \in [0, 0.5]$, we apply our method, SVM, logistic regression and LDA to predict its class label. For LDA approach, we tried different non-singular covariance matrices to evaluate its performance. For example, we used a diagonal matrix with the $i^{th}$ diagonal element being the marginal variances for measurement at each time $i$. Note that the small sample sizes will yield singular sample covariance matrices.
The desirable classifier should be sensitive for prediction towards the closer class.

Figure A.5 displays the two classes with a time shift, and Figure A.6 provides the probability of prediction when the time shift moves from 0 to 0.5. In particular, for the two end points $s = 0$ and $s = 0.5$, it provides the classification performance. We can compute the generalization error from two end points. Assuming equal class prior, Table B.1 provides the generalization errors (GE), the standard errors of GE and 95% confidence interval of GE for each classifier.

Figure A.6 indicates that the new classifier has the highest sensitivity for the correct prediction, followed by the performances of the LDA and logistic regression. The SVM has the lowest sensitivity for the correct prediction. In addition, None of the classifiers has discriminant power when time shifts to the middle with $s = 0.25$, and when time shift $s$ moves to either end, all the above classifiers provide more discriminant power. Nevertheless, the sensitivity of the new classifier for the correct prediction is clearly higher than the other classifiers.

The normal approximation and upper bound for the generalization error rate are not too close to our error rate in the simulation. However, the normal approximation and upper bound decrease as the time shift between two classes increases or the number of repeated measurements increases, as shown in figure A.7 and A.8. The upper bound gives an absolute upper limit to the error rate, whereas the normal approximation relies on how good the approximation is. For this simulation study, the normal approximation does not appear to be close to the true distribution, and the error rate derived from the approximation turns out to be rather conservative. The upper bound is less than 0.5, which implies that the new classifier is at least better than random guessing.
4.2. Different altitudes

For the second simulation study, we generate data with the same functional form but different altitude for the two classes.

The repeated measurements $Y = (Y_{t_1}, \cdots, Y_{t_{50}})$ from class 1 has the mean function

$$E(Y_{t_i}) = 3t_i + 4\sin(t_i) - 2\cos(t_i).$$

The repeated measurements from class 2 has mean

$$E(Y_{t_i}) = 7.5t_i + 10\sin(t_i) - 5\cos(t_i).$$

Given any time point $t_i$, the mean function for class 2 is 2.5 times of mean function in class 1. The covariance of $Y$ for both classes are the same as in simulation study 1. Both sample sizes are 25.

We generate a sequence of random vectors, $Y(c) = (Y(c)_{t_1}, Y(c)_{t_2}, \cdots, Y(c)_{t_{50}})$, where $c \in [1, \cdots, 2.5]$, and the mean function of $Y(c)$ satisfies

$$E(Y(c)_{t_i}) = c\{3t_i + 4\sin(t_i) - 2\cos(t_i)\}$$

This sequence of random vector can be interpreted as follows. When $c = 1$, $Y(c)$ belongs to class 1; as $c$ increases, $Y(c)$ is further away from class 1, and it belongs to class 2 when $c = 2.5$. For each $c$ between 1 and 1.5, we apply our method, SVM and logistic regression to classify the new subject. Figure A.9 displays the two classes, and Figure A.10 provides the probability of prediction when $c$ takes different values. Assuming equal probability prior for two classes, Table B.2 shows the generalization error, the standard
errors of GE and 95% confidence interval of GE for each classifier. In comparison, we get similar results as for the time shift simulation study. Note that the SVM does not have any discriminating power as the other three classifiers.

In contrast to the previous simulation study, the error rate estimator from the normal approximation is much closer to the actual error rate. The theoretical upper bound is also sharper. The upper bound guarantees that the new classifier works much better than random guessing. In this setting, the normal approximation and upper bound decrease as we increase the amplification scale or the number of repeated measurements for each subject, as shown in figure A.11 and A.12.

Both simulation studies show that the new classifier has the highest sensitivity for prediction towards the closest class, followed by the performances of the LDA and logistic regression. The SVM has the lowest sensitivity for the correct prediction. This result is not surprising according to existing learning theory. It is well-known that under normality, LDA outperforms other classifiers including SVM and logistic regression. On the other hand, in finite samples, the estimator of the covariance matrix could be unstable, or could be singular occasionally, therefore the performance of the LDA could be undermined when the sample size is small. However, the new classifier does not require estimating the covariance matrix, and consequently it leads to smaller generalization error.

The above simulations show the performance of the new classifier for continuous responses. The classification procedure can also be applied to discrete responses since our approach is not restricted for continuous case only. In the same venue, we will compare the performance of the new classifier with several other popular classifiers for categorical responses such as naive Bayes classifiers, decision trees, and logistic regression. To explain our selection of these alternative classifiers, note that LDA is mainly suitable for continuous responses, and the SVM is also designed for continuous data. On contrary, decision tree is based on partitioning feature space, and the naive Bayes classifier is derived from Bayes
probability model, are two popular methods for discrete responses, and therefore they are included in our comparison. We also include logistic regression since it is applicable for both continuous and discrete responses.

We use "NaiveBayes" in R-package "e1071" for Naive Bayes classifier, and "rpart" in R-package "rpart" for decision tree. For decision tree, we used grid search on the parameter space to find the best performance.

4.3. Binary responses

We provide two simulation studies to demonstrate the performance of our method for longitudinal binary data. Function \texttt{rmvbin} in R package "bindata" is used to generate the binary data for study 1, and function \texttt{ep} in R package "mvtBinaryEP" for study 2. We will compare the performance of our method with that of the naive Bayes classifier, decision tree, and logistic regression.

4.3.1 Simulation study-1

Let $Y = (Y_{t1}, \ldots, Y_{t100})$ be repeated measurements with binary outcome. The logit of mean function for responses from two classes are

$$
\text{logit}(E(Y_{ti})) = \begin{cases} 
0 & i = 1, \ldots, 100, \text{ if } Y \text{ is from class 1} \\
3t_i & i = 1, \ldots, 100, \text{ if } Y \text{ is from class 2}
\end{cases}
$$

and the covariance of $Y$ has exchangeable correlation structure with the correlation parameter as 0.6. More specifically, the logit of each mean function follows a linear trend, but with different slopes for two classes. Again, the sample sizes for class 1 and class 2 are $n_1 = 25$ and $n_2 = 25$, respectively.

To illustrate the discriminating power of our method, we generate a sequence of
random vectors, $Y(c) = (Y(c)_{t_1}, Y(c)_{t_2}, \cdots, Y(c)_{t_{100}}), c \in [0, 3]$, with the logit of mean

$$logit(E(Y(c)_{t_i})) = ct_i \quad 0 \leq c \leq 3, \ i = 1, \cdots, 100 \quad (4.5)$$

When $c = 0$, $Y(c)$ is generated from class 1; as $c$ increases, $Y(c)$ is further away from class 1, and it belongs to class 2 when $c = 3$. For each $c$ between 0 and 3, we predict the class label applying our method, the naive Bayes classifier, decision tree and logistic regression. Figure A.13 provides the probability of prediction for class 2 when $c$ moves from 0 to 3. Assuming equal probability prior for both classes, Table B.3 provides the generalization error, the standard errors of GE and 95% confidence interval of GE for each classifier. Figure (A.14) and Figure (A.15) show the normal approximation to the misclassification error rate and upper bound, respectively.

Figure A.13 shows that all 4 methods have comparable discriminant power. However, our method has the highest sensitivity for the correct prediction, followed by the naive Bayes classifier and logistic regression, and the decision tree has the worse performance. In addition, none of the classifiers has much discriminant power if the new subject is in between two classes with $c \approx 1.5$. When the new subject gets closer to either class, all four classifiers show more discriminant power. Nevertheless, the sensitivity of the new classifier for correct prediction increases slightly more than the other three classifiers.

Table B.3 also shows that there is discrepancy between the normal approximation to the error rate and the error rate of the new classifier. The normal approximation formula underestimates the error rate. This is because the binary response are typically not appropriate for normal approximation. On the other hand, the theoretical upper bound to the error rate is quite close to the new classifier error rate in our simulation. The normal approximation and upper bound decrease as the time shift between two classes increases or the number of repeated measurements increases, as shown in figure A.14 and
4.3.2 Simulation study-2

In the second simulation study for the binary data, we generate two classes with repeated measurements has logit of mean functions

\[
\text{logit}(E(Y_{t_i})) = \begin{cases} 
  t_i - 0.9 \sin(2t_i) & i = 1, \ldots, 100, \text{ if } Y \text{ is from class 1} \\
  t_i + 0.3 \sin(2t_i) & i = 1, \ldots, 100, \text{ if } Y \text{ is from class 2}
\end{cases}
\]

and \(\text{Cov}(Y)\) has an AR-1 correlation structure with 0.8 correlation coefficient. Or equivalently, the logit of each mean function is a linear combination of time \(t_i\) and a sine function, but with different coefficients for the sine function component. Here the sample sizes for both classes are 25.

To illustrate the discriminant power of our method, we generate a sequence of random vectors for a new subject with repeated measurements

\[Y(c) = (Y(c)_{t_1}, Y(c)_{t_2}, \ldots, Y(c)_{t_{100}}), c \in [-0.9, 0.3].\]

The new subject given each \(c\) has the same AR-1 covariance structure as above, and the logit mean function is

\[
\text{logit}(E(Y(c)_{t_i})) = t_i + csin(2t_i), \; i = 1, \cdots, 100. \quad (4.6)
\]

When \(c = -0.9\), \(Y(c)\) belongs to class 1; as \(c\) increases, \(Y(c)\) is further away from class 1, and it belongs to class 2 when \(c = 0.3\). For each \(c\) between -0.9 and 0.3, we apply the new classifier, the naive Bayes classifier, decision tree and logistic regression to classify the new subject. Figure A.16 displays the probabilities over time for the two classes. Figure A.17 provides the probability of prediction when \(c\) moves from -0.9 to 0.3. Assuming equal probability prior for two classes, Table B.4 provides the generalization error, the standard errors of GE and 95\% confidence interval of GE for each classifier. Figure (A.18)
and Figure (A.19) show the normal approximation to the misclassification error rate and upper bound, respectively.

Figure A.17 shows the performance of four classifiers. The new classifier picks up discriminant power faster among the 4 classifiers when the new subject gets closer to either class. Logistic regression almost has no discriminant power when \( c \) ranges from \(-0.4\) to \(0.1\) as the power curve stays flat around 0.5. The decision tree tends to classify the new subject to class 2 more likely than to class 1. However, the decision tree does not perform as well as the new classifier when \( c \) is closer to either end of class. The naive Bayes classifier turns out to be a comparable classifier in our simulation. It shows similar discriminant power trend as the new classifier, however it grows slower than the new classifier.

Both the normal approximation and the upper bound for the error rate are not very close to the simulation error rates. Similar to the previous setting, the normal approximation underestimates the misclassification error rate for binary responses since the normal approximation is not a good approximation for binary outcomes. The theoretical upper bound is quite loose in this simulation. Nevertheless, the normal approximation and upper bound decrease as the time shift between two classes increases or the number of repeated measurements increases, as shown in figure A.18 and A.19.
5. APPLICATION

In this section, we compare the performance of our method to SVM and logistic regression using two real data sets with continuous responses and two data sets with binary responses.

For two continuous data sets, they both involve multiclass classification. Error rate from Leave-One-Out cross validation will be used to evaluate their performance. We used R version 2.5.0 to implement our method.

5.1. Yeast (Saccharomyces cerevisiae) cell cycle data

The yeast cell microarray data contains the expression profiles of 2467 budding yeast Saccharomyces cerevisiae genes over 79 time points (Spellman et al., 1998), including the cell division cycle after synchronization by alpha factor arrest (ALPH, 18 time points), centrifugal elutriation(ELU, 14 time points), a cdc15 mutant (CDC15, 15 time points), sporulation (SPO, 11 time points), shock by high temperature (HT, 6 time points), reducing agents (D, 4 time points) and low temperature (C, 4 time points). Eisen et al. (1998) conducted a clustering analysis for 75 genes in the centrifugal elutriation time period, and assigned these genes to 5 different functional groups.

We will analyze the 75 genes in the centrifugal elutriation experiment in the following 5 groups: spindle pole body assembly and function, the proteasome, chromatin structure, the ribosome and translation, and DNA replication. Figure A.20 shows individual classes along with the fitted mean function using quadratic inference functions, and Figure A.21 displays all the classes together in one graph.

To model each class, we use cubic truncated power spline basis with equal-spaced
knots. That is, we choose knots, $\kappa_1, \cdots, \kappa_v$ to fit the mean model

$$E(y_i(t)) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + (t - \kappa_1)_+^3 + \cdots + (t - \kappa_v)_+^3$$  \hspace{1cm} (5.1)

We select 3 knots to fit the model and the $\chi^2$ goodness-of-fit test based on the quadratic inference functions (Qu and Li, 2006) shows that it is adequate for the model fitting.

Table B.5 summarizes the classification performance of each classifier on generalization errors. The performance comparison is consistent with our finding in the simulation studies. That is, our classifier has the lowest generalization errors (0.053), and the corresponding standard errors (0.026) are also the lowest, followed by the performances of the LDA, logistic regression and SVM.

5.2. Wild-type flies (*Drosophila melanogaster*) temporal data

Arbeitman et al. (2002) studied the mRNA levels of 4028 genes in wildtype flies (*Drosophila melanogaster*) with cDNA microarrays over 70 time-points spanning fertilization, spanning embryonic, larval, pupal stages and the first 30 days of adulthood. Ma et al. (2006) developed a data-driven clustering method for this data set. We choose 6 clusters consisting of 1120 genes in the fertilization stage from their clustering result to make comparisons for four classifiers. Figure A.22 and Figure A.23 show the individual class, and the combined classes, respectively.

Table B.6 summarizes the generalization error result from the four classifiers. Again, the comparison is consistent with our finding in the simulation studies. That is, the new classifier has the best performance with classification error 0.14, followed by performances of LDA, SVM, and logistic regression has the worst classification error with 0.874.

We notice that the classes configuration in the wild-type flies example, especially
for class 2, 4, 5 and 5, it is more difficult to discriminate between these classes. This could explains the decreasing classification performance for this data set. Also, note that the performances of the LDA and our classifier are quite similar for this data, this is probably due to large sample sizes for each class.

5.3. Smoking data

In this section, we apply our classifier to the naive Bayes classifier, logistic regression and decision tree for the smoking data set.

To understand the risk factors for chronic obstructive lung diseases, residents of both rural and urban areas of Vlagtwedde in Netherland were followed over 21 years (van der Lende et al., 1981). The sampled residents, initially aged 15-44, participated in approximately every 3 years for up to 21 years. At each survey, the relevant demographic and clinical information was collected by questionnaire. In particular, their pulmonary function was determined by a measure of forced expiratory volume (FEV1).

The data set in our analysis consists of a sub-sample of 30 residents aged 36 or older at the initial of the study, and whose smoking status did not change over the 19 years of follow-up. The FEV1 is recorded on whether it is high (> 3) or not for every subject at 0, 3, 6, 9 and 12 years of follow-up. Each participant was either a current or former smoker, where current smoking is defined as smoking at least one cigarette per day. We are interested in developing a classifier to predict a resident is a former or current smoker.

Figure A.24 shows the logit probability of high FEV1 versus low FEV1. Table B.7 displays the performance of the four classifiers. It appears that the new classifier has the best performance with classification error 0.433, followed by the naive Bayes classifier and logistic regression, and the decision tree has the worst performance which classify every
subject wrong. The comparison between the naive Bayes classifier and the new classifier is consistent with our finding in the simulation study.

5.4. PTSD data

The post-traumatic stress disorder (PTSD) data set (Keane et al., 1996; Allison, 2001) contains the interview result of the survivors in the residential fires. These survivors were interviewed at 3, 6, and 12 months after the fire. The response variable, whether the survivor had symptoms of post-traumatic stress (coded as 1), or not (coded as 0), is binary variable. In addition, various demographic variables were recorded. In particular, we are interested in the following two variables, CONTROL, A scale of a person’s perceived control over several areas of life, and PROBLEMS, the total number of problems reported in several areas of life. Specifically, we will group the survivors according to whether their CONTROL is strong, and whether their total number of reported problems is small. These two variables are binary indicator which is 1 if it is above average and 0 otherwise. There are total 4 classes: strong CONTROL, more PROBLEMS; strong CONTROL, less PROBLEMS; less CONTROL, more PROBLEMS; less CONTROL, less PROBLEMS. To illustrate the performance of our method, we randomly take 75 survivors, 20 from class 1, 25 from class 2, 15 from class 3 and 15 from class 4.

We compare the new classifier to the three classifiers for this data. Figure A.25 shows the logit of probabilities from the 4 classes. Table B.8 summarizes the performance of four classifiers. The comparison indicates that four classifier have very similar performances, with our classifier has the lowest classification error, followed by logistic regression and the naive Bayes classifier follow, and decision tree has the worst performance.

One feature of this problem, which might not favor our method, is that each subject
only has 3 repeated measurements. When cluster size is small, our method might not have much advantage compared to other classifiers.

For these two binary data sets, our method still achieves the best on classification error. This is an indication that our method is quite robust for binary data which clearly does not satisfy the elliptically symmetric condition, the required assumption in Theorem 1.
6. SUMMARY AND DISCUSSION

This paper introduces a new classifier for repeated measures based on the quadratic inference functions. By taking advantage of modeling each class and utilizing more information from the data, our classifier gains more discriminating power. Simulation and real data examples show that the new method outperforms commonly used classifiers such as SVM, logistic regression, LDA, the naive Bayes classifier and decision trees.

The quadratic distance used in the new classifier asymptotically has a χ² distribution under normality. This enables one to obtain inference function and provide p-value interpretation to assess the goodness-of-fit for the new subject. While the new classifier is asymptotically equivalent to LDA under normality assumption, the new method has the benefit of not estimating sample covariance matrix which is required for LDA. This provides a more reliable and powerful classification tool for small and moderate sample sizes relative to the number of repeated measures.

The new method can be easily adapted to more general settings such as having more covariates other than time, with different time points, etc. Although the new classifier is introduced in the context of continuous temporal responses, it can be applied to discrete responses as well. The quadratic distance provides a sensible tool for discriminating classes when the likelihood function is not available.

In classification problem, small samples with large numbers of classes are quite common in practice. This imposes a major challenge for most existing classifiers. The data containing small samples usually do not have sufficient information for better and more accurate estimation of parameters corresponding to a specified model. This implies that we should focus on developing a parsimonious statistical model for classification problem.
The new approach requires minimal assumptions for the model with repeated measurements. In theory, the new classifier is asymptotically optimal under the elliptically symmetric distribution condition. However, in our simulation and data examples for binary responses, the new classifier is fairly robust and performs better than common popular classifiers.

In the design of classifiers, the generative approaches taking into account of association between the response and covariates usually achieve higher efficiency than discriminative approaches, and thus yield more powerful classifiers. However, the former approaches require stronger model assumptions, and hence is more likely to produce bias results when the model assumptions are violated. In contrast, discriminative classifiers have less distributional assumptions and tend to be more robust. Our classifier is derived from the generative approach perspective by incorporating the modeling information, but is based on the minimal model assumptions in the repeated measurements context, therefore it is a more robust classifier compared to other generative classifiers.


27. Han, J., Kamber, M., 2006. *Data Mining: Concepts and Techniques* Jiawei Han, Morgan Kaufmann Publishers.


APPENDICES
A APPENDIX Figures

FIGURE A.1: Examples of hand-written digits taken from US zip codes (Bishop, 2006).

FIGURE A.2: An classification example. The straight black line represents a simple model and the red wiggly line represents a complex model.
FIGURE A.3: A support vector machine (SVM) example. The two classes can be separated by more than one lines, e.g., $a_1x + b_1$, $a_2x + b_2$ and $a_3x + b_3$ as shown in this figure. The goal of SVM is to find the separating line which maximizes the geometric margin.
FIGURE A.4: A decision tree example. The decision to play or not play tennis depends on both the weather and Beaver football match. The ellipses are decision nodes and the rectangles are leaf nodes.

FIGURE A.5: The simulation with 2 hypothetical classes and there is a time shift between them. The solide curves are in class 1, and the dashed curves are in class 2.
FIGURE A.7: Normal approximation to the error rate with respect to time shift and number of repeated measurements.
FIGURE A.8: Upper bound of the error rate with respect to time shift and number of repeated measurements.
FIGURE A.9: The 2 hypothetical classes when the mean function of class 2 is 2.5 times as large as that of class 1, and the dashed curves are in class 2. The solid curves are in class 1.
FIGURE A.10: Performance comparison of the new classifier, logistic regression, SVM and LDA. Q - the new classifier, * - logistic regression, v - support vector machine, l - LDA
FIGURE A.11: Normal approximation to the error rate with respect to the amplification factor and number of repeated measurements.
FIGURE A.12: Upper bound of the error rate with respect to the amplification factor and number of repeated measurements.
FIGURE A.13: Performance comparison of the new classifier, logistic regression, decision tree and the naive Bayes classifier. Q - the new classifier, * - Naive Bayes classifier, △ - logistic regression, ■ - decision tree
FIGURE A.14: Normal approximation to the error rate with respect to the coefficients in the linear predictor and number of repeated measurements.

FIGURE A.15: Upper bound of the error rate with respect to the coefficients in the linear predictor and number of repeated measurements.
FIGURE A.16: Logit of probabilities over time for the two classes.

FIGURE A.17: Performance comparison of the new classifier, logistic regression, decision tree and the naive Bayes classifier. Q - the new classifier, * - naive Bayes classifier, △ - logistic regression, ■ - decision tree
FIGURE A.18: Normal approximation to the error rate with respect to the coefficients in the linear predictor and number of repeated measurements.

FIGURE A.19: Upper bound of the error rate with respect to the coefficients in the linear predictor and number of repeated measurements.
FIGURE A.20: The 5 classes from the yeast cell cycle data. The thick lines are the fitted means for each class.
FIGURE A.21: The combined 5 classes from the yeast cell cycle data. The thick lines are the fitted means for each class.
FIGURE A.22: The 6 classes from the fruit fly gene expression data. The thick lines are the fitted means for each class.
FIGURE A.23: The combined 6 classes from the fruit fly gene expression data. The thick lines are the fitted means for each class.
FIGURE A.24: The two classes from the smoking data. The logit of high FEV1 probability for each class is displayed.
FIGURE A.25: The 4 classes from the PTSD data. The logit of probability for each class is displayed.
### TABLE B.1: Simulated generalization error, its standard error and 95% confidence interval for SVM, Logistic regression, LDA, and the new classifier.

<table>
<thead>
<tr>
<th>Classification</th>
<th>Standard error</th>
<th>Confidence interval (95%)</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>0.205</td>
<td>(0.169, 0.241)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>0.104</td>
<td>(0.077, 0.131)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>LDA</td>
<td>0.014</td>
<td>(0.003, 0.025)</td>
<td>0.004</td>
</tr>
<tr>
<td>New classifier</td>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Normal approximation</td>
<td>0.131</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Upper bound</td>
<td>0.443</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

### TABLE B.2: Simulated generalization error, its standard error and 95% confidence interval for SVM, Logistic regression, LDA, and the new classifier.

<table>
<thead>
<tr>
<th>Classification</th>
<th>Standard error</th>
<th>Confidence interval (95%)</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>0.445</td>
<td>(0.400, 0.489)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>0.200</td>
<td>(0.164, 0.235)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>LDA</td>
<td>0.011</td>
<td>(0.002, 0.020)</td>
<td>0.009</td>
</tr>
<tr>
<td>New classifier</td>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Normal approximation</td>
<td>0.0004</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Upper bound</td>
<td>0.082</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
TABLE B.3: Simulated generalization error, its standard error and 95% confidence interval for Decision tree, Logistic regression, the naive Bayes classifier, and the new classifier.

<table>
<thead>
<tr>
<th>Classification</th>
<th>standard error</th>
<th>confidence interval (95%)</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision Tree</td>
<td>0.325</td>
<td>0.021</td>
<td>(0.283, 0.367)</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>0.279</td>
<td>0.020</td>
<td>(0.239, 0.319)</td>
</tr>
<tr>
<td>Naive Bayes</td>
<td>0.263</td>
<td>0.020</td>
<td>(0.223, 0.302)</td>
</tr>
<tr>
<td>New classifier</td>
<td>0.163</td>
<td>0.016</td>
<td>(0.130, 0.195)</td>
</tr>
<tr>
<td>Normal approximation</td>
<td>0.042</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Upper bound</td>
<td>0.182</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

TABLE B.4: Simulated generalization error, its standard error and 95% confidence interval for Decision tree, Logistic regression, the naive Bayes classifier, and the new classifier.

<table>
<thead>
<tr>
<th>Classification</th>
<th>standard error</th>
<th>confidence interval (95%)</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision Tree</td>
<td>0.388</td>
<td>0.022</td>
<td>(0.344, 0.432)</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>0.466</td>
<td>0.022</td>
<td>(0.239, 0.319)</td>
</tr>
<tr>
<td>Naive Bayes</td>
<td>0.323</td>
<td>0.021</td>
<td>(0.281, 0.365)</td>
</tr>
<tr>
<td>New classifier</td>
<td>0.299</td>
<td>0.020</td>
<td>(0.258, 0.340)</td>
</tr>
<tr>
<td>Normal approximation</td>
<td>0.130</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Upper bound</td>
<td>0.441</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

TABLE B.5: Performance of SVM, Logistic regression, LDA, and the new classifier on the yeast cell data.

<table>
<thead>
<tr>
<th>number of errors</th>
<th>Classification error</th>
<th>standard error</th>
<th>confidence interval (95%)</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>30</td>
<td>0.400</td>
<td>0.057</td>
<td>(0.287, 0.513)</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>15</td>
<td>0.200</td>
<td>0.046</td>
<td>(0.108, 0.292)</td>
</tr>
<tr>
<td>LDA</td>
<td>7</td>
<td>0.093</td>
<td>0.033</td>
<td>(0.026, 0.160)</td>
</tr>
<tr>
<td>New classifier</td>
<td>4</td>
<td>0.053</td>
<td>0.026</td>
<td>(0.001, 0.105)</td>
</tr>
</tbody>
</table>
TABLE B.6: Performance of SVM, Logistic regression, LDA, and the new classifier on the fruit fly data

<table>
<thead>
<tr>
<th></th>
<th>number of errors</th>
<th>Classification error</th>
<th>standard error</th>
<th>confidence interval (95%)</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>442</td>
<td>0.395</td>
<td>0.015</td>
<td>(0.365, 0.425)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>979</td>
<td>0.874</td>
<td>0.010</td>
<td>(0.854, 0.894)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>LDA</td>
<td>168</td>
<td>0.150</td>
<td>0.011</td>
<td>(0.128, 0.172)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>New classifier</td>
<td>157</td>
<td>0.140</td>
<td>0.010</td>
<td>(0.120, 0.160)</td>
<td>&lt;0.001</td>
</tr>
</tbody>
</table>

TABLE B.7: Performance of the new classifier, the naive Bayes classifier, decision tree and logistic regression on the smoking data

<table>
<thead>
<tr>
<th></th>
<th>number of errors</th>
<th>Classification error</th>
<th>standard error</th>
<th>confidence interval (95%)</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naive Bayes</td>
<td>16</td>
<td>0.533</td>
<td>0.091</td>
<td>(0.351, 0.716)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>18</td>
<td>0.600</td>
<td>0.089</td>
<td>(0.421, 0.779)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>Decision tree</td>
<td>21</td>
<td>0.700</td>
<td>0.084</td>
<td>(0.533, 0.867)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>New classifier</td>
<td>13</td>
<td>0.433</td>
<td>0.090</td>
<td>(0.252, 0.614)</td>
<td>&lt;0.001</td>
</tr>
</tbody>
</table>

TABLE B.8: Performance of the new classifier, the naive Bayes classifier, decision tree and logistic regression on the PTSD data

<table>
<thead>
<tr>
<th></th>
<th>number of errors</th>
<th>Classification error</th>
<th>standard error</th>
<th>confidence interval (95%)</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naive Bayes</td>
<td>50</td>
<td>0.667</td>
<td>0.0211</td>
<td>(0.625, 0.709)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>50</td>
<td>0.667</td>
<td>0.0211</td>
<td>(0.625, 0.709)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>Decision tree</td>
<td>49</td>
<td>0.653</td>
<td>0.0213</td>
<td>(0.611, 0.700)</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>New classifier</td>
<td>48</td>
<td>0.640</td>
<td>0.0214</td>
<td>(0.600, 0.683)</td>
<td>&lt;0.001</td>
</tr>
</tbody>
</table>
C APPENDIX Proof of Theorem 1

We prove the theorem for two classes, and the proof can be easily generalized to multiple class settings.

Suppose \( f_1 \) is the probability density functions corresponding to the class 1 with mean \( \mu_1 \) and variance-covariance matrix \( V_1 \).

By definition of \( QD_1(y) \), we have,

\[
QD_1(y) = g_1' W_1^{-1} g_1 = [\tilde{T}_1'(y - \hat{\mu}_1)]' W_1^{-1} [\tilde{T}_1'(y - \hat{\mu}_1)] \tag{C.1}
\]

\[
p \left[ T_1'(y - \mu_1)]' (T_1'V_1T_1)^{-1} [T_1'(y - \mu_1)] \right.
\]

\[
\left. = [T_1'(y - \mu_1)]' \Sigma_1^{-1} [T_1'(y - \mu_1)] \text{ as training sample size, } n \to \infty \right.
\]

where \( \Sigma_1^{-1} = (T_1'V_1T_1)^{-1} \)

Similarly,\[
QD_2(y) \tag{C.2}
\]

\[
p \left[ T_2'(y - \mu_2)]' \Sigma_2^{-1} [T_2'(y - \mu_2)] \right. \text{ where } \Sigma_2^{-1} = (T_2'V_2T_2)^{-1}
\]

Under the inverse location regression model (3.7),

\[
T_1'y_1 \overset{d}{=} \mu_1 + \Sigma_1^{\frac{1}{2}} u
\]

\[
\Rightarrow f(T_1'y_1) = |\Sigma_1|^{-\frac{1}{2}} f_0[\Sigma_1^{-\frac{1}{2}}(T_1'(y_1 - \mu_1))] \Sigma_1^{-1}(T_1'(y_1 - \mu_1))]
\]
In the same way, for class 2, we have,

\[ f(T'_2y_2) = |\Sigma_2|^{-\frac{1}{2}} f_0([T'_2(y_2 - \mu_2)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))] \]

Now the optimal classification boundary is determined by,

\[
    r(y) = \begin{cases} 
        1 & \frac{|\Sigma_1|^{-\frac{1}{2}} f_0([T'_1(y_1 - \mu_1)] \Sigma_1^{-1}(T'_1(y_1 - \mu_1)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))]}{\Sigma_2^{-\frac{1}{2}} f_0([T'_2(y_2 - \mu_2)] < 1} \\
        2 & \frac{|\Sigma_1|^{-\frac{1}{2}} f_0([T'_1(y_1 - \mu_1)] \Sigma_1^{-1}(T'_1(y_1 - \mu_1)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))]}{\Sigma_2^{-\frac{1}{2}} f_0([T'_2(y_2 - \mu_2)] < 1}
    \end{cases} \tag{C.3}
\]

Under the inverse location model, \(|\Sigma_i| = |\Sigma_j|, i = 1, \ldots, C.|, and hence the classification rule (C.3) is equivalent to,

\[
    r(y) = \begin{cases} 
        1 & \frac{f_0([T'_1(y_1 - \mu_1)] \Sigma_1^{-1}(T'_1(y_1 - \mu_1)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))]}{f_0([T'_2(y_2 - \mu_2)] < 1} \\
        2 & \frac{f_0([T'_1(y_1 - \mu_1)] \Sigma_1^{-1}(T'_1(y_1 - \mu_1)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))]}{f_0([T'_2(y_2 - \mu_2)] < 1}
    \end{cases} \tag{C.4}
\]

and further,

\[
    r(y) = \begin{cases} 
        1 & f_0([T'_1(y_1 - \mu_1)] \Sigma_1^{-1}(T'_1(y_1 - \mu_1)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))] < f_0([T'_2(y_2 - \mu_2)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))] \\
        2 & f_0([T'_1(y_1 - \mu_1)] \Sigma_1^{-1}(T'_1(y_1 - \mu_1)] < f_0([T'_2(y_2 - \mu_2)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))] \tag{C.5}
    \end{cases}
\]

Using the monotonicity of \(f_0\), we have,

\[
    r(y) = \begin{cases} 
        1 & (T'_1(y_1 - \mu_1)] \Sigma_1^{-1}(T'_1(y_1 - \mu_1)] < (T'_2(y_2 - \mu_2)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))] \\
        2 & (T'_1(y_1 - \mu_1)] \Sigma_1^{-1}(T'_1(y_1 - \mu_1)] < (T'_2(y_2 - \mu_2)] \Sigma_2^{-1}(T'_2(y_2 - \mu_2))] \tag{C.6}
    \end{cases}
\]

Combining (C.1) and (C.2) with (C.6), we complete the proof.
D APPENDIX  Proof of Theorem 2

Suppose $y_1,\cdots,y_n,y_{n+1}$ are independent random vectors, and $X$ is a $p \times 1$ vector of covariates with

$$E(y_i) = X\beta \text{ for } i = 1, \cdots, n$$

$$E(y_{n+1}) = X\beta^*$$

$$Var(y_i) = V \text{ for } i = 1, \cdots, n+1.$$ 

Write $V = D^{\frac{1}{2}}RD^{\frac{1}{2}}$ where $D$ is a diagonal matrix of variance and $R$ is a correlation matrix. We assume $D$ is known (in practice, empirical variances can be used). Let

$$T = [D^{-\frac{1}{2}}M_1D^{-\frac{1}{2}}X \cdots D^{-\frac{1}{2}}M_mD^{-\frac{1}{2}}X]$$

where $M_1, \cdots, M_m$ are known symmetric matrices. A working model for $R$ is that $R^{-1}$ is a linear combination of $M_1, \cdots, M_m$. The following holds even if the working model is incorrect.

Define

$$\bar{y}_n = \frac{1}{n} \sum_{i=1}^{n} y_i$$

$$g_n^\dagger = T'(y_{n+1} - \bar{y}_n)$$

$$C_n^\dagger = \frac{1}{n} \sum_{i=1}^{n} T'(y_i - \bar{y}_n)(y_i - \bar{y}_n)'T$$

$$Q_n^\dagger = (g_n^\dagger)'(C_n^\dagger)^{-1}(g_n^\dagger).$$
Let \( \delta = T'X(\beta^* - \beta) \), \( W = T'VT \), so that

\[
E(g_n^\dagger) = \delta \\
Var(g_n^\dagger) = \frac{n+1}{n}W \\
E(C_n^\dagger) = \frac{n-1}{n}W.
\]

If we assume that the \( y_i \) are approximately normal, then \( g_n^\dagger \) is approximately \( N(\delta, W) \) and \( (g_n^\dagger)'W^{-1}(g_n^\dagger) \) is approximately \( \chi^2(mp, \delta'W^{-1}\delta) \). By the WLLN, \( C_n^\dagger \overset{p}{\to} W \) as \( n \to 0 \), and so \( Q_n^\dagger \) is approximately \( \chi^2(mp, \delta'W^{-1}\delta) \).
E  APPENDIX  Classification error of the new classifier

Let us first assume the training data are generated from 2 populations, \( P_1 \) with mean \( \mu_1 \) and variance covariance \( \Sigma_1 \), and \( P_2 \) with mean \( \mu_2 \) and variance covariance \( \Sigma_2 \). Suppose a new subject, \( y \), is from \( P_1 \). Now we calculate the probability of classifying \( y \) to \( P_2 \). To make it easy to follow, let us assume the training set to be large, and introduce the following notations.

Let \( A_1 = T_1 W_1^{-1} T_1' \), \( A_2 = T_2 W_2^{-1} T_2' \), \( Q = y'(T_1 W_1^{-1} T_1' - T_2 W_2^{-1} T_2')y - 2(\mu_1' T_1 W_1^{-1} T_1' - \mu_2' T_2 W_2^{-1} T_2')y = y'(A_1 - A_2)y - 2(\mu_1' A_1 - \mu_2' A_2)y \), and \( A = A_1 - A_2 \).

Then,

Classify \( y \) into \( P_2 \) \( \iff \) \( QD_1(y) > QD_2(y) \)

\[ \iff (T_1'y - T_1'\mu_1)'W_1^{-1}(T_1'y - T_1'\mu_1) > (T_2'y - T_2'\mu_2)'W_2^{-1}(T_2'y - T_2'\mu_2) \]

\[ \iff (y - \mu_1)'T_1 W_1^{-1} T_1'(y - \mu_1) > (y - \mu_2)'T_2 W_2^{-1} T_2'(y - \mu_2) \]

\[ \iff y'T_1 W_1^{-1} T_1'y - 2y'T_1 W_1^{-1} T_1'\mu_1 + \mu_1' T_1 W_1^{-1} T_1' \mu_1 \]

\[ > y'T_2 W_2^{-1} T_2'y - 2y'T_2 W_2^{-1} T_2' \mu_2 + \mu_2' T_2 W_2^{-1} T_2' \mu_2 \]

\[ \iff y'(A_1 - A_2)y - 2(\mu_1' A_1 - \mu_2' A_2) > \mu_2' A_2 \mu_2 - \mu_1' A_1 \mu_1 \]

Therefore,

\[ P\{\text{Classify } y \text{ into } P_2 | y \text{ is from } P_1\} \]

\[ = P\{y'(A_1 - A_2)y - 2(\mu_1' A_1 - \mu_2' A_2)y > \mu_2' A_2 \mu_2 - \mu_1' A_1 \mu_1\} \]

In applications, it is possible to simulate the classification error, but the analytical prop-
roperties of the error rate remain to be unraveled. Next we investigate some of its statistical properties.

We have,

\[ \begin{align*}
E(y'(A_1 - A_2)y) \\
= & \text{tr}((A_1 - A_2)\Sigma_1) + \mu'_1(A_1 - A_2)\mu_1 \\
= & \text{tr}((A_1 - A_2)\Sigma_1) + \mu'_1(A_1 - A_2)\mu_1 \\
E(2(\mu'_1 A_1 - \mu'_2 A_2)y) \\
= & 2(\mu'_1 A_1 - \mu'_2 A_2)\mu_1 \\
= & 2\mu'_1 A_1\mu_1 - 2\mu'_2 A_2\mu_1 \\
E(Q) \\
= & E(y'(A_1 - A_2)y) - E(2(\mu'_1 A_1 - \mu'_2 A_2)y) \\
= & \text{tr}((A_1 - A_2)\Sigma_1) + \mu'_1(A_1 - A_2)\mu_1 - 2\mu'_1 A_1\mu_1 + 2\mu'_2 A_2\mu_1
\end{align*} \]
\begin{align*}
\text{Var}(y'(A_1 - A_2)y) &= 2\text{tr}((A_1 - A_2)\Sigma_1 (A_1 - A_2)\Sigma_1) + 4\mu_1'(A_1 - A_2)\Sigma_1 (A_1 - A_2)\mu_1 \\
\text{Var}(2(\mu_1' A_1 - \mu_2' A_2)y) &= 4(\mu_1' A_1 - \mu_2' A_2)\Sigma_1 (\mu_1' A_1 - \mu_2' A_2)' \\
&= 4(\mu_1' A_1 - \mu_2' A_2)\Sigma_1 (\mu_1' A_1 - \mu_2' A_2)'
\end{align*}

\begin{align*}
\text{Cov}(y'(A_1 - A_2)y, 2(\mu_1' A_1 - \mu_2' A_2)y) &= \text{Cov}(y'(A_1 - A_2)y, 2(\mu_1' A_1 - \mu_2' A_2)y) \\
&= E[y'(A_1 - A_2)y*2(\mu_1' A_1 - \mu_2' A_2)y] - E[y'(A_1 - A_2)y]E[2(\mu_1' A_1 - \mu_2' A_2)y] \\
&= E[(\mu_1 + \Sigma^3_1 z)'A(\mu_1 + \Sigma^3_1 z) * 2(\mu_1' A_1 - \mu_2' A_2)(\mu_1 + \Sigma^3_1 z)] \\
&\quad - E[y'(A_1 - A_2)y]E[2(\mu_1' A_1 - \mu_2' A_2)y] \\
&= E[2(\mu_1'(A_1 - A_2)\mu_1 + 2\mu_1'(A_1 - A_2)\Sigma^3_1 z + z'\Sigma^3_1 (A_1 - A_2)\Sigma^3_1 z)((\mu_1' A_1 - \mu_2' A_2)\mu_1 \\
&\quad + (\mu_1' A_1 - \mu_2' A_2)\Sigma^3_1 z)] - E[y'(A_1 - A_2)y]E[2(\mu_1' A_1 - \mu_2' A_2)y] \\
&= E(2\mu_1'(A_1 - A_2)\mu_1(\mu_1' A_1 - \mu_2' A_2)\mu_1 + 4\mu_1'(A_1 - A_2)\Sigma^3_1 z(\mu_1' A_1 - \mu_2' A_2)\mu_1 \\
&\quad + 2z'\Sigma^3_1 (A_1 - A_2)\Sigma^3_1 z(\mu_1' A_1 - \mu_2' A_2)\mu_1 + 2\mu_1'(A_1 - A_2)\mu_1(\mu_1' A_1 - \mu_2' A_2)\Sigma^3_1 z \\
&\quad + 4\mu_1'(A_1 - A_2)\Sigma^3_1 z(\mu_1' A_1 - \mu_2' A_2)\Sigma^3_1 z + 2z'\Sigma^3_1 (A_1 - A_2)\Sigma^3_1 z(\mu_1' A_1 - \mu_2' A_2)\Sigma^3_1 z) \\
&\quad - E[y'(A_1 - A_2)y]E[2(\mu_1' A_1 - \mu_2' A_2)y] \\
&= 2\mu_1'(A_1 - A_2)\mu_1(\mu_1' A_1 - \mu_2' A_2)\mu_1 + 2E(z'\Sigma^3_1 (A_1 - A_2)\Sigma^3_1 z(\mu_1' A_1 - \mu_2' A_2)\mu_1) \\
&\quad + 4E(\mu_1'(A_1 - A_2)\Sigma^3_1 z(\mu_1' A_1 - \mu_2' A_2)\Sigma^3_1 z) \\
&\quad - E[y'(A_1 - A_2)y]E[2(\mu_1' A_1 - \mu_2' A_2)y]
\end{align*}
\[= 2\mu' (A_1 - A_2) \mu_1 (\mu'_1 A_1 - \mu'_2 A_2) \mu_1 + 2E(z' \Sigma_{1}^{\frac{1}{2}} (A_1 - A_2) \Sigma_{1}^{\frac{1}{2}} z (\mu'_1 A_1 - \mu'_2 A_2) \mu_1)\]
\[+ 4E(z' \Sigma_{1}^{\frac{1}{2}} (A_1 - A_2) \mu_1 (\mu'_1 A_1 - \mu'_2 A_2) \Sigma_{1}^{\frac{1}{2}} z)\]
\[- E[y'(A_1 - A_2)y]E[2(\mu'_1 A_1 - \mu'_2 A_2)y]\]
\[= 2\mu' (A_1 - A_2) \mu_1 (\mu'_1 A_1 - \mu'_2 A_2) \mu_1 + 2tr(\Sigma_{1}^{\frac{1}{2}} (A_1 - A_2) \Sigma_{1}^{\frac{1}{2}} (\mu'_1 A_1 - \mu'_2 A_2) \mu_1)\]
\[+ 4tr(\Sigma_{1}^{\frac{1}{2}} (A_1 - A_2) \mu_1 (\mu'_1 A_1 - \mu'_2 A_2) \Sigma_{1}^{\frac{1}{2}})\]
\[- E[y'(A_1 - A_2)y]E[2(\mu'_1 A_1 - \mu'_2 A_2)y]\]
\[= 2\mu' (A_1 - A_2) \mu_1 (\mu'_1 A_1 - \mu'_2 A_2) \mu_1 + 2tr(\Sigma_{1}^{\frac{1}{2}} (A_1 - A_2) \Sigma_{1}^{\frac{1}{2}} (\mu'_1 A_1 - \mu'_2 A_2) \mu_1)\]
\[+ 4tr(\Sigma_{1}^{\frac{1}{2}} (A_1 - A_2) \mu_1 (\mu'_1 A_1 - \mu'_2 A_2) \Sigma_{1}^{\frac{1}{2}})\]
\[- [tr((A_1 - A_2) \Sigma_1) + \mu'_1 (A_1 - A_2) \mu_1]2(\mu'_1 A_1 - \mu'_2 A_2) \mu_1\]
\[= 2\mu' (A_1 - A_2) \mu_1 (\mu'_1 A_1 - \mu'_2 A_2) \mu_1 + 2tr((A_1 - A_2) \Sigma_1 (\mu'_1 A_1 - \mu'_2 A_2) \mu_1)\]
\[+ 4tr((A_1 - A_2) \mu_1 (\mu'_1 A_1 - \mu'_2 A_2) \Sigma_1)\]
\[- [tr((A_1 - A_2) \Sigma_1) + \mu'_1 (A_1 - A_2) \mu_1]2(\mu'_1 A_1 - \mu'_2 A_2) \mu_1\]
Hence,

$$Var(Q) = Var(y'(A_1 - A_2)y) + Var(2(\mu'_1 A_1 - \mu'_2 A_2)y)$$

$$- 2Cov(y'(A_1 - A_2)y, 2(\mu'_1 A_1 - \mu'_2 A_2)y)$$

$$= 2tr((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4\mu'_1(A_1 - A_2)\Sigma_1(A_1 - A_2)\mu_1$$

$$+ 4(\mu'_1 A_1 - \mu'_2 A_2)\Sigma_1(\mu'_1 A_1 - \mu'_2 A_2)' - 2[2\mu'_1(A_1 - A_2)\mu_1(\mu'_1 A_1 - \mu'_2 A_2)\mu_1]$$

$$+ 2tr((A_1 - A_2)\Sigma_1(\mu'_1 A_1 - \mu'_2 A_2)\mu_1) + 4tr((A_1 - A_2)\mu_1(\mu'_1 A_1 - \mu'_2 A_2)\Sigma_1)$$

$$- [tr((A_1 - A_2)\Sigma_1) + \mu'_1(A_1 - A_2)\mu_1]2(\mu'_1 A_1 - \mu'_2 A_2)\mu_1]$$

$$= 2tr((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4\mu'_1(A_1 - A_2)\Sigma_1(A_1 - A_2)\mu_1$$

$$+ 4(\mu'_1 A_1 - \mu'_2 A_2)\Sigma_1(\mu'_1 A_1 - \mu'_2 A_2)' - 8tr((A_1 - A_2)\mu_1(\mu'_1 A_1 - \mu'_2 A_2)\Sigma_1)$$

$$= 2tr((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4tr(\mu'_1(A_1 - A_2)\Sigma^\frac{1}{2}_{\Sigma_1}^\frac{1}{2}_{\Sigma_1} (A_1 - A_2)\mu_1)$$

$$+ 4tr((\mu'_1 A_1 - \mu'_2 A_2)\Sigma^\frac{1}{2}_{\Sigma_1}^\frac{1}{2}_{\Sigma_1} (\mu'_1 A_1 - \mu'_2 A_2)')$$

$$- 8tr(\Sigma^\frac{1}{2}_{\Sigma_1} (A_1 - A_2)\mu_1(\mu'_1 A_1 - \mu'_2 A_2)\Sigma^\frac{1}{2}_{\Sigma_1})$$

$$= 2tr((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4tr(\mu'_1(A_1 - A_2)\Sigma^\frac{1}{2}_{\Sigma_1}^\frac{1}{2}_{\Sigma_1} (A_1 - A_2)\mu_1)$$

$$+ 4tr((\mu'_1 A_1 - \mu'_2 A_2)\Sigma^\frac{1}{2}_{\Sigma_1}^\frac{1}{2}_{\Sigma_1} (\mu'_1 A_1 - \mu'_2 A_2)')$$

$$- 8tr(((\mu'_1 A_1 - \mu'_2 A_2)\Sigma^\frac{1}{2}_{\Sigma_1})(\mu'_1 A_1 - \mu'_2 A_2)\Sigma^\frac{1}{2}_{\Sigma_1})$$

$$= 2tr((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4tr(\mu'_1(A_1 - A_2)\Sigma^\frac{1}{2}_{\Sigma_1}^\frac{1}{2}_{\Sigma_1} (A_1 - A_2)\mu_1)$$

$$+ 4tr((\mu'_1 A_1 - \mu'_2 A_2)\Sigma^\frac{1}{2}_{\Sigma_1}^\frac{1}{2}_{\Sigma_1} (\mu'_1 A_1 - \mu'_2 A_2)')$$

$$- 8tr(((\mu'_1 A_1 - \mu'_2 A_2)\Sigma^\frac{1}{2}_{\Sigma_1})(\mu'_1 A_1 - \mu'_2 A_2)\Sigma^\frac{1}{2}_{\Sigma_1})$$
Therefore, the normal approximation to the error rate is given by,

\[ P\{\text{Classify } y \text{ into } P_2 | y \text{ is from } P_1 \} \]

\[= P\{ y'(A_1 - A_2)y - 2(\mu'_1A_1 - \mu'_2A_2)y > \mu'_2A_2\mu_2 - \mu'_1A_1\mu_1 \} \]

\[\approx 1 - \Phi \left( \frac{\mu'_2A_2\mu_2 - \mu'_1A_1\mu_1 - E(Q)}{\sqrt{\text{Var}(Q)}} \right) \]

\[= 1 - \Phi \left( \frac{\mu'_2A_2\mu_2 - \mu'_1A_1\mu_1 - \text{tr}((A_1 - A_2)\Sigma_1) - \mu'_1(A_1 - A_2)\mu_1 + 2\mu'_1A_1\mu_1 - 2\mu'_2A_2\mu_1}{\sqrt{2\text{tr}((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4(\mu'_2 - \mu'_1)A_2\Sigma_1A_2(\mu_2 - \mu_1)}} \right) \]

\[= 1 - \Phi \left( \frac{\mu'_2A_2\mu_2 - \mu'_1A_1\mu_1 - \text{tr}((A_1 - A_2)\Sigma_1) + \mu'_1A_2\mu_1 - 2\mu'_2A_2\mu_1}{\sqrt{2\text{tr}((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4(\mu'_2 - \mu'_1)A_2\Sigma_1A_2(\mu_2 - \mu_1)}} \right) \]

\[= 1 - \Phi \left( \frac{(\mu_2 - \mu_1)'A_2(\mu_2 - \mu_1) - \text{tr}((A_1 - A_2)\Sigma_1)}{\sqrt{2\text{tr}((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4(\mu'_2 - \mu'_1)A_2\Sigma_1A_2(\mu_2 - \mu_1)}} \right) \]
Similarly, we get the following normal approximation,

\[ P\{\text{Classify } y \text{ into } P_1 | y \text{ is from } P_2 \} \approx 1 - \Phi \left( \frac{(\mu_1 - \mu_2)'A_1(\mu_1 - \mu_2) - \text{tr}((A_2 - A_1)\Sigma_2)}{\sqrt{2\text{tr}((A_2 - A_1)\Sigma_2(A_2 - A_1)\Sigma_2) + 4(\mu_1' - \mu_2')A_1\Sigma_2A_1(\mu_1 - \mu_2)}} \right) \]

Assuming equal frequencies for both populations, the normal approximation to the misclassification error rate is given by,

\[ P\{\text{Classify } y \text{ into } P_1 | y \text{ is from } P_2 \text{ or classify } y \text{ into } P_2 | y \text{ is from } P_1 \} \approx 1 - \frac{1}{2}\Phi \left( \frac{(\mu_2 - \mu_1)'A_2(\mu_2 - \mu_1) - \text{tr}((A_1 - A_2)\Sigma_1)}{\sqrt{2\text{tr}((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4(\mu_1' - \mu_2')A_2\Sigma_1A_2(\mu_2 - \mu_1)}} \right) \]

\[ + \frac{1}{2}\Phi \left( \frac{(\mu_1 - \mu_2)'A_1(\mu_1 - \mu_2) - \text{tr}((A_2 - A_1)\Sigma_2)}{\sqrt{2\text{tr}((A_2 - A_1)\Sigma_2(A_2 - A_1)\Sigma_2) + 4(\mu_1' - \mu_2')A_1\Sigma_2A_1(\mu_1 - \mu_2)}} \right) \]

To get an upper bound to the error rate, recall Cantelli’s inequality (Grimmett and Stirzaker, 2001) states the following.

Let \( X \) be a random variable with expected value \( \mu \) and finite variance \( \sigma^2 \). Then for any real number \( k > 0 \),

\[ Pr(X - \mu \geq k\sigma) \leq \frac{1}{1 + k^2} \]
Applying the inequality gives us the following upper bound,

\[
P\{\text{Classify } y \text{ into } P_2 | y \text{ is from } P_1\}
= P\{Q > \mu_2' A_2 \mu_2 - \mu_1' A_1 \mu_1\}
= P\{Q - E(Q) > \mu_2' A_2 \mu_2 - \mu_1' A_1 \mu_1 - E(Q)\}
= P\{Q - E(Q) > \frac{\mu_2' A_2 \mu_2 - \mu_1' A_1 \mu_1 - E(Q)}{\sqrt{\text{Var}(Q)}} \sqrt{\text{Var}(Q)}\}
\leq \frac{1}{1 + \left(\frac{\mu_2' A_2 \mu_2 - \mu_1' A_1 \mu_1 - E(Q)}{\sqrt{\text{Var}(Q)}}\right)^2}
\]

Similarly, we get,

\[
P\{\text{Classify } y \text{ into } P_1 | y \text{ is from } P_2\}
\leq \frac{1}{1 + \left(\frac{\mu_2' A_2 \mu_2 - \mu_1' A_1 \mu_1 - \text{tr}(A_1 - A_2) \Sigma_1 - \mu_1' (A_1 - A_2) \mu_1 + 2\mu_1' A_1 \mu_1 - 2\mu_2' A_2 \mu_1}{\sqrt{\text{Var}(Q)}}\right)^2}
\]

Assuming equal frequencies for both populations, the upper bound to the error rate is given by,
\[ P\{\text{Classify } y \text{ into } P_1 | y \text{ is from } P_2 \text{ or classify } y \text{ into } P_2 | y \text{ is from } P_1 \} \leq \frac{1}{2} \frac{1}{1 + \left( \frac{(\mu_2 - \mu_1)'A_2(\mu_2 - \mu_1) - \text{tr}((A_1 - A_2)\Sigma_1)}{\sqrt{2\text{tr}((A_1 - A_2)\Sigma_1(A_1 - A_2)\Sigma_1) + 4(\mu'_2 - \mu'_1)A_2\Sigma_1 A_2(\mu_2 - \mu_1)}} \right)^2} + \frac{1}{2} \frac{1}{1 + \left( \frac{(\mu_1 - \mu_2)'A_1(\mu_1 - \mu_2) - \text{tr}((A_2 - A_1)\Sigma_2)}{\sqrt{2\text{tr}((A_2 - A_1)\Sigma_1(A_2 - A_1)\Sigma_2) + 4(\mu'_1 - \mu'_2)A_1\Sigma_1 A_2(\mu_2 - \mu_1)}} \right)^2} \]