AN ASYMPTOTICALLY OPTIMAL METHOD FOR
COMBINING SEVERAL $K$-CLASS CLASSIFIERS

A thesis submitted in partial fulfillment of the requirements
For the degree of Master of Science
in Mathematics

by

Andrew Aram Duhancioglu

December 2016
The thesis of Andrew Aram Duhancioglu is approved:

Dr. Mark F. Schilling 

Dr. Ann E. Watkins 

Dr. Majid Mojirsheibani, Chair

California State University, Northridge
Acknowledgments

Dr. Majid Mojirsheibani, I cannot express enough gratitude for constantly providing me with encouragement and motivation. Thank you for your patience throughout this entire process.

Dr. Mark Schilling, thank you for guiding me down the right path from the moment I stepped foot on campus. I will never forget the confidence you expressed in me after I failed my first actuarial exam.

Dr. Ann Watkins and Dr. Mark Schilling, I really appreciate that both of you were willing to be part of my Thesis Committee.

I would like to extend my gratitude to my other professors and colleagues of the CSUN Math Department for the role they played in my development. The relationships developed during my time at this institution will last a life time.

Mom, Dad, and Yaya, thank you for your love and support throughout my life. I am blessed for all the opportunities you provided me.

Lastly I would like to acknowledge my high school teacher and alumni of the CSUN Math Department, Mrs. Anita Joukjian. You discovered my capabilities at a very early stage and helped me realize my potential.
Table of Contents

Signature page ii
Acknowledgments iii
Abstract v
1 Probability Theory 1
2 Classification 3
  2.1 Introduction to Classification .................................. 3
  2.2 Mathematical Framework of Classification ........................ 3
  2.3 Bayes Classifier .................................................. 4
  2.4 F-Errors ......................................................... 7
3 Data Based Classifiers 9
  3.1 Introduction to Data Based Classifiers ................................ 9
  3.2 Linear Discriminant Analysis .................................... 10
  3.3 K-Nearest Neighborhood ........................................ 13
  3.4 Kernel Classifiers .............................................. 16
  3.5 Partition Rules .................................................. 17
  3.6 Support Vector Network .......................................... 18
4 The Proposed Theoretical Combined Classifier 21
  4.1 Introduction to Combining Classifiers ............................ 21
  4.2 Heuristic and Motivational Justification ........................ 22
  4.3 The Theoretical Combined Classifier ............................ 23
  4.4 Sample Based Classifier ........................................ 25
5 Numerical Studies 29
6 Concluding Remarks and Future Work 34
Bibliography 35
Appendix A Bankruptcy Data Table 38
ABSTRACT

AN ASYMPOTOTICALLY OPTIMAL METHOD FOR COMBINING SEVERAL $K$-CLASS CLASSIFIERS

By

Andrew Aram Duhancioglu

Master of Science in Mathematics

Probabilistic classification attempts to classify the unknown nature of an observation. In this context, an observation can be thought of as a collection of numerical measurements. The most appealing quality of classification is that it can be applied to numerous disciplines. For example in the field of biostatistics one might like to classify genes using DNA sequences. The focus of this thesis is to provide the best approach to combining the information of several classifiers.

We present an easy to implement procedure that yields an asymptotically optimal combined classifier composed of $M \geq 2$ different $K$-class classifiers, where $K > 2$. Here asymptotically optimal means that the misclassification error rate of the proposed combined classifier is always asymptotically less than or equal to that of each underlying classifier. The method works by finding nonparametric estimates of the class conditional expectation of a new observation, conditional on the vector of $M$ predicted values corresponding to the $M$ individual classifiers. A data-splitting approach is used to estimate the various class conditional expectations, which can be implemented easily in practice. Numerical examples, are used to evaluate the aforementioned classifier.
1 Probability Theory

This chapter provides a brief overview of the fundamentals from probability theory that are relevant for this thesis. The proofs for these results are omitted. For more information please refer to Serfling (1980), Chow and Teicher (1978), and Durrett (1991).

The following definition is an elementary concept from probability theory, which is used frequently throughout the rest of the chapters,

**Definition 1.1** The indicator function of an event $A$, denoted by $I(A)$, is equal to 1 if event $A$ occurs and 0 otherwise.

The next definition is a basic probability principle that relates the probability of an event to the expected value and indicator of the event.

**Definition 1.2** The probability and expected value of an event $A$ are denoted by $P\{A\}$ and $E\{A\}$, respectively. An expression for $P\{A\}$ is given by:

$$P\{A\} = E\{I(A)\}.$$

Consequently, the following properties of expected values are critical components of the proofs presented in the later chapter. Suppose that $X$, $Y$, and $Z$ are random variables where the expectations for all exist. If $g : \mathbb{R} \to \mathbb{R}$ and $a, b \in \mathbb{R}$, then the following properties hold:

- $E\{aX + bZ \mid Y\} = aE\{X \mid Y\} + bE\{Z \mid Y\}$.
- $E\{X\} = E\{E\{X \mid Y\}\}$.
- $E\{X \mid Y,g(Y)\} \overset{a.s.}{=} E\{X \mid Y\}$.
- $E\{Xg(Y) \mid Y\} \overset{a.s.}{=} g(Y)E\{X \mid Y\}$.

The following theorem is a classical result in probability theory that is used in the proof of Theorem 2.4.

**Theorem 1.1 [Jensen’s Inequality]** Let $f$ be a concave function on a finite or infinite interval of $\mathbb{R}$. In addition, suppose $X$ is a random variable that assumes its values in the aforementioned interval and $E\{X\} < \infty$, then

$$E\{f(X)\} \leq f(E\{X\}).$$

Furthermore, two modes of convergence for random variables are presented. The following definition is a weak form of convergence:
**Definition 1.3** Let \( X_1, \ldots, X_n \) and \( X \) be random variables. \( X_n \) **converges in probability** to \( X \), denoted by \( X_n \xrightarrow{p} X \), if \( \forall \epsilon > 0, \)

\[
\lim_{n \to \infty} P(|X_n - X| < \epsilon) = 1.
\]

The next definition presents two equivalent conditions for a stronger form of convergence:

**Definition 1.4** Let \( X_1, \ldots, X_n \) and \( X \) be random variables. \( X_n \) **converges almost surely** (or with probability 1) to \( X \), denoted by \( X_n \xrightarrow{a.s.} X \), if \( \forall \epsilon > 0 \) and \( \forall m \geq n, \)

\[
\lim_{m \to \infty} P(|X_m - X| < \epsilon) = 1 \iff P\{ \lim_{n \to \infty} X_n = X \} = 1.
\]

The definition above is not the most common technique of proving almost sure convergence. Instead the following results provide an easier method of establishing this mode of convergence.

**Theorem 1.2** [Borel-Cantelli Lemma] Let \( \{A_n \mid n \geq 1\} \) be a sequence of events. We use the following notation to define an event \( A_n \) that occurs infinitely often, \( \{A_n \text{ i.o.}\} = \limsup_{n \to \infty} A_n = \bigcap_{k=1}^{\infty} \bigcup_{n=k}^{\infty} A_n \). If \( \sum_{n=1}^{\infty} P\{A_n\} < \infty \) then \( P\{A_n \text{ i.o.}\} = 0. \)

**Corollary 1.3** Let \( \{X_n \mid n \geq 1\} \), be a sequence of random variables. If \( \forall \epsilon > 0 \)

\[
\sum_{n=1}^{\infty} P\{|X_n - X| > \epsilon\} < \infty,
\]

then \( X_n \xrightarrow{a.s.} X \).
2 Classification

2.1 Introduction to Classification

Classification is a statistical methodology that involves predicting the unknown nature of an observation. Another popular prediction method in statistics is regression. The difference between these procedures is the nature of the underlying variable that must be predicted. In regression this variable is quantitative, whereas in classification it is categorical.

An observation may be a vector of scouting metrics for a college football player, admission data for graduate school, financial records of an individual, or measurements of antihemophilic factors (AHF). The rudimentary motivation for classification is that the nature, or the classes, of these observations is unknown. With respect to the aforementioned examples, the following questions may arise:

1. Which round should a college football prospect be selected in the NFL Draft?
2. What should be the decision for a candidate’s application for graduate school?
3. Do the financial records of an individual indicate that he or she should be audited?
4. Given the AHF records of a patient, is it possible to detect hemophilia?

In classification one must construct a rule, known as a classifier, in order to determine the class of a new observation. Choosing the appropriate classifier or creating the rule for a particular situation is a difficult task. Through trial and error, professionals should try to incorporate their knowledge and experience during the formulation of the classifier. Creating and modifying a rule that can correctly classify every possible observation is impossible. Devroye et. al (1996) stated that: “We have to be prepared to live with imperfect classifiers.” Although there is a best classifier, also known as the Bayes classifier (which will be discussed later), there is no perfect classifier.

A football prospect can have poor scouting marks and end up in the Hall of Fame. It is not possible to create a rule to classify all observations perfectly, rather classification experts strive to achieve optimality for their classifiers. The phenomenon of a classifier assigning an incorrect class to an observation is called misclassification. The optimal classifier is the one that misclassifies observations the least in the long run.

2.2 Mathematical Framework of Classification

The framework of classification involves the mathematical formulation of the components described in the previous section. Classes are represented by the random variable $Y$, where $Y \in \{1, \ldots, K\}$ and $K \geq 2$. The observations are $d$-dimensional random vectors, typically denoted by $X$. Therefore, a probabilistic setting is induced, where $(X, Y) \sim F$ is a random pair.
Definition 2.1 A classifier is a function or mapping \( g \), where \( g : \mathbb{R}^d \rightarrow \{1, \ldots, K\} \).

When evaluating the performance of a classifier it is essential to consider the likelihood that it produces erroneous results. Clearly, the preferred classifiers will minimize this likelihood. The next definition formally describes this measure.

Definition 2.2 The misclassification error rate of a classifier \( g \), is defined below,

\[
L(g) = P\{g(X) \neq Y\}.
\]

The following example illustrates how one might use a classification setting to model the admissions process for graduate school.

Example 1 Suppose that the Math department is reviewing applications for prospective graduate students. The admissions officer will surely consider the undergraduate GPA and GRE score. Perhaps, he or she may also want to consider the number of upper division math courses taken as well. For this institution, there are three possible outcomes of the application: to be denied, conditionally accepted, or fully accepted.

\[
Y = \begin{cases} 
3 & \text{Accepted} \\
2 & \text{Conditionally Accepted} \\
1 & \text{Denied} 
\end{cases} \\
X = \begin{pmatrix} \text{GPA} \\
\text{Number of math classes taken} \\
\text{GRE score} \end{pmatrix}
\]

In the above example, a new observation is an application, which contains the GPA, number of math classes, and GRE score of the candidate. However, the nature of the observation or the application status is not known. The purpose of classification is to derive a rule to sort new observations, referred to as \( x \), to the defined classes.

2.3 Bayes Classifier

As was previously mentioned, the Bayes classifier is the best classifier and denoted by \( g^B \). It satisfies the following property:

\[
g^B(X) = \arg\min_{g: \mathbb{R}^d \rightarrow \{1, \ldots, K\}} P\{g(X) \neq Y\}.
\]

Therefore, \( g^B \) has the lowest classification error rate among all classifiers. However, \( g^B \) relies on the distribution of \( (X, Y) \). In practice, this distribution is usually unknown and solving the Bayes problem may be impossible.

In a perfect world, one would always have access to the distribution of \( (X, Y) \), so that the Bayes classifier could be constructed. As discussed in the previous sections, it has the lowest error rate of all classifiers. For a two class problem, where \( Y = \{0, 1\} \), the Bayes classifier is given by:
\[ g^B(x) = \begin{cases} 
 1 & \text{if } E\{Y|X = x\} > \frac{1}{2} \\
 0 & \text{otherwise.} 
\end{cases} \tag{2.1} \]

The performance of other classifiers is assessed by comparing their error probabilities relative to that of the Bayes classifier. The following result will formally justify the optimality of this classifier for the case of two classes:

**Theorem 2.1** For any classifier \( g \), where \( g: \mathbb{R}^d \to \{0, 1\} \),
\[ L(g^B) = P\{g^B(X) \neq Y\} \leq P\{g(X) \neq Y\} = L(g). \]

**PROOF OF THEOREM 2.1**
A similar proof can be found in section 2.1 of Devroye et al. (1996).
\[
L(g) - L(g^B) = P\{g^B(X) = Y\} - P\{g(X) = Y\} \\
= \sum_{y=0}^{1} P\{g^B(X) = y, Y = y\} - P\{g(X) = y, Y = y\} \\
= \sum_{y=0}^{1} E\{I(g^B(X) = y, Y = y)\} - E\{I(g(X) = y, Y = y)\} \\
= \sum_{y=0}^{1} E\{E\{I(g^B(X) = y, Y = y) | X\}\} - E\{E\{I(g(X) = y, Y = y) | X\}\} \\
= \sum_{y=0}^{1} E\{E\{I(g^B(X) = y)P\{Y = y | X\}\} - E\{I(g(X) = y)P\{Y = y | X\}\} \\
= E\left\{\max_{y=0,1} P\{Y = y | X\}\right\} - \sum_{y=0}^{1} E\{I(g(X) = y)P\{Y = y | X\}\} \geq 0 \\
= E\left\{\max_{y=0,1} P\{Y = y | X\} - (I(g(X) = 0)P\{Y = 0 | X\} + I(g(X) = 1)P\{Y = 1 | X\})\right\} \\
\geq 0 \quad \square 
\]

The next results follow immediately from Theorem 2.1. The first yields an alternate expression of the Bayes error rate and the latter provides a useful shortcut for proofs in future chapters.

**Corollary 2.2** Consider the two-class framework, where the classes are \( Y = 0 \) or \( 1 \), based
on the predictor $X$. Let $\pi(x) = P\{Y = 1 \mid X = x\}$. Then,

$$L(g^B) = E\{\min\{\pi(x), 1 - \pi(x)\}\}.$$ 

**Corollary 2.3** Consider the $K$-class framework, where the classes are $Y = \{1, \ldots, K\}$, based on the predictor $X$ and classifier $g$. Then,

$$L(g) = 1 - \sum_{y=1}^{K} E\{I(g(X) = y, Y = y)\}.$$ 

For more on these results, see Devroye et al. (1996) Chapter 2. The following example, based on an example in section 2.2 of Devroye et al. (1996), illustrates a situation where it is possible to find the exact error rate of the Bayes classifier.

**Example 2** The Casualty Actuarial Society (CAS), wants to predict the exam performance of an actuarial candidate. Let $Y = 1$ if a candidate passes the exam and $Y = 0$ when the candidate is not successful. The CAS has access to the number of hours per week that the candidate studied for the test. This single observation will be the only explanatory variable used in the study, denoted by $X$. While the weekly study hours are definitely a useful indicator of success, there can be other factors that affect the exam performance. For example, the amount and quality of study hours may be limited if the candidate has a sleeping disorder, managerial role requiring long work hours, or family obligations. It is reasonable to assume, that $P\{Y = 1 \mid X = x\}$ or the probability of success, increases with the number of study hours. Suppose that,

$$P\{Y = 1 \mid X = x\} = \frac{x}{x + k}, \text{ where } k > 0.$$ 

Then the Bayes classifier has the following form:

$$g^B(x) = \begin{cases} 
1 & \text{if } E[Y \mid X = x] > \frac{1}{2} \\
0 & \text{otherwise.}
\end{cases}$$

$$= \begin{cases} 
1 & \text{if } x > k \\
0 & \text{otherwise.}
\end{cases}$$

Therefore, the student is predicted to obtain a passing score if $x > k$. Now suppose that $X \sim \text{Uniform}[0, 5k]$. Using the result of the Corollary 2.2, the misclassification error rate is:

$$L(g^B) = E\left\{\min\left(\frac{X}{X + k}, \frac{k}{X + k}\right)\right\}$$

Therefore, the student is predicted to obtain a passing score if $x > k$. Now suppose that $X \sim \text{Uniform}[0, 5k]$. Using the result of the Corollary 2.2, the misclassification error rate is:

$$L(g^B) = E\left\{\min\left(\frac{X}{X + k}, \frac{k}{X + k}\right)\right\}$$
\[
\frac{\min(X, k)}{X + k} = E \left\{ \frac{\min(X, k)}{X + k} \right\} \\
= \frac{1}{5k} \int_0^{5k} \frac{\min\{x, k\}}{x + k} \, dx \\
= \frac{1}{5k} \left( \int_0^k \frac{x}{x + k} \, dx + \int_k^{5k} \frac{k}{x + k} \, dx \right) \\
= \frac{1}{5k} \times ((k - k\log(2)) + k\log(3)) \\
= \frac{1}{5} \times (1 - \log(1.5)) \approx 28.11\%
\]

where \(\log(.)\) is the natural logarithm.

### 2.4 F-Errors

Corollary 2.2 relates the misclassification error of the Bayes classifier to expected value of a concave function of \(\pi(X)\). The following definition provides a more generalized view of the error rate of classifiers.

**Definition 2.3** Suppose that \(F\) is a concave function on [0, 1]. Then the **F-Error** corresponding to the random pair \((X, Y)\) is defined by,

\[
d_F(X, Y) = E\{F(\pi(X))\},
\]

where \(\pi(X) = P\{Y = 1 \mid X = x\}\).

An example of an F-Error, is the Bayes error \(L(g_B)\), where \(F(x) = \min(x, (1 - x))\). Another example, which will be discussed later, is the asymptotic nearest neighborhood error rate, where \(F(x) = 2x(1 - x)\). For more examples, see section 3.6 of Devroye et al. (1996). The following result will demonstrate that the F-Error will increase if \(X\) is transformed by an arbitrary function.

**Theorem 2.4** Let \(t : \mathbb{R}^d \to \mathbb{R}^k\) be an arbitrary function. Then for any distribution of the random pair \((X, Y)\),

\[
d_F(X, Y) \leq d_F(t(X), Y).
\]

**PROOF OF THEOREM 2.4**

The following is an expanded version of the proof found in section 3.6 of Devroye et al. (1996).

Define, \(\pi_t : \mathbb{R}^k \to [0, 1]\) by \(\pi_t(z) = P\{Y = 1 \mid t(X) = z\}\), where \(z \in \mathbb{R}^k\). Notice that,

\[
\pi_t(t(X)) = P\{Y = 1 \mid t(X)\} \\
= E\{I(Y = 1) \mid t(X)\} \\
= E\{E\{I(Y = 1) \mid X, t(X)\} \mid t(X)\}
\]
\[ d_F(t(X), Y) = E\{F(\pi(t(X)))\} \]
\[ = E\{F(E\{\pi(X) \mid t(X)\})\} \]
\[ \geq E\{E\{F(\pi(X) \mid t(X))\}\} \quad \text{(Jensen’s Inequality)} \]
\[ = E\{F(\pi(X))\} = d_f(X, Y) \quad \square \]

**Remark.** Thus, if \( X \) is replaced by \( t(X) \) (e.g. when information is lost) then the \( F \)-error increases or stays the same. The result of this theorem will be used in a future chapter, when providing the heuristic justification for using combined classifiers (discussed later).
3 Data Based Classifiers

3.1 Introduction to Data Based Classifiers

In the previous chapter, we witnessed that the Bayes classifier is optimal. In practice the optimal classifier is virtually unknown because the distribution of $(X, Y)$ is rarely known. If one has access to a training sample $(X_1, Y_1), \ldots, (X_n, Y_n)$, where $D_n = \{(X_i, Y_i)\}_{i=1}^n \overset{iid}{\sim} F$ then it may be possible to construct a classifier with a low probability of error. This sample-based classifier is denoted by $g_n(X; D_n)$ and the misclassification error rate is given by

$$L(g_n) = P\{g_n(X; D_n) \neq Y \mid D_n\}.$$

A sequence of classifiers $\{g_n \mid n \geq 1\}$ is usually called a rule. Whereas classifiers are functions, rules are sequences of functions. The quantity $E(L(g_n))$ would be able to provide insight on the quality of the average data sequence. Ideally, we would like our classifier to have perform similarly to the Bayes classifier. The definitions below, provide a few measures to gauge the performance of classifiers, with respect to the best classifier. For more information, one may refer to Chapter 6 of Devroye et al. (1996).

**Definition 3.1** A rule is said to be (weakly) consistent, for a certain distribution of $(X, Y)$, if

$$L(g_n) \overset{p}{\rightarrow} L(g^B), \quad \text{as } n \rightarrow \infty$$

As the amount of the data increases, a consistent rule ensures that the probability that the error rate is within a small distance of the optimal error rate is close to 1. The next definition, is a stronger form of consistency:

**Definition 3.2** A rule is strongly consistent, for a certain distribution of $(X, Y)$, if

$$L(g_n) \overset{a.s.}{\rightarrow} L(g^B), \quad \text{as } n \rightarrow \infty$$

Clearly, strong consistency implies consistency, since almost sure convergence implies convergence in probability. Except for training sequences of $D_n$ with zero probability, a strongly consistent rule implies that the error rate gets close to the optimum as the amount of the data increases.

Notice that both measures presented above, define consistency for a particular distribution of $(X, Y)$. A more powerful condition is presented below that holds for every distribution of $(X, Y)$:

**Definition 3.3** If a rule is consistent for all distributions of $(X, Y)$ it is said to be universally consistent.

The distribution of $(X, Y)$ is usually unknown, so having a rule that performs well for any distribution is an attractive property. Stone (1977) was able to show that the $K$-
Nearest Neighborhood rule (discussed later) is universally consistent. Before Stone’s breakthrough, many thought that a universally consistent rule did not exist. Subsequently, many other rules were shown to exhibit this property. In what follows, we provide the definitions and properties of several classifiers used in practice.

3.2 Linear Discriminant Analysis

Linear discriminant analysis could be thought of as splitting the space with a hyperplane and assigning a class to each space. Although this particular rule does not necessarily achieve the lowest error rate, it is very easy to implement and is the foundation of other better and popular methods in classification, such as neural networks. The following definitions will describe the rule and the corresponding misclassification error rate:

**Definition 3.4** The linear discriminant (LD) rule is given by the following expression,

\[
g(x) = \begin{cases} 
1 & \text{if } \sum_{i=1}^{d} a_i x^{(i)} + a_0 > 0 \\
0 & \text{otherwise.}
\end{cases}
\]  

(3.1)

where \(a_0, \ldots, a_d\) are the weights and \(x^{(1)}, \ldots, x^{(d)}\) represent the components of \(x\). Denote the error rate of this rule by \(L(a, a_0)\). Then the lowest probability of error within the class is given by

\[
L = \inf_{a \in \mathbb{R}, a_0 \in \mathbb{R}} L(a, a_0),
\]

where \(a = (a_1, \ldots, a_d)\).

Figure 3.1: Example of linear discriminant from Devroye et al. (1996).

In Figure 3.1 one can relate the line to the hyperplane discussed earlier. The LD rule assigns classes based on the region the observation lies in. The upper left region assigns the points to the shaded class, whereas the bottom right region assigns them to the unshaded class. This rule correctly classifies all observations except for the four shaded points below the hyperplane. Becker (1968) proved the following result that provides the best possible bound for this classifier in the univariate case:

**Theorem 3.1** Let \(L\) be as described in Definition 3.4. Suppose that there are two classes
with means, \( m_0 = E[X|Y = 0], m_1 = E[X|Y = 1] \), and variances \( \sigma_0^2 = \text{Var}[X|Y = 0] \) and \( \sigma_1^2 = \text{Var}[X|Y = 1] \). Then the following inequality holds,

\[
L(g^B) \leq L \leq \left( 1 + \frac{(m_0 - m_1)^2}{(\sigma_0^2 + \sigma_1^2)} \right)^{-1}.
\]

The following result yields a more generalized upper bound of the error that uses the outcome of Theorem 3.1. The proof can be found in section 4.2 of Devroye et al. (1996).

**Theorem 3.2** Let \( \mathbf{X}_0 \) and \( \mathbf{X}_1 \) be randomly distributed as \( \mathbf{X} \) given \( Y = 0 \) and \( Y = 1 \) respectively. Set \( m_0 = E[\mathbf{X}_0] \) and \( m_1 = E[\mathbf{X}_1] \). Let the covariance matrices \( \Sigma_1 = E[(\mathbf{X}_1 - E[\mathbf{X}_1])(\mathbf{X}_1 - E[\mathbf{X}_1])^T] \) and \( \Sigma_0 = E[(\mathbf{X}_0 - E[\mathbf{X}_0])(\mathbf{X}_0 - E[\mathbf{X}_0])^T] \). Then,

\[
L(g^B) \leq L \leq \inf_{\mathbf{a} \in \mathbb{R}^d} \left( 1 + \frac{\mathbf{a}^T(m_0 - m_1)^2}{\left( \mathbf{a}^T \Sigma_0 \mathbf{a}^{0.5} + \mathbf{a}^T \Sigma_1 \mathbf{a}^{0.5} \right)^2} \right)^{-1},
\]

where \( L \) is as defined in Definition 3.4.

Fisher (1936) suggested a method on how to use data based values to estimate \( \mathbf{a} \). Suppose the sample means for the two classes are denoted by \( \hat{m}_k \), where \( k = 0 \) or \( 1 \) respectively. Therefore, \( \hat{m}_k = \sum_{i:Y_i = k} \mathbf{X}_i / \sum_{i=1}^n I(Y_i = k) \). Next, the expression for the sample variance for the classes are defined below:

\[
\hat{\sigma}_k^2 = \sum_{i:Y_i = k} (\mathbf{a}^T \mathbf{X}_i - \mathbf{a}^T \hat{m}_k)^2 = \sum_{i:Y_i = k} \mathbf{a}^T (\mathbf{X}_i - \hat{m}_k)(\mathbf{X}_i - \hat{m}_k)^T \mathbf{a}
\]

\[
= \mathbf{a}^T \left( \sum_{i:Y_i = k} (\mathbf{X}_i - \hat{m}_k)(\mathbf{X}_i - \hat{m}_k)^T \right) \mathbf{a}
\]

\[
= \mathbf{a}^T \mathbf{S}_k \mathbf{a},
\]

where \( \mathbf{S}_k = \sum_{i:Y_i = k} (\mathbf{X}_i - \hat{m}_k)(\mathbf{X}_i - \hat{m}_k)^T \). The *Fisher linear discriminant* for the two class problem is said to be the linear function \( \mathbf{a}^T \mathbf{x} \), where the following criterion is maximized:

\[
J(\mathbf{a}) = \frac{(\mathbf{a}^T (m_1 - m_0))^2}{\mathbf{a}^T \mathbf{S}_1 \mathbf{a}}.
\]

Then the expression of \( \mathbf{a} \) that maximizes \( J(\mathbf{a}) \) is given by:

\[
\mathbf{a} = (\mathbf{S}_1 + \mathbf{S}_0)^{-1}(\hat{m}_1 - \hat{m}_0).
\]

Devroye et al. (1996) stated that Fisher’s suggestion was to ultimately replace the training sample with \( (\mathbf{a}^T \mathbf{X}_1, Y_1) \ldots, (\mathbf{a}^T \mathbf{X}_n, Y_n) \) and to perform a one-dimensional discrimination.
The drawback of Fisher’s methodology is that there are certain situations where this technique is not optimal (see section 4.3 of Devroye et al. (1996) for more information). The next example will illustrate this limitation and discuss the rare situation where the Bayes rule is a linear discriminant rule. For more information refer to section 4.4 of Devroye et al. (1996).

**Example 3** Recall that the density function for the multivariate normal distribution is given by,

\[ f(x) = \frac{1}{\sqrt{(2\pi)^d \det(\Sigma)}} e^{-\frac{1}{2}(x-m)^T \Sigma^{-1} (x-m)} . \]

If \( X \) has this density, then \( E\{X\} = m \) and \( E\{(X-m)(X-m)^T\} = \Sigma \), which is denoted by \( X \sim \text{MVN}(m, \Sigma) \). Now suppose that \( X \) has the following mixed density:

\[ (1-p)f_0(x) + pf_1(x) \]

where \( f_i \sim \text{MVN}(m_i, \Sigma_i) \) for \( i = 0, 1 \) and \( p = P\{Y = 1\} \). Below is the Bayes rule for determining the density a new observation belongs to:

\[ g^B(x) = \begin{cases} 
1 & \text{if } pf_1(x) > (1-p)f_0(x) \\
0 & \text{otherwise}.
\end{cases} \]

After taking the logarithms of both sides, one finds that the condition \( pf_1(x) > (1-p)f_0(x) \) is equivalent to,

\[
0 < \log(\det(\Sigma_0)/\det(\Sigma_1)) - 2\log((1-p)/p) \\
+ (x - m_0)^T \Sigma_0^{-1}(x - m_0) - (x - m_1)^T \Sigma_1^{-1}(x - m_1).
\]

Now if \( \Sigma_0 = \Sigma_1 = \Sigma \) (i.e. the covariance matrices of the two densities are equal),

\[
0 < 0 - 2\log((1-p)/p) \\
+ (x - m_0)^T \Sigma^{-1}(x - m_0) - (x - m_1)^T \Sigma^{-1}(x - m_1) \\
= (2\log((1-p)/p) + m_0^T \Sigma^{-1}m_0 - m_1^T \Sigma^{-1}m_1) - 2(m_1 - m_0)^T \Sigma^{-1}x \\
= a_0 + a^T x.
\]

If one denotes \( 2\log((1-p)/p) + m_0^T \Sigma^{-1}m_0 - m_1^T \Sigma^{-1}m_1 \) by \( a_0 \) and \( 2(m_1 - m_0)^T \Sigma^{-1} \) by \( a \), then it is clear that the Bayes rule becomes a linear discriminant rule. If \( \Sigma \) is replaced by the previously described data based estimate then it becomes Fisher’s linear discriminant rule. However, if \( \Sigma_0 \neq \Sigma_1 \) then we obtain a quadratic discriminant rule and Fisher’s linear discriminant becomes sub-optimal.
3.3 *K*-Nearest Neighborhood

The *K*-Nearest Neighborhood (*K*-NN) rule was first introduced in the early 1950s by Fix and Hoges. This popular technique is fairly easy to implement and has been used in both classification and regression. The *K*-NN uses a majority vote rule of the *K* closest neighbors using the underlying sample of data. The distance function defined below is used to determine the nearest neighbors.

**Definition 3.5** Let \( \|W\|_2^2 = (W_1^2 + \ldots + W_d^2) \) be the usual square norm. Then \( X_i \) is said to be the *Kth nearest neighbor* of \( X \) if \( \|X - X_i\| \) is the *Kth* smallest value among, \( \|X - X_1\|, \ldots, \|X - X_n\| \).

where in the case of a distance tie the candidate with the smaller index is considered to be closer \( X \). For example if \( \|X_i - x\| = \|X_j - x\| \), then \( X_i \) is a closer neighbor of \( x \) if \( i < j \).

This is a tie-breaking procedure mentioned above is a sub-optimal method and is no better than choosing the closer neighbor at random. Other techniques will be discussed later in this section. The next definition will give the explicit mathematical formulation for this aforementioned methodology:

**Definition 3.6** The *K*-NN rule is given by the following expression:

\[
g_n(x) = \begin{cases} 
1 & \text{if } \sum_{i=1}^{n} w_{n,i} I(Y_i = 1) > \sum_{i=1}^{n} w_{n,i} I(Y_i = 0) \\
0 & \text{otherwise.} 
\end{cases} \tag{3.2}
\]

where \( w_{n,i} = \frac{1}{K} \) if \( X_i \) is among the *K* nearest neighbors of \( x \) and 0 otherwise.

**Example 4** This example uses a subset of Bankruptcy data from Table 11.4 in Johnson and Wichern (1997). Using Moody’s Industrial Manuals, the authors gathered two key measures in assessing the health of a financial firm: cash flow to total debt and net income to total asset ratios. The objective of this example is to use the *K*-NN rule to determine the class a new observation will be assigned to. The classes are Bankrupt or NonBankrupt firms. This can assist regulators in determining which firms require a higher degree of scrutiny.

Suppose that regulatory officials are tasked with the assignment of assessing the health of Company X. Company X’s cash flow to debt ratio is .20 and their net income to total asset ratio is .08. The following table lists the 10 closest data points (nearest neighbors) in decreasing order (for the full table please refer to the appendix).
The above distance calculations are based on the Euclidean metric. Through the implementation of the 10-NN rule one can derive the following expression:

\[ \sum_{i=1}^{25} w_{25,i} I(Y_i = \text{NonBankrupt}) = .80 > \sum_{i=1}^{25} w_{25,i} I(Y_i = \text{Bankrupt}) = .20, \]

where \( w_{25,i} \) is equal to 0.1 if \( X_i \) is one of the 10 nearest neighbors of \( x \) and 0 otherwise. Therefore, Company X would be assigned to the NonBankrupt class. The regulators may want to allocate their resources to examine other firms. By observing the scatter plot in Figure 3.2, a similar conclusion could be derived without the formal calculations because the Company X point is surrounded by more points from the NonBankrupt class.
In the above example $K$ was chosen to be an even number. If the nearest neighbors had an equal number of Bankrupt and NonBankrupt firms, then we would have a tie. One does not have to resort to the naïve tie-breaking procedure mentioned in Definition 3.5. In fact, multiple sophisticated tie-breaking methods such as Stone’s method, adding a random component, and tie breaking by randomization are presented in section 11.2 of Devroye et al. (1996). Universal consistency is achievable for the adding a random component method. This procedure involves extending the dimension of the sample vectors of the data by 1. Consider the iid real-valued random variables $U, U_1, \ldots, U_n$ that are independent of $X, Y,$ and $D_n$ and their distribution has a density. The new $d + 1$ dimensional vectors are defined to be,

$$
\tilde{X} = (X, U), \tilde{X}_1 = (X_1, U_1), \ldots, \tilde{X}_n = (X_n, U_n).
$$

Through the independence assumption of the random variables, the Bayes error corresponding to $(\tilde{X}, Y)$ is the same as $(X, Y)$. Then, the newly constructed data set is used in place of $D_n$. Ties occur with 0 probability because $X$ and $U$ are independent and $U$ has a density.

The next result will help one assess the asymptotic performance of the nearest neighbor rule. Cover and Hart (1967) provided the justification for this outcome using various continuity conditions. Stone (1977) was able to prove this result without these assumptions.

**Theorem 3.3** For any distribution of $(X, Y)$,

$$
\lim_{n \to \infty} E\{L(g_n)\} = 2E\{\pi(X)(1 - \pi(X))\},
$$

where $\pi(X) = P\{Y = 1 \mid X\}$ and $L(g_n)$ represents the error rate of the $K$-NN rule.

Thus, the nearest neighborhood rule is at most asymptotically twice as bad as the Bayes rule. The next theorem will help establish the strong consistency of the nearest neighbor rule, wherever $X$ has an absolutely continuous distribution. For the proof of this result please refer to Chapter 10 of Devroye and Györfi (1985), Zhao (1987), or section 11.1 of Devroye et al. (1996).

**Theorem 3.4** Let $L(g_n)$ represent the error rate for the $k$-NN rule. Assume that $X$ has a density. If $k/n \to 0$ and $k \to \infty$ then for every $\epsilon > 0$ there is an $n_0$ such that for $n > n_0$

$$
P(L(g_n) - L(g^B) > \epsilon) \leq 2e^{-\frac{\epsilon^2}{(\gamma_d^2)}} ,
$$

where $\gamma_d$ is the minimal number of cones centered at the origin of angle $\pi/6$ that cover $\mathbb{R}^d$. By the result of the Borel-Cantelli Lemma, we get that $L(g_n) \overset{a.s.}{\to} L(g^B)$ because $\forall \epsilon > 0$

$$
\sum_{n=1}^{\infty} 2e^{-\frac{\epsilon^2}{(\gamma_d^2)}} = \frac{2}{e^{\epsilon^2/(\gamma_d^2)} - 1} < \infty.
$$

Therefore, the $K$-NN rule is strongly consistent. Notice that the density assumption of the
above result was made to avoid problems with distance ties. If $X$ has a density distance ties occur with zero probability and are therefore irrelevant.

### 3.4 Kernel Classifiers

The $K$-NN rule provides equal voting weight to all the nearest neighbors. The appealing quality of the Kernel classification rule is that it provides more weight to closer points. A Kernel can be thought of as a specialized probability density function, a formal definition is given below:

**Definition 3.7** A **Kernel** function is a map of the form $K: \mathbb{R}^d \rightarrow \mathbb{R}$ that is usually monotone decreasing and nonnegative.

The Gaussian kernel and Epanechnikov kernel are two of the most popular kernels used in practice. The expression of the Epanechnikov is,

$$K_E(u) = \frac{3}{4}(1 - u^2) \ast I\{|u| < 1\}.$$

See Figure 3.3 for a graphical representation of this kernel. The next definition will use kernels to construct a classifier:

**Definition 3.8** The **Kernel classification rule** is given by

$$g_n(x) = \begin{cases} 
1 & \text{if } \sum_{i=1}^{n} K(\frac{x-x_i}{h})I\{Y_i = 1\} > \sum_{i=1}^{n} K(\frac{x-x_i}{h})I\{Y_i = 0\} \\
0 & \text{otherwise.}
\end{cases} \quad (3.3)$$

where $h$, known as the bandwith, is a smoothing parameter that depends on $n$.

This classifier has a sensible formulation since it provides more weight for being closer to the observation. In addition, it is possible to show that this rule is universally consistent for a large class of kernel functions, called regular kernels:

![Figure 3.3: Graph of Epanechnikov kernel](image)
**Definition 3.9** Let the ball of radius, \( r > 0 \), centered at the origin, be denoted by \( S_{0,r} \). A kernel, \( K \) is said to be **regular** if it satisfies the following conditions:

(i) \( K \) is a nonnegative function.

(ii) There is a constant \( b > 0 \) such that \( K(x) \geq b \{ x \in S_{0,r} \} \).

(iii) \( \int \sup_{z \in x + S_{0,r}} K(z) \, dx < \infty \).

The following theorem in conjunction with the result of the Borel-Cantelli Lemma implies the strong universal consistency of the Kernel classifier. See Devroye and Krzyzak (1989) for the proof of this result.

**Theorem 3.5** Assume \( K \) is a regular kernel. Suppose that \( h \to 0 \) and \( nh^d \to \infty \), as \( n \to \infty \), and then for every \( \epsilon > 0 \) there is an integer \( n_0 \) such that \( \forall n > n_0 \) for the error probability of the kernel rule we have

\[
P(L(g^K) - L(g^B) < \epsilon) \leq 2e^{-n\epsilon^2/(32\rho^2)}.
\]

The constant \( \rho \) depends on \( K \) and the dimension.

### 3.5 Partition Rules

The set of classifiers that partition \( \mathbb{R}^d \) into disjoint **cells**, denoted by \( A_{n1}, A_{n2}, \ldots \), are known as **partition rules**. Let \( \mathcal{P}_n = \{ A_{n1}, A_{n2}, \cdots \} \) be a finite or countably infinite partition of \( \mathbb{R}^d \), that is \( \bigcup_{j=1}^{\infty} A_{nj} = \mathbb{R}^d \) and \( A_{ni} \cap A_{nj} = \emptyset \ \forall i \neq j \). Here the subscript \( n \) of \( \mathcal{P}_n \) shows that the partition may depend on the sample size. Intuitively as \( n \) grows, one would expect the partitions to get finer. Partitioning estimates work by taking a majority vote of the most popular class label among the vectors that fall into a same cell as the observation.

A more formal definition is provided below:

**Definition 3.10** If \( A(x) \) is the cell that contains \( x \), then the partitioning rule has the following form,

\[
g_n(x) = \begin{cases} 
1 & \text{if } \sum_{i=1}^{n} I(Y_i = 1, X_i \in A(x)) > \sum_{i=1}^{n} I(Y_i = 0, X_i \in A(x)) \\
0 & \text{otherwise.}
\end{cases}
\]

Therefore, the classifier uses the labels corresponding to the vectors of the training sample that fall within the same cell to determine the label of \( x \). Devroye et al. (1996) justified the consistency of this classifier under two conditions, given below:

**Condition 3.1** The diameter of a set \( A \), defined by \( \text{diam}(A) = \sup_{x, y \in A} \| x - y \| \), is such that \( \text{diam}(A(X)) \to 0 \).
**Condition 3.2** Let $N(x) = \sum_{i=1}^{n} I(X_i \in A(x))$ be the number of vectors that are in the same cell as $x$. Then $N(X) \overset{p}{\to} \infty$.

The next result establishes the consistency of the partitioning rule. The proof can be found in section 6.3 of Devroye et al. (1996).

**Theorem 3.6** If the partitioning rule satisfies conditions 3.1 and 3.2, then we have,

$$E\{L(g_n)\} \to L(g^B).$$

**Remark 3.1** The two conditions are essentially working against each other. Condition 3.1 requires the cells to be small enough to detect local changes in the distribution. However, Condition 3.2 requires the cells to be large enough to contain a sufficient number of sample vectors within the cell to make a credible decision.

The cubic histogram rule is a special class of partitioning rules that partitions $\mathbb{R}^d$ into cubes of the same size and make a decision. In section 6.4 of Devroye et al. (1996) the universal consistency of this rule is justified. Classification trees partition $\mathbb{R}^d$ into regions, which are generally hyper-rectangles that are parallel to the axes (see Chapter 20 of Devroye et al. (1996) for more information). A random forest classification rule is the collection of a large number of tree classifiers based on bootstrap samples. Each tree votes for the most popular class. Brieman (2001) provided an example in his paper, which illustrated the usefulness of tree classifiers when dealing with higher dimensional data. In this situation, he showed that a single tree classifier was no better than choosing the classes at random. However, when combining the information of several trees he was able to achieve Bayesian accuracy.

### 3.6 Support Vector Network

The support vector network classifier non-linearly maps input vectors into a much higher dimension space, where a decision surface is created (see Cortes and Vapnik (1995)). The final decision function relies only on the supporting patterns, a term first coined by Vapnik (1982).

Recall, the two class framework with $n$ training samples. Let $\tilde{Y}_k$ be 1 if $X_k$ belongs to class 1 and $-1$ otherwise. The training data is said to be linearly separable without error if there exists a vector $w$ and scalar $b$, where

$$\tilde{Y}_i (wX_i + b) \geq 1.$$  

Cortes and Vapnik (1995) discuss the situation where it is not possible to separate the training data without error. If the above expression holds with equality then we call $X_i$ a support vector. The optimal hyper-plane, $w_0x + b_0 = 0$ separates the training data with maximal distance between the support vectors (see Figure 3.4). Boser et al. (1992) solved this optimization problem through the use of Lagrange multipliers. Using their results, the definition of the decision rule is presented below.
Figure 3.4: Figure 2 from Cortes and Vapnik (1995). The support vectors are marked in grey. They define the margin maximum separation between the two classes.

**Definition 3.11** The decision rule for the support vector network is given by the following:

\[
D(x) = \sum_{k=1}^{n} \tilde{Y}_k \alpha_k^* \tilde{K}(x_k, x) + b, \quad \alpha_k^* \geq 0.
\]

The predefined kernel (different from the kernel discussed earlier) \( \tilde{K} \) is a mapping, where \( \tilde{K} : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R} \). \( \alpha_k^* \) are Lagrange multipliers (which are non-zero if \( x_k \) is a support vector) and \( b \) is the bias. If \( D(x) > 0 \) then we say that \( x \) belongs to class 1, otherwise class 0.

Boser et al. (1992) and Cortes and Vapnik (1995) selected a Gaussian kernel for their experimental results. Another choice could be a radial basis function (see Broomhead and Lowe (1988)), which is given below:

\[
\tilde{K}(x_k, x_j) = e^{-\gamma \|x_k - x_j\|^2}, \quad \gamma > 0.
\]

The bias can be selected using an *a priori* estimate, which was first discussed by Vapnik (1982). One may reference Boser et al. (1992) for more information regarding the selection of \( \tilde{K} \) and \( b \). The next result that was justified by section 10.3 of Vapnik (1998), provides insight on the performance of the support vector network.

**Theorem 3.7** Assume the training data is linearly separable without error by an optimal hyperplane. Then

\[
E\{L(g_n)\} \leq \frac{1}{n} \sum_{i=1}^{n} E\{I(\tilde{Y}_i(wX_i + b) = 1)\},
\]

where \( L(g_n) \) represents the error rate of the support vector network and \( n \) represents the size of the training samples.
Therefore, the above result indicates that the misclassification error is bounded above by the ratio of the expected number of support vectors to the sample size of the training data. Cortes and Vapnik (1995) noticed that if there are only a small number of vectors compromising the group of support vectors, then the generalization ability of the optimal hyperplane is high. In fact, they provided an example where the hyperplane generalized well in a billion dimensional space and the aforementioned ratio was as low as .03.
4 The Proposed Theoretical Combined Classifier

4.1 Introduction to Combining Classifiers

The previous chapter presented multiple classifiers based on different theories and concepts. The motivation for combining classifiers is that one rarely knows the distribution of the data, so it will be difficult to evaluate which classifier works best. Each method serves a unique purpose and has its own merits. For example, the popular Fisher’s LDA method works great when trying to separate two normal distributions. However, if the covariance matrices are not the same or the distributions are not normal, the method becomes sub-optimal. Devroye et al. (1996) discuss how there are even some distributions that will cause the Fisher’s LDA method to have a probability of misclassification to be close to 1. A nearest neighbor classifier may be preferable when trying to separate non normal distributions.

In the past few decades, there has been growing interest and developments in constructing a combined classifier, which potentially has higher predictive power than each of the individual estimates. Majority voting methods are widely used in tree-based classification, for example the bootstrap aggregation (bagging) and out-of-bag methods of Brieman (2001) with tree classifiers and Xu et al. (1992) in handwriting recognition. Additionally, Fauvel et al. (2006) applied combined classification to remote sensing images. Adler et al. (2011) used combined classification for paired data. Another approach is to consider the weighted average of estimate class conditional probabilities that are produced by the individual classifier; see for example LeBlanc and Tibshirani (1996), Brieman (1995), Xu et al. (1992), Wolpert (1992), and Mojirsheibani (1997). Mojirsheibani (1999) proposed an alternative nonlinear approach. Recently, Balakrishnan and Mojirsheibani (2015) presented an optimal combined classifier for the two class problem. In addition, they implemented a data splitting approach to construct the sample based combined classifier. The theoretical focus of this paper is to extend this optimal classifier to \( K \) classes.

The concept of combining estimates is not unique to the field of classification. In order to create weights for combining regression estimates van der Laan et al (2007) used a \( V \)-fold cross-validation approach. Furthermore, van der Laan et al. (2006) used cross-validation to choose an optimal model from several candidates. Yang (2000, 2004) reduced the \( L_2 \) risk by using linear combinations of regression estimates. More recently, Biau et al. (2013) created a method for combining regression functions, which perform at the same level or better then the best estimator in the \( L_2 \) sense. The authors of this paper also released an R package called COBRA, which has powerful numerical applications.

There are two categories for combining classifiers, one that starts with a large number of similar homogeneous base classifiers that are then combined to a construct the final rule. An example of this category are the random forest classifiers presented in the work of Brieman (2001). The other category aggregates individual classifiers that are not necessarily homogeneous. The work of LeBlanc and Tibshirani (1996) and Mojirsheibani (1999) also
fall under this category. Furthermore, the results of this thesis fall under this category.

4.2 Heuristic and Motivational Justification

The heuristic justification for the two class problem is straightforward, based on the hypothetical situations where $g$ is non-random. Let $(X, Y) \in \mathbb{R}^d \times \{0, 1\}$ be a random pair and let $g : \mathbb{R}^d \rightarrow \{0, 1\}$ be any nonrandom classifier for predicting $Y$ based on $X$. Also, let $P\{g(X) \neq Y\}$ be the misclassification error of $g$. Now define the new classifier $\tilde{\psi}$ (for predicting the same $Y$) by

$$\tilde{\psi}(g) = \left\{ \begin{array}{ll} 1 & \text{if } E\left[Y \mid g(X) = g(x)\right] > \frac{1}{2} \\ 0 & \text{otherwise} \end{array} \right.$$  (4.1)

Then the following result shows that $\psi$ is at least as good as $g$ in terms of its misclassification error rate:

Theorem 4.1 Let $g$ and $\tilde{\psi}$ be as above. Then

$$L(\tilde{\psi}(g)) = P\{\tilde{\psi}(g(X)) \neq Y\} \leq P\{g(X) \neq Y\} = L(g),$$

where the equality holds when $g$ is the Bayes classifier, i.e. $g(x) = g^B(x)$

PROOF OF THEOREM 4.1

$$L(\tilde{\psi}(g)) = 1 - \sum_{y=0}^{1} E\{I(\tilde{\psi}(g(X)) = i, Y = i)\} \quad \text{(by Corollary 2.3)}$$

$$= 1 - \sum_{y=0}^{1} E\{I(\tilde{\psi}(g(X)) = y)P\{Y = y \mid g(X)\}\}$$

$$= 1 - \sum_{y=0}^{1} E\left\{ \max_{y=0,1} P\{Y = y \mid g(X)\} \right\}$$

$$\leq 1 - \sum_{y=0}^{1} E\{I(g(X) = y)P\{Y = y \mid g(X)\}\}$$

$$= 1 - \sum_{y=0}^{1} E\{E\{I(g(X) = i, Y = i) \mid g(X)\}\}$$

$$= 1 - \sum_{y=0}^{1} E\{I(g(X) = i, Y = i)\} = L(g) \quad \text{(by Corollary 2.3)}$$

If $g(x) = g^B(x)$, then we have the following result:

$$L(g^B) = E\left\{ \min (\pi(x), 1 - \pi(x)) \right\} \quad \text{(by Corollary 2.2; where } \pi(x) = P\{Y = 1 \mid X = x}\)
\[ = 1 - E \left\{ \max_{y=0,1} P\{Y = y \mid X\} \right\} \]
\[ = 1 - E \left\{ \max_{y=0,1} P\{Y = y \mid g(X)\} \right\} = L(\tilde{\psi}(g)) \]  \(\square\)  \hspace{3.7cm} (4.2)

**Remark.** The above proof uses arguments similar to the proof of Theorem 2.1. However, the conditional expectation and probabilities presented above, condition on \(g(X)\), rather than \(X\). This is another way of expressing the error rate of a classifier.

Next, suppose that there are \(M \geq 2\) nonrandom classifiers, \(g_1, \ldots, g_M\), for predicting \(Y\) based on \(X\). Define the combined classifier \(\hat{\psi}_M\) according to
\[ \hat{\psi}_M(g_1(x), \ldots, g_M(x)) = \begin{cases} 1 & \text{if } E\left[Y \mid g_1(X) = g_1(x), \ldots, g_M(X) = g_M(x) \right] > \frac{1}{2} \\ 0 & \text{otherwise.} \end{cases} \]

Consequently the next result follows directly from Theorem 2.4:

**Theorem 4.2** For each individual classifier \(g_k, k = 1, \ldots, M\),
\[ P \left\{ \hat{\psi}_M(g_1(X), \ldots, g_M(X)) \neq Y \right\} \leq P\{\tilde{\psi}(g_k(X)) \neq Y\}, \]
where \(\tilde{\psi}\) is as in (4.1).

**Proof of Theorem 4.2**
This is an immediate consequence of Theorem 2.4.

### 4.3 The Theoretical Combined Classifier

Let \((X, Y) \sim F\) be a random pair, where \(X \in \mathbb{R}^d\) is a vector of covariates that is used to predict a class, \(Y \in \{1, 2, \ldots, K\}\). As mentioned earlier, finding \(g^B\) is impossible since the distribution of \((X, Y)\) is always fully or partially unknown. Thus, if one has a random sample \((X_1, Y_1), \ldots, (X_n, Y_n)\), sample based classifiers \(\hat{g}_n\) must be created. Suppose that there are \(M\) different sample based classifiers. These may include simpler classifiers such as nearest neighbor (NN) classifiers, linear classifiers, quadratic classifiers, etc. Let these \(M\) classifiers be denoted by \(\hat{g}_{n,1}(x), \ldots, \hat{g}_{n,M}(x)\) and
\[ \hat{g}_n(x) = \begin{pmatrix} \hat{g}_{n,1}(x) \\ \vdots \\ \hat{g}_{n,M}(x) \end{pmatrix}. \]

Set,
\[ H_{y,n}(\hat{g}_n(x)) := P\{Y = y \mid \hat{g}_n(X) = \hat{g}_n(x)\} \]  \hspace{3.7cm} (4.3)
Define the combined classifier $\Gamma_n(x): \{1, 2, \ldots K\}^M \to \{1, 2, \ldots K\}$ by,

$$\Gamma_n(\hat{g}_n(x)) = \arg\max_{1 \leq y \leq K} H_{y,n}(\hat{g}_n(x)) \quad (4.4)$$

Once again, the underlying distributions are unknown and cannot be used in practice. How good is (4.4)? The following theorem shows that $\Gamma_n$ is asymptotically optimal in the sense that its overall misclassification error rate:

$$L(\Gamma_n) = P\{\Gamma_n(\hat{g}_n(X)) \neq Y\}, \quad (4.5)$$

is less than or equal to that of any other combined classifier.

**Theorem 4.3** Let $\Gamma_n$ be defined as in (4.4) and $\Psi_n: \{1, 2, \ldots K\}^M \to \{1, 2, \ldots K\}$ be any other function of $\hat{g}_n(x)$. Next, let $L(\Psi_n) = P\{\Psi_n(\hat{g}_n(X)) \neq Y\}$, be the misclassification error rate of $\Psi_n$. Then,

$$L(\Psi_n) - L(\Gamma_n) \geq 0$$

**Proof of Theorem 4.3**

$$L(\Gamma_n) = 1 - \sum_{y=1}^{K} E\{I(\Gamma_n(\hat{g}_n(X)) = y, Y = y)\} \quad \text{(by Corollary 2.3)}$$

$$= 1 - \sum_{y=1}^{K} E\{E\{I(\Gamma_n(\hat{g}_n(X)) = y, Y = y) \mid \hat{g}_n(X)\}\}$$

$$= 1 - \sum_{y=1}^{K} E\{P\{Y = y \mid \hat{g}_n(X)\}I(\Gamma_n(\hat{g}_n(X)) = y)\}$$

$$= 1 - \sum_{y=1}^{K} E\{H_{y,n}(\hat{g}_n(X))I(\Gamma_n(\hat{g}_n(X)) = y)\}$$

$$= 1 - \sum_{y=1}^{K} E\{H_{y,n}(\hat{g}_n(X))I(\arg\max_{1 \leq j \leq K} H_{j,n}(\hat{g}_n(X)) = y)\}$$

$$= 1 - E\{\max_{1 \leq y \leq K} H_{y,n}(\hat{g}_n(X))\} \quad (4.6)$$

$$\leq 1 - E\{H_{n,\Psi_n}(\hat{g}_n(X))\}$$

$$= 1 - \sum_{y=1}^{K} E\{H_{y,n}(\hat{g}_n(X))I(\Psi_n(\hat{g}_n(X)) = y)\}$$

$$= 1 - P\{\Psi_n(\hat{g}_n(X)) = Y\} = L(\Psi_n) \quad \square$$
Remark. Notice that this misclassification error rate is an unconditional probability. This can have some unfortunate implications with respect to the numerical applications. The amount of data within the random sample has to be fairly large for the combined classifier to outperform other estimators.

4.4 Sample Based Classifier

The classifier \( \Gamma_n \) is not useful in practice because it can rarely be computed. It depends on the unknown conditional probabilities \( H_{y,n}(\hat{g}_n(x)) \), where \( y = 1, \ldots, K \), defined in (4.3). A data splitting approach can be used to construct the optimal \( K \)-class sample-based classifier. The approach is easy to implement and useful in practice. First, we split our original data set \( D_n \) into two distinct parts \( D_\ell \) and \( D_m \) of sizes \( \ell \) and \( m \), respectively, where \( m + \ell = n \), \( D_\ell \cap D_m = \emptyset \), and \( D_\ell \cup D_m = D_n \). Let \( \hat{g}_{m,1}, \ldots, \hat{g}_{m,M} \) be \( M \) classifiers based on \( D_m \) only and

\[
\hat{g}_m(x) = \begin{pmatrix}
\hat{g}_{m,1}(x) \\
\vdots \\
\hat{g}_{m,M}(x)
\end{pmatrix}
\]

Let

\[
N_{m,\ell}(x) = \sum_{i: (X_i, Y_i) \in D_\ell} I\{\hat{g}_m(X_i) = \hat{g}_m(x)\}
\]

(4.7)

For large \( K \) and finite samples, \( N_{m,\ell} \) may end up being a small number. And now consider the sample based analog to \( H_{y,n}(\hat{g}_n(x)) \)

\[
\hat{H}_{y,m,\ell}(\hat{g}_m(x)) = \frac{1}{N_{m,\ell}(x)} \sum_{i: (X_i, Y_i) \in D_\ell} I\{Y_i = y, \hat{g}_m(X_i) = \hat{g}_m(x)\}
\]

(4.8)

where \( y = 1, \ldots, K \) and by convention, \( 0/0 := 0 \).

Define the proposed optimal \( K \)-class sample-based classifier as,

\[
\hat{\Gamma}_n(\hat{g}_m(x)) = \arg\max_{1 \leq y \leq K} \hat{H}_{y,m,\ell}(\hat{g}_m(x)).
\]

(4.9)

The next result will prove the asymptotic optimality of \( \hat{\Gamma}_n \) under the following conditions:

Condition 4.1 As \( n \) (and thus \( \ell \)) \( \to \infty \), we have \( N_{m,\ell}(X) \to \infty \) in probability, where \( N_{m,\ell}(X) \) as in (4.7).

Condition 4.2 The error rate of the classifier \( \Gamma_n \) is such that \( L(\Gamma_n) \to c \) for some \( c \in [0, 1] \), as \( n \to \infty \)

Remark. We note that conditional on \( \mathbb{D}_m \), one may view \( N_{m,\ell}(X) \) as the number of successes of \( \ell \) Bernoulli trials. Therefore, it will become infinitely large as \( \ell \to \infty \).
Balakrishnan and Mojirsheibani (2015) used this assumption and Devroye et. al (1996) applied a variation of this assumption.

**Theorem 4.4** Let \( \hat{\Gamma}_n \) be defined as in (4.9). Then under Conditions 4.1 and 4.2, as \( n \to \infty \),
\[
L(\hat{\Gamma}_n) - L(\Gamma_n) \to 0
\]

To prove this theorem we first state a technical lemma.

**Lemma 4.1** Let \( \hat{\Gamma}_n \) and \( \Gamma_n \) be defined as above. Then we have
\[
L(\hat{\Gamma}_n) - L(\Gamma_n) \leq \sum_{y=1}^{K} E \left| \hat{H}_{y,m,\ell}(\hat{g}_m(x)) - H_{y,m}(\hat{g}_m(x)) \right|
\]

The proof of Lemma 4.1 appears after the proof of Theorem 4.4.

**PROOF OF THEOREM 4.4**

In view of Lemma 4.1,
\[
L(\hat{\Gamma}_n) - L(\Gamma_n) = L(\hat{\Gamma}_n) - L(\Gamma_m)) + (L(\Gamma_m) - L(\Gamma_n))
\]
\[
\leq \sum_{y=1}^{K} E \left| \hat{H}_{y,m,\ell}(\hat{g}_m(x)) - H_{y,m}(\hat{g}_m(x)) \right| + (L(\Gamma_m) - L(\Gamma_n))
\]

it is sufficient to show that for \( y = 1, \ldots, K \):
\[
E \left| \hat{H}_{y,m,\ell}(\hat{g}_n(x)) - H_{y,n}(\hat{g}_m(x)) \right| \to 0, \text{ as } n \to \infty
\]

which implies,
\[
\sum_{y=1}^{K} E \left| \hat{H}_{y,m,\ell}(\hat{g}_n(x)) - H_{y,n}(\hat{g}_m(x)) \right| \to 0, \text{ as } n \to \infty
\]

Choose an arbitrary \( y \in (1, \ldots, K) \). Denote \( \mathcal{D}_\ell \) by \([X_1, Y_1], \ldots, (X_\ell, Y_\ell)\) and those in \( \mathcal{D}_m \) by \([X_{\ell+1}, Y_{\ell+1}], \ldots, (X_{\ell+m}, Y_{\ell+m})\), where \( \ell + m = n \). Conditional on the random variables \( \hat{g}_m(X) \) and \( I(\hat{g}_m(X)) = \hat{g}_m(X) \), \( i = 1, \ldots, \ell \) the random variable
\[
N_{m,\ell}(X) \times \hat{H}_{y,m,\ell}(\hat{g}_n(X)) \sim \text{Binomial}(N_{m,\ell}(X), H_{y,m}(\hat{g}_n(X)))
\]

for \( y = 1, \ldots, K \).

Let \( U_i,m(X) := I(\hat{g}_m(X)) = g_m(X) \) where \( i = 1, \ldots, l \). We have,
Thus we have,

\begin{align*}
E \left\{ \left| \hat{H}_{y,m,\ell}(\hat{g}_n(X)) - H_{y,n}(\hat{g}_m(X)) \right|^2 \left| D_{m, X, U_{1,m}(X), \ldots, U_{l,m}(X)} \right\} \right. \\
\leq E \left\{ \left| \frac{N_{m,\ell}(X) \times \hat{H}_{y,m,\ell}(\hat{g}_n(X))}{N_{m,\ell}(X)} - H_{y,n}(\hat{g}_m(X)) \right|^2 \cdot I(N_{m,\ell}(X) > 0) \left| D_{m, X, \{U_{i,m}(X)\}_{i=1}^l} \right\} \right. \\
+ I(N_{m,\ell}(X) > 0) \\
= E \left\{ \left| \frac{Bin(N_{m,\ell}(X), H_{j}(\hat{g}_m(X)))}{N_{m,\ell}(X)} - H_{y,n}(\hat{g}_m(X)) \right|^2 \cdot I(N_{m,\ell}(X) > 0) \left| D_{m, X, \{U_{i,m}(X)\}_{i=1}^l} \right\} \right. \\
+ I(N_{m,\ell}(X) = 0) \\
= \frac{I(N_{m,\ell}(X) > 0)}{4N_{m,\ell}(X)} + I(N_{m,\ell}(X) = 0) \\
\leq \frac{I(N_{m,\ell}(X) > 0)}{4N_{m,\ell}(X)} + I(N_{m,\ell}(X) = 0)
\end{align*}

Thus we have,

\begin{align*}
E \left\{ \left| \hat{H}_{y,m,\ell}(\hat{g}_n(X)) - H_{y,n}(\hat{g}_m(X)) \right|^2 \right. \\
\leq \sqrt{E \left\{ \left| \hat{H}_{y,m,\ell}(\hat{g}_n(X)) - H_{y,n}(\hat{g}_m(X)) \right|^2 \right.} \\
= \sqrt{E \left\{ \left| \hat{H}_{y,m,\ell}(\hat{g}_n(X)) - H_{y,n}(\hat{g}_m(X)) \right|^2 \right.} \cdot \left\{ \hat{g}_m(X), \{U_{i,m}(X)\}_{i=1}^l \right\} \\
\leq \sqrt{E \left\{ \frac{I(N_{m,\ell}(X) > 0)}{4N_{m,\ell}(X)} \right.} + P(N_{m,\ell}(X) = 0) \\
= \sqrt{E \left\{ \frac{I(0 < N_{m,\ell}(X) \leq k)}{4N_{m,\ell}(X)} \right.} + E \left\{ \frac{I(N_{m,\ell}(X) > k)}{4N_{m,\ell}(X)} \right.} + P(N_{m,\ell}(X) = 0) \\
\leq \sqrt{E \left\{ \frac{P(N_{m,\ell}(X) \leq k)}{4} \right.} + \frac{1}{4k} + P(N_{m,\ell}(X) = 0)
\end{align*}

Since \( k \) is arbitrary, we choose \( k \) to be large enough so that \( \frac{1}{4k} \) goes to 0. Since \( N_{m,\ell}(X) \xrightarrow{p} \infty \) then we can choose \( n \) large enough to make the first and third term small. The theorem follows since \( L(\Gamma_m) - L(\Gamma_n) \rightarrow 0 \) as \( n \rightarrow \infty \), by Condition B."
\[
\begin{align*}
&= E \left( \max_{1 \leq y \leq K} H_{y,n}(\hat{g}_m(x)) \right) - E \left( \max_{1 \leq y \leq K} \hat{H}_{y,m,\ell}(\hat{g}_m(x)) \right) \quad \text{(by Equation 4.6)} \\
&= E (H_{\Gamma,n}(\hat{g}_m(x))) - E (\hat{H}_{\Gamma,n,\ell}(\hat{g}_m(x))) \\
&\leq E (H_{\Gamma,n}(\hat{g}_m(x))) - E (\hat{H}_{\Gamma,n,\ell}(\hat{g}_m(x))) \\
&\leq \sum_{y=1}^{K} E \left| \hat{H}_{y,m,\ell}(\hat{g}_m(x)) - H_{y,n}(\hat{g}_m(x)) \right| 
\end{align*}
\]
The objective of this section is to examine the class membership prediction, $Y = \{1, 2, 3\}$ of the vector of covariates $X$, based on several different assumptions of distributions and sample sizes. The class probabilities are all equal, i.e. $P(Y = i) = 1/3$ for $i = 1, 2, 3$. The following are the individual classifiers that are used:

- Random Forests (RF) (Brieman (2001)),
- Support Vector Machines (SVM) (Boser et al. (1992)),
- A one Nearest Neighbor (1-NN) classifier,
- A five Nearest Neighbor (5-NN) classifier,
- Linear Discriminant Analysis (LDA).

We compare the performance of the above classifiers to the combined classifier $\hat{\Gamma}_n$ defined in (4.9).

**Example 5** Suppose that the covariate vector $X \in \mathbb{R}^d$, $d = 5$, where if $Y = 1$ (i.e. class 1 or “Cauchy”) then $X$ has a 5-dimensional Cauchy distribution with independent components. If $Y = 2$ (i.e. class 2 or “Norm1”) then $X \sim N_d(0, c_1 \Sigma)$ or if $Y = 3$ (i.e. class 3 or “Norm2”) then $X \sim N_d(3, c_2 \Sigma)$, where $c_1 = 2$ and $c_2 = 10$ with,

$$\Sigma = (\sigma_{ij})_{i,j=1,...,5} \quad \text{and} \quad \sigma_{ij} = 2^{-|i-j|}.$$

From each distribution, samples of size $n = 50, 100, 200$ were randomly generated. The grid of values of $n$ helps monitor the performance of different classifiers as $n$ increases.

Furthermore, given a sample size $n$, the five individual classifiers listed above were constructed and then used to create the combined classifier. The data-splitting method is slightly different from the approach defined in section 4.4. Rather than using a single split of the data with $m = 0.65n$ and $\ell = 0.35n = n - m$, to estimate $\hat{H}_{y,m,\ell}$ defined in (4.8), one performs multiple data splits and uses the average. In this numerical example there were a total of 20 data splits. The following is a mathematical representation of this revised approach:

$$\hat{H}_{y,m,\ell} = \frac{1}{20} \sum_{i=1}^{20} \hat{H}_{y,m,\ell}^{(i)}.$$

$\hat{H}_{y,m,\ell}^{(i)}$ is derived from the $i^{th}$ data split. This method achieves asymptotic results quicker but requires significantly more computing power. The theoretical justification for this improved numerical accuracy is beyond the scope of this study.
To estimate the misclassification error of the various classifiers an additional sample of 300 observations from each distribution was generated that was used as the test sample. 25 estimates for the misclassification error were obtained by repeating the entire process above 25 times. The table below contains a summary of the average error probabilities for different sample sizes and their respective standard errors:

Table 5.1: Misclassification and Standard Error Summary

<table>
<thead>
<tr>
<th>n</th>
<th>Opt</th>
<th>Maj</th>
<th>LDA</th>
<th>1-NN</th>
<th>5-NN</th>
<th>SVM</th>
<th>RF</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.382</td>
<td>0.399</td>
<td>0.445</td>
<td>0.397</td>
<td><strong>0.379</strong></td>
<td>0.462</td>
<td>0.390</td>
</tr>
<tr>
<td></td>
<td>(0.0035)</td>
<td>(0.0027)</td>
<td>(0.0045)</td>
<td>(0.0025)</td>
<td>(0.0026)</td>
<td>(0.0051)</td>
<td>(0.0037)</td>
</tr>
<tr>
<td>100</td>
<td><strong>0.359</strong></td>
<td>0.384</td>
<td>0.442</td>
<td>0.377</td>
<td>0.366</td>
<td>0.461</td>
<td>0.371</td>
</tr>
<tr>
<td></td>
<td>(0.0034)</td>
<td>(0.0025)</td>
<td>(0.0021)</td>
<td>(0.0035)</td>
<td>(0.0021)</td>
<td>(0.0075)</td>
<td>(0.0030)</td>
</tr>
<tr>
<td>200</td>
<td><strong>0.344</strong></td>
<td>0.372</td>
<td>0.447</td>
<td>0.364</td>
<td>0.359</td>
<td>0.469</td>
<td>0.358</td>
</tr>
<tr>
<td></td>
<td>(0.0026)</td>
<td>(0.0027)</td>
<td>(0.0035)</td>
<td>(0.0025)</td>
<td>(0.0028)</td>
<td>(0.0072)</td>
<td>(0.0029)</td>
</tr>
</tbody>
</table>

The error rate of the strongest classifier in the simulation is boldfaced. Additionally, the standard errors are the figures below the error rates in parenthesis. The classifier labeled Maj is the majority vote combined classifier. In this example, when there are only 50 samples generated from each distribution, the 5-NN classifier is optimal. However, as \( n \) increases the proposed combined classifier (i.e. Opt) starts to outperform all the other classifiers.

Remark. Mojirsheibani and Kong (2016) had two takeaways from their numerical work. The first conclusion was that the combined classifier that consists of several weak performing classifiers may actually outperform the combined classifier that consists of one or more strong classifiers. In other words the optimal combined classifier can better reduce the error rate if it is composed of classifiers of similar power. Secondly, the more classifiers contained within the ensemble the larger the sample must be for the optimal combined classifier to work effectively. One of their proofs can be used to show that if \( n \) (the total number of samples generated from each distribution) is of order \( e^M \) (or larger), then the optimal combined classifier starts to outperform the other classifiers, for the 2-class problem. However, they believed that this was a conservative lower bound on \( n \).

Example 6 The assumptions and parameters used are the same as Example 5 except the ensemble contains a subset of the classifiers used in the example, namely LDA and RF. One of the biggest incentives of reducing the number of classifiers within the combination is that the run time of the simulations are significantly reduced. Another advantage is that proposed combined classifier achieves optimality at smaller sample sizes. The following table summarizes the results of the simulation:

Figure 5.1 represents the results for the scenario where 50 points are generated for each distribution. The horizontal blue line is drawn at the error rate of 2/3. The significance behind this figure is that if one were randomly assigning the classes of new observations, the probability of guessing incorrectly is 2/3. All of the classifiers are well below this line,
Table 5.2: Misclassification and Standard Error Summary

<table>
<thead>
<tr>
<th>$n$</th>
<th>Opt</th>
<th>Maj</th>
<th>LDA</th>
<th>RF</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.385</td>
<td>0.417</td>
<td>0.445</td>
<td>0.389</td>
</tr>
<tr>
<td></td>
<td>(0.0039)</td>
<td>(0.0036)</td>
<td>(0.0036)</td>
<td>(0.0033)</td>
</tr>
<tr>
<td>100</td>
<td>0.371</td>
<td>0.411</td>
<td>0.443</td>
<td>0.374</td>
</tr>
<tr>
<td></td>
<td>(0.0022)</td>
<td>(0.0028)</td>
<td>(0.0034)</td>
<td>(0.0025)</td>
</tr>
<tr>
<td>200</td>
<td>0.353</td>
<td>0.397</td>
<td>0.438</td>
<td>0.355</td>
</tr>
<tr>
<td></td>
<td>(0.0029)</td>
<td>(0.0031)</td>
<td>(0.0037)</td>
<td>(0.0030)</td>
</tr>
</tbody>
</table>

Figure 5.1: Graphical representation of Example 6 for $n = 50$

which implies that every classifier is better than choosing the classes at random. One may also notice that the proposed combined classifier has an error rate less than every other classifier in almost every simulation. This reconciles to the results shown in the table above.

**Example 7** The purpose of this example is to combine weak classifiers, with respect to error rates and track the performance of the proposed combined classifier. Once again, the assumptions and parameters used are the same as before, however the classifiers within the ensemble are different. Here the combined classifier has three components, one of which is the LDA classifier. As seen in Table 5.1, LDA’s error rate has one of the highest error rates. Another classifier used in the ensemble is the nearest-neighborhood with $K = 25$. Recall from Theorem 3.4 that two conditions for the strong consistency of the $K$-NN rule is that $k/n \to 0$ and $k \to \infty$. Thus, selecting a large $K$ such as 25 for small sample sizes will cause this classifier to be less desirable. The last classifier, referred to as Faulty, will assign a new observation, $W$, to the Cauchy class if $||X - W|| \leq 3$. If $||X - W|| > 3$, then this classifier will randomly choose between Norm1 and Norm2. The intuition behind the weakness of this classifier is that one expects the more extreme observations to be from
the Cauchy class. The table below summarizes the results of this simulation:

Table 5.3: Misclassification and Standard Error Summary

<table>
<thead>
<tr>
<th>$n$</th>
<th>Opt</th>
<th>Maj</th>
<th>LDA</th>
<th>25-NN</th>
<th>Faulty</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.421</td>
<td>0.455</td>
<td>0.442</td>
<td>0.457</td>
<td>0.719</td>
</tr>
<tr>
<td></td>
<td>(0.0091)</td>
<td>(0.0036)</td>
<td>(0.0036)</td>
<td>(0.0032)</td>
<td>(0.0029)</td>
</tr>
<tr>
<td>100</td>
<td>0.411</td>
<td>0.438</td>
<td>0.444</td>
<td>0.427</td>
<td>0.711</td>
</tr>
<tr>
<td></td>
<td>(0.0095)</td>
<td>(0.0023)</td>
<td>(0.0035)</td>
<td>(0.0020)</td>
<td>(0.0020)</td>
</tr>
<tr>
<td>200</td>
<td>0.406</td>
<td>0.427</td>
<td>0.436</td>
<td>0.421</td>
<td>0.716</td>
</tr>
<tr>
<td></td>
<td>(0.0091)</td>
<td>(0.0023)</td>
<td>(0.0037)</td>
<td>(0.0022)</td>
<td>(0.0022)</td>
</tr>
</tbody>
</table>

The proposed combined classifier is clearly superior across all sample sizes. The main revelation of this example is that although the individual classifiers within the ensemble are very weak, the proposed combined classifier performs significantly better. Furthermore, an additional tidbit may be found by examining the graph of the results in Figure 5.2. Once more, the results of the table reconcile with the graph as the points representing the proposed combined classifier are clearly the lowest. However, notice how the points representing the Faulty classifier are above the blue line. This implies that guessing the classes of the new observations randomly has a lower error rate than the aforementioned method!

![Figure 5.2: Graphical representation of Example 7 for $n = 100$](image)

**Example 8** The objective of this example is to examine the performance of the proposed combined classifier when one of the distributions are changed. The Norm1 distribution is replaced with a standard Lognormal distribution. In addition, the mean of the Norm2 distribution is multiplied by $-1$. The ensemble contains the same individual classifier as Example 6. The following tables contains the results of the simulation:
Table 5.4: Misclassification and Standard Error Summary

<table>
<thead>
<tr>
<th>$n$</th>
<th>Opt</th>
<th>Maj</th>
<th>LDA</th>
<th>RF</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.303</td>
<td>0.343</td>
<td>0.381</td>
<td>0.305</td>
</tr>
<tr>
<td></td>
<td>(0.0018)</td>
<td>(0.0020)</td>
<td>(0.0025)</td>
<td>(0.0018)</td>
</tr>
<tr>
<td>100</td>
<td>0.290</td>
<td>0.334</td>
<td>0.380</td>
<td>0.292</td>
</tr>
<tr>
<td></td>
<td>(0.0023)</td>
<td>(0.0027)</td>
<td>(0.0027)</td>
<td>(0.0023)</td>
</tr>
<tr>
<td>200</td>
<td>0.275</td>
<td>0.330</td>
<td>0.380</td>
<td>0.278</td>
</tr>
<tr>
<td></td>
<td>(0.0016)</td>
<td>(0.0018)</td>
<td>(0.0021)</td>
<td>(0.0016)</td>
</tr>
</tbody>
</table>

Figure 5.3: Graphical representation of Example 8 for $n = 50$

Once again, the proposed combined classifier is superior to the other classifiers. An interesting tidbit from this example is that the misclassification error rates are lower than the other simulation examples. The reason for this is that the three distributions generated are more heterogeneous. Not only are we using a different distribution (Lognormal) but also changed the mean of Norm2 to be negative. The observations of the classes are more clearly defined, which lowers the misclassification error rates.
Chapter 4 introduced the concept of combining the information of several $K$-class classifiers. Subsequently, the construction of the asymptotically optimal combined classifier was presented. Under two conditions, it was shown that this classifier is theoretically optimal. One potential area for future research is to examine the situation when these two assumptions are relaxed. This will yield a stronger theoretical result.

The numerical work in Chapter 5 justified the aforementioned theoretical results. After a large number of iterations it was shown that the proposed combined classifier outperformed the other classifiers. Recall that the misclassification error rate of this classifier is an unconditional probability. Instead, if this error rate was conditional on the data it is possible that the proposed combined classifier will begin outperforming the rest of the classifiers with fewer iterations. Therefore, this is yet another area to research in the future that could potentially yield stronger numerical results.
References


## Bankruptcy Data Table

<table>
<thead>
<tr>
<th>X1</th>
<th>X2</th>
<th>Class</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.17</td>
<td>0.07</td>
<td>NonBankrupt</td>
<td>0.0010</td>
</tr>
<tr>
<td>0.19</td>
<td>0.05</td>
<td>NonBankrupt</td>
<td>0.0010</td>
</tr>
<tr>
<td>0.16</td>
<td>0.05</td>
<td>NonBankrupt</td>
<td>0.0025</td>
</tr>
<tr>
<td>0.15</td>
<td>0.06</td>
<td>NonBankrupt</td>
<td>0.0029</td>
</tr>
<tr>
<td>0.15</td>
<td>0.05</td>
<td>NonBankrupt</td>
<td>0.0034</td>
</tr>
<tr>
<td>0.15</td>
<td>0.05</td>
<td>Bankrupt</td>
<td>0.0034</td>
</tr>
<tr>
<td>0.14</td>
<td>0.07</td>
<td>NonBankrupt</td>
<td>0.0037</td>
</tr>
<tr>
<td>0.12</td>
<td>0.11</td>
<td>Bankrupt</td>
<td>0.0073</td>
</tr>
<tr>
<td>0.29</td>
<td>0.06</td>
<td>NonBankrupt</td>
<td>0.0085</td>
</tr>
<tr>
<td>0.32</td>
<td>0.07</td>
<td>NonBankrupt</td>
<td>0.0145</td>
</tr>
<tr>
<td>0.14</td>
<td>-0.03</td>
<td>NonBankrupt</td>
<td>0.0157</td>
</tr>
<tr>
<td>0.07</td>
<td>0.02</td>
<td>Bankrupt</td>
<td>0.0205</td>
</tr>
<tr>
<td>0.07</td>
<td>-0.01</td>
<td>Bankrupt</td>
<td>0.0250</td>
</tr>
<tr>
<td>0.37</td>
<td>0.11</td>
<td>Bankrupt</td>
<td>0.0298</td>
</tr>
<tr>
<td>0.01</td>
<td>0.00</td>
<td>Bankrupt</td>
<td>0.0425</td>
</tr>
<tr>
<td>0.01</td>
<td>0.00</td>
<td>Bankrupt</td>
<td>0.0425</td>
</tr>
<tr>
<td>-0.02</td>
<td>0.02</td>
<td>NonBankrupt</td>
<td>0.0520</td>
</tr>
<tr>
<td>0.48</td>
<td>0.09</td>
<td>NonBankrupt</td>
<td>0.0785</td>
</tr>
<tr>
<td>-0.07</td>
<td>-0.06</td>
<td>Bankrupt</td>
<td>0.0925</td>
</tr>
<tr>
<td>-0.10</td>
<td>-0.01</td>
<td>NonBankrupt</td>
<td>0.0981</td>
</tr>
<tr>
<td>-0.07</td>
<td>-0.09</td>
<td>Bankrupt</td>
<td>0.1018</td>
</tr>
<tr>
<td>-0.08</td>
<td>-0.08</td>
<td>Bankrupt</td>
<td>0.1040</td>
</tr>
<tr>
<td>-0.10</td>
<td>-0.09</td>
<td>Bankrupt</td>
<td>0.1189</td>
</tr>
<tr>
<td>-0.14</td>
<td>-0.14</td>
<td>Bankrupt</td>
<td>0.1640</td>
</tr>
</tbody>
</table>