IMBALANCED HIGH DIMENSIONAL CLASSIFICATION AND APPLICATIONS IN PRECISION MEDICINE

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Hui Sun

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

May 2019

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Lingsong Zhang, Co-Chair
Department of Statistics
Dr. Bruce A. Craig, Co-Chair
Department of Statistics
Dr. Arman Sabbaghi
Department of Statistics
Dr. Yu (Michael) Zhu
Department of Statistics

Approved by:

Dr. Jun Xie
Graduate chair of the Department of Statistics

ACKNOWLEDGMENTS

My time as a Ph.D. student at Purdue was a privilege. I feel so fortunate to have interacted with so many talented students, accomplished faculty, and dedicated staff. I wish to highlight and thank several who had a substantial impact on my education.

First and foremost, I would like to thank my two co-advisers, Professor Lingsong Zhang and Professor Bruce Craig. Professor Zhang helped guide me in finding this exciting research topic. Both Professor Zhang and Professor Craig helped instill and develop my creative thinking and problem solving skills during this research. In addition, they taught me better ways to balance time and handle different types of jobs during my time here. This thesis would not be possible without their dual mentorship and guidance.

I would also like to thank my committee members: Professor Arman Sabbaghi and Professor Michael Zhu. Dr. Sabbaghi very generously allowed me to attend and present my research at his group meetings. His comments and suggestions challenged me to think about my research from a different perspective, and improve my ability to present my work to people who do not have prior knowledge of my research area. Michael Zhu taught me the meaning of Statistics and how to be a qualified Ph.D. student.

I would like to thank Professor Hao Zhang as the department head and Professor Jun Xie as the director of the graduate program for ensuring that I was always supported and had ample funding opportunities presented to me. I also want to thank the dedicated department staff. Doug Crabill, who was such a valuable resource for any technological queries I posed. Mary Sigman, Patti Foster, Holly Graef, and Jesse Wallenfang were always so eager to assist with any request I had.

I would like to thank my fellow students for their generous suggestions and discussions, to help me to better understand and interpret my research.

Last but not the least, I want to acknowledge my husband, my parents, and my brother for their love and support throughout this whole process.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

ABBREVIATIONS

DWD	Distance Weighted Discrimination
HDLSS	High Dimension, Low Sample Size
ITR	Individualized Treatment Rule
MDWD	Multicategory Angle-based Distance Weighted Discrimination
MSVM	Multicategory Angle-based Support Vector Machine
OWL	Outcome Weighted Learning

ABSTRACT

Sun, Hui PhD, Purdue University, May 2019. Imbalanced high dimensional classification and applications in precision medicine. Major Professors: Lingsong Zhang, and Bruce A. Craig.

Classification is an important supervised learning technique with numerous applications. This dissertation addresses two research problems in this area. The first is multicategory classification methods for high dimensional data. To handle high dimension low sample size (HDLSS) data with uneven group sizes (i.e., imbalanced data), we develop a new classification method called angle-based multicategory distance-weighted support vector machine (MDWSVM). It is motivated from its binary counterpart and has the merits of both the support vector machine (SVM) and distance-weighted discrimination (DWD) methods while alleviating both the data piling issue of SVM and the imbalanced data issue of DWD. Theoretical results and numerical studies are used to demonstrate the advantages of our MDWSVM method over existing methods.

The second part of the dissertation is on the application of classification methods to precision medicine problems. Because one-stage precision medicine problems can be reformulated as weighted classification problems, the subtle differences between classification methods may lead to different application performances under this setting. Among the margin-based classification methods, we propose to use the distance weighted discrimination outcome weighted learning (DWD-OWL) method. We also extend the model to handle negative rewards for better generality and apply the angle-based idea to handle multiple treatments. The proofs of Fisher consistency for DWD-OWL in both the binary and multicategory cases are provided. Under mild conditions, the insensitivity of DWD-OWL for imbalanced setting is also demonstrated.

1. INTRODUCTION

This dissertation contains two research topics. The first one is on high dimensional multiclass classification (Chapter 3), and the second one is on the application of classification methods to single-stage precision medicine problems (Chapter 4). The first two sections of this chapter provide some general background on these problems and our motivations for addressing these problems. We conclude the chapter with a summary of our research and an outline of the remaining dissertation.

1.1 Background

Classification is an important type of supervised machine learning that is used broadly in areas such as facial recognition [Fathi and Mori, 2008, Kumar et al., 2009], phishing emails detection [Fette et al., 2007], and disease and cancer identification [Al-Hajj et al., 2003, Di Natale et al., 2003]. The goal of classification is to determine a classification rule, or classifier, that can be applied to future observations. This is achieved by using the information from a training data set, where the observations have known category labels. Classification problems are termed binary, when there are only two categories, and multicategory, when there are more than two categories.

For binary classification, numerous methods have been developed. Some of the methods build a linear classifier, like Fisher discriminant analysis [Fisher, 1936, Mika et al., 1999], naive Bayesian classification [Jiang et al., 2019, John and Langley, 1995, Zhang, 2004], logistic regression [Hosmer Jr et al., 2013, Menard, 2002], and support vector machine (SVM) [Cortes and Vapnik, 1995, Cristianini et al., 2000]. Other methods consider a non-linear classifier, such as decision trees [Myles et al., 2004, Rokach and Maimon, 2008], kernel SVM [Hu et al., 2009, Osuna et al., 1997] and neural networks [Haykin, 1994, Haykin et al., 2009]. In general, these methods build a single classifier. Other methods, however, are a combination, or ensemble, of multiple classifiers, like random forests [Liaw et al., 2002, Pal, 2005] and boosting [Friedman, 2002, Schapire, 2003].

For multicategory classification, one common approach is to transform the problem back to a set of binary classification problems using one-versus-one (OvO) or one-versus-rest (OvR) classifications. We will discuss these in more detail in Section 2.2. Alternatively, one can take a direct approach and consider all the categories at once, like multinomial logistic regression [Böhning, 1992], multiclass SVM [Hsu and Lin, 2002], and random forests [Liaw et al., 2002].

A classification method is called unweighted if all observations are treated equally when determining the classifier. In other words, the misclassification of each observation carries the same weight. Weighted classification, on the other hand, involves weighting the misclassification of some observations more heavily than others. It is traditionally used to handle imbalanced group sizes [Frank et al., 2002, Huang and Du, 2005, Qiao et al., 2010].

One very active application area of weighted classification is single-stage precision medicine, where the goal is to determine the optimal individualized treatment rule (ITR) given subject-specific covariates. The motivation for determining such a rule is the well-established fact that patients' responses to treatments are heterogeneous. For example, one patient who severely twists an ankle might only need ice and a compression bandage to be completely healed, while someone else with the same injury may also need nonsteroidal anti-inflammatory drugs. Reasons for this difference may be due to the subjects' cytokine response, age, sex, and body mass index. The goal for this type of single-stage treatment problem is to predict the best ITR for a new patient based on a classifier developed from training data, consisting of patients with various patient-level covariate information, their assigned treatment, and the observed outcome or reward (i.e., the response after taking the assigned treatment).

Zhao et al. [2012] proposed the outcome weighted learning (OWL) framework to obtain the ITR. In a randomized experimental setting, OWL is a weighted classification method with the outcome serving as the weight. Numerous classification methods have been proposed using OWL with SVM [Liu et al., 2016, Zhang et al., 2018, Zhao et al., 2012, Zhou et al., 2017], OWL with tree based methods [Laber and Zhao, 2015, Sies and Van Mechelen, 2017, Tao et al., 2018, Zhang et al., 2012], and OWL with neural network methods [Liang et al., 2018].

1.2 Research Problems and their Motivation

With increased efficiency in information gathering, data storage, global internet communication, and computing systems, more and more data can now be collected and analyzed. In terms of classification, this has resulted in an explosion of potential explanatory variables. When the number of explanatory variables far exceeds the number of cases, we encounter high dimensional low sample size (HDLSS) data [Lee et al., 2013, Marron et al., 2007, Qiao and Zhang, 2015a,b]. It is very common in applications such as micro-array analysis [Brown et al., 2000], chemometrics [Kowalkowski et al., 2006], and sentiment analysis [Pak and Paroubek, 2010].

For HDLSS data, the inverse of the covariance matrix cannot be directly calculated, so traditional methods, like linear discriminant analysis (LDA), do not work. Classification methods utilizing variable selection are commonly used to get around this issue [Dodge, 2012]. With variable selection, however, the estimation of the classifier doesn't always converge to the Bayesian one [Zhao and Yu, 2006]. Since SVM doesn't involve covariance estimation, it can be used without variable selection. However, it suffers from a "data piling" issue, which is a sign of overfitting. We discuss this in more detail in Section 2.1.2.

To solve this "data piling" issue, Marron et al. [2007] proposed a binary Distance Weighted Discrimination (DWD) method. This classification method considers the distance of each observation to the separating hyperplane, and the aim is to maximize these distances (see Section 2.1.3 for more discussion). However, DWD is sensitive to data with uneven group sizes [Qiao and Liu, 2009]. To alleviate this sensitivity and still handle HDLSS data well, Qiao and Zhang [2015a] proposed the binary Distance Weighted Support Vector Machine (DWSVM) method. This method can be viewed as a hybrid of SVM and DWD, and has better performance, in terms of misclassification error, than either SVM or DWD. This method is described in more detail in Section 2.1.4.

The first project of this dissertation is to develop a multicategory classification approach that can handle HDLSS and imbalanced group sizes, similar to the binary DWSVM. Multicategory classification is of great interest in practice [Cai et al., 2011, Hsu and Lin, 2002, Lee et al., 2004, Liu and Shen, 2006]. For example, to predict the emotions of a picture, one might consider multiple categories such as happy, neutral, sad, and angry. Under an angle-based framework, multicategory angle-based SVM (MSVM) and DWD (MDWD) were proposed by Zhang and Liu [2014]. However, we show that these methods suffer the same limitations as their binary counterparts and that a hybrid approach is again warranted.

The second project is the application of classification methods to single-stage precision medicine problems. As reviewed in Section 1.1, identifying the optimal ITR can be cast as a weighted classification problem, where the weights are a function of the reward of the treatment. Driven by the limitations of SVM and DWD in the unweighted classification context, we investigate whether the popular ITR methods will suffer the HDLSS and imbalanced issues. We propose to use the DWD-based ITR method and investigate the theoretical and empirical performance of the DWD-based method.

1.3 Summary of the Main Results

In this dissertation, we propose a new multicategory classification method called Multicategory Distance Weighted Support Vector Machine (MDWSVM) to handle HDLSS data with imbalanced groups. The method can be viewed as a weighted hybrid of Multicategory Support Vector Machine (MSVM) and Multicategory Distance weighted Discrimination (MDWD). We show the proposed MDWSVM method has smaller misclassification error relative to both MDWD and MSVM. In addition, it is closer to the Bayes discriminant direction/optimal decision rule compared to that of MSVM. We prove the Fisher consistency of MDWSVM, as well as demonstrate its insensitivity to imbalanced data.

For the ITR project, we extend our study of classification to finding the optimal ITR in single-stage precision medicine problems. We compare several methods and recommend the binary DWD-OWL method be used in this setting. We also recommend "mirror" projection so that our method can handle outcome rewards that fall on the real line (i.e., handle both positive and negative rewards). Furthermore, we extend this method to the multicategory setting. Through simulation, we show that the DWD-OWL method better handles imbalanced optimal treatment problems compared to SVM-OWL. We show the consistency of our DWD-OWL method under both binary and multicategory treatment settings. In the binary setting, we also show that the excess risk of our DWD-OWL is bounded. And under mild conditions on the outcome rewards, our method demonstrates the desired property that even though one treatment may benefit more patients than the other, our DWD-OWL will not predict that treatment for all patients.

1.4 Structure of the Dissertation

The remainder of the dissertation is organized as follows. In Chapter 2, we review the commonly-used classification methods and their limitations in the HDLSS and imbalanced multicategory groups settings. In Chapter 3, our novel multicategory Distance Weighted Support Vector Machine (MDWSVM) is described. The superiority of this method over the existing Multicategory SVM (MSVM) and multicategory distance weighted discrimination (MDWD) is also demonstrated. Finally, the Fisher consistency and imbalance properties are presented along with a real data application.

In Chapter 4, the current ITR methods are reviewed, and the DWD-OWL method in both the binary and multicategory treatment settings is detailed. Through simulation, we demonstrate that our method outperforms the SVM-OWL method in the imbalanced setting. Some theoretical properties of our method are also demonstrated. An overall summary and future research directions are discussed in Chapter 5.

2. LITERATURE REVIEW

In this chapter, we review some existing methods for both binary classification (Section 2.1) and multicategory classification (Section 2.2). We conclude the chapter with a discussion on the use of outcome-weighted learning (OWL) to estimate the individualized treatment rule (ITR) in a single-stage precision medicine problem.

2.1 Binary Classification Methods

In this section, we provide detail on the common binary classification methods, as well as highlight their limitations when dealing with HDLSS data and imbalanced group sizes.

2.1.1 Binary classification and choice of loss function

Consider a binary classification problem with observed data $(\boldsymbol{x}_i, y_i), i = 1, \ldots, n$. The $\boldsymbol{x}_i \in \mathbb{R}^d$ is a multivariate predictor and the scalar $y_i \in \{1, -1\}$ is the corresponding class label. The goal is to find a decision function (or classifier) f such that its prediction $\hat{y}(\boldsymbol{x}) = \text{sign}(f(\boldsymbol{x}))$ minimizes the misclassification error $E(\hat{Y} \neq Y)$. Note that $yf(\boldsymbol{x}) > 0$ if and only if $f(\boldsymbol{x})$ gives a correct prediction. Thus the misclassification error can be expressed

$$E(\hat{Y} \neq Y) = E(Yf(X) < 0)$$

A natural way to estimate the misclassification error is to use the empirical error

$$1/n \sum \mathbb{I}(\hat{y}_i(\boldsymbol{x}) \neq y_i) = 1/n \sum \mathbb{I}(y_i f(\boldsymbol{x}_i) < 0), \tag{2.1}$$

where $\mathbb{I}(.)$ is the indicator function.

However, due to the discontinuity and nonconvexity of $\mathbb{I}(y_i f(\boldsymbol{x}_i) < 0)$, directly using (2.1) to determine f is computationally inefficient. A common surrogate for the 0-1 loss is a loss function $\ell(.)$ satisfying $\ell(-u) > \ell(u)$ for $u > 0$ and $\ell(0)' \neq 0$ [Lin, 2004]. A variety of commonly used loss functions satisfy this condition, including logistic loss [Kleinbaum et al., 2002]: $\ell(u) = ln(1 + exp(-u))$, and hinge loss [Cortes and Vapnik, 1995]: $\ell(u) = (1 - u)_+$ where $u_+ = u$ if $u > 0$ and 0 otherwise. These two loss functions, along with 0-1 loss and square-error loss, are shown in Figure 2.1. Among these loss functions, square-error loss is commonly used for regression when the responses are continuous. Note that these loss function are all “good” surrogates based on the conditions of Lin [2004].

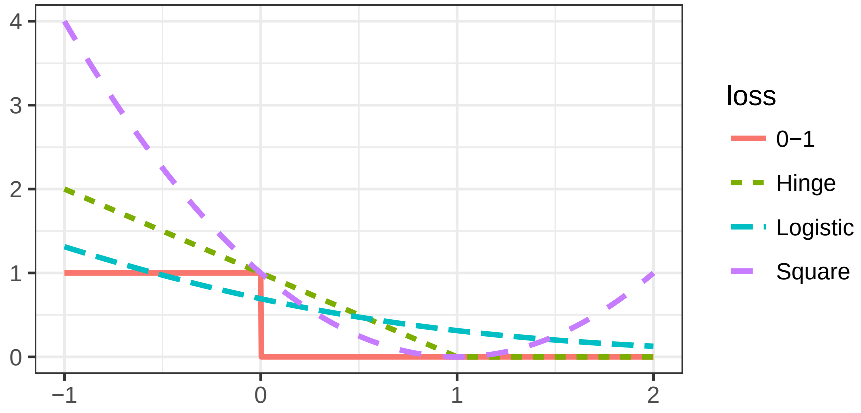

Figure 2.1. Some commonly used loss functions in binary classification.

An optimization problem to find the classifier can be formulated as minimizing the loss function given a constraint:

$$\min_{f \in \mathbb{F}} \quad \sum_{i=1}^{n} \ell(f(\boldsymbol{x}_i), y_i) + \frac{\lambda}{2} J(f) \tag{2.2}$$

Here $\mathbb{F}$ denotes the function space and $J(.)$ is a type of norm used to control the complexity of the method. The function $\ell(f, y)$ represents a loss function surrogate for the 0-1 loss. The tuning parameter λ balances the loss and the norm. For example, the popular linear SVM method uses the hinge loss function $\ell_S(u) = (1-u)_+$ with $u = yf(\boldsymbol{x})$, and $J(.)$ is the L_2 norm. The details of linear SVM classification method are given in the next section.

2.1.2 Support Vector Machine (SVM)

The empirical minimization of hinge loss with L_2 norm is equivalent to the formulation for Support Vector Machine (SVM), which is commonly used method in the classification framework.

Define a hyperplane as a set of points satisfying

$$\boldsymbol{x}^T\boldsymbol{\omega} + \beta = 0.$$

where $\boldsymbol{\omega}$ is a normal vector to this hyperplane. The parameter $\frac{\beta}{\|\boldsymbol{\omega}\|}$ determines the offset of the hyperplane to the origin along the normal vector $\boldsymbol{\omega}$.

If the two groups are linearly separable, one can find two parallel hyperplanes to separate the groups, such that the distance between these two hyperplanes are as far apart as possible. The distance between these two parallel hyperplanes is called the *margin*. The goal of SVM is to find the maximum-margin hyperplane such that the group of points with response $y = -1$ is separated from the group of points with response $y = 1$.

Figure 2.2 shows a maximum-margin hyperplane for a SVM trained with samples from two groups in a linearly separable case. In Figure 2.2, note that the maximum hyperplane is parallel to the two separating hyperplanes and falls in the middle of them. With standardized data, the two parallel hyperplanes are described using the equations

$$\boldsymbol{x}^T\boldsymbol{\omega} + \beta = -1$$

and

$$\boldsymbol{x}^T\boldsymbol{\omega} + \beta = 1.$$

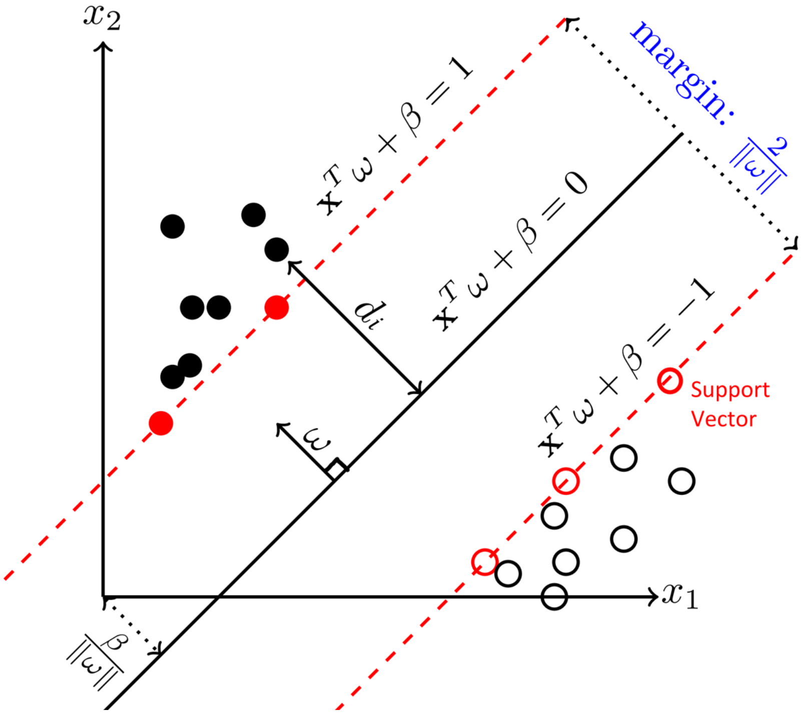
x_2
x_1
margin: $\frac{2}{\|\omega\|}$
$\mathbf{x}^T\omega + \beta = 1$
$\mathbf{x}^T\omega + \beta = 0$
$\mathbf{x}^T\omega + \beta = -1$
d_i
ω
Support Vector
$\frac{\beta}{\|\omega\|}$

Figure 2.2. Geometric margin and maximum-margin hyperplane. The solid line separating two groups (represented by solid and empty circles) is determined by the normal vector ω, which is perpendicular to the hyperplane, and the intercept β, which determines the location of the hyperplane with respect to the origin.

For the points of Group 1, they satisfy $\boldsymbol{x}^T\boldsymbol{\omega} + \beta \geq 1$, and for the points of Group -1, they satisfy $\boldsymbol{x}^T\boldsymbol{\omega} + \beta \leq -1$. Notice that this implies $(\boldsymbol{x}^T\boldsymbol{\omega} + \beta)y \geq 1$ for all points.

The goal is to maximize the distance between the two parallel hyperplanes, which is presented by $2/\left\|\boldsymbol{\omega}\right\|$. This is equivalent to minimizing $\|\boldsymbol{\omega}\|$. Therefore, for linearly separable groups, SVM solves the optimization problem

$$\min \|\boldsymbol{\omega}\|^2$$

$$\text{subject to } (\boldsymbol{x}_i^T\boldsymbol{\omega} + \beta)y_i \geq 1 \text{ for } i = 1, 2, \ldots, n.$$

The solution to this optimization problem, the maximum-margin hyperplane, is determined only by the points that falls on the parallel hyperplanes, which are called *support vectors*. They satisfy $(\boldsymbol{x}^T\boldsymbol{\omega} + \beta)y = 1$.

When the two groups are not linearly separable, the constraint is softened to

$$(\boldsymbol{x}_i^T\boldsymbol{\omega} + \beta)y_i \geq 1 - \xi_i, \text{ with all } \xi_i \geq 0.$$

The ξ_i, also known as penalties, allow each point to fall inside the margin or on the wrong side of the separating hyperplane. If a point $\boldsymbol{x}_i$ falls outside the parallel hyperplane and on the correct side, then $\xi_i = 0$. If a point $\boldsymbol{x}_i$ falls inside the margin but on the right side, then $0 < \xi_i \leq 1$. If a point $\boldsymbol{x}_i$ falls on the wrong side of the separating hyperplane, then $\xi_i > 1$. In general, it is undesirable to have too many points to fall on the wrong side of the separating hyperplane. Therefore a goal is to minimize the sum of the penalties. At the same time, another goal is to maximize the margin (minimize $\|\boldsymbol{\omega}\|$). Thus the optimization becomes

$$\min \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + C\sum_{i=1}^{n}\xi_i$$

$$\text{subject to } (\boldsymbol{x}_i^T\boldsymbol{\omega} + \beta)y_i \geq 1 - \xi_i, \text{ and } \xi_i \geq 0.$$

The parameter C controls the misclassification rate. Notice that

$$\xi_i = \max(0, 1 - (\boldsymbol{x}_i^T\boldsymbol{\omega} + \beta)y_i) = (1 - (\boldsymbol{x}_i^T\boldsymbol{\omega} + \beta)y_i)_+$$

where $(u)_+ = u$ if $u \geq 0$ and 0 otherwise. Therefore, the optimization problem becomes

$$\min C\sum_{i=1}^{n}(1 - (\boldsymbol{x}_i^T\boldsymbol{\omega} + \beta)y_i)_+ + \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega}.$$

Since $f(\boldsymbol{x}_i) = \boldsymbol{x}_i^T\boldsymbol{\omega} + \beta$, the optimization can be viewed as a special case of Equation (2.2) with hinge loss $\ell(f(\boldsymbol{x}), y) = (1 - yf(\boldsymbol{x}))_+$ and L_2 regularization where $J(f) = \boldsymbol{\omega}^T\boldsymbol{\omega}$ [Hastie et al., 2001].

Marron et al. [2007] investigated the performance of SVM on HDLSS data and found that SVM suffers from a *data piling* issue. Data piling refers to the fact that many of the projections to the normal vector are the same. For SVM, data piling means that the support vectors, which tend to be numerous in the HDLSS setting, all pile up at the boundaries of the margin when projected. A toy example is given in Marron et al. [2007] and presented in Figure 2.3 to illustrate this phenomenon.

In the toy example, 20 d-dimensional vectors are randomly generated from both $N(\mu e_1, I_d)$ and $N(-\mu e_1, I_d)$, where $\mu = 2.2$. The vector e_1 is a d-dimensional unit vector where only the first element is 1 and the rest are 0's. Marron et al. [2007] point out that data piling begins as d approaches $n-1$, which is pictured in Figure 2.3 (i.e., $d = 39$).

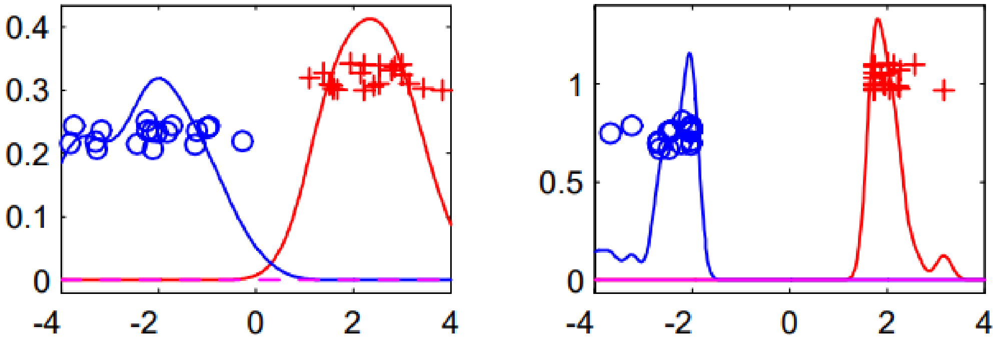

Figure 2.3. Toy example illustrating the data piling of SVM in HDLSS. The blue circled points are from Group 1 and the red cross points are from Group -1. The left panel is a projection of all the points onto the Bayesian normal vector, and the right panel is the projection of all the points onto the SVM normal vector. The smooth curves are estimated densities. This figure is from Marron et al. [2007].

In Figure 2.3, the left panel is the projection of the points to the direction $(1, 0, \ldots, 0)$, which is also the theoretical Bayes direction as the population group difference comes only from the first element. A classification method with normal vector close to this one should perform well. The second panel shows the points projected on the SVM direction. In each panel, the blue circle points belong to Group 1 and the red cross points belong to Group -1. In both panels, the data are jittered vertically for easier viewing. The smooth curves in each plot are the estimated densities for each group. They give some indication of the structure of the unerline population. As expected, the left panel reveals two Gaussian populations, with means of 2.2 and -2.2, respectively. The right panel shows that almost all the points line up in a direction that is orthogonal to the normal vector calculated by SVM.

Data piling is a problem because the corresponding separating hyperplane is highly driven by the particular realization of the training data, i.e., it overfits the data. In the HDLSS setting, it's possible that SVM analyses using two different training sets from the same population will result in completely different classifiers.

2.1.3 Distance weighted Discrimination (DWD)

Data piling is caused by the fact that SVM only considers the support vectors when determining the separating hyperplanes, thereby ignoring the points that are correctly separated. To alleviate the data piling issue under HDLSS setting, all the points should be considered when determining the separating hyperplane. This idea is incorporated into the Distance Weighted Discrimination (DWD) method proposed by Marron et al. [2007].

Define

$$d_i = (\boldsymbol{x}_i^T \boldsymbol{\omega} + \beta) y_i.$$

From Figure 2.2, one can see that the d_i is the signed distance from a single point $\boldsymbol{x}_i$ to the separating hyperplane. The larger a d_i, the farther away the associated point is from the separating hyperplane. SVM requires $d_i \geq 1$ and tries to maximize the distance between the two parallel hyperplanes with $d_i = 1$. For the DWD method, the goal is to no longer maximize the margins, but rather maximize these signed distance. Because $d_i = (\boldsymbol{x}_i^T\boldsymbol{\omega} + \beta)y_i$, one way to increase d_i is to simply increase the scale of $\boldsymbol{\omega}$ and β. To avoid this scaling problem, constraints are imposed on the normal vector $\boldsymbol{\omega}$.

In a linear separable setting, the DWD optimization can be expressed as

$$\begin{aligned} &\min \sum_{i=1}^{n} \frac{1}{d_i} \\ &\text{Subject to } d_i = (\boldsymbol{x}^T\boldsymbol{\omega} + \beta)y, d_i \geq 0 \\ &\text{and } \boldsymbol{\omega}^T\boldsymbol{\omega} \leq 1 \end{aligned}$$

Essentially, DWD minimizes the mean of the inverse distance of all data vectors to the separating hyperplane. In a non-linearly separable setting, a penalty vector η_i is added to each point to allow for misclassification. The corresponding DWD optimization method is then

$$\begin{aligned} &\min \sum_{i=1}^{n} \frac{1}{d_i} + C\sum_{i=1}^{n} \eta_i \\ &\text{subject to } d_i = (\boldsymbol{x}^T\boldsymbol{\omega} + \beta)y + \eta_i, d_i \geq 0, \eta_i \geq 0 \\ &\text{and } \boldsymbol{\omega}^T\boldsymbol{\omega} \leq 1 \end{aligned} \tag{2.3}$$

where C is used to control the misclassification rate. When all $\eta_i = 0$, the optimization simplifies to the linearly separable case.

After some algebraic manipulation, Qiao and Zhang [2015a] point out that the DWD optimization method described in (2.3) is equivalent to minimizing the loss function

$$\min_{f \in \mathbb{F}} \sum_{i=1}^{n} \ell_D(y_i \boldsymbol{f}(\boldsymbol{x}_i))$$

with respect to the constraint $\boldsymbol{\omega}^T\boldsymbol{\omega} \leq 1$, where the corresponding loss function is

$$\ell_D(u) = \begin{cases} 2\sqrt{C} - Cu & u \leq 1/\sqrt{C} \\ 1/u & \text{otherwise.} \end{cases} \tag{2.4}$$

The DWD method minimizes the sum of the inverse signal distances. By taking all the points into consideration, DWD better handles the distribution of the data. However, by allowing all the points to have influence on the separating hyperplane, DWD suffers when the two group sizes are imbalanced [Qiao et al., 2010]. In particular, when the sample size of one group is much larger than the other group, the separating hyperplane will be pushed towards the minority group (group with the smaller size) and in consequence, all future points will be classified to the majority group (group with the larger size).

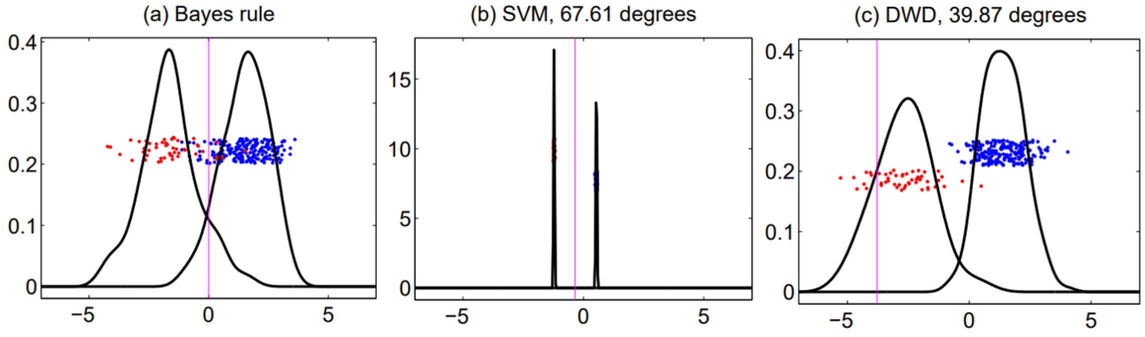

Figure 2.4. Projection plots for illustration of the imbalance issue for DWD method. The panels are the projection plots of data projected to the Bayes normal vector (a), SVM normal vector (b) and DWD normal vector (c). In each panel, the pink line is the separating hyperplane estimated from each method. Red and blues points are points corresponding to two different groups. The plots are presented by the Jitter plot where the vertical coordinated are randomly generated for a better visualization purpose. the black smooth curves are the density curve estimated from the projected points for each of the groups. This figure is from Qiao and Zhang [2015a].

In Qiao and Zhang [2015a], a toy example is used to illustrate this phenomenon. It is shown in Figure 2.4. In the toy example, 200 points were randomly generated from $N_d(u\mathbf{1}_d, I_d)$ and 50 points were randomly generated from $N_d(-u\mathbf{1}_d, I_d)$, where

$d = 300$, $u = 1.35/\sqrt{d} = 0.07794$, $\mathbf{1}_d$ is a d-dimensional vector of ones, and I_d is a $d \times d$ identity matrix. The Bayesian rule for this example has direction $\boldsymbol{\omega} = \mathbf{1}_d/\sqrt{d}$ and intercept $\beta = 0$.

The points in red belong to Group 1 and points in blue belong to Group -1. The pink line is the separating hyperplane. Panel (a) shows the projection of the data to the true mean reference/Bayes direction. It is used as a benchmark for comparison. Panels (b) and (c) are the projections of the data onto the normal vector calculated by SVM and DWD, respectively. Note that for SVM, all the projected points are piled into two points, corresponding to the two groups. It indicates a severe data-piling issue.

The angle between any normal vector $\boldsymbol{\omega}$ and the Bayesian normal vector $\boldsymbol{\omega}_B$ can be calculated as

$$\angle(\boldsymbol{\omega}, \boldsymbol{\omega}_B) = \arccos \frac{\boldsymbol{\omega}^T \boldsymbol{\omega}_B}{\|\boldsymbol{\omega}\| \, \|\boldsymbol{\omega}\|_B}$$

For SVM, the angle between its normal vector and the Bayesian normal vector is 67.61°, which indicates a vector far from ideal. However, because the separating hyperplane is close to 0, the resulting separation into Group 1 and Group -1 are quite good. For DWD, the resulting projections result in two Gaussian distributions. However, the estimation of the location parameter/intercept is shifted towards Group -1, the group with smaller sample size. As a consequence, the method assigns almost all the points to the majority group.

From the toy example, Qiao and Zhang [2015a] conclude that although SVM suffers from data piling under the HDLSS setting, it is not affected by imbalanced group sizes. The estimation of the intercept β is very close to the Bayesian intercept. DWD, on the other hand, handles the HDLSS setting well, but suffers from the imbalanced group sizes, especially if one group has a much larger sample size.

2.1.4 Distance Weighted Support Vector Machine (DWSVM)

Motivated by the accurate estimation of the location parameter using SVM and the normal vector using DWD, Qiao and Zhang [2015a] proposed Distance Weighted Support Vector Machine (DWSVM) to alleviate both data piling and imbalanced data issues in binary classification. It can be viewed as a combination of SVM and DWD. The method has the following form:

$$\begin{aligned} \min_{\boldsymbol{\omega},\beta,\beta_0,\eta_i,\xi_i} \quad & \sum_{i=1}^{n}\{\alpha(\frac{1}{d_i} + C_{dwd}.\eta_i) + (1-\alpha)\xi_i\}, \\ \text{s.t.} \quad & d_i = y_i(x_i^T\boldsymbol{\omega} + \beta_0) + \eta_i, \quad d_i \geq 0 \quad \text{and} \quad \eta_i \geq 0, \\ & C_{svm}y_i(\boldsymbol{x}_i^T\boldsymbol{\omega} + \beta) + \xi_i \geq \sqrt{C_{svm}}, \quad \xi_i \geq 0, \\ & \|\boldsymbol{\omega}\|^2 \leq 1 \end{aligned} \tag{2.5}$$

The parameters $C_{svm} > 0$ and $C_{dwd} > 0$ are used in SVM and DWD, respectively. The parameter $\alpha \in [0, 1)$ is used to control the proportion of DWD and SVM used in the minimization function. When $\alpha = 0$, the method is equivalent to SVM. Notice, however, that the optimization in (2.5) is not a trivial combination of SVM and DWD. There are two intercept/location parameters involved, β and β_0. The estimated separating hyperplane is $\boldsymbol{x}_i^T\hat{\boldsymbol{\omega}} + \hat{\beta} = 0$. For a future point $\boldsymbol{x}$, the prediction function is

$$\hat{y} = \text{sign}(\hat{f}(\boldsymbol{x})) = \text{sign}(\boldsymbol{x}_i^T\hat{\boldsymbol{\omega}} + \hat{\beta}).$$

In other words, it is determined only using the location parameter β estimated from SVM. The intercept/location parameter β_0 estimated from DWD is not used for future prediction.

In binary classification, Qiao and Zhang [2015a] show that by choosing the appropriate α, the DWSVM method results in a smaller misclassification error compared to both DWD and SVM in the HDLSS and imbalanced group context. Furthermore, in terms of the similarity to the Bayesian classifier, DWSVM is similar to DWD and better than SVM.

2.2 Multicategory classification

In multicategory classification, the group labels are $Y \in \{1, 2, \ldots, K\}$. There are two common ways to deal with multicategory classification. The first is to transfer it back to a set of binary classifications, and the second is to extend the binary classification ideas to handle all groups at once.

2.2.1 Transfer to binary

One easy way to transfer the multicategory classification to a set of binary classifications is to use the one vs rest (OvR) method [Bishop, 2006]. OvR trains a single classifier f_k for each group k versus the others. The strategy produces a real value score for each classifier, instead of a label. For a new input $\boldsymbol{x}$, the group classifier f_k with maximum score represents the predicted group. The detailed steps for OvR are outlined below:

```
for $k \leftarrow 1$ to $K$ do
    for $i \leftarrow 1$ to $n$ do
        if $y_i == k$ then
            $y_i^* = 1$;
        else
            $y_i^* =$ -1;
        end
    end
    Calculate binary classifier $\hat{f}_k$ using $(\boldsymbol{x}_i, y_i^*)$, $i = 1, \ldots, n$;
end
$\hat{y} = \arg\max_{1,\ldots,K} \hat{f}_k(\boldsymbol{x})$
```

Though the idea of OvR is easy to implement, it suffers from a variety of problems. First, the scales of $\hat{f}_k$ may be very different, thereby inherently favoring one group over another. Second, when the sample size for each group is the same, combining several groups together results in imbalanced group sizes.

An alternative to OvR is the one versus one (OvO) strategy. In OvO, each pair of groups are extracted to build a classifier. In total, $K(K-1)/2$ classifiers are estimated. For each new observation, one applies all $K(K-1)/2$ classifiers and accumulates the corresponding predicted group labels. The label with highest count wins. For OvO, there can easily be cases where there are ties in the highest count. Furthermore, for a large K, it is computationally expensive to build $K(K-1)/2$ classifiers. For example, when $K = 10$, the number of classifiers needed for OvO is 45.

2.2.2 Extension from binary

Many works have studied the direct extension from binary classification to multicategory classification [Cai et al., 2011, Lee et al., 2004, Liu and Yuan, 2011, Shen et al., 2007, Zhang and Liu, 2013]. A common way is to build a K dimensional function $\boldsymbol{f} = (f_1, f_2, \ldots, f_K)$ is to consider the following algorithm:

$$\begin{aligned} \min_{f \in \mathbb{F}} \quad & \sum_{i=1}^{n} \ell(\boldsymbol{f}(\boldsymbol{x}_i), y_i) + \frac{\lambda}{2} \sum_{j=1}^{K} J(f_j), \\ \text{s.t.} \quad & \sum_{j=1}^{K} f_j(\boldsymbol{x}) = 0. \end{aligned}$$

For example, multicategory SVM [Lee et al., 2004] uses the hinge loss function $\ell(\boldsymbol{f}(\boldsymbol{x}), y) = (1 - f_y(\boldsymbol{x}))_+$. Shen et al. [2007] investigate the generalization error of large margin classifiers in multiclass classification. Liu and Shen [2006] and Liu and Yuan [2011] extended the multicategory SVM into a class of multicategory hinge loss functions.

Unlike binary classification, multicategory classification with a sum to zero constraint does not have a clear geometric explanation. In binary classification, only one classifier is needed. In multicategory classification, it makes sense to have $K-1$, instead of K, classifiers to handle K classes. In addition, the constraint of summing to zero brings along computation difficulty [Zhang and Liu, 2014]. To overcome these limitations, Lange and Wu [2008] proposed the vertex idea where they define $\boldsymbol{f}$ as a $K-1$ dimensional function instead of a K dimensional function. This removes the need for the sum-to-zero constraint. A similar idea was used later by Zhang and Liu [2014] to conduct angle-based classification.

The idea of angle-based classification is to map $\boldsymbol{x}$ to $\boldsymbol{f}(\boldsymbol{x})$, where $\boldsymbol{f} = (f_1, \dots, f_{K-1})$, with a set of K predefined vertices in $\mathbb{R}^{K-1}$. The corresponding label for the vertex with the smallest angle to the projection is the prediction for $\boldsymbol{x}$.

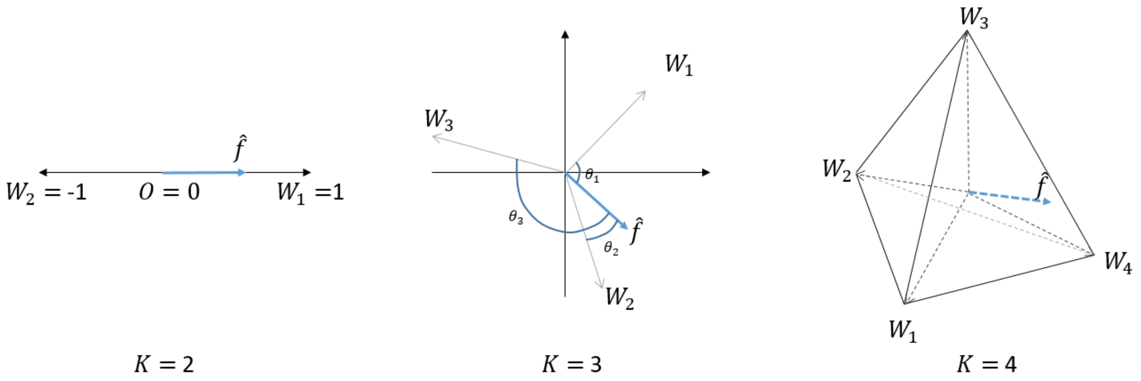

Figure 2.5. Illustration of angle-based classification for $K = 2, 3$, and 4 groups. From left to right, the panels are representing the vertices based on Equation (2.6). The blue vector labeled $\hat{f}$ is the projection of a point $\boldsymbol{x}$ onto the angle-based space in $\mathbb{R}^{K-1}$. The resulting group predictions would be 1, 2, and 4, respectively.

In Zhang and Liu [2014], the vertices $W = (W_1, W_2, \dots, W_K)$ are defined as a collection of K vectors in $\mathbb{R}^{K-1}$ with elements

$$W_j = \begin{cases} (K-1)^{-1/2}\zeta, & j = 1, \\ -(1+K^{1/2})/\{(K-1)^{3/2}\}\zeta + \{K/(K-1)\}^{1/2}e_{j-1}, & 2 \le j \le K. \end{cases} \tag{2.6}$$

where ζ is a unit vector of length $K-1$, and e_j is a vector in $\mathbb{R}^{K-1}$ such that all of its element are 0, except the jth is 1. In this setting, W forms a simplex with K vertices in a $(K-1)$ dimensional space. The center of W is at the origin, and each of the W_j has an Euclidean norm of 1. Furthermore, it is easy to show that the angle between each pair of vertices W_i and W_j, $i \neq j$ is the same. Instead of y_i, W_{y_i} is used to represent the observed class. The prediction function is then $\hat{y} = \arg\max_j \langle W_j, \hat{\boldsymbol{f}} \rangle$, where $\langle ., . \rangle$ denotes the inner product between the two vectors. The larger this inner product, the smaller the angle between $\hat{\boldsymbol{f}}$ and W_j. Figure 2.5 represents the Ws corresponding to the $K = 2, 3$, and 4 settings.

In Figure 2.5, the blue vector is the estimated projection of $\boldsymbol{x}$ onto the angle-based space $\mathbb{R}^{K-1}$. Notice that when $K = 2$, inner product $\langle W_j, \hat{\boldsymbol{f}} \rangle = y\boldsymbol{f}$ simplifies to a binary setting.

For $K = 3$, the W's correspond to the vertices from a regular triangle and $\boldsymbol{x}$ is projected into a 2-dimensional vector $\hat{\boldsymbol{f}}$. In this example, the angle θ_2 between $\hat{\boldsymbol{f}}$ and W_2 is the smallest among all three angles. Therefore Group 2 is the predicted group for this observation. Notice that for $K = 4$, the Ws are corresponding to the vertices from a regular tetrahedron and the predicted group is 4.

With this prediction rule, Zhang and Liu [2014] proposed the following optimization for the angle-based classification:

$$\min_{\boldsymbol{f} \in \mathbb{F}} \frac{1}{n} \sum_{i=1}^{n} \ell(\langle \boldsymbol{f}(\boldsymbol{x}_i), W_{y_i} \rangle) + \frac{\lambda}{2} J(\boldsymbol{f}). \tag{2.7}$$

Here the product $\langle \boldsymbol{f}(\boldsymbol{x}_i), W_{y_i} \rangle$ can be viewed as a new functional margin of $(\boldsymbol{x}, y)$. $J(\boldsymbol{f})$ is used to control the overfitting of the method and λ is its tuning parameter. Because this algorithm is for a general loss function, multicategory SVM (MSVM) and multicategory DWD (MDWD) are two special cases of the angle-based classification method proposed by Zhang and Liu [2014].

Note that Huang et al. [2013] extended the binary DWD to a version of multiclass DWD (MDWDH). It adopts the idea of pairwise comparisons from the one-versus-one idea. MDWDH considers a data point with group label i as being misclassified if the difference of projections on the ith group and the jth group is negative $(i \neq j)$. Even though the approach is shown to have nice theoretical properties and empirical performance that is slightly better than angle-based MDWD, the angle-based methods have better geometric interpretation. In addition, the pair-wise nature of the method will be more computationally expensive. Furthermore, if the angle-based MDWD approach also incorporates the pairwise idea, the two approaches have similar empirical performance.

In multicategory classification, we show that MSVM suffers from data piling in HDLSS setting and MDWD is sensitive to imbalanced group sizes. Motivated by binary DWSVM, we propose multicategory DWSVM (MDWSVM). Details of this approach and its properties are the topic of Chapter 3.

2.3 Determining individualized treatment rules

The goal of a single-stage precision medicine study is to construct an optimal individualized treatment rule (ITR), based on patient-specific covariates, that maximizes the expected clinical outcomes. Such a study can be formulated as a classification problem.

Consider a two-arm clinical trial, or observational study, with underlying distribution $P(X, A, R)$, where $A \in \mathcal{A} = \{-1, 1\}$ is the treatment assigned to a patient, $X = (X_1, X_2, \ldots, X_d) \in \mathbb{R}^d$ is the patient-specific information and $R \in \mathbb{R}^+$ is the outcome response, or reward. The likelihood of (X, A, R) can be expressed as $f_0(X)P(A|X)f_1(R|X, A)$. An Individualized Treatment Rule (ITR) $\mathcal{D}$ [Qian and Murphy, 2011] is defined as a map from the patient information space to the treatment space:

$$\mathcal{D} : X \to \mathcal{D}(\mathcal{X}) \in \{-1, 1\}.$$

Table 2.1.
A toy example for ITR, the grey colored values are the potential outcome which can not be observed. The dark color values are the observed outcome.

ID	X	$P(A\|X)$	$R\|A = 1$	$R\|A = -1$	$\mathcal{D}_1$	$\mathcal{D}_2$
1	1	0.5	1	2	1	-1
2	2	0.5	2	2	1	-1
3	3	0.5	3	2	1	1
4	1	0.5	1	2	1	-1
5	2	0.5	2	2	1	-1
6	3	0.5	3	2	1	1

For an ITR $\mathcal{D}$, we denote the distribution of (X, A, R) by $P^{\mathcal{D}}$ such that the likelihood of (X, A, R) under $\mathcal{D}$ is $f_0(X)\mathbb{I}\{A = \mathcal{D}(X)\}f_1(R|X, A)$. The expected value under $\mathcal{D}$ can be expressed as

$$E^{\mathcal{D}}(R) = \int R dP^{\mathcal{D}} = \int R \frac{\mathbb{I}\{A = \mathcal{D}(X)\}}{P(A|X)} dP = E(\frac{\mathbb{I}\{A = \mathcal{D}(X)\}}{P(A|X)} R).$$

Without loss of generality, we assume larger R represents a better outcome. The goal then is to find the ITR that maximizes the expected value. Therefore, the optimal ITR $\mathcal{D}^*$ is

$$\mathcal{D}^* = \arg\max_{\mathcal{D}} E^{\mathcal{D}}(R) = \arg\max_{\mathcal{D}} E(\frac{\mathbb{I}\{A = \mathcal{D}(X)\}}{P(A|X)} R)$$

Qian and Murphy [2011] point out the optimal decision rule is

$$\mathcal{D}^* = \arg\max_{a} E(R|X, a)$$

In other words, for an ITR with only two treatments $\{-1, 1\}$, the optimal ITR for a patient is determined such that the expected outcome under this treatment is larger than it is under the other treatment. The optimal ITR is explained in a toy example in Table 2.1.

Table 2.1 represents a clinical trial where the treatments are randomly assigned to patients with $P(A=1)=0.5$. Two ITRs $\mathcal{D}_1$ and $\mathcal{D}_2$ are provided, $\mathcal{D}_1 = 1$ always and $\mathcal{D}_2 = 1$ for $X \leq 2$ and -1 otherwise. In Columns 4 and 5, the dark values are the actual rewards of each subject given the assigned treatment, the grey ones are the unobserved rewards had they been assigned the other treatment. For ITR $\mathcal{D}_1$, the empirical expected reward is

$$E_n^{\mathcal{D}_1}(R) = E_n(\frac{\mathbb{I}\{A=\mathcal{D}_1(X)\}}{P(A|X)}R) = \frac{1}{6}(\frac{1+0+3+0+2+0}{0.5}) = 6/3,$$

where $E_n(.)$ is the empirical version of $E(.)$. For ITR $\mathcal{D}_2$, the empirical expected reward is

$$E_n^{\mathcal{D}_2}(R) = E_n(\frac{\mathbb{I}\{A=\mathcal{D}_2(X)\}}{P(A|X)}R) = \frac{1}{6}(\frac{0+2+3+2+0+0}{0.5}) = 7/3.$$

One can conclude that $\mathcal{D}_2$ is the better choice. For subjects with covariate X, the potential rewards are represented in Figure 2.6. It is easy to verify from this figure that $\mathcal{D}_2$ is a better ITR compared to $\mathcal{D}_1$.

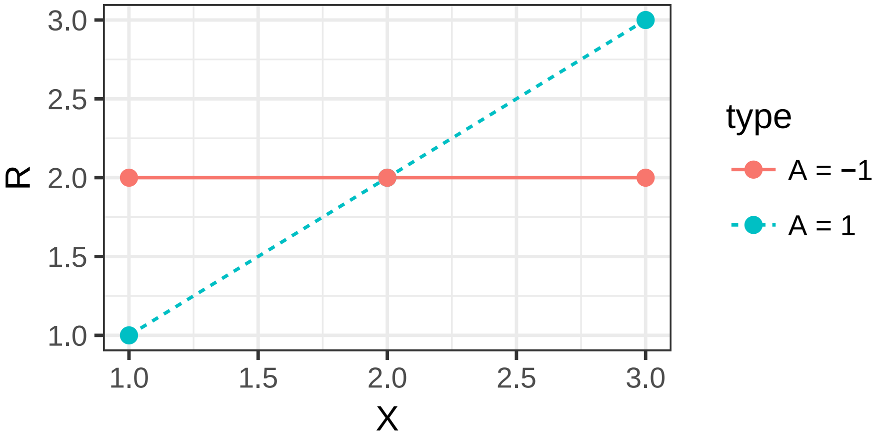

Figure 2.6. A toy example for optimal rule. The red solid line and blue dashed line are the outcome rewards for strategy $\mathcal{D}_1$ and $\mathcal{D}_2$. Note that the optimal regime under this setting is $\mathcal{D}_2$.

There are two main approaches to determining the optimal ITR: the indirect approach and the direct approach. In an indirect approach, a regression model is constructed to predict the outcome using the treatment, covariates (for instance, patient demographic information, clinical information, and genetic information) and the interactions between covariates and treatment as predictors. In other words, a model is used to estimate the conditional expected reward given the covariates and assigned treatment. The predictive model is then used to calculate the corresponding rewards under the different treatments, with the optimal treatment being the one with the largest predicted reward.

Qian and Murphy [2011] built a regression model with l_1 penalty to handle HDLSS data and predict the optimal reward for each patient. Similarly, Lu et al. [2013] proposed a regression model using the adaptive LASSO penalty. It simultaneously estimates the optimal treatment and identifies important variables. Other studies based on the indirect approach include Bang and Robins [2005], Cai et al. [2013], Feldstein et al. [1978], Geng et al. [2015], Stoehlmacher et al. [2004].

The indirect approach emphasizes prediction accuracy instead of optimizing the decision rule. Thus, the success of this approach depends heavily on correctly specifying the model. In contrast, Zhao et al. [2012] propose a weighted classification method, outcome weighted learning (OWL), to directly predict the optimal treatment for each patient. In their paper, they demonstrate the equivalence between maximizing the expected reward and minimizing the reward loss. In other words, they show

$$\arg\max_{\mathcal{D}} E(\frac{\mathbb{I}\{A = \mathcal{D}(X)\}}{P(A|X)} R) = \arg\min_{\mathcal{D}} E(\frac{\mathbb{I}\{A \neq \mathcal{D}(X)\}}{P(A|X)} R). \tag{2.8}$$

Considering a decision function $\mathcal{D}(\boldsymbol{x}) = \text{sign} f(\boldsymbol{x})$, finding the optimal ITR $\mathcal{D}(\boldsymbol{x})$ becomes

$$\arg\min_{\mathcal{D}} E(\frac{R}{P(A|X)} \mathbb{I}\{Af(X) < 0\}). \tag{2.9}$$

Notice that (2.9) can be viewed as minimizing a 0 - 1 loss function $\mathbb{I}\{Af(X) < 0\}$ with weights $R/P(A|X)$ in a classification framework. As the weight is proportional to the outcome R, this approach is called outcome weighted learning (OWL). Due to the non-convexity of the indicator loss function, it is computationally expensive to obtain the minimum of (2.9). To solve this objective function more efficiently, Zhao et al. [2012] propose using a surrogate loss function, in particular hinge loss $\ell(u) = (1-u)_+$ [Vapnik et al., 1995].

Given the observations $(\boldsymbol{x}_i, a_i, r_i)$, for $i = 1, \ldots, n$, the corresponding empirical version of the objective function based on SVM loss (SVM-OWL) is

$$\arg\min_f \frac{1}{n} \sum_{i=1}^{n} \frac{r_i}{P(a_i|\boldsymbol{x}_i)} \ell\{a_i f(\boldsymbol{x}_i)\} + \lambda_n J(f), \tag{2.10}$$

where $\ell(.)$ is the hinge loss. The $P(a_i|\boldsymbol{x}_i)$ is the conditional probability of treatment decision a_i based on subject characteristics. In clinical trials, usually $P(a_i|\boldsymbol{x}_i) = P(a_i)$ due to randomization and can be considered a pre-determined constant. In an observational study, the estimation of this probability using logistic regression is suggested by Zhao et al. [2012]. $J(f)$ is the regularization part used to avoid overfitting, and λ_n is its corresponding tuning parameter. In Zhao et al. [2012], the L_2 regularization is used. The choice of λ_n is determined by cross validation [Devijver and Kittler, 1982, Kohavi et al., 1995]. The estimated optimal ITR is determined by

$$\hat{\mathcal{D}}^*(\boldsymbol{x}) = \text{sign}(\hat{f}(\boldsymbol{x}))$$

SVM-OWL links the single-stage precision medicine to a classification problem. Different loss functions can be applied to the OWL framework to get different methods, like tree-based OWL [Cui et al., 2017, Laber and Zhao, 2015] or neural network-based methods [Liang et al., 2018].

The merit of DWD in the HDLSS setting was discussed in Section 2.1.3. When considering weighted binary classification method, Qiao et al. [2010] showed that weighted DWD has a better prediction performance compared to weighted SVM. Therefore, instead of using SVM-OWL, we suggest using DWD-based OWL (DWD-OWL). In Chapter 4, we discuss the merits of SVM-OWL and DWD-OWL in the precision medicine setting and demonstrate the superiority of DWD-OWL in the imbalanced setting. .

3. MULTICATEGORY DISTANCE WEIGHTED SUPPORT VECTOR MACHINE

3.1 Overview

Classification is important in both statistics and machine learning. The goal of classification is to build a classifier such that it can correctly predict the category of a new observation. Popular classification methods include Fisher's linear discriminant analysis, logistic regression, support vector machine (SVM), and boosting. See Hastie et al. [2001] for an introduction to various classification methods.

SVM [Cristianini et al., 2000, Schölkopf and Burges, 1999] has been shown to be a very popular and powerful method. It is well known that binary SVM searches for a hyperplane in the feature space that maximizes the margin (recall Figure 2.2). SVM has numerous applications, such as image classification [Chapelle et al., 1999, Foody and Mathur, 2006] and cancer detection [Duan et al., 2005, Wang and Huang, 2011].

In a high-dimensional, low sample size (HDLSS) setting, Marron et al. [2007] and Ahn and Marron [2010] observed a *data-piling* phenomenon with binary SVM and other classification methods. A SVM-type linear classifier is a margin-based classifier. It has a separating hyperplane, and its normal vector is essentially the discriminant direction. Data-piling occurs when many of the data projections to the discriminant direction are identical. This indicates that the resulting separating hyperplane might be affected by noise artifacts in the data, resulting in a discrimination direction that can be far away from the Bayesian direction. See more discussion in Ahn and Marron [2010].

To alleviate the *data-piling* issue, Marron et al. [2007] proposed the binary distance-weighted discrimination (DWD) classifier. The idea of DWD is to minimize the total inverse margin of all the data points. This method works quite well in the HDLSS setting. However, because the DWD method uses all the observations to estimate the decision boundary, it is very sensitive to imbalanced sample sizes [Qiao et al., 2010]. In particular, when the sample size of one class is much larger than the other, the classification boundary will be pushed towards the minority class and all future data will be assigned to the majority class.

To deal with both problems, Qiao and Zhang [2015a] proposed a binary distance-weighted support vector machine (DWSVM) method, which can be viewed as a combination of the binary SVM and DWD. The new method inherits both the merits of SVM and DWD yet outperforms both SVM and DWD in the HDLSS and imbalanced context.

In practice, many classification problems have more than two classes. It is more desirable to consider all classes simultaneously. In a multiclass setting, the observed data are $(\boldsymbol{x}_i, y_i), i = 1, \ldots, n$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ is a multivariate predictor, the scalar $y_i \in \{1, \ldots, K\}$ is the corresponding class label, with K as the number of classes. Many classification approaches map x to $\boldsymbol{f}(x) \in \mathbb{R}^K$, and the corresponding prediction rule is $\hat{y} = \arg\max_j f_j(\boldsymbol{x})$, where f_j is the jth element of $\boldsymbol{f}$. In this type of approach, a constraint such as $\sum_{j=1}^K \boldsymbol{f}_j = 0$ is usually imposed to remove redundancy and reduce the dimension of the problem.

It is straightforward to see that the sum-to-zero constraint can be removed if we redefine $\boldsymbol{f}$ in $\mathbb{R}^K$ to be in $\mathbb{R}^{K-1}$, as the degrees of freedom of $\boldsymbol{f}$ is essentially $K - 1$. Several classifiers have been proposed using this fact (Section 2.2.2). One of them is the angle-based method. The angle-based method can be viewed as a natural extension of the binary large margin classifier to the multiclass context. Zhang and Liu [2014] replace the usual functional margin by the angle (or inner product) between the projection $\boldsymbol{f}$ and the vertices. Their simulation results show that these angle-based classifiers have good prediction performance. The Fisher consistency of a family of large margin classifiers is also proved. However, as a specific case in large margin classifiers, the angle-based SVM (MSVM) method is not Fisher consistent because its loss function is not a strictly monotone decreasing function. In Zhang and Liu [2014], Fisher consistency of a proximal SVM was proposed and proven instead.

In Section 3.2, we show that when under the HDLSS and imbalanced data setting, MSVM suffers from *data piling*. Binary DWD, based on the idea of Zhang and Liu [2014], can be extended to multicategory angle-based DWD (MDWD). Though free from the *data piling* concern, MDWD suffers from the imbalanced issue.

In this chapter, we adapt the idea in Qiao and Zhang [2015a] to develop a hybrid of MSVM and MDWD. The work can also be viewed as an extension of the binary DWSVM to the multiclass context. We prove its Fisher consistency and use extensive simulation studies to show the usefulness of our approach. For many cases, the novel approach outperforms both MDWD and MSVM, especially under HDLSS and the imbalanced case.

The rest of this chapter is organized as follows. In Section 3.2, we briefly review the existing multicategory classifiers, and introduce our angle-based distance-weighted support vector machine (MDWSVM) model. In Section 3.3, we present the Fisher consistency and show some imbalance properties of our new approach. In Section 3.4, we perform simulation studies to compare our model with MSVM and MDWD. The sensitivity of the prediction performance in terms of the tuning parameters is also explored in this section. Section 3.5 involves a real application. The proofs of all theorems and lemmas are given in Section 3.6.

3.2 Methodology

In this section, we give a general introduction to classification, including the angle-based multicategory classifier. We then show some drawbacks of this angle-based classification, which motivates our MDWSVM. We conclude with a detailed introduction of our approach, along with its implementation.

3.2.1 Classification and loss function

Consider a binary classification problem with observed data $(\boldsymbol{x}_i, y_i), i = 1, \dots, n$. The $\boldsymbol{x}_i \in \mathbb{R}^d$ is a multivariate predictor and the scalar $y_i \in \{1, -1\}$ is the corresponding class label. The goal is to find a decision function f along with its prediction $\hat{y}(\boldsymbol{x}) = \text{sign}(f(\boldsymbol{x}))$ to minimize the misclassification error $E(\hat{Y} \neq Y)$. Note that when $yf(\boldsymbol{x}) > 0$, $f(\boldsymbol{x})$ gives a correct prediction; otherwise $f(\boldsymbol{x})$ gives a misclassification. A natural way to estimate the misclassification error is to use the empirical error $1/n \sum \mathbb{I}(\hat{y}_i(\boldsymbol{x}) \neq y_i) = 1/n \sum \mathbb{I}(y_i f(\boldsymbol{x}_i) < 0)$, where $\mathbb{I}(.)$ is the indicator function. However, due to the discontinuity and nonconvexity of $\mathbb{I}(y_i f(\boldsymbol{x}_i) < 0)$, it is hard to conduct a direct minimization.

A common surrogate is a convex loss function $\ell(.)$, which is commonly used in large margin classifiers [Hastie et al., 2001]. A large margin classifier can be viewed as minimizing the loss function given a constraint:

$$\min_{f \in \mathbb{F}} \quad \sum_{i=1}^{n} \ell(f(\boldsymbol{x}_i), y_i) + \frac{\lambda}{2} J(f)$$

Here $\mathbb{F}$ denotes the function space and $J(.)$ is a type of norm, which is used to control the complexity of the model. The function $\ell(f, y)$ is a loss function surrogate for the 0-1 loss. The tuning parameter λ balances the loss and the norm. For example, the popular linear SVM uses the hinge loss function $\ell_S(u) = (1 - u)_+$ where $u = yf(\boldsymbol{x})$, and the L_2 norm.

The SVM method can also be viewed as maximizing the smallest distances of all observations to the separating hyperplane. As discussed in Section 2.1.2, SVM suffers from the *data piling* problem in HDLSS setting. Marron et al. [2007] proposed the DWD method, which improves the performance of SVM in the HDLSS setting. Essentially, DWD minimizes the mean of inverse distance of all data vectors to the separating hyperplane. As is discussed in Bartlett et al. [2006], Liu et al. [2011], Qiao and Zhang [2015a], DWD is also a large margin classifier, and its loss function is

$$\ell_D(u) = \begin{cases} 1-u & u \le 1/2 \\ 1/(4u) & \text{otherwise.} \end{cases} \tag{3.1}$$

In practice, lots of applications deal with multicategory rather than binary classification. For multiclass problems, $y_i \in \{1, 2, \ldots, K\}, i = 1, \ldots, n$, with K the number of classes. The common simultaneous procedure is to map $\boldsymbol{x}$ to $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^K$, and the corresponding prediction rule is $\hat{y} = \arg\max_j f_j(\boldsymbol{x})$, where f_j is the jth element of $\boldsymbol{f}$. Commonly a sum to zero constraint on $\boldsymbol{f}$ is used as discussed in Section 2.2 to overcome identifiability issues, see more discussion in Lee et al. [2004], Liu and Yuan [2011], Vapnik and Vapnik [1998].

Many multicategory classification methods can be viewed as the following constrained optimization problem,

$$\begin{aligned} \min_{f \in \mathbb{F}} \quad & \sum_{i=1}^{n} \ell(\boldsymbol{f}(\boldsymbol{x}_i), y_i) + \frac{\lambda}{2} \sum_{j=1}^{K} J(f_j), \\ \text{s.t.} \quad & \sum_{j=1}^{K} f_j(\boldsymbol{x}) = 0. \end{aligned}$$

For example, a multicategory SVM with hinge loss [Vapnik and Vapnik, 1998] uses the loss function $\ell(\boldsymbol{f}(\boldsymbol{x}), y) = (1 - f_y(\boldsymbol{x}))_+$. However, unlike binary classification, multicategory classification with a sum to zero constraint does not have a clear geometric explanation. It also suffers from expensive computation [Zhang and Liu, 2014]. To overcome these limitations, Lange and Wu [2008] proposed the vertex idea where they define $\boldsymbol{f}$ as a $K-1$ dimensional function instead of a K dimensional function. This removes the need for a sum-to-zero constraint. A similar idea is used later by Zhang and Liu [2014] to conduct angle-based classification, which will be discussed next.

3.2.2 Angle-based classification framework

The idea of angle-based classification is to map $\boldsymbol{x}$ to $\boldsymbol{f}(\boldsymbol{x})$, where $\boldsymbol{f} = (f_1, \ldots, f_{K-1})$, with a set of K predefined vertices in $\mathbb{R}^{K-1}$. We then assess which vertex has the smallest angle to the projection $\boldsymbol{f}$, and the corresponding label is the prediction. In Zhang and Liu [2014], the vertices $W = (W_1, W_2, \ldots, W_K)$ are defined as a collection of K vectors in $\mathbb{R}^{K-1}$ with elements

$$W_j = \begin{cases} (K-1)^{-1/2}\zeta, & j = 1, \\ -(1+K^{1/2})/\{(K-1)^{3/2}\}\zeta + \{K/(K-1)\}^{1/2}e_{j-1}, & 2 \leq j \leq K. \end{cases}$$

where ζ is a unit vector of length $K-1$, and e_j is a vector in $\mathbb{R}^{K-1}$ such that all of its element are 0, except the jth is 1.

In this setting, W form a simplex with K vertices in a $(K-1)$ dimensional space. The center of W is at the origin, and each of the $W_j, j = 1, \ldots, K$ has Euclidean norm of 1. Further, it is easy to check that the angle between each pair of vertices W_i and W_j, $i \neq j$ is the same. Instead of y_i, W_{y_i} is used to represent the observed class. The prediction function is $\hat{y} = \arg\max_j \langle W_j, \hat{\boldsymbol{f}} \rangle$, where the inner product $\langle ., . \rangle$ between the two vectors denotes the projection of $\hat{\boldsymbol{f}}$ to W_j. The larger the inner product, the smaller the angle between $\hat{\boldsymbol{f}}$ and W_j.

With this prediction rule, Zhang and Liu [2014] proposed the following optimization for the angle-based classification:

$$\min_{\boldsymbol{f} \in \mathbb{F}} \frac{1}{n} \sum_{i=1}^{n} \ell(\langle \boldsymbol{f}(\boldsymbol{x}_i), W_{y_i} \rangle) + \frac{\lambda}{2} J(\boldsymbol{f}). \tag{3.2}$$

Here the product $\langle \boldsymbol{f}(\boldsymbol{x}_i), W_{y_i} \rangle$ can be viewed as a new functional margin of $(\boldsymbol{x}, y)$. Defining $u = \langle \boldsymbol{f}(\boldsymbol{x}_i), W_{y_i} \rangle$, one of the examples given in Zhang and Liu [2014] is the multicategory angle-based SVM (MSVM) where the loss function is replaced by hinge loss $\ell_S(u) = (1-u)_+$ and with L_2 norm. DWD loss can be applied to this framework as well, along with more generalizations of binary large margin classifiers. See Zhang and Liu [2014] for more details.

3.2.3 From Binary DWSVM to MDWSVM

Through the exploration of the angle-based classification method, we found that MSVM has similar *data piling* issues; while MDWD does not have *data piling* problems, it suffers from imbalanced issues. To demonstrate the *data piling* and imbalanced issues, we show projection plots of a simulated example. In this example, we randomly simulate three classes of observations with 500 covariates, the sample sizes for each class are 100, 50, 50 respectively. For each group, the first two covariates are distributed $N(\mu_j, \sigma^2 I_2)$, where μ_j's are three fixed points equally spaced on the unit circle, and $\sigma = 0.5$. All other covariates are independently and identically distributed $N(0, \sigma^2)$. Note that the data are HDLSS and imbalanced.

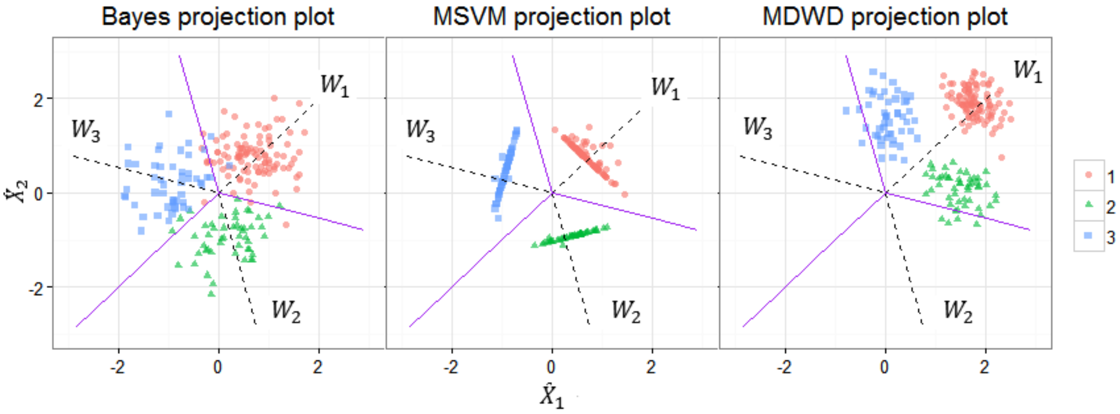

Figure 3.1. Plots of projections and W_j's in $\mathbb{R}^2$ space. Dashed lines are the W_j's, $j = 1, \ldots, 3$; Dots, triangles and squares represent the points from the three different classes. The left panel shows the projection plot for the Bayes classification, the middle one is for angle-based MSVM, and the right one is for the angle-based MDWD. The middle panel shows the MSVM has severe *data piling* issues (the middle panel), and MDWD in the right panel suffers from imbalanced issues (the right panel).

One representation of the projection plots is given in Figure 3.1. This plot is used to visualize the n projections $\boldsymbol{f}(x_i)$'s, $i = 1, \ldots, n$ and vectors W_j's, $j = 1, \ldots, 3$ (dashed lines) in the $\mathbb{R}^2$ plane. The different colors and shapes correspond to different groups. The solid purple lines are the Bayesian decision boundaries. $\tilde{X}_1$ and $\tilde{X}_2$ are the two axes in this $\mathbb{R}^2$ plane.

From Figure 3.1 it is clear that MSVM has severe *data piling* issues since almost all the points project to a single point on the W direction. Furthermore, MDWD suffers from the imbalanced issue as the angle-based classification assigns almost all the points to the dominant Class 1. These findings agree with those in Qiao and Zhang [2015a] for the binary case.

To alleviate both data piling and imbalanced issues, Qiao and Zhang [2015a] proposed binary DWSVM, a combination of SVM and DWD. The method has the following form:

$$\min_{f \in \mathbb{F}} \frac{1}{n} \sum_{i=1}^{n} \alpha \ell_D(y_i f_0(\boldsymbol{x}_i)) + (1-\alpha)\ell_S(y_i f(\boldsymbol{x}_i)) + \frac{\lambda}{2} J(f), \tag{3.3}$$

where $f_0(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{\omega} + \beta_0$ and $f(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{\omega} + \beta$, $\boldsymbol{\omega} \in \mathbb{R}^d$ is the coefficient direction vector and scalar $\beta, \beta_0 \in \mathbb{R}$. The loss function ℓ_D is from formula (3.1) and ℓ_S is the hinge loss. Notice that β is the SVM intercept and β_0 is the DWD intercept, which is called auxillary intercept in the DWSVM paper. The prediction function is $\hat{y} = \text{sign}(f(\boldsymbol{x})) = \text{sign}(\boldsymbol{x}_i^T \boldsymbol{\omega} + \beta)$. The tuning parameter $0 < \alpha < 1$ is used to balance SVM and DWD losses.

Qiao and Zhang [2015a] show that in binary classification, by choosing the appropriate α, the DWSVM method will result in a smaller misclassification error compared to both the DWD method and SVM method in HDLSS and imbalanced data context. Furthermore, in terms of the similarity to the Bayes classifier, the DWSVM are similar to the DWD but better than the SVM.

The DWSVM method motivated us to build a multicategory DWSVM within the angle-based framework. Applying DWSVM to the angle-based framework, we propose a multicategory angle-based DWSVM (MDWSVM)

$$\min_{\boldsymbol{f}\in\mathbb{F}} \frac{1}{n}\sum_{i=1}^{n} \alpha\ell_D(\langle \boldsymbol{f}_0(\boldsymbol{x}_i), W_{y_i}\rangle) + (1-\alpha)\ell_S(\langle \boldsymbol{f}(\boldsymbol{x}_i), W_{y_i}\rangle) + \frac{\lambda}{2}J(\boldsymbol{f}). \tag{3.4}$$

In this model $\boldsymbol{f}(\boldsymbol{x}_i) = \boldsymbol{x}_i B + \beta_0$ and $\boldsymbol{f}_0(\boldsymbol{x}_i) = \boldsymbol{x}_i B + \beta_0^d$, where $B = (B_1, B_2, \ldots, B_{K-1})$, each of the $B_j, j = 1, \ldots, K-1$ is a vector of length d, which does not include the intercept.The parameters $\beta_0, \beta_0^d \in \mathbb{R}^{K-1}$ are intercept vectors. Note that $\boldsymbol{f}_0$ and $\boldsymbol{f}$ are only different in terms of the intercept β_0 and β_0^d respectively. In this model, the interest is to find B, β_0 and β_0^d to minimize the loss function. For prediction $\hat{y} = \arg\max_j \langle W_j, \hat{\boldsymbol{f}}\rangle = \arg\max_j \langle W_j, \boldsymbol{x}_i B + \beta_0\rangle$, however, only β_0 and B are used. This avoids the imbalanced issue cost by β_0^d. Note that the prediction $\arg\max_j \langle W_j, \hat{\boldsymbol{f}}\rangle$ is equivalent to $\arg\min_j \angle(W_j, \hat{\boldsymbol{f}})$ where $\angle(a, b)$ represents the angle between vector a and b . We predict $\boldsymbol{x}$ with the label j such that vertex W_j and $\boldsymbol{f}(\boldsymbol{x})$ has the smallest angle among all $\angle(W_j, \hat{\boldsymbol{f}}), j = 1, \ldots, K$. Observe that $\sum_{j=1}^{K} \langle W_j, \hat{\boldsymbol{f}}\rangle = 0$ for all $\boldsymbol{x}$, which means the angle-based classification framework automatically includes sum-to-zero constraints.

3.2.4 Implementation for MDWSVM

In Qiao and Zhang [2015a], the implementation of the binary DWSVM (3.3) was through second-order cone programming. Mathematically, the DWSVM model can be written as (2.5).

The first constraint $d_i = y_i(x_i^T\boldsymbol{\omega} + \beta) + \eta_i$ is the distance from each data vector i to its separating hyperplane (adding slackness to allow misclassification), which corresponds to the DWD optimization. The second constraint $y_i(x_i^T\boldsymbol{\omega} + \beta) + \xi_i \geq 1$ is a standard constraint in SVM optimization. Both η_i and ξ_i control the misclassification rate, but with different decision boundaries. The third constraint $\|\boldsymbol{\omega}\|^2 \leq C$ is equivalent to the second term in (3.2), the Euclidean norm.

To extend (2.5) to multiclass, we replace the distances (functional margins) to the inner product as introduced earlier in (3.2). Thus our MDWSVM will have the following mathematical form:

$$\begin{aligned} \min_{\boldsymbol{f},\boldsymbol{f_0}} \quad & \sum_{i=1}^{n}\{\alpha(\frac{1}{d_i}+\eta_i)+(1-\alpha)\xi_i\}, \\ \text{s.t.} \quad & d_i = \langle \boldsymbol{f}_0(x_i), W_{y_i}\rangle + \eta_i, \quad d_i \geq 0 \quad \text{and} \quad \eta_i \geq 0, \\ & \langle \boldsymbol{f}(x_i), W_{y_i}\rangle + \xi_i \geq 1, \quad \xi_i \geq 0, \\ & \sum_{j=1}^{k-1} B_j^T B_j \leq C. \end{aligned} \tag{3.5}$$

In this form $\boldsymbol{f_0}$ and $\boldsymbol{f}$ are the same as in (3.4). It is verified in Zhang and Liu [2014] that the first term in the objective function $\sum_{i=1}^{n}(\frac{1}{d_i}+\eta_i)$ along with its constraint is equivalent to the objective of the MDWD method, and the second term in the objective function $\sum_{i=1}^{n}\xi_i$ along with its constraint is equivalent to the objective in the MSVM method. In this case, MDWSVM can be viewed as a convex combination of MDWD and MSVM losses where the parameter α balances the two.

Model (3.5) can be easily implemented in Matlab using the CVX package [Grant et al., 2008]. Notice the only difference between $\boldsymbol{f(x)}$ and $\boldsymbol{f_0(x)}$ is their location vectors β_0 and β_0^d. For prediction we only adopt the location vector from MSVM, which shows insensitivity to the imbalanced issue from Figure 3.1. Moreover, by combining the discriminant direction of MDWD and MSVM, our new model will have a better discriminant direction (closer to the Bayes direction) than the MSVM method alone. Both improvements will be shown in Sections 3.4 and 3.5 using simulations and real examples.

3.3 Theoretical properties

Fisher consistency is a fundamental requirement for a classification method. Fisher consistency implies that when the sample size approaches infinity, the classifier becomes closer and closer to the Bayes classification rule, which corresponds to the minimum misclassification rate. Qiao and Zhang [2015a] explored Fisher consistency of the binary DWSVM model. In the multiclass context, Zhang and Liu [2014] extended Fisher consistency to all large margin classification models under the angle-based framework.

Let $P_j = \Pr(Y = j|X = \boldsymbol{x})$ for $j = 1, \ldots, K$. Note that $\hat{y} = \arg\max_j P_j$ is the Bayes rule. Assume that for a given $\boldsymbol{x}$, the vector $\boldsymbol{f^*(x)}$ minimizes

$$E[\ell\{\langle \boldsymbol{f}(X), W_Y\rangle\}|X = \boldsymbol{x}],$$

and the corresponding decision boundary will then be $\hat{y} = \arg\max_j\langle \boldsymbol{f}^*(\boldsymbol{x}), W_Y\rangle$. Note that this is essentially the limit minimizer of (3.2) when sample size diverges to infinity. Fisher consistency assures that these two decision functions are the same ($\arg\max_j P_j = \arg\max_j\langle \boldsymbol{f}^*(\boldsymbol{x}), W_Y\rangle$).

In this section, we will prove that if using the approximate SVM loss function from Zhang and Liu [2014] in replacement of hinge loss, our MDWSM is Fisher consistent.

Theorem 3.3.1 *The MDWSVM is Fisher consistent for any* $0 < \alpha < 1$.

Fisher consistency in Theorem 3.3.1 ensures that the minimizer of the expected loss function will assign an observation to the same class as what Bayes rule does. Furthermore, in our numerical study in Section 3.4, we notice that for MDWSVM method, as long as C is fixed, different α's will give similar performance in both prediction error and closeness to the Bayes rule. Thus we will fix α to be 0.5 in this paper and not discuss the choice of α further.

In the next theorem, we want to prove that MDWSVM is insensitive to imbalance. Using a similar paradigm as in Owen [2007], we consider the case that the sample size of one class diverges to infinity. Qiao and Zhang [2015a] showed that, in binary classification, the intercept term of DWD diverges, but the intercept of SVM and DWSVM will not diverge. This shows that SVM is not sensitive to imbalance, but DWD will be severely affected. In our multiclass setting, for simplicity, it is assumed that only one of the classes is the dominant one, and the sample size of all other classes are equally fixed. Without loss of generality, we assume their sample sizes are all 1. Under this setting, we can simply assume observation $1, \ldots, K-1$ belongs to the class $1, \ldots, K-1$ respectively, and observations $K \ldots, n$ belong to class K. As n goes to infinity, the classifier tends to classify all the points to the dominant class K. If this happens, $\langle \beta_0, W_{y_K} \rangle$ goes to infinity. In the next proposition, we prove that this will be not be the case for the angle-based SVM. Furthermore, we present in Theorem 3 that the intercept of our MDWSVM model is not sensitive to imbalance either.

Proposition 3.3.1 *In MSVM setting, when the size of the majority class goes to infinity,* $\langle \beta_0, W_{y_K} \rangle < \sqrt{2C} K \max |x_{ij}| + 1$.

Theorem 3.3.2 *In the MDWSVM setting, when the size of the majority class goes to infinity,* $\langle \beta_0, W_{y_K} \rangle < \sqrt{2C} K \max |x_{ij}| + 1$.

Note that Theorem 3.3.2 does not ensure that the MDWSVM method completely overcomes the imbalanced issue. When the sample size of the majority group goes to infinity, the method still will ignore some observations in minority groups.

3.4 Simulation

In this section, we use three simulation examples to demonstrate the performance of our MDWSVM method. We compare it to the angle-based SVM (MSVM) described in Section 3.2 and the angle-based DWD (MDWD) naturally developed using the ideas from Zhang and Liu [2014].

In each example, we simulate a training data set, a tuning data set, and a testing data set. The training data and tuning data have the same sample sizes and are used to estimate the model and to find the optimal tuning parameters. The size of the testing data set is ten times the size of the training data, and is used to evaluate the prediction performance. As we are interested in the misclassification rate in both the dominant class and the minority classes, we will not use the total error rate

$1/n \sum_i I(\hat{y}_i \neq y_i)$ in this paper. Instead, we use the average within-group error rate as follows

$$\frac{1}{K}\sum_{j=1}^{K}\left\{\frac{1}{n_j}\sum_{i:x_i\in C_j} I(\hat{y}_i \neq j|x_i \in C_j)\right\}.$$

Here C_j stands for class j and n_j is the sample size for class j. This measure was previously introduced in Qiao and Liu [2009]. Note that the term within the bracket is the error rate for each group, so r is the arithmetic average of all these error rates.

We also want to measure the closeness of the estimated classifier to the Bayes rule. For the binary case, we can measure the angle between the two linear decision boundaries. For the multiclass case, we develop a similar measure as follows. Note that B is the projection matrix from $\mathbb{R}^d$ to $\mathbb{R}^{K-1}$ (the projection space). In the binary case, B is the discrimination direction vector, we can use the Euclidean inner product $\langle B, B_{\text{Bayes}}\rangle$ to measure the angle between the estimated and the Bayes rule. For the multiclass case, both B and B_{Bayes} are matrices. In matrix form, we want to measure the angle between the jth columns in both B and B_{Bayes}, and then calculate an average of these angles.

In this paper, we use the Frobenius inner product: $\langle B, B_{\text{Bayes}}\rangle_F = \sum_{i,j} B_{ij} B_{\text{Bayes}\ ij}$. Essentially, this is the sum of entries of the Hadamard product between B and B_{Bayes}. One can see that this is the same idea as the inner product of the corresponding columns in B and B_{Bayes}, a scaled mean of the inner products. To make this quantity directly linked to angle, we normalize both B and B_{Bayes} to have Frobenius norm of 1, and thus $\angle\langle B, B_{\text{Bayes}}\rangle = \arccos(\langle B, B_{\text{Bayes}}\rangle_F)$ will be the angle used in this paper.

In all examples, α is set as 0.5, the reason for this is described in Section 3.4.2. We want to choose C in $\mathbb{R}^+$. For convenience, we use the log scale, and set $\log_2 C$ from -3 to 15. For the first two examples, we generate datasets that have signal based on only a few covariates, and then we add pure noise as additional covariates. To better compare the performance for both balanced and imbalanced scenarios, all the examples are conducted under both balance and imbalance cases. Let $p = \Pr(Y = 1)$ and $\Pr(Y = j) = \frac{1}{K-1}(1-p)$ for $j \neq 1$. We will consider $p = 1/K$ for the balanced case and $p = 1/2$ and $p = 1/3$ for the imbalanced case for all examples. The size of the training dataset for each example is 300, 600 and 300 respectively. In each example, five sets of dimensions are considered: 2, 10, 100, 500 and 1000. The noise covariates are identically independent distributed as $N(0, \sigma^2)$. For the third example, all covariates are signal variables. And for all simulation settings, we repeat the experiments 100 times and report the average performance.

3.4.1 Performance comparison

Three experiments are generated according to the following rules:

Example 1 We generate a three class dataset, where the first two covariates are distributed $N(u_j, \sigma^2 I_2)$. In this setting, the u_j's are three points equally spaced on a unit circle, and σ is chosen such that the Bayes error is 0.1. As we can see, this example is similar to the Example 1 in Zhang and Liu [2014] other than that our case considered both the balanced and imbalanced scenarios.

Example 2 We generate a five class dataset, Let $\Pr(Y = 1) = p$, and the first five covariates are distributed $N(u_j, \sigma^2 I_5)$. Here u_j's are five points equally spaced on the sphere of unit ball in $\mathbb{R}^4$, and $\sigma = 0.55$. When dimension is larger than 4, the last $d - 4$ covariates are identically independent distributed as $N(0, \sigma^2)$.

Example 3 A three groups dataset is generated with dimension d, the centers of the three groups are equally distributed on the sphere of an unit ball in $\mathbb{R}^d$. A random noise $N(0, \sigma^2 = 0.55^2)$ was added to each dimension.

We report the average prediction error rate and the average angle to the Bayes rule in Figures 3.2-3.4. Take Figure 3.2, which corresponds to Example 1, as an example. We report both misclassification rate (the top row) and the angle between the estimated classifiers and the Bayes rule. In the plot, we use black solid lines for our MDWSVM method, red dashed lines for MSVM method, and blue dotted lines for MDWD method. The grid points on the x axis represents the different dimensions

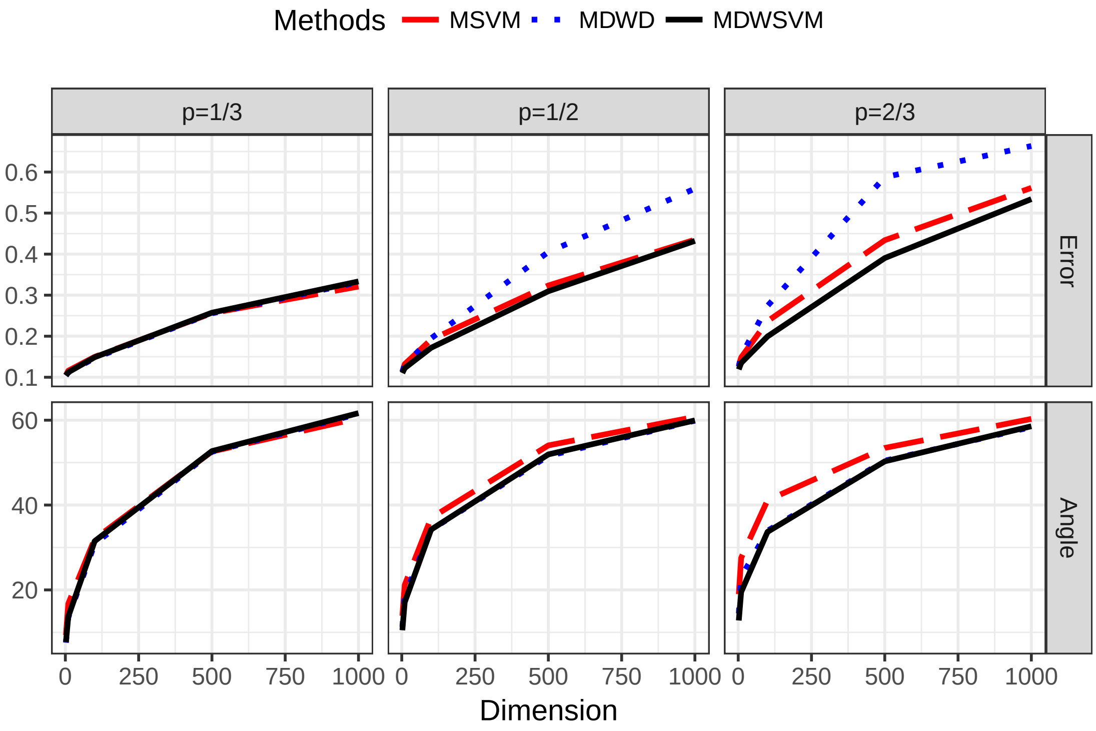

Figure 3.2. Performance comparison plot between the three methods for Examples 1. The top row plots the misclassification rate for different dimensions (the x axis) and different prior probabilities (left, middle and right panels). The bottom row is the angle between the estimated and the Bayes rule. For all measures, smaller implies better result.

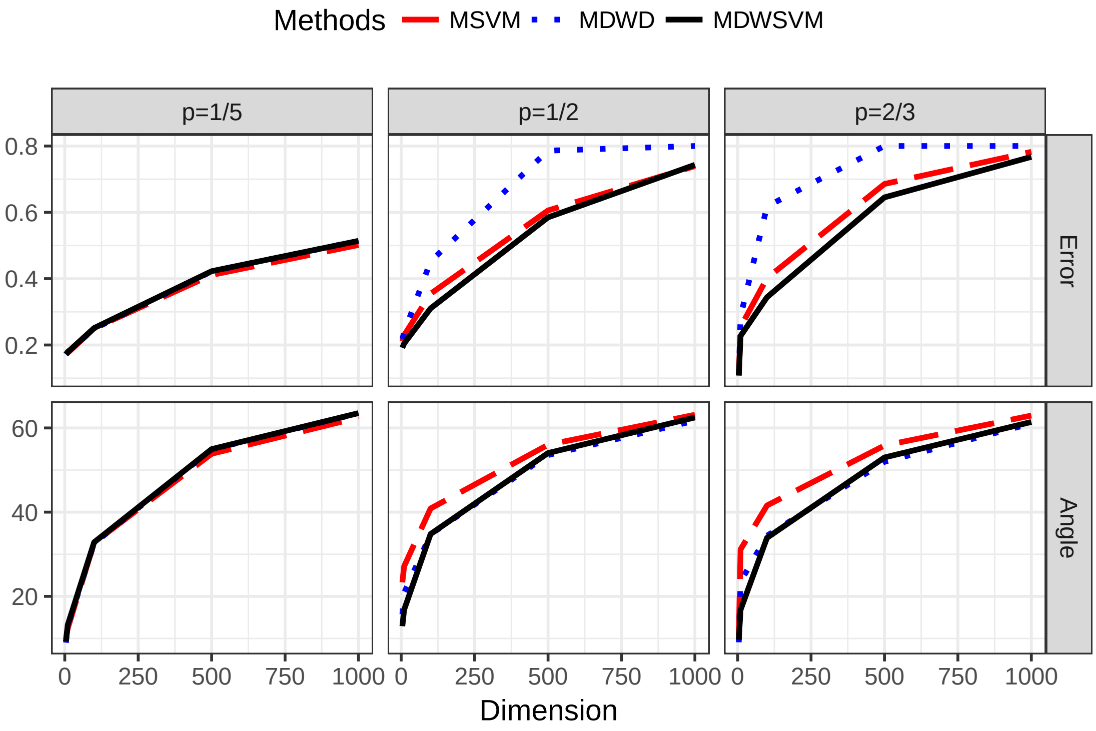

Figure 3.3. Performance comparison plot between the three methods for Examples 2. The top row plots the misclassification rate for different dimensions (the x axis) and different prior probabilities (left, middle and right panels). The bottom row is the angle between the estimated and the Bayes rule. For all measures, smaller implies better result.

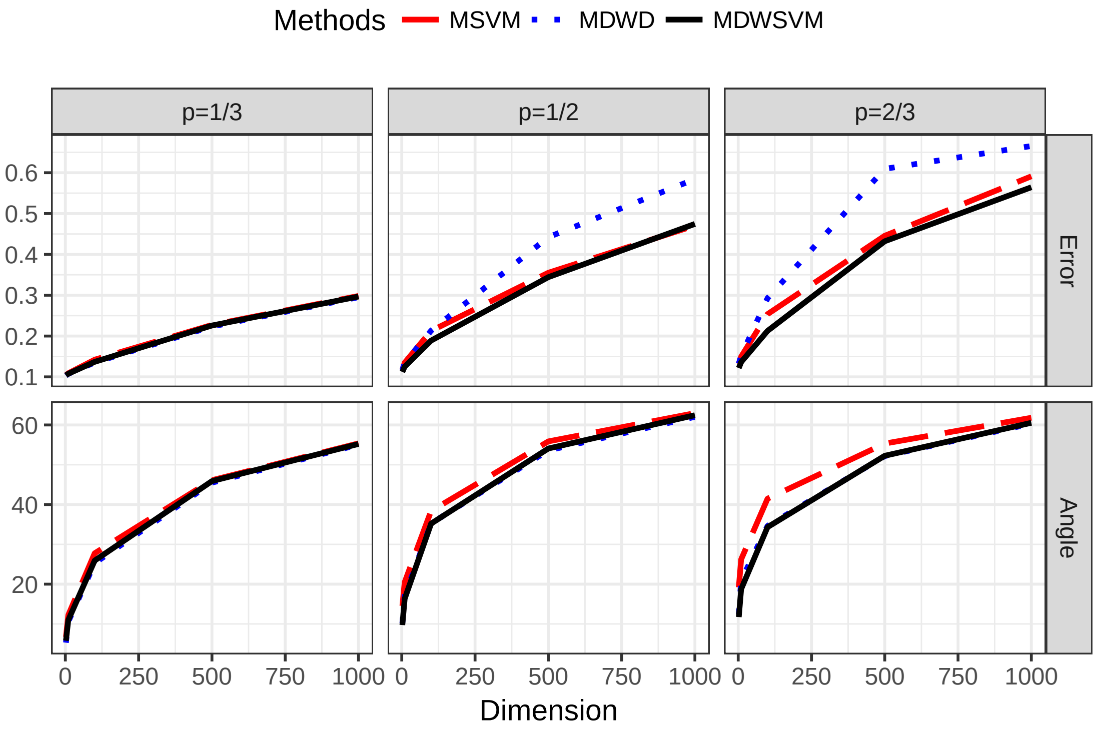

Figure 3.4. Performance comparison plot between the three methods for Examples 3. The top row plots the misclassification rate for different dimensions (the x axis) and different prior probabilities (left, middle and right panels). The bottom row is the angle between the estimated and the Bayes rule. For all measures, smaller implies better result.

d. The y axis corresponds to the performance measure. In our plot, the smaller the y axis value, the better the performance. Different imbalance ratios are visualized in different panels from left to right. The left two panels correspond to the balanced case; the middle panels are mild imbalance ($p = 1/2$); and the right two panels are more severe imbalanced case ($p = 2/3$). For the balanced case, our approach performs similar to the MSVM and MDWD methods. However, for the imbalanced cases (the middle and right panels), we can see clear gaps between the performance of our method and the other two methods, demonstrating that our method outperforms the other two. Note that a similar pattern can also be seen in Figures 3.3 and 3.4. All suggests that the novel approach is better than MSVM and MDWD.

It is also shown in these plots that as the dimension of training data changes from small to large (2 to 1000), the classifier's performances become worse and worse. Furthermore, the performance differences of the three methods become more pronounced. Note that MDWD gives the worst prediction error rate compared to the other two methods, and MSVM gives the worst classification direction compared to the other two methods. Our MDWSVM gives comparably the best performance in both aspects.

3.4.2 Sensitivity to parameters

There are two parameters C and α in our MDWSVM method. We have conducted many simulations to evaluate the performance of these two parameters. In this section, we will only use Example 1 to show the performance. We set α to be fixed, varied C, and evaluated its performance. Then we fixed an optimal C to evaluate the sensitivity of our approach to different α's. At the beginning, we let $\alpha = 0.5$, and allow C to change from 2^{-3} to 2^{12}. The simulation is conducted under different dimensions (100, 500, 1000). All the simulation results are based on 100 replicates.

The left panel of Figure 3.5 is the prediction error under different values of C with different dimensions of training data. It is clearly shown from the graph that as C

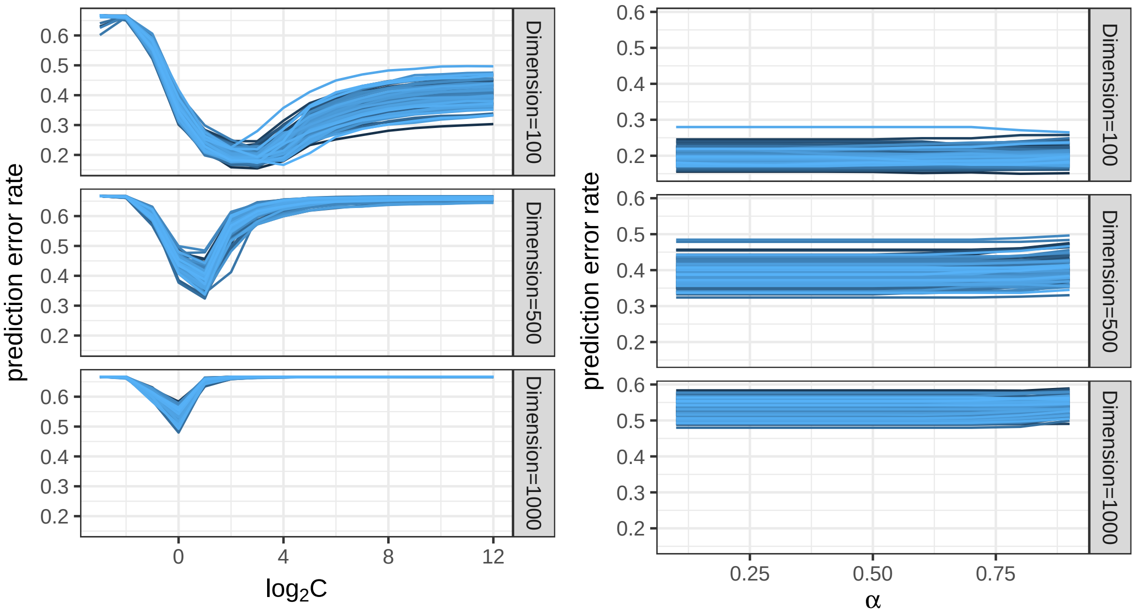

Figure 3.5. Average within-group error rate change under different parameters. Left panel is the prediction error rate change under different C value for fixed $\alpha = 0.5$; right panel is prediction error rate change for different α when C is fixed at its optimal value

increases, the prediction error rate first decreases and then increases. It shows that a minimal prediction error can be reached within this range. The right panel shows the relationship between prediction error rate and different α's. It is also clear that the prediction error rate stays the same as α changes from 0.1 to 0.9, regardless of the slightly increase as α approaches to 1.

Based on the performance from Figure 3.5, the change of α has little impact on the prediction error rate compared to a change of C. The performance is quite stable for different α's. Since the property of the parameters are not the focus in this paper, this simulation gives us an easy suggestion of choosing parameters. One can simply fix $\alpha = 0.5$ and use cross-validation to choose C. This is why we fix $\alpha = 0.5$ in our simulation.

3.4.3 Computation Time

In this section, we will compare the computational time for these methods (MDWSVM, MSVM and MDWD). To test the computational complexity, we only consider the simulation Example 1. We let the dimension change from 10 to 1000. Table 3.1 gives the average computation time in seconds for 100 replicates along with their standard error. All numerical experiments were carried out on an Intel Xeon E3-1284L (2.5 GHz) processor.

Table 3.1.
Computation time comparison for MDWSVM, MDWD and MSVM based on 100 runs of Example 1. The number shows average computing time in seconds, and the number in the parenthesis is the corresponding standard error.

Dimension	MDWSVM	MDWD	MSVM
10	8.31(0.04)	8.22(0.04)	0.88(0.01)
100	14.52(0.07)	12.80(0.05)	2.79(0.01)
1000	41.66(0.20)	27.43(0.12)	9.30(0.07)

Table 3.1 shows that the most efficient method is MSVM and the most time-consuming method is our MDWSVM. Note that MSVM can still be viewed as a quadratic programming problem, while both MDWD and MDWSVM are conic program problems. It is not surprising that MSVM is the most computational efficient one. It is our expectation that MDWSVM would have the longest time to run, since it combines both MSVM and MDWD. From equation 3.5, we can see that the number of parameters can be viewed as the sum of the ones for MDWD and MSVM. Thus the computation times it takes to solve the problem increases as well. It is good enough that the computational time of MDWSVM is shorter than the sum of the computing time of each individual methods. It is worth mentioning that as dimension increases, the CPU times for the three methods increase.

3.5 Real Data Application

In this section, we apply our MDWSVM method to a real data used in Shen and Huang [2005]. The data were gathered at an inbound call center of a major northeastern U.S financial firm in 2002, and describe the call volume from 7:00am-12:00am. Each day is divided into 408 150-second intervals and the number of phone calls is recorded in each interval. Due to equipment malfunctioning, there are 6 missing weekdays within the whole year. The call volume data form a 360 × 408 matrix, where each row corresponds to a day and each column is the call volume for one of the 150-second intervals.

Note that the data have been thoroughly analyzed in Shen and Huang [2005]. Here we simply add some new insights from the data by using our novel approach. According to their analysis, the pattern for Saturday and Sunday is very different from the weekdays. Thus in this analysis, we only focus on the weekdays. Shen and Huang [2005] show that for weekdays, by using singular value decomposition to analyze the number of phone calls, Monday and Friday are slightly different from all other weekdays, see Section 5.3 and Figure 6 of Shen and Huang [2005] for more details. Tuesday, Wednesday and Thursday are hard to tell apart from each other. In this section, we only focus on classifying Monday, Friday and other weekdays (Tuesday, Wednesday and Thursday). In addition, due to the fact that the center has very low volumes on national holidays, we remove the holidays that fall on weekdays. Table 3.2 provides the national holidays excluded in our analysis. These days, along with some other more holidays, were also removed in Shen and Huang [2005].

Table 3.2.
The six national holidays on weekdays that are removed

2002/01/01	New Year	2002/09/02	Labor Day
2002/05/27	Memorial Day	2002/11/28	Thanksgiving
2002/07/04	Independence Day	2002/12/25	Christmas

After removing these holidays, we have 48, 50, 51, 50, 52 days for Monday to Friday respectively. The data are divided into three groups, Group 1: Monday (size 48); Group 2: Tuesday to Thursday (size 151); Group 3: Friday (size 52). This dataset is a typical imbalanced HDLSS dataset with Group 2 as the dominant group. The average number of phone calls on each time interval are presented in Figure 3.6. From Figure 3.6, we can see that the average number of phone calls on Monday is quite distinct from the other days as it is larger than the other two groups. However, Group 2 and 3 are hard to distinguish from each other by only looking at the average number of phone calls. For this data set, we will compare the performance of three classifiers: our MDWSVM, MSVM, MDWD.

To obtain a good evaluation, all three methods use five-fold cross validation. And the evaluation measures are the total error rate (TER) and the average within-group error rate (AER). We also report the prediction error within each group. Table 3.3 provides these results for the three different methods.

From Table 3.3 it is straightforward to conclude that the prediction error of MDWSVM for each of the groups is the smallest, as well as the total error rate and

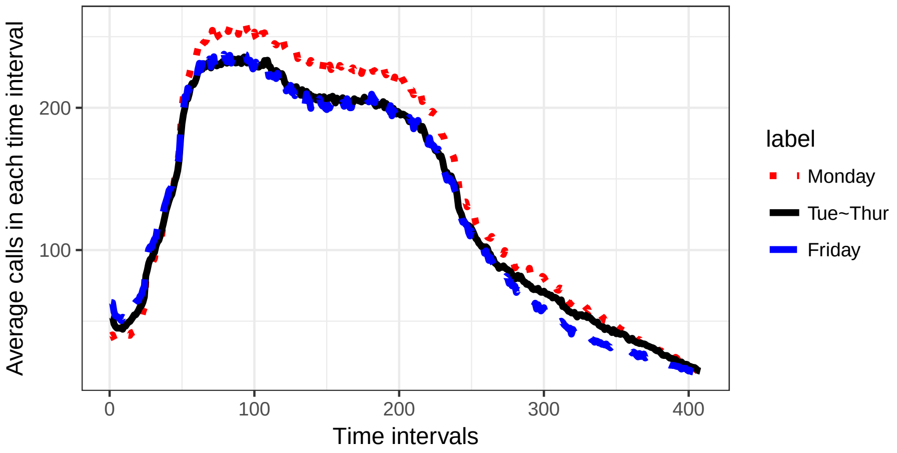

Figure 3.6. Average number of calls in each time interval for the three classes Monday, Tuesday to Thursday and Friday. In this plot, the red dotted line is the number of phone calls for Mondays, the black solid line is for Tuesday to Thursday, and the blue dashed line is for Friday.

Table 3.3.
Prediction error for number of phone calls. The top 3 rows report the cross-validation prediction error for each class, and the bottom two rows are the total error rate and the average within-group error rate.

	MDWD	MSVM	MDWSVM
Mon.	0.8156	0.6200	0.4111
Tue. - Thu.	0.0396	0.0594	0.0985
Fri.	0.6145	0.5527	0.3200
TER	0.3098	0.2702	0.2053
AER	0.4899	0.4107	0.2765

average with-group prediction error rate. Our MDWSVM model works very well for this data set.

Taking another look at the Table 3.3, the prediction error rate for Monday and Friday are more than 30%, which seems to be large. The result could be explained by the fact that the DWSVM used here is a linear classifier, while the nature of the data may not be well classified by a linear classifier. If we incorporate a kernel approach to our classifier, the performance should improve.

3.6 Proofs of Theoretical Properties

In this section we prove the following theorems and proposition from Section 3.3.

3.6.1 Proof of Fisher Consistency for MDWSVM

Lemma 3.6.1 *Zhang and Liu [2014]Suppose we have an arbitrary* $\boldsymbol{f} \in \mathbb{R}^{K-1}$*. For any* u*,* $v \in \{1, \dots, K\}$ *such that* $u \neq v$*, define* $T_{u,v} = W_u - W_v$*. For any scalar* $z \in \mathbb{R}$*,* $\langle(\boldsymbol{f} + zT_{u,v}), W_w\rangle = \langle \boldsymbol{f}, W_w\rangle$*, where* $w \in \{1, \dots, K\}$ *and* $w \neq u, v$*. Furthermore, we have that* $\langle(\boldsymbol{f} + zT_{u,v}), W_u\rangle - \langle \boldsymbol{f}, W_u\rangle = -\langle(\boldsymbol{f} + zT_{u,v}), W_v\rangle + \langle \boldsymbol{f}, W_v\rangle$*.*

The proof of Lemma 3.6.1 is given in Zhang and Liu [2014]. From Lemma 3.6.1, one can see that for a given $\boldsymbol{f}$, if we move it along the direction of $T_{u,v}$, the inner product of $\boldsymbol{f}$ and W_w will stay the same when $w \neq u, v$. Furthermore, the sum of inner product $\langle \boldsymbol{f}, W_u\rangle + \langle \boldsymbol{f}, W_u\rangle, W_u\rangle$ will remain unchanged as well. This lemma will help us to prove the Fisher consistency of the MDWSVM method.

Proof. of Theorem 3.3.1

Recall the definition of $\boldsymbol{f}^*$ is that

$$(\boldsymbol{f}^*, \boldsymbol{f}_0^*) = \arg\min_{\boldsymbol{f}, \boldsymbol{f}_0} E[(1-\alpha)\ell_s\{\langle \boldsymbol{f}(X), W_Y\rangle\} + \alpha\ell_d\{\langle \boldsymbol{f}_0(X), W_Y\rangle\}|X = x].$$

We need to show that when $P_1 > P_2$, $\langle W_1, \boldsymbol{f}^*\rangle > \langle W_2, \boldsymbol{f}^*\rangle$. This can be easily proved by contradiction.

If $\langle W_1, \boldsymbol{f}^* \rangle = \langle W_2, \boldsymbol{f}^* \rangle$, Let $\boldsymbol{f}_0^* = \boldsymbol{f}^* - \Delta$, here we can see that as Δ is only the difference of intercept, which is independent of X. Let $(\boldsymbol{f}^{**}, \boldsymbol{f}_0^{**}) = (\boldsymbol{f}^{**}, \boldsymbol{f}_0^*)$ be such that $\langle W_j, \boldsymbol{f}^{**} \rangle = \langle W_j, \boldsymbol{f}^* \rangle$ for $j \geq 3$ and $\langle W_1, \boldsymbol{f}^{**} \rangle = \langle W_1, \boldsymbol{f}^* \rangle + \epsilon$, $\langle W_2, \boldsymbol{f}^{**} \rangle = \langle W_2, \boldsymbol{f}^* \rangle - \epsilon$. This $(\boldsymbol{f}^{**}, \boldsymbol{f}_0^{**})$ exists based on Lemma 1 and the fact that inner product is continuous. To get the required $\boldsymbol{f}^{**}$, we only need to move $\boldsymbol{f}^*$ along the direction of $T_{1,2}$.

Then it is easy to get

$$\begin{aligned}
&\sum_{j=1}^{K} P_j[(1-\alpha)\ell_s\{\langle \boldsymbol{f}^{**}, W_j \rangle\} + \alpha \ell_d\{\langle \boldsymbol{f}_0^{**}, W_j \rangle\}] - \\
&\sum_{j=1}^{K} P_j[(1-\alpha)\ell_s\{\langle \boldsymbol{f}^{*}, W_j \rangle\} + \alpha \ell_d\{\langle \boldsymbol{f}_0^{*}, W_j \rangle\}] \\
=&\epsilon(P_1 - P_2)(1-\alpha)\ell_s'\{\langle \boldsymbol{f}^*, W_1 \rangle\} + o(\epsilon)
\end{aligned}$$

Since we are using proximal hinge loss, ℓ_s is differentiable, $P_1 - P_2 > 0$, $\ell_s' < 0$ and $0 < \alpha < 1$. we have

$$\begin{aligned}
&\sum_{j=1}^{K} P_j[(1-\alpha)\ell_s\{\langle \boldsymbol{f}^{**}, W_j \rangle\} + \alpha \ell_d\{\langle \boldsymbol{f}_0^{**}, W_j \rangle\}] \\
&< \sum_{j=1}^{K} P_j[(1-\alpha)\ell_s\{\langle \boldsymbol{f}^{*}, W_j \rangle\} + \alpha \ell_d\{\langle \boldsymbol{f}_0^{*}, W_j \rangle\}],
\end{aligned}$$

which is a contradiction.

For $\langle W_1, \boldsymbol{f}^* \rangle < \langle W_2, \boldsymbol{f}^* \rangle$ case, if $P_1\ell_s'\{\langle \boldsymbol{f}^*, W_1 \rangle\} - P_2\ell_s'\{\langle \boldsymbol{f}^*, W_2 \rangle\} < 0$, then choose $(\boldsymbol{f}^{**}, \boldsymbol{f}_0^{**}) = (\boldsymbol{f}^{**}, \boldsymbol{f}_0^*)$ be such that $\langle W_j, \boldsymbol{f}^{**} \rangle = \langle W_j, \boldsymbol{f}^* \rangle$ for $j \geq 3$ and $\langle W_1, \boldsymbol{f}^{**} \rangle = \langle W_1, \boldsymbol{f}^* \rangle + \epsilon$, $\langle W_2, \boldsymbol{f}^{**} \rangle = \langle W_2, \boldsymbol{f}^* \rangle - \epsilon$. Then we have

$$\begin{aligned}
&\sum_{j=1}^{K} P_j[(1-\alpha)\ell_s\{\langle \boldsymbol{f}^{**}, W_j \rangle\} + \alpha \ell_d\{\langle \boldsymbol{f}_0^{**}, W_j \rangle\}] - \\
&\sum_{j=1}^{K} P_j[(1-\alpha)\ell_s\{\langle \boldsymbol{f}^{*}, W_j \rangle\} + \alpha \ell_d\{\langle \boldsymbol{f}_0^{*}, W_j \rangle\}] \\
=&\epsilon(1-\alpha)\{P_1\ell_s'\{\langle \boldsymbol{f}^*, W_1 \rangle\} - P_2\ell_s'\{\langle \boldsymbol{f}^*, W_2 \rangle\}\} + o(\epsilon) \\
&< 0
\end{aligned}$$

We can see that If $P_1\ell_s'\{\langle \boldsymbol{f}^*, W_1\rangle\} - P_2\ell_s'\{\langle \boldsymbol{f}^*, W_2\rangle\} > 0$, then choose $(\boldsymbol{f}^{**}, \boldsymbol{f}_0^{**}) = (\boldsymbol{f}^{**}, \boldsymbol{f}_0^{*})$ be such that $\langle W_j, \boldsymbol{f}^{**}\rangle = \langle W_j, \boldsymbol{f}^*\rangle$ for $j \geq 3$ and $\langle W_1, \boldsymbol{f}^{**}\rangle = \langle W_1, \boldsymbol{f}^*\rangle - \epsilon$, $\langle W_2, \boldsymbol{f}^{**}\rangle = \langle W_2, \boldsymbol{f}^*\rangle + \epsilon$. Then we have

$$\begin{aligned}
&\sum_{j=1}^{K} P_j[(1-\alpha)\ell_s\{\langle \boldsymbol{f}^{**}, W_j\rangle\} + \alpha\ell_d\{\langle \boldsymbol{f}_0^{**}, W_j\rangle\}] - \\
&\sum_{j=1}^{K} P_j[(1-\alpha)\ell_s\{\langle \boldsymbol{f}^{*}, W_j\rangle\} + \alpha\ell_d\{\langle \boldsymbol{f}_0^{*}, W_j\rangle\}] \\
=&\epsilon(1-\alpha)\{-P_1\ell_s'\{\langle \boldsymbol{f}^*, W_1\rangle\} + P_2\ell_s'\{\langle \boldsymbol{f}^*, W_2\rangle\}\} + o(\epsilon) < 0
\end{aligned}$$

We can see that this is a contradiction. This completes the proof.

3.6.2 Proof of property for imbalance setting

In this section, we will first proof that in imbalanced setting, MSVM will not have a intercept that diverges to infinity, then as a special case, we will prove that MDWSVM intercept will not diverge to infinity under imbalance setting.

Proof of Proposition 3.3.1

Assume observations $1, \ldots, K-1$ belong to the class $1, \ldots, K-1$ respectively, and observations $K, \ldots, n$ belong to class K.

$$\begin{aligned}
\text{Loss} =& \sum_{i=1}^{n} \ell_s\{\langle f(x_i), W_{y_i}\rangle\} \\
=& \sum_{i=1}^{K-1} \ell_s\{\langle f(x_i), W_i\rangle\} + \sum_{i=K}^{n} \ell_s\{\langle f(x_i), W_K\rangle\} \\
=& \sum_{i=1}^{K-1} \ell_s\{\langle x_i^T B, W_i\rangle + \langle \beta_0, W_i\rangle\} + \sum_{i=K}^{n} \ell_s\{\langle x_i^T B, W_K\rangle + \langle \beta_0, W_K\rangle\}
\end{aligned}$$

Now we prove that $\forall B \in \mathbb{R}^{p\times(K-1)}$,

$$\begin{aligned}
\langle \beta_0, W_K\rangle &< \sup_i |\langle x_i^T B, W_i\rangle| K + 1 \\
&< \sqrt{2C} K \max |x_{ij}| + 1.
\end{aligned}$$

We can use contradiction to prove it, if $\langle \beta_0, W_K\rangle > \sup_i |\langle x_i^T B, W_i\rangle| K + 1$, then $\ell_s\{\langle x_i^T B, W_K\rangle + \langle \beta_0, W_K\rangle\} = 0$ for all $i \in \{K, \ldots, n\}$ as $\langle x_i^T B, W_K\rangle + \langle \beta_0, W_K\rangle > 1$.

$$\text{Loss} = \sum_{i=1}^{n} \ell_s\{\langle f(x_i), W_{y_i}\rangle\} = \sum_{i=1}^{K-1} \ell_s\{\langle x_i^T B, W_i\rangle + \langle \beta_0, W_i\rangle\}.$$

Then

$$\frac{dL}{d\beta_0} = \sum_{i=1}^{K-1} l_s'\{\langle x_i^T B, W_i\rangle + \langle \beta_0, W_i\rangle\}W_i'.$$

Since

$$\langle \beta_0, W_K\rangle > \sup_i |\langle x_i^T B, W_i\rangle| K + 1,$$

one can get that $u_K = \langle x_K^T B, W_K\rangle + \langle \beta_0, W_K\rangle > 1$. Based on the property of W, we have $\sum_{i=1}^{K}\langle \beta_0, W_i\rangle = 0$.

Furthermore, it is easy to deduct that

$$\min\langle \beta_0, W_i\rangle < -\sup_i |\langle x_i^T B, W_i\rangle| \text{ for } i \in \{1, \ldots, K-1\}.$$

Then

$$\min u_i = \langle x_i^T B, W_i\rangle + \min\langle \beta_0, W_i\rangle < 0 \text{ for } i = 1, \ldots, K-1.$$

Thus we can choose $K-1$ different values $K_1, K_2, , \ldots, K_{K-1}$ from $1, \ldots, K-1$ such that

$$u_{K_1} \geq u_{K_2} \geq \ldots \geq 0 \geq \ldots \geq u_{K_{K-1}}.$$

Assume $i_0 = \max\{i, u_{K_i} < 1\}$, then

$$\frac{dL}{d\beta_0} = -\sum_{i=K_{K-1}}^{K_{K_{i_0}}} W_{K_i}'.$$

One can simply verify that $\frac{dL}{d\beta_0} \neq 0$ based on the property of vertex W. Thus β_0 cannot be the β_0 that minimize the loss function given B. This step completes the prove.

Proof of Theorem 3.3.2

Theorem 3.3.2 can be viewed as a special case of Proposition 3.3.1.

Based on the proof of Proposition 3.3.1, for any $\forall B \in \mathbb{R}^{p\times(K-1)}$,

$$\langle \beta_0, W_K\rangle < \sup_i |\langle x_i^T B, W_i\rangle| K + 1 < \sqrt{2C}K \max|x_{ij}| + 1.$$

Thus for MDWSVM, no matter what B we get, the intercept only comes from MSVM part. Therefore, using the conclusion from Proposition 3.3.1, Theorem 3.3.2 is easily obtained.

4. INDIVIDUALIZED TREATMENT RULE FOR HDLSS DATA

4.1 Overview

Because of individual heterogeneity in genes, environment, and lifestyle, treatment effects differ within a population. A treatment that works effectively for a subgroup of patients with certain characteristics may be ineffective for another subgroup of patients with different characteristics. This phenomenon has been identified in numerous disease studies, including Campbell and Polyak [2007], Crossley [2003], and Lu et al. [2014].

Precision medicine takes this individual heterogeneity into consideration by assessing the expected value of each treatment assignment for a given set of patient characteristics, and determining the optimal individualized treatment rule (ITR). Numerous methods have been proposed to estimate the optimal ITR for single-stage protocols (i.e., one time treatment decision) [Cui et al., 2017, Gunter et al., 2011, Liang et al., 2018, Murphy, 2003, Qian and Murphy, 2011, Zhao et al., 2012, 2014] and multi-stage protocols (i.e., a sequence of treatment decisions) [Moodie et al., 2007, Orellana et al., 2010, Robins, 2004, Zhao et al., 2009].

For single-stage protocols, classification methods are widely used to determine the optimal ITR. Zhao et al. [2012] proposed outcome weighted learning (OWL). OWL, as described in Section 2.3, uses a weighted classification function to predict the optimal treatment for each patient, where the weights are proportional to the outcomes/rewards. There are, however, some limitations with OWL. First, the proposed approach is not designed for there to be negative rewards. To get around this, Zhao et al. [2012] proposed all the rewards be shifted to positive values by adding a constant. Zhou et al. [2017], however, argued that the choice of constant can effect the estimate of the optimal ITR. In addition, a classification method with positive weights tends to predict the treatment that is the same as the one originally received [Zhou et al., 2017].

To handle these problems, Zhou et al. [2017] proposed a residual weighted learning, which involves two steps. In Step 1, a linear regression is used to estimate the outcome using covariates, and the residual for each outcome is calculated. In Step 2, the residuals are treated as the outcomes in OWL. To handle the negative residuals, they proposed using a smoothed ramp loss function. The smoothed ramp loss is a non-convex loss, and as a result, may have many local minimum and stationary points. In other words, the global minimum can not be guaranteed.

To handle the multiple treatment setting, Zhang et al. [2018] extended binary OWL into a multicategory angle-based approach assuming each patient benefits the most from only one of the treatments. They use a generalized large margin loss function as the alternative to 0-1 loss, where hinge loss and DWD loss are two special cases. In contrast, Liang et al. [2018] use a deep learning method and consider the situation where a patient could benefit most from either a single treatment, or a combination of treatments at one time.

In the classification framework, weights are often included to handle uneven training group sizes [Huang and Du, 2005]. For the binary classification with uneven group sizes, Qiao and Zhang [2015a] pointed out DWD is sensitive to uneven group sizes while SVM can handle this imbalanced setting. This was later confirmed in the multicategory setting [Sun et al., 2017, Zhang and Liu, 2014]. However, Qiao et al. [2010] discovered the weighted SVM method has worse performance compared to the weighted DWD method. They also demonstrate the theoretical properties of weighted DWD on high dimensional, low sample size data.

In this chapter, motivated by the impact of weights on SVM and DWD when there are imbalanced group sizes, we conduct an exploration on how this affects OWL using both SVM and DWD loss (i.e., SVM-OWL and DWD-OWL) in both the binary and multicategory treatment settings. Through simulation, we show that in a balanced group setting, DWD-OWL and SVM-OWL performs similarly. However, in the imbalanced group setting, DWD-OWL is superior to SVM-OWL.

The remainder of the chapter is organized as follows. In Section 4.2, we review SVM-OWL and introduce our DWD-OWL. We also describe how to generalize each of them to handle negative rewards and multiple treatments. In Section 4.3, we provide some theoretical properties of DWD-OWL, including Fisher consistency and excess risk. In Section 4.4, some numeric simulations are provided to compare the performance of DWD-OWL to SVM-OWL in both the binary and multicategory treatment settings. The proofs of the theorems are provided in Section 4.5.

4.2 Methodology

In this section, we review outcome weighted learning using hinge loss (SVM-OWL) and introduce our DWD-OWL. Then we describe the ways to extend both SVM-OWL and DWD-OWL procedures to handle negative rewards and multiple treatments.

4.2.1 Outcome Weighted Learning

As discussed in Section 2.3, given the distribution of trajectories (X, A, R), the optimal ITR is the one maximizing $E^{\mathcal{D}}(R)$ [Qian and Murphy, 2011]

$$\mathcal{D}^* = \arg\max_{\mathcal{D}} E^{\mathcal{D}}(R) = \arg\max_{\mathcal{D}} E(\frac{\mathbb{I}\{A = \mathcal{D}(X)\}}{P(A|X)}R)$$

Zhao et al. [2012] proposed using the OWL framework to find this optimal ITR, arguing the equivalence of maximizing the expected reward and minimizing the reward loss

$$\arg\max_{\mathcal{D}} E(\frac{\mathbb{I}\{A = \mathcal{D}(X)\}}{P(A|X)}R) = \arg\min_{\mathcal{D}} E(\frac{\mathbb{I}\{A \neq \mathcal{D}(X)\}}{P(A|X)}R). \tag{4.1}$$

Therefore, for a decision function $\mathcal{D}(\boldsymbol{x}) = \text{sign} f(\boldsymbol{x})$, the optimal ITR $\mathcal{D}(\boldsymbol{x})$ becomes

$$\mathcal{D}^* = \arg\min_{\mathcal{D}} E(\frac{R}{P(A|X)}\mathbb{I}\{Af(X) < 0\}). \tag{4.2}$$

The 0-1 loss in (4.2) is computationally expensive to directly solve and so Zhao et al. [2012] proposed using the hinge loss $\ell(u) = (1-u)_+$. From now on we will refer to this approach as SVM-OWL. Given the observations $(\boldsymbol{x}_i, a_i, r_i)$, for $i = 1, \dots, n$, the corresponding empirical version of their objective function is

$$\arg\min_f \frac{1}{n} \sum_{i=1}^{n} \frac{r_i}{P(a_i|\boldsymbol{x}_i)} \ell\{a_i f(\boldsymbol{x}_i)\} + \lambda_n J(f), \tag{4.3}$$

where $\ell(.)$ is the hinge loss. The $P(a_i|\boldsymbol{x}_i)$ is the conditional probability of decision assigning treatment a_i based on subject i's characteristics. $J(f)$ is used to control the overfitting and λ_n is the tuning parameter. Note that this method can be viewed as a classification method with weight $R/P(A|X)$. In most randomized clinical trials, $P(A|X) = P(A)$ so the weight is proportional to outcome R. This is why it is called outcome weighted learning.

4.2.2 Binary DWD-OWL

Under the OWL framework, numerous loss functions can be used. Liu et al. [2011] proposed a family of loss functions

$$\ell(u) = \begin{cases} 1-u & \text{if } u < \dfrac{c}{1+c} \\ \dfrac{1}{1+c}\Big(\dfrac{b}{(1+c)u - c + b}\Big)^b & \text{Otherwise} \end{cases} \tag{4.4}$$

where $c \geq 0$ determines the connection point of the two pieces, and $b > 0$ is used to control the shape of the loss function when $u > \frac{c}{1+c}$. For $b > 0$ and $c \to +\infty$, this loss function becomes hinge loss. For $b = 1$ and $c = 1$, the loss function becomes DWD loss.

The loss functions of DWD and SVM, along with 0-1 loss are presented in Figure 4.1. Even though the difference between the loss functions for SVM and DWD are subtle, the two methods weight points that are correctly assigned differently. SVM use a weight of 0 for all correctly assigned points. DWD, on the other hand, assigns positive weights to all points, such that points closer to the separating hyperplane carry more weight than those further away. Additional details regarding SVM and DWD can be found in Sections 2.1.2 and 2.1.3.

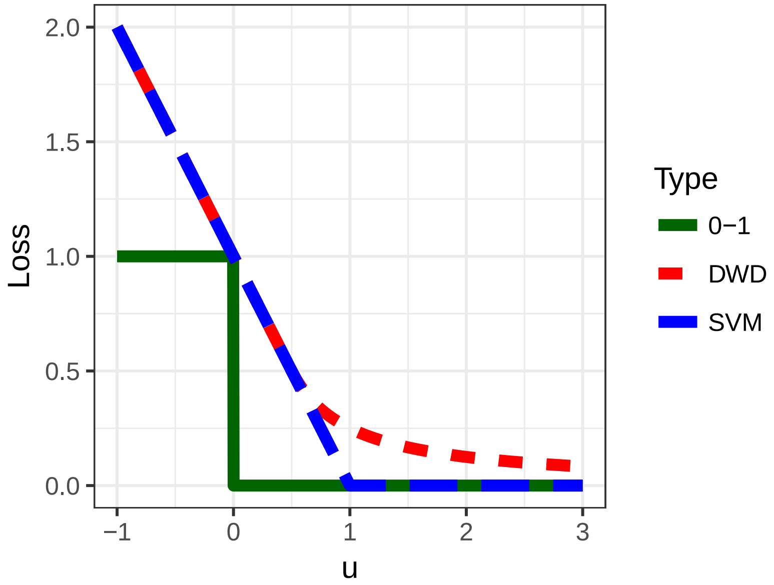

Figure 4.1. Loss function for SVM, DWD and 0-1 loss

In an unweighted classification setting, Qiao and Zhang [2015a] observed that DWD is sensitive to imbalanced group size, while SVM is not. A similar pattern was observed by Sun et al. [2017] in the multicategory classification setting. In contrast, by adding inverse sample size as a weight, Qiao et al. [2010] demonstrates that in binary classification, weighted DWD obtains much better accuracy compared to weighted SVM.

Building on this observation, we propose using DWD loss in the OWL setting (DWD-OWL). This means the empirical objective function is

$$\arg\min_{f} \frac{1}{n} \sum_{i=1}^{n} \frac{r_i}{P(a_i|\boldsymbol{x}_i)} \ell\{a_i f(\boldsymbol{x}_i)\} + \lambda_n J(f), \tag{4.5}$$

where $\ell(.)$ has form

$$\ell(u) = \begin{cases} 1-u & \text{if } u < 1/2 \\ 1/(4u) & \text{Otherwise} \end{cases}. \tag{4.6}$$

$J(f)$ is the regularization item used to avoid overfitting, and λ_n is its corresponding tuning parameter. Note that the loss function in (4.6) is a special case of (2.4) with $C = 4$.

DWD-OWL is a non-trivial alternative of SVM-OWL. For SVM-OWL, the correctly assigned points have no impact on the decision boundary estimation. For DWD-OWL, the correctly assigned points are each given a large penalty, and thus plays a very important role in estimating the decision boundary.

4.2.3 From positive reward to negative reward

To ensure the convexity of the optimization in OWL, the weights $r_i/P(a_i|\boldsymbol{x}_i)$ are assumed positive. In practice, however, there are cases where getting the wrong treatment may be harmful, resulting in a negative reward [Solberg et al., 2005]. Zhao et al. [2012] propose the indirectly approach of shifting the rewards so they are all positive. Issues with this approach are discussed in Section 2.3. To directly handle negative rewards, we adopt the idea of Liu et al. [2016] and propose a "mirror projection." Note that we can rewrite the general ITR objective function as follows:

$$\begin{aligned} & E\Big(\frac{R}{P(A|X)}\mathbb{I}\{A \neq \text{sign}\big(f(X)\big)\}\Big) \\ =& E\Big(\frac{|R|}{P(A|X)}\mathbb{I}\{A\text{sign}(R) \neq \text{sign}\big(f(X)\big)\}\Big) + E\Big(\frac{R_-}{P(A|X)}\Big) \end{aligned} \tag{4.7}$$

where $R_- = R$ if $R \leq 0$ and 0 otherwise. This equation is obviously correct when all $R \geq 0$. If all $R \leq 0$, then

$$E\Big(\frac{|R|}{P(A|X)}\mathbb{I}\{A\text{sign}(R) \neq \text{sign}\big(f(X)\big)\}\Big) = -E\Big(\frac{R}{P(A|X)}\mathbb{I}\{A = \text{sign}\big(f(X)\big)\}\Big),$$

again verifying the equation.

Because the second term of (4.7), $E\left(\frac{R_-}{P(A|X)}\right)$, does not depend on the decision function $f(X)$, we modify DWD-OWL to finding

$$\arg\min_{\mathcal{D}} E\Big(\frac{|R|}{P(A|X)}\ell\big(\text{sign}(R)Af(X)\big)\Big) = \arg\min_{\mathcal{D}} E\Big(\frac{|R|}{P(A|X)}\ell_r\big(Af(X)\big)\Big)$$

where $\ell_r(u) = \ell(u)$ if $r > 0$ and $\ell_r(u) = \ell(-u)$ otherwise. The loss function ℓ_r is shown in Figure 4.2 assuming DWD loss. As $R < 0$ decreases, $|R|$ increases. Because we want ℓ_r to be as small as possible, we want $u = af(\boldsymbol{x}) < 0$. In other words, we want to predict the other treatment, not the one received.

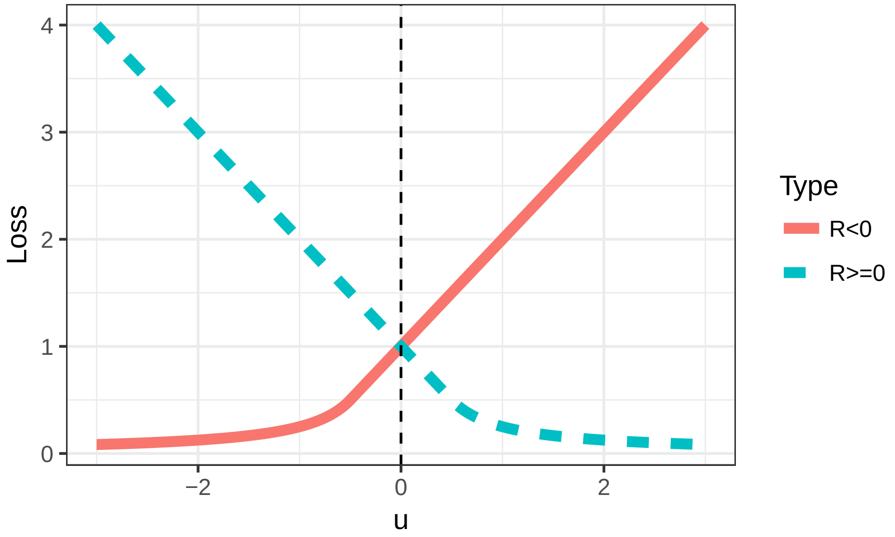

Figure 4.2. Loss function $\ell_r(u)$

The corresponding empirical objective function for DWD-OWL allowing for negative rewards is

$$\min_{\boldsymbol{f}\in\mathbb{F}} \frac{1}{n}\sum_{i=1}^{n} \frac{|r_i|}{P(a_i|\boldsymbol{x}_i)}\ell_r(a_i f(\boldsymbol{x}_i)) + \lambda_n J(\boldsymbol{f}). \tag{4.8}$$

And the optimal ITR is

$$\hat{\mathcal{D}}^*(\boldsymbol{x}) = \text{sign}(\hat{f}(\boldsymbol{x}))$$

This mirror projection allows negative outcomes to be directly implemented into objective function yet does not result in a different optimal ITR if all the rewards are positive. In addition, with only positive rewards, incorrect assignments will have small positive rewards and thus not carry much weight in the objective function. Having negative rewards implies both correct and incorrect treatment assignments will factor into the minimization.

Zhou et al. [2017] proposed centralizing the rewards to take advantage of this property even when all the rewards are positive. This involved subtracting a prediction function $m(X)$ from each observed reward R. They assumed that the reward is a linear combination of an effect due to just the covariates and the interaction between treatment and covariates. In terms of an equation, they assume

$$R = m(X) + Af(X).$$

By subtracting the marginal function $m(X)$ from R, the remaining part $Af(X)$ is the interaction of covariates and treatment, which is the source for reward differences due to treatment assignments. In their paper, $m(X)$ was estimated by a weighted linear regression, minimizing $E(R - \frac{m(X)}{P(A|X)})^2$. In addition to using this function, they also proposed simply subtracting the weighted average $m(X) = E(R/P(A|X))$. In the remainder of this chapter, to avoid the noise introduced by estimating $m(X)$ using covariates X, we will use the weighted mean.

4.2.4 From binary OWL to multicategory OWL

We now consider the multicategory setting with K different treatment, $A = \{1, 2, \ldots, K\}$. Based on the discussion from Section 2.2 and 3.2, one way to build an efficient multicategory classifier is to use the angle-based framework of Zhang and Liu [2014]. Zhang et al. [2018] extend the binary OWL into multicategory OWL by implementing the angle-based framework. Both SVM-OWL and DWD-OWL are special cases of their proposal.

Within the angle-based framework, the empirical objective function for multicategory DWD-OWL is

$$\min_{\boldsymbol{f}\in\mathbb{F}} \frac{1}{n}\sum_{i=1}^{n} \frac{|r_i|}{P(a_i|\boldsymbol{x}_i)}\ell_r(\langle \boldsymbol{f}(\boldsymbol{x}_i), W_{a_i}\rangle) + \frac{\lambda_n}{2}J(\boldsymbol{f}). \tag{4.9}$$

where the inner product $u = \langle \boldsymbol{f}(\boldsymbol{x}_i), W_{a_i}\rangle$ can be viewed as a new functional margin of $(f(\boldsymbol{x}), a)$. The loss function $\ell_r(u)$ is the mirror projection loss described in (4.8). In using this loss function, the reward r_i could be the outcome or the centered outcome. $P(a_i|\boldsymbol{x}_i)$ is the propensity score commonly estimated by logistic regression if unknown, and λ_n is the tuning parameter and $J(f)$ is the regularization term.

Under angle-based DWD-OWL, the optimal ITR is defined as

$$\hat{\mathcal{D}}^*(\boldsymbol{x}) = \arg\max_{j\in\{1,2,\ldots,K\}} \langle \hat{\boldsymbol{f}}(\boldsymbol{x}_i), W_j\rangle$$

In binary DWD-OWL, $W_1 = 1$ and $W_2 = -1$ the method (4.9) becomes identical to (4.8). Thus, the binary case is also the special case for method (4.9).

4.3 Theoretical Properties

In this section, we establish the Fisher consistency of DWD-OWL. Then we show that the loss of the value function under 0-1 loss can be bounded by the excess risk under DWD loss. Later, we demonstrate that when the optimal assignment is imbalanced, DWD-OWL is unaffected.

4.3.1 Fisher consistency

Fisher consistency is one of the most fundamental properties for classification. It guarantees that when the sample size goes to infinity, the best decision function $f^*(\boldsymbol{x})$ is the one maximizing the expected outcome. To demonstrate this, we will first introduce some notation.

ITR $\mathcal{D}(\boldsymbol{x}) = \text{sign}(f(\boldsymbol{x}))$ associated with decision function $f(\boldsymbol{x})$. The risk associated with f is

$$\mathcal{R}(f) = E[\frac{R}{P(A|X)}\mathbb{I}\{A \neq \text{sign}\big(f(X)\big)\}],$$

and the minimal risk (i.e., Bayes risk) is

$$\mathcal{R}^* = \mathcal{R}(f^*) = \min_f \{\mathcal{R}(f) | f : \mathcal{X} \to \mathbb{R}\}.$$

Note that $f^*(\boldsymbol{x}) = \min_f \mathcal{R}(f)$, and the optimal ITR for 0-1 loss is

$$\mathcal{D}^*(\boldsymbol{x}) = \text{sign}(f^*(\boldsymbol{x}))$$

For DWD-OWL, we replace 0-1 loss with the surrogate DWD loss. Define

$$\mathcal{R}_\ell(f) = E[\frac{R}{P(A|X)}\ell(Af(X))],$$

and

$$\mathcal{R}_\ell^* = \inf_f \{\mathcal{R}_\ell(f) | f : \mathcal{X} \to \mathbb{R}\}.$$

If $f^*(\boldsymbol{x}) = \min_f \mathcal{R}_\ell(f)$, then the ℓ loss OWL is Fisher consistent.

Theorem 4.3.1 *For rewards $R \in \mathbb{R}^+$, the binary DWD-OWL classifier is Fisher consistent.*

Notice that Fisher consistency here is determined only for positive outcomes. In fact, for binary OWL, if we replace DWD loss with any surrogate loss that has $\ell'(u) < 0$, this property still holds. See the detailed proof in Section 4.5.

Theorem 4.3.2 *For rewards $R \in \mathbb{R}$, the binary DWD-OWL classifier is Fisher consistent.*

Under some mild conditions. We can also consider the Fisher consistency for multicategory setting.

Theorem 4.3.3 *Consider patient information $\boldsymbol{x}$, and assume the best treatment for this patient is j. Under the condition that*

$$\int_{R<0}(R|X=\boldsymbol{x},A=j)dP > \int_{R<0}(R|X=\boldsymbol{x},A=i)dP$$

for any $i \neq j$. The angle-based multicategory DWD-OWL classifier is Fisher consistent.

The condition mentioned in Theorem 4.3.3 requires that the expected negative outcome under the optimal treatment is going to be larger than the expected negative outcome under the other treatments. This can be easily satisfied in practice. When the outcomes are all positive or all negative, this condition is met. When the marginal distribution of rewards are identical for different treatments except for a shift, the condition is also met.

Fisher consistency guarantees that in a population, the surrogate DWD loss will obtain the optimal decision rule. Zhao et al. [2012] prove the Fisher consistency of SVM-OWL under a binary treatment setting. However, in multicategory OWL, the Fisher consistency of SVM-OWL cannot be guaranteed [Zhang et al., 2018]. This is an important distinction between the two approaches.

4.3.2 Excess risk for $\mathcal{R}(f)$ and $\mathcal{R}_\ell(f)$

For any measurable $f : \mathcal{X} \to \mathbb{R}$, the excess risk of f is the amount by which the risk of f exceeds the Bayes risk. The excess risk under 0-1 loss is $\mathcal{R}(f) - \mathcal{R}^*$ and the excess risk under ℓ loss is $\mathcal{R}_\ell(f) - \mathcal{R}_\ell^*$.

Theorem 4.3.4 *In the binary DWD-OWL method with reward $R \in \mathbb{R}^+$, for any measurable $f : \mathcal{X} \to \mathbb{R}$ and any probability distribution for (X, A, R),*

$$\mathcal{R}(f) - \mathcal{R}^* \leq \mathcal{R}_\ell(f) - \mathcal{R}_\ell^*.$$

This result states that in the binary treatment setting, for any decision function f, the excess risk under 0-1 loss is going to be smaller than the excess risk under

DWD loss. Therefore, the loss of the value function respect to f is bounded by the excess risk under the DWD loss. It implies that if the excess risk of f under the ℓ loss is really small, then the risk under f is going to be close to Bayes risk. We conjecture this is also true for the multicategory setting and plan to investigate it in the near future.

4.3.3 Properties under the imbalanced optimal treatment setting

In this section, we investigate the performance of DWD-OWL when group sizes are imbalanced. As we've done previously, we address the binary setting and leave the multicategory setting as future work. The asymptotic setting we focus on is that the minority group size n_+ is fixed and the majority sample size $n_- \to \infty$. This is similar to the setting in Qiao and Zhang [2015b] for classification problems.

They proved that under the imbalanced setting, DWD will classify all the observations to the majority class. In other words, the minority class will be 100% misclassified. This can be also observed in Figure 2.4.

Under the weighted DWD setting, let $\bar{\boldsymbol{x}}_+^T$ be the sample mean of the minority class. The following theorem states that the intercept term for weighted DWD will not converge to $-\infty$.

Theorem 4.3.5 *In binary classification, let the minority group size n_+ being fixed and the majority group size n_- going to infinity, the intercept $\hat{\beta}$ for weighted DWD, with weight $1/n_-$ for majority group and $1/n_+$ for minority group, does not go to $-\infty$ but rather is bounded*

$$\hat{\beta} > -\frac{1}{2} - \bar{\boldsymbol{x}}_+^T \boldsymbol{w}.$$

Under some mild conditions, we can also show that this is true for DWD-OWL in the binary treatment setting. Defining

$$\eta(\boldsymbol{x}) = \frac{E(R|X = \boldsymbol{x}, A = 1)}{E(R|X = \boldsymbol{x}, A = 1) + E(R|X = \boldsymbol{x}, A = -1)}. \tag{4.10}$$

Note that $\eta(\boldsymbol{x}) > 1/2$ if Treatment 1 is the optimal treatment for $\boldsymbol{x}$, and $\eta(\boldsymbol{x}) < 1/2$ if Treatment -1 is the optimal treatment for $\boldsymbol{x}$. We only consider the setting where $P(A|X) = 1/2$. Defining

$$\mathcal{X}_{++} = \{(\boldsymbol{x}, a) : 2\eta(\boldsymbol{x}) - 1 > 0\&a = 1\},$$

$$\mathcal{X}_{+-} = \{(\boldsymbol{x}, a) : 2\eta(\boldsymbol{x}) - 1 > 0\&a = -1\},$$

$$\mathcal{X}_{-+} = \{(\boldsymbol{x}, a) : 2\eta(\boldsymbol{x}) - 1 \leq 0\&a = 1\},$$

and

$$\mathcal{X}_{--} = \{(\boldsymbol{x}, a) : 2\eta(\boldsymbol{x}) - 1 \leq 0\&a = -1\}$$

where $\eta(\boldsymbol{x})$ is defined in Equation (4.10). The corresponding sizes for each subset are n_{++}, n_{+-}, n_{-+} and n_{--}, respectively. The total sample size is denoted n. Let n_+ be the fixed group size for subjects that benefit more from Treatment 1, and n_- be the group size for subjects that benefit more from Treatment -1. Note that $n = n_+ + n_- = n_{++} + n_{+-} + n_{-+} + n_{--}$.

Theorem 4.3.6 *In single-stage precision medicine setting, let the minority group size n_+ being fixed and the majority group size n_- going to infinity, if*

$$(\frac{\sum_{\mathcal{X}_{--}} r_i}{(\sum_{\mathcal{X}_{-+} \cup \mathcal{X}_{++}} r_i - \sum_{\mathcal{X}_{+-}} r_i)} | \eta(\boldsymbol{x}) < \frac{1}{2}) < \gamma$$

where $\gamma < \infty$, then the DWD-OWL intercept $\hat{\beta}$ will not go to $-\infty$ but be bounded.

Theorem 4.3.6 implies that DWD-OWL will not assign treatment -1 to all the subjects. The mild condition is readily met as the outcome for both correctly assigned subjects and incorrectly assigned subjects are bounded.

Similar to this, we conjecture that in multicategory case, under mild conditions, the DWD-OWL will not assign all the subjects to the treatment that benefits most people, the proof of this conjecture will be our immediate future work.

4.4 Simulation Study

In this section, we compare the performance of SVM-OWL and our DWD-OWL under various degrees of imbalance. There are actually two types of imbalance in a single-stage precision medicine setting. An observed treatment group imbalance, and an optimal treatment group imbalance.

An example of the former is when a new drug is assigned to a larger proportion of participants compared to the reference drug, e.g $P(A = 1|X) = 0.75$. Under the OWL method, the inverse propensity score $1/P(A|X)$ takes care of this imbalance by down-weighting the dominant new drug group. This type of imbalance is not a problem for either SVM-OWL or DWD-OWL.

An example of an optimal treatment imbalance is when the optimal treatment for most of the participants is the new drug. This imbalance will not be observed except through the rewards. Many real world examples have optimal treatment imbalance. This type of imbalance is our focus of this section.

Without loss of generality, we consider simulation scenarios where the observed treatment assignment is balanced. Only the optimal treatment group sizes vary. In the following three simulation settings, each experiment is repeated 100 times and the average performance for each method, along with the standard error, are presented. The empirical measurements of performance are

- Value [Zhao et al., 2012]

$$V = \frac{E_n[\frac{R}{P(A|X)} I(A = D(X))]}{E_n(\frac{1}{P(A|X)})}$$

- Overall accuracy [Zhao et al., 2012]

$$OA = E_n[I(D(X) = A_{opt})]$$

- Mean-within-group accuracy

$$MWGA = \frac{1}{K} \sum_{k=1}^{K} E_n[I(D(X) = A_{opt})|A_{opt} = k]$$

where $E_n(Z) = \frac{1}{n}\sum_{i=1}^{n} z_i$. For all three measurements, a larger value represents a better performance. OA measures the overall accuracy of the prediction compared to the optimal treatment. It is the opposite of the overall misclassification error measured in Zhao et al. [2012]. $MWGA = 1 - MWGE$ [Qiao et al., 2010] is a weighted version of the accuracy, giving the same weight to different optimal treatment groups. Among these three, the weighted expected value V and overall misclassification error $1 - OA$ are commonly used in ITR research [Liang et al., 2018, Zhao et al., 2012, Zhou et al., 2018]. For the imbalanced setting, MWGA is an important measurement as it gives more weight to the observations from the minority group. In practice, the optimal treatment is unknown and cannot be observed. Thus both OA and MWGA cannot be calculated in practice.

We considered a binary treatment setting and two multicategory treatment settings. For each setting, the reward and centered reward are considered. In the summary figures, DWD-OWL and SVM-OWL with centered outcome are referred to as CDWD-OWL and CSVM-OWL, respectively. In all replications, the training data size is fixed as 300, and another data set of size 300 is used to tune the parameter λ_n. The tuning parameter λ_n varies among $(2^{-8}, 2^{-7}, \ldots, 2^{8})$. A data set of 30000 observations serves as the testing data. Each summary in the table is reported as a average of 100 replicates.

4.4.1 Binary treatment setting

The covariates $X \in \mathbb{R}^d$ are assumed $X_j \overset{\text{i.i.d.}}{\sim} Unif(-1, 1)$ for $j = 1, ..., d$. We randomly assign treatments 1 and -1 to the subjects with $P(A = 1) = 0.5$. The reward $R \sim N(\mu, 1)$ where $\mu = 8.0 + 2X_1 + X_2 + 0.5X_3 + 2A(m - X_1 - X_2)$. The constant m is used to control the imbalance. When $m = 0$, the two optimal treatment groups are of equal size. In our figures, we show results for considering $m = 0.0, 0.5, 1.0$, corresponding to balanced, moderately imbalanced and extremely imbalanced settings. The values for dimension d considered are $(5, 10, 50, 100, 150, 200)$.

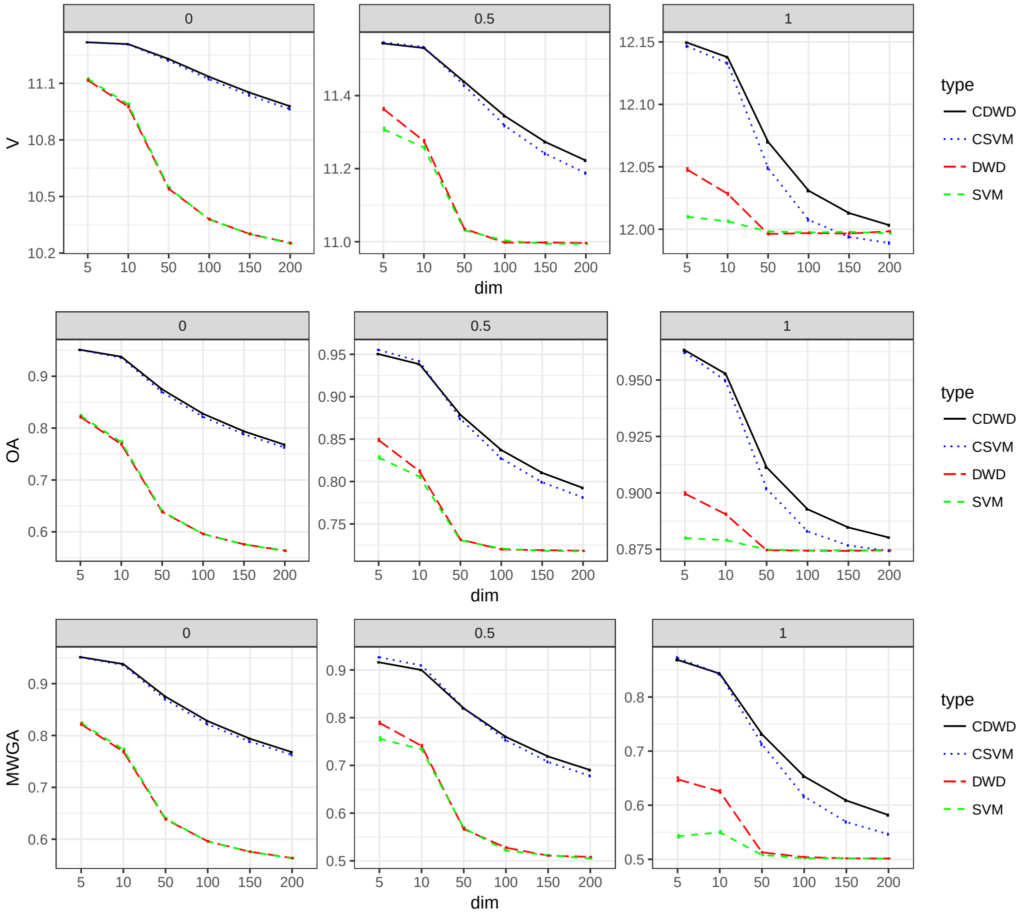

Figure 4.3. Training data for the binary treatment setting when $m = 0.5$ and $d = 3$. Top left subplot is the optimal treatment setup on the projection of data to the first 2 dimensions. Bottom left and top right are the observed rewards for subjects with treatment 1 and -1 respectively. Bottom right subplot is the rewards for all subjects with Treatments 1(triangle) and -1(dots).

Figure 4.3 is an illustration of the training data. For the area where $A_{opt} = 1$, the reward when assign Treatment 1 (represented with a triangle) is larger than when assigned Treatment -1 (represented with a dot). This setting is a typical representation for a real binary clinical trial. For the observations near the boundary of the optimal treatments, there is not much difference on whether it is assigned to Treatment 1 or -1. However, for observations that are far from the boundary, the difference in rewards becomes larger. Thus a wrong assignment carries a larger cost.

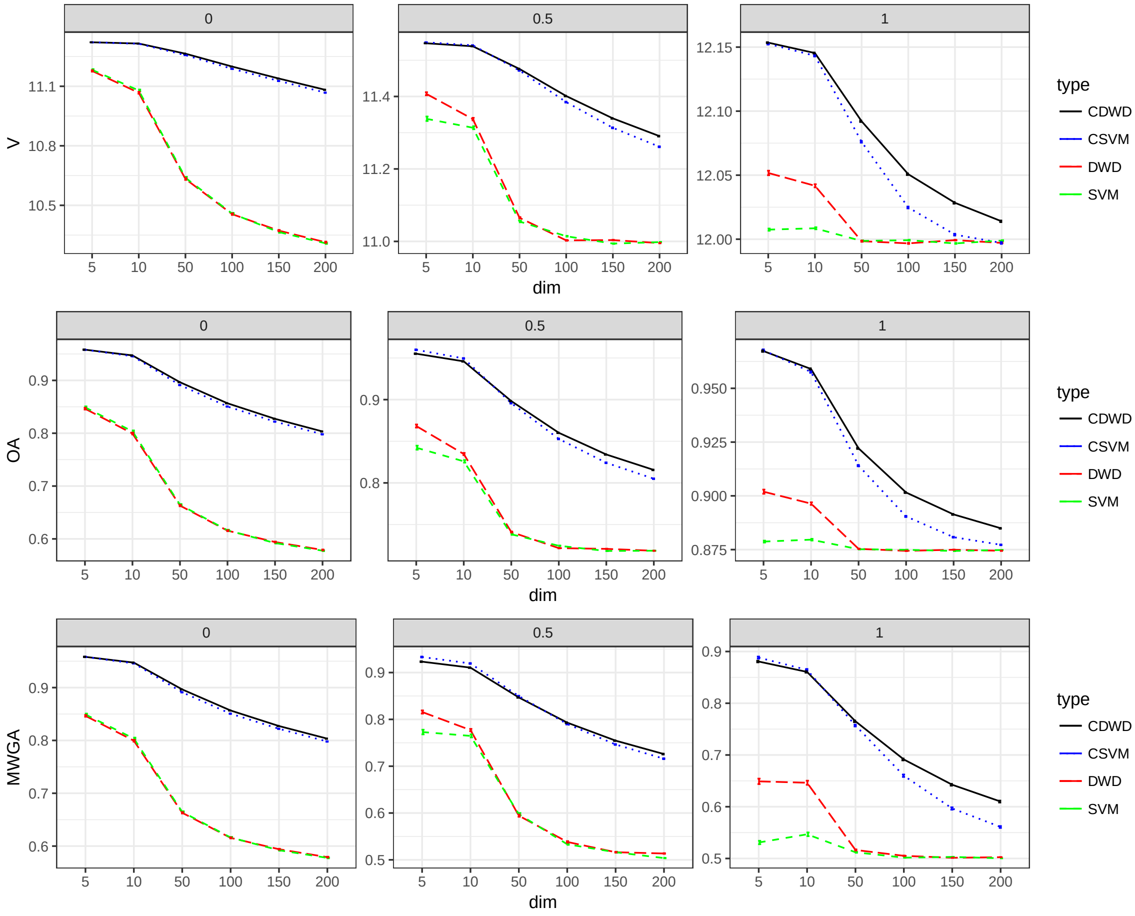

Figure 4.4. Comparison of performance using CDWD-OWL, CSVM-OWL, DWD-OWL and SVM-OWL in the binary treatment setting. In each panel of the figure, the x-axis represents the number of covariates and y represents the performance measures V, OA and $MWGA$ from top to bottom.

The performance of the four different methods are presented in Figure 4.4. Each panel represents the relationship between the number of prognostic variables and performance value (V), overall accuracy (OA) and mean-within-group accuracy (MWGA). The larger the value, the better the performance. As number of covariates increases, performance drops due to more noise added into the data. From left to right, the plots are corresponding to m being 0, 0.5 and 1 respectively. When $m = 0$, we are considering a balanced optimal treatment scenario. Notice that in this case, centered rewards based on DWD and SVM are not different. However, they outperform DWD-OWL and SVM-OWL using the uncentered rewards. This phenomenon is consistent with the conclusion on Zhou et al. [2017]. As the optimal treatment group size becomes moderately imbalanced, CDWD-OWL starts outperforming the other methods in terms of both estimated value and accuracy, especially in the higher dimensional cases. In the extremely imbalanced case, the discrepancy between CDWD-OWL and CSVM-OWL is more pronounced. In general, DWD-based OWL is superior to SVM-based OWL.

4.4.2 Multicategory treatment

In this section, we consider two muticategory treatment simulation settings.

Example 1 The covariates $X \in R^d$ are assumed $X_j \overset{i.i.d.}{\sim} Unif(-1, 1)$ for $j = 1, ..., d$. Treatments 1, 2, and 3 are randomly assigned to subjects with $P(A = k) = 1/3, k = 1, 2, 3$. The reward $R \sim N(\mu, 1)$ where $\mu = 8.0 + \mu_c$. In determining μ_c, we divide the covariate space into 6 areas:

$$C_1 = \{X_1 + X_2 > m \cap A = 1\}$$

$$C_2 = \{X_1 + X_2 > m \cap A \neq 1\}$$

$$C_3 = \{X_1 + X_2 < m \cap X_1 - X_2 > 0 \cap A = 2\}$$

$$C_4 = \{X_1 + X_2 < m \cap X_1 - X_2 > 0 \cap A \neq 2\}$$

$$C_5 = \{X_1 + X_2 < m \cap X_1 - X_2 < 0 \cap A = 3\}$$

$$C_6 = \{X_1 + X_2 < m \cap X_1 - X_2 < 0 \cap A \neq 3\}$$

such that μ_c is defined as

$$\mu_c = \begin{cases} 2(X_1 + X_2 - m) & \text{if } C_1 \\ -2(X_1 + X_2 - m) & \text{if } C_2 \\ -4(X_1 + X_2 - m)(X_1 - X_2) & \text{if } C_3 \cup C_6 \\ 4(X_1 + X_2 - m)(X_1 - X_2) & \text{Otherwise} \end{cases}.$$

The constant m is used to control the imbalance. In our figures, we show results for considering $m = -0.6, 0.0, 0.3, 0.6$, and 1.0. The first two m values represent the imbalanced setting where optimal Treatment 1 has larger group sizes than other two treatments, $m = 0.3$ represents the balanced setting, and the last two m values represent the imbalanced setting where optimal Treatments 2 and 3 have larger groups sizes than Treatment 1. The values for dimension d considered are (2, 5, 10, 50, 100, 200).

Figure 4.5 is the illustration of the training data for multicategory treatment setting. This is a non-trivial extension of the binary setting example. Different values of m determine the level of imbalance. As m decreases, more observations belong to the group with optimal Treatment 1. As m increases, more observations belong to the groups with optimal Treatments 2 or 3. The sizes for groups with optimal Treatments 2 and 3 are always equal.

The performance of the four different methods are presented on Figures 4.6 and 4.7. In Figure 4.6, similar to Figure 4.4, each panel represents the relationship between the number of prognostic variables and a particular performance measure. From left to right, the plots are corresponding to balanced ($m = 0.3$), moderately imbalanced ($m = 0.0$) and extremely imbalanced ($m = -0.6$) optimal treatment groups. For the imbalanced setting, only one of the groups have larger size for optimal treatment, the rest of the groups have equal size for optimal treatment. In other words, the imbalanced setting represents one majority group. The findings in this simulation setting are similar to the ones in the binary ITR setting. As the optimal treatment becomes more imbalanced, CDWD-OWL starts outperforming the rest of the methods

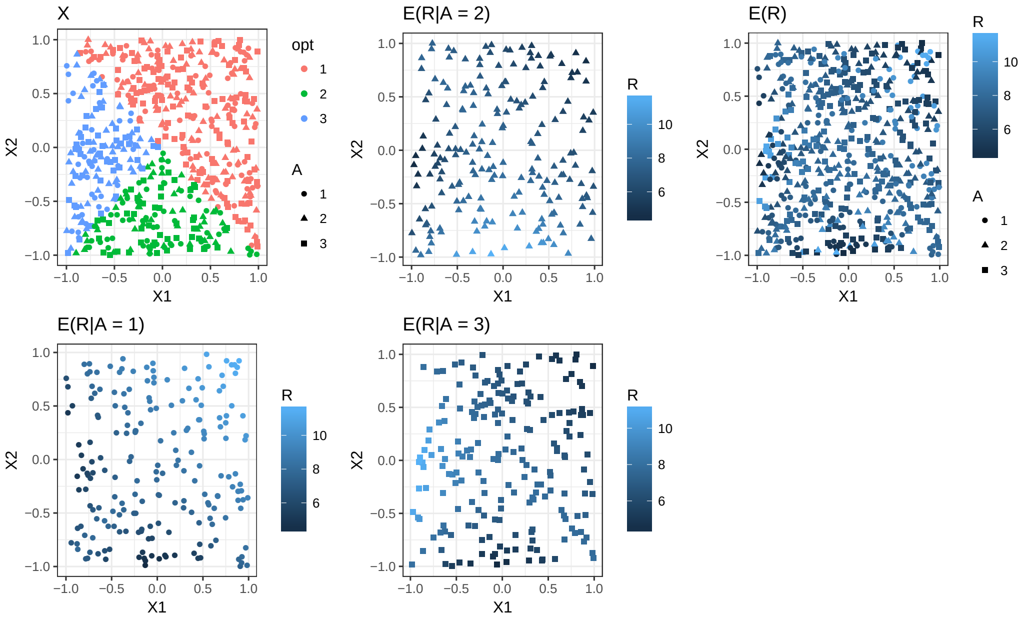

Figure 4.5. Training data for the multicategory treatment setting for Example 1 when $m = 0.0$ and $d = 2$. Top left subplot is the optimal treatment setup on the projection of data to the first 2 dimensions. Bottom left, top middle and bottom middle are the observed rewards for subjects with Treatment 1(represented with a dot), 2(represented with a triangle) and 3(represented with a square) respectively. Top right subplot is the rewards for all subjects with Treatment 1, 2, and 3.

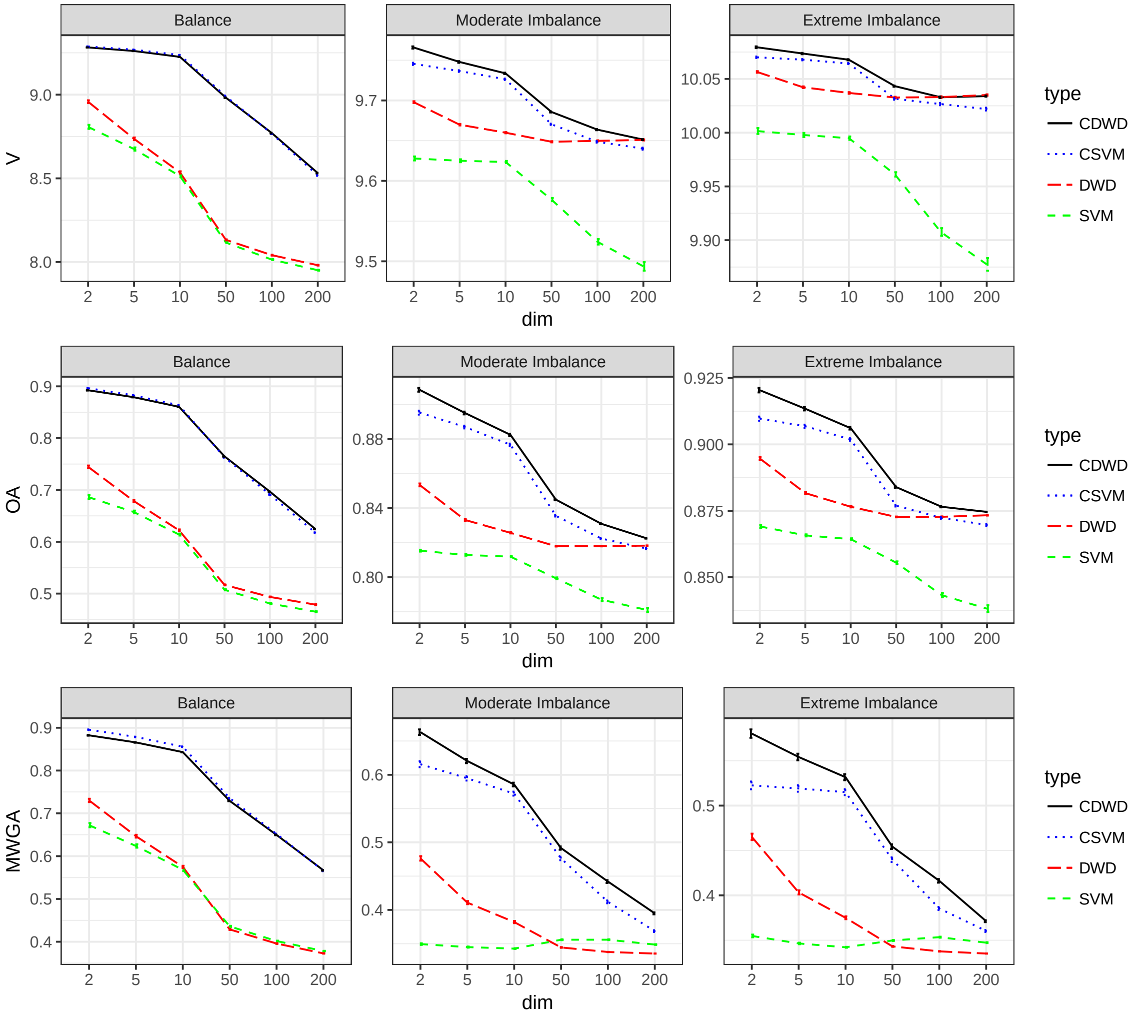

Figure 4.6. Comparison of performance using CDWD-OWL, CSVM-OWL, DWD-OWL and SVM-OWL in the 3 treatments settings. From left to right, the subplots represent the performance for balanced, moderately imbalanced, and extremely imbalanced optimal treatment setting, respectively. In each panel of the figure, the x-axis represents the number of covariates and y represents the performance measures V, OA and MWGA from top to bottom.

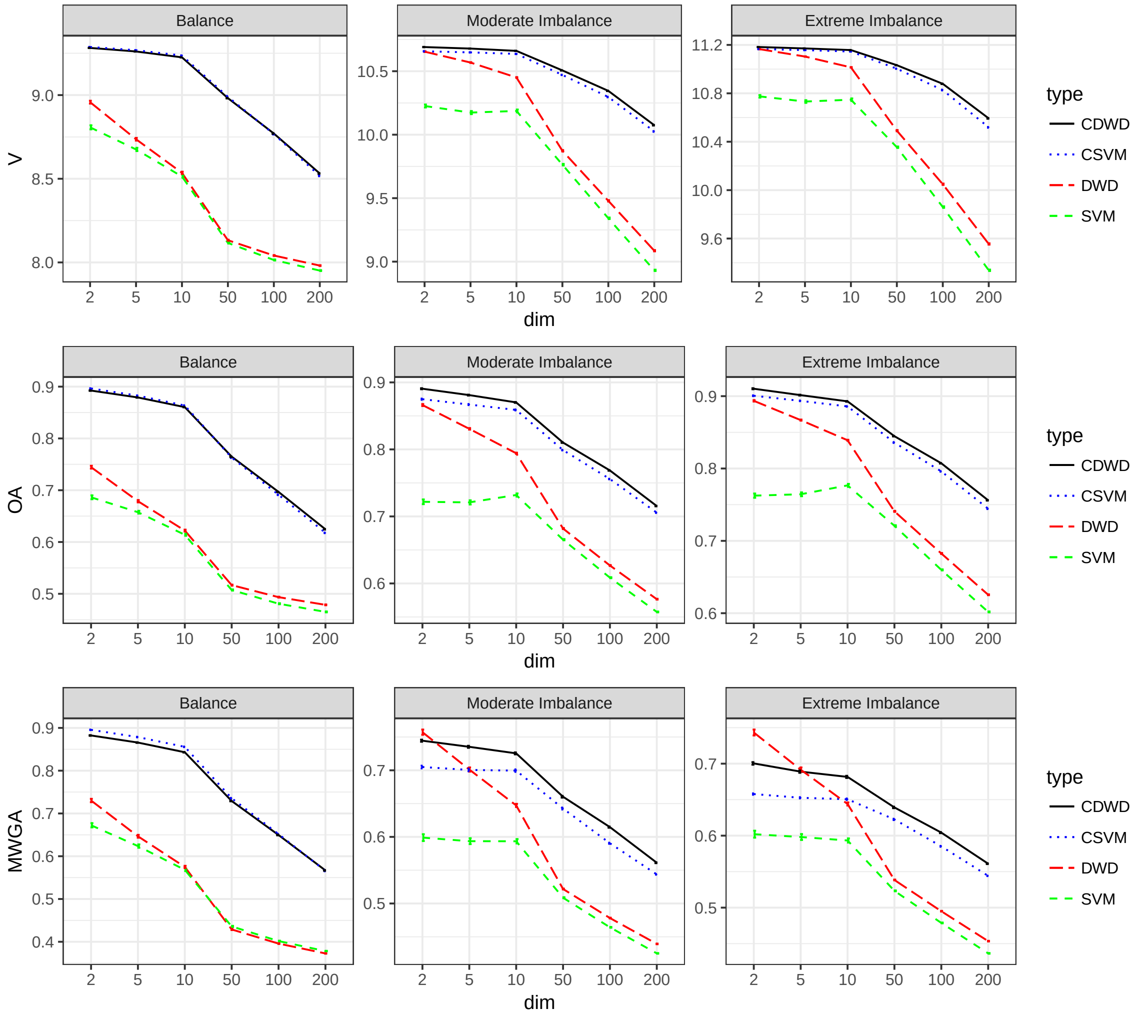

Figure 4.7. Comparison of performance using CDWD-OWL, CSVM-OWL, DWD-OWL and SVM-OWL in the 3 treatments settings. From left to right, the subplots represent the performance for balanced, moderately imbalanced, and extremely imbalanced optimal treatment setting, respectively. In each panel of the figure, the x-axis represents the number of covariates and y represents the performance measures V, OA and MWGA from top to bottom.

on both estimated value and accuracy. However in the balanced scenario, CDWD-OWL and CSVM-OWL give similar results regarding estimated value and accuracy. In general, DWD-based OWL is superior to SVM-based OWL.

Figure 4.7 represents the imbalanced setting where there are two majority groups. In this setting, we still observe the superiority of CDWD-OWL in contrast to CSVM-OWL, and the superiority of DWD-OWL over SVM-OWL.

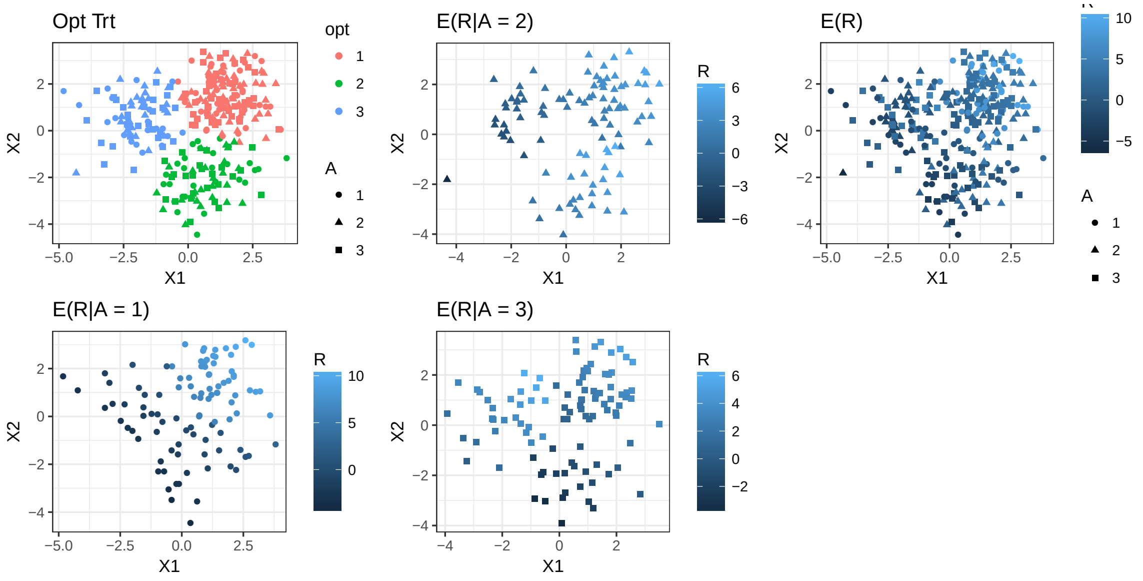

Figure 4.8. Training data for the multicategory treatment setting for Example 2. Top left subplot is the optimal treatment setup on the projection of data to the first 2 dimensions. Bottom left, top middle and bottom middle are the observed rewards for subjects with Treatments 1(dot), 2(triangle) and 3(square) respectively. Top right subplot is the rewards for all subjects with Treatments 1, 2, and 3.

Example 2 Assume three points c_1 c_2 and c_3 on a circle with radius 2 in $\mathbb{R}^2$ such that the distance between each pair is the same. The first 2 dimensions of covariates are randomly generated from

$$X_1, X_2 \sim \begin{cases} N(c_1, I_2) & \text{if } A_{opt} = 1 \\ N(c_2, I_2) & \text{if } A_{opt} = 2 \\ N(c_3, I_2) & \text{if } A_{opt} = 3 \end{cases}$$

Other covariates $X_3, \ldots, X_d$ are independently generated from $N(0, 1)$. Randomly assign Treatments 1, 2, 3 to the subjects with $P(A = i) = 1/3, i = 1, 2, 3$. The reward $R \sim N(\mu, 1)$ where $\mu = 5 + X\beta + 5I(A = A_{opt})$ and $\beta = 1_d$. The dimensions d considered are (2, 5, 10, 50, 100, 500).

Unlike Example 1, where the ratio of rewards for different optimal treatments is a smooth function, the ratio of rewards for different optimal treatments in Example 2 is close to a step function. A real world example is allergies. If a person is allergic to some medicine, it can lead to a very low outcome, otherwise the outcome is very large.

Figure 4.8 presents the training data for this multicategory treatment setting. For each observed treatment group, there is a very clear distinction between the different optimal treatment groups.

The performance is presented in Figure 4.9. Notice that in both balanced setting (0.33) and imbalanced setting (1/2 and 2/3), there is no difference between CDWD-OWL and CSVM-OWL. However, The performance of DWD-OWL and SVM-OWL is similar to Example 1, where DWD-OWL outperforms SVM-OWL especially in imbalanced setting.

4.4.3 Summary of the Simulation Studies

In our simulation study, we consider the following four methods: SVM-OWL, DWD-OWL, CSVM-OWL and CDWD-OWL. The two latter methods use the centered outcome. In our simulation, we only consider the situation where the outcome

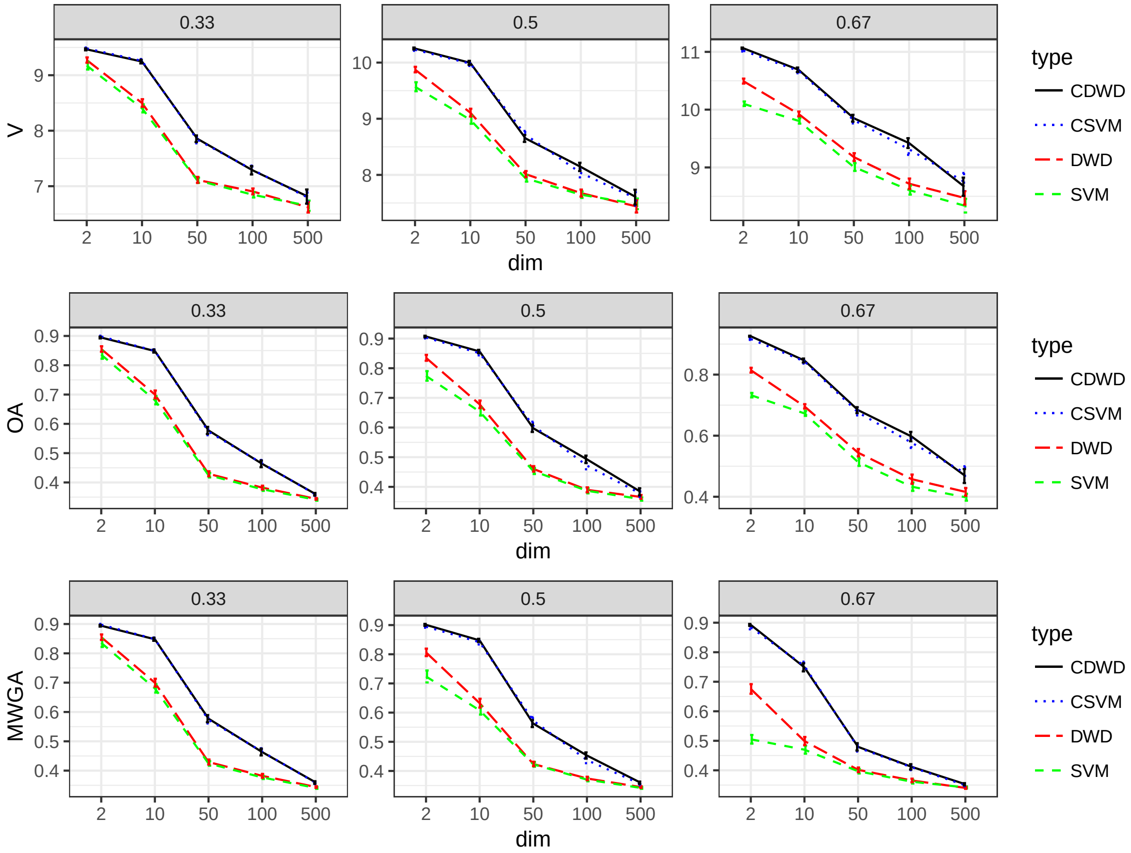

Figure 4.9. Comparison of performance using CDWD-OWL, CSVM-OWL, DWD-OWL and SVM-OWL for 3 treatments setting. From left to right, the subplots represent the performance for balanced, moderately imbalanced, and extremely imbalanced optimal treatment setting, respectively. In each panel of the figure, the x-axis represents the number of covariates and y represents the performance measures V, OA and MWGA from top to bottom.

rewards are positive. By centering them, the outcome rewards have both positive values and negative values.

In general, the DWD-based OWL methods show superiority over SVM-based OWL methods. Especially in the case where the outcomes are all positive. For outcomes with negative values, when the outcome ratio for different treatment is close to a step function, CSVM-OWL and CDWD-OWL shows similar performance. When the outcome rewards ratio of different treatments is a smooth function, CDWD-OWL outperforms CSVM-OWL. This is exemplified in three simulation studies provided in this chapter. More simulations not shown yield the same results.

The difference between DWD-based OWL and SVM-based OWL is more significant in the imbalanced setting, and their performance becomes very similar in the balanced setting. Overall, using centered outcomes will perform better in terms of estimated value and accuracy. In summary, DWD-OWL is superior to SVM-OWL.

4.5 Proofs of Theoretical Properties

In this section, we prove the Fisher consistency for DWD-OWL for both the binary and the multicategory treatment settings. We then provide the excess risk bound for DWD-OWL. The properties under the imbalanced setting is also given.

4.5.1 Proof of Fisher consistency

Define

$$\begin{aligned}
R_+^+(\boldsymbol{x}) &= \int_{R>0} (R|X=\boldsymbol{x}, A=+1)dP,\\
R_+^-(\boldsymbol{x}) &= \int_{R<0} (R|X=\boldsymbol{x}, A=+1)dP,\\
R_-^+(\boldsymbol{x}) &= \int_{R>0} (R|X=\boldsymbol{x}, A=-1)dP,\\
R_-^-(\boldsymbol{x}) &= \int_{R<0} (R|X=\boldsymbol{x}, A=-1)dP.
\end{aligned}$$

Note that $R_+^+ + R_+^- = E(R|X=\boldsymbol{x}, A=+1)$ and $R_-^+ + R_-^- = E(R|X=\boldsymbol{x}, A=-1)$.

Proof of Theorem 4.3.1

Since Theorem 4.3.1 is a special case of Theorem 4.3.2, we will prove Theorem 4.3.2 instead.

Proof of Theorem 4.3.2

Define $P_a(\boldsymbol{x}) = P(A = a|X = \boldsymbol{x})$, we have

$$\begin{aligned}
&E\Big(\frac{|R|}{P(A|X)}\ell_r\big(Af(X)\big)|X = \boldsymbol{x}\Big)\\
=&\sum_{a=1}^{k} E\Big(\frac{|R|}{P(A|X)}\ell_r\big(Af(X)\big)|X = \boldsymbol{x}, A = a\Big)P_a(\boldsymbol{x})\\
=&\sum_{a=1}^{k} E\Big(|R|\ell_r\big(Af(X)\big)|X = \boldsymbol{x}, A = a\Big)
\end{aligned} \tag{4.11}$$

In the binary treatment setting, there are only two treatments. Then the above formula (4.11) becomes

$$\begin{aligned}
&E\Big(\frac{|R|}{P(A|X)}\ell_r\big(Af(X)\big)|X = \boldsymbol{x}\Big)\\
=&\ell_r\big(f(x)\big)E\Big(|R||X = \boldsymbol{x}, A = 1\Big) + \ell_r\big(-f(x)\big)E\Big(|R||X = \boldsymbol{x}, A = -1\Big)\\
=&\big(\ell(f)R_+^+ - \ell(-f)R_+^-\big) + \big(\ell(-f)R_-^+ - \ell(f)R_-^-\big)\\
=&(R_+^+ - R_-^-)\ell(f) + (-R_+^- + R_-^+)\ell(-f)
\end{aligned} \tag{4.12}$$

If +1 is the optimal treatment, then $R_+^+ + R_+^- > R_-^+ + R_-^-$. Therefore, $R_+^+ - R_-^- > R_-^+ - R_+^-$.

Based on the definition of R_+^+, R_+^-, R_-^+, and R_-^-. Both $R_+^+ - R_-^-$ and $R_-^+ - R_+^-$ are positive values. Therefore, the minimizer f^* of (4.12) is the one satisfying $\ell(f^*) < \ell(-f^*)$. Since the DWD loss ℓ is a monotone decreasing function. $\ell(f^*) < \ell(-f^*)$ is equivalent to $f^* > 0$. Thus the proof is completed.

Proof of Theorem 4.3.3

Define

$$\begin{aligned}
R_j^+ &= \int_{R>0} (R|X = \boldsymbol{x}, A = j)dP,\\
R_j^- &= \int_{R<0} (R|X = \boldsymbol{x}, A = j)dP,
\end{aligned}$$

for any $j = 1, 2, \ldots, K$. Define

$$S(\boldsymbol{f}) = E\Big(\frac{|R|}{P(A|X)}\ell_r\big(\langle \boldsymbol{f}(X), W_A\rangle\big)|X = \boldsymbol{x}\Big)$$

Similar to (4.11), one can easily deduct that

$$\begin{aligned} S(\boldsymbol{f}) =& \sum_{j=1}^{K} \ell_r\{\langle \boldsymbol{f}, W_j\rangle\} E(|R||X = \boldsymbol{x}, A = j) \\ =& \sum_{j=1}^{K} [\ell\{\langle \boldsymbol{f}, W_j\rangle\} R_j^+ - \ell\{-\langle \boldsymbol{f}, W_j\rangle\} R_j^-] \end{aligned}$$

Without loss of generality, we assume that Treatment 1 is the best treatment. Since

$$\sum_{j=1}^{K} \langle \boldsymbol{f}, W_j\rangle = 0$$

and

$$\langle \boldsymbol{f}, W_1\rangle = \max_j \langle \boldsymbol{f}, W_j\rangle,$$

the minimizer f^* satisfies $\langle \boldsymbol{f}^*, W_1\rangle > 0$. To prove Theorem 4.3.3, we only need to show that when $R_1^+ + R_1^- > R_2^+ + R_2^-$, $\langle W_1, \boldsymbol{f}^*\rangle > \langle W_2, \boldsymbol{f}^*\rangle$. This can be proved by contradiction.

If $\langle W_1, \boldsymbol{f}^*\rangle \leq \langle W_2, \boldsymbol{f}^*\rangle$, choose $\boldsymbol{f}^{**}$ such that $\langle W_j, \boldsymbol{f}^{**}\rangle = \langle W_j, \boldsymbol{f}^*\rangle$ for $j \geq 3$ and $\langle W_1, \boldsymbol{f}^{**}\rangle = \langle W_1, \boldsymbol{f}^*\rangle + \epsilon$, $\langle W_2, \boldsymbol{f}^{**}\rangle = \langle W_2, \boldsymbol{f}^*\rangle - \epsilon$, where $\epsilon > 0$ and is a small positive value. Such a $\boldsymbol{f}^{**}$ exists based on Lemma 3.6.1 and the fact that inner product is continuous. To achieve the required $\boldsymbol{f}^{**}$, we only need to move $\boldsymbol{f}^*$ along the direction of $T_{1,2}$.

One can verify that

$$
\begin{aligned}
&S(\boldsymbol{f}^*) - S(\boldsymbol{f}^{**}) \\
=&R_1^+[\ell(\langle \boldsymbol{f}^*, W_1\rangle) - \ell(\langle \boldsymbol{f}^{**}, W_1\rangle)] - R_1^-[\ell(-\langle \boldsymbol{f}^*, W_1\rangle) - \ell(-\langle \boldsymbol{f}^{**}, W_1\rangle)] \\
&+ R_2^+[\ell(\langle \boldsymbol{f}^*, W_2\rangle) - \ell(\langle \boldsymbol{f}^{**}, W_2\rangle)] - R_2^-[\ell(-\langle \boldsymbol{f}^*, W_2\rangle) - \ell(-\langle \boldsymbol{f}^{**}, W_2\rangle)] + o(\epsilon) \\
=&R_1^+(-\epsilon)\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle) + R_1^-(-\epsilon)\ell^{'}(-\langle \boldsymbol{f}^*, W_1\rangle) \\
&+ R_2^+(\epsilon)\ell^{'}(\langle \boldsymbol{f}^*, W_2\rangle) + R_2^-(\epsilon)\ell^{'}(-\langle \boldsymbol{f}^*, W_2\rangle) + o(\epsilon) \\
=&\epsilon\Big(R_1^+|\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle)| + R_1^-|\ell^{'}(-\langle \boldsymbol{f}^*, W_1\rangle)| \\
&- R_2^+|\ell^{'}(\langle \boldsymbol{f}^*, W_2\rangle)| - R_2^-|\ell^{'}(-\langle \boldsymbol{f}^*, W_2\rangle)|\Big) + o(\epsilon)
\end{aligned}
$$

As $\langle \boldsymbol{f}^*, W_1\rangle > 0$, we have $0 < \langle W_1, \boldsymbol{f}^*\rangle \leq \langle W_2, \boldsymbol{f}^*\rangle$. For DWD loss

$$
\ell(u) = \begin{cases} 1-u & \text{if } u < 1/2 \\ 1/(4u) & \text{Otherwise} \end{cases}
$$

One can easily verify that the $\ell^{'}(u) < 0$ and $|\ell^{'}(u)|$ is a continuous non-increasing function. Thus for any $u_1 < u_2 \in \mathbb{R}$, $|\ell^{'}(u_1)| \geq |\ell^{'}(u_2)|$.

Based on our assumption $0 < \langle W_1, \boldsymbol{f}^*\rangle \leq \langle W_2, \boldsymbol{f}^*\rangle$ and the property of DWD loss function, it is easy to verify

$$
|\ell^{'}(-\langle \boldsymbol{f}^*, W_2\rangle)| \geq |\ell^{'}(-\langle \boldsymbol{f}^*, W_1\rangle)| > |\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle)| \geq |\ell^{'}(\langle \boldsymbol{f}^*, W_2\rangle)| > 0.
$$

Therefore,

$$
\begin{aligned}
&S(\boldsymbol{f}^*) - S(\boldsymbol{f}^{**}) \\
>&\epsilon\Big(R_1^+|\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle)| + R_1^-|\ell^{'}(-\langle \boldsymbol{f}^*, W_1\rangle)| \\
&- R_2^+|\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle)| - R_2^-|\ell^{'}(-\langle \boldsymbol{f}^*, W_1\rangle)|\Big) + o(\epsilon) \\
=&(R_1^+ - R_2^+)|\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle)| + (R_1^- - R_2^-)|\ell^{'}(-\langle \boldsymbol{f}^*, W_1\rangle)| + o(\epsilon)
\end{aligned}
$$

If $R_1^- - R_2^- > 0$, then

$$
\begin{aligned}
S(\boldsymbol{f}^*) - S(\boldsymbol{f}^{**}) \geq&(R_1^+ - R_2^+)|\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle)| + (R_1^- - R_2^-)|\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle)| \\
\geq&(R_1^+ - R_2^+ + R_1^- - R_2^-)|\ell^{'}(\langle \boldsymbol{f}^*, W_1\rangle)| \\
>&0
\end{aligned}
$$

This contradicts the assumption that $\boldsymbol{f}^*$ is the minimizer of $S(\boldsymbol{f})$, thus the proof is completed.

4.5.2 Proof of Excess Risk

Proof of Theorem 4.3.4

In this proof, we use a more generalized DWD loss proposed by Marron et al. [2007].

$$\ell(u) = \begin{cases} 2\sqrt{C} - Cu & \text{if } u < 1/\sqrt{C} \\ 1/(u) & \text{Otherwise} \end{cases}$$

When $C = 4$, the DWD loss is equivalent to the loss in Equation (4.5)

$$\ell(u) = \begin{cases} 1 - u & \text{if } u < 1/2 \\ 1/(4u) & \text{Otherwise} \end{cases}$$

It is easy to verify that $2\eta(\boldsymbol{x}) - 1$ is the decision boundary for the optimal ITR. To prove the theorem, we first consider the case with discrete rewards. Argument for continuous rewards setting follows similarly. Define $\eta_r(x) = p(A = 1|R = r, X = \boldsymbol{x})$, $q_r(\boldsymbol{x}) = rp(R = r|X = \boldsymbol{x})$ and $\pi(X) = P(A|X)$. We can write

$$\begin{aligned} \mathcal{R}(f) =& E[\sum_r rp(R = r|X)E(\frac{I(A \neq sign(f))}{A\pi(X) + (1-A)/2}|R = r, X)] \\ =& E[\sum_r q_r(X)(\frac{\eta_r(X)}{\pi(X)}I(sign(f) \neq 1) + \frac{1-\eta_r(X)}{1-\pi(X)}I(sign(f) \neq -1))] \\ =& E[c_0(X)(\eta(X)I(sign(f) \neq 1) + (1-\eta(X))I(sign(f) \neq -1))] \end{aligned}$$

with

$$c_0(\boldsymbol{x}) = \sum_r q_r(\boldsymbol{x})[\eta_r(\boldsymbol{x})/\pi(\boldsymbol{x}) + (1 - \eta_r(\boldsymbol{x})/(1 - \pi(\boldsymbol{x})))],$$

and

$$\eta(\boldsymbol{x}) = \sum_r q_r(\boldsymbol{x})\eta_r(\boldsymbol{x})/(\pi(\boldsymbol{x})c_0(\boldsymbol{x})),$$

which is the same as defined in Equation (4.10). Similarly,

$$\mathcal{R}_\ell(f) = E[c_0(X)(\eta(X)\ell(f) + (1 - \eta(X))\ell(-f))].$$

Define $C(\eta, f) = \eta\ell(f) + (1-\eta)\ell(-f)$ and $\Delta C(\eta, f) = C(\eta, f) - \inf_{f\in\mathbb{R}} C(\eta, f)$. The optimal $\ell-$risk satisfies

$$\mathcal{R}_\ell^* = \inf_{f\in\mathbb{R}} E[c_0(X)C(\eta(X), f)] = E[c_0(X) \inf_{f\in\mathbb{R}} C(\eta(X), f)]$$

and

$$\mathcal{R}_\ell - \mathcal{R}_\ell^* = E[c_0(X)(C(\eta(X), f) - \inf_{f\in\mathbb{R}} C(\eta(X), f))]$$

Define $f^* = \arg\inf_{f^*\in\mathbb{R}} C(\eta, f)$, then $\mathcal{R}_\ell^* = E[c_0(X)(C(\eta(X), f^*)]$. For DWD loss

$$\ell_D(u) = \begin{cases} 1/u & u > 1/\sqrt{C} \\ 2\sqrt{C} - Cu & \text{Otherwise} \end{cases},$$

it is easy to verify that

$$f^* = \begin{cases} \frac{1}{\sqrt{C}}\sqrt{\frac{\eta}{1-\eta}} & \eta > 1/2 \\ -\frac{1}{\sqrt{C}}\sqrt{\frac{1-\eta}{\eta}} & \text{Otherwise} \end{cases}.$$

And the corresponding

$$C(\eta, f^*) = \begin{cases} 2\sqrt{C}[1-\eta+\sqrt{\eta(1-\eta)}] & \eta > 1/2 \\ 2\sqrt{C}[\eta+\sqrt{\eta(1-\eta)}] & \text{Otherwise} \end{cases}.$$

Thus, we can directly compute

$$\begin{aligned} \Delta C(\eta, f) &= C(\eta, f) - C(\eta, f^*) \\ &= \eta(\ell(f) - \ell(f^*)) + (1-\eta)(\ell(-f) - \ell(-f^*)) \end{aligned}$$

and

$$\Delta C(\eta, 0) = \begin{cases} 2\sqrt{C}\sqrt{\eta(1-\eta)}(\sqrt{\frac{\eta}{1-\eta}} - 1) & \eta > 1/2 \\ 2\sqrt{C}\sqrt{\eta(1-\eta)}(\sqrt{\frac{1-\eta}{\eta}} - 1) & \text{Otherwise} \end{cases}.$$

It is straightforward to verify that $\Delta C(\eta,0) > 2\sqrt{C}|\eta-1/2|$. As $2\eta(x)-1$ is the decision boundary for optimal ITR, we have n

$$\begin{aligned}\mathcal{R}(f)-\mathcal{R}^* =&\mathcal{R}(f)-\mathcal{R}(\eta-1/2)\\ =&E[I(sign(f)\neq sign(\eta(X)-1/2))|c_0(X)(2\eta(X)-1)|]\\ =&E[I(f(\eta(X)-1/2)<0)c_0(X)|(2\eta(X)-1)|]\\ \leq&E[I(f(\eta(X)-1/2)<0)c_0(X)\Delta C(\eta,0)]\\ \leq&E[I(f(\eta(X)-1/2)<0)c_0(X)(C(\eta(X),0)-\inf_{a\in\mathbb{R}}C(\eta(X),a))]\\ \leq&E[I(f(\eta(X)-1/2)<0)c_0(X)(C(\eta(X),f)-\inf_{a\in\mathbb{R}}C(\eta(X),a))]\\ \leq&E[c_0(X)(C(\eta(X),f)-\inf_{a\in\mathbb{R}}C(\eta(X),a))]\\ =&\mathcal{R}_\ell-\mathcal{R}_\ell^*\end{aligned} \tag{4.13}$$

To show that the equality (4.13) holds, it is sufficient to prove that $f(2\eta-1)<0$ implies $C(\eta,0)<C(\eta,f)$. This can be verified in the following two scenarios:

- $\eta\geq 0.5$: For DWD loss, we have $f^*\geq 1/\sqrt{C}>0$. $(2\eta-1)f<0$ implies $f<0$. Thus $0\in[f,f^*]$ due to the convexity of $C(\eta,f)$ with respect to f. Furthermore, we obtained that $C(\eta,0)<max(C(\eta,f),C(\eta,f^*))=C(\eta,f)$.
- $\eta<0.5$: For DWD loss, we have $f^*<-1/\sqrt{C}<0$. $(2\eta-1)f<0$ implies $f>0$. Thus $0\in[f^*,f]$ due to the convexity of $C(\eta,f)$ with respect to f. Furthermore, we obtained that $C(\eta,0)<max(C(\eta,f),C(\eta,f^*))=C(\eta,f)$.

Thus in either case the result is proven. ■

4.5.3 Proof of Imbalance optimal treatment properties

Proof of Theorem 4.3.5

Defining

$$\mathcal{X}_+=\{\boldsymbol{x}\in\mathcal{X},2\eta(\boldsymbol{x})-1>0\},\mathcal{X}_-=\{\boldsymbol{x}\in\mathcal{X},2\eta(\boldsymbol{x})-1\leq 0\},$$

and assuming two classes are linearly separable. To prove the theorem, we can simply assume that the classification boundary will be pushed towards the minority class since the sample size for majority class is infinity. In this case, the functional margin $u_i = a_i f(\boldsymbol{x}_i)$ for the ith vector from the minority class is small and its DWD loss is $1 - u_i = 1 - f(\boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{X}_{--}$. Similarly, the function margin for the majority negative class is large and the corresponding loss is $1/(4a_i f(\boldsymbol{x}_i)) = -1/(4f(\boldsymbol{x}_i)) = -1/4(\boldsymbol{x}_i^T \boldsymbol{w} + \beta)$. Therefore, the objective function for wDWD is

$$\ell_D = \frac{1}{n_+ + n_-}\{\frac{1}{n_+}\sum_{i=1}^{n_+}[1 - (\boldsymbol{x}_i^T \boldsymbol{w} + \beta)] - \frac{1}{4n_-}\sum_{i=1}^{n_-}\frac{1}{\boldsymbol{x}_i^T \boldsymbol{w} + \beta}\} + \lambda/2||\boldsymbol{w}||^2.$$

Taking the derivative over β, we can get

$$\frac{\partial l_D}{\partial \beta} = \frac{1}{n_+ + n_-}\{-1 + \frac{1}{4n_-}\sum_{i=1}^{n_-}[(\boldsymbol{x}_i^T \boldsymbol{w} + \beta)^{-2}]\}$$

If $\bar{\boldsymbol{x}}_+^T \boldsymbol{w} + \beta \geq 0$, then $\beta > -\bar{\boldsymbol{x}}_+^T \boldsymbol{w} > -1/2 - \boldsymbol{x}_+^T \boldsymbol{w}$. Otherwise, assuming $\bar{\boldsymbol{x}}_+^T \boldsymbol{w} + \beta < 0$, for any $\boldsymbol{x}$ in negative/majority class, one can verify that $\boldsymbol{x}^T \boldsymbol{w} + \beta < 0$, and $\bar{\boldsymbol{x}}_+$ is on the same side of the classification boundary as the negative/majority class. Thus for any $\boldsymbol{x} \in \mathcal{X}_-$, $(\boldsymbol{x}_i^T \boldsymbol{w} + \beta)^2 > (\bar{\boldsymbol{x}}_+^T \boldsymbol{w} + \beta)^2$. Therefore,

$$\begin{aligned}\frac{\partial l_D}{\partial \beta} =& \frac{1}{n_+ + n_-}\{-1 + \frac{1}{4n_-}\sum_{i=1}^{n_-}[(\boldsymbol{x}_i^T \boldsymbol{w} + \beta)^{-2}]\} \\ <& \frac{1}{n_+ + n_-}\{-1 + 1/4(\bar{\boldsymbol{x}}_+^T \boldsymbol{w} + \beta)^{-2}\}\end{aligned}$$

Supposing that $\beta < -\sqrt{\frac{1}{4}} - \bar{\boldsymbol{x}}_+^T \boldsymbol{w} < 0$, then $-4 + (\bar{\boldsymbol{x}}_+^T \boldsymbol{w} + \beta)^{-2} < 0$ and $\frac{\partial l_D}{\partial \beta} < 0$. Since ℓ_D is a strictly convex function, the minimizer $\hat{\beta} > -\frac{1}{2} - \bar{\boldsymbol{x}}_+^T \boldsymbol{w}$. ■

Proof of theorem 4.3.6

Assuming two groups are linearly separable. We can prove this theorem by contradiction. Assuming that the intercept

$$\hat{\beta} < -max(\boldsymbol{x}_i^T \boldsymbol{w}) - \frac{1}{2}[1 + \sqrt{\frac{\sum_{\mathcal{X}_{--}} r_i}{(\sum_{\mathcal{X}_{-+} \cup \mathcal{X}_{++}} r_i - \sum_{\mathcal{X}_{+-}} r_i)}}].$$

Then $\hat{\beta} + max(\boldsymbol{x}_i^T \boldsymbol{w}) < 0$. It is easy to verify that the decision boundary is pushed towards the positive/minority class and $f(\boldsymbol{x}_i) < 0$ for $i = 1, \ldots, n$. Therefore, the DWD loss function becomes

$$l_D(f) = \begin{cases} \frac{1}{4u} = -\frac{1}{4f} & (x_i, a_i) \in \mathcal{X}_{--} \cup \mathcal{X}_{+-} \\ 1 - u = 1 - f & (x_i, a_i) \in \mathcal{X}_{-+} \cup \mathcal{X}_{++} \end{cases}$$

The objective loss for DWD-OWL becomes

$$\begin{aligned} & -\frac{1}{4n}\sum_{\mathcal{X}_{--}} r_i[(\boldsymbol{x}_i^T \boldsymbol{w} + \beta)^{-1}] + \frac{1}{n}\sum_{\mathcal{X}_{-+}} r_i[1 - (\boldsymbol{x}_i^T \boldsymbol{w} + \beta)] \\ & -\frac{1}{4n}\sum_{\mathcal{X}_{+-}} r_i[(\boldsymbol{x}_i^T \boldsymbol{w} + \beta)^{-1}] + \frac{1}{n}\sum_{\mathcal{X}_{++}} r_i[1 - (\boldsymbol{x}_i^T \boldsymbol{w} + \beta)] + \frac{\lambda}{2}||\boldsymbol{w}||^2 \end{aligned}$$

Taking the derivative over β,

$$\begin{aligned} \frac{\partial l_D}{\partial \beta} &= -\frac{1}{n}\sum_{\mathcal{X}_{-+}\cup\mathcal{X}_{++}} r_i + \frac{1}{4n}\sum_{\mathcal{X}_{--}} r_i[(\boldsymbol{x}_i^T \boldsymbol{w} + \beta)^{-2}] + \frac{1}{4n}\sum_{\mathcal{X}_{+-}} r_i[(\boldsymbol{x}_i^T \boldsymbol{w} + \beta)^{-2}] \\ &< -\frac{1}{n}\sum_{\mathcal{X}_{-+}\cup\mathcal{X}_{++}} r_i + \frac{1}{4n}\sum_{\mathcal{X}_{--}} r_i[(\bar{\boldsymbol{x}}_+^T \boldsymbol{w} + \beta)^{-2}] + \frac{1}{n}\sum_{\mathcal{X}_{+-}} r_i. \end{aligned}$$

For the inequality part, we use the fact that as the decision boundary are moving towards the positive group, for any $(\boldsymbol{x}_i, a_i) \in \mathcal{X}_{-+}$, one can verify that $|f(\boldsymbol{x}_i)| > |f(\bar{\boldsymbol{x}}_+)|$.

Notice that when

$$\beta < -\sqrt{\frac{\sum_{\mathcal{X}_{--}} r_i}{4(\sum_{\mathcal{X}_{-+}\cup\mathcal{X}_{++}} r_i - \sum_{\mathcal{X}_{+-}} r_i)}} - \bar{\boldsymbol{x}}_+^T \boldsymbol{w},$$

one can verify that $\frac{\partial l_D}{\partial \beta} < 0$. Due to the convexity of l_D and the continuity of $\frac{\partial l_D}{\partial \beta}$, the minimizer

$$\hat{\beta} > -\sqrt{\frac{\sum_{\mathcal{X}_{--}} r_i}{4(\sum_{\mathcal{X}_{-+}\cup\mathcal{X}_{++}} r_i - \sum_{\mathcal{X}_{+-}} r_i)}} - \bar{\boldsymbol{x}}_+^T \boldsymbol{w}$$

When the size of the majority/negative class goes to infinity, the right part goes to $-\sqrt{\frac{\gamma}{4}} - \bar{\boldsymbol{x}}_+^T \boldsymbol{w}$. It contradicts with the assumption thus complete the proof. ■

5. DISCUSSION AND FUTURE WORK

In this dissertation we focused on two specific classification problems. The first is to propose a multicategory classification method that is not sensitive to unbalanced HDLSS data. We show that our novel method has smaller misclassification error relative to both MDWD and MSVM in this setting and comparable results in other settings. In addition, when in the unbalanced HDLSS setting, our method provides a classifier that is closer to the Bayes discriminant direction than that of MSVM.

In the single-stage precision medicine setting, we compared DWD-OWL and SVM-OWL extending both methods to handle negative rewards and multiple treatments. We showed through simulation that our linear DWD-OWL has better performance compared to SVM-OWL method for both binary and multicategory treatments.

The following sections summarize additional work we plan in each of these areas.

5.1 Future work for MDWSVM

Our focus in this dissertation was entirely on linear classifiers. To deal with non-linear classification, a kernel trick [Schölkopf, 2001] can be implemented. We've extended our MDWSVM method to use this kernel trick and are in the process of assessing/comparing the performance of various methods.

Our MDWSVM is implemented in Matlab and uses interior point estimation to find the optimal decision rule. Another way to solve our optimization problem is to use gradient descent or stochastic gradient descent [Bottou, 2010, Mandic, 2004], which can handle large scales very easily. The implementation of gradient descent to MDWSVM in R is a work in progress. A goal is to eventually have an R package that will include both our MDWSVM and the kernel version.

Our MDWSVM approach uses the squared norm as the regularization component so it does not have a variable selection property. To better deal with data of high dimension, variable selection penalties can be added to the method, e.g., LASSO [Tibshirani, 1996] or Elastic net [Zou and Hastie, 2005]. Work on this type of generalization will follow.

5.2 Future work for optimal ITRs

For our DWD-OWL method, we provided the Fisher consistency for both the binary and the multicategory settings. The excess risk and the insensitivity is only provided for the binary treatment setting. We conjecture that similar properties hold for the multicategory treatment setting. The demonstration of this conjecture is going to be our immediate future work.

In random clinical trials, the propensity score $P(A|X)$ is pre-determined and doesn't depend on X. However, in observational studies, one needs to estimate $P(A|X)$. If the sample size of one observed treatment is small or if a predictor is strongly associated with one of the treatments, the estimated propensity score may be biased. When the observed treatment group sizes are imbalanced, the estimation of the propensity score may have a huge impact on the optimal ITR, especially if there are some propensity scores close to 0. Therefore, a method needs to be explored to push the estimation of the propensity scores away from 0. Other than logistic regression estimation proposed by Zhao et al. [2012], an alternative of the propensity score estimation might be considered, like Firth bias-adjusted estimates Firth [1993], which use Jeffreys invariant prior to reduce the bias.

REFERENCES

J. Ahn and J. Marron. The maximal data piling direction for discrimination. *Biometrika*, 97(1):254–259, 2010.

M. Al-Hajj, M. S. Wicha, A. Benito-Hernandez, S. J. Morrison, and M. F. Clarke. Prospective identification of tumorigenic breast cancer cells. *Proceedings of the National Academy of Sciences*, 100(7):3983–3988, 2003.

H. Bang and J. M. Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005.

P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

C. M. Bishop. *Pattern recognition and machine learning.* springer, 2006.

D. Böhning. Multinomial logistic regression algorithm. *Annals of the institute of Statistical Mathematics*, 44(1):197–200, 1992.

L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

M. P. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. W. Sugnet, T. S. Furey, M. Ares, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences*, 97(1):262–267, 2000.

T. Cai, L. Tian, D. Lloyd-Jones, and L. Wei. Evaluating subject-level incremental values of new markers for risk classification rule. *Lifetime data analysis*, 19(4): 547–567, 2013.

X. Cai, F. Nie, H. Huang, and C. Ding. Multi-class l2, 1-norm support vector machine. In *2011 IEEE 11th International Conference on Data Mining*, pages 91–100. IEEE, 2011.

L. L. Campbell and K. Polyak. Breast tumor heterogeneity: cancer stem cells or clonal evolution? *Cell cycle*, 6(19):2332–2338, 2007.

O. Chapelle, P. Haffner, and V. N. Vapnik. Support vector machines for histogram-based image classification. *IEEE Transactions on Neural Networks*, 10(5):1055–1064, 1999.

C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

N. Cristianini, J. Shawe-Taylor, et al. *An introduction to support vector machines and other kernel-based learning methods.* Cambridge university press, 2000.

M. L. Crossley. let me explain: narrative emplotment and one patient's experience of oral cancer. *Social Science & Medicine*, 56(3):439–448, 2003.

Y. Cui, R. Zhu, and M. Kosorok. Tree based weighted learning for estimating individualized treatment rules with censored data. *Electronic journal of statistics*, 11(2):3927–3953, 2017.

P. A. Devijver and J. Kittler. *Pattern recognition: A statistical approach*. Prentice hall, 1982.

C. Di Natale, A. Macagnano, E. Martinelli, R. Paolesse, G. D'Arcangelo, C. Roscioni, A. Finazzi-Agro, and A. D'Amico. Lung cancer identification by the analysis of breath by means of an array of non-selective gas sensors. *Biosensors and Bioelectronics*, 18(10):1209–1218, 2003.

Y. Dodge. *Statistical data analysis based on the L1-norm and related methods*. Birkhäuser, 2012.

K.-B. Duan, J. C. Rajapakse, H. Wang, and F. Azuaje. Multiple svm-rfe for gene selection in cancer classification with expression data. *IEEE Transactions on Nanobioscience*, 4(3):228–234, 2005.

A. Fathi and G. Mori. Action recognition by learning mid-level motion features. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

M. L. Feldstein, E. D. Savlov, and R. Hilf. A statistical model for predicting response of breast cancer patients to cytotoxic chemotherapy. *Cancer research*, 38(8):2544–2548, 1978.

I. Fette, N. Sadeh, and A. Tomasic. Learning to detect phishing emails. In *Proceedings of the 16th international conference on World Wide Web*, pages 649–656. ACM, 2007.

D. Firth. Bias reduction of maximum likelihood estimates. *Biometrika*, 80(1):27–38, 1993.

R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

G. M. Foody and A. Mathur. The use of small training sets containing mixed pixels for accurate hard image classification: Training on mixed spectral responses for classification by a svm. *Remote Sensing of Environment*, 103(2):179–189, 2006.

E. Frank, M. Hall, and B. Pfahringer. Locally weighted naive bayes. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pages 249–256. Morgan Kaufmann Publishers Inc., 2002.

J. H. Friedman. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

Y. Geng, H. H. Zhang, and W. Lu. On optimal treatment regimes selection for mean survival time. *Statistics in medicine*, 34(7):1169–1184, 2015.

M. Grant, S. Boyd, and Y. Ye. Cvx: Matlab software for disciplined convex programming, 2008.

L. Gunter, J. Zhu, and S. Murphy. Variable selection for qualitative interactions in personalized medicine while controlling the family-wise error rate. *Journal of biopharmaceutical statistics*, 21(6):1063–1078, 2011.

T. Hastie, R. Tibshirani, and J. Friedman. The elements of statistical learning. 2001. *Springer*, 2001.

S. Haykin. *Neural networks*, volume 2. Prentice hall New York, 1994.

S. S. Haykin, S. S. Haykin, S. S. Haykin, K. Elektroingenieur, and S. S. Haykin. *Neural networks and learning machines*, volume 3. Pearson Upper Saddle River, 2009.

D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant. *Applied logistic regression*, volume 398. John Wiley & Sons, 2013.

C.-W. Hsu and C.-J. Lin. A comparison of methods for multiclass support vector machines. *IEEE transactions on Neural Networks*, 13(2):415–425, 2002.

M. Hu, Y. Chen, and J. T.-Y. Kwok. Building sparse multiple-kernel svm classifiers. *IEEE Transactions on Neural Networks*, 20(5):827–839, 2009.

H. Huang, Y. Liu, Y. Du, C. M. Perou, D. N. Hayes, M. J. Todd, and J. S. Marron. Multiclass distance-weighted discrimination. *Journal of Computational and Graphical Statistics*, 22(4):953–969, 2013.

Y.-M. Huang and S.-X. Du. Weighted support vector machine for classification with uneven training class sizes. In *2005 International Conference on Machine Learning and Cybernetics*, volume 7, pages 4365–4369. IEEE, 2005.

J. Jiang, D. Wu, Y. Chen, D. Yu, L. Wang, and K. Li. Fast artificial bee colony algorithm with complex network and naive bayes classifier for supply chain network management. *Soft Computing*, pages 1–17, 2019.

G. H. John and P. Langley. Estimating continuous distributions in bayesian classifiers. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 338–345. Morgan Kaufmann Publishers Inc., 1995.

D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, and M. Klein. *Logistic regression*. Springer, 2002.

R. Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.

T. Kowalkowski, R. Zbytniewski, J. Szpejna, and B. Buszewski. Application of chemometrics in river water classification. *Water research*, 40(4):744–752, 2006.

N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *2009 IEEE 12th International Conference on Computer Vision*, pages 365–372. IEEE, 2009.

E. Laber and Y. Zhao. Tree-based methods for individualized treatment regimes. *Biometrika*, 102(3):501–514, 2015.

K. Lange and T. Wu. An mm algorithm for multicategory vertex discriminant analysis. *Journal of Computational and Graphical Statistics*, 17(3):527–544, 2008.

M. H. Lee, J. Ahn, and Y. Jeon. Hdlss discrimination with adaptive data piling. *Journal of Computational and Graphical Statistics*, 22(2):433–451, 2013.

Y. Lee, Y. Lin, and G. Wahba. Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data. *Journal of the American Statistical Association*, 99(465):67–81, 2004.

M. Liang, T. Ye, and H. Fu. Estimating individualized optimal combination therapies through outcome weighted deep learning algorithms. *Statistics in medicine*, 37 (27):3869–3886, 2018.

A. Liaw, M. Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

Y. Lin. A note on margin-based loss functions in classification. *Statistics & probability letters*, 68(1):73–82, 2004.

Y. Liu and X. Shen. Multicategory ψ-learning. *Journal of the American Statistical Association*, 101(474):500–509, 2006.

Y. Liu and M. Yuan. Reinforced multicategory support vector machines. *Journal of Computational and Graphical Statistics*, 20(4):901–919, 2011.

Y. Liu, H. H. Zhang, and Y. Wu. Hard or soft classification? large-margin unified machines. *Journal of the American Statistical Association*, 106(493):166–177, 2011.

Y. Liu, Y. Wang, M. R. Kosorok, Y. Zhao, and D. Zeng. Robust hybrid learning for estimating personalized dynamic treatment regimens. *arXiv preprint arXiv:1611.02314*, 2016.

W. Lu, H. H. Zhang, and D. Zeng. Variable selection for optimal treatment decision. *Statistical methods in medical research*, 22(5):493–504, 2013.

Y.-F. Lu, D. B. Goldstein, M. Angrist, and G. Cavalleri. Personalized medicine and human genetic diversity. *Cold Spring Harbor perspectives in medicine*, page a008581, 2014.

D. P. Mandic. A generalized normalized gradient descent algorithm. *IEEE signal processing letters*, 11(2):115–118, 2004.

J. Marron, M. J. Todd, and J. Ahn. Distance-weighted discrimination. *Journal of the American Statistical Association*, 102(480):1267–1271, 2007.

S. Menard. *Applied logistic regression analysis*, volume 106. Sage, 2002.

S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.-R. Mullers. Fisher discriminant analysis with kernels. In *Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468)*, pages 41–48. Ieee, 1999.

E. E. Moodie, T. S. Richardson, and D. A. Stephens. Demystifying optimal dynamic treatment regimes. *Biometrics*, 63(2):447–455, 2007.

S. A. Murphy. Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2):331–355, 2003.

A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, and S. D. Brown. An introduction to decision tree modeling. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 18(6):275–285, 2004.

L. Orellana, A. Rotnitzky, and J. M. Robins. Dynamic regime marginal structural mean models for estimation of optimal dynamic treatment regimes, part i: main content. *The International Journal of Biostatistics*, 6(2), 2010.

E. Osuna, R. Freund, F. Girosi, et al. Training support vector machines: an application to face detection. In *CVPR*, volume 97, page 99, 1997.

A. B. Owen. Infinitely imbalanced logistic regression. *Journal of Machine Learning Research*, 8(Apr):761–773, 2007.

A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 1320–1326, 2010.

M. Pal. Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, 26(1):217–222, 2005.

M. Qian and S. A. Murphy. Performance guarantees for individualized treatment rules. *Annals of statistics*, 39(2):1180, 2011.

X. Qiao and Y. Liu. Adaptive weighted learning for unbalanced multicategory classification. *Biometrics*, 65(1):159–168, 2009.

X. Qiao and L. Zhang. Distance-weighted support vector machine. *Statistics and Its Interface*, 8(3):331–345, 2015a.

X. Qiao and L. Zhang. Flexible high-dimensional classification machines and their asymptotic properties. *Journal of Machine Learning Research*, 16:1547–1572, 2015b.

X. Qiao, H. H. Zhang, Y. Liu, M. J. Todd, and J. S. Marron. Weighted distance weighted discrimination and its asymptotic properties. *Journal of the American Statistical Association*, 105(489):401–414, 2010.

J. M. Robins. Optimal structural nested models for optimal sequential decisions. In *Proceedings of the second seattle Symposium in Biostatistics*, pages 189–326. Springer, 2004.

L. Rokach and O. Z. Maimon. *Data mining with decision trees: theory and applications*, volume 69. World scientific, 2008.

R. E. Schapire. The boosting approach to machine learning: An overview. In *Nonlinear estimation and classification*, pages 149–171. Springer, 2003.

B. Schölkopf. The kernel trick for distances. In *Advances in neural information processing systems*, pages 301–307, 2001.

B. Schölkopf and C. J. Burges. *Advances in kernel methods: support vector learning*. MIT press, 1999.

H. Shen and J. Z. Huang. Analysis of call centre arrival data using singular value decomposition. *Applied Stochastic Models in Business and Industry*, 21(3):251–263, 2005.

X. Shen, L. Wang, et al. Generalization error for multi-class margin classification. *Electronic Journal of Statistics*, 1:307–330, 2007.

A. Sies and I. Van Mechelen. Comparing four methods for estimating tree-based treatment regimes. *The international journal of biostatistics*, 13(1), 2017.

T. K. Solberg, Ø. P. Nygaard, K. Sjaavik, D. Hofoss, and T. Ingebrigtsen. The risk of getting worse after lumbar microdiscectomy. *European Spine Journal*, 14(1): 49–54, 2005.

J. Stoehlmacher, D. Park, W. Zhang, D. Yang, S. Groshen, S. Zahedy, and H. Lenz. A multivariate analysis of genomic polymorphisms: prediction of clinical outcome to 5-fu/oxaliplatin combination chemotherapy in refractory colorectal cancer. *British journal of cancer*, 91(2):344, 2004.

H. Sun, B. A. Craig, and L. Zhang. Angle-based multicategory distance-weighted svm. *The Journal of Machine Learning Research*, 18(1):2981–3001, 2017.

Y. Tao, L. Wang, D. Almirall, et al. Tree-based reinforcement learning for estimating optimal dynamic treatment regimes. *The Annals of Applied Statistics*, 12(3):1914–1938, 2018.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

V. Vapnik, I. Guyon, and T. Hastie. Support vector machines. *Mach. Learn*, 20(3): 273–297, 1995.

V. N. Vapnik and V. Vapnik. *Statistical learning theory*, volume 1. Wiley New York, 1998.

H. Wang and G. Huang. Application of support vector machine in cancer diagnosis. *Medical Oncology*, 28(1):613–618, 2011.

B. Zhang, A. A. Tsiatis, M. Davidian, M. Zhang, and E. Laber. Estimating optimal treatment regimes from a classification perspective. *Stat*, 1(1):103–114, 2012.

C. Zhang and Y. Liu. Multicategory large-margin unified machines. *The Journal of Machine Learning Research*, 14(1):1349–1386, 2013.

C. Zhang and Y. Liu. Multicategory angle-based large-margin classification. *Biometrika*, 101(3):625–640, 2014.

C. Zhang, J. Chen, H. Fu, X. He, Y. Zhao, and Y. Liu. Multicategory outcome weighted margin-based learning for estimating individualized treatment rules. 2018.

H. Zhang. The optimality of naive bayes. *AA*, 1(2):3, 2004.

P. Zhao and B. Yu. On model selection consistency of lasso. *Journal of Machine learning research*, 7(Nov):2541–2563, 2006.

Y. Zhao, M. R. Kosorok, and D. Zeng. Reinforcement learning design for cancer clinical trials. *Statistics in medicine*, 28(26):3294–3315, 2009.

Y. Zhao, D. Zeng, A. J. Rush, and M. R. Kosorok. Estimating individualized treatment rules using outcome weighted learning. *Journal of the American Statistical Association*, 107(499):1106–1118, 2012.

Y.-Q. Zhao, D. Zeng, E. B. Laber, R. Song, M. Yuan, and M. R. Kosorok. Doubly robust learning for estimating individualized treatment with censored data. *Biometrika*, 102(1):151–168, 2014.

X. Zhou, N. Mayer-Hamblett, U. Khan, and M. R. Kosorok. Residual weighted learning for estimating individualized treatment rules. *Journal of the American Statistical Association*, 112(517):169–187, 2017.

X. Zhou, Y. Wang, and D. Zeng. Sequential outcome-weighted multicategory learning for estimating optimal individualized treatment rules. 2018.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2): 301–320, 2005.

VITA

Hui (Sophie) Sun is born in Zhongxiang in Hubei province, China. She received her Bachelor's degree in Mathematics in Huazhong University of Science and Technology in June 2010. And later got her Master's degree in Statistics from Nankai University in June 2013. In August 2013, she entered the PhD program in Statistics at Purdue University.