Journal of Machine Learning Research 18 (2017) 1-21 Submitted 1/17; Revised 5/17; Published 7/17

Angle-based Multicategory Distance-weighted SVM

Hui Sun SUN400@PURDUE.EDU
Department of Statistics
Purdue University
West Lafayette, IN 47906, USA

Bruce A. Craig BACRAIG@PURDUE.EDU
Department of Statistics
Purdue University
West Lafayette, IN 47906, USA

Lingsong Zhang LINGSONG@PURDUE.EDU
Department of Statistics
Purdue University
West Lafayette, IN 47906, USA

Editor: John Shawe-Taylor

Abstract

Classification is an important supervised learning technique with numerous applications. We develop an angle-based multicategory distance-weighted support vector machine (MDWSVM) classification method that is motivated from the binary distance-weighted support vector machine (DWSVM) classification method. The new method has the merits of both support vector machine (SVM) and distance-weighted discrimination (DWD) but also alleviates both the *data piling* issue of SVM and the imbalanced data issue of DWD. Theoretical and numerical studies demonstrate the advantages of MDWSVM method over existing angle-based methods.

Keywords: Discriminant analysis, Imbalanced data, High dimension, Support vector machine, Distance-weighted discrimination

1. Introduction

Classification is important in both statistics and machine learning. The goal of classification is to build a classifier such that it can 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 (Schölkopf and Burges, 1999; Cristianini and Shawe-Taylor, 2000) has been shown to be a very popular and powerful method. It is well known that the binary SVM searches for a hyperplane in the feature space (that is a type of projection space from the original data) that maximizes the margin (a gap between the two groups). SVM has numerous applications, such as image classification (Chapelle et al., 1999; Foody and Mathur, 2006) and cancer diagnostics (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 the 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 is the phenomenon when projecting the data points to the discriminant direction that many of these projections are identical. This phenomenon indicates that the resulting separating hyperplane might be affected by noise artifacts in the data. Noise artifacts result in a discrimination direction far away from the Bayes 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. See more discussion in Qiao and Zhang (2015a,b) for the HDLSS overfitting issue of SVM and imbalanced data issue of DWD. 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. A natural way to deal with multiclass classification problems is to take a one-versus-one or one-versus-rest approach, see examples in Hastie et al. (1998) and Allwein et al. (2000). Though both approaches are intuitive, the one-versus-one approach can lead to a tie-in-vote problem and the one-versus-rest approach suffers from inconsistency when there is no dominant class (Lee et al., 2004).

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 $\hat{y} = \arg\max_i f_j(\boldsymbol{x})$, 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. See Zhu and Hastie (2001); Lee et al. (2004); Liu and Shen (2006); Liu (2007); Liu and Yuan (2011) for more discussion. Fisher consistency for several existing multicategory hinge loss functions are also provided in Liu (2007).

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. For example, Lange and Wu (2008) proposed a vertex based model that maps the data $\boldsymbol{x}$ in $\mathbb{R}^d$ to a $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^{K-1}$, and predicts the label based on the distance of $\boldsymbol{f}(\boldsymbol{x})$ to K predefined vertices. Zhang and Liu (2014) proposed a similar idea where they define the same K vertices in the $\mathbb{R}^{k-1}$ space, and use the angle between $\boldsymbol{f}$ to these vertex vectors to predict labels. More details are given in Section 2 of this paper.

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 our experiment in Section 2, under HDLSS and imbalanced data setting, we observed that MSVM suffers from *data piling* issue. 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. Both these issues were previously observed in the binary case (Marron et al., 2007; Qiao and Zhang, 2015a).

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 one-versus-one idea. MDWDH considers a data point with label i being misclassified if the difference of projections on ith member and jth member is negative ($i \neq j$). Even though it has nice theoretical properties and empirical performance as shown in their paper, the angle-based methods have better geometric interpretation. In addition, the pair-wise natural of the method will have more expensive computational cost, compared to angle-based approaches. We also observed that the performance is slightly better than angled-based MDWD method. If the angle-based MDWD approach also incorporates the pairwise idea, the two approaches will have similar empirical performance.

In this paper, 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 article is organized as follows. In Section 2, we briefly review the existing multicategory classifiers, and introduce our angle-based distance-weighted support vector machine (MDWSVM) model. In Section 3, we prove the Fisher consistency and show some imbalance properties of our new approach. In Section 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 5 involves a real application and Section 6 discusses some future work for this model. The proofs of all theorems and lemmas are given in the appendix.

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.

2.1 Classification and 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 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

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

where $\mathbb{F}$ 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 1, 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

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

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 1 to overcome identifiability issues, see more discussion in Vapnik and Vapnik (1998); Lee et al. (2004); Liu and Yuan (2011).

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.

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

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

The unit vector ζ is 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 optimization model for the angle-based classification

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

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.

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

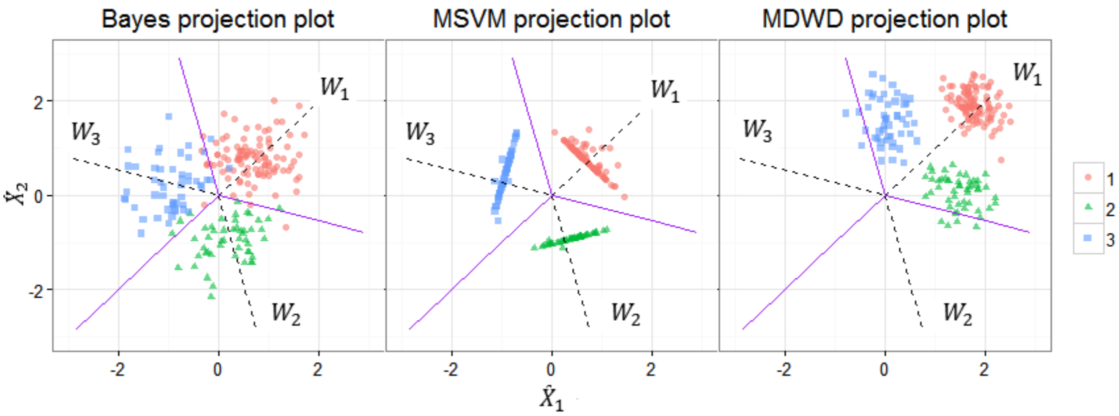


Figure 1: Plots of projections and W_j's in $\mathbb{R}^2$ space. Dashed lines are the W_j's, $j = 1, \ldots, 3$; Dots, triangles and squares represent the points from the three different classes. The left panel shows the projection plot for the Bayes classification, the middle one is for angle-based 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).

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.

One representation of the projection plots is given in Figure 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 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 form

$$\min_{f \in \mathbb{F}} \frac{1}{n} \sum_{i=1}^{n} \alpha \ell_D(y_i f_0(\boldsymbol{x}_i)) + (1-\alpha)\ell_S(y_i f(\boldsymbol{x}_i)) + \frac{\lambda}{2} J(f), \tag{3}$$

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 (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)

$$\min_{\boldsymbol{f}\in\mathbb{F}} \frac{1}{n}\sum_{i=1}^{n} \alpha\ell_D(\langle \boldsymbol{f}_0(\boldsymbol{x}_i), W_{y_i}\rangle) + (1-\alpha)\ell_S(\langle \boldsymbol{f}(\boldsymbol{x}_i), W_{y_i}\rangle) + \frac{\lambda}{2}J(\boldsymbol{f}). \tag{4}$$

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.

2.4 Implementation of MDWSVM

In Qiao and Zhang (2015a), the implementation of the binary DWSVM (3) was through second-order cone programming. Mathematically, the DWSVM model can be written as

$$\begin{aligned} \min_{\boldsymbol{\omega},\beta,\beta_0,\eta_i,\xi_i} \quad & \sum_{i=1}^{n} \{\alpha(\frac{1}{r_i} + \eta_i) + (1-\alpha)\xi_i\}, \\ \text{s.t.} \quad & r_i = y_i(x_i^T\boldsymbol{\omega} + \beta) + \eta_i, \quad r_i \geq 0 \quad \text{and} \quad \eta_i \geq 0, \\ & y_i(x_i^T\boldsymbol{\omega} + \beta) + \xi_i \geq 1, \quad \xi_i \geq 0, \\ & \|\boldsymbol{\omega}\|^2 \leq C. \end{aligned} \tag{5}$$

The first constraint $r_i = y_i(x_i^T\boldsymbol{\omega} + \beta) + \eta_i$ 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 (2), the Euclidean norm.

To extend (5) to multiclass, we replace the distances (functional margins) to the inner product as introduced earlier in (2). Thus our MDWSVM will have the following mathematical form:

$$\begin{aligned}\min_{\boldsymbol{f},\boldsymbol{f_0}} \quad & \sum_{i=1}^{n}\{\alpha(\frac{1}{r_i}+\eta_i)+(1-\alpha)\xi_i\},\\ \text{s.t.} \quad & r_i=\langle \boldsymbol{f}_0(x_i),W_{y_i}\rangle+\eta_i, \quad r_i\geq 0 \quad \text{and} \quad \eta_i\geq 0,\\ & \langle \boldsymbol{f}(x_i),W_{y_i}\rangle+\xi_i\geq 1, \quad \xi_i\geq 0,\\ & \sum_{j=1}^{k-1}B_j^TB_j\leq C.\end{aligned} \tag{6}$$

In this form $\boldsymbol{f_0}$ and $\boldsymbol{f}$ are the same as in (4). It is verified in Zhang and Liu (2014) that the first term in the objective function $\sum_{i=1}^{n}(\frac{1}{r_i}+\eta_i)$ 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 (6) 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 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 4 and 5 using simulations and real examples.

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 $\hat{y}=\arg\max_j\langle\boldsymbol{f^*(x)},W_Y\rangle$. Note that this is essentially the limit minimizer of (2) when sample size diverges to infinity. Fisher consistency assures that these two decision functions are the same ($\arg\max_j P_j=\arg\max_j\langle\boldsymbol{f^*(x)},W_Y\rangle$).

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

Fisher consistency in Theorem 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 4.2, 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 2 *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 *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 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.

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 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

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

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 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.

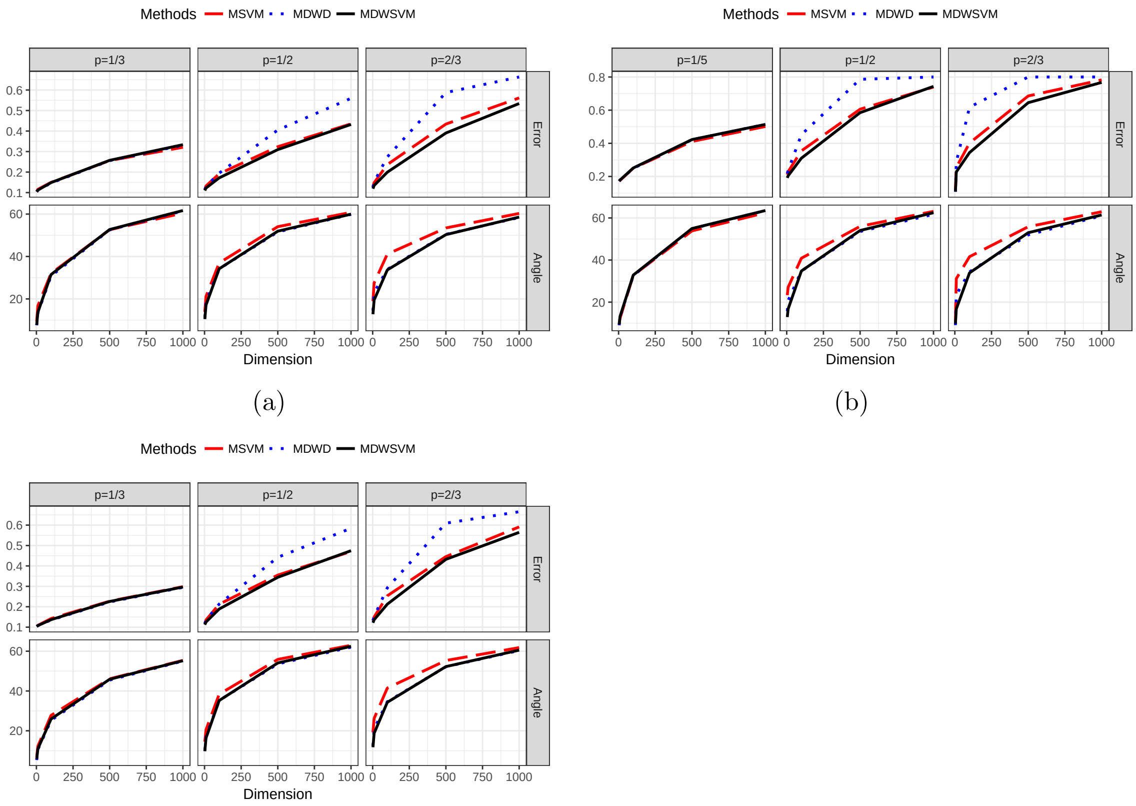
4.1 Performance Comparison

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 2. Take Figure 2(a), which corresponds to Example 1, as an example. We report



(a)

(b)

(c)

Figure 2: Performance comparison plot between the three methods for Examples 1, 2, and 3. The top row of each graph 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. We can see that our method (the solid black line) performs almost the same as the other two methods (MSVM, the red dashed line, MDWD, the dotted blue line) for balanced case, but outperforms the other two for the imbalanced cases.

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 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 2(b) and 2(c). 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.

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 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 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, 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.

4.3 Computation Time

In this subsection, we will compare the computational time for these methods (MDWSVM, MSVM and MDWD). To test the computational complexity, we only consider the simulation

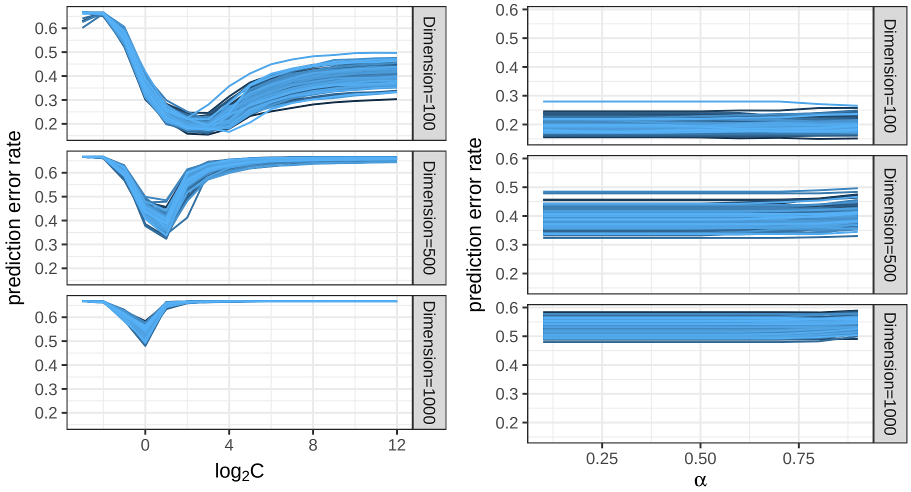


Figure 3: Average within-group error rate change under different parameters. Left panel is the prediction error rate change under different C value for fixed $\alpha = 0.5$; right panel is prediction error rate change for different α when C is fixed at its optimal value

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 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.

of Example 1. We let the dimension change from 10 to 1000. Table 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 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 6, 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.

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

Table 2: Six national holidays on weekdays that are removed

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 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).

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 4. From Figure 4, 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 provides these results for the three different methods.

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

Taking another look at the Table 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

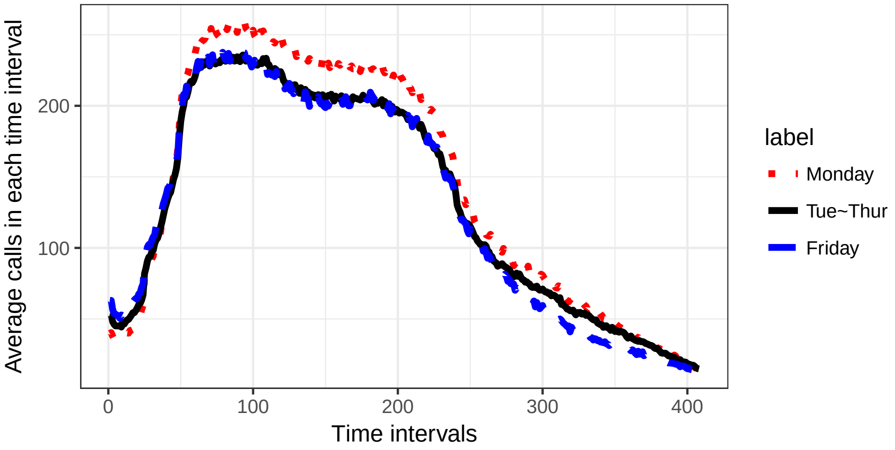


Figure 4: Average number of calls in each time interval for the three classes Monday, Tuesday to Thursday and Friday. In this plot, the red dotted line is the number of phone calls for Mondays, the black solid line is for Tuesday to Thursday, and the blue dashed line is for Friday.

	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

Table 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.

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.

6. Discussion

In this paper, we proposed a new angle-based method MDWSVM. We show in this paper that in HDLSS and the imbalanced case, our novel MDWSVM method has smaller misclassification error relative to both MDWD and MSVM. In addition, it is closer to the Bayes discriminant direction compared to that of MSVM. The MDWSVM can be viewed as an extension of the binary DWSVM to the multicategory case. Furthermore, as our model considers both SVM and DWD loss, it can be treated as a weighted hybridization of MSVM and MDWD.

One of the limitations of DWD-type methods is that it suffers from slow computation time compared to SVM-type methods (See Chapter 4.3 for examples). Recently an alternating direction method of multipliers (ADMM) algorithm has been implemented by Lam et al. (2016) for the DWD type methods, which can handle large size data more efficiently. A similar implementation for our MDWSVM method will be explored in the future.

Note that we only consider linear classifiers for simplicity. This method can be extended to a kernel approach (Liu and Yuan, 2011) by allowing $\boldsymbol{f}$ to be a nonlinear mapping. The implementation of such a generalization will be our immediate future work.

Our MDWSVM 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 model, *e.g.* LASSO (Tibshirani, 1996) or Elastic net (Zou and Hastie, 2005). Work on this type of generalization will follow.

Acknowledgments

We would like to thank the reviewers for the valuable comments and suggestions. Also thank Cheng Li (Purdue University) for the insightful discussion on proof of theorems.

Appendix A.

In this appendix we prove the following theorems and proposition from Section 3:

Lemma 1 *(Zhang and Liu, 2014): Suppose we have an arbitrary* $\boldsymbol{f} \in \mathbb{R}^{K-1}$*. For any* u*,* $v \in \{1, \ldots, K\}$ *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* $w \in \{1, \ldots, K\}$ *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 1 is given in Zhang and Liu (2014). From Lemma 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.

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

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

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

Proof. 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 $\forall B \in \mathbb{R}^{p \times (K-1)}$, we have

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

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

Theorem 3. *In the MDWSVM setting, when the size of the majority class goes to infinity,* $\langle \beta_0, W_{y_K}\rangle < \sqrt{2C}K \max|x_{ij}| + 1$.

Proof. Based on the proof of Proposition 2, for any $\forall B \in \mathbb{R}^{p\times(K-1)}$, we have

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

Thus for MDWSVM model, no matter what B we get, the intercept only comes from MSVM part. Therefore, from the conclusion of Proposition 2, Theorem 3 is proved. ■

References

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

Erin L Allwein, Robert E Schapire, and Yoram Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1(Dec): 113–141, 2000.

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

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

Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods.* Cambridge University Press, 2000.

Kai-Bo Duan, Jagath C Rajapakse, Haiying Wang, and Francisco Azuaje. Multiple svm-rfe for gene selection in cancer classification with expression data. *IEEE Transactions on Nanobioscience*, 4(3):228–234, 2005.

Giles M Foody and Ajay 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.

Michael Grant, Stephen Boyd, and Yinyu Ye. Cvx: Matlab software for disciplined convex programming, 2008.

Trevor Hastie, Robert Tibshirani, et al. Classification by pairwise coupling. *The Annals of Statistics*, 26(2):451–471, 1998.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. The elements of statistical learning. 2001. *Springer*, 2001.

Hanwen Huang, Yufeng Liu, Ying Du, Charles M Perou, D Neil Hayes, Michael J Todd, and James Stephen Marron. Multiclass distance-weighted discrimination. *Journal of Computational and Graphical Statistics*, 22(4):953–969, 2013.

Xin Yee Lam, JS Marron, Defeng Sun, and Kim-Chuan Toh. Fast algorithms for large scale generalized distance weighted discrimination. *arXiv preprint arXiv:1604.05473*, 2016.

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

Yoonkyung Lee, Yi Lin, and Grace 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.

Yufeng Liu. Fisher consistency of multicategory support vector machines. In *AISTATS*, pages 291–298, 2007.

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

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

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

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

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

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

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

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

Xingye Qiao, Hao Helen Zhang, Yufeng Liu, Michael J Todd, and James Stephen Marron. Weighted distance weighted discrimination and its asymptotic properties. *Journal of the American Statistical Association*, 105(489):401–414, 2010.

Bernhard Schölkopf and Christopher JC Burges. *Advances in kernel methods: support vector learning*. MIT press, 1999.

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

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

Vladimir Naumovich Vapnik and Vlamimir Vapnik. *Statistical learning theory*, volume 1. Wiley New York, 1998.

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

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

Ji Zhu and Trevor Hastie. Kernel logistic regression and the import vector machine. In *Advances in Neural Information Processing Systems*, pages 1081–1088, 2001.

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