Multiclass Distance Weighted Discrimination with Applications to Batch Adjustment

Hanwen Huang[1], Yufeng Liu[1], Michael J. Todd[2], Ying Du[3], Charles M. Perou[3], D. Neil Hayes[3], and J. S. Marron[1]

November 9, 2010

Abstract

Combining different microarray data sets across platforms has the potential to generally increase statistical power. Distance Weighted Discrimination (DWD) is a powerful tool for solving binary classification problems in high dimensions. It has also been shown to provide an effective approach to binary cross-platform batch adjustment. In this paper, we extend the binary DWD to the multicategory case. In addition to the usual extensions which combine several binary DWD classifiers, we propose a global multiclass DWD (MDWD) which finds a single classifier that considers all classes at once. Our theoretical results show that MDWD is Fisher consistent, even in the particularly challenging case when there is no dominating class. The performances of different multiclass DWD methods are assessed through simulation studies. While the idea is applicable to general multicategory classification problems, we focus our application on batch adjustment. The effectiveness of the MDWD batch adjustment method is demonstrated through the application to a real microarray data set.

1 Introduction

Support Vector Machine (SVM), Vapnik (1998), Cristianini and Taylor (2000), Hastie et al. (2001) and Distance Weight Discrimination (DWD), Marron et al. (2007) are two commonly used large margin based classification methods. In the *binary* (2-class) classification problem, SVM seeks to find the separating hyperplane to maximize the minimum distance from each data point to the hyperplane. SVM will usually suffer from data-piling problems in High Dimension Low Sample Size (HDLSS) settings. DWD is a recently developed classification method which overcomes this issue and improves the generalizability in HDLSS data settings.

[1]Department of Statistics and Operations Research, University of North Carolina, Chapel Hill, NC 27599
[2]School of Operations Research and Information Engineering, Cornell University, Ithaca, NY 14853
[3]Lineberger Comprehensive Cancer Center, University of North Carolina, Chapel Hill, NC 27599

DWD seeks to find the separating hyperplane to minimize a notion of the average inverse distance from data points to the separating hyperplane.

Binary classification is a well studied special case. In many applications, *multicategory* problems are important as well. Binary classification methods can be generalized in many ways to handle multiple classes. Generalizations from binary SVM to multiclass SVM have been well studied in the literature. Two general strategies are commonly used to tackle the multicategory SVM problem. One strategy is to solve the multicategory problem by solving a series of binary problems. Examples include One-Versus-One (OVO) and One-Versus-The-Rest (OVR) approaches (Duda et al. (2000); Hastie et al. (2009)). The second strategy treats the population in a simultaneous fashion and considers all classes at once. Various methods along the line of the second strategy include Weston and Watkins (1999); Crammer and Singer (2000); Lee et al. (2004); Liu and Shen (2006); Liu and Yuan (2010). However, to our knowledge, generalization from binary DWD to multiclass DWD has not been studied. This article involves the study of the extension of DWD from the binary case to the multicategory case using both strategies.

For multiclass classification methods, one needs either to construct several binary classifiers or to solve a larger optimization problem which involves all classes at the same time. The OVO and OVR methods are computationally simple, and the global method is computationally more complex. However, the OVO method has the disadvantage of potential variance increase, because a smaller number of observations are used to learn each classifier. For the OVR method, it may fail under the circumstances when there is no dominating class, see Friedman (1996) and Lee et al. (2004). This leads to an interesting question of whether a more sophisticated method can achieve stronger results than the combination of several simple binary methods.

For multiclass SVM problems, comparisons of these three methods have been studied. Hsu and Lin (2002) conducted large-scale experiments and claimed that the OVO method is more suitable for practical use than the other methods. Lee et al. (2004) and Liu and Yuan (2010) demonstrated the superiority of their global method over the simple OVR method through some numerical studies. Rifkin and Klautau (2004) claimed that a simple OVR method is as accurate as any other approach. They supported their position by a critical review of the existing literatures and some experimental work. It is interesting to consider whether similar results can be obtained using the DWD method. We will carry out some simulation studies in this paper for all three methods and indicate the situations under which each specific method is preferred.

Microarray analysis has become a powerful tool in biological science. Microarray technologies allow for the measurement of thousands of gene expression levels simultaneously. The primary goal of a microarray study is to extract useful information from differential expression and provide insight into biological effects. However, nonbiological experimental variation such as batch effects are commonly observed in microarray experiments due to different experimental features. Large batch effects can make it difficult to obtain meaningful and accurate biological results and also make it difficult to integrate data from several sources or from multiple independent studies. Disregarding batch effects could result in misleading conclusions. Therefore, it is important and necessary to identify and adjust batch effects prior to microarray data analysis. Common approaches include mean/median centering, Singular Value Decomposition

(SVD Alter et al. (2000)) and ANOVA-like modeling (Wolfinger et al. (2001)) to balance the expression measurement across experiments. More sophisticated procedures have also been developed including an empirical Bayes method (Tibshirani et al. (2002); Johnson et al. (2007)), DWD (Benito et al. (2004); Liu et al. (2009)), and XPN (Shabalin et al. (2008)). See Scherer (2009) for a good review of this area.

The DWD classification method has been shown to provide effective batch adjustment for microarray data by Benito et al. (2004), and Liu et al. (2009). They also demonstrated that DWD can work better than SVM and SVD for the adjustment of systematic microarray effects. Benito et al. (2004) implement batch adjustment by first projecting the data onto the DWD normal direction and then moving the means of the two classes to a common point along that direction. When there are more than two batches, they take a stepwise approach. For example, for data including three batches, they first made a batch adjustment between Batches 1 and 2 (combined) and Batch 3. Next, they applied the same method to the adjusted data, to separate Batch 1 from Batch 2. This stepwise method creates an additional level of complexity especially when the number of batches considered is large because we need to decide which pair should be chosen in each step. For a K class problem, our proposed global multiclass DWD (MDWD) method will simultaneously produce K direction vectors which provide the basis of our new batch adjustment method. The K normal direction vectors determine a subspace which contains each class mean. We move each class in such a way that the class means move to a common point in this subspace. In Section 2 we will show how our new multiclass batch adjustment method gives better performance than any combination of binary methods.

The rest of the article is organized as follows. In Section 2 we present the batch adjustment results for a real data set using our MDWD method. Different types of multiclass DWD methods including OVO, OVR and MDWD are introduced in Section 3. Some theoretical properties of multiclass DWD are explored in Section 4. In Section 5 we present numerical results on simulated data to compare the performances of different methods. We provide some discussions in Section 6 and collect proofs of the theoretical results in the Appendix.

2 Batch Adjustment and Real Data

As mentioned in Verhaak et al. (2010), Glioblastoma Multiforme (GBM) is one of the most common forms of malignant brain cancer in adults. For the purposes of the current analysis, we selected a cohort of patients from The Cancer Genome Atlas Research Network with GBM cancer whose brain samples were assayed on three gene expression platforms (Affymetrix HuEx array, Affymetrix U133A array, and Agilent 244K array) and combined into a single unified data set. Four clinically relevant subtypes were identified using integrated genomic analysis in Verhaak et al. (2010), they are Proneural, Neural, Classical, and Mesenchymal. The data set came from seven batches and contained 168 patients with 1510 genes. Among the 168 samples, there are 56 Mesenchymal samples, 24 Neural samples, 52 Proneural samples, and 36 Classical samples.

Figure 1 studies the raw GBM data using a scatter plot matrix visualization based on the first four Principal Component (PC) axes. Observations from different batches are distin-

guished by different colors. The symbol types indicate the biological classes. The plots on the diagonal show the one-dimensional projections of the data onto each PC direction vector. A different height is added to each symbol just for convenient visual separation. In each diagonal plot we also include several smooth histograms, colored according to the batch label. The off-diagonal plots are projections of the data onto 2-d planes, determined by the various pairs of the PC directions. Note that Batch 5 (red color) is clearly separated from the rest of the batches in the PC1 direction. Figure 1 gives some suggestion of biological classes; for example in the PC 4 direction, Proneural (circle) seems to separate itself from the rest. However, this class is not very distinct in the sense the distances between batches are large relative to the distances between biological classes. Therefore, it will be very useful to remove the batch effects before doing data analysis.

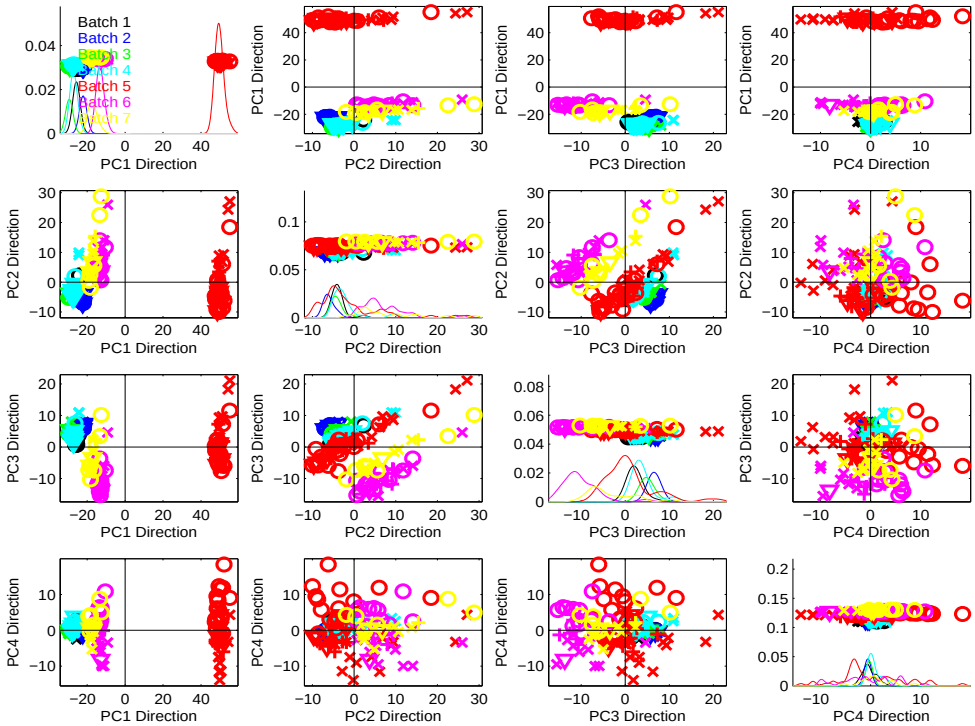

Figure 1: PCA projection scatter plot view of raw GBM data, showing 1D (diagonal) and 2D projections of raw data onto PC directions. Groupings of colors indicate batch biases. Samples from Classical, Mesenchymal, Proneural, and Neural are indicated by "+", "x", circle and triangle symbols respectively. This shows a very strong batch effect, so that adjustment is essential before combining data sets.

The steps of the proposed MDWD batch adjustment are as follows: (1) The MDWD direction vectors generate a subspace. (2) The subpopulations (e.g. respective batch subsets) are all projected onto that subspace. (3) The coordinates of the subpopulation projected means are computed. (4) Each subpopulation is shifted in such a way that its projected mean is moved in the subspace to a fixed point which is common to all subpopulations. An important advantage of the MDWD adjustment over PCA adjustment is that it preserves the variation that is not due to batch effects, because the MDWD directions maximize the separations between the

batches and ignore the variation in the data.

Figure 2 shows (using the same view) the same data after the MDWD adjustment. Now in all of the PC directions, the huge differences among batches visible in Figure 1 have disappeared, because the colors, representing the seven batches, are very well mixed. This means that the systematic sample batch effects in the data have been effectively removed. From Figure 2, Proneural (to the right) seems to separate from Mesenchymal (to the left) in the PC 2 direction, and the batch differences are much smaller in magnitude than the biological features in this data.

To further test the performance of our method, we apply a newly developed statistical tool called Standardized Within class Sum of Squares (SWISS Cabanski et al. (2010)) to this data set. SWISS allows for the comparison of different methods on a dataset in terms of how well they cluster a priori biological classes. A lower SWISS score means better separation of the cluster. The calculated SWISS score for the raw data and MDWD adjusted data are 0.9695 and 0.9361 respectively. The reduced SWISS score implies that our MDWD method is efficient at adjusting for the batch effects.

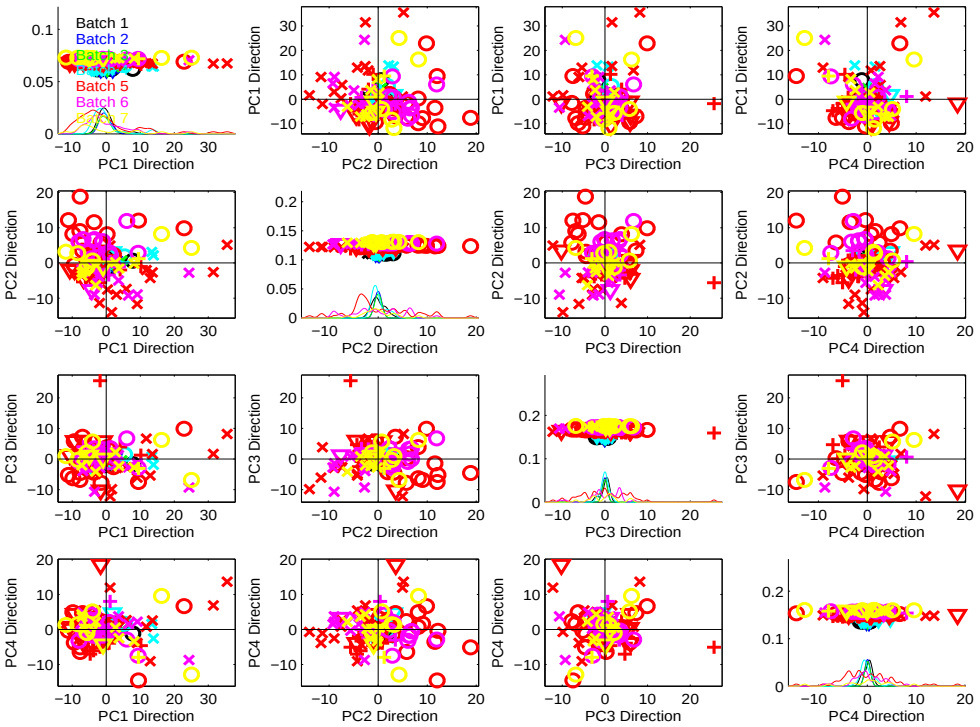

Figure 2: PCA scatter plot view of MDWD adjusted GBM data (labels are the same as in Figure 1), showing effective removal of batch biases. Note that biological class differences are now much more clear.

Adjusting batch effects in microarray data sets with more than two batches using the OVR and OVO methods can be implemented by the combination of a series of binary adjustments. The stepwise approach described in Benito et al. (2004) is based on the OVR DWD method. The batch adjustment using the OVO method also takes a stepwise approach as follows. In

each step, a pair of classes are combined together through a binary adjustment. So the number of unadjusted classes is reduced by one after each step. This process is repeated until all classes have been combined together.

The main drawback of the OVR and OVO adjustment methods is that their results depend on the order, i.e, which pair of classes are used in each binary problem. In each step, the number of options in constructing the binary problem increases with the number of total classes. Therefore, in the case where the number of classes considered is big, this can be a complicated problem because it is hard to find the optimal order among so many options. Moreover, the class size can be quite unbalanced which will further complicate the problem as shown in Qiao et al. (2010). A significant advantage of the MDWD method over the OVR and OVO methods is that it provides a convenient way to do batch adjustment for data sets with more than two batches. The MDWD method considers all batches at once and makes adjustment simultaneously for all batches.

3 Methodology

In the classification problem, we are given a training dataset consisting of n observations $(\mathbf{x}_i, y_i)$ for $i = 1, \cdots, n$. Here $\mathbf{x}_i \in R^d$ represents an input vector, and $y_i \in \{1, \cdots, K\}$ denotes the corresponding output class label. We assume that each $(\mathbf{x}_i, y_i)$ are independent random vectors distributed according to some unknown distribution function $P(\mathbf{x}, y)$. The task is to build a classification rule $\phi(\mathbf{x}) : R^d \to \{1, \cdots, K\}$ which can be used to predict the class label for a new input $\mathbf{x}$. In this section, we generalize binary DWD to the multiclass case. We first define OVR and OVO DWD which are based on solving several binary DWD classifications. Then we introduce MDWD which considers all classes in a single optimization.

3.1 OVR and OVO DWD

The OVR constructs K binary classifiers, each one trained to distinguish the examples in the single class from the examples in all remaining classes. When it is desired to classify a new example, the K classifiers are run, and the classifier which outputs the largest value is chosen.

In contrast to SVM, which seeks to maximize the smallest residual distance to the separating hyperplane, DWD aims to minimize the sum of inverse residuals. In particular, for the ith DWD classifier which is trained with all of the examples in the ith class with positive labels and all other examples with negative labels, we solve the following problem

$$\min_{\mathbf{w}^i, \beta^i, \xi^i} \sum_k \Big(\frac{1}{r_k} + C\xi_k^i \Big), \tag{1}$$

$$\begin{aligned} \text{subject to} \quad & r_k = (\mathbf{x}_k^T \mathbf{w}^i + \beta^i) + \xi_k^i, \text{ for k} : y_k = i, \\ & r_k = -(\mathbf{x}_k^T \mathbf{w}^i + \beta^i) + \xi_k^i, \text{ for k} : y_k \neq i, \\ & \mathbf{w}^{iT} \mathbf{w}^i \leq 1, \; r_k \geq 0, \; \xi_k^i \geq 0. \end{aligned} \tag{2}$$

After solving (1), there are K decision functions and we say $\mathbf{x}$ is in the class which has the largest value of the decision function, i.e. class of $\mathbf{x} = \text{argmax}_i(\mathbf{x}^T\mathbf{w}^i + \beta^i)$.

The OVO approach constructs $K(K-1)/2$ classifiers where each one is trained on data from two classes. For the classifier i, j which is trained on data from the ith class and the jth class, we solve the similar binary classification problem

$$\min_{\mathbf{w}^{ij},\beta^{ij},\xi^{ij}} \sum_{k:y_k=i \text{ or } y_k=j} \left(\frac{1}{r_k} + C\xi_k^{ij}\right), \tag{3}$$

$$\begin{aligned} \text{subject to} \quad & r_k = (\mathbf{x}_k^T\mathbf{w}^{ij} + \beta^{ij}) + \xi_k^{ij}, \text{ for k} : y_k = i, \\ & r_k = -(\mathbf{x}_k^T\mathbf{w}^{ij} + \beta^{ij}) + \xi_k^{ij}, \text{ for k} : y_k = j, \\ & \mathbf{w}^{ijT}\mathbf{w}^{ij} \leq 1, \ r_k \geq 0, \ \xi_k^{ij} \geq 0. \end{aligned} \tag{4}$$

There are different methods for combining the results of all $K(K-1)/2$ classifiers. The most commonly used method is called Friedman's "Max-wins" voting strategy: if $\text{sign}(\mathbf{x}_k^T\mathbf{w}^{ij} + \beta^{ij})$ says $\mathbf{x}$ is in the ith class, then the vote total for the ith class is increased by one; otherwise the vote total for the jth class is increased by one. Then we predict $\mathbf{x}$ is in the class with the largest vote total.

3.2 MDWD

Here we propose an approach for multiclass DWD problems by considering all classes at once and solving one single optimization problem simultaneously. We will show that the generalized formulation encompasses that of the two category DWD, regaining the desirable properties of the binary DWD. Consider a K-class classification problem. There are many different ways to represent classifiers. One of the most natural ways is to introduce a vector of discriminant functions $\mathbf{f} = (f_1, \cdots, f_K)$, where each component represents one class. For any new input $\mathbf{x}$, its label is estimated via a decision rule $\hat{y} = \text{argmax}_j f_j(\mathbf{x})$, where $f_j(\mathbf{x}) = \mathbf{x}^T\mathbf{w}_j + \boldsymbol{\beta}_j$.

For extension of DWD from the binary to the multiclass case, the objective function can be naturally constructed in such a way that it encourages f_y to be the largest among K functions. Here we formulate multiclass DWD in terms of the following optimization problem

$$\min_{\mathbf{w},\boldsymbol{\beta},\xi} \sum_{i=1}^{n} \sum_{k \neq j} \left(\frac{1}{r_{jk}^i} + C\xi_{jk}^i\right), \tag{5}$$

$$\begin{aligned} \text{subject to } & r_{jk}^i = f_j(x_i) - f_k(x_i) + \xi_{jk}^i, \text{ for } y_i = j, \\ & r_{jk}^i \geq 0, \ \xi_{jk}^i \geq 0, \ \sum_{j=1}^{K} \beta_j = 0, \ \sum_{k=1}^{K} ||\mathbf{w}_k||^2 = 1. \end{aligned} \tag{6}$$

Note that the i-th individual's contribution to the first term in the objective function (5) is the sum of the inverse of the differences between $f_{y_i}(\mathbf{x}_i)$ and all the other functions. This represents a natural generalization of the term $y_i f(\mathbf{x}_i)$ appearing in the binary DWD loss function. The parameter C in the second term in (5) controls the penalty on the variable ξ, the amount of

violation of classification. It also plays the role of tuning parameter. Similar to the binary case, using additional variables and constraints, the optimization problem (5) can be reformulated as a second-order cone programming problem.

The following Theorem shows that the solution of (5) satisfies the sum-to-zero constraint.

Theorem 1. *Let* $\mathbf{f}^*$ *be the minimizer of (5). Then* $\sum_{j=1}^{K} f_j^* = 0$.

Proof of this Theorem and other proofs are given in the Appendix. Note that in multiclass SVM, this sum-to-zero relationship is introduced as one of the constraints to ensure the uniqueness of the optimal solution. But here for multiclass DWD, we can show that the solution for $\mathbf{w}$ given in (5) automatically satisfies this sum-to-zero constraint. Therefore only $K-1$ direction vectors are independent which makes the minimizer unique and reduces the dimension of the original problem. If $K=2$, it is easy to show that the problem (5) reduces to the original binary DWD.

4 Theoretical Properties

In this section we study some of the statistical properties of multiclass DWD. We will focus on Fisher consistency. Fisher consistency is a desired condition for a classification method although a consistent method may not always give better classification accuracy. Fisher consistency has been well investigated for binary classification methods. However, it turns out that one can lose consistency in the generalization from the binary SVM to the multiclass SVM. We first study the Fisher consistency of OVO and OVR DWD in Section 4.1 and then study the Fisher consistency of MDWD in Section 4.2.

4.1 Fisher consistency of OVO and OVR DWD

It is easy to study the consistency property of the OVO type of approach to the multiclass classification problem, assuming that the properties of the corresponding binary classifiers have been well studied. Friedman (1996) pointed out that the "Max-wins" rule is equivalent to the Bayes rule when the class posterior probabilities $p_i = P(y=i|\mathbf{x})$ are known:

$$\text{argmax}_i(p_i) = \text{argmax}_i \Big[\sum_{j \neq i} I\Big(\frac{p_i}{p_i + p_j} > \frac{p_j}{p_i + p_j} \Big) \Big]. \tag{7}$$

Equation (7) suggests that the OVO method will be Fisher consistent as long as the consistency of its underlying binary classifiers is satisfied. This allows us to conclude that the OVO DWD is Fisher consistent since the Fisher consistency of binary DWD has been proved in Qiao et al. (2010).

For the OVR SVM, Lee et al. (2004) argued that Fisher consistency holds only in the case when there exists a dominating class, i.e., a class j with $p_j > 1/2$, because only the support vectors appear in each optimization, resulting in a flat region of the loss. More specifically, the

minimizer of the OVR SVM classifier is $\text{sign}(p_i - \frac{1}{2})$ for $i = 1, \cdots, K$. If there is a class j with $p_j > \frac{1}{2}$, then we can easily pick the majority class j because f_j would be near 1 and all of the other f_i would be close to -1. However, if there is no dominating class, then all f_i's would be close to -1, making the classifier inconsistent.

In sharp contrast, since DWD uses all data points, the resulting loss is smoothly decreasing, so Fisher consistency hold much more broadly in the sense that the solution satisfies $f_i^* > f_j^*$ if $p_i > p_j$ regardless of whether p_i is bigger than $\frac{1}{2}$ or not. The following theorem establishes Fisher consistency of the OVR DWD:

Theorem 2. *Let f_i^* be the minimizer of the ith binary DWD classifier defined in the OVR DWD method (5). Assume that the unique maximum of p_i for $i = 1, \cdots, K$ exists. Then $argmax_i(f_i^*) = argmax_i(p_i)$.*

4.2 Fisher Consistency of MDWD

Qiao et al. (2010) proved the Fisher consistency of binary DWD by using an equivalent formulation of the DWD optimization. We will show the Fisher consistency of multiclass DWD based on the extension of the equivalent formulation from binary case to multiclass case.

For each $i = 1, \cdots, n$ and $k = \{1, \cdots, K\}/\{y_i\}$, we define $f_{y_i k}^i = f_{y_i}^i - f_k^i = (\mathbf{x}_i^T \mathbf{w}_{y_i} + \beta_{y_i}) - (\mathbf{x}_i^T \mathbf{w}_k + \beta_k)$. The multiclass DWD optimization problem (5) can be shown to be equivalent to the following problem

$$\min_{\mathbf{w}, \boldsymbol{\beta}: \mathbf{w}^T \mathbf{w} \leq 1} \min_{\xi \geq 0} \sum_{i=1}^{n} \sum_{k \neq j} \left(\frac{1}{f_{y_i k}^i + \xi_{y_i k}^i} + C \xi_{y_i k}^i \right). \tag{8}$$

It can be shown that the optimal solution for the inside optimization part of (8) is given by $(\xi_{y_i k}^i)^* = \frac{1}{\sqrt{C}} - f_{y_i k}^i$ if $f_{y_i k}^i \leq \frac{1}{\sqrt{C}}$; $(\xi_{y_i k}^i)^* = 0$ otherwise. Then the multiclass DWD problem amounts to

$$\min_{\mathbf{w}, \boldsymbol{\beta}} \sum_{i=1}^{n} \sum_{k \neq y_i} \left([2\sqrt{C} - C f_{y_i k}^i] I \left[f_{y_i k}^i \leq \frac{1}{\sqrt{C}} \right] + \frac{1}{f_{y_i k}^i} I \left[f_{y_i k}^i \geq \frac{1}{\sqrt{C}} \right] \right) \tag{9}$$

$$\text{subject to } \sum_{k=1}^{K} ||\mathbf{w}_k||^2 = 1. \tag{10}$$

If we define the multiclass DWD loss function as

$$V(\mathbf{f}, y) \;=\; \sum_{j \neq y} l(f_{yj}), \tag{11}$$

where

$$l(f_{yj}) \;=\; \begin{cases} 2\sqrt{C} - C f_{yj} & \text{if} f_{yj} \leq \frac{1}{\sqrt{C}} \\ \frac{1}{f_{yj}} & \text{otherwise,} \end{cases}$$

then the multiclass DWD optimization is $\min_{\mathbf{w},\boldsymbol{\beta}} \sum_{i=1}^{n} V(\mathbf{f}(\mathbf{w},\boldsymbol{\beta}), y_i),\ s.t.\ \sum_{k=1}^{K} ||\mathbf{w}_k||^2 \leq 1$.

Consider $y \in \{1, \cdots, K\}$ and let $p_j(\mathbf{x}) = P(y = j|\mathbf{x})$. For any classification function $\mathbf{f} = (f_1, \cdots, f_K)$, the expected multiclass DWD loss, that is, the risk, is $R(\mathbf{f}) = E(E(V(\mathbf{f}(\mathbf{x}), y)|\mathbf{x}))$. Fisher consistency requires that $\text{argmax}_j f_j^* = \text{argmax}_j p_j$, where $\mathbf{f}^*(\mathbf{x}) = (f_1^*(\mathbf{x}), \cdots, f_k^*(\mathbf{x}))$ denotes the minimizer of $R(\mathbf{f})$. Theorem 3 shows the Fisher consistency of multiclass DWD.

Theorem 3. *Let $\mathbf{f}^*$ be the global minimizer of $R(\mathbf{f}) = E(E(V(\mathbf{f}(\mathbf{x}), y)|\mathbf{x}))$, where $V(\cdot)$ is the multiclass DWD loss given in (11). Assume that the unique maximum of p_j for $j = 1, \cdots, K$ exists. Then $argmax_j(f_j^*) = argmax_j(p_j)$.*

There are previous studies on Fisher consistency of multiclass SVM methods such as Zhang (2004); Lee et al. (2004); Tewari and Bartlett (2005); Liu (2007); Zou et al. (2008). Liu (2007) summarized the Fisher consistency properties of four commonly used SVM loss functions:

(a) (Zou et al. (2008)) $[1 - f_y(\mathbf{x})]_+$;

(b) (Lee et al. (2004)) $\sum_{j \neq y} [1 + f_j(\mathbf{x})]_+$;

(c) (Vapnik (1998); Weston and Watkins (1999); Bredensteiner and Bennett (1999)) $\sum_{j \neq y} [1 - (f_y(x) - f_j(x))]_+$;

(d) (Crammer and Singer (2000); Liu and Shen (2006)) $[1 - \min_j (f_y(\mathbf{x}) - f_j(\mathbf{x}))]_+$.

It was shown in Liu (2007) that, under the sum-to-zero constraint, except for loss (b), these losses are not always Fisher consistent. Two approaches were proposed in Liu (2007) to modify inconsistent losses to be consistent. It is interesting to see that the DWD loss function we used in (11) is related to the SVM counterpart (c). But the DWD loss function yields a Fisher consistent classifier without modification. The reason is that the loss function (11) is continuously differentiable as opposed to the SVM loss function which is not differentiable. This appealing property of DWD is due to the fact that all data points have a direct influence, instead of only the support vectors.

5 Simulations

In this section, simulations are conducted to investigate the performance of the proposed OVR, OVO and MDWD methods. For comparison, the results from the Bayes classifiers, which are derived based on the true underlying distributions, are also included.

The simulated data sets include training, tuning and test sets. We generated the tuning and testing data from the same distributions as the training data. For the reason noted in Shao (1993), we set the sample sizes of tuning sets equal to that of the training sets. The sizes of the test sets are taken to be 10 times bigger than that of the training sets. Each experiment was replicated 100 times. Tuning sets are used to choose the tuning parameter C through a grid search, and the testing errors, evaluated on independent testing data, are used to measure the accuracy of various classifiers.

We have tried many different settings, including both low dimensional and high dimensional. To save space, we only report the results from the high-dimensional settings since the focus of MDWD is on the high-dimensional situations. We consider HDLSS settings with $d = 1000$ in all simulations.

Our simulation results show that in situations where each class can be well separated from the rest, the performance of all three multiclass DWD methods are quite similar and are close to the optimal Bayes rule. We do not explicitly show these results here. The first example we show belongs to the situations where not all individual classes can be well separated from the rest. The data include three classes with the sample size of each training class being 50. The three classes are generated using three different Gaussian distributions with unit covariance and the first two components of the mean vectors as $(-5, 0)$, $(5, 0)$ and $(0, 1)$. The rest of the $d - 2$ dimensions are pure noise, i.e., all sampled from the standard normal distribution. If it is known that one should look in the direction of the first two coordinates, then the three classes are easy to separate, as shown by the tiny test error of the Bayes rule. However, in high dimensions, it can be quite challenging to find those directions. To investigate the generalizability property of the different methods, we exhibit the average performance over 100 replications in the first row of Table 1. The table summarizes the mean and standard error (over the 100 replications) of the proportion (out of 1500 members of each test data set) of incorrect classifications. Note that none of the three methods can achieve results close to optimal. But both OVO and MDWD are quite comparable, and much better than OVR, which is consistent with the ideas of Friedman (1996).

Table 1: Test errors (in percentage) over 100 replications

	OVR	OVO	MDWD	Bayes
Example 1	16.18	7.59	7.59	0.72
	(0.11)	(0.08)	(0.08)	(0.02)
Example 2	9.45	8.64	6.96	4.67
	(0.15)	(0.12)	(0.14)	(0.05)
Example 3	19.36	19.81	18.02	15.23
	(0.06)	(0.05)	(0.13)	(0.07)
Example 4	29.00	24.63	15.28	0.70
	(0.17)	(0.19)	(0.20)	(0.02)
Example 5	30.96	28.75	19.08	3.39
	(0.17)	(0.18)	(0.27)	(0.06)

Examples 2 is a case where MDWD is the best of these three methods. The data include three classes with the same sample size as Example 1. The first two components of the distributions for the three classes are Gaussians with means $(-10, 0)$, $(10, 0)$ and $(0, 2)$ and variances $(5, 1)$, $(5, 1)$ and $(1, 2)$. Figure 3 shows the projections of the data points and the decision boundaries onto the first two directions. The first, second and third classes have $n_1 = n_2 = n_3 = 50$ data vectors denoted by red plus, blue square, and white circle signs respectively. The optimal Bayes decision boundary is quadratic for this case due to the fact

that the variances are different among three classes. In this example, classes 1 and 2 can be easily separated and it is challenging to separate class 3 from the other two. The MDWD classifier results in a decision area for class 3 which is close to the one provided by the Bayes rule. In contrast, the OVR and the OVO decision areas for class 3 are either too thin or too wide near the bottom of plot where most data lie. The small distances and different covariances among classes make it difficult to do separation using the OVR and OVO methods. The MDWD method can provide improvement in this situation as shown by both the test error rate in the second row of Table 1 and the illustration in Figure 3.

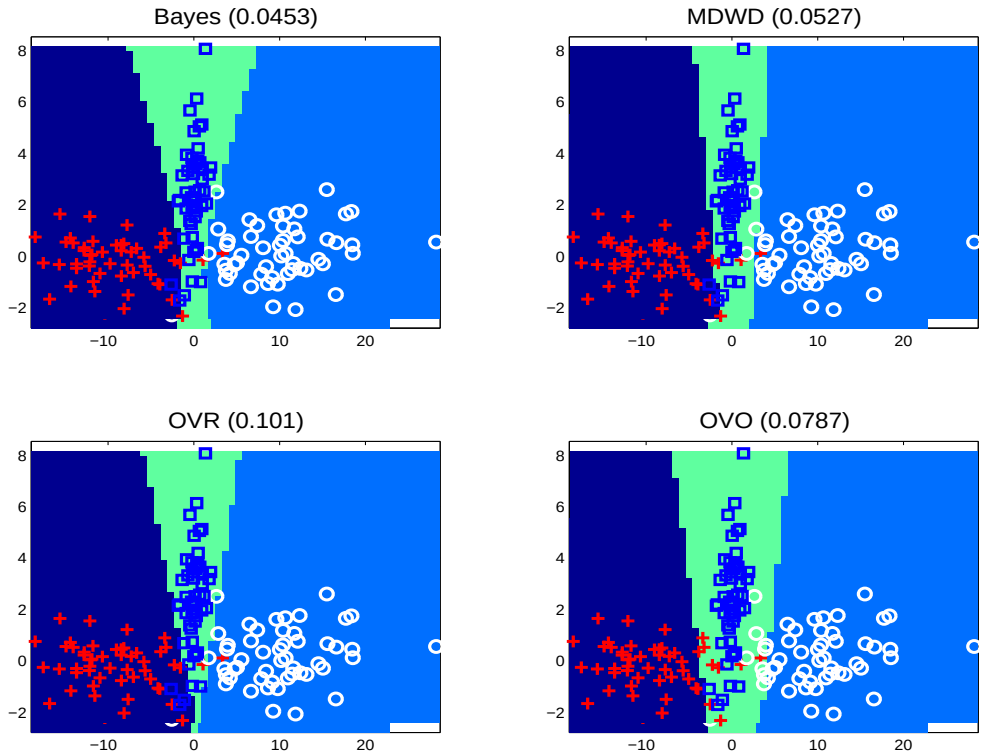

Figure 3: Plots of data points and decision boundaries in the first two coordinate axis directions for one training set of Example 2. Upper left panel for Bayes boundary, upper right for MDWD, lower left for OVR, lower right for OVO. The numbers in the parentheses show the test errors for this set.

Example 3 also includes three classes with the same sample size as the previous two examples. The distributions of the first two classes are single Gaussians with the first two components of mean vectors as $(-10, 0)$ and $(10, 0)$. However, the third class is a mixture of two Gaussian distributions with the first two components of mean vectors as $(10, 1)$ and $(10, 20)$. We have 60% of the data from the first Gaussian component and the remaining from the second component. The small distance between the second component of class 3 and class 2 makes it difficult to separate these two classes using binary DWD. The MDWD method can provide improvement in this situation. It considers all data points from the three classes simultaneously and the impact of the second component in class 3 can help the separation between classes 2 and 3. Thus it improves the test error rate over the OVO method as shown in the third row of Table 1.

All three of the above examples are balanced designs, i.e., the sample size of each class is the same. Examples 4 and 5 are unbalanced cases where different classes have different sample sizes. Example 4 includes three classes with training sample sizes being 50, 20, and 30 respectively. The distributions of the three classes are the same as those in Example 1. The test errors for this example (the fourth row of Table 1) show clear improvement of the MDWD method over the other two. Example 5 includes four classes with training sample sizes being 50, 20, 30, and 10 respectively. The distributions of the four classes are Gaussian with unit covariance and the first three components of the mean vectors being (-5,0,0), (5,0,0), (0,2,0) and (0,0,2) respectively. The outperformance of the MDWD method over the other methods for this example can be shown in the fifth row of Table 1. Examples 4 and 5 show that the MDWD method can give a big improvement in classification error rate over the OVO and OVR methods in unbalanced situations. This is a quite appealing property of MDWD because real data are often unbalanced.

6 Discussion

In this article we have extended the DWD classification method to the multicategory case. In addition to the OVR and OVO approaches which solve the multicategory problem via a sequence of binary DWD, we have proposed a new MDWD approach which generalizes the binary DWD to a simultaneous multicategory formulation. Our theoretical results show that MDWD is Fisher consistent even in the absence of a dominating class for multicategory problems. The simulation studies show that our MDWD method can always work as well as, and frequently better than, the existing OVR and OVO methods in multicategory problems.

An important direct application of our MDWD is to provide a powerful method for the adjustment of various types of systematic biases such as source and batch effects in microarray experiments. We have demonstrated the usefulness of this method through application to a microarray data set. We recommend MDWD as a general approach for removing systematic bias effects from microarray data and for merging different data sets.

Although our focus in this article is on the application of batch adjustment, the proposed MDWD method can also be applied to general multicategory classification problems, as indicated by our simulation studies. An important future research issue is the HDLSS asymptotics. Hall et al. (2005) showed that under certain conditions, there exists a geometric representation of data in the high dimensional case. This representation has been successfully applied to study the asymptotic properties of binary classifiers such as SVM, DWD, and BDD (Hall et al. (2005); Qiao et al. (2010); Huang et al. (2010)). However, no HDLSS asymptotic studies have been carried out for multiclass classifiers even for the SVM method. In future research, we will use this geometric representation to study the asymptotic behaviors of the proposed multicategory DWD classifier in HDLSS settings.

Appendix

Proof of Theorem 1: Note that the objective function (5) only depends on the direction vectors $\mathbf{w}$ through the form of pairwise differences $(\mathbf{w}_j - \mathbf{w}_k)$. Thus adding any constant vector $\mathbf{w}_0$ to all $\mathbf{w}_j$ will not change the objective function value. We can rewrite the direction vectors $\mathbf{w}$ as $\mathbf{w}_1 = \mathbf{w}_0, \mathbf{w}_2 = \mathbf{w}_0 + \mathbf{w}_{12}, \cdots, \mathbf{w}_K = \mathbf{w}_0 + \cdots + \mathbf{w}_{K-2,K-1}$. Then the $\mathbf{w}_0$ term only appears in the constraint $S = \sum_{k=1}^{K} ||\mathbf{w}_k||^2 = ||\mathbf{w}_0||^2 + ||\mathbf{w}_0 + \mathbf{w}_{12}||^2 + \cdots + ||\mathbf{w}_0 + \cdots + \mathbf{w}_{K-2,K-1}||^2 = 1$. Since both the objective function (5) and the constraints (6) are continuously differentiable with respect to $\mathbf{w}$, according to the Karush-Kuhn-Tucker (KKT) condition, the solution $\mathbf{w}^*$ should satisfy $\frac{\partial S}{\partial \mathbf{w}_0^*} = 2(\mathbf{w}_0^* + (\mathbf{w}_0^* + \mathbf{w}_{12}^*) + \cdots + (\mathbf{w}_0^* + \cdots + \mathbf{w}_{K-2,K-1}^*)) = \sum_{j=1}^{K} \mathbf{w}_j^* = \mathbf{0}$. Combining this equation with the constraint $\sum_{j=1}^{K} \beta_j = 0$ we immediately get $\sum_{j=1}^{K} f_j^* = 0$. □

Proof of Theorem 2:

From Qiao et al. (2010), we get that the minimizer for the ith binary classifier in the One-vs-the-rest DWD method is

$$f_i^* = \frac{1}{\sqrt{C}} \begin{cases} \sqrt{\frac{p_i}{1-p_i}} & \text{if } p_i > \frac{1}{2} \\ \sqrt{\frac{1-p_i}{p_i}} & \text{if } p_i < \frac{1}{2}. \end{cases}$$

Thus, we can easily show that $f_i^* > f_j^*$ if $p_i > p_j$ regardless whether p_i is bigger than $\frac{1}{2}$ or not. Hence the Theorem immediately follows. □

Proof of Theorem 3: Note that $R(\mathbf{f}) = E(E(V(\mathbf{f}(\mathbf{x}), y)|\mathbf{x}))$, We can minimize $R(\mathbf{f})$ by minimizing $E(V(\mathbf{f}(\mathbf{x}), y)|\mathbf{x})$ for every $\mathbf{x}$. For any fixed $\mathbf{x}$, $E(V(\mathbf{f}(\mathbf{x}), y)|\mathbf{x})$ can be written as $\sum_{j=1}^{K} p_j(\mathbf{x})[\sum_{k \neq j} l(f_{jk})]$. For any given $\mathbf{X} = \mathbf{x}$, assume that $p_j(\mathbf{x}) > p_k(\mathbf{x})$. Then we can conclude that the solution of $f_j^*(\mathbf{x}) \leq f_k^*(\mathbf{x})$. To show this, suppose that $f_j^*(\mathbf{x}) < f_k^*(\mathbf{x})$, it is easy to see that switching $f_j^*(\mathbf{x})$ and $f_k^*(\mathbf{x})$ will yield a smaller objective value due to the decreasing property of l. Without loss of generality, assume that $p_1(\mathbf{x}) > p_2(\mathbf{x}) \geq p_3(\mathbf{x}) \cdots \geq p_K(\mathbf{x})$, which implies that the minimizer must satisfy $f_1^*(\mathbf{x}) \geq f_2^*(\mathbf{x}) \geq \cdots \geq f_K^*(\mathbf{x})$. We need to show that $f_1^*(\mathbf{x}) > f_2^*(\mathbf{x})$. Consider $f_1 - f_2 = s_1$, $f_2 - f_3 = s_2$, $\cdots$, $f_{K-1} - f_K = s_{K-1}$, The problem reduces to

$$\min_{\boldsymbol{s}} L(\boldsymbol{s}) \tag{12}$$

$$\text{subject to } s_j \geq 0, \ j = 1, \cdots, K-1, \tag{13}$$

where

$$L(\boldsymbol{s}) = \sum_{k=1}^{K} p_k(l(-s_1 - \cdots - s_{k-1}) + \cdots + l(-s_{k-1}) + l(s_k) + \cdots + l(s_k + \cdots + s_{K-1})).$$

Since both the objective function and the constraints are continuously differentiable, the opti-

mal solution of (12) must satisfy Karush-Kuhn-Tucker (KKT) condition, i.e.

$$\begin{aligned}
& \frac{\partial L(\boldsymbol{s})}{\partial s_i} - \alpha_i \\
= & \sum_{k=1}^{i} p_k(l'(s_k + \cdots + s_i) + \cdots + l'(s_k + \cdots + s_{K-1})) \\
& + \sum_{k=i+1}^{K} p_k(l'(-s_1 - ... - s_{k-1}) - \cdots + l'(-s_i - \cdots - s_{k-1})) - \alpha_i \\
= & \sum_{k=1}^{i} p_k(l'(s_k + \cdots + s_i) + \cdots + l'(s_k + \cdots + s_{K-1})) + iC \sum_{k=i+1}^{K} p_k - \alpha_i \\
= & \ 0, \qquad (14)
\end{aligned}$$

where

$$\alpha_i \geq 0 \ and \ \alpha_i s_i^* = 0, \ for \ all \ i = 1, \cdots .K-1 \qquad (15)$$

It is sufficient to show that $s_1^* = 0$ is not a minimizer. Toward this end, suppose that $s_1^* = 0$, we have

$$\begin{aligned}
\frac{\partial L}{\partial s_1} &= p_1(l'(s_1) + \cdots + l'(s_1 + \cdots + s_{K-1})) + \sum_{k=2}^{K} Cp_k \\
&= p_1(l'(0) + l'(s_2^*) \cdots + l'(s_2^* \cdots + s_{K-1}^*)) + \sum_{k=2}^{K} Cp_k = \alpha_1, \qquad (16)
\end{aligned}$$

and

$$\begin{aligned}
\frac{\partial L}{\partial s_2} &= p_1(l'(s_1 + s_2) + \cdots + l'(s_1 + \cdots + s_{K-1})) \\
& \quad + p_2(l'(s_2) + \cdots + l'(s_2 + \cdots + s_{K-1})) + 2\sum_{k=3}^{K} Cp_k \\
&= (p_1 + p_2)l'(s_2) + \cdots + l'(s_2 + \cdots + s_{K-1}) + 2\sum_{k=3}^{K} Cp_k = \alpha_2. \qquad (17)
\end{aligned}$$

From (16) we have

$$l'(s_2) + \cdots + l'(s_2 + \cdots + s_{K-1}) = \frac{\alpha_1 - \sum_{k=2}^{K} Cp_k + Cp_1}{Cp_1} = \frac{\alpha_1 - C + 2Cp_1}{Cp_1}.$$

Substitute into (17), we have

$$\begin{aligned}
\alpha_2 &= (p_1 + p_2)\frac{\alpha_1 - C + 2Cp_1}{p_1 C} + 2\sum_{k=3}^{K} Cp_k \\
&= \frac{(p_1 + p_2)\alpha_1 + C(p_1 - p_2)}{Cp_1} > 0, \qquad (18)
\end{aligned}$$

which implies $s_2^* = 0$ from the fact that $\alpha_2 s_2 = 0$.

Suppose that $\alpha_j = 0$ for all $j = 1, \cdots, i-1$. From (16), we have

$$l'(s_i) + \cdots + l'(s_i + \cdots + s_{K-1}) = \frac{\alpha_1 - \sum_{k=2}^{K} Cp_k + (i-1)Cp_1}{Cp_1} = \frac{\alpha_1 - C + iCp_1}{Cp_1}.$$

Then substitute into the ith formulae, we have

$$\begin{aligned} \alpha_i &= (p_1 + \cdots + p_i)(l'(s_i) + \cdots + l'(s_i + \cdots + s_{K-1})) + i \sum_{k=i+1}^{K} Cp_k \\ &= (p_1 + \cdots + p_i)\frac{\alpha_1 - C + iCp_1}{Cp_1} + iC(1 - (p_1 + \cdots + p_i)) \\ &= \frac{(p_1 + \cdots + p_i)\alpha_1 + iCp_1 - C(p_1 + \cdots + p_i)}{Cp_1} > 0, \end{aligned} \tag{19}$$

thus we have $s_i^* = 0$. We conclude that $s_j^* = 0$ for all $j = 1, \cdots, K-1$. But from (16), we have that

$$\alpha_1 = (K-1)p_1 l'(0) + \sum_{k=2}^{K} Cp_k = C(\sum_{k=2}^{K} Cp_k - (K-1)p_1 < 0,$$

which is contradict to the KKT requirement that $\alpha_1 \geq 0$. Thus $s_1^* = 0$ can not be the minimizer which implies f_1^* is the unique maximum. □

References

Alter, O., P. O. Brown, and D. Botstein (2000). Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences of the United States of America 97*(18), 10101–10106.

Benito, M., J. Parker, Q. Du, L. Skoog, A. Lindblom, C. M. Perou, and J. S. Marron (2004). Adjustment of systematic microarray data biases. *Bioinformatics 20*, 105–144.

Bredensteiner, E. J. and K. P. Bennett (1999). Multicategory classification by support vector machines. *Computational Optimizations and Applications 12*, 53–79.

Cabanski, C. R., Y. Qi, X. Yin, E. Bair, M. C. Hayward, C. Fan, J. Li, M. D. Wilkerson, J. S. Marron, C. M. Perou, and D. N. Hayes (2010, 03). Swiss made: Standardized within class sum of squares to evaluate methodologies and dataset elements. *PLoS ONE 5*(3), e9905.

Crammer, K. and Y. Singer (2000). On the learnability and design of output codes for multiclass problems. In *In Proceedings of the Thirteenth Annual Conference on Computational Learning Theory*, pp. 35–46.

Duda, R. O., P. E. Hart, and D. G. Stork (2000). *Pattern Classification*. Wiley-Interscience Publication.

Friedman, J. H. (1996). Another approach to polychotomous classification. Technical report, Department of Statistics, Stanford University.

Hall, P., J. Marron, and A. Neeman (2005). Geometric representation of high dimension, low sample size data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67*(3), 427–444.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning* (second ed.). Springer.

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

Huang, H., Y. Liu, and J. S. Marron (2010). Bi-directional discrimination with application to data visualization. Submitted.

Johnson, W. E., C. Li, and A. Rabinovic (2007). Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics 8*(1), 118–127.

Lee, Y., Y. Lin, and G. Wahba (2004). 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), 67–82.

Liu, X., J. Parker, C. Fan, C. M. Perou, and J. S. Marron (2009). Visualization of cross-platform microarray normalization. In *Batch Effects and Noise in Microarray Experiments: Sources and Solutions (A. Scherer, ed.)*, pp. 167–181. Wiley,New York.

Liu, Y. (2007). Fisher consistency of multicategory support vector machines. In *Eleventh International Conference on Artificial Intelligence and Statistics*, pp. 289–296.

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

Liu, Y. and M. Yuan (2010). Reinforced multicategory support vector machines. *Journal of Computational and Graphical Statistics*, to appear.

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

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

Rifkin, R. and A. Klautau (2004). In defense of one-vs-all classification. *The Journal of Machine Learning Research 5*, 101–141.

Scherer, A. (2009). *Batch Effects and Noise in Microarray Experiments: Sources and Solutions.* Wiley,New York.

Shabalin, A. A., H. Tjelmeland, C. Fan, C. M. Perou, and A. B. Nobel (2008). Merging two gene-expression studies via cross-platform normalization. *Bioinformatics 24*(9), 1154–1160.

Shao, J. (1993). Linear model selection by cross-validation. *Journal of the American Statistical Association 88*(422), pp. 486–494.

Tewari, A. and P. L. Bartlett (2005). On the consistency of multiclass classification methods. In P. Auer and R. Meir (Eds.), *COLT*, Volume 3559 of *Lecture Notes in Computer Science*, pp. 143–157. Springer.

Tibshirani, R., T. Hastie, B. Narasimhan, and G. Chu (2002). Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences of the United States of America 99*(10), 6567–6572.

Vapnik, V. N. (1998). *Statistical Learning Theory*. Springer.

Verhaak, R. G., K. A. Hoadley, E. Purdom, V. Wang, Y. Qi, M. D. Wilkerson, C. R. Miller, L. Ding, T. Golub, J. P. Mesirov, G. Alexe, M. Lawrence, M. O'Kelly, P. Tamayo, B. A. Weir, S. Gabriel, W. Winckler, S. Gupta, L. Jakkula, H. S. Feiler, J. G. Hodgson, C. D. James, J. N. Sarkaria, C. Brennan, A. Kahn, P. T. Spellman, R. K. Wilson, T. P. Speed, J. W. Gray, M. Meyerson, G. Getz, C. M. Perou, D. N. Hayes, and Cancer Genome Atlas Research Network (2010). Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in pdgfra, idh1, egfr, and nf1. *Cancer cell 17*(1), 98–110.

Weston, J. and C. Watkins (1999). Support vector machines for multi-class pattern recognition. In *Proceedings of the Seventh European Symposium on Artificial Neural Networks*, pp. 219–224.

Wolfinger, R. D., G. Gibson, E. D. Wolfinger, L. Bennett, H. Hamadeh, P. Bushel, C. Afshari, and R. S. Paules (2001). Assessing gene significance from cdna microarray expression data via mixed models. *J. Comput. Biol. 8*, 625–637.

Zhang, T. (2004). Statistical analysis of some multi-category large margin classification methods. *J. Mach. Learn. Res. 5*, 1225–1251.

Zou, H., J. Zhu, and T. Hastie (2008). New multicategory boosting algorithms based on multicategory fisher-consistent losses. *The Annals of Applied Statistics*, 1290–1306.