Statistica Sinica Preprint No: SS-2016-0260R2

Title	Stability Enhanced Large-Margin Classifier Selection
Manuscript ID	SS-2016-0260R2
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202016.0260
Complete List of Authors	Will Wei Sun Guang Cheng and Yufeng Liu
Corresponding Author	Will Wei Sun
E-mail	sunweisurrey8@gmail.com

Statistica Sinica: Preprint
doi:10.5705/ss.202016.0260

Stability Enhanced Large-Margin Classifier Selection

Running title: Large-Margin Classifier Selection

Will Wei Sun[*], Guang Cheng[†], Yufeng Liu[‡]

Abstract

Stability is an important aspect of a classification procedure as unstable predictions can potentially reduce users' trust in a classification system and harm the reproducibility of scientific conclusions. We introduce a concept of classification instability, decision boundary instability (DBI), and incorporate it with the generalization error (GE) as a standard for selecting the *most accurate and stable* classifier. For this, we implement a two-stage algorithm: (i) select a subset of classifiers whose estimated GEs are not significantly different from the minimal estimated GE among all the candidate classifiers; (ii) take the optimal classifier to be the one achieving the minimal DBI among the subset selected in stage (i). This selection principle applies to both linear and nonlinear classifiers. Large-margin classifiers are used as a prototypical example to illustrate this idea. Our selection method is shown to be consistent in the sense that the optimal classifier simultaneously achieves the minimal GE and the minimal DBI.

[*] Assistant Professor, Department of Management Science, University of Miami, FL 33156, Email: wsun@bus.miami.edu. This work was carried out during Will's PhD period at Purdue University.

[†] Corresponding Author. Professor, Department of Statistics, Purdue University, West Lafayette, IN 47906, Email: chengg@purdue.edu. Office phone: 765-496-9549. Partially supported by NSF CAREER Award DMS-1151692, DMS-1418042 and Office of Naval Research (ONR N00014-15-1-2331).

[‡] Professor, Department of Statistics and Operations Research, Department of Genetics, Department of Biostatistics, Carolina Center for Genome Sciences, Lineberger Comprehensive Cancer Center, University of North Carolina Chapel Hill, NC 27599, Email: yfliu@email.unc.edu. Partially supported by NSF DMS-1407241, NIH/NCI P01 CA-142538, and NSF IIS 1632951.

Various simulations and examples further demonstrate the advantage of our method over alternative approaches.

Keywords: Asymptotic normality, Large-margin, Model selection, Selection consistency, Stability.

1 Introduction

Classification aims to identify the class label of a new subject using a classifier constructed from training data whose class memberships are given. It has been widely used in such fields as medical diagnosis, fraud detection, and natural language processing. Classification methods have been successfully developed with classical approaches such as Fisher's linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), and logistic regression (see Hastie et al. (2001) for a comprehensive review), and modern approaches such as the support vector machine (SVM) (Cortes and Vapnik (1995)) and boosting (Freund and Schapire (1997)). Liu et al. (2011) proposed a platform, large-margin unified machine (LUM), for unifying various large margin classifiers ranging from soft to hard.

In the literature, much of the research has focused on improving the predictive accuracy of classifiers and hence generalization error (GE) is often the primary criterion for selecting the optimal one from the rich pool of existing classifiers; see Vapnik (1998) and Steinwart (2007). Recently, researchers have started to explore alternative measures to evaluate the performance of classifiers. For instance, besides prediction accuracy, computational complexity and training time of classifiers are considered in Lim et al. (2000). Wu and Liu (2007) proposed the robust truncated hinge loss SVM to improve the robustness of the standard SVM. Qiao and Liu (2009) and Wang (2013) investigated several measures of cost-sensitive weighted generalization errors for highly unbalanced classification tasks since, in this case, GE itself is not sufficiently informative. In this paper, we focus on the stability of a classification procedure. Stability has received attention in statistics and machine learning. For example, Wang (2010) employed clustering instability as a criterion to select the number of clusters; Adomavicius and Zhang (2010) introduced stability as a new performance measure for recommender systems; Meinshausen and Bühlmann (2010) and Shah and Samworth (2013) used stability for variable selection; Sun et al. (2013) applied variable selection stability for model selection, and Lim and Yu (2016) incorporated estimation stability into the tuning parameter selection of regularized regression models. While successes of stability have been reported in these works, little has been done for classification stability itself, expect for some results on nearest neighbor classifiers (Sun et al. (2016)). Consequently, there is a need for a systematic study of stability in a general classification context.

We introduce a notion of decision boundary instability (DBI) to assess the stability (Breiman (1996)) of a classification procedure arising from the randomness of training samples. Providing a stable prediction plays a crucial role on users' trust of a classification system. In the psychology literature, for example, it has been shown that advice-giving agents with larger variability in past opinions are considered less informative and less helpful than those with a more consistent pattern of opinions (Gershoff et al. (2003); Van Swol and Sniezek (2005)). Then too, scientific conclusions should be reproducible with respect to small perturbations of data. Reproducible research has recently received much attention in statistics (Yu (2013)), biostatistics (Kraft et al. (2009); Peng (2009)), computational science (Donoho et al. (2009)) and other scientific communities (Ioannidis (2005)). A classification procedure with more stable prediction performance is preferred when researchers aim to reproduce the reported results from randomly generated samples.

We attempt to select the *most accurate and stable* classifier by incorporating DBI into our selection process. We suggest a two-stage selection procedure: (i) eliminate the classifiers whose GEs are significantly larger than the minimal one among all the candidate classifiers; (ii) select the optimal classifier as that has the minimal DBI among the remaining classifiers.

In the first stage, we show that the cross-validation estimator for the difference of GEs induced from two large-margin classifiers is asymptotically Gaussian, which enables us to construct a confidence interval for the GE difference. If this confidence interval contains 0, the classifiers are considered indistinguishable in terms of GE. By applying this, we can obtain a collection of potentially good classifiers whose GEs are close enough to the minimal value. In the second stage, we check whether the collection of potentially good classifiers perform well in terms of their stability by invoking a further selection criterion DBI. This measure can precisely reflect the visual variability in the decision boundaries due to perturbed training samples.

This two-stage selection algorithm is shown to be consistent in the sense that the selected optimal classifier simultaneously achieves the minimal GE and the minimal DBI. The proof is nontrivial because of the stochastic nature of the two-stage algorithm. Our method is distinguished from bias-variance analysis in classification since the latter focuses on the decomposition of GE, e.g., Valentini and Dietterich (2004). Our DBI also differs from the stability-oriented measure of Bousquet and Elisseeff (2002), which was defined as the maximal difference of the decision functions trained from the original datasets and the leave-one-out datasets. More discussion of the connection with other variability measures is given in Section 3.3. In the end, extensive experiments illustrate the advantage of our selection algorithm over alternative approaches in terms of both classification accuracy and stability.

For simplicity, we focus on linear classifiers. The nonlinear extension is conceptually feasible by mapping the nonlinear feature space into a higher dimensional linear space; see the Appendix for further discussion. The rest of the article is organized as follows. Section 2 reviews the large-margin classifiers that are used as prototypical examples to illustrate our method. Section 3 describes the main properties of our classifier selection procedure. Section 4 establishes the selection consistency of the proposed selection procedure. Simulations and examples are in Section 5, followed by a brief discussion in Section 6. The Appendix and Supplementary Materials are devoted to technical details and a notation table.

2 Large-Margin Classifiers

This section briefly reviews the large-margin classifiers, that serve as prototypical examples to illustrate our two-stage classifier selection technique. The proposed method is broadly applicable to general classifiers.

Let $(\boldsymbol{X}, Y) \in \mathbb{R}^d \times \{1, -1\}$ be random variables from an underlying distribution $\mathcal{P}(\boldsymbol{X}, Y)$. Denote the conditional probability of class $Y = 1$ given $\boldsymbol{X} = \boldsymbol{x}$ as $p(\boldsymbol{x}) = P(Y = 1|\boldsymbol{X} = \boldsymbol{x})$, where $p(\boldsymbol{x}) \in (0, 1)$ to exclude the degenerate case. Let the input variable be $\boldsymbol{x} = (x_1, \ldots, x_d)^T$, $\tilde{\boldsymbol{x}} = (1, x_1, \ldots, x_d)^T$, with coefficient $\boldsymbol{w} = (w_1, \ldots, w_d)^T$ and parameter $\boldsymbol{\theta} = (b, \boldsymbol{w}^T)^T$. The linear decision function is defined as $f(\boldsymbol{x}; \boldsymbol{\theta}) = b + \boldsymbol{x}^T\boldsymbol{w} = \tilde{\boldsymbol{x}}^T\boldsymbol{\theta}$ with the decision boundary $S(\boldsymbol{x}; \boldsymbol{\theta}) = \{\boldsymbol{x} : f(\boldsymbol{x}; \boldsymbol{\theta}) = 0\}$. The performance of the classifier $\text{sign}\{f(\boldsymbol{x}; \boldsymbol{\theta})\}$ is measured by the classification risk $E[\mathbb{1}\{Y \neq \text{sign}\{f(\boldsymbol{X}; \boldsymbol{\theta})\}\}]$, where the expectation is with respect to $\mathcal{P}(\boldsymbol{X}, Y)$. Since the direct minimization of this risk is NP hard (Zhang (2004)), various convex surrogate loss functions $L(\cdot)$ have been proposed to deal with this computational issue. Denote the surrogate risk as $\mathcal{R}_L(\boldsymbol{\theta}) = E[L(Yf(\boldsymbol{X}; \boldsymbol{\theta}))]$, and assume that the minimizer of $\mathcal{R}_L(\boldsymbol{\theta})$ is obtained at $\boldsymbol{\theta}_{0L} = (b_{0L}, \boldsymbol{w}_{0L}^T)^T$. Here $\boldsymbol{\theta}_{0L}$ depends on the loss function L.

Given the training sample $\mathcal{D}_n = \{(\boldsymbol{x}_i, y_i); i = 1, \ldots, n\}$ drawn from $\mathcal{P}(\boldsymbol{X}, Y)$, a large-margin classifier minimizes the empirical risk

$$O_{nL}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} L\Big(y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)\Big) + \frac{\lambda_n}{2}\boldsymbol{w}^T\boldsymbol{w}, \tag{1}$$

where λ_n is some positive tuning parameter. The estimator minimizing $O_{nL}(\boldsymbol{\theta})$ is denoted by $\widehat{\boldsymbol{\theta}}_L = (\widehat{b}_L, \widehat{\boldsymbol{w}}_L^T)^T$. Common large-margin classifiers employ squared loss $L(u) = (1 - u)^2$, exponential loss $L(u) = e^{-u}$, logistic loss $L(u) = \log(1+e^{-u})$, and hinge loss $L(u) = (1-u)_+$. There seems to be no general guideline for selecting loss functions in practice, except for cross-validation error. Ideally, if we had access to an arbitrarily large test set, we would choose the classifier for which the test error was the smallest. In reality where only limited samples

Statistica Sinica: Preprint
doi:10.5705/ss.202016.0260

are available, cross-validation error may not be able to accurately approximate the testing error. Our goal is to establish a practically useful selection criterion by incorporating DBI with the cross-validation error.

3 Classifier Selection Algorithm

In this section, we propose a two-stage classifier selection algorithm that selects candidate classifiers whose estimated GEs are relatively small and deems the optimal classifier the one with the smallest DBI from those chosen.

3.1 Stage 1: Initial Screening via GE

We show that the difference of the cross-validation errors obtained from two large-margin classifiers is asymptotically Gaussian, which enables us to construct a confidence interval for their GE difference. We further propose a perturbation-based resampling approach to construct this confidence interval.

Given a new input $(\boldsymbol{X}_0, Y_0)$ from $\mathcal{P}(\boldsymbol{X}, Y)$, we define the GE induced by the loss function L as

$$D_{0L} = \frac{1}{2} E|Y_0 - \text{sign}\{f(\boldsymbol{X}_0; \widehat{\boldsymbol{\theta}}_L)\}|, \tag{2}$$

where $\widehat{\boldsymbol{\theta}}_L$ is based on the training sample $\mathcal{D}_n$, and the expectation is with respect to both $\mathcal{D}_n$ and $(\boldsymbol{X}_0, Y_0)$. The GE in (2) is equivalent to the mis-classification risk $E[\mathbb{1}\{Y_0 \neq \text{sign}\{f(\boldsymbol{X}_0; \widehat{\boldsymbol{\theta}}_L)\}\}]$. In practice, the GE, which depends on the underlying distribution $\mathcal{P}(\boldsymbol{X}, Y)$, needs to be estimated using $\mathcal{D}_n$. The empirical generalization error $\widehat{D}_L \equiv \widehat{D}(\widehat{\boldsymbol{\theta}}_L)$ with $\widehat{D}(\boldsymbol{\theta}) = (2n)^{-1} \sum_{i=1}^n |y_i - \text{sign}\{f(\boldsymbol{x}_i; \boldsymbol{\theta})\}|$ as an estimate suffers from the problem of overfitting (Wang and Shen (2006)). We use the K-fold cross-validation procedure to estimate the GE; this can significantly reduce the bias (Jiang et al. (2008)). We randomly split $\mathcal{D}_n$ into K disjoint subgroups and denote the kth subgroup as I_k. For $k = 1, \ldots, K$, we obtain the estimator $\widehat{\boldsymbol{\theta}}_{L(-k)}$ from all the data except those in I_k, and calculate the empirical average

$\widehat{D}(\widehat{\boldsymbol{\theta}}_{L(-k)}) = (2|I_k|)^{-1}\sum_{i\in I_k}|y_i - \text{sign}\{f(\boldsymbol{x}_i;\widehat{\boldsymbol{\theta}}_{L(-k)})\}|$ with $|I_k|$ the cardinality of I_k. The K-fold cross-validation (K-CV) error is thus computed as

$$\widehat{\mathcal{D}}_L = K^{-1}\sum_{k=1}^{K}\widehat{D}(\widehat{\boldsymbol{\theta}}_{L(-k)}). \tag{3}$$

We took $K = 5$ for our numerical experiments.

To establish the asymptotic normality of the K-CV error $\widehat{\mathcal{D}}_L$ for a general loss $L(\cdot)$, we require certain regularity conditions.

(L1) The probability distribution function of $\boldsymbol{X}$ and the conditional probability $p(\boldsymbol{x})$ are continuously differentiable.

(L2) The parameter $\boldsymbol{\theta}_{0L}$ is bounded and unique.

(L3) The map $\boldsymbol{\theta} \mapsto L(yf(\boldsymbol{x};\boldsymbol{\theta}))$ is convex.

(L4) The map $\boldsymbol{\theta} \mapsto L(yf(\boldsymbol{x};\boldsymbol{\theta}))$ is differentiable at $\boldsymbol{\theta} = \boldsymbol{\theta}_{0L}$ a.s., and $G(\boldsymbol{\theta}_{0L})$ is element-wisely bounded, where

$$G(\boldsymbol{\theta}_{0L}) = E\Big[\nabla_{\boldsymbol{\theta}}L(Yf(\boldsymbol{X};\boldsymbol{\theta}))\nabla_{\boldsymbol{\theta}}L(Yf(\boldsymbol{X};\boldsymbol{\theta}))^T\Big]\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{0L}}.$$

(L5) The surrogate risk $\mathcal{R}_L(\boldsymbol{\theta})$ is bounded and twice differentiable at $\boldsymbol{\theta} = \boldsymbol{\theta}_{0L}$ with positive definite Hessian matrix $H(\boldsymbol{\theta}_{0L}) = \nabla^2_{\boldsymbol{\theta}}\mathcal{R}_L(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{0L}}$.

Assumption (L1) ensures that the uniform law of large numbers can be applied. Assumption (L3) ensures that the uniform convergence theorem for convex functions (Pollard (1991)) can be applied; it is satisfied by all the large-margin loss functions considered in this paper. Assumptions (L4) and (L5) are required to obtain the local quadratic approximation to the surrogate risk function around $\boldsymbol{\theta}_{0L}$. Assumptions (L2)–(L5) were previously used by Rocha et al. (2009) to prove the asymptotic normality of $\widehat{\boldsymbol{\theta}}_L$.

Our result establishes the asymptotic normality of $\widehat{\mathcal{D}}_L$ for any large-margin classifier, generalizing the result for the SVM in Jiang et al. (2008).

Theorem 1 *If (L1)–(L5) hold and $\lambda_n = o(n^{-1/2})$, for any fixed K,*

$$\mathcal{W}_L = \sqrt{n}\Big(\widehat{\mathcal{D}}_L - D_{0L}\Big) \xrightarrow{d} N\Big(0, E(\psi_1^2)\Big) \quad \textit{as } n \to \infty, \tag{4}$$

where $\psi_1 = \frac{1}{2}|Y_1 - sign\{f(\boldsymbol{X}_1;\boldsymbol{\theta}_{0L})\}| - D_{0L} - \dot{d}(\boldsymbol{\theta}_{0L})^T H(\boldsymbol{\theta}_{0L})^{-1} M_1(\boldsymbol{\theta}_{0L})$ with $\dot{d}(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} E(\widehat{D}(\boldsymbol{\theta}))$, and $M_1(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} L(Y_1 f(\boldsymbol{X}_1;\boldsymbol{\theta}))$.

The proof of Theorem 1 is in Section S.1 of the online supplement. An immediate application compares competing loss functions L_1 and L_2. Take their GE difference Δ_{12} and its consistent estimate $\widehat{\Delta}_{12}$ to be $D_{02} - D_{01}$ and $\widehat{\mathcal{D}}_2 - \widehat{\mathcal{D}}_1$, respectively. To test whether the GEs induced by L_1 and L_2 are significantly different, we need to establish an approximate confidence interval for Δ_{12} based on the distribution of $\mathcal{W}_{\Delta_{12}} \equiv \mathcal{W}_2 - \mathcal{W}_1 = n^{1/2}(\widehat{\Delta}_{12} - \Delta_{12})$. We apply the perturbation-based resampling procedure of Park and Wei (2003) to approximate the distribution of $\mathcal{W}_{\Delta_{12}}$, this in common with Jiang et al. (2008) who employed it to construct the confidence interval of SVM's GE. Specifically, let $\{G_i\}_{i=1}^n$ be i.i.d. random variables drawn from the exponential distribution with unit mean and unit variance, and let

$$\widehat{\boldsymbol{\theta}}_j^* = \arg\min_{b,\boldsymbol{w}} \left\{ \frac{1}{n}\sum_{i=1}^n G_i L_j\Big(y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)\Big) + \frac{\lambda_n}{2}\boldsymbol{w}^T\boldsymbol{w} \right\}. \tag{5}$$

Conditionally on $\mathcal{D}_n$, the randomness of $\widehat{\boldsymbol{\theta}}_j^*$ merely comes from that of $G_1, \ldots, G_n$. Take $W_{\Delta_{12}}^* = W_2^* - W_1^*$, with

$$W_j^* = n^{-1/2}\sum_{i=1}^n \left\{ \frac{1}{2}\Big| y_i - \text{sign}\{f(\boldsymbol{x}_i, \widehat{\boldsymbol{\theta}}_j^*)\}\Big| - \widehat{D}_j \right\} G_i. \tag{6}$$

By repeatedly generating a set of random variables $\{G_i, i = 1, \ldots, n\}$, we can obtain a large number of realizations of $W_{\Delta_{12}}^*$ to approximate the distribution of $\mathcal{W}_{\Delta_{12}}$. The proof of the following is in the online supplement.

Theorem 2 *Suppose that the assumptions in Theorem 1 hold. Then as $n \to \infty$,*

$$\mathcal{W}_{\Delta_{12}} \xrightarrow{d} N\Big(0, Var(\psi_{12} - \psi_{11})\Big),$$

where ψ_{11} and ψ_{12} are defined in Section S.2 of the online supplement, and

$$W^*_{\Delta_{12}} \overset{d}{\Longrightarrow} N\Big(0, Var(\psi_{12} - \psi_{11})\Big) \quad \textit{conditional on } \mathcal{D}_n,$$

where "$\Longrightarrow$" means conditional weak convergence in the sense of Hoffmann-Jorgensen (1984).

Our algorithm summarizes the resampling procedure for establishing the confidence interval of the GE difference Δ_{12}.

Algorithm 1 (Generalization Error Comparison Algorithm)

Input: Training sample $\mathcal{D}_n$ and candidate loss functions L_1 and L_2.

```
Step 1. Calculate K-CV errors $\widehat{\mathcal{D}}_1$ and $\widehat{\mathcal{D}}_2$ induced from $L_1$ and $L_2$, respectively.

Step 2. For $r = 1, \ldots, N$, repeat the following steps:

   (a) generate i.i.d. samples $\{G_i^{(r)}\}_{i=1}^n$ from Exp(1);

   (b) find $\widehat{\boldsymbol{\theta}}_j^{*(r)}$ via (5), $W_j^{*(r)}$ via (6), and calculate $W^{*(r)}_{\Delta_{12}} = W_2^{*(r)} - W_1^{*(r)}$.

Step 3. Construct the 100(1 − $\alpha$)% confidence interval for $\Delta_{12}$ as

   $$\Big[\widehat{\Delta}_{12} - n^{-1/2}\phi_{1,2;\alpha/2}, \widehat{\Delta}_{12} - n^{-1/2}\phi_{1,2;1-\alpha/2}\Big],$$

   where $\widehat{\Delta}_{12} = \widehat{\mathcal{D}}_2 - \widehat{\mathcal{D}}_1$ and $\phi_{1,2;\alpha}$ is the $\alpha$th upper percentile of $\{W^{*(1)}_{\Delta_{12}}, \ldots, W^{*(N)}_{\Delta_{12}}\}$.
```

In our experiments, we repeated the resampling procedure $N = 100$ times in Step 2, and fix $\alpha = 0.1$. The effect of the choice of α is discussed at the end of Section 3.4. The GEs of two classifiers induced from L_1 and L_2 are judged as significantly different if the confidence interval established in Step 3 does not contain 0. We apply *Algorithm 1* to eliminate the classifiers whose GEs are significantly different from the minimal GE of a set of candidate classifiers.

Employing statistical testing for classifier comparison has been successfully applied in practice (Dietterich (1998); Demsar (2006)). In particular, Demsar (2006) reviewed several statistical tests in comparing two classifiers on multiple data sets and recommended the Wilcoxon sign rank test, which examined whether two classifiers were significantly different by calculating the relative rank of their corresponding performance scores on multiple data sets. Compared to the Wilcoxon sign rank test, our perturbed cross-validation estimator has the advantage of being theoretically justified without assuming measured performance scores have no sampling error.

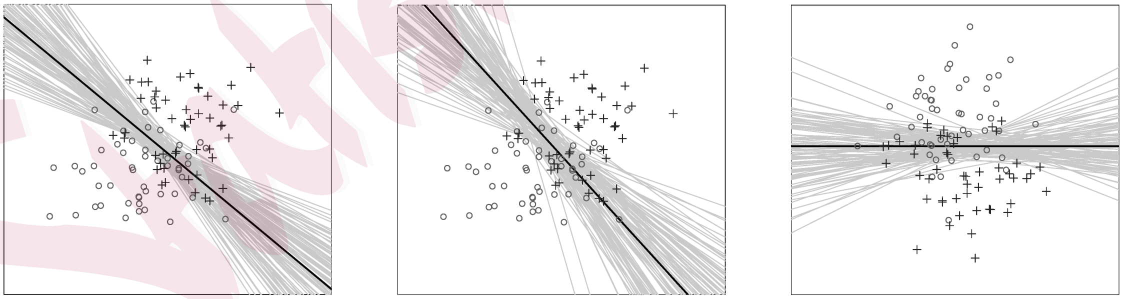
The classifiers that emerge from *Algorithm 1* are potentially good. However, their decision boundaries may change dramatically following small perturbations of the training sample, indicating prediction instability. We introduce the DBI to capture the prediction instability, and embed it into our classifier selection algorithm.

3.2 Stage 2: Final Selection via DBI

In this section, we define the DBI and then provide an efficient way to estimate it in practice.

Example: To motivate the DBI, we start with a simulated example using two classifiers: squared loss L_1 and hinge loss L_2. We generated 100 observations from a mixture of two Gaussian distributions with equal probability: $N((-0.5,-0.5)^T, I_2)$ and $N((0.5,0.5)^T, I_2)$ with I_2 an identity matrix of dimension two. In Figure 1, we plot the decision boundary $S(\boldsymbol{x};\widehat{\boldsymbol{\theta}}_j)$ based on $\mathcal{D}_n$, and 100 perturbed decision boundaries $\{S(\boldsymbol{x};\widehat{\boldsymbol{\theta}}_j^{*(1)}),\ldots,S(\boldsymbol{x};\widehat{\boldsymbol{\theta}}_j^{*(100)})\}$ for $j=1,2$; see Step 2 of *Algorithm 1*. Figure 1 reveals that the perturbed decision boundaries of the squared loss are more stable than those of the SVM given a small perturbation of the training sample. To quantify the variability of the perturbed decision boundaries with respect to the original unperturbed decision boundary $S(\boldsymbol{x};\widehat{\boldsymbol{\theta}}_j)$ is a nontrivial task since the boundaries spread over a d-dimensional space, $d=2$ in Figure 1. We transform the data in such a way that the variability can be fully measured in a single dimension. We find a $d \times d$ transformation matrix R_L, orthogonal with determinant 1, such that the decision boundary based on the transformed data $\mathcal{D}_n^\dagger = \{(\boldsymbol{x}_i^\dagger, y_i), i = 1, \ldots, n\}$ with $\boldsymbol{x}_i^\dagger = R_L \boldsymbol{x}_i$ is parallel to the $\mathcal{X}_1, \ldots, \mathcal{X}_{d-1}$ axes; see the supplementary material S.3 for the calculation of R_L. The variability of the perturbed decision boundaries with respect to the original unperturbed decision boundary then reduces to the variability along the last axis $\mathcal{X}_d$. To illustrate, we apply the data-transformation idea to the SVM plotted in the middle plot of Figure 1. From the right plot in Figure 1, the variability of the transformed perturbed decision boundaries (in gray) with respect to the transformed unperturbed decision boundary (in black) reduces to the variability along the $\mathcal{X}_2$ axis only; the transformed unperturbed decision boundary is parallel to the $\mathcal{X}_1$ axis. The choice of data transformation is not unique as, for example, we could interchange the role of the two axes; the DBI measure we introduce is transformation invariant.

Figure 1: Two classes are shown in circles and crosses. The black line is the decision boundary based on the original training sample, and the gray lines are 100 decision boundaries based on perturbed samples. The left (middle) panel corresponds to the least square loss (SVM). The perturbed decision boundaries of SVM after data transformation are shown on the right.



Given the loss function L, we define the coefficient estimator based on transformed data $\mathcal{D}_n^\dagger$ as $\widehat{\boldsymbol{\theta}}_L^\dagger$ and the coefficient estimator of its corresponding perturbed decision boundary as

$\widehat{\boldsymbol{\theta}}_L^{\dagger *}$. We find the following relationship via the transformation matrix R_L:

$$\widehat{\boldsymbol{\theta}}_L \equiv \begin{pmatrix} \widehat{b}_L \\ \widehat{\boldsymbol{w}}_L \end{pmatrix} \Rightarrow \widehat{\boldsymbol{\theta}}_L^{\dagger} \equiv \begin{pmatrix} \widehat{b}_L \\ R_L \widehat{\boldsymbol{w}}_L \end{pmatrix} \text{ and } \widehat{\boldsymbol{\theta}}_L^{*} \equiv \begin{pmatrix} \widehat{b}_L^{*} \\ \widehat{\boldsymbol{w}}_L^{*} \end{pmatrix} \Rightarrow \widehat{\boldsymbol{\theta}}_L^{\dagger *} \equiv \begin{pmatrix} \widehat{b}_L^{*} \\ R_L \widehat{\boldsymbol{w}}_L^{*} \end{pmatrix}.$$

This can be shown by replacing $\boldsymbol{x}_i$ with $R_L \boldsymbol{x}_i$ in (1) and (5) and using the property of R_L. Given $\widehat{\boldsymbol{\theta}}_L^{\dagger *} = (\widehat{b}_L^{*}, \widehat{w}_{L,1}^{\dagger *}, \ldots, \widehat{w}_{L,d}^{\dagger *})^T$, we define the d-th dimension of $S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_L^{\dagger *})$ as

$$S_d := -\frac{\widehat{b}_L^{\dagger *}}{\widehat{w}_{L,d}^{\dagger *}} - \sum_{j=1}^{d-1} \frac{\widehat{w}_{L,j}^{\dagger *}}{\widehat{w}_{L,d}^{\dagger *}} X_j. \tag{7}$$

DBI is then the variability of the transformed perturbed decision boundary $S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_L^{\dagger *})$ with respect to the transformed unperturbed decision boundary $S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_L^{\dagger})$ along its d-th dimension.

Definition 1 *The decision boundary instability (DBI) of $S(\boldsymbol{x}; \widehat{\boldsymbol{\theta}}_L)$ is*

$$DBI\Big(S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_L)\Big) = E\left[Var\Big(S_d | \boldsymbol{X}_{(-d)}^{\dagger}\Big)\right], \tag{8}$$

where S_d is defined in (7) *and $\boldsymbol{X}_{(-d)}^{\dagger} = (X_1^{\dagger}, \ldots, X_{d-1}^{\dagger})^T$.*

Remark 1 The conditional variance $Var(S_d | \boldsymbol{X}_{(-d)}^{\dagger})$ in (8) captures the variability of the transformed perturbed decision boundary along the dth dimension based on a given sample. After data transformation, the transformed unperturbed decision boundary is parallel to the $\mathcal{X}_1, \ldots, \mathcal{X}_{d-1}$ axes. This conditional variance precisely measures the variability of the perturbed decision boundary with respect to the unperturbed decision boundary conditioned on the given sample. The expectation in (8) then averages out the randomness in the sample.

Example Continuation: We give an illustration of (8) via the 2-dimensional example shown in the right plot of Figure 1. For each sample, the conditional variance in (8) was estimated via the sample variability of the projected X_2 values on the perturbed decision boundary. Then the final DBI was estimated by averaging over all samples.

In Appendix A.1, we demonstrate an efficient way to simplify (8) by approximating the conditional variance via the weighted variance of $\widehat{\boldsymbol{\theta}}^\dagger_L$. The idea is to connect the conditional variance of the d-th dimension of decision boundary with the variance of the coefficients of the corresponding decision function. We show that

$$DBI\Big(S(\boldsymbol{X};\widehat{\boldsymbol{\theta}}_L)\Big) \approx (w^\dagger_{L,d})^{-2} E\left[\tilde{\boldsymbol{X}}^{\dagger T}_{(-d)}\left(n^{-1}\Sigma^\dagger_{0L,(-d)}\right)\tilde{\boldsymbol{X}}^\dagger_{(-d)}\right], \tag{9}$$

where $w^\dagger_{L,d}$ is the last entry of the transformed coefficient $\boldsymbol{\theta}^\dagger_{0L}$, and $n^{-1}\Sigma^\dagger_{0L,(-d)}$ is the asymptotic variance of the first d dimensions of $\widehat{\boldsymbol{\theta}}^\dagger_L$. Therefore, DBI can be viewed as a proxy measure of the asymptotic variance of the decision function.

We propose a plug-in estimate for the approximate version of DBI in (9). Direct estimation of DBI in (8) is possible, but it requires perturbing the transformed data. To reduce the computational cost, we can take advantage of the resampling results in Stage 1 based on the relationship between $\Sigma^\dagger_{0L}$ and Σ_{0L}. We can estimate $\Sigma^\dagger_{0L}$ by

$$\widehat{\Sigma}^\dagger_L = \begin{pmatrix} \widehat{\Sigma}_b & \widehat{\Sigma}_{b,\boldsymbol{w}} R^T_L \\ R_L \widehat{\Sigma}_{\boldsymbol{w},b} & R_L \widehat{\Sigma}_{\boldsymbol{w}} R^T_L \end{pmatrix} \text{ given that } \widehat{\Sigma}_L = \begin{pmatrix} \widehat{\Sigma}_b & \widehat{\Sigma}_{b,\boldsymbol{w}} \\ \widehat{\Sigma}_{\boldsymbol{w},b} & \widehat{\Sigma}_{\boldsymbol{w}} \end{pmatrix}, \tag{10}$$

where $\widehat{\Sigma}_L$ is the sample variance of $\widehat{\boldsymbol{\theta}}^*_L$ obtained from Stage 1 as a byproduct. Hence, combining (9) and (10), we propose the estimator

$$\widehat{DBI}\Big(S(\boldsymbol{X};\widehat{\boldsymbol{\theta}}_L)\Big) = \frac{\sum_{i=1}^n \widetilde{\boldsymbol{x}}_i{}^{\dagger T}_{(-d)} \widehat{\Sigma}^\dagger_{L,(-d)} \widetilde{\boldsymbol{x}}_i{}^{\dagger}_{(-d)}}{(n\widehat{w}^\dagger_{L,d})^2}, \tag{11}$$

where $\widehat{w}^\dagger_{L,d}$ is the last entry of $\widehat{\boldsymbol{\theta}}^\dagger_L$, and $\widehat{\Sigma}^\dagger_{L,(-d)}$ is obtained by removing the last row and last column of $\widehat{\Sigma}^\dagger_L$ defined in (10). The DBI estimate in (11) is the one we use in numerical experiments.

Statistica Sinica: Preprint
doi:10.5705/ss.202016.0260

3.3 Relationship of DBI with Other Variability Measures

DBI may appear to be related to the asymptotic variance $E(\psi_1)^2$ in Theorem 1. But these two quantities are quite different. When data are nearly separable, reasonable perturbations to the data may only lead to a small variation in the K-CV error, while small changes in the data (especially those support points near the decision boundary) may lead to a large variation in the decision boundary which implies a large DBI. In Section 5, we provide examples to show that these variation measures generally lead to different choices of loss functions, and the loss function with the smallest DBI often corresponds to the classifier that is more accurate and stable.

While the stability-oriented measure of Bousquet and Elisseeff (2002) shares a similar spirit as our DBI, they focus on the variability of the decision function as opposed to the decision boundary. Their procedure is not transformation invariant while ours is.

In the experiments, we compare our classifier selection algorithm with approaches using these two alternatives. Our method achieves superior performance in classification accuracy and stability.

3.4 Summary of Classifier Selection Algorithm

Algorithm 2 (Two-Stage Classifier Selection Procedure):

Input: Training sample $\mathcal{D}_n$ and a collection of candidate loss functions $\{L_j : j \in J\}$.

Step 1. Obtain the K-CV errors $\widehat{\mathcal{D}}_j$ for each $j \in J$, with minimal value $\widehat{\mathcal{D}}_t$.

Step 2. Apply *Algorithm 1* to establish the pairwise confidence interval for each GE difference Δ_{tj}. Eliminate the loss L_j if the corresponding confidence interval does not cover zero. The set of potentially good classifiers is

$$\Lambda = \Big\{ j \in J : \widehat{\Delta}_{tj} - n^{-1/2}\phi_{t,j;\alpha/2} \leq 0 \Big\},$$

where $\widehat{\Delta}_{tj}$ and $\phi_{t,j;\alpha/2}$ are defined in Step 3 of *Algorithm 1.*

Step 3. Estimate DBI for each L_j, $j \in \Lambda$ via (11). The optimal loss function is L_{j^*} with

$$j^* = \arg\min_{j \in \Lambda} \widehat{DBI}\Big(S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_j)\Big). \tag{12}$$

In Step 2, we fix the confidence level $\alpha = 0.1$ since it provides a sufficient but not too stringent confidence level. Our experiment in Section 6.1 further shows that the set Λ is quite stable against α within a reasonable range around 0.1. The optimal loss function L_{j^*} selected in (12) is not necessarily unique. However, according to our experiments, multiple optimal loss functions are quite uncommon. In principle we can perform an additional significance test for DBI in Step 3, but the related computational cost is high given that DBI is already a second-moment measure. We choose not to include this test in our algorithm.

4 Selection Consistency

This section investigates the selection consistency of our algorithm by showing that the selected classifier achieves the minimal GE and minimal DBI asymptotically. To simplify the presentation, we establish our selection consistency via the large-margin unified machines (LUM, Liu et al. (2011)); the extension to other large-margin classifiers is straightforward.

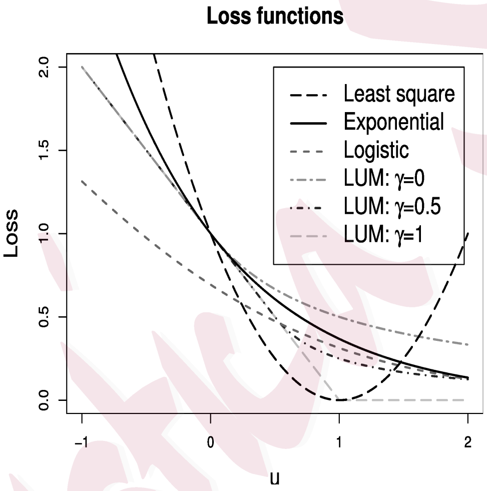
The LUM offers a platform unifying various large margin classifiers ranging from soft to hard ones. A soft classifier estimates the class conditional probabilities explicitly and makes the class prediction via the largest estimated probability, while a hard classifier directly estimates the classification boundary without a class-probability estimation (Wahba (2002)). The class of LUM loss functions can be written as

$$L_\gamma(u) = \begin{cases} 1 - u & \text{if } u < \gamma \\ (1-\gamma)^2(\frac{1}{u-2\gamma+1}) & \text{if } u \geq \gamma, \end{cases} \tag{13}$$

Statistica Sinica: Preprint
doi:10.5705/ss.202016.0260

where the index parameter $\gamma \in [0, 1]$. As shown by Liu et al. (2011), when $\gamma = 1$ the LUM loss reduces to the hinge loss of SVM, which is a typical example of hard classification; when $\gamma = 0.5$ the LUM loss is equivalent to the DWD classifier, which can be viewed as a classifier that is between hard and soft; and when $\gamma = 0$ the LUM loss is a soft classifier that has an interesting connection with the logistic loss. Therefore, the LUM framework approximates many of the soft and hard classifiers in the literature. Figure 2 displays LUM loss functions for various values of γ and compares them with some commonly used loss functions.

Figure 2: Plots of least square, exponential, logistic, and LUM loss functions with $\gamma = 0, 0.5, 1$.



In the LUM framework, we denote the true risk as $\mathcal{R}_\gamma(\boldsymbol{\theta}) = E[L_\gamma(yf(\boldsymbol{x};\boldsymbol{\theta}))]$, the true parameter as $\boldsymbol{\theta}_{0\gamma} = \arg\min_{\boldsymbol{\theta}} \mathcal{R}_\gamma(\boldsymbol{\theta})$, the GE as $D_{0\gamma}$, the empirical generalization error as $\widehat{D}_\gamma$, and the K-CV error as $\widehat{\mathcal{D}}_\gamma$. Given data $\mathcal{D}_n$, LUM solves

$$\widehat{\boldsymbol{\theta}}_\gamma = \arg\min_{b,\boldsymbol{w}} \left\{ \frac{1}{n} \sum_{i=1}^{n} L_\gamma\Big(y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)\Big) + \frac{\lambda_n \boldsymbol{w}^T\boldsymbol{w}}{2} \right\}. \tag{14}$$

In Corollaries 1 and 2 provided in Section S.4 of the online supplement, we establish the asymptotic normality of $\widehat{\boldsymbol{\theta}}_\gamma$ and $\widehat{\mathcal{D}}_\gamma$, respectively. These preliminary results are used to develop the selection consistency of our two-stage classifier selection algorithm.

For the LUM class, the set of potentially good classifiers is

$$\widehat{\Lambda}_0 = \Big\{\gamma \in [0,1] : \widehat{\mathcal{D}}_\gamma \le \widehat{\mathcal{D}}_{\widehat{\gamma}_0^*} + n^{-1/2}\phi_{\gamma,\widehat{\gamma}_0^*;\alpha/2}\Big\}, \tag{15}$$

where $\widehat{\gamma}_0^* = \arg\min_{\gamma\in[0,1]} \widehat{\mathcal{D}}_\gamma$, based on $\mathcal{D}_n$. Its population version is defined as those classifiers achieving the minimal GE, denoted

$$\Lambda_0 = \Big\{\gamma \in [0,1] : D_{0\gamma} = D_{0\gamma_0^*}\Big\}, \tag{16}$$

where $\gamma_0^* = \arg\min_{\gamma\in[0,1]} D_{0\gamma}$. To show the selection consistency, we require an additional assumption on the Hessian matrix $H(\boldsymbol{\theta}_{0\gamma})$ defined in Corollary 1, see the online supplement.

(B1) The smallest eigenvalue of the true Hessian matrix $\lambda_{\min}(H(\boldsymbol{\theta}_{0\gamma})) \ge c_1$, and the largest eigenvalue of the true Hessian matrix $\lambda_{\max}(H(\boldsymbol{\theta}_{0\gamma})) \le c_2$, where the positive constants c_1, c_2 do not depend on γ.

As seen in the proof of Corollary 1, the true Hessian matrix $H(\boldsymbol{\theta}_{0\gamma})$ is positive definite for any fixed $\gamma \in [0,1]$. Therefore, Assumption (B1) is slightly stronger in the uniform sense. It is required to guarantee the uniform convergence results, ($S.15$) and ($S.17$), in Section S.7 of the online supplement.

Lemma 1 *If (L1), (B1), and (A1) in the online supplement hold, for* $\lambda_n = o(n^{-1/2})$,

$$\Big|\widehat{\mathcal{D}}_{\widehat{\gamma}_0^*} - D_{0\gamma_0^*}\Big| = O_P(n^{-1/2}). \tag{17}$$

In the second stage, we denote the index of the selected optimal classifier as

$$\widehat{\gamma}_0 = \arg\min_{\gamma\in\widehat{\Lambda}_0} \widehat{DBI}\Big(S(\boldsymbol{X};\widehat{\boldsymbol{\theta}}_\gamma)\Big), \tag{18}$$

and its population version as

$$\gamma_0 = \arg\min_{\gamma \in \Lambda_0} DBI\Big(S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_\gamma)\Big). \tag{19}$$

Theorem 3 *If the assumptions in Lemma 1 hold, as $N \to \infty$,*

$$\left|\widehat{DBI}\Big(S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_{\widehat{\gamma}_0})\Big) - DBI\Big(S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_{\gamma_0})\Big)\right| = o_P(n^{-1}), \tag{20}$$

where N is the number of resamplings in Step 2 of Algorithm 1.

Theorem 3 implies that the estimated DBI of the selected classifier converges to the DBI of the true optimal classifier, that has the smallest DBI. Therefore, the proposed two-stage algorithm is able to select the classifier with the minimal DBI among those classifiers having the minimal GE. In summary, we have shown that the selected optimal classifier has achieved the minimal GE and the minimal DBI asymptotically.

5 Experiments

In this section, we first demonstrate the DBI estimation procedure introduced in Section 3.2, and then illustrate the applicability of our classifier selection method in various examples. In all experiments, we compared our selection procedure, denoted as "cv+dbi", with two alternative methods: "cv+varcv", the two-stage approach selecting the loss with the minimal variance of the K-CV error in Stage 2; "cv+be", the two-stage approach selecting the loss with the minimal classification stability, as in Bousquet and Elisseeff (2002), in Stage 2. Stage 1 of each alternative approach is the same as ours. We consider six large-margin classifier candidates: least squares loss, exponential loss, logistic loss, and LUM with $\gamma = 0, 0.5, 1$. In all the large-margin classifiers, the tuning parameter λ_n was selected via cross-validation.

Statistica Sinica: Preprint
doi:10.5705/ss.202016.0260

5.1 Illustration

This subsection demonstrates the DBI estimation procedure and checks the sensitivity of the confidence level α in Algorithm 2.

We generated labels $y \in \{-1, 1\}$ with equal probability. Given $Y = y$, the predictor vector (x_1, x_2) was generated from a bivariate normal $N((\mu y, \mu y)^T, I_2)$ with the signal level $\mu = 0.8$.

We first compared the estimated DBIs with the true DBIs for various sample sizes. We varied the sample size n as 50, 100, 200, 500, and 1000. The classifier with the least squares loss was investigated due to its simplicity. Simple algebra gives the true parameter $\boldsymbol{\theta}_{0L} = (0, 0.351, 0.351)$ and the transformed parameter $\boldsymbol{\theta}^{\dagger}_{0L} = (0, 0, 0.429)$. The covariance matrix Σ_{0L} and the transformed covariance matrix $\Sigma^{\dagger}_{0L}$ were computed as

$$\Sigma_{0L} = \begin{pmatrix} 0.439 & 0 & 0 \\ 0 & 0.268 & -0.170 \\ 0 & -0.170 & 0.268 \end{pmatrix} \text{ and } \Sigma^{\dagger}_{0L} = \begin{pmatrix} 0.439 & 0 & 0 \\ 0 & 0.439 & 0 \\ 0 & 0 & 0.098 \end{pmatrix},$$

given the transformation matrix

$$R_L = \begin{pmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{pmatrix}.$$

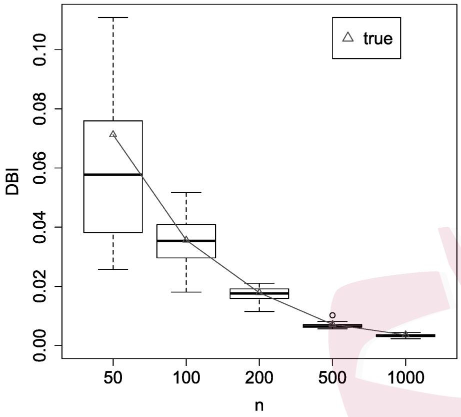
Plugging these terms into (9) led to

$$DBI\Big(S(\boldsymbol{X}; \widehat{\boldsymbol{\theta}}_L)\Big) \approx \frac{3.563}{n}. \tag{21}$$

Figure 3 compares the estimated DBIs in (11) with the true DBIs in (21). They match well for various sample sizes and their difference vanishes as the sample size increases.

To show the sensitivity of the confidence level α to the set Λ in Algorithm 2, we randomly selected one replication and found the proportion of potentially good classifiers over all six

Figure 3: Comparison of true and estimated DBIs in Example 6.1. The true DBI for each n is denoted as a triangle and the estimated DBIs from replicated experiments are illustrated by box plots.



classifiers. As α increases, the confidence interval for the difference of GEs narrows, and hence the size of Λ will be smaller. The change of the proportion reflects exactly the change of Λ since Λ is monotone with respect to α. For each $\alpha \in \{l/100;\ l = 0, \ldots, 50\}$, we computed the proportion of potentially good classifiers and observed that the proportion was stable in a reasonable large range around 0.1.

5.2 Simulations

In this section, we recount the performance of our method using four simulated examples. These simulations were previously studied by Liu et al. (2011). In each simulation, the size of training data sets was 100 and that of testing data sets was 1000. All procedures were repeated 100 times and the averaged test errors and averaged test DBIs of the selected classifier were reported.

Simulation 1: Two predictors were uniformly generated over $\{(x_1, x_2) : x_1^2 + x_2^2 \leq 1\}$. The class label y was 1 when $x_2 \geq 0$ and -1 otherwise. We generated 100 samples and then contaminated the data by randomly flipping the labels of 15% of the instances.

Simulation 2: The setting of Simulation 1 except that we contaminated the data by randomly flipping the labels of 25% of the instances.

Simulation 3: The setting of Simulation 1 except that we contaminated the data by randomly flipping the labels of 80% of the instances whose $|x_2| \geq 0.7$.

Simulation 4: Two predictors were uniformly generated over $\{(x_1, x_2) : |x_1| + |x_2| \leq 2\}$. Conditionally on $X_1 = x_1$ and $X_2 = x_2$, the class label y took 1 with probability $e^{3(x_1+x_2)}/(1 + e^{3(x_1+x_2)})$ and -1 otherwise.

We demonstrate the mechanism of our proposed method for one repetition of Simulation 1. As shown in the upper left plot of Figure 4, exponential loss and LUMs with $\gamma = 0.5$ or 1 are potentially good classifiers in Stage 1; they happen to have the same K-CV error. Their corresponding DBIs are compared in the second stage. As shown in the upper right plot of Figure 4, LUM with $\gamma = 0.5$ gives the minimal DBI and is selected as the final classifier. In this example, while exponential loss gives the minimal K-CV error, its decision boundary is unstable compared to that of LUM with $\gamma = 0.5$. To show that our DBI estimation is reasonable, we display the perturbed decision boundaries for these three potentially good classifiers on the bottom of Figure 4. The relationship among their instabilities is captured by our DBI estimate: compared with exponential loss and LUM with $\gamma = 1$, LUM with $\gamma = 0.5$ is more stable.

We report the averaged test errors and averaged test DBIs of the classifier selected from our method as well as two alternative approaches, see Table 1. In the four simulated examples, "cv+dbi" achieves the smallest test errors, while the difference among test errors of all algorithms is generally not significant. All methods are the same during the first stage and those left from Stage 1 are all potentially good in terms of classification accuracy, but "cv+dbi" is able to choose the classifiers with minimal test DBIs in all simulations and the improvements over other algorithms are significant. Overall, our method is able to choose the classifier with outstanding performance in both classification accuracy and stability.

Figure 4: The K-CV error, the DBI estimate, and the perturbed decision boundaries in Simulation 1 with flipping rate 15%. The minimal K-CV error and minimal DBI estimate are indicated with triangles. The labels Ls, Exp, Logit, LUM0, LUM0.5, and LUM1 refer to least squares loss, exponential loss, logistic loss, and LUM loss with index $\gamma = 0, 0.5, 1$, respectively.

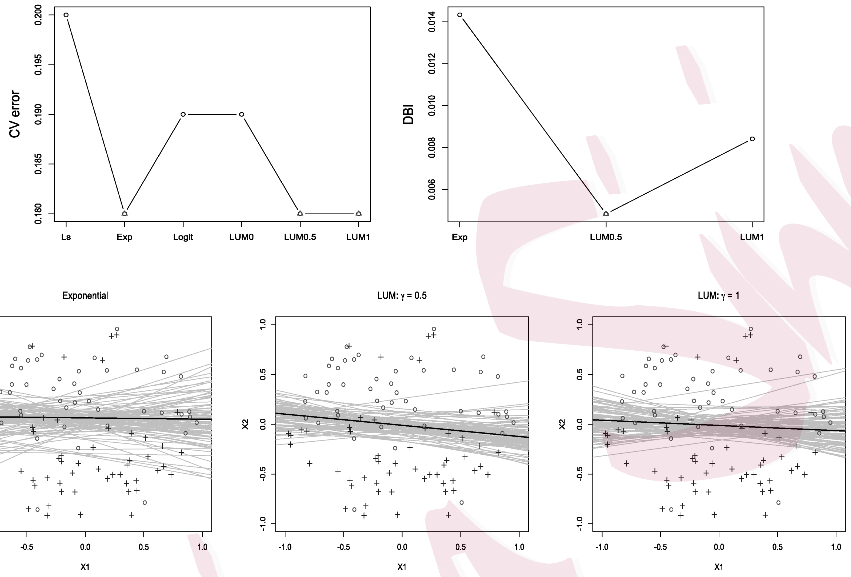


Table 1: The averaged test errors and averaged test DBIs (multiplied by 100) of all methods: "cv+varcv" is the two-stage approach which selects the loss with the minimal variance of the K-CV error in Stage 2; "cv+be" is the two-stage approach which in Stage 2 selects the loss with the minimal classification stability defined in Bousquet and Elisseeff (2002); "cv+dbi" is our method. The smallest value in each case is given in bold. Standard errors are given in subscript.

Simulations		cv+varcv	cv+be	cv+dbi
Sim 1	Error	$0.191_{0.002}$	$0.194_{0.002}$	$\mathbf{0.190}_{0.002}$
	DBI	$0.139_{0.043}$	$0.135_{0.019}$	$\mathbf{0.081}_{0.002}$
Sim 2	Error	$0.296_{0.002}$	$0.303_{0.003}$	$\mathbf{0.295}_{0.002}$
	DBI	$0.291_{0.044}$	$0.318_{0.036}$	$\mathbf{0.229}_{0.012}$
Sim 3	Error	$0.218_{0.006}$	$0.234_{0.006}$	$\mathbf{0.209}_{0.004}$
	DBI	$0.124_{0.008}$	$0.291_{0.037}$	$\mathbf{0.107}_{0.003}$
Sim 4	Error	$0.120_{0.001}$	$0.121_{0.001}$	$\mathbf{0.119}_{0.001}$
	DBI	$0.884_{0.207}$	$0.414_{0.106}$	$\mathbf{0.235}_{0.038}$

Statistica Sinica: Preprint
doi:10.5705/ss.202016.0260

5.3 Examples

In this subsection, we compare our method with the alternatives on three datasets in the UCI Machine Learning Repository (Frank and Asuncion (2010)).

The first data set is the liver disorders data set (*liver*) that consists of 345 samples with 6 variables of blood test measurements. The class label splits the data into two classes with sizes 145 and 200. The second data set is the breast cancer data set (*breast*) which consists of 683 samples after removing missing values (Wolberg and Mangasarian (1990)). Each sample has 10 experimental measurement variables and one binary class label indicating whether the sample is benign or malignant. These 683 samples arrived periodically as Dr. Wolberg reported his clinical cases. In total, there are 8 groups of samples which reflect the chronological order of the data. It is expected that a good classification procedure should generate a classifier that is stable across these groups of samples. The third data set is the credit approval data set (*credit*) which consists of 690 samples with 15 features, among which 307 samples have a positive class label and the rest have a negative class label.

For each dataset, we randomly split the data into 2/3 training samples and 1/3 testing samples, and reported the averaged test errors and averaged test DBIs based on all classifier selection algorithms over 50 replications, see Table 2. Compared with the alternatives, “cv+dbi” obtains significant improvements in DBIs and simultaneously attains satisfactory test errors that are minimal or statistically indistinguishable to the minimal one.

Table 2: The averaged test errors and averaged test DBIs of all methods in real example. The smallest value in each case is given in bold. Standard errors are given in subscript.

Data		cv+varcv	cv+be	cv+dbi
Liver	Error	$0.331_{0.006}$	$0.335_{0.006}$	$\mathbf{0.327}_{0.006}$
	DBI	$0.140_{0.013}$	$0.157_{0.024}$	$\mathbf{0.113}_{0.012}$
Breast	Error	$\mathbf{0.038}_{0.002}$	$\mathbf{0.038}_{0.002}$	$\mathbf{0.038}_{0.002}$
	DBI	$0.388_{0.066}$	$0.152_{0.028}$	$\mathbf{0.124}_{0.023}$
Credit	Error	$\mathbf{0.135}_{0.004}$	$0.138_{0.004}$	$0.136_{0.004}$
	DBI	$0.229_{0.101}$	$0.157_{0.042}$	$\mathbf{0.112}_{0.023}$

6 Discussion

This paper proposes a two-stage classifier selection procedure based on GE and DBI. It selects the classifier with the most stable decision boundary among those classifiers with relatively small estimated GEs. The concept of DBI is quite general, and its extension to a broader framework, e.g., multi-category classification (Shen and Wang (2007); Zhang and Liu (2013)) or high-dimensional classification (Fan et al. (2012)), is conceptually simple. In particular, in the multi-category classification, we suggest using the one-versus-all idea (Rifkin and Klautau (2004)) to extend our DBI measure. For K classes, we compute DBI_k as the DBI between the k-th class and the other $K-1$ classes, then average the DBIs to obtain the final DBI as $K^{-1}\sum_{k=1}^{K}\mathrm{DBI}_k$. When $K=2$, this reduces to our original DBI.

The extension to the nonlinear classifiers is also feasible. We give detailed discussions of the nonlinear extension in Appendix A.2. Briefly, in Stage 1, the asymptotic normality of the nonlinear K-CV error remains valid due to Hable (2012); in Stage 2, measuring the instability of the nonlinear decision boundaries is possible by mapping the nonlinear decision boundaries to a higher-dimensional space where the projected decision boundaries are linear.

Supplementary Materials

In the online supplement, we provide all proofs, discuss the calculation of the transformation matrix, and provide a notation table.

Acknowledgment

The authors thank the Co-Editor, an associate editor, and two referees for their constructive comments and suggestions, which have led to a significantly improved paper. We thank SAMSI for hospitality when part of the work was being done. Will Wei Sun thanks Yongheng Zhang for discussions on the transformation matrix.

Appendix: Technical Details

In the Appendix, we discuss an efficient approximation of DBI, and propose a nonlinear extension of our two-stage classifier selection algorithm.

A.1. Approximating DBI via (9)

We propose an approximate version of DBI, (9), which is easily estimated in practice. According to (8), we can calculate $DBI(S(\boldsymbol{X};\widehat{\boldsymbol{\theta}}_L))$ as

$$E\left[\tilde{\boldsymbol{X}}^{\dagger T}_{(-d)} Var\Big(\widehat{\boldsymbol{\eta}}^{\dagger *}_L | \boldsymbol{X}^{\dagger}_{(-d)}\Big) \tilde{\boldsymbol{X}}^{\dagger}_{(-d)}\right], \tag{A.1}$$

where $\tilde{\boldsymbol{X}}^{\dagger}_{(-d)} = (1, \boldsymbol{X}^{\dagger T}_{(-d)})^T$ and $\widehat{\boldsymbol{\eta}}^{\dagger *}_L = \Big(-\widehat{b}^{\dagger *}_L/\widehat{w}^{\dagger *}_{L,d}, -\widehat{w}^{\dagger *}_{L,1}/\widehat{w}^{\dagger *}_{L,d} \ldots, -\widehat{w}^{\dagger *}_{L,d-1}/\widehat{w}^{\dagger *}_{L,d}\Big)$. To further simplify (A.1), we need the following as an intermediate step.

Theorem 4 *If (L1)–(L5) hold and $\lambda_n = o(n^{-1/2})$, as $n \to \infty$,*

$$\sqrt{n}(\widehat{\boldsymbol{\theta}}_L - \boldsymbol{\theta}_{0L}) \xrightarrow{d} N(0, \Sigma_{0L}), \tag{A.2}$$

$$\sqrt{n}(\widehat{\boldsymbol{\theta}}^*_L - \widehat{\boldsymbol{\theta}}_L) \overset{d}{\Longrightarrow} N(0, \Sigma_{0L}) \quad \textit{conditional on } \mathcal{D}_n, \tag{A.3}$$

where $\Sigma_{0L} = H(\boldsymbol{\theta}_{0L})^{-1}G(\boldsymbol{\theta}_{0L})H(\boldsymbol{\theta}_{0L})^{-1}$. After data transformation, as $n \to \infty$,

$$\sqrt{n}(\widehat{\boldsymbol{\theta}}^{\dagger}_L - \boldsymbol{\theta}^{\dagger}_{0L}) \xrightarrow{d} N(0, \Sigma^{\dagger}_{0L}), \tag{A.4}$$

$$\sqrt{n}(\widehat{\boldsymbol{\theta}}^{\dagger *}_L - \widehat{\boldsymbol{\theta}}^{\dagger}_L) \overset{d}{\Longrightarrow} N(0, \Sigma^{\dagger}_{0L}) \quad \textit{conditional on } \mathcal{D}^{\dagger}_n, \tag{A.5}$$

where $\boldsymbol{\theta}^{\dagger}_{0L} = (b_{0L}, \boldsymbol{w}^T_{0L} R^T_L)^T$ and

$$\Sigma^{\dagger}_{0L} = \begin{pmatrix} \Sigma_b & \Sigma_{b,\boldsymbol{w}} R^T_L \\ R_L \Sigma_{\boldsymbol{w},b} & R_L \Sigma_{\boldsymbol{w}} R^T_L \end{pmatrix} \quad \textit{if we partition } \Sigma_{0L} \textit{ as } \begin{pmatrix} \Sigma_b & \Sigma_{b,\boldsymbol{w}} \\ \Sigma_{\boldsymbol{w},b} & \Sigma_{\boldsymbol{w}} \end{pmatrix}.$$

We omit the proof of Theorem 4 since (*A.2*) and (*A.3*) directly follow from (S.1) and Appendix D in Jiang et al. (2008), and (A.4) and (A.5) follow from the Delta method.

Let $\widehat{\boldsymbol{\eta}}_L^\dagger = \Big(-\widehat{b}_L^\dagger/\widehat{w}_{L,d}^\dagger, -\widehat{w}_{L,1}^\dagger/\widehat{w}_{L,d}^\dagger \ldots, -\widehat{w}_{L,d-1}^\dagger/\widehat{w}_{L,d}^\dagger \Big)$. According to (A.4) and (A.5), we know that $Var(\widehat{\boldsymbol{\eta}}_L^{\dagger *}|\boldsymbol{X}_{(-d)}^\dagger)$ is a consistent estimate of $Var(\widehat{\boldsymbol{\eta}}_L^\dagger)$ because $\widehat{\boldsymbol{\eta}}_L^{\dagger *}$ and $\widehat{\boldsymbol{\eta}}_L^\dagger$ can be written as the same function of $\widehat{\boldsymbol{\theta}}_L^{\dagger *}$ and $\widehat{\boldsymbol{\theta}}_L^\dagger$, respectively. Hence, we claim that

$$DBI\Big(S(\boldsymbol{X};\widehat{\boldsymbol{\theta}}_L)\Big) \approx E\left(\tilde{\boldsymbol{X}}_{(-d)}^{\dagger T} Var(\widehat{\boldsymbol{\eta}}_L^\dagger)\tilde{\boldsymbol{X}}_{(-d)}^\dagger\right).$$

Furthermore, we can approximate $Var(\widehat{\boldsymbol{\eta}}_L^\dagger)$ by $(w_{L,d}^\dagger)^{-2}[n^{-1}\Sigma_{0L,(-d)}^\dagger]$, where $n^{-1}\Sigma_{0L,(-d)}^\dagger$ is the asymptotic variance of the first d dimensions of $\widehat{\boldsymbol{\theta}}_L^\dagger$, since $\widehat{w}_{L,d}^\dagger$ is asymptotically normal with mean $w_{L,d}^\dagger$ and variance converging to 0 as n grows (Hinkley (1969)). Finally, we can get the desirable approximation (9) for DBI. ■

A.2. Nonlinear Extension

The extension of our two-stage algorithm to nonlinear classifiers contains two aspects: asymptotic normality of the K-CV error in Stage 1; the application of DBI in Stage 2. The former is still valid due to Hable (2012), and the latter is feasible by mapping the nonlinear decision boundaries to a higher dimensional space where the projected decision boundaries are linear.

Extension of Stage 1: We first modify several key concepts. The loss $L : \mathcal{X} \times \mathcal{Y} \times \mathbb{R} \to [0, \infty)$ is convex if it is convex in its third argument for every $(x, y) \in \mathcal{X} \times \mathcal{Y}$. A reproducing kernel Hilbert space (RKHS) H is a space of functions $f : \mathcal{X} \to \mathbb{R}$ which is generated by a kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Here the kernel k could be a linear kernel, a Gaussian RBF kernel, or a polynomial kernel.

Given i.i.d training samples $\mathcal{D}_n = \{(\boldsymbol{x}_i, y_i); i = 1, \ldots, n\}$ drawn from $P = (X, Y)$, the empirical function $f_{L,\mathcal{D}_n,\lambda_n}$ solves

$$\min_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} L(x_i, y_i, f(x_i)) + \lambda_n \|f\|_{\mathcal{H}}^2.$$

In the nonparametric case, the optimization problem of minimizing population risk is ill-posed because a solution is not necessarily unique, and small changes in P may have large effects on the solution. Therefore it is common to impose a bound on the complexity of the predictor and estimate a smoother approximation to the population version (Hable, 2012). For a fixed $\lambda_0 \in (0,\infty)$, we denote f_{L,P,λ_0} as the population function which solves

$$\min_{f\in\mathcal{H}} \int L(x,y,f(x))P(d(x,y)) + \lambda_0 \|f\|_{\mathcal{H}}^2.$$

The following conditions are assumed in Hable (2012) to prove the asymptotic normality of the estimated kernel decision function.

(N1) The loss L is a convex, P-square-integrable Nemitski loss function of order $p \in [1,\infty)$.

(N2) The partial derivatives $L^{'}(x,y,t) := \frac{\partial L}{\partial t}(x,y,t)$ and $L^{''}(x,y,t) := \frac{\partial^2 L}{\partial^2 t}(x,y,t)$ exist for every $(x,y,t) \in \mathcal{X}\times\mathcal{Y}\times\mathbb{R}$ and are continuous.

(N3) For every $a \in (0,\infty)$, there is $b_a^{'} \in L_2(P)$ and $b_a^{''} \in [0,\infty)$ such that, for every $(x,y) \in \mathcal{X}\times\mathcal{Y}$, $\sup_{t\in[-a,a]} |L^{'}(x,y,t)| \le b_a^{'}(x,y)$ and $\sup_{t\in[-a,a]} |L^{''}(x,y,t)| \le b_a^{''}$.

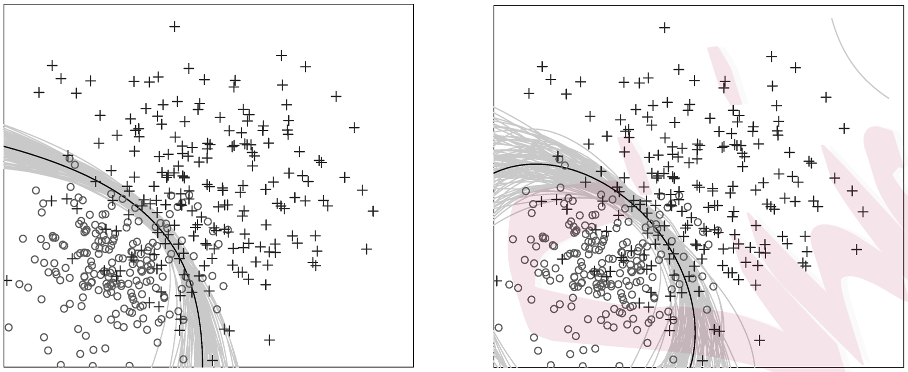
Proposition 1 *(Theorem 3.1, Hable (2012)) If (N1)-(N3) hold and* $\lambda_n = \lambda_0 + o(n^{-1/2})$*, for every* $\lambda_0 \in (0,\infty)$*, there is a tight, Borel-measurable Gaussian process* $\mathbb{H} : \Omega \to H$ *such that* $\sqrt{n}\Big(f_{L,\mathcal{D}_n,\lambda_n} - f_{L,P,\lambda_0}\Big) \to \mathbb{H}$.

Remark 2 Least squares, exponential, and logistic losses all satisfy (N1)-(N3), while LUM loss is not differentiable and does not satisfy (N2). Hable (2012) showed that any Lipschitz-continuous loss function (e.g. LUM loss) can be modified as a differentiable $\epsilon-$version of the loss function such that (N1)-(N3) are satisfied; see Remark 3.5 in Hable (2012).

In the nonlinear case, the GE D_{0L} and the K-CV error $\widehat{\mathcal{D}}_L$ are modified accordingly. The asymptotic normality of $\mathcal{W}_L = \sqrt{n}(\widehat{\mathcal{D}}_L - D_{0L})$ follows from Proposition 1, Corollary 3.3 in Hable (2014), and a slight modification of the proof of our Theorem 1. Then a perturbation-based resampling approach can be constructed analogously to Algorithm 1.

Extension of Stage 2: The concept of DBI is defined for linear decision boundaries. In order to measure the instability of nonlinear decision boundaries, we map the nonlinear decision boundaries to a higher dimensional space where the projected decision boundaries are linear.

Figure A1: The nonlinear perturbed decision boundaries for the least squares loss (left) and SVM (right) in the bivariate normal example with unequal variances.



Here we illustrate the estimation procedure via a bivariate normal example with sample size $n = 400$. Assume the underlying distributions of the two classes are $f_1 = N((-1,-1)^T, I_2)$ and $f_2 = N((1,1)^T, 2I_2)$ with equal prior probability. We map the input $\{x_1, x_2\}$ to the polynomial basis $\{x_1, x_2, x_1x_2, x_1^2, x_2^2\}$ and fit the linear large-margin classifiers using the expanded inputs. The instability of the original nonlinear decision boundary comes down to the instability of the linear boundaries in the expanded space. Figure A1 demonstrates 100 nonlinear perturbed decision boundaries for the least squares and SVM losses, where the former is visually more stable than the latter. Indeed, their corresponding DBI estimations in the expanded space capture this relationship in that the estimated DBI of the former is 0.017 and that of the latter is 0.354. ■

References

Adomavicius, G. and Zhang, J. (2010), On the Stability of Recommendations Algorithms, *ACM Conference on Recommender Systems*, 47-54.

Bousquet, O. and Elisseeff, A. (2002), Stability and Generalization, *Journal of Machine Learning Research*, **2**, 499-526.

Breiman, L. (1996), Heuristics of Instability and Stabilization in Model Selection, *Annals of Statistics*, **24**, 2350-2383.

Cortes, C. and Vapnik, V. (1995), Support-Vector Networks, *Machine Learning*, **20**, 273-279.

Demsar, J. (2006), Statistical Comparisons of Classifiers over Multiple Data Sets, *Journal of Machine Learning Research*, **7**, 1-30.

Dietterich, T. (1998), Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms, *Neural Computation*, **10**, 1895-1923.

Donoho, D.L., Maleki, A., Shahram, M., Rahman, I.U. and Stodden, V. (2009), Reproducible Research in Computational Harmonic Analysis., *IEEE Computing in Science and Engineering*, **11**, 8-18.

Efron, B. (1975), The Efficiency of Logistic Regression Compared to Normal Discriminant Analysis, *Journal of American Statistical Association*, **70**, 892-898.

Fan, J., Feng, Y., and Tong, X. (2012), A ROAD to Classification in High Dimensional Space, *Journal of the Royal Statistical Society, Series B*, **74**, 745-771.

Frank, A. and Asuncion, A. (2010), UCI Machine Learning Repository, *http://archive.ics.uci.edu/ml.*

Freund, Y. and Schapire, R. (1997), A Decision Theoretic Generalization of On-line Learning and An Application to Boosting, *Journal of Computer and System Sciences*, **55**, 119-139.

Gershoff, A., Mukherjee, A., and Mukhopadhyay, A. (2003), Consumer Acceptance of Online Agent Advice: Extremity and Positivity Effects, *Journal of Consumer Psychology*, **13**, 161-170.

Hable, R. (2012), Asymptotic Normality of Support Vector Machine Variants and Other Regularized Kernel Methods, *Journal of Multivariate Analysis*, **106**, 92-117.

Hable, R. (2014), Asymptotic Confidence Sets for General Nonparametric Regression and Classification by Regularized Kernel Methods, *Technical Report*.

Hastie, T., Tibshirani, R., and Friedman, J. (2009), The Elements of Statistical Learning: Data Mining, Inference, and Prediction, *Springer-Verlag: New York*.

Hoffmann-Jorgensen, J. (1984), Stochastic Processes on Polish Spaces. Unpublished.

Ioannidis, J.P.A. (2005), Why Most Published Research Findings Are False, *PLoS Medicine*, **2**, 696-701.

Jiang, B., Zhang, X., and Cai, T. (2008), Estimating the Confidence Interval for Prediction Errors of Support Vector Machine Classifiers, *Journal of Machine Learning Research*, **9**, 521-540.

Koo, J., Lee, Y., Kim, Y., and Park, C. (2008), A Bahadur Representation of the Linear Support Vector Machine, *Journal of Machine Learning Research*, **9**, 1343-1368.

Kraft, P., Zeggini, E. and Ioannidis, J.P.A. (2009), Replication in Genome-Wide Association Studies., *Statistical Science*, **24**, 561-573.

Lim, C. and Yu, B. (2016), Estimation Stability with Cross Validation, *Journal of Computational and Graphical Statistics*, To Appear.

Lim, T., Loh, W.Y., and Shih, Y.S. (2000), A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-three Old and New Classification Algorithms, *Machine Learning*, **40**, 203-229.

Liu, Y., Zhang, H., and Wu, Y. (2011), Hard or Soft Classification? Large-Margin Unified Machines, *Journal of American Statistical Association*, **106**, 166-177.

Meinshausen, N. and Bühlmann, P. (2010), Stability Selection, *Journal of the Royal Statistical Society, Series B*, **72**, 414-473.

Park, Y. and Wei, L.J. (2003), Estimating Subject-specific Survival Functions under the Accelerated Failure Time Model, *Biometrika*, **90**, 717-723.

Peng, R. D. (2009), Reproducible Research and Biostatistics, *Biostatistics*, **10**, 405-408.

Pollard, D. (1991), Asymptotics for Least Absolute Deviation Regression Estimators, *Econometric Theory*, **7**, 186-199.

Qiao, X. and Liu, Y. (2009), Adaptive Weighted Learning for Unbalanced Multicategory Classification, *Biometrics*, **65**, 159-168.

Rifkin, R. and Klautau, A. (2004), In Defense of One-Vs-All Classification, *Journal of Machine Learning Research*, **5**, 101-141.

Rocha, G., Wang, X., and Yu, B. (2009), Asymptotic distribution and sparsistency for $l1$ penalized parametric M-estimators, with applications to linear SVM and logistic regression, *Technical Report*.

Shah, R. and Samworth, R. (2013), Variable Selection with Error Control: Another Look at Stability Selection, *Journal of the Royal Statistical Society, Series B*, **75**, 55-80.

Shen, X. and Wang, L. (2007), Generalization Error for Multi-class Margin Classification, *Electronic Journal of Statistics*, **1**, 307-330.

Steinwart, I. (2007), How to Compare Different Loss Functions and Their Risks, *Constructive Approximation*, **26**, 225-287.

Sun, W., Wang, J., and Fang, Y. (2013), Consistent Selection of Tuning Parameters via Variable Selection Stability, *Journal of Machine Learning Research*, **14**, 3419-3440.

Sun, W., Qiao, X., and Cheng, G. (2016), Stabilized Nearest Neighbor Classifier and Its Statistical Properties, *Journal of American Statistical Association*, To Appear.

Valentini, G. and Dietterich, T. (2004), Bias-Variance Analysis of Support Vector Machines for the Development of SVM-Based Ensemble Methods, *Journal of Machine Learning Research*, **5**, 725-775.

Van Swol, L. and Sniezek, J. (2005), Factors Affecting the Acceptance of Expert Advice, *British Journal of Social Psychology*, **44**, 443-461.

Vapnik, V. (1998), Statistical Learning Theory, *John Wiley and Sons, New York.*

Wahba, G. (2002), Soft and Hard Classification by Reproducing Kernel Hilbert Space Methods, *Proceedings of the National Academy of Sciences*, **99**, 16524-16530.

Wang, J. (2010), Consistent Selection of the Number of Clusters via Cross Validation, *Biometrika*, **97**, 893-904.

Wang, J. (2013), Boosting the Generalized Margin in Cost-Sensitive Multiclass Classification, *Journal of Computational and Graphical Statistics*, **22**, 178-192.

Wang, J. and Shen, X. (2006), Estimation of Generalization Error: Random and Fixed Inputs, *Statistica Sinica*, **16**, 569-588.

Wolberg, W. H. and Mangasarian, O.L. (1990), Multisurface Method of Pattern Separation for Medical Diagnosis Applied to Breast Cytology, *Proceedings of the National Academy of Sciences*, **87**, 9193-9196.

Wu, Y. and Liu, Y. (2007), Robust Truncated Hinge Loss Support Vector Machines, *Journal of American Statistical Association*, **102**, 974-983.

Yu, B. (2013), Stability, *Bernoulli*, **19**, 1484–1500.

Zhang, C. and Liu, Y. (2013), Multicategory Large-margin Unified Machines, *Journal of Machine Learning Research*, **14**, 1349-1386.

Zhang, T. (2004), Statistical Behavior and Consistency of Classification Methods Based on Convex Risk Minimization, *Annals of Statistics*, **32**, 56-134.