Statistica Sinica **16**(2006), 569-588

ESTIMATION OF GENERALIZATION ERROR: RANDOM AND FIXED INPUTS

Junhui Wang and Xiaotong Shen

University of Minnesota

Abstract: In multicategory classification, an estimated generalization error is often used to quantify a classifier's generalization ability. As a result, quality of estimation of the generalization error becomes crucial in tuning and combining classifiers. This article proposes an estimation methodology for the generalization error, permitting a treatment of both fixed and random inputs, which is in contrast to the conditional classification error commonly used in the statistics literature. In particular, we derive a novel data perturbation technique, that jointly perturbs both inputs and outputs, to estimate the generalization error. We show that the proposed technique yields optimal tuning and combination, as measured by generalization. We also demonstrate via simulation that it outperforms cross-validation for both fixed and random designs, in the context of margin classification. The results support utility of the proposed methodology.

Key words and phrases: Averaging, logistic, margins, penalization, support vector.

1. Introduction

In classification, the generalization error is often used as a means to measure a classifier's accuracy of generalization. Estimating the generalization error therefore becomes important in tuning as well as combining classifiers in order to maximize the accuracy of classification. The central topic this article addresses is estimation of the generalization error when inputs can be both random and fixed.

In statistics, estimation of the conditional prediction error given fixed inputs has been extensively investigated, c.f., Efron (1983, 1986, 2004) and Shen and Huang (2005) for some discussions, but that of the generalization error for random inputs has not yet received much attention at all. In the context of linear regression, Breiman and Spector (1992) argued that ignoring randomness in design variables could lead to highly biased estimation of the prediction error, although the regression estimates remain unchanged regardless of random designs or not. Evidently, special attention is necessary with regard to random inputs in estimation of the generalization error.

Conventional techniques for estimating the generalization error are mainly based on cross-validation (CV), which uses one part of data for training while retaining the rest for testing. It is well known that CV has high variability resulting in instable estimation and selection (Devroye, Gyorfi and Lugosi (1996)). Efron (2004) showed that a covariance penalty is more accurate than CV in that it has a smaller variance while having essentially the same amount of bias when a conditional loss is used. In this article, we further develop the concept of covariance penalty in the context of estimating of the generalization error.

In our framework, we derive a random covariance penalty for a general classifier, together with a correction term that accounts for random inputs, where the correction term automatically reduces to zero when inputs are fixed. Furthermore, we derive a method that jointly perturbs input-output pairs to estimate both the random covariance penalty and the correction term. We show that the estimated generalization error based on the covariance penalty and the correction term is asymptotically optimal in generalization. This, together with our simulation, suggests that our random version of estimated covariance penalty is again more accurate than CV, which achieves our goal in optimal tuning and combination.

This paper is organized as follows. Section 2 formulates the problem of estimating generalization error, and proposes a data perturbation methodology. Section 3 establishes an optimality property of the proposed methodology. Section 4 applies the proposed technique to yield optimal tuning and optimal combination of large margin classifiers, followed by some numerical results. Section 5 discusses the methodologies. Technical details are given in Section 6.

2. Estimating Generalization Error

For k-class classification, a classifier (learner) ϕ is trained via a training sample $(X_i, Y_i)_{i=1}^n$, independent and identically distributed according to an unknown $P(x, y)$, where ϕ maps from $\mathbb{R}^d \to \{0, \ldots, k-1\}$, with $k > 1$ and d is the dimension of X. To analyze a learning scenario, accuracy on inputs outside the training set is examined. This is performed through an error function that measures the ability of generalization, and is known as the generalization error (GE). For any classifier ϕ, GE is defined as

$$GE(\phi) = P(Y \neq \phi(X)) = E(I(Y \neq \phi(X))), \tag{1}$$

where $I(\cdot)$ is the indicator, (X, Y) is independent and identically distributed according to $P(x, y)$, and independent of $(X_i, Y_i)_{i=1}^n$. The empirical version of GE, called the empirical generalization error (EGE), is defined as

$$EGE(\phi) = \frac{1}{n} \sum_{i=1}^{n} I(Y_i \neq \phi(X_i)). \tag{2}$$

In training, the classifier ϕ often involves a tuning parameter C, vector or scalar, whose value controls the trade-off between fitting and generalization. In what follows, we write ϕ as $\hat{\phi}_C$ to indicate its dependency on $(X_i, Y_i)_{i=1}^n$ and C.

The quantity $GE(\hat{\phi}_C)$ compares classifiers indexed by different values of C. If $P(\cdot, \cdot)$ were known, we could select an optimal classifier by minimizing $GE(\hat{\phi}_C)$ over the range of tuning parameter C or, equivalently, over a class of classifiers. In practice, $P(\cdot, \cdot)$ is unknown, so $GE(\hat{\phi}_C)$ needs to be estimated from data.

2.1. Motivation: Binary case

For motivation, we first examine the case with $Y \in \{0, 1\}$, and then generalize it to the multicategory case. Our basic strategy of estimating $GE(\hat{\phi}_C)$ is to seek the optimal loss estimator in a class of candidate loss estimators that yields an approximately unbiased estimate of $GE(\hat{\phi}_C)$. Note that there does not exist an exact unbiased estimate of $GE(\hat{\phi}_C)$. Naturally, one might estimate $GE(\hat{\phi}_C)$ by $EGE(\hat{\phi}_C)$, the empirical version of $GE(\hat{\phi}_C)$. However, $EGE(\hat{\phi}_C)$ suffers from the problem of overfitting, in contrast to $GE(\hat{\phi}_C)$. This is evident from the fact that the tuning parameter C yielding the smallest training error usually does not give the optimal performance in generalization or prediction. To prevent overfitting from occurring, we introduce a class of candidate loss estimators of the form:

$$EGE(\hat{\phi}_C) + \lambda(X^n, \hat{\phi}_C), \tag{3}$$

where $X^n = \{X_i\}_{i=1}^n$, and λ is an overfitting penalty function that is to be determined optimally. For optimal estimation of $GE(\hat{\phi}_C)$, we choose to minimize the L_2-distance between GE and (3),

$$E[GE(\hat{\phi}_C) - (EGE(\hat{\phi}_C) + \lambda(X^n, \hat{\phi}_C))]^2, \tag{4}$$

where the expectation E is taken with respect to $(X^n, Y^n) = (X_i, Y_i)_{i=1}^n$. Minimizing (4) with respect to $\lambda(X^n, \hat{\phi}_C)$ produces optimal $\lambda_o(X^n, \hat{\phi}_C)$, given expression in Theorem 1.

Theorem 1. *The optimal* $\lambda_o(X^n, \hat{\phi}_C)$ *that minimizes* (4) *over* $\lambda(X^n, \hat{\phi}_C)$ *is*

$$\lambda_o(X^n, \hat{\phi}_C) = 2n^{-1} \sum_{i=1}^{n} \mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n) + D_{1n}(X^n, \hat{\phi}_C) + D_{2n}(X^n), \tag{5}$$

where $D_{1n}(X^n, \hat{\phi}_C) = E(E(E(Y|X) - \hat{\phi}_C(X))^2 - n^{-1}\sum_i (E(Y_i|X_i) - \hat{\phi}_C(X_i))^2|X^n)$, *and* $D_{2n}(X^n) = E(\mathrm{Var}\,(Y|X)) - n^{-1}\sum_i \mathrm{Var}\,(Y_i|X_i)$.

In (5), $n^{-1}\sum_i \mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$ is averaged over covariances between Y_i and its predicted value $\hat{\phi}_C(X_i)$ at each observation (X_i, Y_i), which evaluates the

accuracy of prediction of $\hat{\phi}_C$ on X^n. Note that $\mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$ depends on the scale of Y_i. Thus the generalized degree of freedom of the classifier $\hat{\phi}$ is defined as $n(\sum_i \mathrm{Var}\,(Y_i|X_i))^{-1} \sum_i \mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$, which measures the degree of freedom cost in classification as well as tuning and combining.

The term D_{1n} can be decomposed as a difference between the true model error $E(E(Y|X)-\hat{\phi}_C(X))^2$ and its empirical version $n^{-1}\sum_i (E(Y_i|X_i)-\hat{\phi}_C(X_i))^2$. The disparity between these two errors comes from potential randomness of X, when sampled from an unknown distribution. In the situation of *fixed* design, D_{1n} is identical to zero, since the empirical distribution X^n is the same as that of X. In the situation of *random* design, D_{1n} is usually non-zero and needs to be estimated, in view of the result of Breiman and Spector (1992) and Breiman (1992) in a different context.

The term D_{2n}, on the other hand, is independent of $\hat{\phi}_C$. For the purpose of comparison, it suffices to use the comparative GE, which is defined as $CGE(\hat{\phi}_C) = GE(\hat{\phi}_C) - D_{2n}(X^n)$, as opposed to the original GE. With GE replaced by CGE in (4), we find the optimal $\lambda_o(X^n, \hat{\phi}_C)$ for CGE to be $2n^{-1}\sum_i \mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n) + D_{1n}(X^n, \hat{\phi}_C)$.

2.2. Estimation

Using (3) and (5) in Theorem 1, we propose to estimate $CGE(\hat{\phi}_C)$ by

$$\widehat{CGE}(\hat{\phi}_C) = EGE(\hat{\phi}_C) + 2n^{-1}\sum_{i=1}^{n} \widehat{\mathrm{Cov}}\,(Y_i, \hat{\phi}_C(X_i)|X^n) + \widehat{D}_{1n}(X^n, \hat{\phi}_C), \quad (6)$$

with $\widehat{\mathrm{Cov}}$ the estimated covariance, and $\widehat{D}_{1n}$ the estimated D_{1n}. In the situation of *fixed* design, $D_{1n} \equiv 0$, and (6) reduces to $\widehat{CGE}(\hat{\phi}_C) = EGE(\hat{\phi}_C) + 2n^{-1}\sum_i \widehat{\mathrm{Cov}}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$.

There are two major difficulties in estimating CGE in (6). First, there does not exist an exact unbiased estimate of $\sum_i \mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$, because Y_i follows a Bernoulli distribution. Second, only one realization of data is available for estimating the unobserved $\sum_i \mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$ and D_{1n}. Consequently, a resampling method of some type is required. However, it is known that the conventional bootstrap may not work when classification involves tuning and combining with discontinuity, c.f., Denby, Landwehr and Mellows (2004).

To overcome these difficulties, we propose a novel data perturbation technique based on swapping values of inputs and outputs (labels) to estimate $\sum_i \mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$ and D_{1n}. The learning accuracy of the classifier based on perturbed data estimates the sensitivity of classification, which yields an estimated GE.

First perturb X_i, $i = 1, \ldots, n$, via its empirical distribution $\hat{F}$, followed by flipping the corresponding label Y_i with a certain probability given the perturbed X_i. This generates perturbations for assessing accuracy of generalization of a classifier. More precisely, for $i = 1, \ldots, n$, let

$$X_i^* = \begin{cases} X_i \text{ with probability } 1 - \tau, \\ \tilde{X}_i \text{ with probability } \tau, \end{cases} \tag{7}$$

where $\tilde{X}_i$ is sampled from $\hat{F}$. This step can given an X-fixed design. A perturbed Y_i^* is

$$Y_i^* = \begin{cases} Y_i \text{ with probability } 1 - \tau, \\ \tilde{Y}_i \text{ with probability } \tau, \end{cases} \tag{8}$$

where $0 \le \tau \le 1$ is the size of perturbation, and $\tilde{Y}_i \sim \text{Bin}(1, \hat{p}_i(X_i^*))$, with $\hat{p}_i(X_i^*)$ an initial probability estimate of $E(Y_i|X_i^*)$, obtained via the same classification method that defines $\hat{\phi}_C$, or logistic regression if the classification method does not yield an probability estimate, such as in the case of support vector machine.

For simplicity, denote by E^*, Var^* and Cov^* the conditional mean, variance, and covariance with respect to $Y^{*n} = \{Y_i^*\}_{i=1}^n$, given (X^{*n}, Y^n), with $X^{*n} = \{X_i^*\}_{i=1}^n$. The perturbed Y_i^* has the following properties: (1) its conditional mean $E^*Y_i^* = (1-\tau)Y_i + \tau E(\tilde{Y}_i|X_i^*) = (1-\tau)Y_i + \tau\hat{p}_i(X_i^*)$, and (2) its conditional variance $\text{Var}^*(Y_i^*) = E^*(Y_i^{*2}) - (E^*(Y_i^*))^2 = \tau\text{Var}^*(\tilde{Y}_i) + \tau(1-\tau)(Y_i - E(\tilde{Y}_i|X_i^*))^2 = \tau\hat{p}_i(X_i^*)(1-\hat{p}_i(X_i^*)) + \tau(1-\tau)(Y_i - \hat{p}_i(X_i^*))^2$.

We now provide some heuristics for our proposed estimator. To estimate $\text{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$, note that it equals $\text{Var}\,(Y_i|X_i)[\text{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)/\text{Var}\,(Y_i|X_i)]$. Then we can estimate $\text{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n)/\text{Var}\,(Y_i|X_i)$ by $\text{Cov}^*(Y_i^*, \hat{\phi}_C^*(X_i^*)|X^{*n})/\text{Var}^*(Y_i^*)$. Additionally, $\text{Var}\,(Y_i|X_i)/\text{Var}^*(Y_i^*)$ is estimated by $1/K(Y_i, \hat{p}_i(X_i^*))$ with $K(Y_i, \hat{p}_i(X_i^*)) = \tau + \tau(1-\tau)(Y_i - \hat{p}_i(X_i^*))^2/[\hat{p}_i(X_i^*)(1-\hat{p}_i(X_i^*))]$, when $\text{Var}\,(Y_i|X_i)$ is estimated by $\text{Var}^*(\tilde{Y}_i) = \hat{p}_i(X_i^*)(1-\hat{p}_i(X_i^*))$. This leads to our proposed estimator

$$\widehat{\text{Cov}}\,(Y_i, \hat{\phi}_C(X_i^*)|X^{*n}) = \frac{1}{K(Y_i, \hat{p}_i(X_i^*))}\text{Cov}^*(Y_i^*, \hat{\phi}_C^*(X_i^*)|X^{*n}), \quad i = 1, \ldots, n, \tag{9}$$

where $\hat{\phi}_C^*$ is an estimated decision function via the same classification routine applied to $(X_i^*, Y_i^*)_{i=1}^n$.

To estimate D_{1n}, note that $E(E(Y|X) - \hat{\phi}_C^*(X))^2$ can be estimated by $n^{-1}\sum_i(\hat{p}(X_i) - \hat{\phi}_C^*(X_i))^2$ when $E(Y|X) = p(X)$ is estimated by $\hat{p}_i(X)$, while $(E(Y_i|X_i) - \hat{\phi}(X_i))^2$ may be estimated by $(\hat{p}_i(X_i^*) - \hat{\phi}_C^*(X_i^*))^2$ when $E(Y_i|X_i)$ is

replaced by $\hat{p}_i^*(X_i^*)$, $i = 1, \ldots, n$. This leads to

$$\begin{aligned}&\widehat{D}_{1n}(X^n, \hat{\phi}_C)\\ &= E^*\Big(n^{-1}\sum_{i=1}^{n}(\hat{p}_i(X_i) - \hat{\phi}_C^*(X_i))^2 - n^{-1}\sum_{i=1}^{n}(\hat{p}_i^*(X_i^*) - \hat{\phi}_C^*(X_i^*))^2 | X^{*n}\Big), \end{aligned} \tag{10}$$

where $\hat{\phi}_C^*$ is trained via $(X_i^*, Y_i^*)_{i=1}^n$, and $\hat{p}_i^*(X_i^*)$ is an estimated $E(Y_i^*|X_i^*)$.

Based on (9) and (10), we obtain $\widehat{CGE}(\hat{\phi}_C)$ in (6). Note that the proposed estimator $\widehat{CGE}(\hat{\phi}_C)$ is constructed based on perturbed data, and can be generally computed via Monte Carlo (MC) approximation. In some situations, however, $\widehat{CGE}(\hat{\phi}_C)$ can be computed analytically without recourse to MC methods, permitting fast implementation, as in Fisher's linear discrimination. For problems considered in this article, we use a MC numerical approximation for implementation. First, generate D perturbed samples $\{X_i^{*l}\}_{i=1}^n$ according to (7), $l = 1, \ldots, D$. Second, for each sample $\{X_i^{*l}\}_{i=1}^n$, generate D perturbed samples $\{Y_i^{*lm}\}_{i=1}^n$ according to (8), $m = 1, \ldots, D$. For $l, m = 1, \ldots, D$, $i = 1, \ldots, n$, compute $\widehat{\mathrm{Cov}}^*(Y_i^*, \hat{\phi}_C^*(X_i^*)|X^n) = (D^2-1)^{-1}\sum_{l,m}\hat{\phi}_C^{*lm}(X_i^{*l})(Y_i^{*lm} - \overline{Y}_i^*)$, where $\hat{\phi}_C^{*lm}$ is trained via $\{X_i^{*l}, Y_i^{*lm}\}_{i=1}^n$, and $\overline{Y}_i^* = D^{-2}\sum_{l,m} Y_i^{*lm}$. Now (9) is approximated by the corresponding sample MC covariance, i.e.,

$$\begin{aligned}&\widehat{\mathrm{Cov}}\,(Y_i, \hat{\phi}_C(X_i)|X^n)\\ &\approx \frac{1}{D^2-1}\sum_{l,m=1}^{D}\frac{1}{K(Y_i, \hat{p}_i(X_i^{*l}))}\hat{\phi}_C^{*lm}(X_i^{*l})(Y_i^{*lm} - \overline{Y}_i^*); \quad i = 1, \ldots, n, \end{aligned} \tag{11}$$

while (10) is approximated as

$$\begin{aligned}&\widehat{D}_{1n}(X^n, \hat{\phi}_C)\\ &\approx \frac{1}{n(D^2-1)}\sum_{i=1}^{n}\sum_{l,m=1}^{D}\Big((\hat{p}_i(X_i) - \hat{\phi}_C^{*lm}(X_i))^2 - (\hat{p}_i^*(X_i^{*l}) - \hat{\phi}_C^{*lm}(X_i^{*l}))^2\Big). \end{aligned} \tag{12}$$

The estimated CGE in (6) is now MC-approximated, with approximated $\widehat{\mathrm{Cov}}$ and $\widehat{D}_{1n}$ given in (11) and (12). By the Law of Large Numbers, (11) and (12) converge to (9) and (10), respectively, and hence MC-approximated CGE converges to (6), as $D \to \infty$. In practice, we recommend that D be at least $n^{1/2}$ to ensure the precision of MC approximation.

2.3. Sensitivity with respect to τ and initial probabilities

Our proposed estimator of CGE in (6) depends on the value of $0 < \tau < 1$, with $\tau = 0.5$ recommended in implementation based on our limited numerical

experience. This dependency on τ can be removed by a data-driven selection routine that may be computationally intensive. One proposal is to employ CV, or our proposed method once again, to seek the optimal τ by minimizing CV or (6) with respect to $\tau \in (0,1)$. For the problem considered in this article, we fix $\tau = 0.5$ for simplicity and ease of computation. A sensitivity study of our proposed method with respect to τ is summarized in Section 4.1.

The initial probability estimation for $p_i(x_i)$ and $p_i^*(x_i)$ may be also important for in (6). The dependency of initial probability estimation may be removed at the expense of additional computational cost. Specifically, suppose that different probability estimation methods are indexed by θ, the optimal θ can be obtained by minimizing the estimated Kullback-Leibler (KL) loss between the true and estimated probabilities, over θ,

$$\widehat{K}(p, \hat{p}(\theta)) = -n^{-1} \sum_{i=1}^{n} \log L(Y_i|\hat{p}_i(\theta)) + n^{-1} \sum_{i=1}^{n} \widehat{\mathrm{Cov}}\,(\log(\hat{p}_i(\theta)) - \log(1-\hat{p}_i(\theta)), Y_i), \tag{13}$$

where $L(Y_i|\hat{p}_i(\theta))$ is the likelihood function with parameter $\hat{p}(\theta)$, c.f., Shen et al. (2004) for details. In this article, for simplicity, we use logistic regression to estimate the initial probabilities $p_i(x_i)$ and $p_i^*(x_i)$, $i = 1, \ldots, n$, which is sensible, as shown in a sensitivity study in Section 4.1.

2.4. Multicategory case

To treat the multicategory case, we introduce a mapping $t : \{0, \ldots, k-1\} \to \{0,1\}^k$, which permits a treatment of the multicategory case via the result in the binary case. Precisely, $t(j)$ is defined as $(\underbrace{0, \ldots, 0}_{j}, 1, \underbrace{0, \ldots, 0}_{k-j-1})$ for $j \in \{0, \ldots, k-1\}$. With this mapping, Y and $\hat{\phi}_C(X)$ are converted to vector representations, denoted by $t(Y)$ and $t(\hat{\phi}_C(X))$. Now let $Z = (Z^{(0)}, \ldots, Z^{(k-1)}) = t(Y)$, and the corresponding classification rule be $\hat{\phi}_C^t = (\hat{\phi}_C^{t(0)}, \ldots, \hat{\phi}_C^{t(k-1)}) = t(\hat{\phi}_C)$. By definition, $\{\hat{\phi}_C, (X_i, Y_i)_{i=1}^n\}$ maps one-to-one onto $\{\hat{\phi}_C^t, (X_i, Z_i)_{i=1}^n\}$, with $Z_i = t(Y_i)$. Therefore, $GE(\hat{\phi}_C) = GE(\hat{\phi}_C^t) = P(Z \neq \hat{\phi}_C^t(X))$. More importantly, under this new setting, $GE(\hat{\phi}_C)$ can be written as a sum of the GE's of k binary problems.

Lemma 1. *We have*

$$GE(\hat{\phi}_C) = GE(\hat{\phi}_C^t) = P(Z \neq \hat{\phi}_C^t(X)) = \frac{1}{2} \sum_{j=0}^{k-1} P(Z^{(j)} \neq \hat{\phi}_C^{t(j)}(X)). \tag{14}$$

In addition, $EGE(\hat{\phi}_C) = EGE(\hat{\phi}_C^t) = (1/2) \sum_{j=0}^{k-1} EGE(\hat{\phi}_C^{t(j)})$.

Note that the $\hat{\phi}_C^{t(j)}(X)$'s in Lemma 1 are internally consistent, that is, if $\hat{\phi}_C^{t(j_o)}(X) = 1$, then $\hat{\phi}_C^{t(j)}(X) = 0$ for all $j \neq j_o$. Therefore the decomposition in (14) differs from the usual decomposition in multicategory classification with k separate components, and is applicable to classifiers with different class codings, such as one-vs-rest SVM with coding $\{1, \ldots, k\}$, and multicategory SVM with vector coding (Lee, Lin and Wahba (2004)).

An application of Lemma 1 and (6) yields the estimated GE of $\hat{\phi}_C$ as

$$\widehat{GE}(\hat{\phi}_C) = EGE(\hat{\phi}_C) + \sum_{j=0}^{k-1} \Big(\frac{1}{n} \sum_{i=1}^{n} \widehat{\mathrm{Cov}}\,(Z_i^{(j)}, \hat{\phi}_C^{t(j)}(X_i)|X^n) + \frac{1}{2}\widehat{D}_{1n}(X^n, \hat{\phi}_C^{t(j)}) + \frac{1}{2}\widehat{D}_{2n}(X^n, Z^{n(j)}) \Big),$$

where $Z^{n(j)} = (Z_i^{(j)})_{i=1}^n$ and $\widehat{D}_{2n}(X^n, Z^{n(j)}) = E(\mathrm{Var}\,(Z^{(j)}|X^n)) - n^{-1}\sum_i \mathrm{Var}\,(Z_i^{(j)}|X_i)$, which leads to the corresponding comparative GE $CGE(\hat{\phi}_C)$, as well as the estimator

$$\widehat{CGE}(\hat{\phi}_C) = EGE(\hat{\phi}_C) + \sum_{j=0}^{k-1} \Big(\frac{1}{n} \sum_{i=1}^{n} \widehat{\mathrm{Cov}}\,(Z_i^{(j)}, \hat{\phi}_C^{t(j)}(X_i)|X^n) + \frac{1}{2}\widehat{D}_{1n}(X^n, \hat{\phi}_C^{t(j)}) \Big), \tag{15}$$

where $EGE(\hat{\phi}_C)$ is the training error.

To compute $\widehat{\mathrm{Cov}}\,(Z_i^{(j)}, \hat{\phi}_C^{t(j)}(X_i)|X^n)$ and $\widehat{D}_{1n}$ in (15), we now modify the data perturbation technique in the binary case. Let $M(p_{i0}(X_i), \ldots, p_{i,k-1}(X_i))$ denote the conditional distribution of Y_i given X_i, where $p_{ij}(X_i) = P(Y_i = j|X_i)$ and $\sum_j p_{ij}(X_i) = 1$. Generate X_i^*, $i = 1, \ldots, n$, as

$$X_i^* = \begin{cases} X_i \text{ with probability } 1 - \tau, \\ \tilde{X}_i \text{ with probability } \tau, \end{cases}$$

where $\tilde{X}_i$ is sampled from the empirical distribution of X^n. Generate Y_i^*, $i = 1, \ldots, n$, as

$$Y_i^* = \begin{cases} Y_i \text{ with probability } 1 - \tau, \\ \tilde{Y}_i \text{ with probability } \tau, \end{cases}$$

where $\tilde{Y}_i$ is sampled from $M(\hat{p}_{i0}(X_i^*), \ldots, \hat{p}_{i,k-1}(X_i^*))$, with $\hat{p}_{ij}(X_i^*)$ the estimated $P(Y_i = j|X_i^*)$.

Let $Z_i^* = t(Y_i^*)$ be the transformed perturbed response, and $\hat{\phi}_C^{t*}$ be the corresponding transformed classifiers based on $(X_i^*, Z_i^*)_{i=1}^n$. The MC approximations

of $\widehat{\mathrm{Cov}}$ and $\widehat{D}_{1n}$ in (15) are given as

$$\begin{aligned}
&\widehat{\mathrm{Cov}}\,(Z_i^{(j)}, \phi_C^{t(j)}(X_i)|X^n)\\
&\approx \frac{1}{D^2-1}\sum_{j=0}^{k-1}\sum_{l,m=1}^{D}\frac{1}{K(Z_i^{(j)}, \hat{p}_{ij}(X_i^{*l}))}\hat{\phi}_C^{t*lm(j)}(X_i^{*l})(Z_i^{*lm(j)}-\bar{Z}_i^{*(j)});\ i=1,\ldots,n,\\
&\widehat{D}_{1n}(X^n, \hat{\phi}_C^{t(j)})\\
&\approx \frac{1}{n(D^2-1)}\sum_{i=1}^{n}\sum_{j=0}^{k-1}\sum_{l,m=1}^{D}\left((\hat{p}_{ij}(X_i)-\hat{\phi}_C^{t*lm(j)}(X_i))^2-(\hat{p}_{ij}^*(X_i^{*l})-\hat{\phi}_C^{t*lm(j)}(X_i^{*l}))^2\right),
\end{aligned}$$

where $\hat{p}_{ij}(X_i)$ and $\hat{p}_{ij}^*(X_i^{*l})$ are estimates of $P(Y_i = j|X_i)$ and $P(Y_i^* = j|X_i^{*l})$, respectively. Substituting these two approximations into (15), we obtain the proposed MC approximated CGE in the multicategory case.

3. Theory

In this section, we develop a theory concerning the proposed data perturbation technique. Particularly, we show that $\hat{C}$ minimizing (15) recovers the ideal performance that knowledge of the true $P(x, y)$ would have brought. That is to say our proposed technique yields the optimal tuning parameter $\hat{C}$, and hence the optimal classification rule $\hat{\phi}_{\hat{C}}$ against any other classifier in terms of generalization. To establish the theory, the following technical assumptions are made.

(C.1): (Integrability) For some $\delta > 0$, $E\sup_{\tau\in(0,\delta)}|\hat{\lambda}(X^n, \hat{\phi}_C)| < +\infty$.

(C.2): (Loss and risk) $\lim_{n\to\infty}\sup_C |GE(\hat{\phi}_C)/E(GE(\hat{\phi}_C))-1| = 0$ in probability.

(C.3): (Consistency of initial estimates) For almost all X, $\hat{p}_{ij}(X) \to p_{ij}(X)$, as $n \to \infty$; $i = 1, \ldots, n$, $j = 1, \ldots, k$.

(C.4): (Positive generalization error) Assume that $\inf_C E(GE(\hat{\phi}_C)) > 0$.

Assumption (C.1) which ensures that expectations and limits can be exchanged. Assumption (C.2) specifies the relationship between loss and risk, analogous to (2.4) of Li (1987). Assumption (C.3) requires that the initial probability estimates of perturbation be sufficiently good. As a consequence, the perturbed data approximately follows the same distribution as the original training data. Assumption (C.4) secures the validity of the comparison ratios in (16) and (17).

Theorem 2. *For k-class classification with $Y \in \{0, \ldots, k-1\}$ and $k > 1$, let $\hat{C}$ be the minimizer of* (15) *with respect to C. Under* (C.1) *and* (C.3),

$$\lim_{n\to\infty}\left(\lim_{\tau\to 0^+} E(GE(\hat{\phi}_{\hat{C}}))/\inf_C E(GE(\hat{\phi}_C))\right) = 1. \tag{16}$$

If additionally, (C.2) *holds, then*

$$\lim_{n\to\infty}\Big(\lim_{\tau\to 0^+} GE(\hat{\phi}_{\hat{C}})/\inf_{C} GE(\hat{\phi}_C)\Big) = 1. \tag{17}$$

There is an interesting connection between Theorem 2 and the Rao-Blackwell decomposition of Efron (2004) for fixed designs. Similarly, our estimated covariance $\widehat{\mathrm{Cov}}\,(Y_i, \hat{\phi}_C(X_i)|X^n)$ in (6) has the same Rao-Blackwell decomposition, which means that it has smaller variance while retaining the same bias. This means that $\widehat{CGE}(\hat{\phi}_C)$ may yield substantially higher accuracy in estimating GE in the finite sample situation, while retaining asymptotic optimality for both fixed and random inputs.

4. Numerical Examples

The choice of C is crucial to many classifiers, such as penalized logistic regression, a support vector machine (SVM, Cortes and Vapnik (1995)), and ψ-learning (Shen et al. (2003)). Optimal estimation of C may be necessary to minimize GE with respect to C. This section examines some examples, including simulations and data, to illustrate accuracy of our proposed method with respect to tuning and combining. The optimal $\hat{C}$ is obtained by minimizing (15), where the regularization solution path of SVM (Hastie et al. (2004)) is used to reduce computational cost. Experiments are performed in R 2.0.0.

4.1. Selection of tuning parameters

We investigate the effectiveness of our proposed method, denoted as GDF, and compare it against cross-validation (CV) in the selection of tuning parameters. Here the crystal ball estimate(CB, Breiman (1996)) is used as a baseline for comparison, estimating GE on a left-out testing sample. The performance of a given method is measured by the testing error, averaged over 100 simulation replications. The amount of improvement of GDF over CV is defined as

$$\frac{(T(CV) - T(CB)) - (T(GDF) - T(CB))}{T(CV) - T(CB)},$$

where $T(\cdot)$ is the testing error for a given method, and $T(\cdot) - T(CB) \geq 0$ for all methods.

Simulations are conducted for classifiers of three types: linear SVM, nonlinear SVM with Gaussian kernel, and penalized logistic regression. The SVM in the binary case can be obtained from

$$\min_f \; C\sum_{i=1}^{n} L(y_i, f(x_i)) + \frac{1}{2}J(f), \tag{18}$$

where $L(y_i, f(x_i)) = (1 - y_i f(x_i))_+$ is the so-called hinge loss, and $J(f) = \|f\|_K^2$ with $K(\cdot, \cdot)$ a kernel. In our context, $K(x, y)$ is $\langle x, y \rangle$ in the linear case, or $K(x, y)$ is $\exp(-\sigma^{-2}\|x - y\|^2)$. For simplicity, σ^2 is set to be the dimension of the input matrix X^n, a default value in the "svm" routine of R. This is because C plays a similar role as σ^2, and it is easier to optimize with respect to C if σ^2 is estimated. Similarly, penalized logistic regression can be obtained from (18), with hinge loss $(1 - y_i f(x_i))_+$ replaced by $\log(1 + e^{-y_i f(x_i)})$ and $K(x, y)$ by $\langle x, y \rangle$.

Two simulation are considered, together with four benchmark examples from the UCI repository (Blake and Merz (1998)): Wisconsin Breast Cancer (WBC), Diabetes, Iris and Wine. In each example, the optimal C is obtained by minimizing CB, GDF and 10-fold CV with respect to a discretized grid of $C \in [10^{-3}, 10^3]$, via a grid search.

Example 1. Data $\{(X_{i1}, X_{i2}, Y_i),\ i = 1, \ldots, 1,000\}$ are generated as follows. First, $\{X_{i1}, X_{i2}\}$, $i = 1, \ldots, 1,000$, are sampled according to the uniform distribution over a unit disk $\{(X_1, X_2) : X_1^2 + X_2^2 \leq 1\}$. Second, $Y_i = 1$ if $X_{i1} \geq 0$ and -1 otherwise, $i = 1, \ldots, n$. Third, a random choice of 10% of the sample have their labels flipped to generate the nonseparable situation. A random 100 instances are selected for training, the remaining 900 instances are for testing.

Example 2. Data $\{(X_{i1}, \ldots, X_{i10}, Y_i),\ i = 1, \ldots, 1,000\}$ are generated as follows. First, X_{ij}, $i = 1, \ldots, 1,000$, $j = 1, \ldots, 10$, are generated from the standard normal distribution. Second labels are assigned to each observation as $Y_i = \text{Sign}(\log((X_{i1} + \cdots + X_{i5})^2) + \sin(X_{i6} + \cdots + X_{i10}))$. A random selection of 100 instances are for training, the remaining 900 instances are for testing.

Benchmarks. Four benchmark examples, WBC, Diabetes, Iris and Wine, are examined. The first two are binary, while the last two use three categories All data examples are randomly divided into halves, for training and testing. Description of these data examples is given in Table 1.

Table 1. Description of all data examples in our simulations.

Data	training size	testing size	# covariates	# categories
Example1	100	900	2	2
Example2	100	900	10	2
WBC	341	341	9	2
Diabetes	384	384	8	2
Iris	75	75	4	3
Wine	89	89	13	3

Under the same setting, CB, GDF and CV are compared with respect to the three types of classifiers. Their performances, averaged over 100 simulations replications, are reported in Table 2.

Table 2. Averaged testing errors as well as the estimated standard errors (in parenthesis) with respect to $\hat{C}$ via three selection methods over 100 simulation replications. Here SVM_G, SVM_L and PLR represent Gaussian kernel SVM, linear SVM, and penalized logistic regression, respectively.

Dataset	Classifier	CB	GDF	CV	% Improv
Example1	SVM_G	0.143(0.0020)	0.158(0.0024)	0.163(0.0023)	25.0%
	SVM_L	0.140(0.0028)	0.149(0.0033)	0.153(0.0026)	30.8%
	PLR	0.148(0.0026)	0.152(0.0024)	0.153(0.0026)	20.0%
Example2	SVM_G	0.320(0.0012)	0.336(0.0018)	0.341(0.0016)	23.8%
	SVM_L	0.337(0.0005)	0.342(0.0029)	0.354(0.0040)	70.6%
	PLR	0.337(0.0005)	0.344(0.0027)	0.354(0.0034)	58.8%
WBC	SVM_G	0.029(0.0006)	0.033(0.0007)	0.037(0.0008)	50.0%
	SVM_L	0.026(0.0006)	0.033(0.0009)	0.034(0.0008)	12.5%
	PLR	0.027(0.0006)	0.033(0.0007)	0.035(0.0007)	25.0%
Diabetes	SVM_G	0.233(0.0016)	0.244(0.0016)	0.245(0.0019)	8.3%
	SVM_L	0.223(0.0015)	0.231(0.0016)	0.232(0.0016)	11.1%
	PLR	0.223(0.0016)	0.231(0.0017)	0.232(0.0017)	11.1%
Iris	SVM_G	0.033(0.0018)	0.047(0.0024)	0.052(0.0025)	26.3%
	SVM_L	0.021(0.0015)	0.042(0.0025)	0.045(0.0026)	12.5%
Wine	SVM_G	0.016(0.0013)	0.023(0.0015)	0.027(0.0017)	36.4%
	SVM_L	0.013(0.0012)	0.029(0.0017)	0.031(0.0017)	11.1%

Table 3. Sensitivity study. Averaged testing errors as well as the estimated standard errors (in parenthesis) of GDF as a function of perturbation size τ, based on 100 simulation replications.

Dataset	Classifier	$\tau = 0.1$	$\tau = 0.3$	$\tau = 0.5$	$\tau = 0.7$	$\tau = 0.9$
Example1	SVM_G	0.161	0.160	0.158	0.162	0.164
		(0.0024)	(0.0025)	(0.0024)	(0.0026)	(0.0031)
	SVM_L	0.150	0.150	0.149	0.149	0.149
		(0.0021)	(0.0032)	(0.0033)	(0.0031)	(0.0035)
	PLR	0.151	0.150	0.152	0.153	0.153
		(0.0025)	(0.0025)	(0.0024)	(0.0026)	(0.0029)
Example2	SVM_G	0.335	0.336	0.336	0.336	0.334
		(0.0018)	(0.0017)	(0.0018)	(0.0020)	(0.0027)
	SVM_L	0.340	0.341	0.342	0.343	0.342
		(0.0028)	(0.0029)	(0.0029)	(0.0029)	(0.0031)
	PLR	0.344	0.344	0.344	0.345	0.345
		(0.0026)	(0.0027)	(0.0027)	(0.0029)	(0.0029)

Table 4. Sensitivity study. Averaged testing errors as well as the estimated standard errors (in parenthesis) of GDF with and without adaptive estimated initial probabilities, based on 30 simulation replications.

Dataset	Classifier	w/ adp.	w/o adp.
Example1	SVM_G	0.158(0.0023)	0.158(0.0024)
	SVM_L	0.148(0.0028)	0.149(0.0033)
Example2	SVM_G	0.335(0.0011)	0.336(.0018)
	SVM_L	0.343(0.0026)	0.342(0.0029)

From Table 2, we note that GDF outperforms CV in all examples, with improvement ranging from 8.3% to 50.0%. The amount of improvement, however, depends on the examples and the type of classifiers. Furthermore, the choice of C appears to be more critical to nonlinear classifiers.

We now investigate the sensitivity of the performance of GDF to the perturbation size τ and initial probability estimation, respectively, via a small simulation study. The simulation study is conducted in Examples 1 and 2, with $\tau = 0.1, 0.3, 0.5, 0.7, 0.9$, and initial probability estimation. As indicated in Table 3, the performance of GDF hardly varies as a function of τ, and $\tau = 0.5$ seems to be a good empirical choice in all situations. Similarly, Table 4 suggests that the performance of GDF with initial probabilities estimated by the logistic regression is close to that of GDF when they are estimated by adaptively minimizing (13) with respect to different levels of penalization in penalized logistic regression.

Finally, we examine the accuracy of GDF and CV in estimating GE on a randomly chosen training sample in Example 1. As illustrated in Figure 1, GDF is closer to CB, and has lower variability compared to CV, although both GDF and CV capture the trend of GE. Note that CB converges to the true GE when the size of testing sample tends to infinity. Consequently, GDF yields a minimizer that is closer to that of GE, whereas the minimizer estimated by CV is skewed to the right of the true one.

4.2. Combining SVM's

We now apply the proposed methodology to combine SVM classifiers with different types of kernel to yield better performance than each individual classifier. The idea outlined here may be useful for combining classifiers of any types.

Now consider, two SVM classifiers: linear SVM $\text{Sign}(\hat{f}_1)$ and Gaussian kernel SVM $\text{Sign}(\hat{f}_2)$, where $\hat{f}_1(X) = \langle \hat{w}_1, X\rangle + b_1$ takes a linear form, while $\hat{f}_2(X) = \langle \hat{w}_2, X\rangle_K + b_2$ is defined by the Gaussian kernel $K(x, z)$ as in Section 4.1. Our goal is to seek the optimal weight $0 \le a \le 1$ such that the combined decision function $\hat{f}_a = a\hat{f}_1 + (1-a)\hat{f}_2$, equivalently, $\hat{f}_C = a^{-1}\hat{f}_a = \hat{f}_1 + C\hat{f}_2$, with $C = [(1-a)/a] \in [0, \infty]$, yields a classifier $\hat{\phi}_C = \text{Sign}(\hat{f}_C)$ that can outperform both $\text{Sign}(\hat{f}_i); i = 1, 2$.

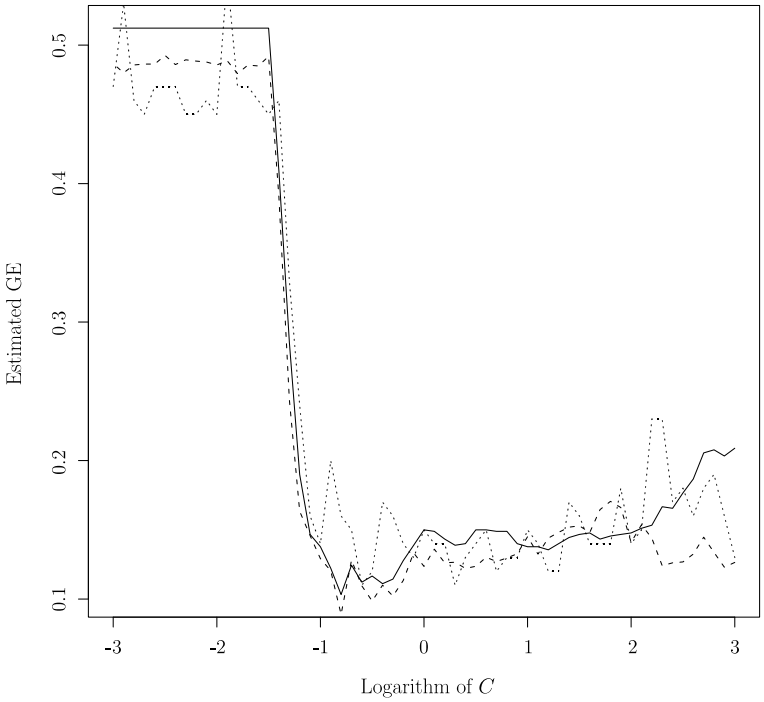

Figure 1. Plot of the estimated GE, by GDF and CV, as a function of tuning parameter C in Example 1. The solid line represents optimal approximate of true GE via CB, the dash line represents the estimated GE by GDF, and the dotted line represents the estimated GE by CV.

The estimated CGE for the combined classifier is given in (6). Minimization of (6) with respect to C yields $\hat{C}$, which is obtained via the same grid search as described in Section 4.1. Our final combined classifier is $\text{Sign}(\hat{f}_1 + \hat{C}\hat{f}_2)$, with $\hat{C}$ is the minimizer of (6).

To examine effectiveness of the proposed combining strategy, we perform a simulation study in the first four binary data examples. First, linear SVM and Gaussian kernel SVM are trained with the optimal tuning parameters, obtained via minimizing (6). Second, C is estimated respectively via GDF and CB. The testing errors averaged over 100 simulation replications are used to evaluate the performance, which are summarized in Table 5.

Evidently, the testing error of the combined classifier via GDF is closer to that via CB, and smaller than that via any individual SVM classifier in Example 1 and the WBC example, whereas it approximates the best individual testing error in Example 2 and the diabetes example. This empirical result shows that our proposed technique achieves the goal of combining two classifiers to outperform their component classifiers, or at least to do no worse than the best individual one.

Table 5. Average testing errors as well as the estimated standard errors (in parenthesis). Here the combined classifier combines two SVM with the optimal weights, estimated by GDF and CB respectively, where each individual SVM is trained with the optimal tuning parameter.

Dataset	SVM_G	SVM_L	Combined via GDF	Combined via CB
Example1	0.160(0.0026)	0.153(0.0030)	0.148(0.0028)	0.141(0.0025)
Example2	0.336(0.0019)	0.345(0.0019)	0.336(0.0023)	0.334(0.0010)
WBC	0.033(0.0006)	0.033(0.0008)	0.032(0.0007)	0.029(0.0006)
Diabetes	0.242(0.0019)	0.231(0.0016)	0.232(0.0016)	0.224(0.0014)

5. Summary and Discussions

This article studied a number of issues in estimating GE as well as its applications in tuning and combining. In contrast to most statistical methodologies, assuming X-fixed designs and conditioning on X, we introduced a new framework of estimation of GE and a technique of data perturbation, applicable to both X-fixed and X-random designs. This framework permits more accurate and efficient evaluation of any classifiers, binary or multicategory, margin-based or likelihood-based.

An application of our framework to 1-norm SVM (Bradley and Mangasarian (2000)) is also straightforward, which permits feature selection and classification simultaneously.

Acknowledgement

This research is supported by NSF grant IIS-0328802. We thank the reviewers for helpful comments and suggestions.

Appendix

Proof of Theorem 1. Note that $E(GE(\hat{\phi}_C) - (EGE(\hat{\phi}_C) + \lambda(X^n, \hat{\phi}_C)))^2 = E((GE(\hat{\phi}_C) - EGE(\hat{\phi}_C)) - \lambda(X^n, \hat{\phi}_C))^2$. Minimizing this with respect to λ yields $\lambda(X^n, \hat{\phi}_C) = E(GE(\hat{\phi}_C)|X^n) - E(EGE(\hat{\phi}_C)|X^n)$, which can be simplified to

$$
\begin{aligned}
&E(E(Y - \hat{\phi}_C(X))^2|X^n) - E(n^{-1}\sum_{i=1}^{n}(Y_i - \hat{\phi}_C(X_i))^2|X^n) \\
&= E(E(Y - E(Y|X))^2|X^n) + E(E(E(Y|X) - \hat{\phi}_C(X))^2|X^n) \\
&\quad - E(n^{-1}\sum_{i=1}^{n}(Y_i - E(Y_i|X_i))^2|X^n) - E(n^{-1}\sum_{i=1}^{n}(E(Y_i|X_i) - \hat{\phi}_C(X_i))^2|X^n) \\
&\quad - E(2n^{-1}\sum_{i=1}^{n}(Y_i - E(Y_i|X_i))(E(Y_i|X_i) - \hat{\phi}_C(X_i))|X^n),
\end{aligned}
$$

which yields the desired result in Theorem 1.

Proof of Lemma 1. It suffices to prove the last equation. It is easy to see that $\{Z \neq \hat{\phi}_C^t(X)\} = \bigcup_{j=0}^{k-1}\{Z^{(j)} \neq \hat{\phi}_{Cj}^t(X)\}$. Let $A_j = \{Z^{(j)} \neq \hat{\phi}_{Cj}^t(X)\}$. By the generating scheme of Z and $\hat{\phi}_C^t$, the intersection of any three or more A_j's is empty. Therefore, we have

$$\begin{aligned} P\{Z \neq \hat{\phi}_C^t(X)\} &= P\Big(\bigcup_{j=0}^{k-1}\{Z^{(j)} \neq \hat{\phi}_{Cj}^t(X)\}\Big) = P\Big(\bigcup_{j=0}^{k-1} A_j\Big) \\ &= \sum_{j=0}^{k-1} P(A_j) - \sum_{i\neq j} P(A_i \cap A_j) \\ &= \sum_{j=0}^{k-1} P(A_j) - \frac{1}{2}\sum_{j=0}^{k-1} P(A_j)\sum_{i\neq j} P(A_i|A_j). \end{aligned}$$

Note that $\{A_i|i = 0,\ldots,k-1; i \neq j\}$ are exhaustive and mutually exclusive, given A_j. Thus, $\sum_{i\neq j} P(A_i|A_j) = P(\bigcup_{i\neq j} A_i|A_j) = 1$; $0 \leq j \leq k-1$, and the result follows.

Lemma 2. *Let $V(Y_i, p_i(X_i))$ be $p_i(X_i)$ when $Y_i = 0$, and 0 when $Y_i = 1$. Then $E(V(Y_i, p_i(X_i))|X^n) = \mathrm{Var}\,(Y_i|X_i)$ and $\mathrm{Cov}\,(Y_i, \hat{\phi}_C(X_i)|X^n) = E(V(Y_i, p_i(X_i))\frac{\partial}{\partial Y_i}\hat{\phi}_C(X_i)|X^n) = \mathrm{Var}\,(Y_i|X_i)\,\frac{\partial}{\partial Y_i}\hat{\phi}_C(X_i)\Big|_{Y_i=0}$, where*

$$\frac{\partial}{\partial Y_i}\hat{\phi}_C(X_i) = \begin{cases} \hat{\phi}_C(X_i)\Big|_{Y_i=1} - \hat{\phi}_C(X_i)\Big|_{Y_i=0}, & \text{if } Y_i = 0, \\ 0, & \text{otherwise.} \end{cases}$$

Assume that $\hat{p}_i(X_i) \to p_i(X_i)$ as $n \to \infty$, a.s. Then $\hat{p}_i(X_i^*) \to p_i(X_i)$ as $n \to \infty$, $\tau \to 0^+$. Furthermore,

$$\lim_{n\to\infty}\lim_{\tau\to 0^+} E\Big(\Big(\frac{1}{K(Y_i, \hat{p}_i(X_i^*))} - \frac{V(Y_i, p_i(X_i))}{\mathrm{Var}^*(Y_i^*)}\Big)\mathrm{Cov}^*\big(Y_i^*, \hat{\phi}_C^*(X_i^*)|X^{*n}\big)\Big) = 0. \tag{19}$$

Proof of Lemma 2. We only prove (19). The proof for other parts is straightforward, and is omitted. By the definition of $V(Y_i, p_i)$ and (C.1), the left hand side of (19) becomes, after exchanging limit with expectation,

$$\lim_{n\to\infty}\lim_{\tau\to 0^+} E\Big(\Big(\frac{\mathrm{Var}^*(Y_i^*)}{K(Y_i, \hat{p}_i(X_i^*))} - V(Y_i, p_i(X_i))\Big)\frac{\mathrm{Cov}^*\big(Y_i^*, \hat{\phi}_C^*(X_i^*)|X^{*n}\big)}{\mathrm{Var}^*(Y_i^*)}\Big)$$

$$= \lim_{n\to\infty}\lim_{\tau\to 0^+} E\Big(\Big(\frac{\mathrm{Var}^*(Y_i^*)}{K(Y_i,\hat{p}_i(X_i^*))} - V(Y_i,p_i(X_i))\Big)\frac{\partial}{\partial Y_i^*}\,\hat{\phi}_C^*(X_i^*)\Big|_{Y_i^*=0}\Big)$$

$$= \lim_{n\to\infty}\lim_{\tau\to 0^+} E\Big(E\Big(\,\frac{\mathrm{Var}^*(Y_i^*)}{K(Y_i,\hat{p}_i(X_i^*))} - V(Y_i,p_i(X_i))\Big|\,(X^n,X^{*n})\Big)$$

$$\frac{\partial}{\partial Y_i^*}\,\hat{\phi}_C^*(X_i^*)\Big|_{Y_i^*=0}\Big)$$

$$= \lim_{n\to\infty}\lim_{\tau\to 0^+} E\Big(\Big(E\Big(\,\frac{\mathrm{Var}^*(Y_i^*)}{K(Y_i,\hat{p}_i(X_i^*))}\Big|\,(X^n,X^{*n})\Big) - p_i(X_i)(1-p_i(X_i))\Big)$$

$$\frac{\partial}{\partial Y_i^*}\hat{\phi}_C^*(X_i^*)\Big|_{Y_i^*=0}\Big).$$

Note that $K(1,\hat{p}_i(X_i^*)) = \tau(1-\tau+\tau\hat{p}_i(X_i^*))/\hat{p}_i(X_i^*)$ and $K(0,\hat{p}_i(X_i^*)) = \tau(1-\tau\hat{p}_i(X_i^*))/[1-\hat{p}_i(X_i^*)]$, while $\mathrm{Var}^*(Y_i^*)|_{Y_i=1} = (1-\tau+\tau\hat{p}_i(X_i^*))(\tau-\tau\hat{p}_i(X_i^*))$ and $\mathrm{Var}^*(Y_i^*)|_{Y_i=0} = (1-\tau\hat{p}_i(X_i^*))\tau\hat{p}_i(X_i^*)$. It thus can be verified that

$$\lim_{n\to\infty}\lim_{\tau\to 0^+} E\Big(\,\frac{\mathrm{Var}^*(Y_i^*)}{K(Y_i,\hat{p}_i(X_i^*))}\Big|\,(X^n,X^{*n})\Big) = p_i(X_i)(1-p_i(X_i)).$$

The result in (19) then follows.

Proof of Theorem 2. By Lemma 1, we only need to prove the binary case. Note that when $k=2$, (15) reduces to (6), and D_{2n} is independent of $\hat{\phi}_C$. It is sufficient to show that

$$\lim_{n\to\infty}\left(\lim_{\tau\to 0^+} E(CGE(\hat{\phi}_{\hat{C}}))/E(CGE(\hat{\phi}_{\tilde{C}}))\right) \le 1, \tag{20}$$

where $\tilde{C}$ is any estimate of C. For any C, we have

$$\begin{aligned} E(CGE(\hat{\phi}_C)) &= E(EGE(\hat{\phi}_C)) + \lambda_o(X^n,\hat{\phi}_C) \\ &= E(EGE(\hat{\phi}_C) + \hat{\lambda}(X^n,\hat{\phi}_C) + (\lambda_o(X^n,\hat{\phi}_C) - \hat{\lambda}(X^n,\hat{\phi}_C))). \end{aligned}$$

For any estimator $\tilde{C}$, $EGE(\hat{\phi}_{\tilde{C}}) + \hat{\lambda}(X^n,\hat{\phi}_{\tilde{C}}) \ge EGE(\hat{\phi}_{\hat{C}}) + \hat{\lambda}(X^n,\hat{\phi}_{\hat{C}})$. For (20), it suffices to prove that $\lim_{n\to\infty}\lim_{\tau\to 0^+} E[\lambda_o(X^n,\hat{\phi}_C) - \hat{\lambda}(X^n,\hat{\phi}_C)] = 0$ for any given C. By (C.1), it follows from the Dominated Convergence Theorem that we may interchange the limits and expectation in the following derivation. Assumption

(C.3) together with Lemma 2 ensures that we may work on $V(Y_i, p_i)$. Then

$$
\begin{aligned}
&\lim_{n\to\infty}\lim_{\tau\to 0^+} E\Big(2n^{-1}\sum_{i=1}^{n}\widehat{\mathrm{Cov}}\,(Y_i,\hat\phi_C(X_i^*)|X^{*n})\Big)\\
&=\lim_{n\to\infty} 2n^{-1}\sum_{i=1}^{n}\lim_{\tau\to 0^+} E\Big(\frac{V(Y_i,p_i(X_i))}{\mathrm{Var}^*(Y_i^*)}\mathrm{Cov}^*\big(Y_i^*,\hat\phi_C^*(X_i^*)|X^{*n}\big)\Big)\\
&=\lim_{n\to\infty} 2n^{-1}\sum_{i=1}^{n} E\Big(V(Y_i,p_i(X_i))E^*\lim_{\tau\to 0^+}\frac{\partial}{\partial Y_i^*}\hat\phi_C^*(X_i^*)\frac{V^*(Y_i^*,p_i^*(X_i^*))}{\mathrm{Var}^*(Y_i^*)}\Big)\\
&=\lim_{n\to\infty} 2n^{-1}\sum_{i=1}^{n} E\Big(V(Y_i,p_i(X_i))\frac{\partial}{\partial Y_i}\hat\phi_C(X_i)E^*\frac{V^*(Y_i^*,p_i^*(X_i^*))}{\mathrm{Var}^*(Y_i^*)}\Big)\\
&=\lim_{n\to\infty} 2n^{-1}\sum_{i=1}^{n} E(\mathrm{Cov}\,(Y_i,\hat\phi_C(X_i)|X^n))\\
&=\lim_{n\to\infty} E(2n^{-1}\sum_{i=1}^{n}\mathrm{Cov}\,(Y_i,\hat\phi_C(X_i)|X^n)).
\end{aligned}
$$

Furthermore, the distribution of (X_i^*, Y_i^*) tends to the distribution of (X_i, Y_i) as $\tau\to 0$, therefore, $\lim_{n\to\infty}\lim_{\tau\to 0^+} E(\widehat{D}_{1n}(X^n,\hat\phi_C))$ is

$$
\begin{aligned}
&\lim_{n\to\infty}\lim_{\tau\to 0^+} E\Big(E^*\Big(\frac{1}{n}\sum_{i=1}^{n}(\hat p_i(X_i)-\hat\phi_C^*(X_i))^2-\frac{1}{n}\sum_{i=1}^{n}(\hat p_i^*(X_i^*)-\hat\phi_C^*(X_i^*))^2\,\Big|\,X^{*n}\Big)\Big)\\
&=\lim_{n\to\infty} E\Big(\lim_{\tau\to 0^+} E^*\Big(\frac{1}{n}\sum_{i=1}^{n}(\hat p_i(X_i)-\hat\phi_C^*(X_i))^2-E(E(Y|X)-\hat\phi_C^*(X))^2\ |X^{*n}\Big)\Big)\\
&\quad+\lim_{n\to\infty} E\Big(\lim_{\tau\to 0^+} E^*\Big(E(E(Y|X)-\hat\phi_C^*(X))^2-\frac{1}{n}\sum_{i=1}^{n}(\hat p_i^*(X_i^*)-\hat\phi_C^*(X_i^*))^2\,\Big|\,X^{*n}\Big)\Big)\\
&=\lim_{n\to\infty} E\Big(E\Big(\frac{1}{n}\sum_{i=1}^{n}(\hat p_i(X_i)-\hat\phi_C(X_i))^2-E(E(Y|X)-\hat\phi_C(X))^2\,\Big|\,X^n\Big)\Big)\\
&\quad+\lim_{n\to\infty} E\Big(E\Big(E(E(Y|X)-\hat\phi_C(X))^2-\frac{1}{n}\sum_{i=1}^{n}(\hat p_i(X_i)-\hat\phi_C(X_i))^2\,\Big|\,X^n\Big)\Big)\\
&=\lim_{n\to\infty} E\Big(E\Big(E(E(Y|X)-\hat\phi_C(X))^2-\frac{1}{n}\sum_{i=1}^{n}(E(Y_i|X_i)-\hat\phi_C(X_i))^2\,\Big|\,X^n\Big)\Big)\\
&=\lim_{n\to\infty} E(D_{1n}(X^n,\hat\phi_C)),
\end{aligned}
$$

where the second to last equality follows from (C.3) and the one-sided Uniform Convergence Theorem in Vapnik and Chervonenkis (1991), by noting that

$0 \le E(E(Y|X) - \hat{\phi}_C(X))^2 \le 4$. Combining the above two equalities, we have $\lim_{n\to\infty} \lim_{\tau\to 0^+} E(\hat{\lambda}(X^n, \hat{\phi}_C)) = \lim_{n\to\infty} E(\lambda_o(X^n, \hat{\phi}_C))$. Therefore,

$$\begin{aligned}
&\lim_{n\to\infty} \lim_{\tau\to 0^+} E(CGE(\hat{\phi}_{\hat{C}})) \\
&= \lim_{n\to\infty} \lim_{\tau\to 0^+} E\left(EGE(\hat{\phi}_{\hat{C}}) + \hat{\lambda}(X^n, \hat{\phi}_{\hat{C}}) + (\lambda_o(X^n, \hat{\phi}_{\hat{C}}) - \hat{\lambda}(X^n, \hat{\phi}_{\hat{C}}))\right) \\
&= \lim_{n\to\infty} \lim_{\tau\to 0^+} E(EGE(\hat{\phi}_{\hat{C}}) + \hat{\lambda}(X^n, \hat{\phi}_{\hat{C}})) \\
&\le \lim_{n\to\infty} \lim_{\tau\to 0^+} E(EGE(\hat{\phi}_{\tilde{C}}) + \hat{\lambda}(X^n, \hat{\phi}_{\tilde{C}})) \\
&= \lim_{n\to\infty} \lim_{\tau\to 0^+} E\left(EGE(\hat{\phi}_{\tilde{C}}) + \hat{\lambda}(X^n, \hat{\phi}_{\tilde{C}}) + (\lambda_o(X^n, \hat{\phi}_{\tilde{C}}) - \hat{\lambda}(X^n, \hat{\phi}_{\tilde{C}}))\right) \\
&= \lim_{n\to\infty} E(CGE(\hat{\phi}_{\tilde{C}})).
\end{aligned}$$

The result then follows.

References

Blake, C. L. and Merz, C. J. (1998). UCI Repository of machine learning databases. http:// www.ics.uci.edu/ mlearn/MLRepository.html. University of California, Irvine, Department of Information and Computer Science.

Bradley, P. S. and Mangasarian, O. L. (2000). Massive data discrimination via linear support vector machines. *Optimization Methods and Software* **13**, 1-10.

Breiman, L. (1992). The little bootstrap and other methods for dimensionality selection in regression: X-fixed prediction error. *J. Amer. Statist. Assoc.* **87**, 738-754.

Breiman, L. and Spector, P. (1992). Submodel selection and evaluation in regression – the X - Random case. *Internat. Rev. Statist.* **3**, 291-319.

Breiman, L. (1996). Heuristics of instability and stabilization in model selection. *Ann. Statist.* **24**, 2350-2383.

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

Devroye L, Gyorfi L. and Lugosi G. (1996). *A probabilistic theory of pattern recognition*, Springer-Verlag, New York.

Denby, L., Landwehr, J. M. and Mallows, C. L. (2004). Discussion of Efron's paper entitled "The estimation of prediction error: covariance penalties and Cross-Validation". *J. Amer. Statist. Assoc.* **99**, 633-634.

Efron, B. (1983). Estimating the error rate of a prediction rule: improvement on cross-validation. *J. Amer. Statist. Assoc.* **78**, 316-331.

Efron, B. (1986). How biased is the apparent error rate of a prediction rule?. *J. Amer. Statist. Assoc.* **81**, 461-470.

Efron, B. (2004). The estimation of prediction error: covariance penalties and cross-validation. *J. Amer. Statist. Assoc.* **99**, 619-632.

Hastie, T., Rosset, S., Tibshirani, R. and Zhu, J. (2004). The entire regularization path for the support vector machine. *J. of Machine Learning Research* **5**, 1391-1415.

Lee, Y., Lin, Y. and Wahba, G. (2003). Multicategory support vector machines, theory and application to the classification of microarray data and satellite radiance data. *J. Amer. Statist. Assoc.* **99**, 67-81.

Li, K. C. (1987). Asymptotic optimality for C_p, C_L, cross-validation and generalized cross-validation: Discrete index set. *Ann. Statist.* **15**, 958-975.

Shen, X. and Huang, H.-C. (2005). Optimal model assessment, selection and combination. *J. Amer. Statist. Assoc.*, to appear.

Shen, X., Huang, H.-C. and Ye, J. (2004). Adaptive model selection and assessment for exponential family distributions. *Technometrics* **46**, 306-317.

Shen, X., Tseng, G. C. Zhang, X. and Wong, W. H. (2003). On Psi-learning. *J. Amer. Statist. Assoc.* **98**, 724-734.

Vapnik, V. and Chervonenkis, A. (1991). The necessary and sufficient conditions for consistency of the method of empirical risk minimization. *Pattern Recogn. Image Anal.* **1**, 284-305.

School of Statistics, University of Minnesota, Minneapolis, MN 55455, U.S.A.

E-mail: wangjh@stat.umn.edu

School of Statistics, University of Minnesota, Minneapolis, MN 55455, U.S.A.

E-mail: xshen@stat.umn.edu

(Received March 2005; accepted July 2005)