Performance of Error Estimators for Classification

Edward R. Dougherty,[1,2,3,*] Chao Sima,[2] Jianping Hua,[2] Blaise Hanczar[4] and Ulisses M. Braga-Neto[1]

[1]*Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA*

[2]*Computational Biology Division, Translational Genomics Research Institute, Phoenix, AZ, USA*

[3]*Department of Pathology, University of Texas M. D. Anderson Cancer Center, Houston, TX, USA*

[4]*LIPADE, Universite Paris Descartes, Paris, France*

Abstract: Classification in bioinformatics often suffers from small samples in conjunction with large numbers of features, which makes error estimation problematic. When a sample is small, there is insufficient data to split the sample and the same data are used for both classifier design and error estimation. Error estimation can suffer from high variance, bias, or both. The problem of choosing a suitable error estimator is exacerbated by the fact that estimation performance depends on the rule used to design the classifier, the feature-label distribution to which the classifier is to be applied, and the sample size. This paper reviews the performance of training-sample error estimators with respect to several criteria: estimation accuracy, variance, bias, correlation with the true error, regression on the true error, and accuracy in ranking feature sets. A number of error estimators are considered: resubstitution, leave-one-out cross-validation, 10-fold cross-validation, bolstered resubstitution, semi-bolstered resubstitution, .632 bootstrap, .632+ bootstrap, and optimal bootstrap. It illustrates these performance criteria for certain models and for two real data sets, referring to the literature for more extensive applications of these criteria. The results given in the present paper are consistent with those in the literature and lead to two conclusions: (1) much greater effort needs to be focused on error estimation, and (2) owing to the generally poor performance of error estimators on small samples, for a conclusion based on a small-sample error estimator to be considered valid, it should be supported by evidence that the estimator in question can be expected to perform sufficiently well under the circumstances to justify the conclusion.

Keywords: Classification, epistemology, error estimation, validity.

INTRODUCTION

Classification plays a key role in bioinformatics – for instance, in the genomic or proteomic discrimination between phenotypes. Error estimation is critical to classification because the validity of the resulting classifier model, composed of the classifier and its error estimate, is based on the accuracy of the error estimation procedure [1]. Given a large set of sample data, the data can be split between training and test data, with a classifier being designed on the training data and its error being estimated on the test data. The downside in splitting the data is that there are less data available for design, thereby hurting the design process. This negative impact is negligible when there is an abundance of data, but can be significant when samples are small. In this paper our focus is on using the same data for training and testing.

Since it is impossible to know the accuracy of a particular error estimate for a specific sample, estimation quality is judged based on the properties of the estimation procedure. Performance can be judged in various ways. We consider error-estimation performance relative to accuracy, correlation with the true error, regression between the true and estimated errors, conditional bounds on the true error, the ability to rank feature sets, and the manner in which error estimation affects feature selection when used as an estimation procedure within a feature-selection algorithm. Performance of an error estimator depends in an interactive way on the distribution of the features and labels, the classification rule used to design the classifier, and the sample size.

We assume that both training and testing are performed on a random sample $S = \{(\mathbf{X}_1, Y_1), (\mathbf{X}_2, Y_2),\ldots, (\mathbf{X}_n, Y_n)\}$ of feature-label pairs drawn from a feature-label distribution $\mathbf{F}$ defined over $\Re^d \times \{0, 1\}$. $\mathbf{X}_i$ is a d-dimensional vector and Y_i is a binary label. There is a classification rule Ψ defined on the space of random samples, so that $\psi = \Psi(S)$ is a binary-valued function defined on $\Re^d$. Given the class-conditional distributions, $\mathbf{F}|0$ and $\mathbf{F}|1$, the classifier ψ defines a decision procedure on $\Re^d$. Its error as a discriminator between $\mathbf{F}|0$ and $\mathbf{F}|1$ is given by $\varepsilon_{\mathbf{F}}[\psi] = P(\psi(\mathbf{X}) \neq Y)$, the probability being with respect to $\mathbf{F}$. To emphasize the fact that ψ is a function of Ψ and S, this error can be denoted as $\varepsilon_{\Psi,\mathbf{F}}[S]$, where S is a random set. In this sense, $\varepsilon_{\Psi,\mathbf{F}}[S]$ is the error for the classification rule Ψ on the feature-label distribution $\mathbf{F}$ for the random sample S. The error is estimated via the sample S by an estimation rule Ξ. $\Psi(S)$ is a classifier and $\hat{\varepsilon}_{\Xi,\mathbf{F}}[S] = \Xi(S)$ is an estimate of the error of $\Psi(S)$. For notational simplicity, we will typically write ε_n and $\hat{\varepsilon}_n$ in place of $\varepsilon_{\Psi,\mathbf{F}}[S]$ and $\hat{\varepsilon}_{\Xi,\mathbf{F}}[S]$, respectively, where n is the size of the sample.

The performance of an error estimator concerns the relation between the true error and the estimate. The full probabilistic relation is characterized by the joint distribution of the random vector $(\varepsilon_n, \hat{\varepsilon}_n)$ of the true and estimated errors. Certain moments of this joint distribution play key roles: the expected values, $E[\varepsilon_n]$ and $E[\hat{\varepsilon}_n]$, of the true and estimated

*Address correspondence to this author at the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843-3128, USA; Tel: 979-862-8896; Fax: 979-845-6259;
Email: edward@mail.ece.tamu.edu

1574-8936/10 $55.00+.00

errors, the variances, $Var[\varepsilon_n]$ and $Var[\hat{\varepsilon}_n]$, of the true and estimated errors, and the covariance, $Cov[\varepsilon_n, \hat{\varepsilon}_n]$, between the errors. Besides being important in their own right, these moments are used to compute various quantities relating to the performance of an error estimator: the bias of the error estimator, the correlation between the true and estimated errors, the variance of the deviation between the estimated and true errors, and the root-mean-square (RMS) error for the estimated error as an approximation for the true error. These are given by

$$Bias[\hat{\varepsilon}_n] = E[\hat{\varepsilon}_n] - E[\varepsilon_n] \tag{1}$$

$$\rho[\varepsilon_n, \hat{\varepsilon}_n] = \frac{Cov[\varepsilon_n, \hat{\varepsilon}_n]}{\sqrt{Var[\varepsilon_n]Var[\hat{\varepsilon}_n]}} \tag{2}$$

$$Var_{dev}[\hat{\varepsilon}_n] = Var[\hat{\varepsilon}_n - \varepsilon_n] = Var[\hat{\varepsilon}_n] + Var[\varepsilon_n] - 2\rho[\varepsilon_n, \hat{\varepsilon}_n]\sqrt{Var[\hat{\varepsilon}_n]Var[\varepsilon_n]} \tag{3}$$

$$RMS[\hat{\varepsilon}_n] = \sqrt{E[|\hat{\varepsilon}_n - \varepsilon_n|^2]} = \sqrt{Var_{dev}[\hat{\varepsilon}_n] + Bias^2[\hat{\varepsilon}_n]} \tag{4}$$

respectively. The most important of these is the RMS, because it provides a direct measure of the difference between the estimated and true errors. The others appear within the computation of the RMS; indeed, all of the five basic moments, the expectations, variances, and covariance, appear within the RMS. In particular, note the role of the deviation $\hat{\varepsilon}_n - \varepsilon_n$. The RMS is composed of its variance and its mean, which is the bias of $\hat{\varepsilon}_n$. Hence, the RMS is determined by the centrality and dispersion of the deviation distribution. Observing the definitions, we see that estimation performance is enhanced by small bias, small deviation variance, and large correlation. It is in this light that these are considered as performance measures.

ERROR ESTIMATORS

We consider several error estimators.

Resubstitution

A simple approach is to estimate the error of a designed classifier ψ by applying ψ to the sample (which is being used for both training and testing). The *resubstitution estimate*, $\hat{\varepsilon}_n^{res}$, is the fraction of errors made by ψ on the sample. The resubstitution estimator is typically (but not always) low-biased, meaning $E[\hat{\varepsilon}_n^{res}] < E[\varepsilon_n]$, and this bias can be severe for small samples, depending on the complexity of the classification rule.

Bolstering

In resubstitution there is no distinction between points near and far from the decision boundary; the *bolstered-resubstitution* estimator is based on the heuristic that, relative to making an error, more confidence should be attributed to points far from the decision boundary than points near it [2]. This is achieved by placing a distribution, called a *bolstering kernel*, at each point. A key issue is the amount of bolstering (spread of the bolstering kernels), and a method has been proposed to compute this spread based on the data [2]. Fig. (**1**) illustrates the error for linear classification when the bolstering kernels are uniform circular distributions. By normalizing their total volume to 1, the collection of bolstering kernels form a probability density

$$\mathbf{f}^{\nabla}(\mathbf{x}, y) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{f}_i^{\nabla}(\mathbf{x} - \mathbf{x}_i) I_{y=y_i} \tag{5}$$

where $\mathbf{f}_i^{\nabla}(\mathbf{x} - \mathbf{x}_i)$ is the bolstering kernel at the sample point $\mathbf{x}_i$ and I_A is the indicator function, $I_A = 1$ if A is true and $I_A = 0$ otherwise. The bolstered resubstitution error estimate is obtained by integrating all bolstering kernels over their corresponding error regions (rather than simply counting the erroneously classified points as with resubstitution):

$$\hat{\varepsilon}_n^{bol} = \frac{1}{n}\sum_{i=1}^{n}\left(I_{y_i=0}\int_{\{\mathbf{x}:\psi(\mathbf{x})=1\}} \mathbf{f}_i^{\nabla}(\mathbf{x}-\mathbf{x}_i)\,d\mathbf{x} + I_{y=y_i}\int_{\{\mathbf{x}:\psi(\mathbf{x})=0\}} \mathbf{f}_i^{\nabla}(\mathbf{x}-\mathbf{x}_i)\,d\mathbf{x} \right) \tag{6}$$

When resubstitution is heavily low-biased, it may not be good to spread incorrectly classified data points because that increases the optimism of the error estimate (low bias). The *semi-bolstered-resubstitution* estimator does not bolster (spread) incorrectly classified points.

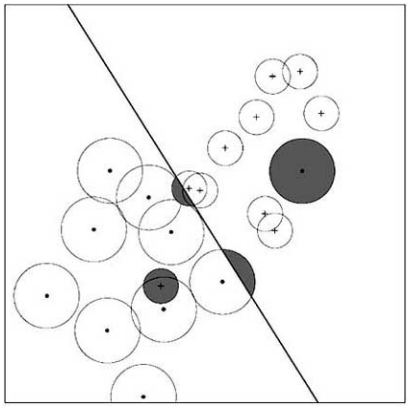


Fig. (1). Bolstering kernels for linear classification.

Cross-Validation

Cross-validation is a re-sampling strategy in which classifiers are designed from parts of the sample, each is tested on the remaining data, and ε_n is estimated by averaging the errors. In *k-fold cross-validation*, the sample S is partitioned into k folds $S_{(i)}$, for $i = 1, 2, \ldots, k$. Each fold is left out of the design process, a (*surrogate*) classifier $\psi_{(i)}$ is designed on $S - S_{(i)}$, the error of $\psi_{(i)}$ is estimated as the resubstitution error it makes on $S_{(i)}$, and the estimate, $\hat{\varepsilon}_n^{cv(k)}$, of ε_n is the average error committed on all folds. A k-fold cross-validation estimator is unbiased as an estimator of $E[\varepsilon_{n-n/k}]$, meaning $E[\hat{\varepsilon}_n^{cv(k)}] = E[\varepsilon_{n-n/k}]$, where $\varepsilon_{n-n/k}$ is the error arising from design on a sample of size $n - n/k$. The special case of n-fold cross-validation yields the *leave-one-out estimator*, $\hat{\varepsilon}_n^{loo}$, which is an unbiased estimator of $E[\varepsilon_{n-1}]$.

While not suffering from severe bias, cross-validation has large variance with small samples and therefore its use is problematic [3]. Consider leave-one-out cross-validation, for which $E[\hat{\varepsilon}_n^{loo}] = E[\varepsilon_{n-1}]$, so that $E[\hat{\varepsilon}_n^{loo} - \varepsilon_n] \approx 0$. Thus, the expected difference between the error estimator and the error is approximately 0. But our main concern is not with the expected difference between the true and estimated errors, but with $RMS[\hat{\varepsilon}_n^{loo}]$, which measures the precision of the error estimator. From Eq. 4 we see that, unless the cross-validation deviation variance is small, which it is not for small samples, the RMS will not be small.

A key difference between leave-one-out cross validation and leaving out $n/k > 1$ points is that with leaving out one point, all n folds are used in the computation, whereas with leaving out n/k points, there is a much greater number of possible folds and in practice usually only a random subset of these is chosen for cross-validation. Thus, as with resubstitution and bolstered resubstitution, given the sample, leave-one-out cross-validation is deterministic, but leave-(n/k)-out cross-validation is random. Thus, whereas, the expectation and variance, for leave-one-out cross-validation are relative to the set of all samples S, for random-fold cross-validation they are with respect to all samples S and all possible fold selections K. For expectation this means that

$$E[\hat{\varepsilon}_n^{cv(k)}] = E_S[E_K[\hat{\varepsilon}_n^{cv(k)} \mid S]] \tag{7}$$

where the subscripts S and K indicate with respect to which variable the expectation is taken. The error for a given sample S, or "sample error," is the conditional expectation, $E_K[\hat{\varepsilon}_n^{cv(k)} \mid S]$. In the case of the variance, the basic variance formula yields

$$VAR[\hat{\varepsilon}_n^{cv(k)}] = VAR_S[E_K[\hat{\varepsilon}_n^{cv(k)} \mid S]] + E_S[VAR_K[\hat{\varepsilon}_n^{cv(k)} \mid S]] \tag{8}$$

The variance of the random-fold estimator is decomposed into the variance over all samples of the sample error plus the expected value over all samples of the variance of the random-fold estimator given the sample. The first variance is "exterior" to any particular sample and the second is "interior" to the re-sampling procedure. In words, the variance of the random-fold cross-validation error estimator is equal to the *external variance* plus the expected value of the *internal variance*.

We close this subsection by noting that bolstering can be applied to cross-validation. For instance, *bolstered leave-one-out* estimation involves bolstering the error counting on the surrogate classifiers.

Bootstrap

A *bootstrap sample* consists of n equally-likely draws with replacement from the original sample S [4]. Some points may appear multiple times, whereas others may not appear at all. For the basic *zero bootstrap* estimator, $\hat{\varepsilon}_n^{boo}$, the classifier is designed on the bootstrap sample and tested on the points left out, this is done repeatedly for randomly drawn bootstrap samples, and the bootstrap estimate is the average error made on the left-out points. The basic bootstrap estimator tends to be high-biased because the number of points available for design is on average only $0.632n$. The *.632 bootstrap* estimator tries to correct this bias via a weighted average of $\hat{\varepsilon}_n^{boo}$ and resubstitution [5],

$$\hat{\varepsilon}_n^{b632} = 0.368\hat{\varepsilon}_n^{res} + 0.632\hat{\varepsilon}_n^{boo} \tag{9}$$

The .632 bootstrap estimator is a convex combination of the low-biased resubstitution estimator and the high-biased *zero bootstrap* estimator.

More generally, a *convex estimator* is of the form

$$\hat{\varepsilon}_n^{a,b} = a\hat{\varepsilon}_n^{(1)} + b\hat{\varepsilon}_n^{(2)} \tag{10}$$

where $\hat{\varepsilon}_n^{(1)}$ and $\hat{\varepsilon}_n^{(2)}$ are error estimators and convexity means that $a + b = 1$, with the weights a and b being nonnegative [6]. Its bias is given by

$$Bias[\hat{\varepsilon}_n^{a,b}] = aBias[\hat{\varepsilon}_n^{(1)}] + bBias[\hat{\varepsilon}_n^{(2)}] \tag{11}$$

If the weights are chosen so that the bias is eliminated (or approximately eliminated), then the bias term can be dropped from the RMS. Since the deviation of a convex estimator is given by

$$\hat{\varepsilon}_n^{a,b} - \varepsilon_n = a(\hat{\varepsilon}_n^{(1)} - \varepsilon_n) + b(\hat{\varepsilon}_n^{(2)} - \varepsilon_n) \tag{12}$$

the deviation variance satisfies

$$Var_{dev}[\hat{\varepsilon}_n^{a,b}] = a^2\, Var_{dev}[\hat{\varepsilon}_n^{(1)}] + b^2\, Var_{dev}[\hat{\varepsilon}_n^{(2)}] + 2ab\rho[\hat{\varepsilon}_n^{(1)}, \hat{\varepsilon}_n^{(2)}]\sqrt{Var_{dev}[\hat{\varepsilon}_n^{(1)}]Var_{dev}[\hat{\varepsilon}_n^{(2)}]}$$

$$\leq \max\left(Var_{dev}[\hat{\varepsilon}_n^{(1)}], Var_{dev}[\hat{\varepsilon}_n^{(2)}]\right), \tag{13}$$

[6]. A convex combination of a low-biased and high-biased estimator can mitigate both biases and this should be done using low-variance component estimators.

An optimal convex estimator results from finding weights a and b that minimize the RMS of a convex estimator. Our interest here is with the optimal bootstrap estimator, which results from optimizing a and b in Eq. 10 with $\hat{\varepsilon}_n^{(1)} = \hat{\varepsilon}_n^{res}$ and $\hat{\varepsilon}_n^{(2)} = \hat{\varepsilon}_n^{boo}$ to arrive at

$$\hat{\varepsilon}_n^{bopt} = w_{bopt}\hat{\varepsilon}_n^{res} + (1 - w_{bopt})\hat{\varepsilon}_n^{boo} \tag{14}$$

[6]. A practical difficulty with the optimal bootstrap estimator is that it requires the feature-label distribution to derive the optimal weights. Given that the feature-label distribution is unknown, the weights must be estimated based on either some prior assumptions or on prior assumptions in conjunction with the data. Nonetheless, for performance evaluation it provides a benchmark because the feature-label distribution is known and the weights can be derived explicitly.

An alternative to the optimal bootstrap is to take an adaptive approach to finding the weights. The .632+ estimator is of this kind [7]. The basic idea is to adjust the weight on the resubstitution estimator when resubstitution is exceptionally low-biased, which indicates strong overfitting. Letting the sample be $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$ and defining

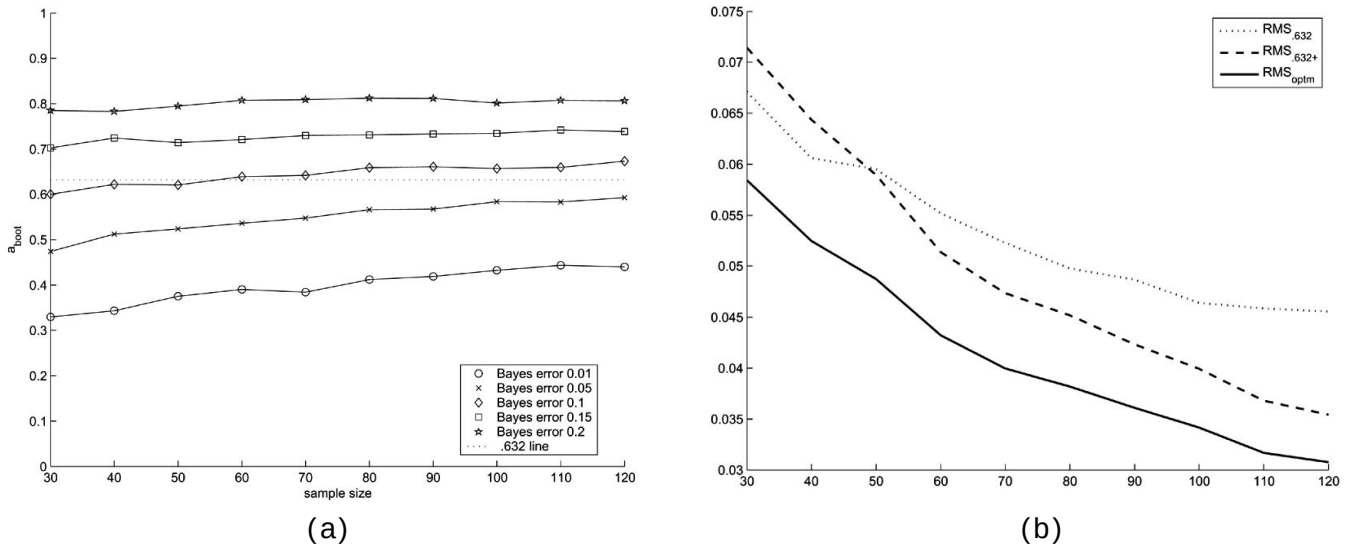


Fig. (2). Bootstrap performance as a function of sample size for 3NN classification: (**a**) optimal zero-bootstrap weight for different Bayes errors; (**b**) RMS for different bootstrap estimators.

$$\hat{\gamma} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} I_{y_i \neq \psi_n(\mathbf{x}_j)} \tag{15}$$

the relative overfitting rate is defined by

$$\hat{R} = \frac{\hat{\varepsilon}_n^{boo} - \hat{\varepsilon}_n^{res}}{\hat{\gamma} - \hat{\varepsilon}_n^{res}} \tag{16}$$

To be certain that $\hat{R} \in [0, 1]$, we set $\hat{R} = 1$ if $\hat{\varepsilon}_n^{boo} > \hat{\gamma}$ and set $\hat{R} = 0$ if $\hat{\varepsilon}_n^{boo} < \hat{\varepsilon}_n^{res}$ or $\hat{\gamma} < \hat{\varepsilon}_n^{res}$. $\hat{R}$ measures the degree of overfitting via the relative difference between the bootstrap and resubstitution estimates, a larger difference indicating more overfitting. $\hat{R}$ is "relative" because the difference is normalized by $\hat{\gamma} - \hat{\varepsilon}_n^{res}$. Although we shall not go into detail, $\hat{\gamma} - \hat{\varepsilon}_n^{res}$ approximately represents the most that resubstitution can be low-biased, so that usually $\hat{\varepsilon}_n^{boo} - \hat{\varepsilon}_n^{res} \leq \hat{\gamma} - \hat{\varepsilon}_n^{res}$. The weight

$$\hat{w} = \frac{0.632}{1 - 0.368\hat{R}} \tag{17}$$

replaces 0.632 in the original .632 bootstrap, except when $\hat{R} = 1$, in which case the .632+ bootstrap has the same form as the .632 bootstrap but with $\hat{\gamma}$ replacing $\hat{\varepsilon}_n^{res}$.

Fig. **2**(**a**) shows how far the weights of the optimal bootstrap can deviate from those of the .632 bootstrap. This is for a Gaussian model in which the class conditional densities have covariance matrices **I** and $(1.5)^2$**I** and the classification rule is 3-nearest-neighbor [6]. Part (b) of the figure shows the RMS errors for .632, .632+. and the optimal bootstrap for the same model for increasing sample size when the Bayes error is 0.2. Three points should be noticed: (1) the optimal bootstrap significantly outperforms the .632 bootstrap; (2) as the sample size increases, the .632+ bootstrap outperforms the .632 bootstrap but is still substantially inferior to the optimal bootstrap; and (3) for very small sample sizes, the .632 bootstrap actually outperforms the .632+ bootstrap. As with every performance analysis for error estimation, the results depend upon the feature-label distribution and the classification rule.

Finally, since bootstrap involves re-sampling internal to the sample, like random-fold cross-validation, computation of the expected bootstrap error and the variance must take this into account. In particular, the variance decomposes into external and internal variance.

Model

We make comparisons of estimator performance in several different settings using the following error estimators: resubstitution (resb), leave-one-out cross-validation (loo), 10-fold cross-validation (cv10), bolstered resubstitution (bresb), semi-bolstered resubstitution (sresb), .632 bootstrap (b632), .632+ bootstrap (b632+), and optimal bootstrap (bopt). Bolstering kernels are Gaussian and we use the method of [2] to find the spread of the kernels. We do not provide an exhaustive study across different classification rules and different feature-label distributions. For coherence, we focus on a single model and use two classification rules. We refer the reader to the cited references in each section for the relevant extensive studies.

We use a feature-label distribution consisting of two equally likely Gaussian class-conditional distributions with covariance matrices $\mathbf{K}_0$ and $\mathbf{K}_1$, with $\mathbf{K}_1 \neq \mathbf{K}_0$. The optimal classifier for this model has a quadratic decision boundary determined by a discriminant function, $d_k(\mathbf{x})$, point $\mathbf{x}$ being classified as $Y = 1$ if $d_1(\mathbf{x}) > d_0(\mathbf{x})$. While we use the unequal covariance model as the feature-label distribution, we apply the discriminant for the optimal classifier when $\mathbf{K}_1 = \mathbf{K}_0$, in which case the decision boundary would be linear and the optimal classifier is determined by

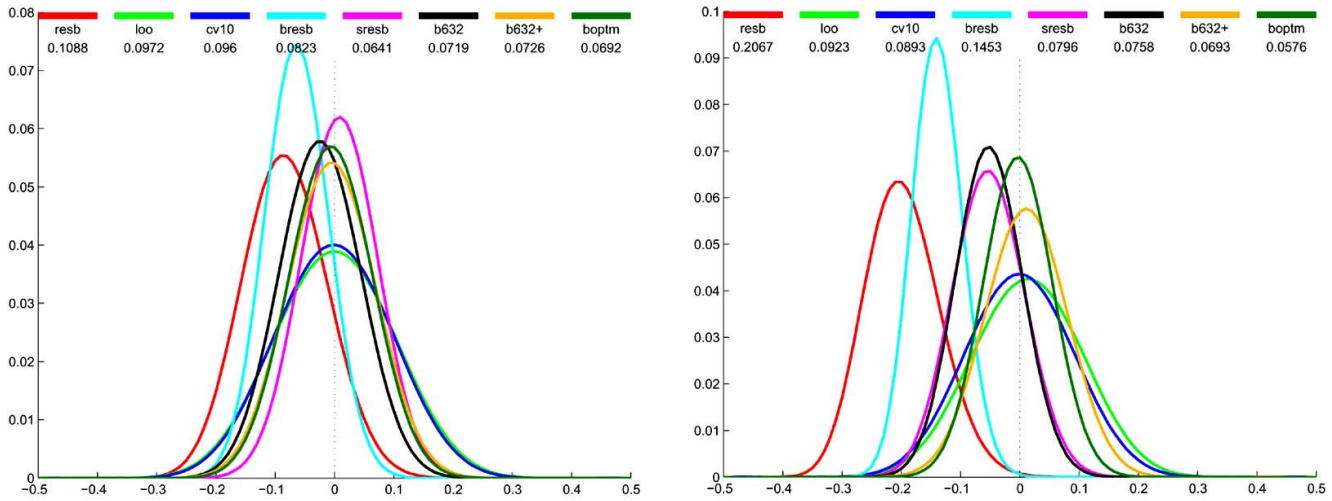


Fig. (3). Deviation distributions for error estimation methods in the quadratic model: (**a**) LDA classification rule; (**b**) 3NN classification rule.

$$d_k(\mathbf{x}) = -(\mathbf{x} - \mathbf{u}_k)'\mathbf{K}^{-1}(\mathbf{x} - \mathbf{u}_k) + 2\log f(k) \quad (18)$$

for $k = 0, 1$, which characterizes *linear discriminant analysis* (*LDA*). LDA is applied to sample data via the sample estimates of the means and covariance matrix. LDA can be applied to sample data even if the underlying feature-label distribution is not Gaussian; indeed, in practice, one rarely knows the feature-label distribution. Letting $\mathbf{I}$ denote the identity matrix, herein, class 0 has mean at $(0, 0,\ldots, 0)$ with $\mathbf{K}_0 = \sigma_0^2\mathbf{I}$, $\sigma_0 = 1$, and class 1 has mean at $(a_1, a_2,\ldots, a_D)$, where $a_1, a_2,\ldots, a_D$ have been sampled independently from a beta distribution, Beta(0.75, 2), with $\mathbf{K}_1 = \sigma_1^2\mathbf{I}$, $\sigma_1 = 1.5$. With these covariance matrices, the features are independent and can be ranked according to the values of $a_1, a_2,\ldots, a_D$, better features resulting from larger values.

In addition to LDA, we use the *k-nearest-neighbor* (*k*NN) *rule*, *k* odd, where the designed classifier ψ_n is defined for each $\mathbf{x} \in \Re^d$ by letting $\psi_n(\mathbf{x})$ be 0 or 1 according to which is the majority among the labels of the *k* points closest to $\mathbf{x}$. We use the popular choice $k = 3$.

When feature selection is used in classification, it is part of the classification rule and the entire set of potential features constitutes the feature set relative to the classification rule. Feature selection constrains the space of functions from which a classifier might be chosen, but it does not reduce the number of features in the design process. In this review we consider only a single feature-selection method, one that involves error estimation within the selection process, so that error estimators can be evaluated based on their impact to feature selection.

We briefly describe the *sequential forward floating selection* (*SFFS*) algorithm used in this study. To begin with, the *sequential forward selection* (*SFS*) algorithm starts with a small set of features, perhaps one, and iteratively builds the feature set. When there are *k* features, $x_1, x_2,\ldots, x_k$, in the growing feature set, all feature sets of the form $\{x_1, x_2,\ldots, x_k, w\}$ are compared and the best one chosen to form the feature set of size $k + 1$. The *SFS look-back* algorithm allows the deletion of features that may not perform well in combination with other features. For it, when there are *k* features, $x_1, x_2,\ldots, x_k$, in the growing feature set, all feature sets of the form $\{x_1, x_2,\ldots, x_k, w, z\}$ are compared and the best one chosen. Then all $(k + 1)$-element subsets are checked to allow the possibility of one of the earlier chosen features to be deleted, the result being the $k + 1$ features that will form the basis for the next stage of the algorithm. Flexibility is added with the SFFS algorithm, in which the number of features to be adjoined and deleted is not fixed [8].

PERFORMANCE ANALYSIS

This section describes various methodologies for performance analysis of error estimators, along with illustrative applications using the preceding models and methods.

DEVIATION DISTRIBUTION

Among the performance measures discussed in this paper, RMS is the most important, and RMS performance analysis is directly related to the deviation distribution [3]. Fig. (**3**) shows RMS values and beta-distribution fits (1000 points) for the deviation distributions arising from (a) LDA and (b) 3NN classification using 3 features with the *quadratic model*, $\sigma_0 = 1$ and $\sigma_1 = 1.5$, with sample size 40. A centered density indicates low bias and a narrow density indicates low variance. Notice the large dispersions for cross-validation. Note that bolstered resubstitution has the minimum deviation variance among the rules but that in both cases it is low-biased. Semi-bolstered resubstitution mitigates the low bias, even being slightly high-based in the LDA case. The correction is sufficiently good for LDA that semi-bolstered resubstitution has the minimum RMS among all the classification rules considered. For LDA, the .632+ bootstrap does not outperform the .632 bootstrap on account of increased variance, even though it is less biased. For 3NN, the .632+ bootstrap outperforms the .632 bootstrap, basically eliminating the low bias of the .632 bootstrap. This occurs even though the .632+ bootstrap shows increased variance in comparison to the .632 bootstrap. Both the .632 and .632+ bootstraps are significantly outperformed by the optimal bootstrap. Note the generally poor performance of cross-validation.

The preceding examples do not include feature selection. When there is feature selection, cross-validation and boot-

strap re-sampling must be carried out with respect to the full collection of features because feature selection is part of the classification rule and therefore must be incorporated within the error estimation procedure. In the case of *k*-fold cross-validation, when there is feature selection, given a sample S_D of size *n* from the feature-label distribution and having used feature selection to obtain a feature set and design a classifier ψ^d having *d* features for its arguments, the cross-validation estimate is computed in the following manner:

a) Randomly generate a cross-validation partition $S_{D,1},\ldots, S_{D,k}$ from from S_D, leaving out n/k points.

b) Using $S_D - S_{D,1}$, apply feature selection to find a feature set and classifier ψ^d.

c) Apply ψ^d to $S_{D,1}$, and let ε^1 be the error rate on $S_{D,1}$.

d) Repeat steps b and c for *k* times to obtain ε^2, $\varepsilon^3,\ldots, \varepsilon^k$.

e) The cross-validation estimate for ψ^d is $\hat{\varepsilon}_n^{cv(k)} = [\varepsilon^1 + \varepsilon^2 + \ldots + \varepsilon^k]/k$.

An analogous procedure is used to obtain the bootstrap estimate. As demonstrated in [3], and illustrated in Fig. (**3**), cross-validation suffers from a large deviation variance. The situation can be much worse when feature-selection is involved [9, 10]. As presently developed bolstering cannot handle a very large number of features because in such a situation the current method of choosing the kernel variances does not work. It can be used within a feature-selection algorithm but not to estimate the error of the final designed classifier on the final chosen feature set.

NONLINEAR REGRESSION

RMS, bias, and deviation variance are global quantities in that they relate to the full joint distribution of the true and error estimated errors. They do not relate directly to the knowledge obtained from a specific estimate. The knowledge derived from a specific estimate $\hat{\varepsilon}_n$ is embodied in the conditional distribution of the true error given the estimated error. In particular, the conditional expectation, $E[\varepsilon_n|\hat{\varepsilon}_n]$, called the *nonlinear regression* of the true error on the estimated error, gives the average value of the true error given the estimate $\hat{\varepsilon}_n$ and has been used for performance analysis [11]. It is the best mean-square-error estimate of the true error given the estimate. In practice, $E[\varepsilon_n|\hat{\varepsilon}_n]$ is not known; however, when evaluating the performance of an error estimator for a specific feature-label distribution, it is known. Since $E[\varepsilon_n|\hat{\varepsilon}_n]$ is the best mean-square estimator of the true error given the estimate, we would like to have $\hat{\varepsilon}_n \approx E[\varepsilon_n|\hat{\varepsilon}_n]$. This is because in practice we will use the estimate, not the conditional expectation of the true error given the estimate because we will not know the latter.

One can also consider the conditional distribution of the estimated error given the true error, in particular, the conditional expectation $E[\hat{\varepsilon}_n|\varepsilon_n]$. The accuracy of the conditional expectation as a representation of the overall conditional distribution depends on the conditional variance. Hence, the conditional variances $Var[\varepsilon_n|\hat{\varepsilon}_n]$ and $Var[\hat{\varepsilon}_n|\varepsilon_n]$ are also of interest.

CONDITIONAL BOUNDS

If one is concerned with having the true error below some tolerance given the estimate, then interest would center on finding a conditional bound for the true error, given the estimate. From this perspective, a way to evaluate an error estimator is to consider a $(1 - \alpha)$% interval of the form $[0, \tau]$ conditioned on the error estimate. Specifically, given the estimate $\hat{\varepsilon}_n$, we would like a bound $\tau(\alpha, \hat{\varepsilon}_n)$ on ε_n such that

$$P(\varepsilon_n < \tau(\alpha, \hat{\varepsilon}_n)|\hat{\varepsilon}_n) = 1 - \alpha \tag{19}$$

Since

$$P(\varepsilon_n < \tau(\alpha, \hat{\varepsilon}_n)|\hat{\varepsilon}_n) = \int_0^{\tau(\alpha,\hat{\varepsilon}_n)} dF(\varepsilon_n|\hat{\varepsilon}_n) \tag{20}$$

where $F(\varepsilon_n|\hat{\varepsilon}_n)$ is the conditional distribution for ε_n given $\hat{\varepsilon}_n$, determining the conditional bound depends on obtaining the required conditional distributions, which can be derived from the joint distribution of the true and estimated errors. It is rare to possess an analytic expression for either the nonlinear regression or the conditional bounds, and the problem of conditional bounds has been studied mostly via simulations [11], which is the approach we will take here.

Fig. (**4**) shows both the nonlinear regression of the true error on the estimated error and 0.95 conditional-bound lines for various error estimators for LDA, and the model used in this paper with sample size 60 and 5 features selected from 200 by the t-test (ignore erratic behavior at the curve ends resulting from very little data in that part of the distribution). Notice the lack of regression (flatness of the curves), especially for the cross-validation methods. Bolstered resubstitution and bootstrap show more regression. Fig. (**5**) shows the corresponding curves for sample size 100 and we see some increase in regression, especially for bolstering. Lack of regression, and therefore poor error estimation, is commonplace when samples are not large, especially for the randomized error estimators [12].

CORRELATION AND LINEAR REGRESSION

Since $\hat{\varepsilon}_n$ is used to estimate ε_n, we would like $\hat{\varepsilon}_n$ and ε_n to be strongly correlated. From the perspective of RMS, the desire for strong correlation is evident in Eq. 3, which shows that high correlation can help to mitigate the deviation variance caused by the individual variances. If the sample is large, then the individual variances tend to be small, so that the deviation variance is small; however, when the sample is small, the individual variances tend to be large, so that a large correlation is needed to offset these variances. Thus, the correlation between the true and estimated errors plays a vital role in assessing the goodness of the error estimator [12].

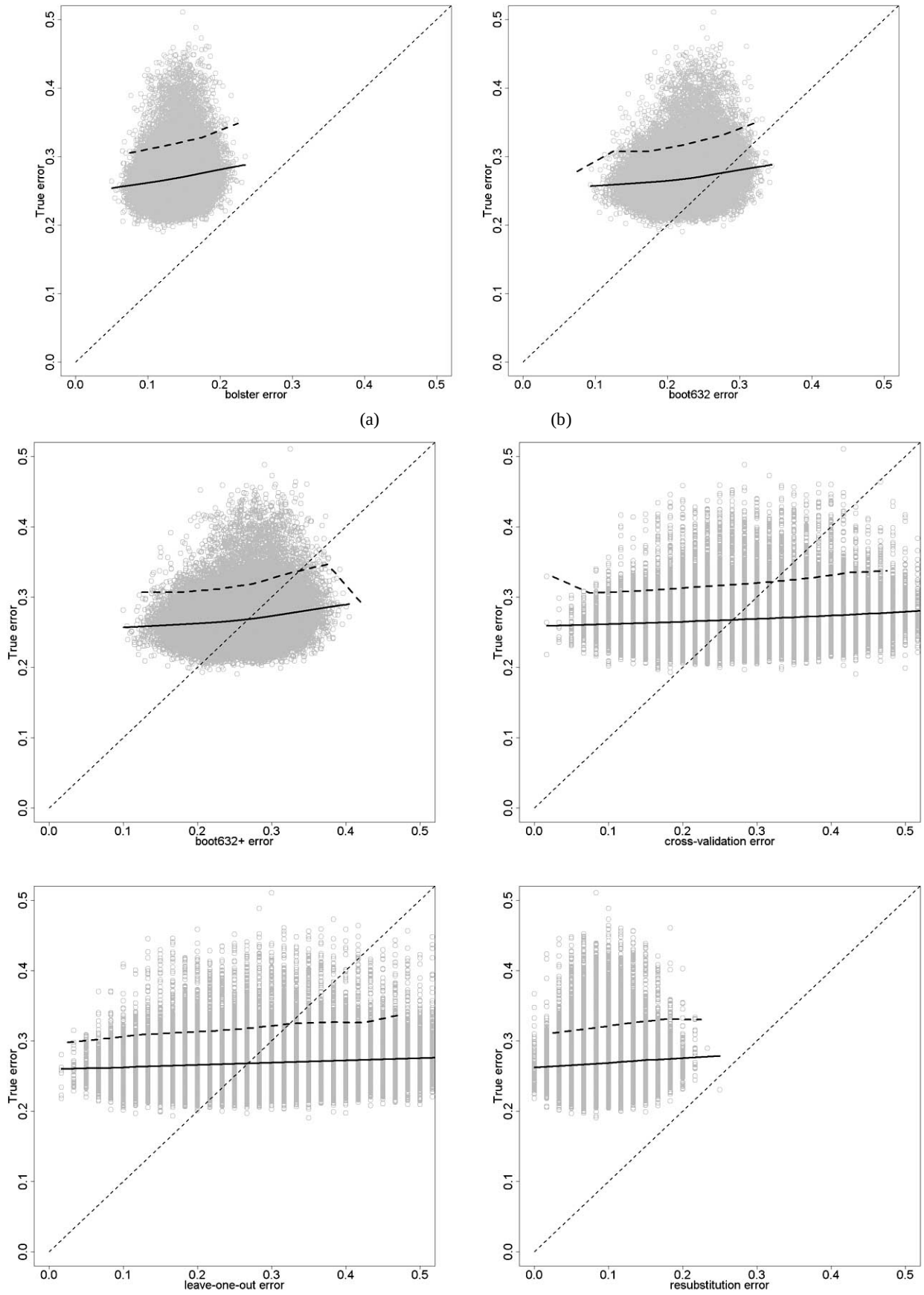
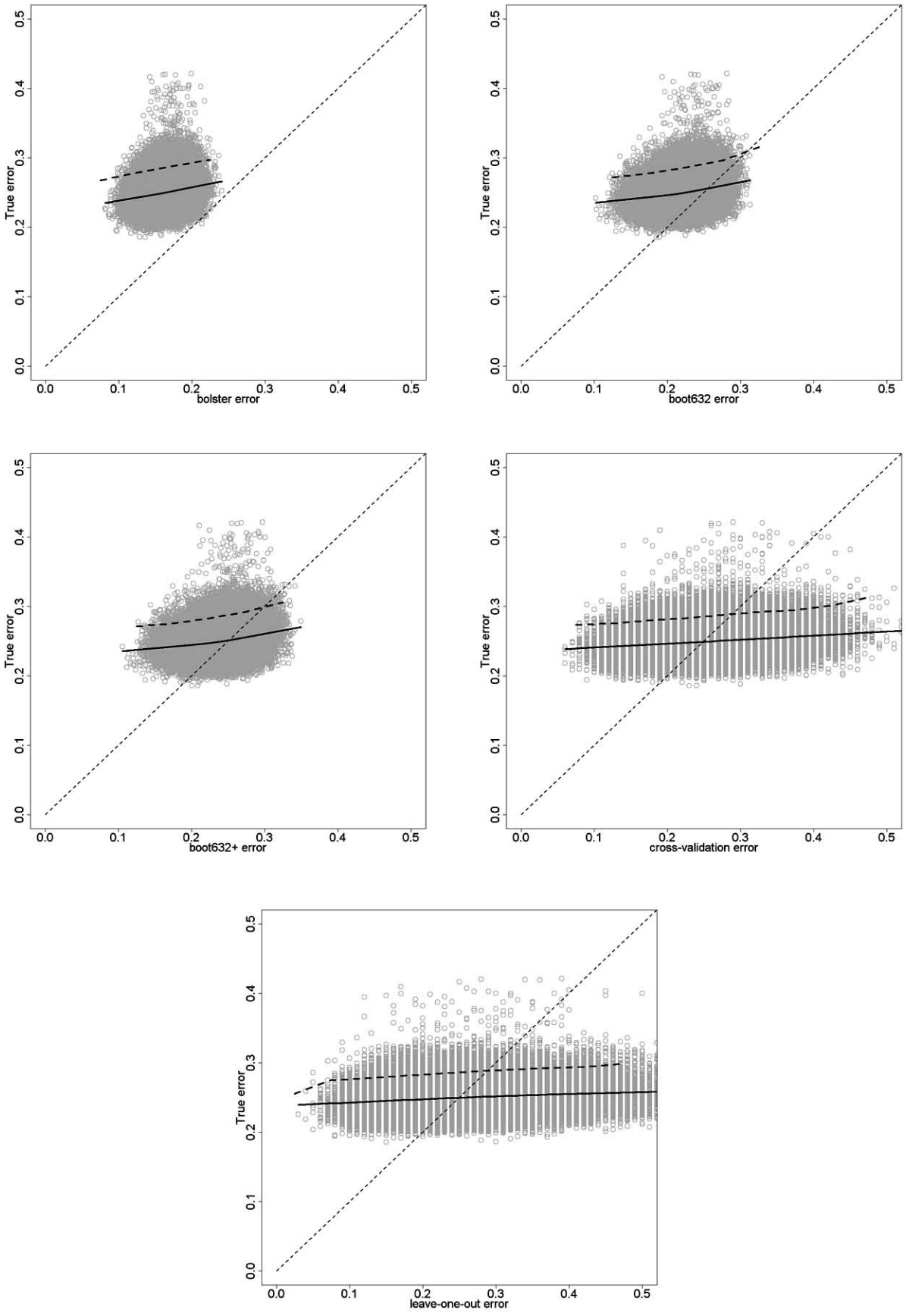


Fig. (4). Nonlinear regression (solid line) and 0.95 conditional bound (dashed line) for LDA with sample size 60: (**a**) bolstered resubstitution; (**b**) .632 bootstrap; (**c**) .632+ bootstrap; (**d**) 10-fold cross-validation; (**e**) leave-one-out cross-validation; (**f**) resubstitution.



(e)

Fig. (5). Nonlinear regression (solid line) and 0.95 conditional bound (dashed line) for LDA with sample size 100: (**a**) bolstered resubstitution; (**b**) .632 bootstrap; (**c**) .632+ bootstrap; (**d**) 10-fold cross-validation; (**e**) leave-one-out cross-validation; (**f**) resubstitution.

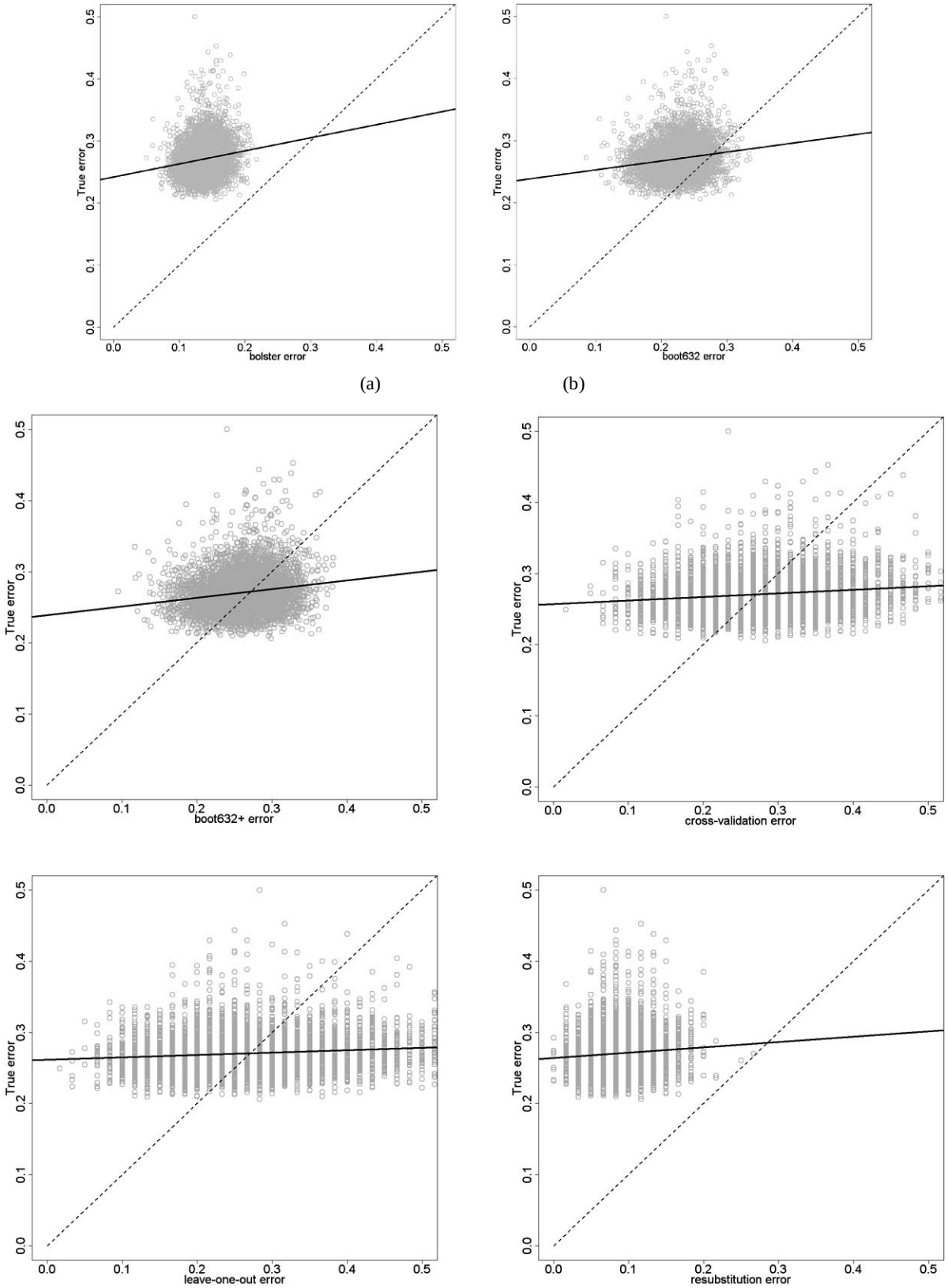


Fig. (6). Linear regression for LDA with sample size 60: (**a**) bolstered resubstitution; (**b**) .632 bootstrap; (**c**) .632+ bootstrap; (**d**) 10-fold cross-validation; (**e**) leave-one-out cross-validation; (**f**) resusbstitution.

The correlation is related to the linear regression between the true and estimated errors. Whereas nonlinear regression of the true error on the estimated error corresponds directly to the conditional expectation $E[\varepsilon_n|\hat{\varepsilon}_n]$, the linear regression

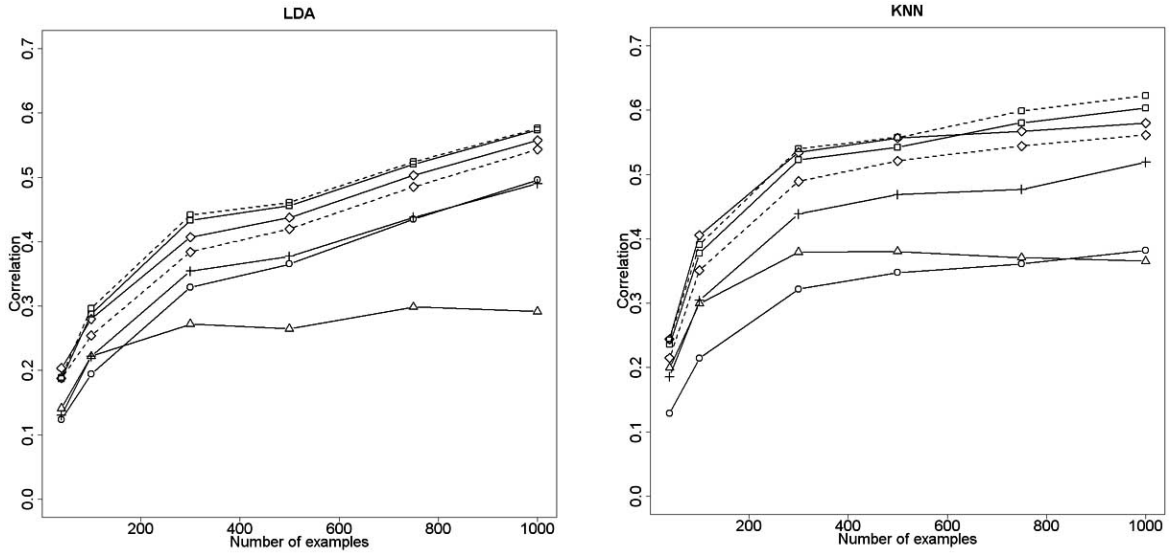


Fig. (7). Correlation between estimated and true errors as a function of the sample size: (**a**) LDA; (**b**) 3NN. Each curve represents an estimation method. Dot curve for resubstitution, triangle curve for leave-one-out, cross curve for cross-validation, diamond curve for bolstering, diamond curve with dotted line for semi-bolstering, square curve for bootstrap 632, square curve with dotted line for bootstrap 632+.

of the true on the estimated error arises from approximating the conditional expectation by the linear model

$$\mu_{tru|est} = a\,\hat{\varepsilon}_n + b \tag{21}$$

Based on the sample data, the least-squares estimate of the regression coefficient a is given by

$$\hat{a} = \frac{\hat{\sigma}_{tru}}{\hat{\sigma}_{est}}\hat{\rho} \tag{22}$$

where $\hat{\sigma}_{tru}$, $\hat{\sigma}_{est}$, and $\hat{\rho}$ are the sample-based estimates of the standard deviation of the true error, the standard deviation of the estimated error, and $\rho[\varepsilon_n, \hat{\varepsilon}_n]$, respectively. The closer $\hat{a}$ is to 0, the weaker the regression. Thus, the closer $\hat{\rho}$ is to 0, the weaker the regression. It is not unusual to have $\hat{\sigma}_{tru}/\hat{\sigma}_{est} \geq 1$, in which case small $\hat{a}$ is solely due to lack of correlation [12].

Fig. (**6**) shows the linear regression of the true error on the estimated error for various error estimators for LDA and the model used in this paper with sample size 60 and 5 features selected from 200 by the t-test. The scatter plots are less dense than in Figs. (**4** and **5**) because we require less data points for linear regression. As with nonlinear regression we observe a lack of regression. Indeed, the linear regression looks much like the nonlinear regression, which should not be surprising since the nonlinear regression curves are close to being flat straight lines. Again, bolstered resubstitution and bootstrap show more regression. Although the plots are not shown, as with nonlinear regression, there is some improvement with sample size 100, and similar results hold for other classification rules [12].

The lack of correlation is evident in Fig. (**6**). Large samples can increase this correlation. Fig. (**7**) shows the correlation between true and estimated errors as a function of the sample size, parts (**a**) and (**b**) showing LDA and 3NN, respectively. Note the very low value of correlation for small samples: less than 0.3 for n = 100 with LDA and less than 0.4 for n = 100 with 3NN. There is a strong increase of correlation between 50 and 300 then the increase (if any) is weaker. Fig. (**8**) shows the correlation as a function of the number of features, parts (**a**) and (**b**) showing LDA and 3NN, respectively. The correlation is decreasing with the dimensionality for both classification rules and all estimation methods. The decrease is very strong between 20 and 100, and then weakens. From D = 500, the correlation is less than 0.2 with LDA and less than 0.3 with 3NN.

Feature Set Ranking

If the features represent some physical variables, then good-performing feature sets may indicate that their multivariate activity bears some underlying relation to the target variable. For instance, in genomics, one may wish to find sets composed of genes for which there is evidence of their molecular relationship with a certain phenotype. The idea is that good feature sets may provide good candidates for diagnosis and therapy. Given a family of gene sets discovered by some classification rule, one issue is to rank them based on error. Thus, a natural measure of worth for an error estimator is its ranking accuracy for feature sets. Two measures of merit have been considered [13]. Each compares ranking based on true and estimated errors – under the condition that the true error is less than t. $R_1^K(t)$ is the number of feature sets in the truly top K feature sets that are also among the top K feature sets based on error estimation. It measures how well the error estimator finds top feature sets. $R_2^K(t)$ is the mean-absolute rank deviation for the K best feature sets. Fig. (**9**) shows graphs obtained by averaging these measures over 2000 trials with K = 40 for the model used throughout the paper and sample size n = 40. An exhaustive search of all

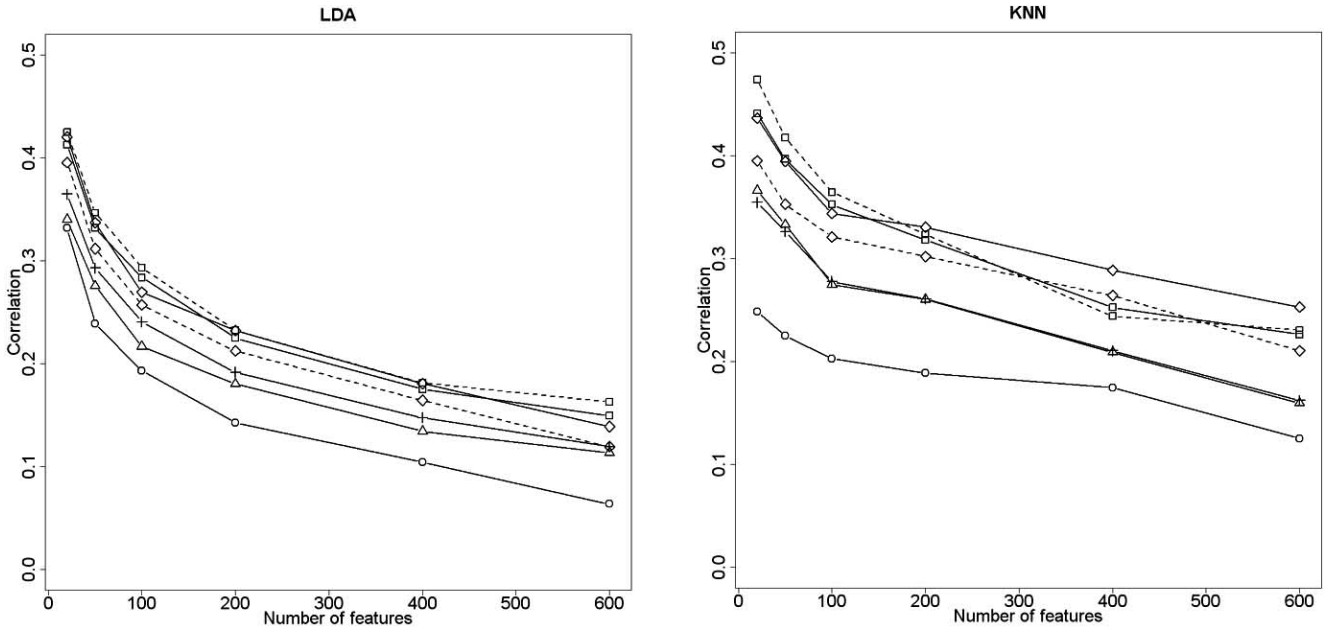


Fig. (8). Correlation between estimated and true errors as a function of the number of features: (**a**) LDA; (**b**) 3NN. Each curve represents an estimation method. Dot curve for resubstitution, triangle curve for leave-one-out, cross curve for cross-validation, diamond curve for bolstering, diamond curve with dotted line for semi-bolstering, square curve for bootstrap 632, square curve with dotted line for bootstrap 632+.

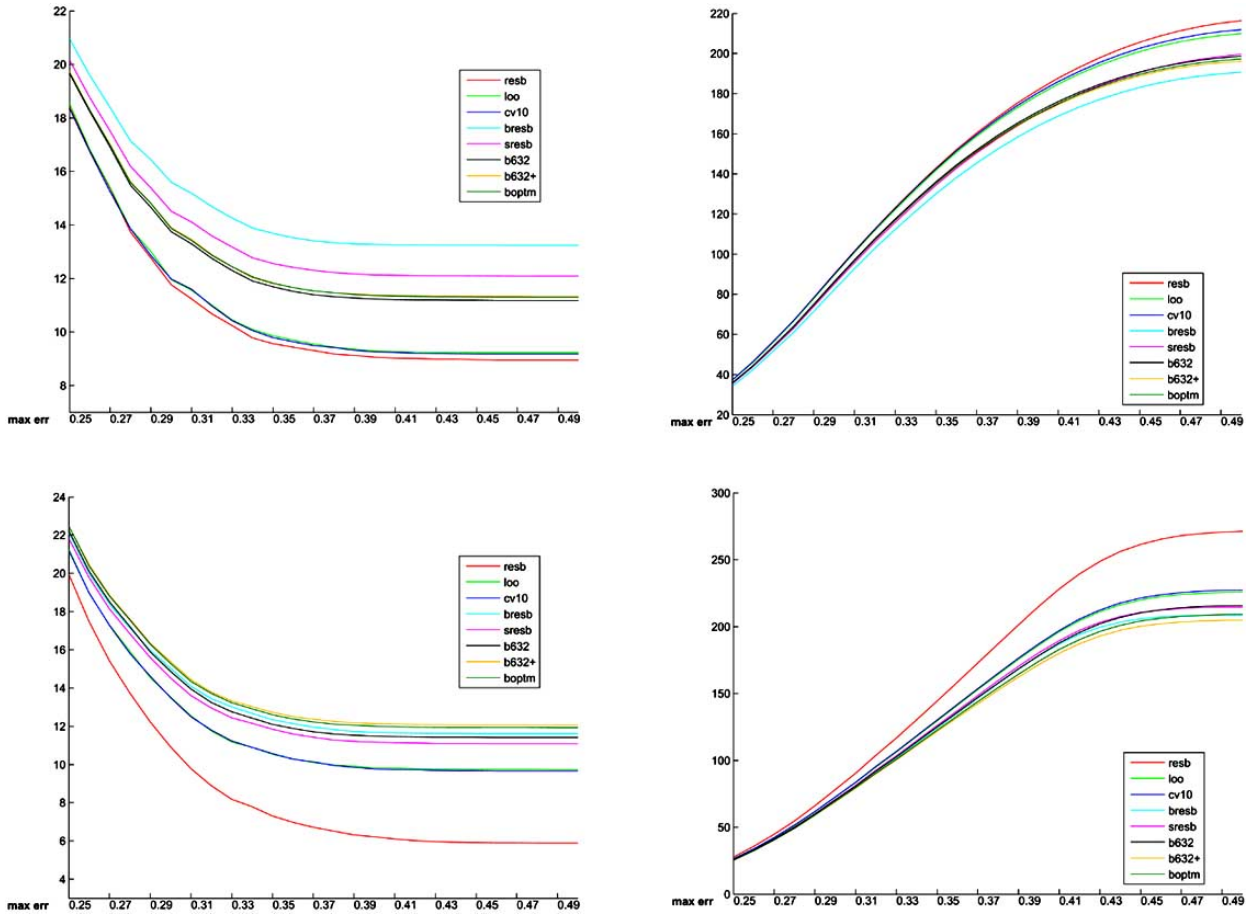


Fig. (9). Feature-set ranking measures: (**a**) LDA, $R_1^{40}(t)$; (**b**) LDA, $R_2^{40}(t)$; (**c**) 3NN, $R_1^{40}(t)$; (**d**) 3NN, $R_2^{40}(t)$.

feature sets has been performed for the true ranking with each feature set being of size $d = 3$ out of the $D = 20$ total number of features. Cross-validation generally performs poorer than the bootstrap and bolstered estimators.

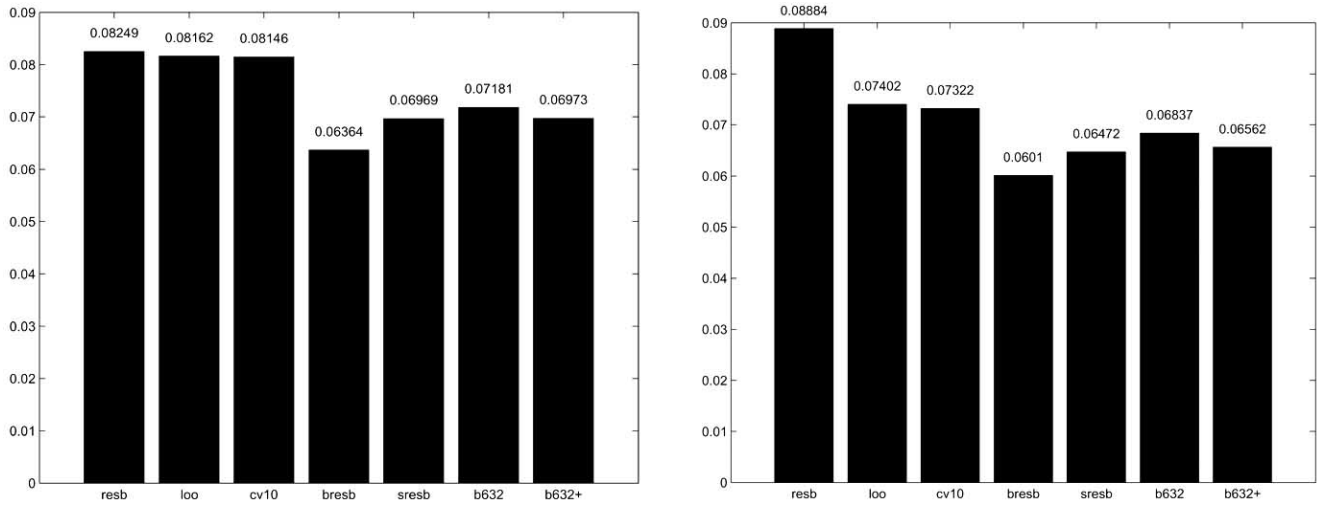


Fig. (10). Errors arising from different estimation methods inside the SFFS algorithm: (**a**) LDA; (**b**) 3NN.

Impact of Error Estimation on Feature Selection

When selecting features via an algorithm like SFFS that employs error estimation within it, the choice of error estimator impacts feature selection, the degree depending on the classification rule and feature-label distribution in [14]. As demonstrated in [14], SFFS can perform close to using an exhaustive search when the true error is used within the SFFS algorithm; however, this is not possible in practice, where error estimates must be used within the SFFS algorithm. It was also shown that the choice of error estimator can make a greater difference than the manner of feature selection. Indeed in that study there were examples using LDA and 3NN where the resulting error was less using SFFS with bolstering or bootstrap than performing a full search using leave-one-out cross-validation. Figure 10 illustrates the impact of error estimation within SFFS feature selection to select $d = 3$ features from $D = 50$ features in the quadratic model with with $\sigma_0 = 1$, $\sigma_1 = 1.5$, a_i selected form Beta(0.75, 2), and sample size $n = 40$. The errors in the figure are the average true errors of the resulting designed classifiers for 1000 replications. Parts (a) and (b) are from LDA and 3NN, respectively. In both cases, bolstered resubstitution performs the best, with semi-bolstered resubstitution performing second best.

Using Real Data

Using real data to evaluate error estimator performance is problematic from two perspectives, one methodological and the other epistemological. Methodologically, the data set has to be sufficiently large so that it can be randomly split into two subsets: (a) a training sample on which to train a classifier and estimate its error using a training-sample-based error estimator and (b) a disjoint test set on which to obtain a precise estimate of the true error. The data set has to be large enough that the disjoint test set is large enough for precise estimation and that the training and test sets obtained in successive random splits can be taken to be independent, when in fact they are not.

Epistemologically, one supposes that the data set has been generated from some unknown feature-label distribution and that the estimated error rate corresponds to the true error of the classifier on this unknown distribution; indeed, the whole idea of error estimation presupposes an error to estimate. Practically, the data have been obtained from some measurement process, say microarray readings on a sample of cancer patients in which the patients are divided into two classes. If the data set is large enough to satisfy the methodological demands, then the epistemological ground of the overall procedure is that, when applied to the current and future population of all cancer patients in the two defined classes, then the classifier will have a fixed error rate and that the error rate computed on the test sample estimates this error rate. There is a serious epistemological problem here: What is meant by the "defined classes?" If these classes are generated by a tightly controlled experimental protocol, then one has some confidence that the classes are meaningful; however, if they are merely defined by some set of measurements across a widely diverse population, they may, in fact, not be well-defined because, given a patient, it may not be decidable with a strong degree of certainty as to which class the patient belongs. To the degree that there is a tight experimental protocol resulting in two labeled classes, hypothesizing an unknown theoretical distribution from which the data have been drawn has justification and error estimation is meaningful; to the extent that data are grouped in some *ad hoc* manner, labeling loses its justification and, not only is the hypothesis of a feature-label distribution problematic, the entire notion of prediction, and therefore prediction accuracy, is untenable.

Supposing that the data set we have justifies performance evaluation, we will illustrate error regression and correlation using microarray data. We use a breast cancer dataset containing 295 patients divided into two classes, with 115 patients in the good-prognosis class and 180 patients in the poor-prognosis class [15]. The dataset is reduced to a selection of the 2000 genes with highest variance and these are reduced to 50 by using t-test feature selection. In the simulations, we divide the data into two sets. The first set consists of 50 examples drawn without replacement from the full data set. It is used for both training and training-sample-based error estimation. The remaining examples are used as a hold-

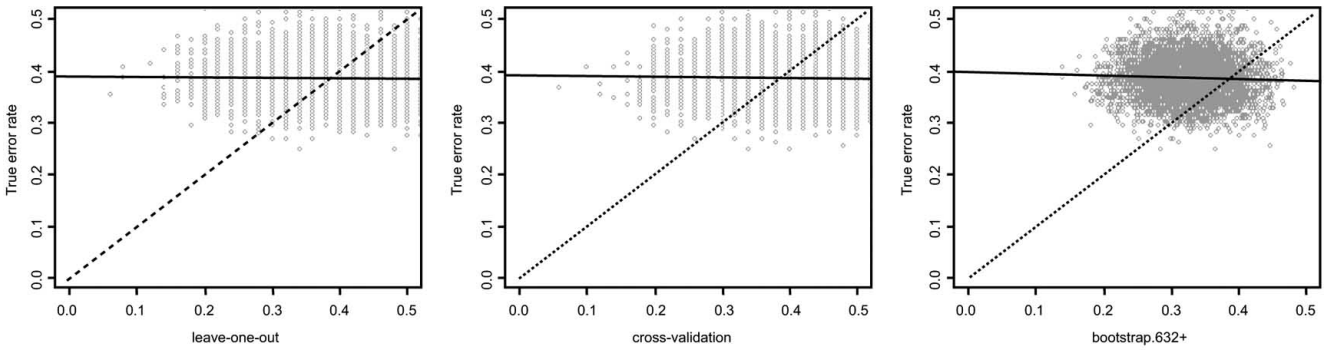


Fig. (11). Comparison of true and estimated errors in breast cancer dataset with LDA: left panel for leave-one-out, middle panel for 10-fold cross-validation, and right panel for bootstrap .632+.

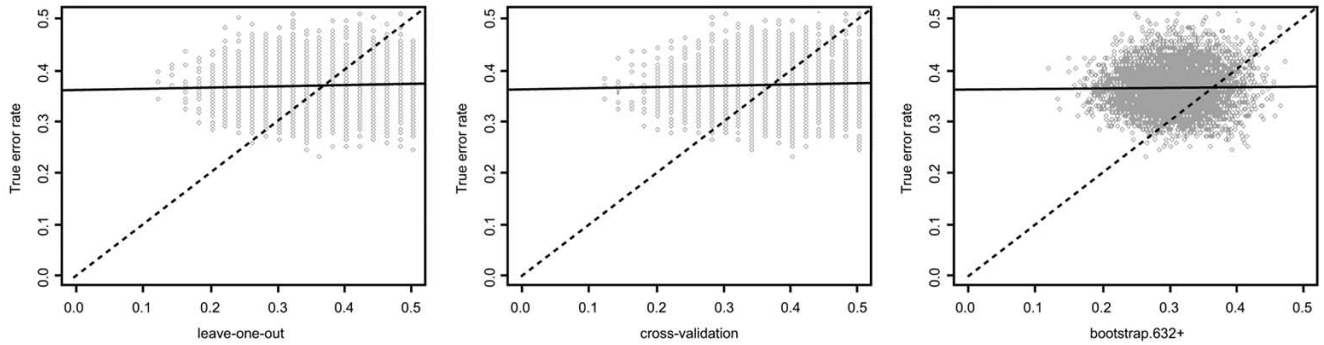


Fig. (12). Comparison of true and estimated errors in breast cancer dataset with 3NN: left panel for leave-one-out, middle panel for 10-fold cross-validation, and right panel for bootstrap .632+.

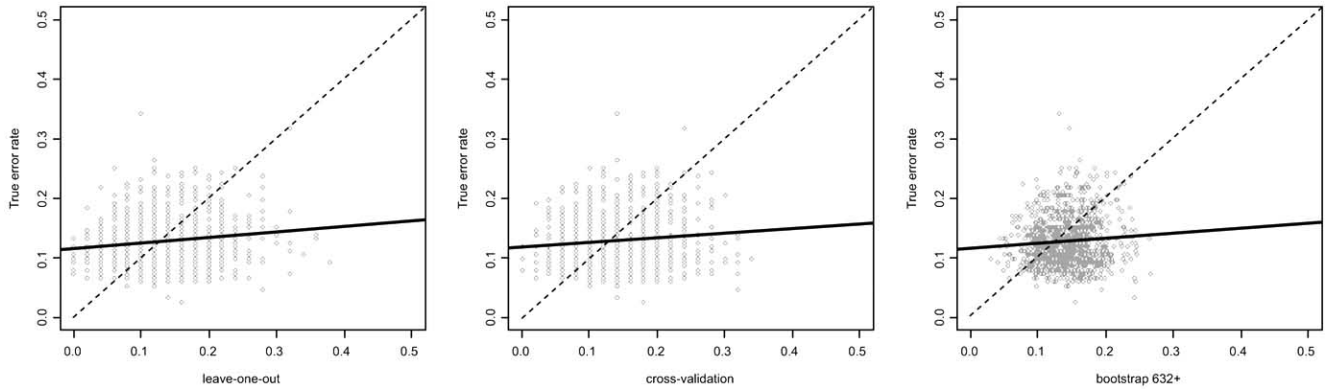


Fig. (13). Comparison of true and estimated errors in lung cancer dataset with LDA: left panel for leave-one-out, middle panel for 10-fold cross-validation, and right panel for bootstrap .632+.

out test set to get an accurate estimate of the true error, which is taken as the true error. This procedure is repeated 10,000 times. The results are shown in Fig. (**11**), LDA, and Fig. (**12**), 3NN, in which the vertical and horizontal axes represent the true and estimated errors, respectively, using leave-one-out cross-validation, 10-fold cross-validation and bootstrap .632+. We see that with both classification rules and all estimation methods, the regression line is virtually parallel to the x-axis; indeed, in Fig. (**11**) we observe a slight negative slope.

As a second illustration with patient data, we use a lung cancer dataset containing 203 patients, 139 patients with adenocarcinomas and 64 patients with cancers of another type [16]. Again, the dataset is reduced to the 2000 genes with highest variance, these are reduced to 50 by using t-test feature selection, and in the simulations the data are split into two sets, the first consisting of 50 examples drawn without replacement from the full dataset to be used for both training and training-sample-based error estimation, and the remaining examples being held-out as a test set to estimate the true error. This procedure is repeated 10,000 times. The results are shown in Fig. (**13**), LDA, and (**14**), 3NN. As with the breast cancer dataset, for both classification rules and all estimation methods, the regression line is virtually parallel to the x-axis.

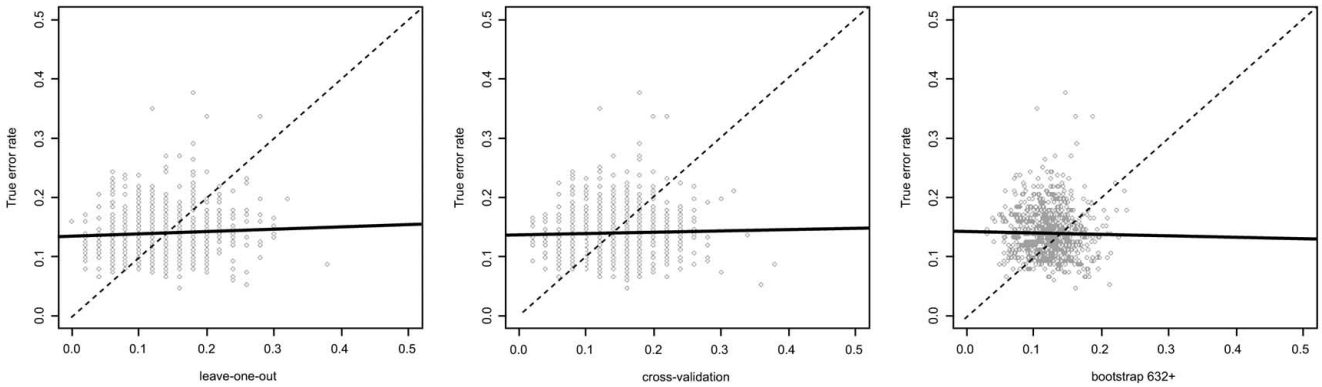


Fig. (14). Comparison of true and estimated errors in lung cancer dataset with 3NN: left panel for leave-one-out, middle panel for 10-fold cross-validation, and right panel for bootstrap .632+.

Figs. (**11**) through (**14**) illustrates a lack of regression relative to the estimated error serving as an estimate of the true error for the real data sets considered for a sample size of 50. Such small samples are not unusual in microarray classification studies owing to the difficulty of obtaining patient tissue samples and study cost. These issues continue to be an obstacle for obtaining large samples. For instance, consider the following sample sizes and error estimators reported in the literature: glioma, sample size 50 (only 21 classic tumors used for class prediction), leave-one-out [17]; leukemia, sample size 37, cross validation [18]; lung cancer, sample size 41, leave-one-out [19]; oral tongue carcinoma, sample size 39, leave-one-out [20]; colorectal cancer, sample size 53, leave-one-out [21]; and stage II colon cancer, sample size 50, cross validation [22]. When confronted with a small sample in which decent error estimation is impossible, one alternative is to report a list of feature sets and classifiers with small estimated errors, so that there is good likelihood that one of the classifiers actually does perform well [23-25].

CONCLUDING REMARKS

As noted at the outset, error estimation is critical to classification because the validity of the resulting classifier model is based on the accuracy of the error estimation procedure [1]. As can be seen from the examples presented in the paper, the commonly employed error estimators used when both training and testing are done on the same data suffer from performance problems with small samples, and performance is further degraded by high dimensional feature sets. Typically, with small samples there is very little correlation and a lack of regression between the estimated and true errors. Fig. (**11**) is striking in this regard because, in it, we actually observe a negative correlation between the estimated and true errors. One might conjecture that this is an artifact of the simulation, but, in fact, negative correlation with small samples can be demonstrated analytically in some classification settings [26]. These results lead to two conclusions: (1) much greater effort needs to be focused on error estimation, and (2) a conclusion based on a small-sample estimator should be supported by evidence that the estimator in question can be expected to perform sufficiently well under the circumstances to justify the conclusion – specifically, the known properties of the error estimator determine the validity of the conclusion. On the latter point, we are faced with fundamental epistemological and validation issues when constructing classifier models with small samples, and especially so when there is a large number of features. Small samples and high dimensionality are ubiquitous to genomic and proteomic classification. Let us conclude by saying that the epistemological problems in genomics with regard to classification and other issues have not gone unnoticed [1, 27-35].

ACKNOWLEDGEMENT

We appreciate the support of the National Science Foundation (CCF-0634794, CCF-0845407) for this work.

REFERENCES

[1] Dougherty ER, Braga-Neto UM. Epistemology of computational biology: Mathematical models and experimental prediction as the basis of their validity. *Biol Syst* **2006**; 14: 65-90.

[2] Braga-Neto UM, Dougherty ER. Bolstered error estimation. *Pattern Recognit* **2004**; 37: 1267-81.

[3] Braga-Neto UM, Dougherty ER. Is cross-validation valid for small-sample microarray classification. *Bioinformatics* **2004**; 20: 374-80.

[4] Efron B. Bootstrap methods: another look at the jacknife. *Ann Stat* **1979**; 7: 1-26.

[5] Efron B. Estimating the error rate of a prediction rule: Improvement on cross-validation. *J Am Stat Soc* **1983**; 78: 316-31.

[6] Sima C, Dougherty ER. Optimal convex error estimators for classification. *Pattern Recognit* **2006**; 39: 1763-80.

[7] Efron B, Tibshirani R. Improvements on cross-validation: The 632+ bootstrap method. *J Am Stat Assoc* **1997**; 92: 548-60.

[8] Pudil P, Novovicova J, Kittler J. Floating search methods in feature selection. *Pattern Recognit Lett* **1994**; 15: 1119-25.

[9] Molinaro AM, Simon R, Pfeiffer RM. Prediction error estimation: A comparison of resampling methods. *Bioinformatics* **2005**; 21: 3301-07.

[10] Xiao Y, Hua J, Dougherty ER. Quantification of the impact of feature selection on cross-validation error estimation precision. *EURASIP J Bioinform Syst Biol* **2007**; Article ID 16354, 11 pages, 2007.

[11] Xu Q, Hua J, Braga-Neto UM, Xiong Z, Suh E, Dougherty ER. Confidence intervals for the true classification error conditioned on the estimated error. *Tech Cancer Res Treat* **2006**; 5: 579-90.

[12] Hanczar B, Hua J, Dougherty ER. Decorrelation of the true and estimated classifier errors in high-dimensional settings. *EURASIP J Bioinform Syst Biol* **2007**; Article ID 38473, 12 pages, doi:10.1155/2007/38473, 2007.

[13] Sima C, Braga-Neto UM, Dougherty ER. Superior feature-set ranking for small samples using bolstered error estimation. *Bioinformatics* **2005**; 21: 1046-54.

[14] Sima C, Attoor S, Braga-Neto UM, Lowey J, Suh E, Dougherty ER. Impact of error estimation on feature-selection algorithms. *Pattern Recognit* **2005**; 38: 2472-82.

[15] van de Vijver MJ, He YD, van't Veer LJ, *et al.* A gene expression signature as a predictor of survival in breast cancer. *Eng J Med* **2002**; 347: 1999-2009.

[16] Bhattacharjee A, Richards WG, Staunton J, *et al.* Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. *Proc Natl Acad Sci USA* **2001**; 96: 6745-50.

[17] Nutt CL, Mani DR, Betensky RA, *et al.* Gene expression-based classification of malignant gliomas correlates better with survival than histological classification. *Cancer Res* **2003**; 63: 1602-07.

[18] Armstrong SA, Staunton JE, Silverman LB, *et al.* MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nat Gene* **2002**; 30: 41-7.

[19] Moriya Y, Iyoda A, Kasai Y, *et al.* Prediction of lymph node metastasis by gene expression profiling in patients with primary resected lung cancer. *Lung Cancer* **2009**; 64: 86-91.

[20] Watanabe H, Mogushi K, Miura M, *et al.* Prediction of lymphatic metastasis based on gene expression profile analysis after brachytherapy for early-stage oral tongue carcinoma. *Radiat Oncol* **2008**; 87: 237-42.

[21] Watanabe T, Kobunai T, Toda E, *et al.* Gene expression signature and the prediction of ulcerative colitis-associated colorectal cancer by DNA microarray. *Clin Cancer Res* **2007**; 13: 415-20.

[22] Barrier A, Boelle PY, Roser F, *et al.* Stage II colon cancer prognosis prediction by tumor gene expression profiling. *J Clin Oncol* **2006**; 24: 4685-91.

[23] Kim S, Dougherty ER, Shmulevich I, *et al.* Identification of combination gene sets for glioma classification. *Mol Cancer Therap* **2002**; 1: 1229-36.

[24] Kobayashi T, Yamaguchi M, Kim S, *et al.* Gene expression profiling identifies strong feature genes that classify de novo $CD5^+$ and $CD5^-$ diffuse large B-cell lymphoma and mantle cell lymphoma. *Cancer Res* **2003**; 63: 60-66.

[25] Zhao C, Ivanov I, Dougherty ER, *et al.* Non-invasive detection of candidate molecular biomarkers in subjects with a history of insulin resistance and colorectal adenomas. *Cancer Prev Res* **2009**; 2: 590-97.

[26] Braga-Neto UM, Dougherty ER. Exact correlation between actual and estimated errors in discrete classification **2010**; in press.

[27] Dougherty ER. Small-sample issues for microarray-based classification. *Comp Funct Genomics* **2001**; 2: 28-34.

[28] Potter JD. At the interfaces of epidemiology, genetics and genomics. *Nat Rev/Gene* **2001**; 2: 142-47.

[29] Mehta T, Murat T, Allison DB. Towards sound epistemological foundations of statistical methods for high-dimensional biology. *Nat Gene* **2004**; 36: 943-47.

[30] Michiels S, Koscielny S, Hill C. Prediction of cancer outcome with microarrays; A multiple random validation strategy. *Lancet* **2005**; 365: 482-88.

[31] Dougherty ER, Hua J, Bittner ML. Validation of computational methods in genomics. *Curr Genomics* **2007**; 8: 1-19.

[32] Dupuy A, Simon RM. Critical review of published microarray studies for cancer outcome and guidelines on statistical analysis and reporting. *J Nat Cancer Ins* **2007**; 99: 147-57.

[33] Braga-Neto UM. Fads and fallacies in the name of small-sample microarray classification. *IEEE Signal Proc Mag* **2007**; 24: 91-99.

[34] Dougherty ER. On the epistemological crisis in genomics. *Curr Genomics* **2008**; 9: 69-79.

[35] Wolkenhauer O. Why systems biology Is (Not) called systems biology. *Bioforum Eur* **2007**; 4: 2-3.

Received: January 15, 2009 Revised: July 13, 2009 Accepted: July 16, 2009