EURASIP Journal on Bioinformatics and Systems Biology
a SpringerOpen Journal

RESEARCH Open Access

Unbiased bootstrap error estimation for linear discriminant analysis

Thang Vu[1], Chao Sima[2], Ulisses M Braga-Neto[1,2*] and Edward R Dougherty[1,2]

Abstract

Convex bootstrap error estimation is a popular tool for classifier error estimation in gene expression studies. A basic question is how to determine the weight for the convex combination between the basic bootstrap estimator and the resubstitution estimator such that the resulting estimator is unbiased at finite sample sizes. The well-known 0.632 bootstrap error estimator uses asymptotic arguments to propose a fixed 0.632 weight, whereas the more recent 0.632+ bootstrap error estimator attempts to set the weight adaptively. In this paper, we study the finite sample problem in the case of linear discriminant analysis under Gaussian populations. We derive exact expressions for the weight that guarantee unbiasedness of the convex bootstrap error estimator in the univariate and multivariate cases, without making asymptotic simplifications. Using exact computation in the univariate case and an accurate approximation in the multivariate case, we obtain the required weight and show that it can deviate significantly from the constant 0.632 weight, depending on the sample size and Bayes error for the problem. The methodology is illustrated by application on data from a well-known cancer classification study.

Keywords: Bootstrap; Error estimation; Bias; Linear discriminant analysis; Gene expression classification

Introduction

The bootstrap method [1-7] has been used in a wide range of statistical problems. The asymptotic behavior of bootstrap has been studied [8-11], while small-sample properties have been studied under simplifying assumptions, such as considering the estimator based on all possible bootstrap samples (the 'complete' bootstrap) [12-14]. The small-sample properties of the usual bootstrap are not well understood, in particular when it comes to estimating the error rates of classification rules [15,16].

There has been, on the other hand, interest in the application of bootstrap to error estimation in classification problems and, in particular, gene expression classification studies [17-20]. Of particular interest is the issue of classifier error estimation [21,22]. Bootstrap methods have generally been shown to outperform more traditional error estimation techniques, such as resubstitution and cross-validation, in terms of root-mean-square (RMS) error [4,5,7,23-35]. Bootstrap error estimation is typically performed via a convex combination of the (generally) pessimistic basic bootstrap estimator, known as the zero bootstrap, and the (generally) optimistic resubstitution estimator. A basic problem is how to choose the weight that yields an unbiased estimator.

The problem of unbiased convex error estimation was previously considered in [36-38] for a convex combination of resubstitution and cross-validation estimators, and in [4,7,23] for a combination between resubstitution and the basic bootstrap estimator. In the former case, a fixed suboptimal weight of 0.5 was proposed in [36,38], while an asymptotic analysis to find the optimal weight was provided in [37]. In the latter case, our case of interest, a fixed suboptimal weight of 0.632 was proposed in [4], leading to the well-known 0.632 bootstrap estimator, while in [7], a suboptimal weight is computed by means of a sample-based procedure, which attempts to counterbalance the effect of overfitting on the bias, leading to the so-called 0.632+ bootstrap error estimator; the problem of finding the optimal weight for finite sample cases was addressed via a numerical approach in [23].

Here, we determine the optimal weight for finite sample cases analytically, in the case of linear discriminant analysis under Gaussian populations. In the univariate case, no

*Correspondence: ulisses@ece.tamu.edu
[1]Department of Electrical and Computer Engineering, Texas A&M University, 3128 TAMU, College Station, TX 77843, USA
[2]Center for Bioinformatics and Genomic Systems Engineering, Texas A&M University, 101 Gateway, Suite A, College Station, TX 77845, USA

other assumptions are made. In the multivariate case, it is assumed that the populations are homoskedastic and that the common covariance matrix is known and used in the discriminant. In either case, no simplifications are introduced to the bootstrap error estimator; it is the usual one, based on a finite number of random bootstrap samples.

The analysis in this paper follows in the steps of previous papers that have provided analytical representations for the moments of error-estimator distributions [39,40]. In the univariate case, exact expressions are given for the expectation of the zero bootstrap error estimator, in the general heteroskedastic (general-variance) Gaussian case. By using similar expressions for the expected true and resubstitution error [39], this allows the exact calculation of the required weight. In the multivariate case, the expectation of the zero bootstrap error estimator is expressed as a probability involving the ratio of two noncentral chi-square variables, in the homoskedastic Gaussian case, assuming that the true common covariance matrix is used in the discriminant. The resulting expression is exact but necessitates approximation for its numerical computation. This is done in this paper via the Imhof-Pearson three-moment method, which is accurate in small-sample cases [41]. Use of similar expressions for the expected true and resubstitution error [40] then allows the exact calculation of the required weight.

In the homoskedastic case, the required weight for unbiasedness is shown to be a function only of the Bayes error and sample size. Accordingly, plots and tables of the required weight for varying values of Bayes error and sample size are presented; if the Bayes error can be estimated for a problem, this provides a way to obtain the optimal weight to use. In the univariate case, it was observed that as the sample size increases, the optimal weight settles on an asymptotic value of around 0.675, thus slightly over the heuristic value 0.632; by contrast, in the multivariate case ($d = 2$), the asymptotic value appears to be strongly dependent on the Bayes error, being as a rule significantly smaller than 0.632, except for very small Bayes error.

This paper is organized as follows. The 'Bootstrap classification' section defines linear discriminant analysis as well as its application under bootstrap sampling. The 'Bootstrap error estimation' section reviews convex bootstrap error estimation. The 'Unbiased bootstrap error estimation' section contains the main theoretical results in the paper, providing the analytical expressions for the computation of the required convex bootstrap weight in the univariate and multivariate cases. The 'Gene expression classification example' section contains a demonstration of the usage of the optimal weight in bootstrap error estimation using data from the breast cancer classification study in [42,43]. Lastly, the 'Conclusions' section contains a summary and concluding remarks.

All the proofs are presented in the Appendix.

Bootstrap classification

Classification involves a predictor vector $X \in R^d$, also known as a *feature* vector, which represents an individual from one of two populations Π_0 and Π_1 (we consider here only this binary classification problem). The classification problem is to assign X correctly to its population of origin. The populations are coded into a discrete *label* $Y \in \{0, 1\}$. Therefore, given a feature vector X, classification attempts to predict the corresponding value of the label Y. We assume that there is a joint *feature-label distribution* F_{XY} for the pair (X, Y) characterizing the classification problem. In particular, it determines the probabilities $c_0 = P(X \in \Pi_0) = P(Y = 0)$ and $c_1 = P(X \in \Pi_1) = P(Y = 1)$, which are called the *prior probabilities*.

Given a fixed sample size n, the *sample data* is an i.i.d. sample $S_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ from F_{XY}. The population-specific sample sizes are given by $n_0 = \sum_{i=1}^n I_{Y_i=0}$ and $n_1 = \sum_{i=1}^n I_{Y_i=1} = n - n_0$, which are random variables, with $n_0 \sim \text{Binomial}(n, c_0)$ and $n_1 \sim \text{Binomial}(n, c_1)$. When we need to emphasize that n_0 and n_1 are random variables, we will use capital letters N_0 and N_1, respectively. This sampling design, which is the most commonly found one in contemporary pattern recognition, is known as *mixture sampling* [44].

A *classification rule* Ψ_n is used to map the training data S_n into a designed classifier $\psi_n = \Psi_n(S_n)$, where ψ_n is a function taking on values in the set $\{0, 1\}$, such that X is assigned to population Π_0 or Π_1 according to whether $\psi_n(X) = 0$ or 1, respectively. The *classification error rate* ε_n of classifier ψ_n is the probability that the assignment is erroneous:

$$\begin{aligned}\varepsilon_n &= c_0\, P(\psi_n(X) = 1 \mid Y = 0) + c_1\, P(\psi_n(X) = 0 \mid Y = 1) \\ &\stackrel{\text{def}}{=} c_0\, \varepsilon_n^0 + c_1\, \varepsilon_n^1,\end{aligned} \tag{1}$$

where (X, Y) is an independent test point and $\varepsilon_n^i = P(\psi_n(X) = 1 - i \mid Y = i)$ is the error rate specific to population Π_i, for $i = 0, 1$. Since the training set S_n is random, ε_n is a random variable, with *expected classification error rate* $E[\varepsilon_n]$; this gives the average performance over all possible training sets S_n, for fixed sample size n.

Linear discriminant analysis (LDA) employs Anderson's W discriminant [45], which is defined as follows:

$$W(X) = \left(X - \frac{\hat{\mu}_0 + \hat{\mu}_1}{2}\right)^T \Sigma^{-1} \left(\hat{\mu}_0 - \hat{\mu}_1\right) \tag{2}$$

where

$$\hat{\mu}_0 = \frac{1}{n_0} \sum_{i=1}^n X_i I_{Y_i=0} \quad \text{and} \quad \hat{\mu}_1 = \frac{1}{n_1} \sum_{i=1}^n X_i I_{Y_i=1} \tag{3}$$

are the sample means relative to each population, and Σ is a matrix, which can be either (1) the true common covariance matrix of the populations, assuming it is known (this is the approach followed, for example, in [39,40,46]), or (2) the sample covariance matrix based on the pooled sample S_n, which leads to the general LDA case. In this paper, we will assume case (1) throughout.

The corresponding LDA classifier is given by

$$\psi_n(X) = \begin{cases} 1\,, & \text{if } W(X) < 0 \\ 0\,, & \text{if } W(X) \geq 0 \end{cases}, \tag{4}$$

that is, the sign of $W(X)$ determines the classification of X.

A *bootstrap sample* S_n^* contains n instances drawn uniformly, with replacement, from S_n. Hence, some of the instances in S_n may appear multiple times in S_n^*, whereas others may not appear at all. Let C be a vector of size n, where the ith component $C(i)$ equals the number of appearances in S_n^* of the ith instance in S_n. The vector C will be referred to as a *bootstrap vector*.

For a given S_n, the vector C uniquely determines a bootstrap sample S_n^*, which we denote by S_n^C. Note that the original sample itself is included: if $C = (1, \ldots, 1) \stackrel{\text{def}}{=} \mathbf{1}_n$, then $S_n^C = S_n$, since each original instance appears once in the bootstrap sample. Note also that the number of distinct bootstrap samples, i.e., values for C, is equal to $\binom{2n-1}{n}$; even for small n, this is a large number. For example, the total number of possible bootstrap samples of size $n = 20$ is larger than 6.8×10^{10}.

The vector C has a multinomial distribution with parameters $(n, 1/n, \ldots, 1/n)$,

$$P(C = (i_1, \ldots, i_n)) = \frac{1}{n^n} \frac{n!}{i_1! \cdots i_n!}, \quad i_1 + \cdots + i_n = n\,. \tag{5}$$

Starting from a classification rule Ψ_n, one may design a classifier $\psi_n^C = \Psi_n(S^C)$ on a bootstrap training set S^C. Its classification error ε_n^C is given as in (1), namely, $\varepsilon_n^C = c_0 \varepsilon_n^{C,0} + c_1 \varepsilon_n^{C,1}$ where $\varepsilon_n^{C,i} = P(\psi_n^C(X) = 1 - i \mid Y = i)$ is the error rate specific to population Π_i, for $i = 0, 1$. In this paper, we apply this scheme to the LDA classification rule defined previously. Notice the distinction between a bootstrap LDA classifier and a 'bagged' (bootstrap-aggregated) LDA classifier [47,48]; these correspond to distinct classification rules. The bootstrap LDA classifier is employed here as an auxiliary tool to analyze the problem of unbiased bootstrap error estimation for the plain LDA classifier.

Bootstrap error estimation

Since the feature-label distribution is typically unknown, the classification error rate ε_n has to be estimated by a sample-based statistic $\hat{\varepsilon}_n$, commonly referred to as an *error estimator*. Data in practice are often limited, and the training sample S_n has to be used for both designing the classifier ψ_n and as the basis for the error estimator $\hat{\varepsilon}_n$. The simplest and fastest way to estimate the error of a designed classifier ψ_n is to compute its error on the sample data itself:

$$\hat{\varepsilon}_n^{\,r} = \frac{1}{n} \sum_{i=1}^{n} \left(I_{\psi_n(X_i)=1}\, I_{Y_i=0} + I_{\psi_n(X_i)=0}\, I_{Y_i=1} \right). \tag{6}$$

This *resubstitution* estimator, or *apparent error*, is often optimistically biased, that is, it is often the case that $\mathrm{Bias}\left(\hat{\varepsilon}_n^{\,r}\right) = E\left[\hat{\varepsilon}_n^{\,r}\right] - E[\varepsilon_n] < 0$, though this is not always so. The bias tends to worsen with more complex classification rules [49].

The basic bootstrap error estimator is the *zero bootstrap* error estimator [4], which is introduced next. Given the training data S_n, B bootstrap samples are randomly drawn from it. Denote the corresponding (random) bootstrap vectors by $\{C_1, \ldots, C_B\}$. The zero bootstrap error estimator is defined as the average error committed by the B bootstrap classifiers on sample points that do not appear in the bootstrap samples:

$$\hat{\varepsilon}_n^{\,\mathrm{boot}} = \frac{1}{B} \sum_{i=1}^{B} \left[\frac{1}{n(C_i)} \sum_{j:\, C_i(j)=0} \left(I_{\psi_n^{C_i}(X_j)=1}\, I_{Y_j=0} + I_{\psi_n^{C_i}(X_j)=0}\, I_{Y_j=1} \right) \right], \tag{7}$$

where $n(C)$ is the number of zeros in C.

The bootstrap zero estimator tends to be pessimistically biased, since the amount of distinct training instances available for designing the classifier is on average $(1 - e^{-1})n \approx 0.632n < n$. Pessimistic bias in an error estimator can be mitigated by forming a convex combination with an optimistic error estimator [23]. In the case of bootstrap error estimation, the standard approach is to form a convex combination of the zero bootstrap with resubstitution,

$$\hat{\varepsilon}_n^{\,\mathrm{conv}} = (1 - w)\, \hat{\varepsilon}_n^{\,r} + w\, \hat{\varepsilon}_n^{\,\mathrm{boot}}\,. \tag{8}$$

Selecting the appropriate weight $w = w^*$ leads to an unbiased error estimator, $E[\hat{\varepsilon}_n^{\,\mathrm{conv}}] = E[\varepsilon_n]$.

In [4], the weight w is heuristically set to $w = 0.632$ to reflect the average ratio of original training instances that appear in a bootstrap sample. This is known as the *.632 bootstrap estimator*

$$\hat{\varepsilon}_n^{\,\mathrm{b632}} = (1 - 0.632)\, \hat{\varepsilon}_n^{\,r} + 0.632\, \hat{\varepsilon}_n^{\,\mathrm{boot}}\,, \tag{9}$$

which has been heavily employed in the machine learning field.

Unbiased bootstrap error estimation

The 0.632 bootstrap error estimator reviewed in the previous section is not guaranteed to be unbiased. In this

section, we will examine the necessary conditions for setting the weight $w = w^*$ in (8) to achieve unbiasedness. We will then particularize the analysis to the Gaussian linear discriminant case, where exact expressions for w^* will be derived, both in the univariate and multivariate cases.

The bias of the convex estimator in (8) is given by

$$E\left[\hat{\varepsilon}_n^{\text{conv}} - \varepsilon_n\right] = (1-w)E\left[\hat{\varepsilon}_n^r\right] + wE\left[\hat{\varepsilon}_n^{\text{boot}}\right] - E\left[\varepsilon_n\right]. \tag{10}$$

Setting this to zero yields the exact weight

$$w^* = \frac{E\left[\hat{\varepsilon}_n^r\right] - E\left[\varepsilon_n\right]}{E\left[\hat{\varepsilon}_n^r\right] - E\left[\hat{\varepsilon}_n^{\text{boot}}\right]} \tag{11}$$

that produces an unbiased error estimator.

Now, applying expectation on both sides of (7) produces

$$E\left[\hat{\varepsilon}_n^{\text{boot}}\right] = \sum_C E\left[\varepsilon_n^C \mid C\right] p(C), \tag{12}$$

where $p(C)$ is given by (5) and the sum is taken over all possible values of C (an efficient procedure for listing all multinomial vectors is provided by the NEXCOM routine given in [50], Chapter 5). Equations (11) and (12) allow the computation of the weight w^* given the knowledge of $E[\,\varepsilon_n]$, $E\left[\hat{\varepsilon}_n^r\right]$, and $E\left[\varepsilon_n^C \mid C\right]$. We will present next exact formulas for these expectations in the case of the LDA classification rule under Gaussian populations.

Univariate case

In the univariate case, the common variance term cancels and the W statistic and LDA classifier become greatly simplified, with

$$\psi_n(X) = \begin{cases} 1, & \text{if } \left(X - \frac{\hat{\mu}_0+\hat{\mu}_1}{2}\right)(\hat{\mu}_0 - \hat{\mu}_1) < 0 \\ 0, & \text{otherwise} \end{cases}. \tag{13}$$

The following functions will be useful. Let $\Phi(u) = P(Z \le u)$ and $\Phi(u, v; \rho) = P((Z_1, Z_2) \le (u, v))$, where Z is a zero-mean, unit-variance Gaussian random variable, and Z_1, Z_2 are zero-mean, unit-variance random variables that are jointly Gaussian distributed, with correlation coefficient ρ.

Assume that population Π_i is distributed as $N(\mu_i, \sigma_i)$, for $i = 0, 1$, where $\sigma_0 \neq \sigma_1$ in general.

Under these conditions, John obtained in [39] an exact expression for the expectation of the true classification error for *fixed* sample sizes n_0 and n_1 (this is known as *separate* sampling [44]). John's result can be written as follows:

$$E\left[\varepsilon_n^0 \mid N_0 = n_0\right] = \Phi(a, b; \rho_e) + \Phi(-a, -b; \rho_e), \tag{14}$$

where

$$a = \frac{\mu_1 - \mu_0}{\sqrt{\frac{\sigma_0^2}{n_0} + \frac{\sigma_1^2}{n_1}}}, \quad b = \frac{\mu_0 - \mu_1}{\sqrt{\left(4 + \frac{1}{n_0}\right)\sigma_0^2 + \frac{\sigma_1^2}{n_1}}},$$
$$\rho_e = \frac{\frac{\sigma_0^2}{n_0} - \frac{\sigma_1^2}{n_1}}{\sqrt{\left(\frac{\sigma_0^2}{n_0} + \frac{\sigma_1^2}{n_1}\right)\left(\left(4 + \frac{1}{n_0}\right)\sigma_0^2 + \frac{\sigma_1^2}{n_1}\right)}}. \tag{15}$$

The corresponding result for $E[\,\varepsilon_n^1 \mid N_0 = n_0]$ is obtained by simply interchanging all indices 0 and 1 in the previous expressions. The expected error rate can then be found by using conditioning and Equation (1):

$$\begin{aligned} E[\,\varepsilon_n] &= \sum_{n_0=0}^{n} E[\,\varepsilon_n \mid N_0 = n_0]\, P(N_0 = n_0) \\ &= \sum_{n_0=0}^{n} \left(c_0 E\left[\varepsilon_n^0 \mid N_0 = n_0\right] + c_1 E\left[\varepsilon_n^1 \mid N_0 = n_0\right]\right) \\ &\quad \times P(N_0 = n_0). \end{aligned} \tag{16}$$

where

$$P(N_0 = n_0) = \binom{n}{n_0} c_0^{n_0} c_1^{n_1}. \tag{17}$$

As for resubstitution, Hills provided in [51] exact expressions for the expected error for fixed n_0 and n_1. However, his expression applies only to the case $\sigma_0 = \sigma_1$. Theorem 3 in [52] provides a generalization of this result to the case of populations of unequal variances. First, note that

$$\hat{\varepsilon}_n^r = \frac{n_0}{n}\hat{\varepsilon}_n^{r,0} + \frac{n_1}{n}\hat{\varepsilon}_n^{r,1}, \tag{18}$$

where

$$\begin{aligned} \hat{\varepsilon}_n^{r,0} &= \frac{1}{n_0}\left[\sum_{i=1}^{n} I_{\psi(X_i)=1}\, I_{Y_i=0}\right] \quad \text{and} \\ \hat{\varepsilon}_n^{r,1} &= \frac{1}{n_1}\left[\sum_{i=1}^{n} I_{\psi(X_i)=0}\, I_{Y_i=1}\right] \end{aligned} \tag{19}$$

are the apparent error rates specific to class 0 and 1, respectively. The result in [52] can be written as

$$E\left[\hat{\varepsilon}_n^{r,0} \mid N_0 = n_0\right] = \Phi(c, d; \rho_r) + \Phi(-c, -d; \rho_r), \tag{20}$$

where

$$c = \frac{\mu_1 - \mu_0}{\sqrt{\frac{\sigma_0^2}{n_0} + \frac{\sigma_1^2}{n_1}}}, \quad d = \frac{\mu_0 - \mu_1}{\sqrt{\left(4 - \frac{3}{n_0}\right)\sigma_0^2 + \frac{\sigma_1^2}{n_1}}},$$
$$\rho_r = -\frac{\frac{\sigma_0^2}{n_0} + \frac{\sigma_1^2}{n_1}}{\sqrt{\left(\frac{\sigma_0^2}{n_0} + \frac{\sigma_1^2}{n_1}\right)\left(\left(4 - \frac{3}{n_0}\right)\sigma_0^2 + \frac{\sigma_1^2}{n_1}\right)}}. \tag{21}$$

The corresponding result for $E[\hat{\varepsilon}_n^{r,1} \mid N_0 = n_0]$ is obtained by interchanging all indices 0 and 1. The expected resubstitution error rate can then be found by using conditioning and Equation (18):

$$\begin{aligned} E\left[\hat{\varepsilon}_n^r\right] &= \sum_{n_0=0}^{n} E\left[\hat{\varepsilon}_n^r \mid N_0 = n_0\right] P(N_0 = n_0) \\ &= \sum_{n_0=0}^{n} \left(\frac{n_0}{n} E\left[\hat{\varepsilon}_n^{r,0} \mid N_0 = n_0\right] + \frac{n_1}{n} E\left[\hat{\varepsilon}_n^{r,1} \mid N_0 = n_0\right]\right) \\ &\quad \times P(N_0 = n_0). \end{aligned} \tag{22}$$

Finally, let us consider the expected bootstrap error. Given C, the bootstrap LDA classifier is obtained by replacing $\hat{\mu}_i$ by $\hat{\mu}_i^C$, $i = 0, 1$, in (13):

$$\psi_n^C(X) = \begin{cases} 1, & \text{if } \left(X - \frac{\hat{\mu}_0^C + \hat{\mu}_1^C}{2}\right)\left(\hat{\mu}_0^C - \hat{\mu}_1^C\right) < 0 \\ 0, & \text{otherwise} \end{cases}, \tag{23}$$

where

$$\hat{\mu}_0^C = \frac{\sum_{i=1}^n C(i) X_i I_{Y_i=0}}{\sum_{i=1}^n C(i) I_{Y_i=0}} \quad \text{and} \quad \hat{\mu}_1^C = \frac{\sum_{i=1}^n C(i) X_i I_{Y_i=1}}{\sum_{i=1}^n C(i) I_{Y_i=1}} \tag{24}$$

are *bootstrap sample means*.

Now, note that with $N_0 = n_0$ fixed, the training data labels Y_i, $i = 1, \ldots, n$, are no longer random. Since all classification rules of interest are invariant to reordering of the training data, we can, without loss of generality, reorder the sample points so that $Y_i = 0$ for $i = 1, \ldots, n_0$, and $Y_1 = 1$ for $i = n_0 + 1, \ldots, n$. Let the same reordering be applied to a given bootstrap vector C. The next theorem extends John's result to the classification error of the bootstrapped LDA classification rule defined by (23).

Theorem 1. *Assume that population Π_i is distributed as $N\left(\mu_i, \sigma_i^2\right)$, for $i = 0, 1$. Then the expected error rate of the bootstrap LDA classification rule defined by (23) is given by:*

$$E\left[\varepsilon_n^{C,0} \mid N_0 = n_0, C\right] = \Phi(e, f; \rho_c) + \Phi(-e, -f; \rho_c), \tag{25}$$

where

$$e = \frac{\mu_1 - \mu_0}{\sqrt{s_0\sigma_0^2 + s_1\sigma_1^2}}, \quad f = \frac{\mu_0 - \mu_1}{\sqrt{(4 + s_0)\sigma_0^2 + s_1\sigma_1^2}},$$
$$\rho_c = \frac{s_0\sigma_0^2 - s_1\sigma_1^2}{\sqrt{\left((4 + s_0)\sigma_0^2 + s_1\sigma_1^2\right)\left(s_0\sigma_0^2 + s_1\sigma_1^2\right)}}, \tag{26}$$

with

$$s_0 = \frac{\sum_{i=1}^{n_0} C(i)^2}{\left(\sum_{i=1}^{n_0} C(i)\right)^2} \quad \textit{and} \quad s_1 = \frac{\sum_{i=1}^{n_1} C(n_0 + i)^2}{\left(\sum_{i=1}^{n_1} C(n_0 + i)\right)^2}, \tag{27}$$

The corresponding result for $E\left[\varepsilon_n^{C,1} \mid N_0 = n_0, C\right]$ is obtained by interchanging all indices 0 and 1.

Proof. See the Appendix.

It is easy to check that the result in Theorem 1 reduces to the one in (14) and (15) when $C = \mathbf{1}_n$. Following (16), we can then write

$$\begin{aligned} E\left[\varepsilon_n^C \mid C\right] &= \sum_{n_0=0}^{n} E\left[\varepsilon_n^C \mid N_0 = n_0, C\right] P(N_0 = n_0) \\ &= \sum_{n_0=0}^{n} \left(c_0 E\left[\varepsilon_n^{C,0} \mid N_0 = n_0, C\right] \right. \\ &\quad \left. + c_1 E\left[\varepsilon_n^{C,1} \mid N_0 = n_0, C\right]\right) P(N_0 = n_0). \end{aligned} \tag{28}$$

The expected bootstrap error rate $E[\hat{\varepsilon}_n^{\text{boot}}]$ can now be computed via (12).

The weight w^* for unbiased bootstrap error estimation can now be computed exactly by means of Equations (11), (12), (14) to (17), (20) to (22), and (25) to (28).

In the special case $\sigma_0 = \sigma_1 = \sigma$ (homoskedasticity), it follows easily from the previous expressions that $E[\varepsilon_n]$, $E[\hat{\varepsilon}_n^r]$, and $E[\hat{\varepsilon}_n^{\text{boot}}]$ depend only on the sample size n and on the Mahalanobis distance between the populations $\delta = |\mu_1 - \mu_0|/\sigma$, and therefore so does the weight w^*, through (11). Since the optimal (Bayes) classification error in this case is $\varepsilon^* = \Phi(-\delta/2)$, there is a one-to-one correspondence between Bayes error and the Mahalanobis distance. Therefore, in the homoskedastic case, the weight w^* *is a function only of the Bayes error ε^* and the sample size n.*

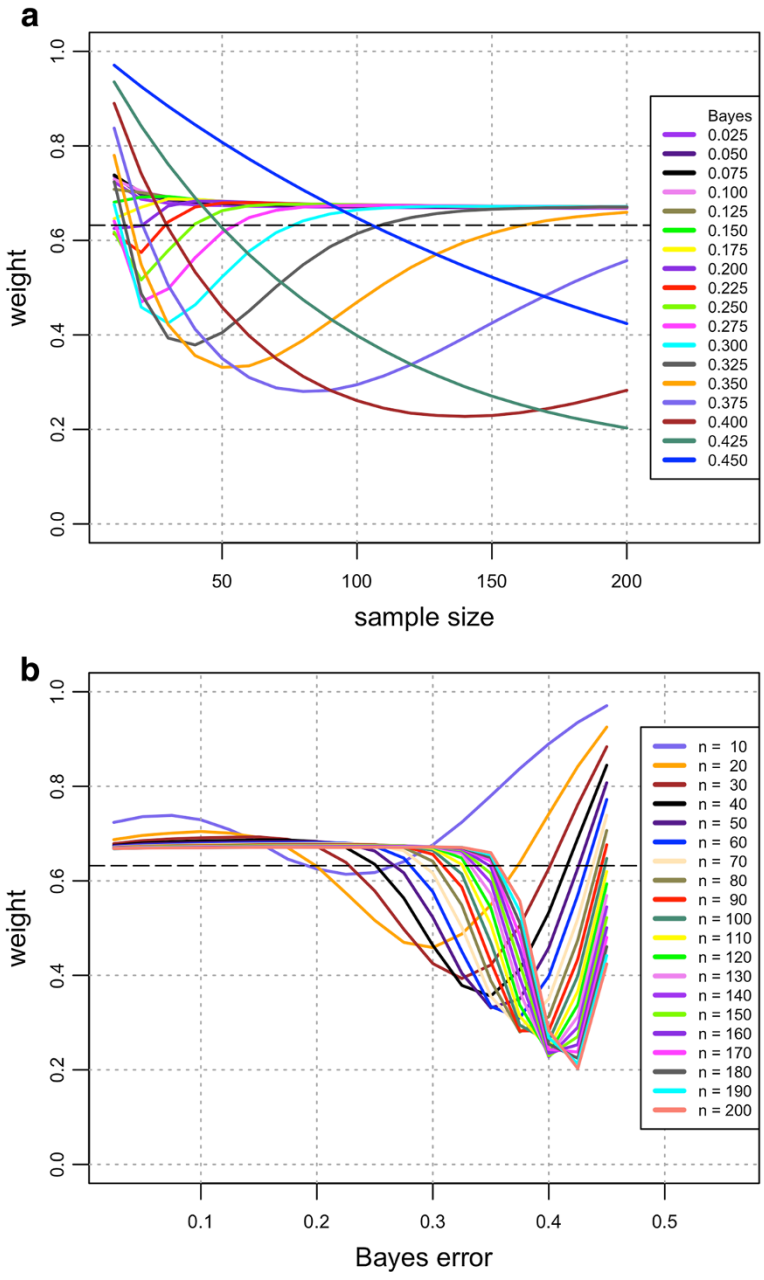

Figure 1 Univariate case. Required weight w^* for unbiased convex bootstrap estimation plotted against **(a)** sample size and **(b)** Bayes error.

Figure 1 and Table 1 display the value of w^* in the homoskedastic case, for several sample sizes and Bayes errors. In order to extend the plots up to $n = 200$, it is necessary to approximate $E[\hat{\varepsilon}_n^{\text{boot}}]$ in (12) by a Monte Carlo procedure; this is done by generating $M = 100 \times n^2$ independent random vectors $\{C_i \mid i = 1, \ldots, M\}$ and letting $E[\hat{\varepsilon}_n^{\text{boot}}] \approx (1/M) \sum_{i=1}^{M} E[\epsilon_n^{C_i} \mid C_i]$. We find that this value of M is large enough to obtain an accurate approximation. All other quantities are computed exactly, as described previously. One can see in Figure 1a that w^* varies wildly and can be very far from the heuristic 0.632 weight; however, as the sample size increases, w^* appears to settle around an asymptotic fixed value. This asymptotic value is approximately 0.675, being thus slightly larger than 0.632. In addition, Figure 1b allows one to see that convergence to the asymptotic value is faster for smaller Bayes errors. These facts help explain the good performance of the original convex 0.632 bootstrap error

Table 1 Univariate case: required weight w^* for unbiased convex bootstrap estimation

	$n = 10$	$n = 20$	$n = 30$	$n = 40$	$n = 50$	$n = 60$	$n = 70$	$n = 80$	$n = 90$	$n = 100$
$\epsilon^* = 0.025$	0.724	0.687	0.679	0.675	0.674	0.672	0.671	0.671	0.670	0.670
$\epsilon^* = 0.050$	0.736	0.696	0.685	0.680	0.678	0.676	0.674	0.673	0.672	0.672
$\epsilon^* = 0.075$	0.738	0.701	0.689	0.683	0.679	0.677	0.676	0.674	0.674	0.673
$\epsilon^* = 0.100$	0.729	0.704	0.691	0.684	0.681	0.678	0.677	0.675	0.674	0.673
$\epsilon^* = 0.125$	0.708	0.701	0.692	0.686	0.682	0.679	0.677	0.676	0.675	0.674
$\epsilon^* = 0.150$	0.681	0.692	0.693	0.687	0.683	0.680	0.678	0.677	0.676	0.675
$\epsilon^* = 0.175$	0.646	0.670	0.688	0.687	0.683	0.680	0.678	0.677	0.676	0.675
$\epsilon^* = 0.200$	0.625	0.631	0.673	0.683	0.683	0.681	0.679	0.677	0.676	0.675
$\epsilon^* = 0.225$	0.614	0.574	0.639	0.671	0.679	0.680	0.679	0.677	0.676	0.675
$\epsilon^* = 0.250$	0.617	0.516	0.579	0.635	0.663	0.673	0.676	0.677	0.676	0.675
$\epsilon^* = 0.275$	0.641	0.470	0.498	0.563	0.617	0.648	0.664	0.671	0.673	0.674
$\epsilon^* = 0.300$	0.676	0.459	0.425	0.464	0.523	0.577	0.616	0.641	0.656	0.665
$\epsilon^* = 0.325$	0.724	0.487	0.393	0.379	0.405	0.451	0.502	0.548	0.587	0.614
$\epsilon^* = 0.350$	0.780	0.549	0.422	0.356	0.331	0.334	0.356	0.389	0.428	0.469
$\epsilon^* = 0.375$	0.837	0.639	0.505	0.412	0.350	0.310	0.288	0.280	0.282	0.295
$\epsilon^* = 0.400$	0.890	0.741	0.626	0.533	0.458	0.398	0.350	0.312	0.283	0.261
$\epsilon^* = 0.425$	0.935	0.842	0.761	0.690	0.627	0.570	0.519	0.474	0.434	0.399
$\epsilon^* = 0.450$	0.971	0.925	0.884	0.845	0.808	0.772	0.739	0.707	0.676	0.647
	$n = 110$	$n = 120$	$n = 130$	$n = 140$	$n = 150$	$n = 160$	$n = 170$	$n = 180$	$n = 190$	$n = 200$
$\epsilon^* = 0.025$	0.669	0.669	0.669	0.669	0.669	0.669	0.669	0.668	0.668	0.668
$\epsilon^* = 0.050$	0.671	0.671	0.671	0.671	0.670	0.670	0.670	0.669	0.670	0.669
$\epsilon^* = 0.075$	0.672	0.672	0.671	0.671	0.671	0.671	0.670	0.670	0.670	0.670
$\epsilon^* = 0.100$	0.673	0.672	0.672	0.671	0.671	0.671	0.671	0.670	0.670	0.670
$\epsilon^* = 0.125$	0.673	0.673	0.672	0.672	0.672	0.671	0.671	0.671	0.670	0.670
$\epsilon^* = 0.150$	0.674	0.673	0.673	0.672	0.672	0.672	0.671	0.671	0.671	0.671
$\epsilon^* = 0.175$	0.674	0.673	0.673	0.672	0.672	0.672	0.672	0.671	0.671	0.671
$\epsilon^* = 0.200$	0.674	0.673	0.673	0.673	0.672	0.672	0.672	0.671	0.671	0.671
$\epsilon^* = 0.225$	0.675	0.674	0.673	0.672	0.672	0.672	0.672	0.672	0.671	0.671
$\epsilon^* = 0.250$	0.675	0.674	0.673	0.673	0.672	0.672	0.672	0.672	0.671	0.671
$\epsilon^* = 0.275$	0.674	0.674	0.673	0.673	0.673	0.673	0.672	0.671	0.671	0.671
$\epsilon^* = 0.300$	0.669	0.671	0.672	0.672	0.672	0.672	0.672	0.672	0.672	0.672
$\epsilon^* = 0.325$	0.635	0.648	0.657	0.663	0.666	0.668	0.669	0.670	0.671	0.671
$\epsilon^* = 0.350$	0.508	0.543	0.572	0.597	0.615	0.630	0.642	0.649	0.655	0.660
$\epsilon^* = 0.375$	0.313	0.337	0.365	0.394	0.425	0.455	0.484	0.511	0.536	0.557
$\epsilon^* = 0.400$	0.245	0.234	0.229	0.228	0.229	0.235	0.243	0.254	0.268	0.283
$\epsilon^* = 0.425$	0.367	0.338	0.313	0.290	0.270	0.253	0.238	0.224	0.213	0.203
$\epsilon^* = 0.450$	0.620	0.594	0.569	0.545	0.522	0.501	0.480	0.461	0.442	0.424

estimator with moderate sample sizes and small Bayes errors.

Multivariate case

Assume that population Π_i is distributed as a multivariate Gaussian $N(\mu_i, \Sigma)$, for $i = 0, 1$. Under these conditions, John obtained in [39] an exact expression for the expectation of the error of the LDA classification rule, defined by (2) to (4), for the case where $N_0 = n_0$ is fixed. This result is stated by Moran in [40] as follows:

$$E\left[\varepsilon_n^0 \mid N_0 = n_0\right] = P\left(\frac{W_1}{W_2} > \frac{1-\rho_e}{1+\rho_e}\right), \tag{29}$$

where W_1 and W_2 are independently distributed as non-central chi-square variables with d degrees of freedom

(d being the dimensionality) and noncentrality parameters λ_1 and λ_2, with

$$\lambda_1 = \frac{n_0 n_1}{2(1+\rho_e)} \left(\frac{1}{\sqrt{n_0+n_1}} - \frac{1}{\sqrt{n_0+n_1+4n_0n_1}} \right)^2 \delta^2,$$

$$\lambda_2 = \frac{n_0 n_1}{2(1-\rho_e)} \left(\frac{1}{\sqrt{n_0+n_1}} + \frac{1}{\sqrt{n_0+n_1+4n_0n_1}} \right)^2 \delta^2,$$

$$\rho_e = \frac{n_1-n_0}{\sqrt{(n_0+n_1)(n_0+n_1+4n_0n_1)}}, \tag{30}$$

where $\delta^2 = (\mu_1-\mu_0)^T \Sigma^{-1} (\mu_1-\mu_0)$ is the squared Mahalanobis distance between the populations. The corresponding result for $E[\varepsilon_n^1 \mid N_0 = n_0]$ is obtained by interchanging n_0 and n_1. The expected true error rate can then be found by using (16).

Moran also provided the following expression for the expectation of the resubstitution error estimator in the multivariate case, for fixed $N_0 = n_0$ [40]:

$$E\left[\hat{\varepsilon}_n^{r,0} \mid N_0 = n_0\right] = P\left(\frac{W_3}{W_4} > \frac{1-\rho_r}{1+\rho_r} \right), \tag{31}$$

where W_3 and W_4 are independently distributed as noncentral chi-square variables with d degrees of freedom and noncentrality parameters λ_3 and λ_4, with

$$\lambda_3 = \frac{n_0 n_1}{2(1+\rho_r)} \left(\frac{1}{\sqrt{n_0+n_1}} - \frac{1}{\sqrt{n_0-3n_1+4n_0n_1}} \right)^2 \delta^2,$$

$$\lambda_4 = \frac{n_0 n_1}{2(1-\rho_r)} \left(\frac{1}{\sqrt{n_0+n_1}} + \frac{1}{\sqrt{n_0-3n_1+4n_0n_1}} \right)^2 \delta^2,$$

$$\rho_r = -\sqrt{\frac{n_0+n_1}{n_0-3n_1+4n_0n_1}}, \tag{32}$$

The corresponding result for $E[\hat{\varepsilon}_n^{r,1}]$ is obtained by interchanging n_0 and n_1. The expected resubstitution error rate can then be found by using (22).

The bootstrap LDA classifier in the multivariate case is given by

$$\psi_n^C(X) = \begin{cases} 1, & \text{if } \left(X - \frac{\hat{\mu}_0^C + \hat{\mu}_1^C}{2}\right)^T \Sigma^{-1} \left(\hat{\mu}_0^C - \hat{\mu}_1^C\right) < 0 \\ 0, & \text{otherwise} \end{cases}, \tag{33}$$

where $\hat{\mu}_0^C$ and $\hat{\mu}_1^C$ are defined in (24). The next theorem generalizes John's result for the multivariate classification error to the case of the bootstrapped LDA classification rule.

Theorem 2. *Assume that population Π_i is distributed as $N(\mu_i, \Sigma)$, for $i = 0, 1$. Then, the expected error rate of the bootstrap LDA classification rule defined by (33) is given by*

$$E\left[\varepsilon_n^{C,0} \mid N_0 = n_0, C\right] = P\left(\frac{W_5}{W_6} > \frac{1-\rho_c}{1+\rho_c} \right), \tag{34}$$

where W_5 and W_6 are independently distributed as noncentral chi-square variables with d degrees of freedom and noncentrality parameters λ_5 and λ_6, with

$$\lambda_5 = \frac{1}{2(1+\rho_c)} \left(\frac{1}{\sqrt{s_0+s_1}} - \frac{1}{\sqrt{s_0+s_1+4}} \right)^2 \delta^2,$$

$$\lambda_6 = \frac{1}{2(1-\rho_c)} \left(\frac{1}{\sqrt{s_0+s_1}} + \frac{1}{\sqrt{s_0+s_1+4}} \right)^2 \delta^2,$$

$$\rho_c = \frac{s_0-s_1}{\sqrt{(s_0+s_1)(s_0+s_1+4)}}, \tag{35}$$

where s_0 and s_1 are defined in (27). The corresponding result for $E[\varepsilon_n^{C,1} \mid N_0 = n_0, C]$ is obtained by interchanging s_0 and s_1.

Proof. See the Appendix.

It is easy to check that the result in Theorem 2 reduces to the one in (29) and (30) when $C = \mathbf{1}_n$.

As in the univariate case, Theorem 2 can be used in conjunction with Equations (12) and (28) to compute $E[\hat{\varepsilon}_n^{\text{boot}}]$.

The weight w^* for unbiased bootstrap error estimation can now be computed exactly by means of Equations (11), (12), (16) to (17), (22), (28), (29) to (32), and (34) to (35).

An issue that arises in the multivariate case is the computation of the probabilities in (29), (31), and (34). This computation is very difficult since it involves the ratio of noncentral chi-square random variables, which has a doubly noncentral F distribution. Computation of this distribution is a hard problem. Moran proposes in [40] a complex procedure, based on work by Price [53], to compute this probability, which only applies to even dimensionality d. We employ a simpler procedure, namely, the Imhof-Pearson three-moment method, which is applicable to even and odd dimensionality [41]. This consists of approximating a noncentral $\chi^2_d(\lambda)$ random variable with a central χ^2_h random variable, by equating the first three moments of their distributions. This approach was also

employed in [52], where it was found to be very accurate. To fix ideas, we consider (29). The Imhof-Pearson three-moment approximation is given by

$$E\left[\varepsilon_n^0\right] = P\left(\frac{W_1}{W_2} > \frac{1-\rho_e}{1+\rho_e}\right) \simeq P\left(\chi_h^2 > y\right), \quad (36)$$

where χ_h^2 is a central chi-square random variable with h degrees of freedom, with

$$\begin{aligned} h &= \frac{c_2^3}{c_3^2}, \\ y &= h - c_1\sqrt{\frac{h}{c_2}}, \end{aligned} \quad (37)$$

and

$$\begin{aligned} c_i &= (1+\rho_e)^i (d+i\lambda_1) + (-1)^i (1-\rho_e)^i (d+i\lambda_2), \\ i &= 1,2,3. \end{aligned} \quad (38)$$

The approximation is valid only for $c_3 > 0$ [41]. If $c_3 < 0$, one uses the approximation

$$E\left[\varepsilon_n^0\right] = P\left(\frac{W_1}{W_2} > \frac{1-\rho_e}{1+\rho_e}\right) \simeq P\left(\chi_h^2 < y\right), \quad (39)$$

where h and y are as in (37), and

$$\begin{aligned} c_i &= (-1)^i (1+\rho_e)^i (d+i\lambda_1) + (1-\rho_e)^i (d+i\lambda_2), \\ i &= 1,2,3. \end{aligned} \quad (40)$$

The same approximation method applies to (31) and (34) by substituting the appropriate values.

As in the univariate case, the assumption of a common covariance matrix Σ makes the expectations $E[\varepsilon_n]$, $E[\hat{\varepsilon}_n^r]$, and $E[\hat{\varepsilon}_n^{\text{boot}}]$ and thus also the weight w^*, functions only of n and δ. Since $\varepsilon^* = \Phi(-\delta/2)$, this means that the weight w^* is a function only of the Bayes error ε^* and the sample size n.

Figure 2 and Table 2 display the value of w^* computed with the previous expressions in this section, for several sample sizes and Bayes errors. As in the univariate case, $E[\hat{\varepsilon}_n^{\text{boot}}]$ in (12) is approximated by a Monte Carlo procedure, with the same number $M = 100 \times n^2$ of MC vectors. All other quantities are computed exactly, as described previously, save for the Imhof-Pearson approximation. We can see in Figure 2 that there is considerable variation in the value of w^* and it can be far from the heuristic 0.632 weight; however, as the sample size increases, w^* appears to settle around an asymptotic fixed value. In contrast to the univariate case, these asymptotic values here appear to be strongly dependent on the Bayes error and are significantly smaller than the heuristic 0.632 except for very small Bayes errors. As in the univariate case, convergence to the apparent asymptotic value is faster for smaller Bayes errors. These facts again help explain the good performance of the original convex 0.632 bootstrap error estimator for moderate sample sizes and small Bayes errors.

Gene expression classification example

Here we demonstrate the application of the previous theory in comparing the performance of the bootstrap error estimator using the optimal weight versus the use of the fixed $w = 0.632$ weight, using gene expression data from the well-known breast cancer classification study in [42], which analyzed expression profiles from 295 tumor specimens, divided into $N_0 = 115$ specimens belonging to the 'good-prognosis' population (class 1 here) and $N_1 = 180$ specimens belonging to the 'poor-prognosis' population (class 0).

Our experiment was set up in the following way. We selected two genes among the previously published 70-gene prognosis profile [43]. These genes were selected for their approximate homoskedastic Gaussian distributions (see Figure 3). Since the real prior probabilities c_0 and c_1 for the good- and poor-prognosis populations are unknown, we assumed three different scenarios corresponding to $c_0 = 1/3$, $c_0 = 1/2$, and $c_0 = 2/3$ and *downsampled* randomly one or the other set of specimens to obtain new sample sizes $(90, 180)$, $(115, 115)$, and $(115, 68)$, respectively, so as to reflect the assumed prior probabilities. In each of the three cases, we then drew 2,000 random samples of size $n = 30$ from the pooled data, computed for each the true error, resubstitution, basic bootstrap, and convex bootstrap error rates. Bias and root-mean-square (RMS) error for each estimator were estimated by averaging over the 2,000 repetitions. We considered both the fixed 0.632 weight and the optimal weight prescribed by our analysis. For the latter, we estimated for each value of c_0 the Bayes error using the full data set and read off Table 2 the optimal weight corresponding to the estimated Bayes error and sample size $n = 30$. The results are displayed in Table 3. Despite the approximate nature of the results, given that the simulated training samples are not independent from each other, we can see that the bias and RMS were always smaller for the estimator using the optimal weight than using the fixed 0.632 weight (all bootstrap estimators vastly outperforming resubstitution).

Conclusions

Exact expressions were derived for the required weight for unbiased convex bootstrap error estimation in the finite sample case, for linear discriminant analysis of Gaussian populations. The results not only provide the practitioner with a recommendation of what weight to use

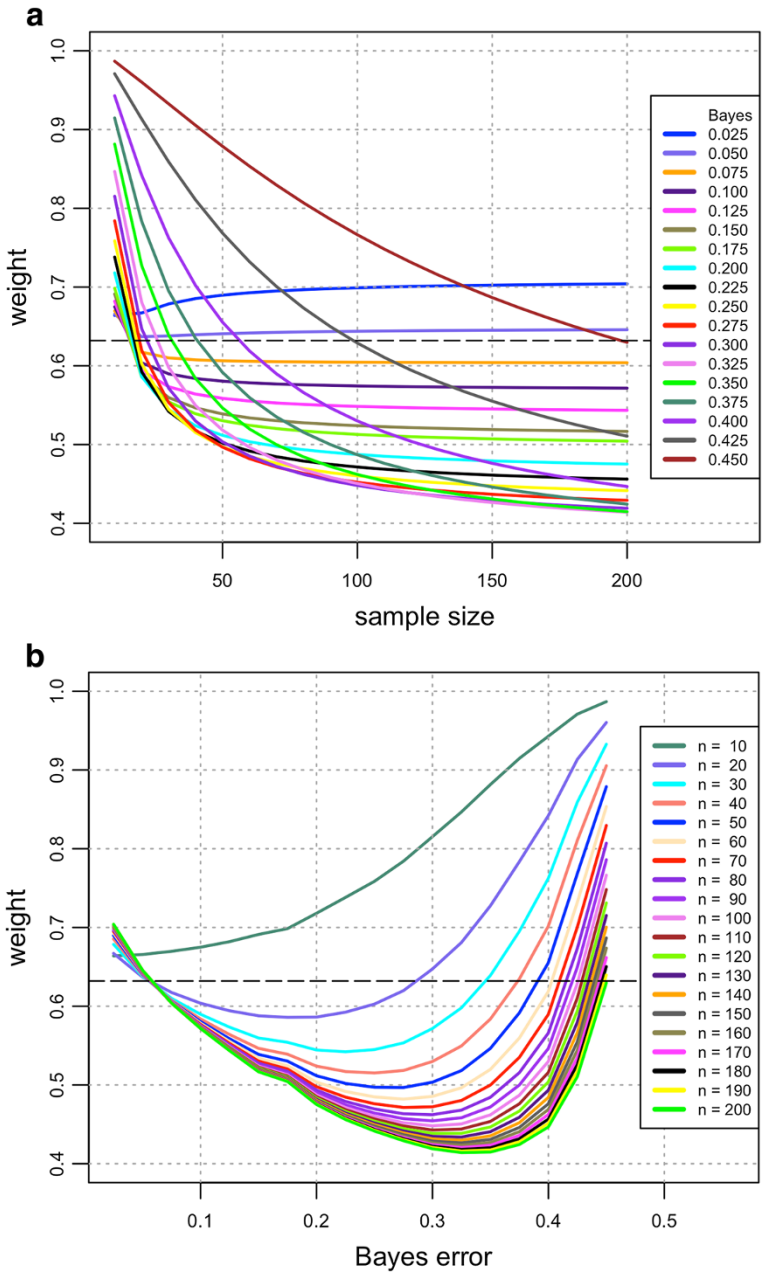

Figure 2 Bivariate case. Required weight w^* for unbiased convex bootstrap estimation plotted against **(a)** sample size and **(b)** Bayes error.

given the sample size and problem difficulty, but also offer insight into the choice of the 0.632 weight for the classic 0.632 bootstrap error estimator. It was observed that the required weight for unbiasedness can deviate significantly from the 0.632 weight, particularly in the multivariate case, where the required weight for unbiasedness appears to settle on an asymptotic value that is strongly dependent on the Bayes error, being as a rule smaller than 0.632. The results were illustrated by application to gene expression data from a well-known breast cancer study.

Appendix

Proof of Theorem 1

Following the same technique used in [40], we write

$$\begin{aligned} E\left[\varepsilon_C^0 \mid C\right] &= P\left(\psi_n^C(X) = 1 \mid X \in \Pi_0, C\right) \\ &= P\left(\widehat{\mu}_1^C > \widehat{\mu}_0^C, X > \frac{\widehat{\mu}_0^C + \widehat{\mu}_1^C}{2} \mid X \in \Pi_0, C\right) \\ &\quad + P\left(\widehat{\mu}_1^C \leq \widehat{\mu}_0^C, X \leq \frac{\widehat{\mu}_0^C + \widehat{\mu}_1^C}{2} \mid X \in \Pi_0, C\right) \\ &= P(UV > 0 \mid X \in \Pi_0, C), \end{aligned} \tag{41}$$

Table 2 Bivariate case: required weight w^* for unbiased convex bootstrap estimation

	$n = 10$	$n = 20$	$n = 30$	$n = 40$	$n = 50$	$n = 60$	$n = 70$	$n = 80$	$n = 90$	$n = 100$
$\epsilon^* = 0.025$	0.664	0.667	0.679	0.685	0.690	0.693	0.695	0.697	0.698	0.699
$\epsilon^* = 0.050$	0.666	0.637	0.638	0.639	0.641	0.642	0.642	0.643	0.644	0.644
$\epsilon^* = 0.075$	0.670	0.617	0.610	0.608	0.606	0.606	0.605	0.605	0.605	0.605
$\epsilon^* = 0.100$	0.675	0.604	0.590	0.584	0.581	0.578	0.577	0.576	0.575	0.574
$\epsilon^* = 0.125$	0.682	0.594	0.573	0.564	0.559	0.555	0.553	0.551	0.550	0.548
$\epsilon^* = 0.150$	0.691	0.588	0.560	0.547	0.539	0.534	0.530	0.528	0.526	0.524
$\epsilon^* = 0.175$	0.699	0.586	0.554	0.539	0.530	0.524	0.520	0.517	0.515	0.513
$\epsilon^* = 0.200$	0.718	0.586	0.544	0.524	0.512	0.504	0.498	0.493	0.490	0.487
$\epsilon^* = 0.225$	0.738	0.592	0.542	0.517	0.502	0.492	0.485	0.479	0.475	0.471
$\epsilon^* = 0.250$	0.759	0.603	0.545	0.515	0.497	0.485	0.476	0.469	0.464	0.460
$\epsilon^* = 0.275$	0.784	0.620	0.553	0.518	0.497	0.482	0.471	0.463	0.457	0.452
$\epsilon^* = 0.300$	0.815	0.647	0.572	0.530	0.503	0.485	0.472	0.462	0.454	0.448
$\epsilon^* = 0.325$	0.847	0.681	0.598	0.550	0.518	0.496	0.480	0.468	0.458	0.450
$\epsilon^* = 0.350$	0.882	0.728	0.639	0.584	0.546	0.520	0.500	0.484	0.472	0.462
$\epsilon^* = 0.375$	0.915	0.784	0.695	0.635	0.592	0.560	0.535	0.516	0.500	0.487
$\epsilon^* = 0.400$	0.943	0.842	0.763	0.702	0.655	0.619	0.590	0.566	0.546	0.530
$\epsilon^* = 0.425$	0.971	0.914	0.859	0.811	0.769	0.732	0.701	0.673	0.650	0.629
$\epsilon^* = 0.450$	0.987	0.960	0.933	0.905	0.879	0.853	0.830	0.807	0.786	0.766
	$n = 110$	$n = 120$	$n = 130$	$n = 140$	$n = 150$	$n = 160$	$n = 170$	$n = 180$	$n = 190$	$n = 200$
$\epsilon^* = 0.025$	0.700	0.701	0.701	0.702	0.702	0.703	0.703	0.704	0.704	0.704
$\epsilon^* = 0.050$	0.644	0.645	0.645	0.645	0.645	0.645	0.645	0.646	0.646	0.646
$\epsilon^* = 0.075$	0.604	0.604	0.604	0.604	0.604	0.604	0.604	0.604	0.604	0.604
$\epsilon^* = 0.100$	0.574	0.573	0.573	0.573	0.573	0.572	0.572	0.572	0.572	0.572
$\epsilon^* = 0.125$	0.548	0.547	0.546	0.546	0.545	0.545	0.544	0.544	0.544	0.543
$\epsilon^* = 0.150$	0.523	0.522	0.521	0.520	0.519	0.518	0.518	0.517	0.517	0.517
$\epsilon^* = 0.175$	0.511	0.510	0.509	0.508	0.507	0.506	0.506	0.505	0.505	0.504
$\epsilon^* = 0.200$	0.485	0.483	0.482	0.480	0.479	0.478	0.477	0.477	0.476	0.475
$\epsilon^* = 0.225$	0.469	0.466	0.464	0.463	0.461	0.460	0.459	0.458	0.457	0.456
$\epsilon^* = 0.250$	0.457	0.454	0.452	0.449	0.448	0.446	0.445	0.443	0.442	0.441
$\epsilon^* = 0.275$	0.448	0.444	0.442	0.439	0.437	0.435	0.433	0.432	0.430	0.429
$\epsilon^* = 0.300$	0.443	0.438	0.435	0.432	0.429	0.426	0.424	0.422	0.420	0.419
$\epsilon^* = 0.325$	0.444	0.439	0.434	0.430	0.426	0.423	0.421	0.418	0.416	0.414
$\epsilon^* = 0.350$	0.454	0.447	0.441	0.435	0.431	0.427	0.423	0.420	0.417	0.415
$\epsilon^* = 0.375$	0.476	0.467	0.459	0.452	0.446	0.441	0.436	0.432	0.428	0.424
$\epsilon^* = 0.400$	0.516	0.504	0.493	0.484	0.476	0.469	0.462	0.457	0.451	0.447
$\epsilon^* = 0.425$	0.611	0.594	0.580	0.567	0.555	0.544	0.535	0.526	0.518	0.511
$\epsilon^* = 0.450$	0.748	0.731	0.715	0.700	0.687	0.674	0.662	0.650	0.640	0.630

where $U = \widehat{\mu}_1^C - \widehat{\mu}_0^C$ and $V = X - \frac{\widehat{\mu}_0^C + \widehat{\mu}_1^C}{2}$. From (303), it is clear that, given C, $\hat{\mu}_0^C$ and $\hat{\mu}_1^C$ are independent Gaussian random variables, such that $\hat{\mu}_i^C \sim N(\mu_i, s_i\sigma_i^2)$, for $i = 0, 1$, where s_1 and s_2 are defined in (27). It follows that U and V are jointly Gaussian random variables, with the following parameters:

$$
\begin{aligned}
&E[U \mid X \in \Pi_0, C] = \mu_1 - \mu_0, \mathrm{Var}(U \mid X \in \Pi_0, C) = s_0\sigma_0^2 + s_1\sigma_1^2,\\
&E[\, V \mid X \in \Pi_0, C] = \frac{\mu_0 - \mu_1}{2},\\
&\mathrm{Var}(V \mid X \in \Pi_0, C) = \left(1 + \frac{s_0}{4}\right)\sigma_0^2 + \frac{s_1}{4}\sigma_1^2, \qquad (42)\\
&\mathrm{Cov}(U, V \mid X \in \Pi_0, C) = \frac{s_0\sigma_0^2 - s_1\sigma_1^2}{2}.
\end{aligned}
$$

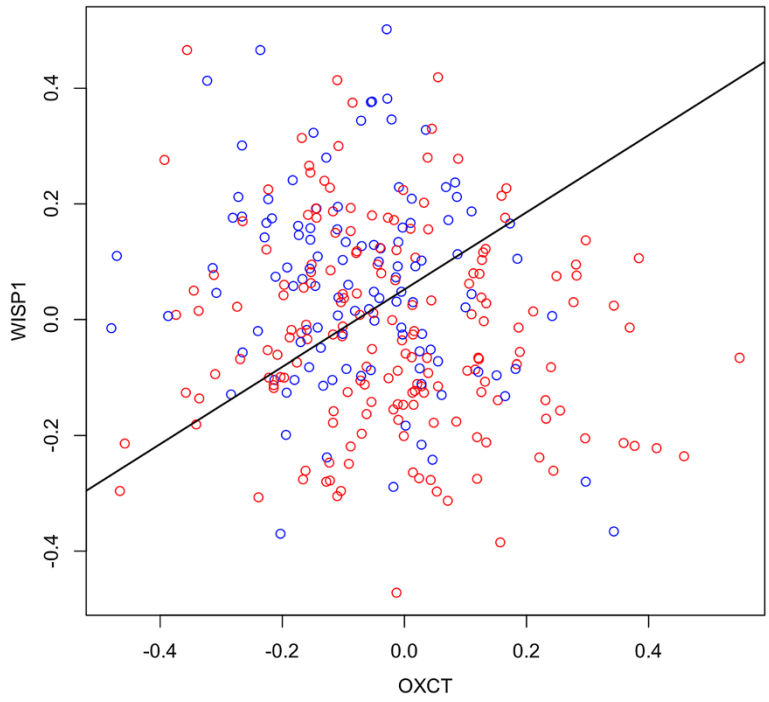

Figure 3 Data used in the gene expression experiment. The plot shows the optimal (linear) classifier superimposed on the sample for the genes OXCT and WISP1, from the breast cancer study in [42]. We can see that both populations are approximately Gaussian with equal dispersion. Bad prognosis = red. Good prognosis = blue.

The result then follows after some algebraic manipulation. By symmetry, to obtain $E[\,\varepsilon_C^1 \mid C]$, one needs only to interchange all indices 0 and 1. □

Proof of Theorem 2

Following the same technique used in [32], we write

$$\begin{aligned} E[\,\varepsilon_C^0 \mid C] &= P(\psi_n^C(X) = 1 \mid X \in \Pi_0, C) \\ &= P\left((\hat{\mu}_1^C - \hat{\mu}_0^C)^T \Sigma^{-1}\left(X - \frac{\hat{\mu}_0^C + \hat{\mu}_1^C}{2}\right)\right. \\ &\left. > 0 \mid X \in \Pi_0, C\right) = P\left(U^T V > 0 \mid X \in \Pi_0, C\right) \\ &= P((U+V)^T(U+V) - (U-V)^T(U-V) \\ &> 0 \mid X \in \Pi_0, C) \\ &= P\left(\frac{(U+V)^T(U+V)}{(U-V)^T(U-V)} > 1 \mid X \in \Pi_0, C\right), \end{aligned} \tag{43}$$

where $U = (s_0 + s_1)^{-\frac{1}{2}} \Sigma^{-\frac{1}{2}} (\hat{\mu}_1^C - \hat{\mu}_0^C)$ and $V = 2\,(s_0 + s_1 + 4)^{-\frac{1}{2}} \Sigma^{-\frac{1}{2}} \left(X - \frac{\hat{\mu}_0^C + \hat{\mu}_1^C}{2}\right)$. It can be readily checked that $U + V$ and $U - V$ are independent Gaussian random vectors, such that

$$\begin{aligned} E\,[U+V \mid X \in \Pi_0, C] &= \left[(s_0+s_1)^{-\frac{1}{2}} - (s_0+s_1+4)^{-\frac{1}{2}}\right] \\ &\quad \times \Sigma^{-1/2}(\mu_1 - \mu_0), \\ E\,[U-V \mid X \in \Pi_0, C] &= \left[(s_0+s_1)^{-\frac{1}{2}} + (s_0+s_1+4)^{-\frac{1}{2}}\right] \\ &\quad \times \Sigma^{-1/2}(\mu_1 - \mu_0)\,, \\ \Sigma_{U+V} \mid X \in \Pi_0, C &= 2(1+\rho_c)I\,, \quad \Sigma_{U-V} \mid X \in \Pi_0, \\ C &= 2(1-\rho_c)I\,, \end{aligned} \tag{44}$$

Table 3 Bias and RMS of estimators considered in the experiment with expression data from genes 'OXCT' and 'WISP1'

c_0	n	ε^*	$E[\,\varepsilon_n]$	Resub		Basic boot		Opt boot		0.632 boot	
				Bias	RMS	Bias	RMS	Bias	RMS	Bias	RMS
0.33	30	0.4043	0.4206	−0.0702	0.1061	0.0008	0.0820	−0.0161	0.0803	−0.0253	0.0817
0.50	30	0.3969	0.4266	−0.0719	0.1060	0.0072	0.0830	−0.0116	0.0798	−0.0219	0.0806
0.67	30	0.3893	0.4131	−0.0914	0.1185	−0.0181	0.0878	−0.0355	0.0885	−0.0451	0.0909

Also displayed are the assumed values for the prior probability c_0, sample size n, the estimated value of the Bayes error ε^*, and the expected classification error $E[\varepsilon_n]$.

where ρ_c is defined as in (35) and I denotes the identity matrix of dimension d. It follows that

$$W_5 = \frac{1}{2(1+\rho_c)}(U+V)^T(U+V)\,,$$
$$W_6 = \frac{1}{2(1-\rho_c)}(U-V)^T(U-V) \tag{45}$$

are independent noncentral chi-squared random variables with d degrees of freedom and noncentrality parameters λ_5 and λ_6 defined in (35). The result then follows from (62). Following along the same lines, one can show that $E[\,\varepsilon_C^1 \mid C]$ is obtained by interchanging s_0 and s_1 in the result for $E[\,\varepsilon_C^0 \mid C]$ (the details are omitted for brevity). □

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
TV proved Theorems 1 and 2. TV and SC conducted numerical experiments to compute Figures 1 and 2 and Tables 1 and 2. SC conducted the numerical experiments with gene expression data. UMB conceived the study and wrote the first draft of the manuscript. ERD contributed ideas on convex error estimation and revised the manuscript. All authors read and approved the final manuscript.

Acknowledgements
The authors acknowledge the support of the National Science Foundation, through NSF awards CCF-0845407 (Braga-Neto) and CCF-0634794 (Dougherty).

Received: 17 February 2014 Accepted: 18 August 2014
Published online: 03 October 2014

References
1. B Efron, Bootstrap methods: another look at the jackknife. Ann. Stat. **7**(1), 1–26 (1979). [Online]. http://projecteuclid.org/euclid.aos/1176344552
2. B Efron, Computers and the theory of statistics: thinking the unthinkable. SIAM Rev. **21**(4), 460–480 (1979). [Online]. http://www.jstor.org/stable/2030104
3. B Efron, Nonparametric standard errors and confidence intervals. Can. J. Stat. **9**(2), 139–158 (1981)
4. B Efron, Estimating the error rate of a prediction rule: improvement on cross-validation. J. Am. Stat. Assoc. **78**(382), 316–331 (1983). [Online]. http://dx.doi.org/10.2307/2288636
5. B Efron, G Gong, A leisurely look at the bootstrap, the jackknife, and cross-validation. Am. Stat. **37**(1), 36–48 (1983). [Online]. http://dx.doi.org/10.2307/2685844
6. B Efron, R Tibshirani, *An Introduction to the Bootstrap*. (Chapman & Hall, New York, 1993)
7. B Efron, R Tibshirani, Improvements on cross-validation: the .632+ bootstrap method. J. Am. Stat. Assoc. **92**(438), 548–560 (1997). [Online]. http://dx.doi.org/10.2307/2965703
8. K Singh, On the asymptotic accuracy of Efron's bootstrap. Ann. Stat. **9**, 1187–1195 (1981)
9. P Bickel, D Freedman, Some asymptotic theory for the bootstrap. Ann. Stat. **9**, 1196–1217 (1981)
10. R Beran, Estimated sampling distributions: the bootstrap and competitors. Ann. Stat. **10**(1), 212–225 (1982). [Online]. http://www.jstor.org/stable/2240513
11. P Hall, *The Bootstrap and Edgeworth Expansion*. (Springer, New York, 1992)
12. F Scholz, *The Bootstrap Small Sample Properties*. (University of, Washington, Seattle, 2007)
13. P Porter, S Rao, J-Y Ku, R Poirot, M Dakins, Small sample properties of nonparametric bootstrap t confidence intervals. J. Air Waste Manag. Assoc. **47**(11), 1197–1203 (1997)
14. K Chan, S Lee, An exact iterated bootstrap algorithm for small-sample bias reduction. Comput. Stat. Data Anal. **36**(1), 1–13 (2001)
15. G Young, Bootstrap: more than a stab in the dark? With discussion and a rejoinder by the author. Stat. Sci. **9**(3), 382–415 (1994)
16. J Shao, D Tu, *The Jackknife and Bootstrap*. (Springer, New York, 1995). [Online]. http://www.worldcat.org/isbn/0387945156
17. D Pils, D Tong, G Hager, E Obermayr, S Aust, G Heinze, M Kohl, E Schuster, A Wolf, J Sehouli, I Braicu, I Vergote, T Van Gorp, S Mahner, N Concin, P Speiser, R Zeillinger, A combined blood based gene expression and plasma protein abundance signature for diagnosis of epithelial ovarian cancer–a study of the OVCAD consortium. BMC Cancer. **13**(178) (2013). doi: 10.1186/1471-2407-13-178
18. S Paul, P Maji, muHEM for identification of differentially expressed miRNAs using hypercuboid equivalence partition matrix. BMC Bioinformatics. **14**(266) (2013). doi:10.1186/1471-2105-14-266
19. S Student, K Fujarewicz, Stable feature selection and classification algorithms for multiclass microarray data. Biol Direct. **7**, 33 (2012). doi:10.1186/1745-6150-7-33
20. T Hwang, CH Sun, T Yun, GS Yi, FiGS: a filter-based gene selection workbench for microarray data. BMC Bioinformatics. **11**(50) (2010). doi:10.1186/1471-2105-11-50
21. G McLachlan, *Discriminant Analysis and Statistical Pattern Recognition*. (Wiley, New York, 1992)
22. L Devroye, L Gyorfi, G Lugosi, *A Probabilistic Theory of Pattern Recognition*. (Springer, New York, 1996)
23. C Sima, E Dougherty, Optimal convex error estimators for classification. Pattern Recognit. **39**(6), 1763–1780 (2006)
24. M Chernick, V Murthy, C Nealy, Application of bootstrap and other resampling techniques: evaluation of classifier performance. Pattern Recognit. Lett. **3**(3), 167–178 (1985). [Online] http://www.sciencedirect.com/science/article/B6V15-48MPVCK-55/2/32754228bc17ac0655b9fa9a7a60ca90
25. K Fukunaga, R Hayes, Estimation of classifier performance. IEEE Trans. Pattern Anal. Mach. Intell. **11**(10), 1087–1101 (1989)
26. G McLachlan, Error rate estimation in discriminant analysis: recent advancesAdv. Multivariate Stat. Anal, 233–252 (1987)
27. A Davison, P Hall, On the bias and variability of bootstrap and cross-validation estimates of error rate in discrimination problems. Biometrika. **79**(2), 279–284 (1992). [Online] http://www.jstor.org/stable/2336839
28. M Chernick, *Bootstrap Methods: A Guide for Practitioners and Researchers (Wiley Series in Probability and Statistics)*, 2nd ed. (Wiley-Interscience, Hoboken, 2007). [Online]. http://www.worldcat.org/isbn/0471756210
29. S Chatterjee, S Chatterjee, Estimation of misclassification probabilities by bootstrap methods. Comput. **12**, 645–656 (1983)
30. A Jain, R Dubes, C Chen, Bootstrap techniques for error estimation. IEEE Trans. Pattern Anal. Mach. Intell. **9**(5), 628–633 (1987)
31. S Raudys, On the accuracy of a bootstrap estimate of the classification erro, in *Proceedings of Ninth International Joint Conference on Pattern Recognition,* (Rome 14–17 Nov 1988, p. 1230–1232(1988)
32. U Braga-Neto, E Dougherty, Bolstered error estimation. Pattern Recognit. **37**(6), 1267–1281 (2004). [Online] http://www.sciencedirect.com/science/article/B6V14-4BNMG7H-1/2/752fe2e9105d351b8850e48577ba182c
33. U Braga-Neto, R Hashimoto, E Dougherty, D Nguyen, R Carroll, Is cross-validation better than re-substitution for ranking genes? Bioinformatics. **20**(2), 253–258 (2004). [Online] http://bioinformatics.oxfordjournals.org/cgi/content/abstract/20/2/253
34. U Braga-Neto, E Dougherty, Is cross-validation valid for small-sample microarray classification? Bioinformatics. **20**(3), 374–380 (2004). [Online]. http://bioinformatics.oxfordjournals.org/cgi/content/abstract/20/3/374
35. R Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection. (IJCAI), 1137–1145 (1995). [Online]. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.48.529
36. G Toussaint, An efficient method for estimating the probability of misclassification applied to a problem in medical diagnosis. Comput. Biol. Med. **4**, 269 (1975)

37. G McLachlan, A note on the choice of a weighting function to give an efficient method for estimating the probability of misclassification. Pattern Recognit. **9**(2), 147–149 (1977)
38. S Raudys, A Jain, Small sample size effects in statistical pattern recognition: recommendations for practitioners. IEEE Trans. Pattern Anal. Mach. Intell. **13**(3), 4–37 (1991)
39. S John, Errors in discrimination. Ann. Math. Stat. **32**(4), 1125–1144 (1961). [Online]. http://www.jstor.org/stable/2237911
40. M Moran, On the expectation of errors of allocation associated with a linear discriminant function. Biometrika. **62**(1), 141–148 (1975). [Online]. http://www.jstor.org/stable/2334496
41. J Imhof, Computing the distribution of quadratic forms in normal variables. Biometrika. **48**(3/4), 419–426 (1961)
42. MJ van de Vijver, YD He, LJ van't Veer, H Dai, AAM Hart, DW Voskuil, GJ Schreiber, JL Peterse, C Roberts, MJ Marton, M Parrish, D Astma, A Witteveen, A Glas, L Delahaye, T van der Velde, H Bartelink, S Rodenhuis, ET Rutgers, SH Friend, R Bernards, A gene-expression signature as a predictor of survival in breast cancer. N. Engl. J. Med. **347**(25), 1999–2009 (2002)
43. LJ van't Veer, H Dai, MJ van de Vijver, YD He, AAM Hart, M Mao, HL Peterse, K van der Kooy, MJ Marton, AT Witteveen, GJ Schreiber, RM Kerkhoven, C Roberts, PS Linsley, R Bernards, SH Friend, Gene expression profiling predicts clinical outcome of breast cancer. Nature. **415**, 530–536 (2002)
44. UM Braga-Neto, A Zollanvari, ER Dougherty, Cross-validation under separate sampling: strong bias and how to correct it. Bioinformatics (2014). doi:10.1093/bioinformatics/btu527
45. T Anderson, Classification by multivariate analysis. Psychometrika. **16**, 31–50 (1951)
46. S Raudys, Comparison of the estimates of the probability of misclassification, in *Proc. 4th Int. Conf. Pattern Recognition* Kyoto, Japan, 1978), pp. 280–282
47. L Breiman, Bagging predictors. Mach. Learn. **24**(2), 123–140 (1996)
48. T Vu, U Braga-Neto, Is bagging effective in the classification of small-sample genomic and proteomic data?. URASIP J. Bioinformatics Syst. Biol. **2009**, Article ID 158368 (2009)
49. V Vapnik, *Statistical Learning Theory*. (Wiley, New York, 1998)
50. A Nijenhuis, H Wilf, *Combinatorial Algorithms,* 2nd ed. (Academic Press, New York, 1978)
51. M Hills, Allocation rules and their error rates. J. R. Stat. Soc. Series B (Methodological). **28**(1), 1–31 (1966). [Online]. http://www.jstor.org/stable/2984268
52. A Zollanvari, U Braga-Neto, E Dougherty, On the sampling distribution of resubstitution and leave-one-out error estimators for linear classifiers. Pattern Recognit. **42**(11), 2705–2723 (2009)
53. R Price, Some non-central *f*-distributions expressed in closed form. Biometrika. **51**, 107–122 (1964)

doi:10.1186/s13637-014-0015-0
Cite this article as: Vu *et al.*: **Unbiased bootstrap error estimation for linear discriminant analysis.** *EURASIP Journal on Bioinformatics and Systems Biology* 2014 **2014**:15.