BIOINFORMATICS ORIGINAL PAPER

Vol. 27 no. 13 2011, pages 1822–1831
doi:10.1093/bioinformatics/btr272

Gene expression

Advance Access publication May 5, 2011

Application of the Bayesian MMSE estimator for classification error to gene expression microarray data

Lori A. Dalton[1,*] and Edward R. Dougherty[1,2,3]

[1]Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, [2]Computational Biology Division,Translational Genomics Research Institute, Phoenix, AZ 85004 and [3]Department of Bioinformatics and Computational Biology, University of Texas M. D. Anderson Cancer Center, Houston, TX 77030 USA

Associate Editor: Joaquin Dopazo

ABSTRACT

Motivation: With the development of high-throughput genomic and proteomic technologies, coupled with the inherent difficulties in obtaining large samples, biomedicine faces difficult small-sample classification issues, in particular, error estimation. Most popular error estimation methods are motivated by intuition rather than mathematical inference. A recently proposed error estimator based on Bayesian minimum mean square error estimation places error estimation in an optimal filtering framework. In this work, we examine the application of this error estimator to gene expression microarray data, including the suitability of the Gaussian model with normal–inverse-Wishart priors and how to find prior probabilities.

Results: We provide an implementation for non-linear classification, where closed form solutions are not available. We propose a methodology for calibrating normal-inverse-Wishart priors based on discarded microarray data and examine the performance on synthetic high-dimensional data and a real dataset from a breast cancer study. The calibrated Bayesian error estimator has superior root mean square performance, especially with moderate to high expected true errors and small feature sizes.

Availability: We have implemented in C code the Bayesian error estimator for Gaussian distributions and normal–inverse-Wishart priors for both linear classifiers, with exact closed-form representations, and arbitrary classifiers, where we use a Monte Carlo approximation. Our code for the Bayesian error estimator and a toolbox of related utilities are available at http://gsp.tamu.edu/Publications/supplementary/dalton11a. Several supporting simulations are also included.

Contact: ldalton@tamu.edu

Supplementary Information: Supplementary data are available at *Bioinformatics* online.

Received on January 25, 2011; revised on April 8, 2011; accepted on April 24, 2011

1 INTRODUCTION

Classification is a major constituent of bioinformatics, in particular, phenotypic discrimination, which can be accomplished via many different data types, such as gene expression, protein expression or sequence data. The misclassification error of a classifier quantifies its predictive capacity, the key aspect of any scientific model. Thus, accuracy of the error estimation represents the salient epistemological issue in classification, model validity (Dougherty and Braga-Neto, 2006). The main measure of error estimation accuracy is the root mean square (RMS) error of the estimator,

$$\mathrm{RMS} = \sqrt{\mathrm{E}\left[\left(\varepsilon_{\mathrm{tru}} - \varepsilon_{\mathrm{est}}\right)^2\right]},$$

where $\varepsilon_{\mathrm{tru}}$ and $\varepsilon_{\mathrm{est}}$ are the true and estimated errors of the classifier and E is expectation with respect to the random sampling procedure. Given a large data sample, the data can be split between training and test data, the classifier designed on the training data and classifier error estimated on the test data. In this scenario, there is a satisfactory distribution-free bound, $\mathrm{RMS} \leq 1/2\sqrt{m}$, where m is the size of the test sample (Devroye *et al.*, 1996). However, when the sample is small, splitting the data is unacceptable because the classifier will be trained on too small a set, thereby resulting in poor classifier design. Thus, in small sample settings (the concern of this article), a classifier is trained and its error estimated on the same data.

A number of training data-based error estimators have been proposed in the past and we will consider several in this article. Perhaps the one most commonly employed in bioinformatics is cross-validation. In this method, the data are partitioned into k folds (subsets); at each state of the procedure, one fold is held out, a surrogate classifier trained on the remaining folds and its error estimated on the held-out fold. The error of the classifier (originally trained on the full sample) is estimated by the average surrogate errors on the left-out folds. In the special case $k=n$, the sample size, each held-out fold consists of one point and the error method is termed 'leave-one-out'. For leave-one-out, there is only one partition of folds; however, when $k<n$ evaluating all combinations of partitions is computationally prohibitive. Hence, in this case partitions are randomly chosen to make the estimation. Although cross-validation is close to being unbiased if k is not too small, it tends to have a large variance for small samples (Braga-Neto and Dougherty, 2004b; Devroye *et al.*, 1996) and also to be poorly correlated with the true error (Hanczar *et al.*, 2007), the two combining to create a large RMS for small samples [for a review of error estimation performance, see Dougherty *et al.* (2010)].

A natural question arises: can cross-validation be used for small samples or, equivalently, are there small-sample cases in which the RMS of cross-validation is sufficiently small so that it can be considered a valid error estimator? To answer this question, one might first ask if it is possible to use distribution-free bounds. Not

*To whom correspondence should be addressed.

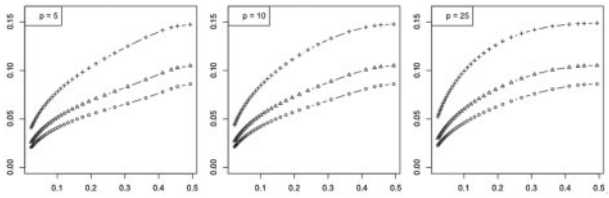


Fig. 1. Leave-one-out RMS versus Bayes error for LDA. (plus) is 20 samples, (triangle) 40 samples and (circle) 60 samples.

only are there very few cases in which such bounds are known, but also when they are known they are so loose as to be useless in practice. For instance, consider the following leave-one-out RMS bound for the k-nearest neighbor classification rule with random tie breaking (Devroye and Wagner, 1979):

$$\text{RMS} \leq \sqrt{\left(1+24\sqrt{\frac{k}{2\pi}}\right)\frac{1}{n}}.$$

If $k=3$ and the sample size is $n=100$, then the bound is approximately 0.353, which is useless.

Let us now consider bounds when there are distributional assumptions. We consider a feature-label distribution having two equally probable Gaussian class-conditional densities sharing a known covariance matrix and the linear discriminant analysis (LDA) classification rule. For this model, we possess analytic representation of the joint distribution of the true error with the leave-one-out estimator (Zollanvari *et al.*, 2010). Figure 1 shows the RMS to be a one-to-one increasing function of the Bayes error for dimensions $p=5,10,25$, and sample sizes $n=20,40,60$, the RMS and Bayes errors being on the y and x axes, respectively. In this model, where the Bayes error is a function of the distance between the class-conditional means, the maximum RMS is bounded and does not exceed 0.15, even with only 20 sample points. Moreover, if one wishes to bound the RMS below some tolerance, τ, one need to only make an assumption on the minimum distance between the means, which corresponds to a maximum Bayes error. This kind of behavior, where the RMS of leave-one-out is tolerable when the Bayes error is small, is often observed—indeed, we see this in Figure 5 of this article—but it has only been quantified in a small number of cases (Braga-Neto and Dougherty, 2010; Zollanvari *et al.*, 2010).

The upshot of these considerations is that if cross-validation is going to be used when the sample size is small, there must be modeling assumptions to make the RMS acceptable. Hence, why not take a Bayesian minimum mean square error (MMSE) approach and thereby guarantee that the average RMS across the model family for the resulting error estimator is minimal among all possible error estimators? That is what is done in Dalton and Dougherty (2011a, b), where a parameterized family of class-conditional feature distributions is assumed, a prior distribution is applied to the parameters of the model, and this prior along with observed data are used to compute an unbiased, MMSE estimate of classification error. An advantage of this approach, besides achieving average minimum RMS across the model family, is that it depends only on the form of the designed classifier, not the classification rule used to design the classifier. In particular, it is independent of the feature selection method, which is part of the classification rule.

Two problems naturally arise. First, how does one arrive at a prior distribution governing the model? This issue arises in any Bayesian method and, as previously explained, would arise in the context of small-sample error estimation even if one were to use a classical error estimator. The current paper proposes a method to determine a prior distribution when using microarray data. The second issue is the difficulty of deriving an analytic expression for the Bayesian MMSE estimator. This is done for discrete classification under a family of generalized beta prior distributions in Dalton and Dougherty (2011a) and for linear classifiers applied to Gaussian distributions under normal–inverse-Wishart prior distributions in Dalton and Dougherty (2011b). While we are not advocating the abandonment of analytic methods, it is practically useful to have software that can arrive at the Bayesian MMSE estimator via Monte Carlo methods. Currently, approximation is necessary when using a non-linear classifier, where a closed form solution for the model is not known. This article develops and provides publicly available software.

2 SYSTEMS AND METHODS

2.1 Modeling microarray data

We assume two classes and require the training sample to consist of normalized log ratios. Thus, use of normalization schemes such as total intensity normalization or the LOESS method, which are popular transformations before high-level analysis is applied, are required. Log-transformed gene expression values have nearly Gaussian class-conditional distributions (with unknown parameters) (Autio *et al.*, 2009; Hoyle *et al.*, 2002). To further validate a Gaussian modeling assumption, during feature selection we will permit only features that pass a Shapiro–Wilk Gaussianity test. Note that Bayesian error estimators designed under the Gaussian model are robust in the sense that performance is still good when the true distributions are Johnson distributions (Dalton and Dougherty, 2011b), which are a class of non-Gaussian distributions with four free parameters to control mean, variance, skewness and kurtosis.

Normal–inverse-Wishart priors compose a flexible class of distributions with many degrees of freedom to facilitate calibration of the priors to gene expression microarrays. Further, this family of priors possesses a fast closed-form solution when used with linear classification. In problems where the Gaussian model applies and one wishes to use a linear classifier, the benefit one might gain by having more control over the prior is not worth the much greater amount of time required to run an integral approximation code and the effort of designing a specialized model, especially for small samples where one cannot afford a very complex model anyway. Hence, we focus on calibrating normal–inverse-Wishart priors.

Assuming the parameters between classes are fairly independent, we have justified the assumptions posed by Dalton and Dougherty (2011a), the others being that the class-conditional distributions are relatively Gaussian and that normal–inverse-Wishart priors are adequate for representing prior knowledge. We are left to devise a method of generating priors for the mean and covariance of each class.

2.2 The Bayesian error estimator for Gaussian distributions with normal–inverse-Wishart priors

This section summarizes essentials from Dalton and Dougherty (2011b), using similar notation. Consider the class $y \in \{0,1\}$ in a binary classification problem. D multivariate Gaussian features are used for classification and we define the distribution parameters for these D features to be the mean and covariance, $\theta=\{\mu,\Sigma\}$. We assume Σ is invertible with probability 1, and for

invertible Σ our priors are of the form:

$$\pi(\theta)=\pi(\mu|\Sigma)\pi(\Sigma),$$

where

$$\pi(\mu|\Sigma)\sim\mathcal{N}\left(\mathbf{m},\Sigma/\nu\right),$$
$$\pi(\Sigma)\propto|\Sigma|^{-(\kappa+D+1)/2}\exp\left(-\frac{1}{2}\text{trace}\left(S\Sigma^{-1}\right)\right).$$

That is, the mean conditioned on the covariance is Gaussian with mean $\mathbf{m}$ and covariance Σ/ν, and the marginal distribution of the covariance is an inverse-Wishart distribution. The hyperparameters of $\pi(\theta)$ are a real number $\nu\geq 0$, a length D real vector $\mathbf{m}$, a real number κ and a non-negative definite $D\times D$ matrix S. For linear classification, we also restrict κ to be an integer to guarantee a closed form solution. The hyperparameters $\mathbf{m}$ and S can be viewed as targets for the mean and the shape of the covariance, respectively. The larger ν is the more localized the prior is about $\mathbf{m}$, and the larger κ is the less the shape of Σ is allowed to wiggle.

Given n_y observed sample points, we update the prior for class y to a posterior, π^*. This posterior has the same form as the prior, with updated hyperparameters given by

$$\kappa^*=\kappa+n_y,$$
$$S^*=(n_y-1)\widehat{\Sigma}+S+\frac{n_y\nu}{n_y+\nu}(\widehat{\mu}-\mathbf{m})(\widehat{\mu}-\mathbf{m})^T,$$
$$\nu^*=\nu+n_y,$$
$$\mathbf{m}^*=\frac{n_y\widehat{\mu}+\nu\mathbf{m}}{n_y+\nu},$$

where $\widehat{\mu}$ and $\widehat{\Sigma}$ are the sample mean and sample covariance of points from class y, respectively. To ensure a proper prior, we require $\kappa>D-1$, S positive definite, and $\nu>0$. These restrictions are not mandatory as long as the posterior is proper with $\kappa^*>D-1$, S^* positive definite and $\nu^*>0$.

Assuming the *a priori* probability of class 0 is uniform between 0 and 1 and assuming prior (and posterior) independence between this and the distribution parameters in each class, the Bayesian MMSE error estimator can be expressed as

$$\widehat{\varepsilon}=\frac{n_0+1}{n+2}\text{E}_{\pi^*}[\varepsilon_n^0]+\frac{n_1+1}{n+2}\text{E}_{\pi^*}[\varepsilon_n^1],$$

where $n=n_0+n_1$ is the total number of sample points. $\text{E}_{\pi^*}[\varepsilon_n^y]$ may be viewed as the posterior expectation of the error contributed by class y. With a fixed classifier and given θ, the true error, $\varepsilon_n^y(\theta)$, is deterministic and

$$\text{E}_{\pi^*}[\varepsilon_n^y]=\int_{\Theta_y}\varepsilon_n^y(\theta)\pi^*(\theta)d\theta, \tag{1}$$

where Θ_y is the parameter space of class y. For non-linear classifiers, this integral must be approximated with Monte Carlo methods.

For a linear classifier, i.e. a classifier of the form

$$\psi_n(\mathbf{x})=\begin{cases}0 \text{ if } g(\mathbf{x})\leq 0\\ 1 \text{ if } g(\mathbf{x})>0\end{cases},$$

where $g(\mathbf{x})=\mathbf{a}^T\mathbf{x}+b$ with some constant vector $\mathbf{a}$ and constant scalar b,

$$\text{E}_{\pi^*}[\varepsilon_n^y]=\frac{1}{2}\left(1+\text{sgn}(A)I\left(\frac{A^2}{A^2+\mathbf{a}^TS^*\mathbf{a}};\frac{1}{2},\frac{\kappa^*-D+1}{2}\right)\right), \tag{2}$$

where

$$A=(-1)^y g(\mathbf{m}^*)\sqrt{\frac{\nu^*}{\nu^*+1}},$$

and $I\left(x;\alpha,\beta\right)$ is the regularized incomplete beta function. For positive integer N, $I\left(x;\frac{1}{2},\frac{N}{2}\right)$ has a closed form solution, in particular, $I\left(1;\frac{1}{2},\frac{N}{2}\right)=1$ and for $0\leq x<1$,

$$I\left(x;\frac{1}{2},\frac{N}{2}\right)$$
$$=\begin{cases}(2/\pi)\sin^{-1}\left(\sqrt{x}\right) & \text{if } N=1\\ (2/\pi)\sin^{-1}\left(\sqrt{x}\right) & \\ \quad+\dfrac{2\sqrt{x}}{\pi}\displaystyle\sum_{k=1}^{(N-1)/2}\frac{(2k-2)!!}{(2k-1)!!}\left(1-x\right)^{k-\frac{1}{2}} & \text{if } N>1 \text{ is odd}\\ \sqrt{x}\displaystyle\sum_{k=0}^{(N-2)/2}\frac{(2k-1)!!}{(2k)!!}\left(1-x\right)^k & \text{if } N>1 \text{ is even,}\end{cases} \tag{3}$$

where !! is the double factorial.

2.3 Implementation of exact and approximate Bayesian error estimators

Assuming a Gaussian model with normal–inverse-Wishart priors for the Gaussian distribution parameters, with fixed hyperparameters for the priors of each class, we use the observed sample to update the hyperparameters of the posteriors. We also check that these posteriors are valid density functions, and if they are not, by default the code reports the error contributed by that class to be 0.5. Note that the Bayesian error estimator is most useful in a small sample setting, but the sample size must not be so small that the posterior is not a valid density function. This may happen, for instance, if we use a flat prior with $\kappa+D+2=0$ and the sample size for class y is $n_y\leq 2D+1$, so that $\kappa^*=\kappa+n_y\leq D-1$. In such cases, the Bayesian error estimator is meaningless because the available information is not sufficient for estimation, but generally there are also too few sample points for any error estimator to provide meaningful results.

Given valid normal–inverse-Wishart posteriors, the closed form Bayesian error estimator in Equation (2) for linear classification is easily evaluated. For arbitrary classifiers, we approximate the Bayesian error estimator in Equation (1) with a Monte Carlo approach. For each class, we generate a random mean and covariance pair according to the specified posterior normal–inverse-Wishart distribution. Several algorithms for generating normal–inverse-Wishart distributed multivariate sample points are available, see Johnson (1987). For each mean and covariance pair, the true error contributed by the class for the designed classifier is approximated by generating 10 000 sample points from the Gaussian distribution having the specified mean and covariance, and finding the error of these sample points on the classifier. The Bayesian error estimator is computed by averaging these true errors over 2500 random sets of mean and covariance pairs.

A toolbox of C code for Bayesian error estimation is publicly available. This includes the exact Bayesian error estimator for linear classifiers, the approximation code described above for arbitrary classifiers, a three-stage feature selection algorithm discussed in Section 2.4, as well as code implementing the method of generating priors described in Section 2.5. Simulations demonstrating the accuracy of this approximation with synthetic data and LDA classification are available in the Supplementary Material.

2.4 Feature selection

We use a three-stage feature selection method based on the t-test and a Gaussianity test to reduce the original feature set to D features. Since this article is not focused on optimizing a classification scheme, but rather on investigating the performance of error estimators, this feature selection scheme is intended to be a simple possible scheme to produce highly differentially expressed Gaussian features.

In the first stage, only highly differentially expressed features or features with a high likelihood of biological significance are selected. These may be selected by a t-test or based on biological knowledge. This stage reduces the number of features from tens of thousands to a few hundred. The second stage applies a Shapiro–Wilk hypothesis test (Shapiro and Wilk, 1965) on each feature of each class. Only features passing the Shapiro–Wilk test with

95% confidence in both classes are used, unless there are not enough features passing the test, in which case we select a fixed number of features with the highest sum of the Shapiro–Wilk test statistics in each class. In the final stage of feature selection, we reduce the feature set to D features. This is done either by applying a t-test if it has not already been applied in the first stage or by using the same t-test statistics from the first stage to pick the D most differentially expressed Gaussian features.

This implementation employs classifier-independent feature selection schemes, such as the t-test and Shapiro–Wilk test. However, even for classifier-dependent schemes, once the feature selection and classification schemes have been implemented, the Bayesian error estimator (BEE) may be calculated as a deterministic function of the fixed classifier. This is in contrast to cross-validation, which uses surrogate classifiers to estimate the error of the designed classifier.

2.5 Estimating prior hyperparameters

When calibrating priors for microarrays, what data should be used and how? With the explosion of microarray experimentation over the last decade, the genomics community has amassed an enormous database of gene expression data, and trends in the entire history of microarray experimentation could be used to find a prior, perhaps conditioned on a particular organism, tissue, gene and/or type of abnormality, depending on the nature of the experiment at hand. However, different microarray experiments are currently very difficult to compare, although there have been some recent efforts to normalize and integrate different datasets (Autio *et al.*, 2009).

The method employed here uses discarded gene expression data, consisting of a subset of the features from the microarray data that are not used for classification, to calibrate the priors of the Bayesian error estimator. Though these features are not used in the actual classifier, they may implicitly contain useful calibration information such as the varying concentrations of DNA material used in each microarray, background intensities and other characteristics of the digitized images of a microarray slide. And although calibration requires a large amount of data and in microarray gene expression analysis we typically expect a very small sample setting, the huge number of discarded features ensures that there is enough data for a successful calibration of the hyperparameters.

It is possible to define a prior on the entire feature set and to compute the Bayesian error estimator over the reduced feature set based on the marginal distribution of this prior on only the selected features. However, the following approach directly defines a prior on only the selected features.

We essentially use a method of moments approach to calibrate the hyperparameters; however, estimating a vector $\mathbf{m}$ and matrix S may be problematic for a small number of sample points, so to simplify the analysis we assume the following structure on these hyperparameters:

$$\mathbf{m} = m[1,1,\ldots,1]^T,$$

$$S = \sigma^2 \begin{bmatrix} 1 & \rho & \cdots & \rho \\ \rho & 1 & \cdots & \rho \\ \vdots & \vdots & \ddots & \vdots \\ \rho & \rho & \cdots & 1 \end{bmatrix},$$

where m is a real number, $\sigma^2 \geq 0$, and $-1 \leq \rho \leq 1$. This structure is justified because prior to observing the data, there is no reason to think that any feature, or pair of features, should have unique properties. With this simplification, our problem is now reduced to estimating five scalers for each class: ν, m, κ, σ^2 and ρ.

In the first stage of a method of moments approach, we find the theoretical first and second moments of the random variables μ and Σ (random because of the prior distribution applied to them) in terms of the hyperparameters we wish to estimate. Throughout the remainder of this section, a subscript i represents the i-th element of a vector, and a subscript jk represents the j-th row, k-th column element of a matrix.

First consider the parameter Σ, with a marginal prior having an inverse-Wishart distribution with hyperparameters κ and S. The mean of this distribution is well known (Rowe, 2003),

$$\mathrm{E}[\Sigma] = \frac{S}{\kappa - D - 1},$$

and given the previously defined structure on S, we obtain

$$\sigma^2 = (\kappa - D - 1)\mathrm{E}[\Sigma_{11}], \tag{4}$$

$$\rho = \frac{\mathrm{E}[\Sigma_{12}]}{\mathrm{E}[\Sigma_{11}]}. \tag{5}$$

Due to our imposed structure, only $\mathrm{E}[\Sigma_{11}]$ and $\mathrm{E}[\Sigma_{12}]$ are needed.

The variance of the j-th diagonal element in inverse-Wishart distributed Σ may be expressed as

$$\mathrm{Var}(\Sigma_{jj}) = \frac{2(S_{jj})^2}{(\kappa - D - 1)^2(\kappa - D - 3)} = \frac{2(\mathrm{E}[\Sigma_{11}])^2}{\kappa - D - 3},$$

where we have applied Equation (4) in the second equality. Solving for κ,

$$\kappa = \frac{2(\mathrm{E}[\Sigma_{11}])^2}{\mathrm{Var}(\Sigma_{11})} + D + 3. \tag{6}$$

We next consider the mean, μ, which is parameterized by the hyperparameters ν and m. The marginal distribution of the mean is a multivariate Student's t-distribution given by Rowe (2003):

$$\pi(\mu) = \frac{\Gamma\left(\frac{\kappa+1}{2}\right)}{\Gamma\left(\frac{\kappa-D+1}{2}\right)} \sqrt{\frac{\nu^D}{\pi^D} \frac{|S|^{-1}}{\left(1 + \nu(\mu - \mathbf{m})^T S^{-1}(\mu - \mathbf{m})\right)^{\kappa+1}}}.$$

The mean and covariance of this distribution are well known:

$$\mathrm{E}[\mu] = \mathbf{m},$$

$$\mathrm{Var}(\mu) = \frac{S}{(\kappa - D - 1)\nu} = \frac{\mathrm{E}[\Sigma]}{\nu}.$$

With the assumed structure on $\mathbf{m}$, we obtain

$$m = \mathrm{E}[\mu_1], \tag{7}$$

$$\nu = \frac{\mathrm{E}[\Sigma_{11}]}{\mathrm{Var}(\mu_1)}. \tag{8}$$

Finally, our objective is to approximate the expectations in Equations (4) through (8) using calibration features left out of the classification scheme. Suppose the calibration data for the current class consists of n sample points with $E \gg D$ features. Let $\widehat{\mu}^E$ be the sample mean and $\widehat{\Sigma}^E$ be the sample covariance matrix of the complete set of E features in the calibration data. From these we wish to find several sample moments of μ and Σ in our original D feature problem, that is, to find $\widehat{\mathrm{E}}[\mu_1]$, $\widehat{\mathrm{Var}}(\mu_1)$, $\widehat{\mathrm{E}}[\Sigma_{11}]$, $\widehat{\mathrm{E}}[\Sigma_{12}]$ and $\widehat{\mathrm{Var}}(\Sigma_{11})$, where the hats indicate the sample moment of the corresponding quantity. All these are scaler quantities.

To compress the set of E features in the calibration data to solve an estimation problem on just D features, and ultimately to find these scaler sample moments in a balanced way, we emulate the feature selection process by assuming that the selected features are drawn uniformly. Since any of the E features is equally likely to be selected as the i-th feature, the sample mean of the mean of the i-th feature, $\widehat{\mathrm{E}}[\mu_i]$, is computed as the average of the sample means of all E features in the calibration data. This result is the same for all i, and we use $\widehat{\mathrm{E}}[\mu_1]$ to represent all features. In particular,

$$\widehat{\mathrm{E}}[\mu_1] = \frac{1}{E}\sum_{i=1}^{E} \widehat{\mu}_i^E. \tag{9}$$

Thanks to uniform feature selection, all other moments are balanced over all features or any pair of distinct features. The remaining sample moments are obtained in a similar manner:

$$\widehat{\mathrm{Var}}(\mu_1) = \frac{1}{E-1}\sum_{i=1}^{E} \left(\widehat{\mu}_i^E - \widehat{\mathrm{E}}[\mu_1]\right)^2, \tag{10}$$

$$\widehat{\mathrm{E}}[\Sigma_{11}] = \frac{1}{E}\sum_{i=1}^{E} \widehat{\Sigma}_{ii}^E, \tag{11}$$

$$\widehat{\mathrm{E}}[\Sigma_{12}] = \frac{2}{E(E-1)} \sum_{i=2}^{E} \sum_{j=1}^{i-1} \widehat{\Sigma}_{ij}^{E}, \tag{12}$$

$$\widehat{\mathrm{Var}}(\Sigma_{11}) = \frac{1}{E-1} \sum_{i=1}^{E} \left(\widehat{\Sigma}_{ii}^{E} - \widehat{\mathrm{E}}[\Sigma_{11}] \right)^2. \tag{13}$$

Here, $\widehat{\mathrm{Var}}(\mu_1)$ represents the variance of each feature in the mean. We also have $\widehat{\mathrm{E}}[\Sigma_{11}]$ and $\widehat{\mathrm{E}}[\Sigma_{12}]$ representing the sample mean of diagonal elements and off-diagonal elements in Σ, respectively. Finally, $\widehat{\mathrm{Var}}(\Sigma_{11})$ is the sample variance of the diagonal elements in Σ.

Plugging our sample moments into Equations (4) through (8), we obtain

$$\sigma^2 = 2\widehat{\mathrm{E}}[\Sigma_{11}] \left(\frac{(\widehat{\mathrm{E}}[\Sigma_{11}])^2}{\widehat{\mathrm{Var}}(\Sigma_{11})} + 1 \right), \tag{14}$$

$$\rho = \frac{\widehat{\mathrm{E}}[\Sigma_{12}]}{\widehat{\mathrm{E}}[\Sigma_{11}]}, \tag{15}$$

$$\kappa = \frac{2(\widehat{\mathrm{E}}[\Sigma_{11}])^2}{\widehat{\mathrm{Var}}(\Sigma_{11})} + D + 3, \tag{16}$$

$$m = \widehat{\mathrm{E}}[\mu_1], \tag{17}$$

$$\nu = \frac{\widehat{\mathrm{E}}[\Sigma_{11}]}{\widehat{\mathrm{Var}}(\mu_1)}. \tag{18}$$

Note Equation (6) for κ was plugged into Equation (4) to obtain the final σ^2.

In sum, calibration for the prior hyperparameters is defined by Equations (14) through (18), the sample moments being given in Equations (9) through (13). The estimates of κ and ν can be unstable, since they rely on second moments, $\widehat{\mathrm{Var}}(\Sigma_{11})$ and $\widehat{\mathrm{Var}}(\mu_1)$, in a denominator. These parameters can be made more stable by discarding outliers when computing the sample moments. Herein, we discard the 10% of the $\widehat{\mu}_i^E$ with largest magnitude and the 10% of the $\widehat{\Sigma}_{ii}^E$ with largest value.

This method is one of many possible approaches; for simplicity and to avoid an over-defined system of equations, we do not incorporate the covariance between distinct features in μ [that is, $\mathrm{Cov}(\mu)_{12}$], the variance of off-diagonal elements in Σ [that is, $\mathrm{Var}(\Sigma_{12})$], or the correlation between distinct elements in Σ, though it may be possible to use these to improve the estimates of the hyperparameters. It may also be feasible to use other estimation methods, such as maximum likelihood. Furthermore, the method proposed here to calibrate the priors is a purely data-driven technique for easy and general application to microarray experiments. Ideally, the best way to calibrate priors would be to incorporate data and biological knowledge specific to the particular features selected for classification.

3 RESULTS AND DISCUSSION

We present two sets of results demonstrating good performance of Bayesian error estimators, one on synthetic high-dimensional data with three-stage feature selection and a second based on breast cancer data with two stages of feature selection.

3.1 High-dimensional synthetic data

In this section, we apply our Bayesian prior estimation method to synthetic high-dimensional microarray data. We use the same synthetic data model provided in Hua *et al.* (2009), which models many observations made in microarray expression-based studies, including blocked covariance matrices to model groups of interacting variables with negligible interactions between groups.

Our model emulates a full feature-label distribution with 20 000 total features. Features are categorized as either 'markers' or 'non-markers'. Markers represent features that have different class-conditional distributions in the two classes and are further divided into two subtypes: global markers and heterogeneous markers. Non-markers have the same distributions for both classes and thus have no discriminatory power, and are also divided into two subtypes: high-variance non-markers and low-variance non-markers. A summary of the feature types is shown in Figure 2.

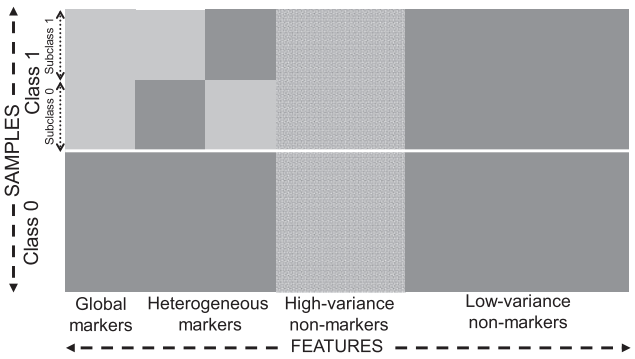


Fig. 2. Different feature types in constructing the high-dimensional synthetic data model (Hua *et al.*, 2009).

Twenty features are global markers, which are homogeneous in each class. In particular, the set of all global markers in class i has a Gaussian distribution with mean μ_i^{gm} and covariance matrix Σ_i^{gm}.

Within class 1, we assume each sample point belongs to one of two equally likely subclasses named 0 and 1, representing different stages or subtypes of cancer. Each subclass is associated with 50 heterogeneous markers, which are jointly Gaussian with mean μ_1^{hm} and covariance Σ_1^{hm}. Sample points associated with the other subclass have the same distribution as class 0, which is Gaussian with mean μ_0^{hm} and covariance Σ_0^{hm}. Each heterogeneous marker may only be associated with one subclass, thus there are 100 total heterogeneous markers in the model.

We simplify the model by assuming that μ_i^{gm} and μ_i^{hm} have the form $m_i \times (1, 1, \ldots, 1)$ for fixed scalers m_i. We assume Σ_i^{gm} and Σ_i^{hm} have the form $\sigma_i^2 \Sigma$, where σ_i^2 are constants and Σ, not to be confused with the definition in Section 2.2, has a block covariance structure, i.e.

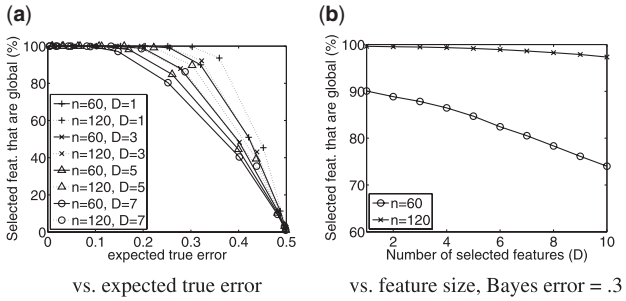
$$\Sigma = \begin{bmatrix} \Sigma_\rho & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \Sigma_\rho \end{bmatrix},$$

with Σ_ρ being a 5×5 matrix with 1 on the diagonal and $\rho = 0.8$ off the diagonal. That is, we group markers into blocks of five features, where the blocks are independent from each other, and the markers within each block are correlated with a relatively high correlation coefficient to emulate a pathway.

We generate 2000 high-variance non-marker features, which have independent mixed Gaussian distributions given by $pN(m_0, \sigma_0^2) + (1-p)N(m_1, \sigma_1^2)$, where m_i and σ_i^2 are the same scalers defined for markers. The random variable p is selected independently for each feature with a uniform distribution over $[0, 1]$ and is applied to all sample points of both classes. These features can be viewed as genes regulated by mechanisms unrelated to those that regulate the class 0 and class 1 phenotypes. The remaining features are low-variance non-marker features, each having independent univariate Gaussian distributions with mean m_0 and variance σ_0^2.

Table 1. Synthetic high-dimensional data model parameters

Parameters	Values/description
Total features	20 000
Global markers	20
Subclasses in class 1	2
Heterogeneous markers	50 per subclass (100 total)
High-variance features	2000
Low-variance features	17 880
Mean	$m_0 = 0$, $m_1 = 1$
Variances	$\sigma^2 = \sigma_0^2 = \sigma_1^2$ (controls Bayes error)
Block size	5
Block correlation	0.8
a priori probability of class 0	0.5



vs. expected true error vs. feature size, Bayes error = .3

Fig. 3. Percentage of three-stage selected features that are global features in the synthetic high-dimensional data model. (**a**) versus expected true error; (**b**) versus feature size, Bayes error = 0.3.

In this model, heterogeneous markers are Gaussian within each subclass, but the class-conditional distribution for class 1 is a mixed Gaussian distribution (mixing the distributions of the subclasses), and is thus not Gaussian. Further, the high-variance features are also mixed Gaussian distributions, so this model incorporates both Gaussian and non-Gaussian features to challenge the Shapiro–Wilk Gaussianity test in the feature selection scheme.

To simplify our simulations, we set the *a priori* probability of both classes to 0.5 and fix the parameters $m_0 = 0$ and $m_1 = 1$. We also define a single parameter $\sigma^2 = \sigma_0^2 = \sigma_1^2$, which specifies the difficulty of the classification problem. A summary of our synthetic high-dimensional data model parameters is given in Table 1. In all simulations, the values for σ^2 are chosen so that a single global feature (note that all global features are identical) has a specific Bayes error. We call this the 'Bayes error' in the remainder of this section, and it is given by $\varepsilon^* = \Phi\left(-1/(2\sigma)\right)$, where Φ is the unit normal Gaussian cumulative distribution function, so for instance, we use $\sigma = 0.9537$ for a Bayes error of 0.3.

Under this high-dimensional model, we run several Monte Carlo simulations. In each experiment, we fix the training sample size, n, the number of selected features, D, and the difficulty of the classification problem via σ. The synthetically generated samples are non-stratified, meaning that in each iteration the sample size of each class is not fixed but determined by a binomial $(0.5, n)$ experiment, and the corresponding sample points are randomly generated according to the distributions defined for each class.

Once the sample has been generated, we apply the three-stage feature selection scheme outlined in Section 2.4. In the first stage, we apply a t-test to obtain 1000 highly differentially expressed features by removing most non-informative features. In the second stage, we apply a Shapiro–Wilk Gaussianity test and eliminate features that do not pass the test with 95% confidence. The number of features output in this stage is variable. If there are not at least 30 features that pass the test, then we return the 30 features with the highest sum of the Shapiro–Wilk test statistics for both classes. In the final stage, we use the same t-test values computed before to obtain the final set of D highly differentially expressed Gaussian features, which will be used to design our classifier. The $1000 - D$ features that pass the first stage of feature selection but are not used for classification are saved as calibration data.

The feature selected training data are then used to train an LDA classifier. With the classifier fixed, 5000 testing points are drawn from exactly the same distribution as the training data and used expressly to approximate the true error. Subsequently, several training data error estimators are computed, including leave-one-out (loo), 5-fold cross-validation (cv), 0.632 bootstrap (boot) and bolstered resubstitution (bol) (see the Supplementary Material for details). Two Bayesian error estimators are also applied, one with flat non-informative priors defined by $\pi(\theta) = 1$ (flat BEE), and the other with priors calibrated as described in Section 2.5 (calibrated BEE). Since the classifier is linear, these BEEs are computed exactly. This entire process is repeated 120 000 times to approximate the RMS deviations from the true error for each error estimator.

We first analyze the quality of features selected by the three-stage feature selection algorithm. Figure 3a shows the percentage of selected features that are global features with respect to the expected true error of the designed classifier. We would like to graph performance with respect to Bayes error, which is a more pure measure of the difficulty of a classification problem, but evaluating Bayes error on our high-dimensional model is difficult and it may not be close to the true error of the designed classifier. Hence, in our graphs we focus on performance with respect to expected true error. Similarly, Figure 3b graphs against feature size with a fixed Bayes error of 0.3. Recall that this model uses 20 000 features, of which only 20 are global features that most effectively discriminate the classes. As long as the feature size is reasonable given the difficulty of the problem (expected true error and sample size), this percentage is quite large. However, in Figure 3b for sample size 60 we see that a feature size larger than 7 will result in <80% of the selected features being global features. This illustrates the necessity of restricting feature size in a small sample setting, and is consistent with earlier studies showing the difficulty of finding good feature sets when the number of features is large and the sample is small (Dougherty *et al.*, 2009; Sima and Dougherty, 2006).

The graphs in Figure 4 show the percentage of selected feature sets that are not rejected by a multivariate Shapiro–Wilk test on either class at a 95% significance level. There are several multivariate Gaussianity tests based on the Shapiro–Wilk statistic. We used Villasenor Alvaa and Estradaa (2009), which generalizes the classical univariate Shapiro–Wilk test to the multivariate case by transforming the data into a set of approximately independent standard normal random variables, and essentially summing up the standard Shapiro–Wilk statistic on each dimension. The results show that even though the three-stage feature selection algorithm only uses a univariate Gaussianity test, and univariate normality does not

imply multivariate normality, the resulting feature set still tends to have a high probability of passing the multivariate Gaussianity test.

We next turn our attention to the RMS performance of error estimators under our synthetic high-dimensional model, where a summary of all simulation settings are available in Table 2. Our first battery of simulations in Figure 5 shows RMS deviation from true error for all error estimators with respect to expected true error for LDA classification with 1, 3, 5 or 7 selected features and either 60 or 120 sample points. Given the sample sizes, it is prudent to keep the number of selected features small to have satisfactory feature selection (Sima and Dougherty, 2006) and to avoid the peaking phenomena (Hua *et al.*, 2005, 2009). Lines marked with 'o' represent the Bayesian error estimator with flat priors, and lines marked with 'x' represent the Bayesian error estimator with the calibrated priors. The key point in these graphs is that the calibrated BEE has best performance in the mid and high range. For an expected true error of about 0.25 and $n=60$, the RMS for the calibrated BEE outperforms 5-fold cross-validation for $D=1,3,5$ and 7 by 0.0507, 0.0300, 0.0335 and 0.0379, respectively, representing 64, 32, 30 and 29% decrease in RMS, respectively. For $n=120$, the decrease in RMS for $D=1,3,5$ and 7 is 0.0366, 0.0175, 0.0192 and 0.0198, respectively, for 67, 34, 35, and 33 percent decrease in RMS, respectively. All other error estimators typically have best performance for low expected true errors, with the flat BEE having even better performance than the classical error estimation schemes. Indeed, all graphs except Figure 5g demonstrate that either the flat or calibrated Bayesian error estimator is the best scheme over the whole range of expected true error.

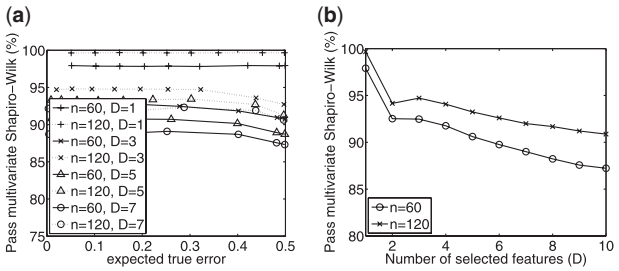


Fig. 4. Percentage of three-stage selected features that are not rejected by a multivariate Shapiro–Wilk test on either class at a 95% significance level with the synthetic high-dimensional data model. (**a**) versus expected true error; (**b**) versus feature size, Bayes error = 0.3.

Our next set of graphs in Figure 6 show simulation results with respect to feature size. For reference, graphs of the expected true error for these simulations are shown in Figure 7. Calibrated priors provide the best performance, except when combining large feature and small sample sizes, in which case a flat prior performs best. In fact, performance of the calibrated BEE in Figure 6 tends to be best precisely in the rage of feature sizes with the highest percentage of global features and the lowest true errors. For example, the calibrated BEE in Figure 6a for sample of size 60 has the best performance up to 7 features, where in Figure 3b the percentage of selected features being global is greater than about 80% and in Figure 7 the true error has started to level off. Note, also, the consistently superior performance of the calibrated BEE over the non-Bayesian estimators for $n=60$; indeed, throughout the range of feature sizes, the calibrated BEE has an RMS of at least 0.0263 smaller than the best performing non-Bayesian error estimator, which represents an improvement of at least 14%.

Note the upward RMS trend in Figure 6a and the downward trend in Figure 6b for the non-Bayesian error estimators. Although it can be dangerous to generalize about the behavior of error estimators, let us at least conjecture. We see in Figure 7 that the true error is large for $n=60$, with little improvement as we increase the number of features and, in fact, increasing true error as the number of features passes 7, which is a clear sign of the peaking phenomenon. Thus, for $n=60$, adding features creates a more difficult estimation problem that is not offset by easing error estimation on account of small true errors. On the other hand, in Figure 7 we see a fast reduction of true error for $n=120$ as more features are added, thereby greatly easing the error estimation problem and resulting in the declining RMS trend in Figure 6b. While these comments apply directly to the non-Bayesian error estimators, they apply to the Bayesian estimators relative to their change of slope. The flat BEE is relatively constant in Figure 6a but falls along with the non-Bayesian error estimators in Figure 6b, whereas the calibrated BEE consistently rises in Figure 6a but remains relatively flat in Figure 6b.

3.2 Empirical breast cancer data

We applied the Bayesian error estimator to normalized gene expression measurements from a breast cancer study

Table 2. Data model and classification settings for simulation with synthetic high-dimensional data

Data model	Classifier	Sample size		Feature selection				BEE calibration	Iteration
Bayes error		Training	Test	Original	1st t-test	Shapiro–Wilk test	2nd t-test		
0.05–0.45	LDA	$n=60$	5000	20 000	1000	95% confidence	$D=1$	$1000-D$	120 000
0.05–0.45	LDA	$n=60$	5000	20 000	1000	95% confidence	$D=3$	$1000-D$	120 000
0.05–0.45	LDA	$n=60$	5000	20 000	1000	95% confidence	$D=5$	$1000-D$	120 000
0.05–0.45	LDA	$n=60$	5000	20 000	1000	95% confidence	$D=7$	$1000-D$	120 000
0.05–0.45	LDA	$n=120$	5000	20 000	1000	95% confidence	$D=1$	$1000-D$	120 000
0.05–0.45	LDA	$n=120$	5000	20 000	1000	95% confidence	$D=3$	$1000-D$	120 000
0.05–0.45	LDA	$n=120$	5000	20 000	1000	95% confidence	$D=5$	$1000-D$	120 000
0.05–0.45	LDA	$n=120$	5000	20 000	1000	95% confidence	$D=7$	$1000-D$	120 000
0.3	LDA	$n=60$	5000	20 000	1000	95% confidence	1 to 10	$1000-D$	120 000
0.3	LDA	$n=120$	5000	20 000	1000	95% confidence	1 to 10	$1000-D$	120 000

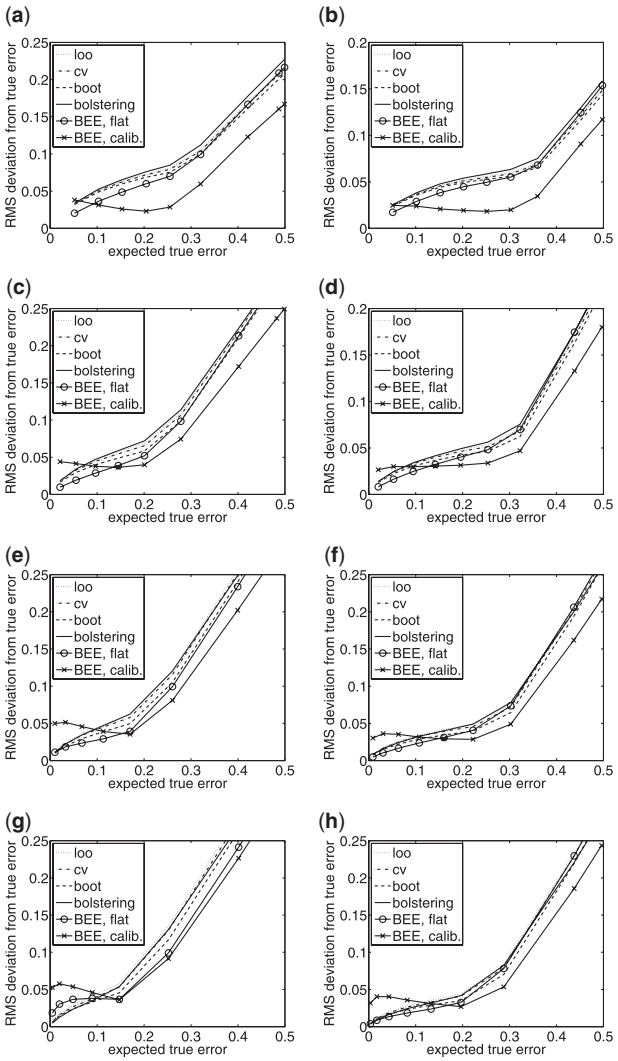


Fig. 5. RMS deviation from true error for the synthetic high-dimensional data model with LDA classification versus expected true error. (**a**) $n=60$, $D=1$; (**b**) $n=120$, $D=1$; (**c**) $n=60$, $D=3$; (**d**) $n=120$, $D=1$; (**e**) $n=60$, $D=5$; (**f**) $n=120$, $D=5$; (**g**) $n=60$, $D=7$; (**h**) $n=120$, $D=7$.

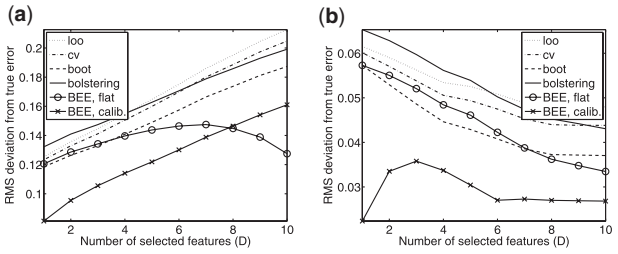


Fig. 6. RMS deviation from true error for the synthetic high-dimensional data model with LDA classification versus feature size. See Figure 7 for the expected true errors in these graphs. (**a**) $n=60$, Bayes error = 0.3; (**b**) $n=120$, Bayes error = 0.3.

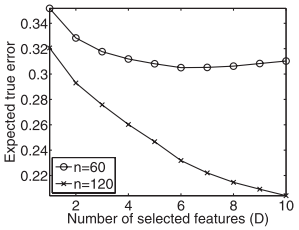


Fig. 7. Expected true error for the synthetic high-dimensional data model with LDA classification versus feature size, Bayes error = 0.3.

(van de Vijver *et al.*, 2002). This study used 295 sample points, with 180 assigned to class 0 (good prognosis) and 115 in class 1 (bad prognosis), and provides a 70 feature prognosis profile. From the original 295 points, we randomly draw a non-stratified training sample of size n. Since the number of features in the dataset is relatively small, we apply only the last two stages of our feature selection scheme in Section 2.4. The first stage selects features passing a Shapiro–Wilk Gaussianity test with 95% confidence and must report at least D features, while the second stage selects D features with the highest t-test statistic. The $70-D$ features not used for classification are retained as calibration data for Bayesian error estimation. After feature selection, we train an LDA, QDA or 3NN classifier.

The remaining sample points are used as holdout data to approximate the true error of the designed classifier. The previously considered error estimators are also evaluated from the training samples [except in the case of 3NN where semibolstering is used instead of bolstering owing to its superior performance for 3NN (Braga-Neto and Dougherty, 2004a)], along with exact Bayesian error estimators (for LDA) or approximate Bayesian error estimators (for QDA and 3NN). Both flat and calibrated priors are applied. This process is repeated either 100 000 times (for LDA) or 10 000 times (for QDA and 3NN) to estimate the average RMS deviation of each error estimator from the true error.

The Bayesian error estimator priors are calibrated as discussed in Section 2.5. A typical prior with 2 features and 40 sample points is $\nu=16.80$, $m=-0.004$, $\kappa=12$, $\sigma^2/(\kappa-D-1)=0.042$ and $\rho=0.020$ for class 0, and $\nu=2.78$, $m=-0.068$, $\kappa=10$, $\sigma^2/(\kappa-D-1)=0.024$ and $\rho=0.073$ for class 1. These indicate that the good prognosis class (0) has a distribution with a more concentrated mean (since ν is much larger) and the mean is close to 0, which is expected since the data have been normalized. On the other hand, κ is fairly large for both classes, suggesting that the variance of each feature in either class is probably close to the prior expected variance, $\sigma^2/(\kappa-D-1)$. Interestingly, the variance is a bit larger for class 0 and ρ is usually small but positive.

Figures 8, 9 and 10 provide simulation results for LDA, QDA and 3NN, respectively. Each figure contains subplots representing fixed feature sizes between one and five, and one figure showing the expected true error for all simulations with the corresponding classifier. A summary of the simulation settings is shown in Table 3. The uniform prior performs well over a wide range of sample and feature sizes, and generally shows significant improvement over the classical error estimators. Prior calibration can have even more pronounced improvement, especially for small feature sets. Also, although the uniform prior often performs better than the calibrated

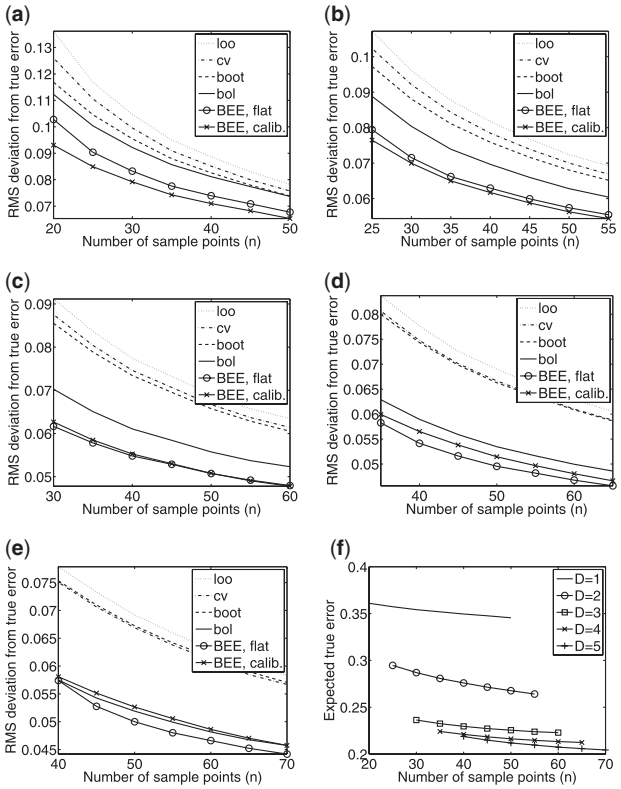


Fig. 8. RMS deviation from true error and expected true error with LDA classification of empirical measurements from a breast cancer study. (**a**) 1 feature; (**b**) 2 features; (**c**) 3 features; (**d**) 4 features; (**e**) 5 features; (**f**) expected true error.

prior for high feature sizes, see for example Figure 8e for 5 features, we observe in Figure 8f that true error does not improve much, and may actually get worse, for as little as 5 features. This may indicate that when there is not enough calibration data for good prior design, there is also probably insufficient data for good classifier design.

3.3 Concluding remarks

Our synthetic data simulations demonstrate the power of prior knowledge in two ways: we may assume a low Bayes error by using a flat prior and outperform the classical error estimators where they perform best, or we may calibrate a prior, even using purely data-driven methods, and obtain superior performance in the midrange of Bayes errors. Also note that for moderately difficult classification problems which are typical in a small sample biological setting, the midrange is precisely where training data error estimation is needed. One might argue that there is a risk with postulating a low-Bayes-error prior since, although it will show excellent performance if the Bayes error is truly low, it will suffer for large Bayes errors. In Figure 5, not only does performance deteriorate with increasing Bayes error for the Bayesian MMSE estimator, but also the performance of cross-validation. This should not be surprising because the use of cross-validation presupposes that the Bayes error is small because its performance seriously degrades for increasing Bayes error. This behavior, noted more than 30 years ago in a simple 1D Gaussian model (Glick, 1978), has been

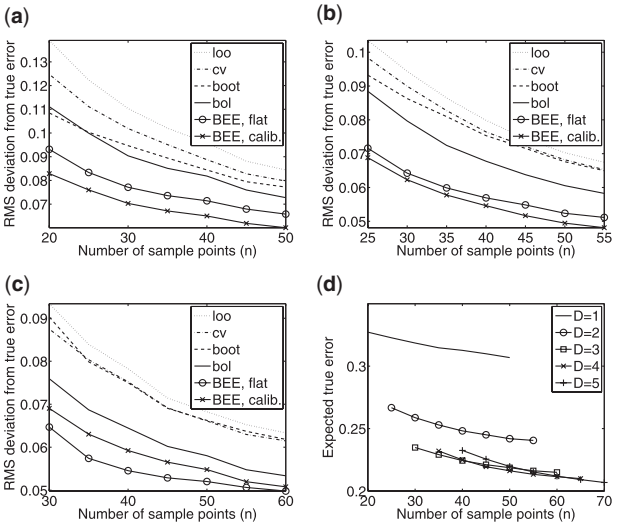


Fig. 9. RMS deviation from true error and expected true error with QDA classification of empirical measurements from a breast cancer study.(**a**) 1 feature; (**b**) 2 features; (**c**) 3 features; (**d**) expected true error.

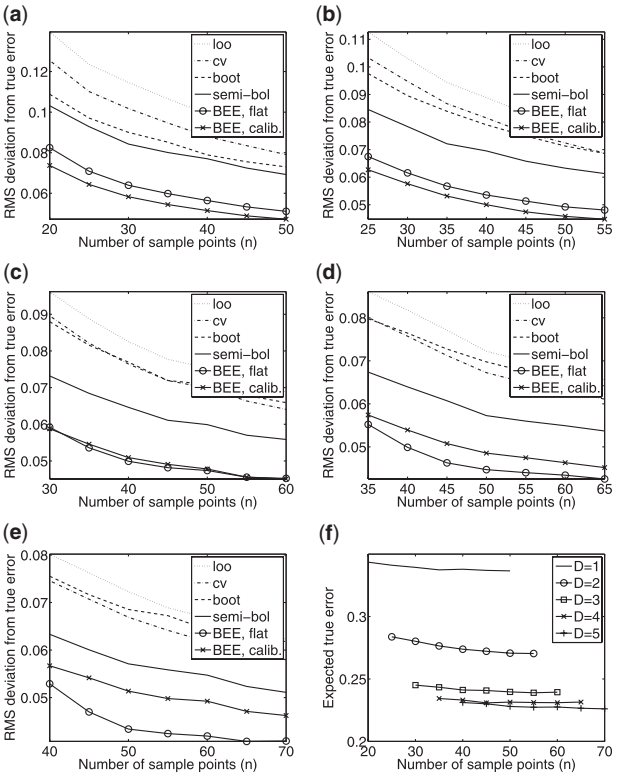


Fig. 10. RMS deviation from true error and expected true error with 3NN classification of empirical measurements from a breast cancer study.(**a**) 1 feature; (**b**) 2 features; (**c**) 3 features; (**d**) 4 features; (**e**) 5 features; (**f**) expected true error.

Table 3. Classification schemes and settings for simulation with real breast cancer data

Classifier	Sample size		Feature selection			BEE calibration	Iteration
	Training	Test	Original	Shapiro–Wilk test	2nd t-test		
LDA	20–50	$295-n$	70	95% confidence	$D=1$	$70-D$	100 000
LDA	25–55	$295-n$	70	95% confidence	$D=2$	$70-D$	100 000
LDA	30–60	$295-n$	70	95% confidence	$D=3$	$70-D$	100 000
LDA	35–65	$295-n$	70	95% confidence	$D=4$	$70-D$	100 000
LDA	40–70	$295-n$	70	95% confidence	$D=5$	$70-D$	100 000
QDA	20–50	$295-n$	70	95% confidence	$D=1$	$70-D$	10 000
QDA	25–55	$295-n$	70	95% confidence	$D=2$	$70-D$	10 000
QDA	30–60	$295-n$	70	95% confidence	$D=3$	$70-D$	10 000
3NN	20–50	$295-n$	70	95% confidence	$D=1$	$70-D$	10 000
3NN	25–55	$295-n$	70	95% confidence	$D=2$	$70-D$	10 000
3NN	30–60	$295-n$	70	95% confidence	$D=3$	$70-D$	10 000
3NN	35–65	$295-n$	70	95% confidence	$D=4$	$70-D$	10 000
3NN	40–70	$295-n$	70	95% confidence	$D=5$	$70-D$	10 000

demonstrated via large simulations for both discrete and Gaussian models (Dalton and Dougherty, 2011a, b), and has been analytically proven in the Gaussian model (Zollanvari *et al.*, 2010). In other words, unless one is not interested in error estimator performance, use of cross-validation carries with it implicitly assumed prior knowledge. If one knows that the Bayes error is low, then why not define a prior model based on this assumption to design a Bayesian error estimator with even better performance?

ACKNOWLEDGMENTS

The authors would like to thank Yidong Chen for his helpful discussions on modeling microarray data.

Funding: Philanthropic Educational Organization (P.E.O.); National Science Foundation (CCF-0634794) in part.

Conflict of Interest: none declared.

REFERENCES

Autio,R. *et al.* (2009) Comparison of Affymetrix data normalization methods using 6,926 experiments across five array generations. *BMC Bioinformatics*, **10**(**Suppl. 1**), Article S24.

Braga-Neto,U. and Dougherty,E.R. (2004a) Bolstered error estimation. *Pattern Recogn.*, **37**, 1267–1281.

Braga-Neto,U.M. and Dougherty,E.R. (2004b) Is cross-validation valid for small-sample microarray classification? *Bioinformatics*, **20**, 374–380.

Braga-Neto,U. and Dougherty,E.R. (2010) Exact correlation between actual and estimated errors in discrete classification. *Pattern Recogn. Lett.*, **31**, 407–412.

Dalton,L.A. and Dougherty,E.R. (2011a) Bayesian minimum mean-square error estimation for classification error–part I: definition and the Bayesian MMSE error estimator for discrete classification. *IEEE Trans. Signal Process.*, **59**, 115–129.

Dalton,L.A. and Dougherty,E.R. (2011b) Bayesian minimum mean-square error estimation for classification error–part II: the Bayesian MMSE error estimator for linear classification of Gaussian distributions. *IEEE Trans. Signal Process.*, **59**, 130–144.

Devroye,L. and Wagner,T. (1979) Distribution-free inequalities for the deleted and hold-out error estimates. *IEEE Trans. Inf. Theory*, **25**, 202–207.

Devroye,L. *et al.* (1996) *A Probabilistic Theory of Pattern Recognition*. Springer, New York.

Dougherty,E.R. and Braga-Neto,U. (2006) Epistemology of computational biology: mathematical models and experimental prediction as the basis of their validity. *J. Biol. Syst.*, **14**, 65–90.

Dougherty,E.R. *et al.* (2009) Performance of feature selection methods. *Curr. Genomics*, **10**, 365–374.

Dougherty,E.R. *et al.* (2010) Performance of error estimators for classification. *Curr. Bioinformatics*, **5**, 53–67.

Glick,N. (1978) Additive estimators for probabilites of correct classification. *Pattern Recogn.*, **10**, 211–222.

Hanczar,B. *et al.* (2007) Decorrelation of the true and estimated classifier errors in high-dimensional settings. *EURASIP J. Bioinform. Syst. Biol.*, 38473.

Hoyle,D.C. *et al.* (2002) Making sense of microarray data distributions. *Bioinformatics*, **18**, 576–584.

Hua,J. *et al.* (2005) Optimal number of features as a function of sample size for various classification rules. *Bioinformatics*, **21**, 1509–1515.

Hua,J. *et al.* (2009) Performance of feature-selection methods in the classification of high-dimension data. *Pattern Recogn.*, **42**, 409–424.

Johnson,M.E. (1987) *Multivariate Statistical Simulation*. John Wiley and Sons, New York.

Rowe,D.B. (2003) *Multivariate Bayesian Statistics: Models for Source Separation and Signal Unmixing*. Chapman & Hall/CRC, Boca Raton, FL.

Shapiro,S.S. and Wilk,M.B. (1965) An analysis of variance test for normality (complete samples). *Biometrika*, **3**, 591–611.

Sima,C. and Dougherty,E.R. (2006) What should be expected from feature selection in small-sample settings. *Bioinformatics*, **22**, 2430–2436.

van de Vijver,M.J. *et al.* (2002) A gene-expression signature as a predictor of survival in breast cancer. *N. Engl. J. of Med.*, **347**, 1999–2009.

Villasenor Alvaa,J.A. and Estradaa,E.G. (2009) A generalization of Shapiro-Wilk's test for multivariate normality. *Comm. Statist. Theory Methods*, **38**, 1870–1883.

Zollanvari,A. *et al.* (2010) On the joint sampling distribution between the actual classification error and the resubstitution and leave-one-out error estimators for linear classifiers. *IEEE Trans. Inf. Theory*, **56**, 784–804.