BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 9 2005, pages 1979–1986
doi:10.1093/bioinformatics/bti294

Gene expression

Estimating misclassification error with small samples via bootstrap cross-validation

Wenjiang J. Fu*, Raymond J. Carroll and Suojin Wang

Department of Statistics, Texas A & M University, 447 Blocker Building, 3143 TAMU, College Station, TX 77843, USA

Received on October 20, 2004; revised on January 20, 2005; accepted on January 21, 2005
Advance Access publication February 2, 2005

ABSTRACT

Motivation: Estimation of misclassification error has received increasing attention in clinical diagnosis and bioinformatics studies, especially in small sample studies with microarray data. Current error estimation methods are not satisfactory because they either have large variability (such as leave-one-out cross-validation) or large bias (such as resubstitution and leave-one-out bootstrap). While small sample size remains one of the key features of costly clinical investigations or of microarray studies that have limited resources in funding, time and tissue materials, accurate and easy-to-implement error estimation methods for small samples are desirable and will be beneficial.

Results: A bootstrap cross-validation method is studied. It achieves accurate error estimation through a simple procedure with bootstrap resampling and only costs computer CPU time. Simulation studies and applications to microarray data demonstrate that it performs consistently better than its competitors. This method possesses several attractive properties: (1) it is implemented through a simple procedure; (2) it performs well for small samples with sample size, as small as 16; (3) it is not restricted to any particular classification rules and thus applies to many parametric or non-parametric methods.

Contact: wfu@stat.tamu.edu

1 INTRODUCTION

Cross-validation (CV) has been widely used in estimating prediction errors in many statistical models such as regressions and classifications. It is well-known that CV provides unbiased estimation and is easy to implement. However, recent discussions about the role of CV in estimating misclassification error in microarray data analysis raised concerns over its performance since CV presents large variability with small samples. Braga-Neto and Dougherty (2004) studied cases where classifiers were trained based on a small number of genes that investigators may be interested in. Ambroise and McLachlan (2002) studied cases where classifiers were trained based on a large number of genes available in microarray studies. While small sample size remains one of the key features of microarray studies, it is of great interest to develop methodologies that potentially provide more accurate estimation with small samples (Dougherty, 2001; Brun *et al*., 2003; Kim *et al*., 2002). Efron (1983) and Efron and Tibshirani (1997) studied the leave-one-out bootstrap (LOOBT), 0.632 bootstrap (BT632) and 0.632+ bootstrap (BT632+) methods for small sample classification. Their methods improved error estimation by bootstrap resampling with training set separated from test set and yielded slightly biased estimation. Ambroise and McLachlan (2002) applied 10-fold CV and BT632+ methods to highly fit microarray data, where the number of genes is huge, in the order of thousands, while the sample size is relatively small, in the order of tens or hundreds. Their findings with small samples are encouraging. We are thus motivated to investigate whether bootstrap resampling improves CV with small samples and study the bootstrap cross-validation (BCV) method, a simple and straightforward method of estimating misclassification error. We conclude that in many cases BCV performs better than LOOBT, BT632 and BT632+ with small samples in terms of mean relative squared error.

The paper is organized as follows. In Section 2 we introduce the BCV method and show why BCV works. Section 3 compares BCV with its competitors in estimating misclassification errors through simulation studies. In Section 4 we apply BCV and other methods to a microarray study on breast cancer patient prognosis and demonstrate that BCV performs better than its competitors. Section 5 provides some concluding remarks.

2 BOOTSTRAP CROSS-VALIDATION METHOD

2.1 The BCV procedure

We propose BCV by combining the bootstrap (Efron and Tibshirani, 1993) with CV (Geisser, 1975; Stone, 1974). Assume that we have a sample $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ of size n, where x_i represents the feature the observation (x_i, y_i) possesses and y_i is the class label of the observation. We draw bootstrap samples with replacement, $S_b^* = \{(x_1^*, y_1^*), \ldots, (x_n^*, y_n^*)\}$, $b = 1, \ldots, B$ for some large B between 50 and 200. We require that each bootstrap sample have at least three distinct observations in every class. For each sample S_b^*, we carry out CV with a predetermined classification rule, such as the linear discriminant analysis (LDA) and obtain an error estimation r_b. We then repeat this procedure B times and calculate the averaged error estimation $r_{\mathrm{BCV}} = B^{-1} \sum_{b=1}^{\mathrm{B}} r_b$ over all B bootstrap samples. We call r_{BCV} the BCV error. Note that <3 distinct observations in one class can be problematic in training LDA classifier with cross-validated bootstrap samples.

By definition, BCV is in the framework of Breiman's bagging predictors (Breiman, 1996). Breiman studied a number of classification rules and showed that bagging through bootstrap-aggregating improves the performance of unstable estimators, such as the classification and regression trees (CART), but does not yield much improvement for stable estimators, such as k-nearest neighbors

*To whom correspondence should be addressed.

(KNN). The bagging predictor is defined as follows: for a given sample S and a predetermined classification rule $\mathcal{C}$, such as CART or KNN, draw B bootstrap samples $S_1^*, \ldots, S_B^*$ from S with B between 50 and 200. Train classifier $\mathcal{C}$ on each bootstrap sample S_b^*, $b = 1, \ldots, B$, and estimate its misclassification error $r_{\mathcal{C}}(S_b^*)$ on S_b^* with a predetermined method, such as CV. The bagging predictor error is defined as the averaged misclassification error over B bootstrap samples, $r = B^{-1} \sum_{b=1}^{B} r_{\mathcal{C}}(S_b^*)$. By definition BCV is a bagging predictor, but LOOBT is not. Neither is BT632 nor BT632+ as a weighted average of LOOBT and resubstitution. Breiman's conclusion was supported by extensive simulation results with sample size $\geq$200. Further properties of bagging estimators in terms of bias, variance, mean squared errors (MSE) and asymptotics have been studied by Buja and Stuetzle (2000a,b) (available at http://www-stat.wharton.upenn.edu/~buja/), Buhlmann and Yu (2002), Chen and Hall (2003) and Friedman and Hall (2000). Although bagging predictors, such as random forest via bagged trees, were discussed in Dudoit *et al*. (2002) for small sample studies in microarray data analysis, BCV has not been studied specifically for samples as small as 20.

2.2 Why BCV works

Since overlap of training and test data sets for classification may induce bias and may lead to underestimation of misclassification error, the above BCV procedure no doubt raises concerns with duplicate observations in bootstrap samples. We now explain why BCV works for small samples.

Consider the CV procedure for a given sample $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ of size n as aforementioned. Assume the underlying distribution of the feature x is F in a feature-class label pair (x, y). At the i-th step, $i = 1, \ldots, n$, the observation (x_i, y_i) is left out. A classification model G is trained with the remaining data $S^{(-i)}$ and is denoted by $G^{(-i)}$. If the sample size n is large enough, the remaining sample $S^{(-i)}$ still represents the empirical underlying distribution $\widehat{F}$ well. The left-out observation x_i has a range of distances to the sample points in $S^{(-i)}$, from small to large. Thus the predicted class label $\widehat{y}_i = G^{(-i)}(x_i)$ has a large probability to be equal to y_i and the prediction performs well. This leads to a good performance of CV. However, if the sample size n is small, the data in S are sparse. Assume there are no duplicate observations. It implies that the sample $S^{(-i)}$ tends to have a large distance from the left-out observation (x_i, y_i). Thus prediction $G^{(-i)}(x_i)$ does not perform well and has a large variance. This is why CV performs poorly with small samples.

It is known that bootstrap samples S^* have duplicates. Here we use the superscript * to indicate bootstrap samples or observations in bootstrap samples. The duplicate copies of (x_i^*, y_i^*) in the remaining sample $S^{*(-i)}$, when the observation (x_i^*, y_i^*) is left out, may be regarded as copies of observations (x, y_i^*) with jittered feature x_i^* that are close to x but $x_i^* \neq x$. Thus the observation (x_i^*, y_i^*) is close enough to the remaining sample $S^{*(-i)}$ to improve the performance of prediction $G^{*(-i)}(x_i^*)$. Therefore, prediction by CV performs better in bootstrap samples than in the original sample when sample size is small. Heuristically for the same reason, BCV performs better than LOOBT with small samples since LOOBT is computed in such a way that it only counts errors from test data sets that have no overlap with training data sets. Since BT632 and BT632+ have a large weight on LOOBT, they do not perform as well as BCV in error estimation with small samples.

In the next section, we demonstrate through simulations, that BCV performs better than its competitors in estimating misclassification error with small samples. We will focus on numerical comparison of the performance of BCV with other error estimation methods, but will not study the properties of BCV, such as how bias and standard error of the estimate of misclassification error depend on sample size and choice of classification rule, in this paper.

3 SIMULATION STUDIES

We compare BCV with its competitors in estimating misclassification error based on random samples generated from two populations featuring in either one-dimensional or multi-dimensional space. Here, we only consider cases where observations belong to definitive categories with no exception. For complicated cases with observations difficult to assign a category to, we refer readers to the work on outlier detection by Wang *et al*. (1997). For one-dimensional feature space, we consider both symmetric and asymmetric distributions. We generate small, moderate and large random samples of size n with equal or unequal number of observations from known populations. Different values of n are used in separate simulations with $n = 16$, 20, 30, 50 and 100. Although microarray data typically have thousands of genes available, classifiers are usually trained based on a small number of genes or features that are of interest to investigators, where features can be extracted through dimension reduction methods, such as principal component analysis (singular value decomposition) or partial least squares (Nguyen and Rocke, 2002, 2004). We chose to have a low-dimensional feature space in simulations and a small number of genes in a case study in the next section for the purpose of accurate computation of true conditional error because high-dimensional data suffer from the curse of dimensionality (Hastie and Tibshirani, 1990), where data are further apart in high dimensional space and result in inaccurate error estimation. For cases where only a few genes are of interest and are involved in training classifiers, our BCV method applies directly and yields accurate error estimation. For cases where a large number of genes are involved in training classifiers, our BCV method has not been assessed and may not perform as well if it is directly applied to a large number of genes. However, dimension reduction methods can be applied to extract a few important features; and our BCV method applies to the small number of important features to yield accurate error estimation. Although our simulation studies only consider two class discriminate analysis, the results can be applied to multiple classes because a multiple class problem of m classes ($m > 2$) can be converted, for the purpose of error estimation but not classification, into m separate problems of two classes, class i and non-class i ($1 \leq i \leq m$), with unequal number of observations. The above error estimation methods and our simulation results are readily applicable to multiple class discriminate analysis.

To compare BCV with its competitors, we estimate the conditional error based on one given sample $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ randomly generated. The conditional error estimation is of interest in many studies since true distributions are unknown in practice and all inferences are made based on given samples.

3.1 Simulation study 1: two classes of equal number of samples with one-dimensional normal feature

We generated data from one-dimensional normal distributions Normal $(0, 1)$ and Normal $(\Delta, 1)$ with $\Delta > 0$. We demonstrate that BCV performed consistently better in terms of mean squared relative

Table 1. Mean relative deviation, square-root of MSRE and their standard errors of QDA misclassification errors by BCV, CV, LOOBT and BT632 in 1000 runs

Δ	n	BCV	CV	LOOBT	BT632
		Relative deviation $\overline{R}$ (SE)			
1	20	0.0055 (0.0089)*	0.0810 (0.0129)	0.1917 (0.0108)	0.0851 (0.0101)
	30	0.0284 (0.0077)	0.0451 (0.0091)	0.1553 (0.0091)	0.0784 (0.0085)
	50	0.0248 (0.0066)	0.0252 (0.0072)	0.0985 (0.0075)	0.0505 (0.0070)
	100	0.0049 (0.0047)*	0.0053 (0.0049)*	0.0345 (0.0049)	0.0158 (0.0049)
		$\sqrt{\text{MSRE}}$ (SE)			
	20	0.2828 (0.0055)	0.4171 (0.0149)	0.3926 (0.0084)	0.3314 (0.0069)
	30	0.2452 (0.0053)	0.2917 (0.0074)	0.3265 (0.0070)	0.2805 (0.0062)
	50	0.2098 (0.0038)	0.2293 (0.0055)	0.2557 (0.0057)	0.2283 (0.0050)
	100	0.1480 (0.0030)	0.1559 (0.0032)	0.1612 (0.0034)	0.1543 (0.0032)
		Relative deviation $\overline{R}$ (SE)			
3	20	0.0469 (0.0230)	0.1277 (0.0262)	0.4393 (0.0269)	0.2085 (0.0252)
	30	0.0339 (0.0200)*	0.1104 (0.0228)	0.2704 (0.0221)	0.1396 (0.0213)
	50	0.0057 (0.0158)*	0.0614 (0.0174)	0.1322 (0.0165)	0.0618 (0.0163)
	100	0.0191 (0.0117)*	0.0442 (0.0128)	0.0769 (0.0120)	0.0436 (0.0121)
		$\sqrt{\text{MSRE}}$ (SE)			
	20	0.7290 (0.0195)	0.8387 (0.0237)	0.9557 (0.0269)	0.8233 (0.0233)
	30	0.6319 (0.0135)	0.7281 (0.0166)	0.7495 (0.0175)	0.6867 (0.0155)
	50	0.4979 (0.0113)	0.5552 (0.0130)	0.5372 (0.0134)	0.5177 (0.0122)
	100	0.3719 (0.0086)	0.4061 (0.0094)	0.3878 (0.0093)	0.3847 (0.0090)

*Statistically nonsignificant deviation since the 95% confidence interval (Relative deviation $\pm$ 1.96 SE) contains 0.
Data generated from Normal $(0, 1)$ and Normal $(\Delta, 1)$. Two hundred bootstrap samples were generated for BCV, LOOBT and BT632.

errors (MSRE) than its competitors including CV, LOOBT and BT632.

For comparison purposes, we standardized the conditional errors and calculated their relative deviation from the true conditional misclassification error as follows. First, we generated a random sample $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ from the aforesaid two populations Normal $(0, 1)$ and Normal $(\Delta, 1)$ with $n/2$ observations from each. In the sample S, y_i is the class label of observation (x_i, y_i), 1 for Normal $(0, 1)$ and 2 for Normal $(\Delta, 1)$ and x_i is the feature of the observation with $x_i \sim$ Normal $(0, 1)$ if $y_i = 1$ or $x_i \sim$ Normal $(\Delta, 1)$ if $y_i = 2$. A quadratic discriminate analysis (QDA) classifier $\mathcal{C}(\mathcal{S})$ was then trained based on the sample S, and its true conditional error $r(S) = \text{P}_{\mathcal{C}(S)}\{(x, y)$ misclassified $|S\}$ was computed through 10 000 newly generated random observations (x, y) from Normal $(0, 1)$ and Normal $(\Delta, 1)$, 5000 from each. A QDA allows unequal variances of the training data in two different classes. We then computed the estimated conditional misclassification errors r for the given sample S by BCV, CV, LOOBT and BT632. A total of 200 bootstrap samples were generated in the computation of BCV, LOOBT and BT632. We then computed the relative deviation $R(S) = \{r - r(S)\}/r(S)$ for each estimation method. Thus a negative deviation indicates underestimation of true error while a positive deviation indicates overestimation of true error.

Second, we repeated the above procedure 1000 times with different randomly generated samples S_i, averaged the relative deviations with $\overline{R} = 1000^{-1} \sum_{i=1}^{1000} R(S_i)$, and calculated the MSRE with MSRE $= 1000^{-1} \sum_{i=1}^{1000} \{R(S_i)\}^2$ to compare different error estimation methods. We also compared these methods with absolute scale of the deviation and MSE, and observed results similar to the ones with relative scale. We present the results in relative scale only and omit the ones in absolute scale.

Table 1 displays the mean relative deviation $\overline{R}$, square-root of MSRE and their standard errors (SE) over 1000 simulation runs by BCV, CV, LOOBT and BT632 for sample size $n = 20$, 30, 50 and 100 and $\Delta = 1$ and 3. The square-root of MSRE allows direct comparison with the mean deviation $\overline{R}$ to identify the dominating term in $\sqrt{\text{MSRE}}$, either bias or variance. BCV had the smallest absolute deviation and the smallest $\sqrt{\text{MSRE}}$ consistently in all cases. In several cases, BCV had nonsignificant deviation since the 95% confidence interval (relative deviation $\pm$ 1.96 SE) contains 0, while CV, LOOBT and BT632 had significant positive deviations. It is demonstrated that in general BCV tends to yield more accurate estimation than its competitors with small, moderate and large samples.

Figure 1 shows error densities in the relative scale by BCV, CV, LOOBT and BT632 for $\Delta = 1$. These densities were estimated with density estimation function 'density' in the statistical analysis package R with Gaussian window type (Silverman, 1986). CV was positively skewed with a long tail for small sample size 20 and was shifted away from 0 for moderate sample sizes 30 and 50. LOOBT and BT632 were slightly positively shifted for small and moderate sample sizes 20, 30 and 50. BCV seemed to have a symmetric distribution about 0. Large sample size 100 made all densities symmetric about 0.

3.2 Simulation study 2: two classes of unequal number of samples with one-dimensional normal feature

In this simulation study, we assess the performance of BCV, CV, LOOBT and BT632 with unequal number of observations from two populations Normal $(0, 1)$ and Normal $(\Delta, 1)$ in one-dimensional feature space similar to study 1. For sample size $n = 20$, 30 and 50, we generated a random sample of 7, 10 and 20 observations from distribution Normal $(0, 1)$, respectively, and the remaining from

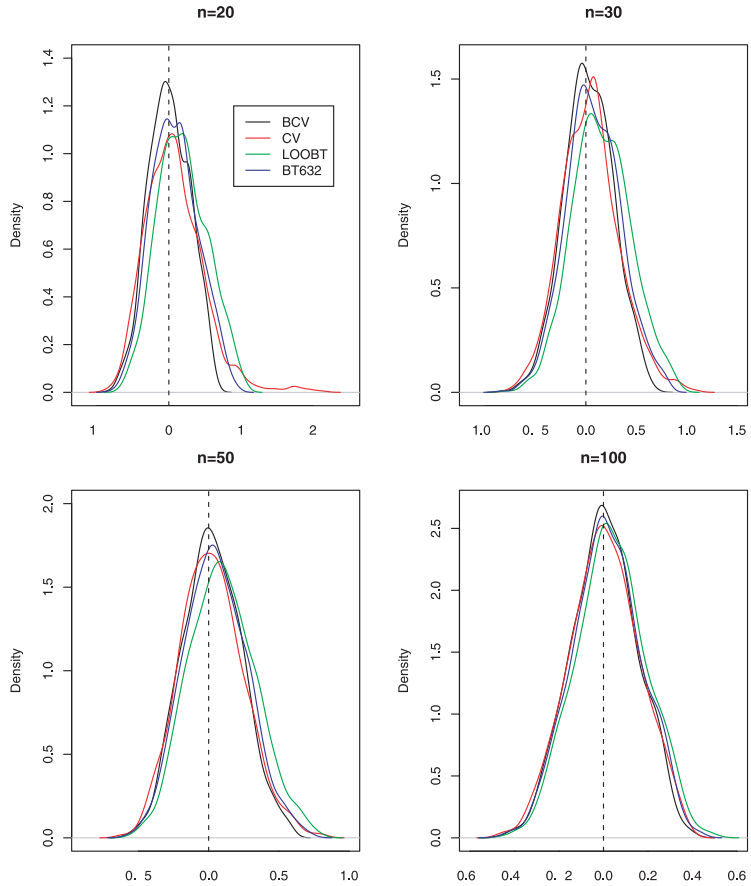

Fig. 1. Densities of conditional misclassification errors in relative scale estimated by different methods from 1000 simulation runs with data generated from distributions Normal $(0, 1)$ and Normal $(1, 1)$.

Normal $(\Delta, 1)$ for $\Delta = 1$ and 3 in two separate studies. We also assessed the 5-fold CV (5FCV) method for error estimation. Table 2 displays the mean deviation and square-root of MSRE and their SE. Similar to study 1, BCV achieved the smallest $\sqrt{\text{MSRE}}$ except for one case with $\Delta = 3$ and $n = 30$, in which LOOBT had a slightly smaller $\sqrt{\text{MSRE}}$ (0.6682) than BCV (0.6761) but the difference was not significant relative to the SEs. Also shown in this study is that 5FCV had the largest $\sqrt{\text{MSRE}}$ compared with BCV, CV, LOOBT and BT632, which indicates that 5FCV does not perform as well as other error estimation methods. This is consistent with the discussion about 10-fold CV in Ambroise and McLachlan (2002). We thus chose not to include 5FCV in subsequent simulation studies.

3.3 Simulation study 3: two classes of equal number of observations with one-dimensional asymmetric feature

We generated random samples with asymmetric Chi-square distributions χ_1^2 and χ_{df}^2 with the degrees of freedom $df = 3$ and 5 in two separate simulations. We trained a five nearest neighbor (5NN) classifier and computed the relative deviation and $\sqrt{\text{MSRE}}$ of misclassification errors by BCV, CV, LOOBT and BT632 with 1000 simulation runs. Table 3 displays the mean deviation and $\sqrt{\text{MSRE}}$ for sample size $n = 16$, 20, 30 and 50. BCV achieved the smallest $\sqrt{\text{MSRE}}$ in all cases except for one with $df = 3$ and $n = 50$, where BT632 achieved a slightly smaller $\sqrt{\text{MSRE}}$ than BCV with $\sqrt{\text{MSRE}} = 0.1921$ and 0.2093, respectively.

3.4 Simulation study 4: two classes of equal number of observations with multi-dimensional normal feature

We generated a random sample S of size n from two population distributions in q-dimensional space with $n/2$ observations from each distribution and $q = 5$ and 10 in two separate simulations. The first distribution is a q-dimensional normal distribution $[\text{Normal}\ (0, 1)]^q$, i.e. each coordinate follows a standard normal distribution and the second distribution is $[\text{Normal}\ (\Delta, 1)]^q$ with $\Delta = 1$ and 0.8 for $q = 5$ and 10, respectively. We chose a smaller Δ value for $q = 10$ because the curse of dimensionality makes data points further apart in higher-dimensional space. We trained a three nearest neighbor (3NN) classifier on a given random sample and computed the true conditional misclassification error by testing the 3NN classifier on 100 000 and 1 000 000 random points generated from the population distributions for $q = 5$ and 10. Again, we tested on a larger set of sample points for 10-dimensional feature due to the curse of dimensionality. We then computed the relative deviation and $\sqrt{\text{MSRE}}$ for BCV, CV, LOOBT and BT632 with 500 simulation runs, where one new random sample S was generated for each simulation run and 50 bootstrap samples were generated to compute BCV, LOOBT and BT632.

Table 2. Mean relative deviation, square-root of MSRE and their standard errors of QDA misclassification errors by BCV, CV, LOOBT, BT632 and 5FCV in 1000 runs

Δ	n	BCV	CV	LOOBT	BT632	5FCV
		Relative deviation $\overline{R}$ (SE)				
1	20	−0.0574 (0.0085)	−0.0543 (0.0097)	0.0306 (0.0095)	−0.0308 (0.0092)	−0.0290 (0.0101)
	30	−0.0969 (0.0071)	−0.0735 (0.0082)	−0.0318 (0.0077)	−0.0648 (0.0076)	−0.0645 (0.0083)
	50	−0.0306 (0.0061)	−0.0229 (0.0066)	0.0108 (0.0066)	−0.0121 (0.0064)	−0.0149 (0.0067)
		$\sqrt{\text{MSRE}}$ (SE)				
	20	0.2758 (0.0056)	0.3125 (0.0065)	0.3015 (0.0060)	0.2919 (0.0058)	0.3198 (0.0071)
	30	0.2431 (0.0055)	0.2698 (0.0059)	0.2465 (0.0050)	0.2500 (0.0053)	0.2702 (0.0062)
	50	0.1965 (0.0044)	0.2096 (0.0048)	0.2084 (0.0045)	0.2042 (0.0045)	0.2127 (0.0049)
		Relative deviation $\overline{R}$ (SE)				
3	20	−0.0893 (0.0218)	−0.0184 (0.0250)	0.1261 (0.0235)	−0.0208 (0.0228)	0.0513 (0.0269)
	30	−0.1580 (0.0168)	−0.0774 (0.0199)	0.0171 (0.0178)	−0.0757 (0.0178)	−0.0595 (0.0202)
	50	−0.0686 (0.0149)	−0.0184 (0.0165)	0.0382 (0.0155)	−0.0183 (0.0155)	−0.0052 (0.0164)
		$\sqrt{\text{MSRE}}$ (SE)				
	20	0.7496 (0.0093)	0.7945 (0.0095)	0.7569 (0.0102)	0.7528 (0.0097)	0.8172 (0.0098)
	30	0.6761 (0.0089)	0.7159 (0.0093)	0.6682 (0.0096)	0.6770 (0.0093)	0.7167 (0.0094)
	50	0.6234 (0.0091)	0.6511 (0.0093)	0.6265 (0.0095)	0.6302 (0.0093)	0.6501 (0.0092)

Data generated from Normal $(0, 1)$ and Normal $(\Delta, 1)$ and 7, 10 and 20 observations generated from Normal $(0, 1)$ for sample size $n = 20$, 30 and 50, respectively. Two hundred bootstrap samples were generated for BCV, LOOBT and BT632.

Table 3. Mean relative deviation, square-root of MSRE and their standard errors of 5NN misclassification errors by BCV, CV, LOOBT and BT632 in 1000 runs

df	n	BCV	CV	LOOBT	BT632
		Relative deviation $\overline{R}$ (SE)			
3	16	0.0437 (0.0098)	0.1255 (0.0161)	0.2575 (0.0123)	0.0751 (0.0108)
	20	−0.0380 (0.0085)	0.0673 (0.0137)	0.2225 (0.0111)	0.0478 (0.0098)
	30	−0.1060 (0.0067)	0.0237 (0.0110)	0.1492 (0.0088)	−0.0107 (0.0078)
	50	−0.1297 (0.0052)	−0.0224 (0.0086)	0.1150 (0.0067)	−0.0318 (0.0060)
		$\sqrt{\text{MSRE}}$ (SE)			
	16	0.3126 (0.0062)	0.5235 (0.0138)	0.4672 (0.0088)	0.3510 (0.0071)
	20	0.2718 (0.0063)	0.4390 (0.0103)	0.4153 (0.0081)	0.3124 (0.0066)
	30	0.2375 (0.0053)	0.3497 (0.0083)	0.3165 (0.0066)	0.2480 (0.0056)
	50	0.2093 (0.0041)	0.2739 (0.0064)	0.2415 (0.0052)	0.1921 (0.0042)
		Relative deviation $\overline{R}$ (SE)			
5	16	0.1222 (0.0149)	0.1228 (0.0202)	0.3234 (0.0184)	0.1278 (0.0162)
	20	0.0120 (0.0128)	0.0298 (0.0169)	0.2662 (0.0163)	0.0804 (0.0142)
	30	−0.0370 (0.0105)	0.0540 (0.0151)	0.2230 (0.0135)	0.0543 (0.0118)
	50	−0.0734 (0.0086)	0.0184 (0.0119)	0.1697 (0.0111)	0.0169 (0.0097)
		$\sqrt{\text{MSRE}}$ (SE)			
	16	0.4869 (0.0115)	0.6515 (0.0181)	0.6642 (0.0161)	0.5265 (0.0130)
	20	0.4032 (0.0086)	0.5360 (0.0136)	0.5794 (0.0134)	0.4559 (0.0099)
	30	0.3335 (0.0075)	0.4815 (0.0127)	0.4809 (0.0115)	0.3782 (0.0090)
	50	0.2828 (0.0058)	0.3778 (0.0090)	0.3888 (0.0090)	0.3082 (0.0067)

Data generated from χ^2_1 and χ^2_{df}. Two hundred bootstrap samples were generated for BCV, LOOBT and BT632.

Table 4 displays the mean deviation and square-root of MSRE. BCV achieved the smallest $\sqrt{\text{MSRE}}$. Figure 2 shows the densities of the errors estimated via different methods for sample size $n = 16$ and 30, $q = 5$ in the upper panels and $q = 10$ in the lower panels. CV had a long tail to the right for small sample size $n = 16$ and $q = 5$. LOOBT and BT632 had a tail longer than that of BCV. BCV achieved the highest peak and seemed to be symmetric about 0.

4 APPLICATIONS TO MICROARRAY DATA

We assess the performance of BCV and its competitors with a microarray study of breast cancer patients' prognosis, in which a gene expression profiling method was proposed in predicting the prognosis for a patient with breast cancer based on 70 genes (van't Veer *et al*., 2002; van de Vijver *et al*., 2002). We chose this study

Table 4. Mean relative deviation, square-root of MSRE and their standard errors of 3NN misclassification errors by BCV, CV, LOOBT and BT632 in 500 runs

(q, Δ)	n	BCV	CV	LOOBT	BT632
		Relative deviation $\overline{R}$ (SE)			
(5, 1)	16	0.0441 (0.0153)	0.1324 (0.0255)	0.3910 (0.0197)	0.0714 (0.0171)
	20	−0.0281 (0.0131)	0.0737 (0.0213)	0.3707 (0.0177)	0.0470 (0.0150)
	30	−0.1038 (0.0100)	0.0073 (0.0170)	0.2913 (0.1402)	−0.0029 (0.0118)
		$\sqrt{\text{MSRE}}$ (SE)			
	16	0.3481 (0.0111)	0.5918 (0.0202)	0.5926 (0.0167)	0.3932 (0.0121)
	20	0.2934 (0.0095)	0.5198 (0.0184)	0.5424 (0.0157)	0.3373 (0.0111)
	30	0.2472 (0.0073)	0.3805 (0.0125)	0.4277 (0.0120)	0.2640 (0.0083)
		Relative deviation $\overline{R}$ (SE)			
(10, 0.8)	16	0.0684 (0.0144)	0.1096 (0.0259)	0.4293 (0.0191)	0.0872 (0.0167)
	20	0.0316 (0.0125)	0.1263 (0.0212)	0.4664 (0.0162)	0.1160 (0.0141)
	30	−0.0373(0.0112)	0.0645 (0.0189)	0.3924 (0.0155)	0.0597 (0.0133)
		$\sqrt{\text{MSRE}}$ (SE)			
	16	0.3291 (0.0105)	0.5892 (0.0185)	0.6051 (0.0164)	0.3835 (0.0126)
	20	0.2857 (0.0095)	0.5003 (0.0171)	0.5948 (0.0150)	0.3410 (0.0108)
	30	0.2540 (0.0081)	0.4291 (0.0146)	0.5243 (0.0141)	0.3041 (0.0104)

Data generated from [Normal $(0, 1)]^q$ and [Normal $(\Delta, 1)]^q$. Fifty bootstrap samples were generated for BCV, LOOBT and BT632 methods.

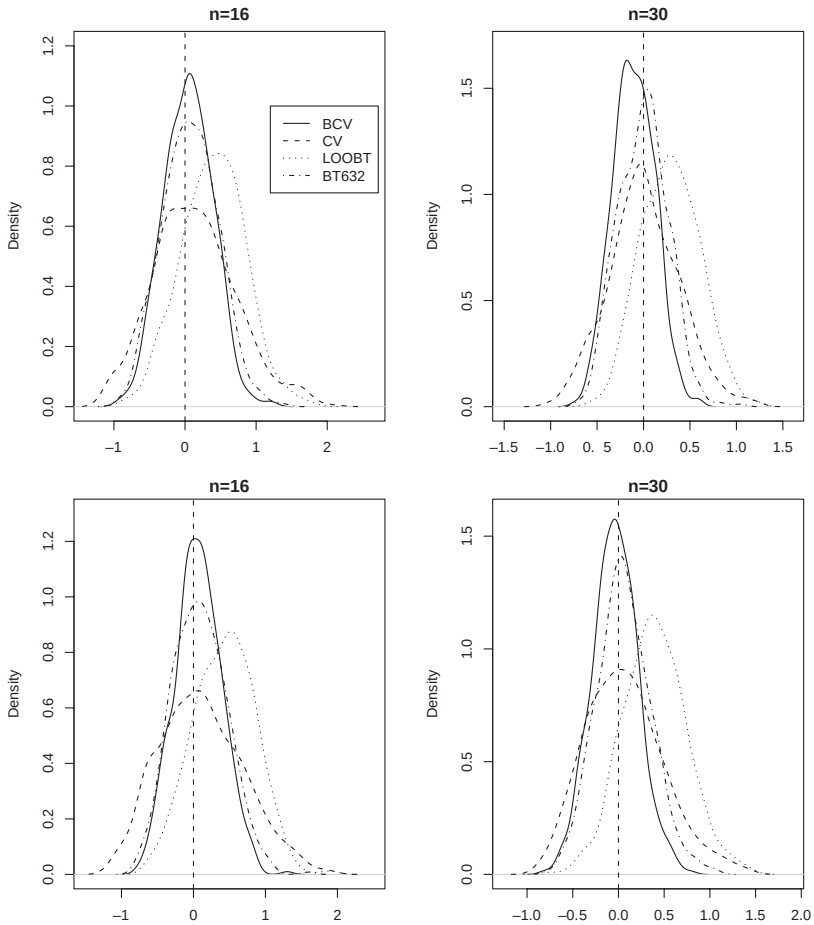

Fig. 2. Densities of conditional misclassification errors in relative scale estimated by different methods from 500 simulation runs with data generated from distributions [Normal $(0, 1)]^q$ and [Normal $(\Delta, 1)]^q$. Upper panels: $\Delta = 1$ and $q = 5$; Lower panels: $\Delta = 0.8$ and $q = 10$.

Table 5. Mean relative deviation, square-root of MSRE and their standard errors of KNN misclassification errors by BCV, CV, LOOBT and BT632 in 1000 runs

k	n	BCV	CV	LOOBT	BT632
		Relative deviation $\overline{R}$ (SE)			
3	20	−0.1409 (0.0090)	0.0233 (0.0150)	0.2186 (0.0125)	−0.0320 (0.0105)
	30	−0.1864 (0.0074)	−0.0132 (0.0121)	0.1504 (0.0102)	−0.0787 (0.0087)
		$\sqrt{\text{MSRE}}$ (SE)			
	20	0.3183 (0.0064)	0.4734 (0.0110)	0.4528 (0.0100)	0.3324 (0.0072)
	30	0.3003 (0.0058)	0.3835 (0.0087)	0.3568 (0.0084)	0.2872 (0.0063)
		Relative deviation $\overline{R}$ (SE)			
5	20	−0.0620 (0.0100)	0.0347 (0.0148)	0.2357 (0.0131)	0.0299 (0.0112)
	30	−0.0894 (0.0087)	0.0257 (0.0123)	0.1914 (0.0114)	0.0116 (0.0100)
		$\sqrt{\text{MSRE}}$ (SE)			
	20	0.3208 (0.0065)	0.4702 (0.0124)	0.4761 (0.0108)	0.3564 (0.0080)
	30	0.2879 (0.0059)	0.3894 (0.0092)	0.4071 (0.0091)	0.3156 (0.0071)

Data generated from the breast cancer prognosis study. Five genes were used in training the KNN classifiers and 200 bootstrap samples were generated for BCV, LOOBT and BT632.

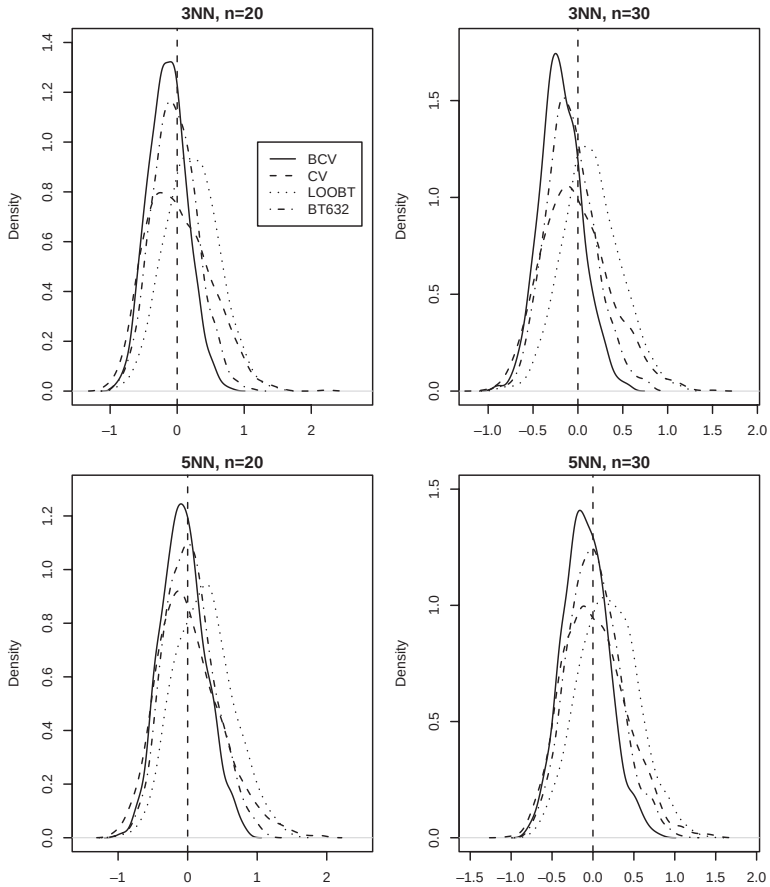

Fig. 3. Densities of conditional misclassification errors in relative scale estimated by different methods from 1000 runs of generating random sample S of size n from the breast cancer prognosis data and testing on the remaining data. Upper panels: 3NN classifier; Lower panels: 5NN classifier.

because the large sample size of 295 patients makes it possible to accurately estimate the conditional misclassification error for a given small random sample. Although the gene expression profiling was established with 70 genes, we chose 5 genes that are most highly correlated with the patient's prognosis and trained a KNN classifier based on the random sample. We were aware of the bias in such a gene selection procedure (Ambroise and McLachlan, 2002) and chose to do so because gene selection is not the focus of this study.

In biological experiments, investigators are often interested in certain genes based on their knowledge and gene selection is thus subject to selection bias as well. Therefore, we used these five genes to serve the purpose of comparing different error estimation methods.

For the comparison, we carried out the following steps. (1) Take a random sample S of size $n = 20$ or 30 with half the number having good prognosis and the other half having poor prognosis. (2) Train a KNN classifier based on the random sample S and compute its empirical 'true' misclassification error by comparing the predicted class of the remaining samples based on their gene expression levels with the true clinical prognosis class. (3) Compute the conditional error for the given random sample S by BCV, CV, LOOBT and BT632. (4) Calculate the relative deviation of the conditional errors from the 'true' error. (5) Repeat 1000 times the above steps (1)–(4) and calculate the mean relative deviation $\overline{R}$ and $\sqrt{\text{MSRE}}$.

Table 5 shows the mean deviation and $\sqrt{\text{MSRE}}$. BCV achieved the smallest $\sqrt{\text{MSRE}}$ in all cases except one, where a 3NN classifier was trained with a sample size $n = 30$ and BT632 performed slightly better ($\sqrt{\text{MSRE}} = 0.2872$) than BCV ($\sqrt{\text{MSRE}} = 0.3003$). Although CV achieved small relative deviation, its large variance and large MSE made it less competitive. LOOBT had large relative deviation and large MSRE. In general, BCV performed better than its competitors in terms of MSRE with small samples. Figure 3 shows densities of the errors estimated with different methods. BCV had a short tail and a higher peak $\sim$0 while others had longer tails and lower peaks. BCV had a density $\sim$0 while LOOBT had a shift towards overestimation except for the 3NN classifier with $n = 30$, where BCV, CV and BT632 had a shift toward underestimation. In summary, BCV performed better than its competitors in error estimation with small samples in this breast cancer patient prognosis study.

5 DISCUSSION

Estimating misclassification error with small samples is a key issue in statistics and bioinformatics, especially in microarray studies, where sample sizes are usually small. Although CV provides unbiased estimation in general, it presents large variability with small samples and is thus not satisfactory. Other methods, such as LOOBT, BT632 and BT632+, perform better than CV, but still yield biased estimation.

In this paper, we proposed BCV, a simple procedure through bootstrap resampling, applying CV on each bootstrap sample and averaging the errors across all bootstrap samples. Although BCV is in the framework of bagging predictors, it has not been studied particularly due to the concerns on overlapped training and test data in cross-validated bootstrap samples. We found that such overlapping of training and test data in the cross-validated bootstrap samples may facilitate accurate error estimation as an advantage rather than a disadvantage for small samples. Simulation studies demonstrated that BCV performed consistently better than its competitors, including the LOOBT and BT632. This result also implies that BCV performs better than BT632+ in error estimation because BT632+ takes a larger weight on LOOBT than BT632 (Ambroise and McLachlan, 2002; Efron and Tibshirani, 1997) and would, if included in our simulation studies, make BT632+ further biased toward overestimation of errors. Our application to a microarray data set also confirmed that BCV provided accurate error estimation.

BCV is a simple statistical procedure, and performs well with samples of sizes as small as 16. It is not restricted to any specific classification rules and thus applies to many parametric or non-parametric classification methods. While BCV has advantages compared to its competitors for small sample error estimation, its performance with large samples is not critical since many simple and computationally less expensive methods, such as CV, perform well and serve the needs of error estimation with large samples. Consequently, methods based on bootstrap resampling, such as BCV, LOOBT, BT632 and BT632+ are computationally expensive and hence are not recommended for large sample studies.

ACKNOWLEDGEMENTS

W.J.F. was supported by a grant from the National Cancer Institute (R25-CA90301). R.J.C. and S.W. were supported by a grant from the National Cancer Institute (5R01-CA57030) and by the Texas A&M Center for Environmental and Rural Health via a grant from the National Institute of Environmental Health Sciences (P30-ES09106).

REFERENCES

Ambroise,C. and McLachlan,G.J. (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. *Proc. Natl Acad. Sci. USA*, **99**, 6562–6566.

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

Breiman,L. (1996) Bagging predictors. *Mach. Learn.*, **24**, 123–140.

Brun,M. *et al*. (2003) Corrected small-sample estimation of the Bayes error. *Bioinformatics*, **19**, 944–951.

Buhlmann,P. and Yu,B. (2002) Analyzing bagging. *Ann. Statist.*, **30**, 927–961.

Buja,A. and Stuetzle,W. (2000a) Smoothing effects of Bagging. *Technical report, AT& T Labs. Research*, Florham Park, NJ, pp. 1–10.

Buja,A. and Stuetzle,W. (2000b) The effect of bagging on variance, bias, and mean squared error. *Technical report, AT& T Labs. Research*, Florham Park, NJ, pp. 1–20.

Chen,S.X. and Hall,P. (2003) Effects of bagging and bias correction on estimators defined by estimating equations. *Statist. Sinica*, **13**, 97–109.

Dougherty,E.R. (2001) Small-sample issues for microarray-based classification. *Comp. Funct. Genom.*, **2**, 28–34.

Dudoit,S. *et al*. (2002) Comparison of discrimination methods for the classification of tumors using gene expression data. *J. Am. Stat. Assoc.*, **97**, 77–87.

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

Efron,B. and Tibshirani,R. (1993) *An Introduction to the Bootstrap*, Chapman and Hall, London.

Efron,B. and Tibshirani,R. (1997) Improvements on cross-validation: the .632+ bootstrap method. *J. Am. Stat. Assoc.*, **92**, 548–560.

Friedman,J.H. and Hall,P. (2000) On bagging and nonlinear estimation. *Technical report, Department of Statistics, Stanford University*, Stanford, CA, pp. 1–17.

Geisser,S. (1975) The predictive sample reuse method with applications. *J. Am. Stat. Assoc.*, **70**, 320–328.

Hastie,T. and Tibshirani,R. (1990) *Generalized Additive Models*. Chapman and Hall, New York, USA.

Kim,S. *et al*. (2002) Strong feature sets from small samples. *J. Comp. Biol.*, **9**, 127–146.

Nguyen,D.V. and Rocke,D.M. (2002) Tumor classification by partial least squares using microarray gene expression data. *Bioinformatics*, **18**, 39–50.

Nguyen,D.V. and Rocke,D.M. (2004) On partial least squares dimension reduction for microarray-based classification: a simulation study. *Comp. Stat. Data Anal.*, **46**, 407–425.

Silverman,B. (1986) *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, London.

Stone,M. (1974) Cross-validatory choice and assessment of statistical predictions. *J. Roy. Statist. Soc. B.*, **36**, 111–147.

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

van't Veer,L.J. *et al*. (2002) Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, **415**, 530–536.

Wang,S. *et al*. (1997) A new test for outlier detection from a multivariate mixture distribution, *J. Comp. Grap. Stat.*, **6**, 285–299.