Original Paper

Gene Expression

Calculating Confidence Intervals for Prediction Error in Microarray Classification Using Resampling

Wenyu Jiang*[§], Sudhir Varma[§] and Richard Simon[§]
Biometric Research Branch, Division of Cancer Treatment and Diagnosis, National Cancer Institute, Rockville MD 20892-7434, USA

[§]These authors contributed equally to this work
[*]To whom correspondence should be addressed.

Running Head: Confidence Intervals for Prediction Error

Calculating Confidence Intervals for Prediction Error in Microarray Classification Using Resampling

Abstract

Motivation:

Cross-validation based point estimates of prediction accuracy are frequently reported in microarray class prediction problems. However these point estimates can be highly variable, particularly for small sample numbers, and it would be useful to provide confidence intervals of prediction accuracy.

Results:

We performed an extensive study of existing confidence interval methods and compared their performance in terms of empirical coverage and width. We developed a bootstrap case cross-validation (BCCV) resampling scheme and defined several confidence interval methods using BCCV with and without bias-correction.

The widely used approach of basing confidence intervals on an independent binomial assumption of the leave-one-out cross-validation errors results in serious under-coverage of the true prediction error. Two split-sample based methods previously proposed in the literature tend to give overly conservative confidence intervals. Using BCCV resampling, the percentile confidence interval method was also found to be overly conservative without bias-correction, while the bias corrected accelerated (BCa) interval method of Efron returns substantially anti-conservative confidence intervals. We propose a simple bias reduction on the BCCV percentile interval. The method provides mildly conservative inference under all circumstances studied and outperforms the other methods in microarray applications with small to moderate sample sizes.

Availability: Matlab and R files are available on requests from the authors.

Contact: WJ: jiangwen@mail.nih.gov SV: varmas@mail.nih.gov
RS: rsimon@mail.nih.gov

1 INTRODUCTION

Microarray based gene expression profiling is widely used in oncology research to predict clinical outcome such as response to therapy and occurrence of metastasis. Microarray experiments are typically conducted with a small to moderate number (*n*) of cases but a large number (*p*) of genes; the number of genes is usually orders of magnitude larger than the number of specimens ($n<<p$). Introduction to microarray technology and discussion of relevant issues in statistics can be found in Simon, Korn et al. (2003). For each specimen in the sample, the data consist of the gene expression measurements generated from the microarray experiment along with a class label indicating the outcome category. Given the observed data, a prediction model (predictor) can be trained to predict the outcome for future observations. An accurate prediction model helps to improve the correct diagnosis and/or proper treatment assignment for patients. Performance of the class prediction procedures are usually assessed by prediction error rates.

In such class prediction problems, it is common to report a point estimate for prediction error to assess how well a prediction model built on the observed sample generalizes to future data. If we have a large sample, we can split it into a training set (on which we develop the prediction model) and a test set on which we can evaluate the prediction model. The performance of the prediction model on the test set provides a point estimate for the true prediction error.

Because of the limited number of cases in most microarray studies, the split-sample approaches represents an inefficient use of the data relative to leave-one-out cross-validation (LOOCV) (Lachenbruch and Mickey, 1968) or other resampling methods (Molinaro, Simon and Pfeiffer, 2005). When the number of cases is small or moderate, both split-sample and cross-validation based estimates of prediction error are imprecise and confidence intervals should be reported.

With the split-sample method, the number of prediction errors in the test set has a Binomial distribution and a confidence interval for the prediction error can be based on that distribution. For LOOCV, however, the test set on which prediction of the *i*th case has n-2 specimens in common with the training set on which prediction of the *j*th is based, hence, the number of prediction errors is not binomial (Radmacher, McShane and Simon, 2002). Nevertheless, such Binomial distribution based confidence intervals are often used in the literature (Martin and Hirschberg, 1996).

Michiels, Koscielny and Hill (2005) applied a multiple random validation strategy to construct confidence intervals for prediction error. This method randomly splits the sample into learning and test sets of pre-specified sizes a large number of times. For each random split, it calculates a split-sample estimate of prediction error. Confidence intervals are obtained from the empirical percentiles of these estimates obtained from the random splits. In their paper, the method was applied directly to a number of microarray datasets. However, the validity and performance of this method were neither established nor investigated in their paper.

Limited work has been done in the literature to evaluate the merits of the fore-mentioned confidence interval procedures for microarray data. Through extensive empirical study, we find these existing methods are problematic in the microarray situation.

In this paper we propose a new method for estimating confidence intervals. We first develop a bootstrap case cross-validation procedure (BCCV) which is a modification of the resampling procedure of Fu, Carroll and Wang (2005). Their resampling procedure was originally proposed for the purpose of obtaining point estimates for prediction error in small sample problems. However, their method seriously underestimates the true prediction error because of the overlaps between the learning and test sets in boostrap samples (Jiang and Simon, 2006). In our proposed BCCV resampling procedure, the resampled learning and test sets do not overlap. We construct bootstrap percentile confidence intervals (Efron and Tibshirani, 1998) based on BCCV, but find the method tends to give overly-conservative results. We further employ the bias-corrected accelerated (BCa) method (Efron, 1987) on the BCCV resampling, but it tends to produce confidence intervals with serious under-coverage in microarray situations. We then propose a simple bias reduction approach to correct on the direct percentile confidence intervals using BCCV.

2 METHODS

2.1 Review of Confidence Interval Methods for Prediction Error

In this section, we first describe the general framework for microarray class prediction and then review some existing confidence interval estimation methods for prediction error.

In a microarray experiment with *n* independent specimens, we observe $x_i = (t_i, y_i)$, $i = 1, ..., n$, where t_i is a p-dimensional vector containing gene expression measurements and y_i is the response for subject i. Suppose the observations $x_1, ..., x_n$ are realizations of an underlying random variable $X = (T, Y)$ and the dichotomous response Y takes 0 or 1 values distinguishing the two classes of outcome. We are interested in the true prediction error ($\theta_n = E\left[I\left\{Y \neq r(T, x)\right\}\right]$) as a prediction accuracy measurement, where $I\{\}$ is an indicator function. The true prediction error is the probability that the prediction model $r(\cdot, \cdot)$ built on the observed data $x = \left(x_1, ..., x_n\right)$ misclassifies a future item arising from the random mechanism of X.

Since the number of genes (features) greatly exceeds the number of specimen, it is necessary to select a collection of genes to include in prediction modeling. This step is known as feature selection and is an important part of the algorithm of building prediction model $r(\cdot, \cdot)$. When point or interval estimation of the prediction error is based on resampling approaches, feature selection needs to be performed on every learning set arising from cross-validation or other type of resampling of the data. Failure to conduct feature selections within the loops of the resampling procedures results in serious overestimation on prediction accuracy (Simon, Radmacher et al. 2003; Ambroise and McLachlan, 2002).

Binomial Interval Based on Leave-One-Out Cross-Validation (LOOCV-Bin)

Leave-one-out cross-validation (LOOCV) is a popular method for estimating prediction error for small samples. The LOOCV estimate can be expressed as $\hat{\theta}_n{}^{LOOCV} = 1/n\sum_{i=1}^{n} I\left\{y_i \neq r(t_i, x_{(-i)})\right\}$, where $x_{(-i)}$ represents the leave-one-out learning set, which is the collection of data with observation x_i removed . It calculates the rate of misclassification when predicting for each specimen using a learning set containing all other observations in the sample. The distribution of the LOOCV estimate is typically unknown; hence it is not easy to measure the variability of the estimate or to construct the corresponding confidence intervals.

For specimen i , a LOOCV test error ($I\left\{y_i \neq r(t_i, x_{(-i)})\right\}$) takes 0 or 1 value, indicating whether the prediction model built on $x_{(-i)}$ misclassifies the response of the specimen. Inference for the true prediction error θ_n is often based upon an assumption that the LOOCV errors are independent Bernoulli trials with mean θ_n, and their sum ($n\hat{\theta}_n{}^{LOOCV}$) follows a binomial distribution denoted by Bin(*n*, θ_n). The corresponding $100(1-\alpha)\%$ binomial confidence interval is given by

$$\{\theta_n : \Pr(\mathrm{B} \leq n\hat{\theta}^{LOOCV} \mid x)\} < \alpha,$$

which consists of all plausible values of θ_n such that the probability of observing a binomial variable B (from Bin(*n*, θ_n)) no greater than the actual number of LOOCV errors is smaller than α .

Although the method is commonly used in the conventional framework when *n*>*p*, it is known that the assumption is invalid; in fact, the LOOCV errors are shown to have positive covariance (Lemma 1, Nadeau and Bengio, 2001). We will assess the performance of this confidence interval method in microarray application with *n*<<*p*.

Binomial Interval Based on Split-Sample (Split-Bin)

Split-sample method divides the sample x into a learning set x^{learn} of size n^{learn} and a test set x^{test} of size n^{test} such that $n^{learn} + n^{test} = n$; a prediction model $r(\cdot, x^{learn})$ is built on the learning set; the split sample estimate $\hat{\theta}_n^{Split}$ for prediction error is the average error rate of applying the prediction model on the test set. For specimen i in the test set, a test error ($I\left\{y_i \neq r(t_i, x^{learn})\right\}$) takes 0 or 1 value, indicating whether the prediction model misclassifies the response of the specimen.

Unlike the LOOCV error mentioned in the previous method, the test errors for the split sample method are truly independent Bernoulli variables. A Binomial confidence interval for θ_n can be derived by using the Bernoulli distribution on the test errors. This approach provides exact inference for the true prediction error of the split-sample learning set x^{learn}. It should work well for θ_n on a sample large enough so that building a prediction model on the split-sample learning set is almost as accurate as on the sample itself, and the test set is also large enough.

We study this method in microarray applications where the sample sizes are typically not large, and we investigate the performance of this method under different learning-and-test-set allocations (i.e. n^{learn} versus n^{test}).

Multiple Random Validation Percentile Interval (MRVP)

Michiels, Koscielny and Hill (2005) proposed a multiple random validation strategy for confidence interval estimation for the true prediction error. With pre-specified sizes of learning and test sets, the method randomly split the samples into learning and test sets and obtains a split-sample estimate of the true prediction error; this procedure is repeated a large number of times to obtain replicates of such estimates. The $100(1-\alpha)\%$ percentile of these estimates is used as the $100(1-\alpha)\%$ upper confidence limit.

In the original paper, the authors applied the method directly to a number of datasets with a variety of learning-and-test set allocations; the number of patients in the learning sets ranges from ten to a maximum value, which can give rise to test sets with as few as one patient from each class. In our opinion, the learning-and-test-set allocation should be assessed with caution. On one hand, when the learning set size n^{learn} is much smaller than the sample size *n*, the split sample estimates will seriously overestimate the true prediction error (Molinaro et al. 2005). We will evaluate how this affects the performance of the confidence intervals. On the other hand, when the test set size n^{test} becomes too small, the split-sample estimates will be too discrete to produce meaningful inference. For example, with two patients in each test set, the MRVP intervals will only take three values, 0, 0.5 and 1.

2.2 Confidence Intervals using Bootstrap Case Cross-Validation

Our new confidence interval methods will be based on the following **bootstrap case cross-validation (BCCV)** procedure. From the original sample *x*, we draw a bootstrap sample of size n using simple random sampling with replacement. This is repeated *B* times and we denote the bootstrap samples by $x_1^{*,b},...,x_n^{*,b}$ where $b=1,...B$. Define a vector $(m_1^b,...,m_n^b)$ whose entry m_i^b is the number of times that the original observation x_i is selected in bootstrap sample *b*. We then apply a cross-validation procedure on a bootstrap sample. That is, we leave out *all the replications of an original observation (case)* at a time and use what remains in the bootstrap sample to predict for the left-out observation. The resulting cross-validation estimate can be formulated as

$$\hat{\theta}^{*,b}=\frac{1}{n}\sum_{i=1}^{n}m_i^b I\{y_i \neq r(t_i,x_{(-i)}^{*,b})\}$$

where $x_{(-i)}^{*,b}$ is the bootstrap sample *b* excluding the replications of the original observation *i*. In this way, the BCCV resampling avoids overlaps between the learning set and the test set generated in the cross validation procedure.

Bootstrap Case Cross-Validation Percentile Interval (BCCVP)

The BCCV resampling gives rise to an estimate $\hat{\theta}^{BCCV}=\frac{1}{B}\hat{\theta}^{*,b}$ for the true prediction error. We can construct percentile confidence intervals utilizing the empirical distribution of the cross-validation estimates calculated on the bootstrap replications; an approximate $100(1-\alpha)\%$ level upper confidence interval for the true prediction error can be obtained as $(0, \hat{\theta}^*_{(1-\alpha)}]$ where $\hat{\theta}^*_{(1-\alpha)}$ is the $100(1-\alpha)$ empirical percentile of $\hat{\theta}^{*,b}$, $b = 1, \ldots B$. This procedure is in the general framework of bootstrap percentile interval approach (Efron and Tibshirani, 1998).

Bias Corrected Accelerated Interval Using Bootstrap Case Cross-Validation (BCCV-BCa)

The BCCV procedure is a modification on the bootstrap cross-validation of Fu, Carroll and Wang (2005). The BCCV procedure avoids overlaps between the resampled learning and test sets and can be used for interval estimation. However, a simple probability calculation indicates that the resampled learning set $x^{*,b}_{(-i)}$ in the BCCV contains only about .632(n-1) unique observations (Efron, 1983). This leads to an overestimation on the true prediction error and can result in overly conservative inference.

Efron (1987) proposed a bias corrected accelerated (BCa) method which is known to improve on percentile intervals for a population parameter in traditional *n*>*p* framework. We apply the BCa algorithm as described in Efron and Tibshirani (1998) as an initial approach to correct on the BCCVP intervals and the details of the algorithm are presented in the supplementary materials. The BCa method assumes the existence of a transformation such that the parameter of interest and its estimate are transformed into a statistic having an asymptotically normal distribution. However, this assumption may not be valid in prediction error estimation problems. In Section 3.1, we evaluate how the BCa method corrects on the BCCVP intervals.

Bootstrap Case Cross-Validation Percentile Interval with Bias Reduction (BCCVP-BR)

We propose a simpler approach to correct the BCCVP confidence intervals through bias reduction. That is, we approximate the $100(1-\alpha)\%$ level upper confidence interval by $(0, \hat{\theta}^*_{(1-\alpha)} - (\hat{\theta}^{BCCV} - \hat{\theta}^{LOOCV})]$, where $\hat{\theta}^*_{(1-\alpha)}$ is the upper confidence limit using the

BCCVP method. The term $\hat{\theta}^{BCCV} - \hat{\theta}^{LOOCV}$ corresponds to the amount that the BCCV estimate exceeds the LOOCV estimate $\hat{\theta}^{LOOCV}$. It approximates the bias of the BCCV estimate since the LOOCV procedure is known to give an almost unbiased estimate for the true prediction error.

3 RESULTS

In this section, we compare the performance of the existing confidence interval methods and the BCCV based methods through extensive simulations and an application to a lymphoma dataset.

Though all these methods are capable of estimating two-sided confidence intervals, we focus on upper confidence intervals. An upper confidence interval for the true prediction error is of the form $[0,u]$ where $u \leq 1$ is the value below which the true prediction error is expected to lie with probability $(1-\alpha)$. In microarray class prediction context, an upper confidence interval for true prediction error is of primary interest since it helps to assess whether a prediction model predicts the classes better than chance.

3.1 Simulation Study

We compare the methods described in Section 2 through simulations. Synthetic data are generated with half of the patients in class 0, half in class 1. Gene expression levels for each patient are generated from a normal distribution with covariance matrix $\Sigma = (\sigma_{ij}), i, j = 1, \ldots, p$, where the nonzero entries are $\sigma_{ii} = 1$ and $\sigma_{ij} = 0.2$ with $0 <| i - j |\leq 5$. For class 0 patients, gene expression levels are generated with mean 0. For class 1 patients, we generate a fixed proportion of the genes with mean μ and the rest with mean 0.

In each simulation run, we generate a sample of *n* patients, each with *p* genes. We compute upper confidence limits in each simulation run for a number of nominal coverage levels using the aforementioned confidence interval procedures. In each simulation run, we also generate 1000 independent data from the same random mechanism as above, use it as a test set to calculate the prediction error rate of the sample and call it the "true" prediction error $\tilde{\theta}_n$. A simulation study contains R=1000 simulation runs. We compute the average and the standard deviation of the upper confidence limits across all simulation runs, as well as the empirical coverage probability, which is the proportion of the simulation runs with upper confidence limits greater than the corresponding "true" prediction errors $\tilde{\theta}_n$.

We conduct eight simulation studies by varying the sample size, the number of genes, the proportion of differentially expressed genes and the classifier. Simulation 1 is composed of n=40 patients in each sample with p=1000 genes, 2% of the genes in class 1 have non-zero means $\mu = 0.8$. Simulation 2 considers the "null" case where there is no difference between the two classes. Simulation 3 considers a smaller sample size of n=20 but otherwise has the same composition as Simulation 1. Simulation 4 considers the *n*>*p* situation with n=40 and p=10, half of the genes in class 1 are differentially expressed with mean $\mu = 0.8$; all genes are used in prediction and no feature selection is done. In the *n*<<*p* situations, 10 features with the largest absolute-value *t*-statistics are selected for every prediction model, and the feature selection steps are repeated on all split-sample or resampled learning sets generated throughout the procedures to compute confidence intervals. In Simulations 1-4, we use the diagonal linear discriminant analysis (DLDA) as the classifier to discriminate the classes and the algorithm is available through the function "dlda" in the library "supclust" in the statistical package R (Ihaka and Gentleman, 1996). For simplicity of presentation, we only include outcome of Simulations 1-4 in the paper and descriptions for Simulations 1-4 are listed in Table 1. Descriptions and results for Simulations 5-8 are presented in the supplementary material.

Insert Table 1 about here.

Table 2 presents the 80% and 90% upper confidence intervals using the BCCVP, BCCVP-BR, BCCV-BCa, LOOCV-Bin, Split-Bin (1/3) with 1/3 patients in the test set, and MRVP (1/3) . The MRVP (1/3) method randomly splits the sample 100 times; each split results in a test set of 1/3 and a learning set of 2/3 of the patients in the sample. In addition to the empirical coverage probability, we report the average and standard deviation of the upper confidence limits calculated across all simulation replications. We also include for each simulation in Table 1 the "true" prediction errors $\tilde{\theta}_n$ averaged across all simulation runs. Comparisons of the coverage properties of these methods are illustrated in Figure 1. We plot the empirical coverage probabilities of the upper confidence intervals against the nominal coverage levels of 70%, 80%, 90% and 95%. The yellow reference line indicates perfect agreements between the empirical and the nominal coverage.

The BCCVP-BR intervals are slightly conservative and are more accurate than the BCCVP intervals in terms of coverage in all simulations. The LOOCV-Bin and the BCCV-BCa methods work well in Simulation 4 in the traditional *n*>*p* scenario, but they both suffer from substantial under-coverage in Simulations 1-3 with *n*<<*p*, hence are not suitable for data typical of microarray applications. The BCCV-BCa method is less competitive than other methods also because it gives confidence limits with the largest standard deviation. The Split-Bin (1/3) and the MRVP (1/3) both give conservative confidence intervals. The MRVP (1/3) has more serious over-coverage than the BCCVP-BR, although the two methods perform about the same at 95% level in Simulations 1 and 3. In Simulations 1 and 2, the Split-Bin (1/3) is comparable to the BCCVP-BR at nominal levels of 90% and 95% in terms of coverage but is more conservative than the BCCVP-BR at levels below 90%. The BCCVP-BR intervals are more accurate than the Split-Bin (1/3) intervals in terms of coverage in Simulation 3 with smaller sample size and in Simulation 4 with *n*>*p*. Overall, the BCCVP-BR intervals perform the best among the methods across all simulation studies and all nominal coverage levels of practical interest.

Insert Table 2 and Figure 1 about here.

The Split-Bin (1/3) and the MRVP (1/3) have good performance at the extreme nominal coverage levels. We further examine how these methods behave with different learning-and-test-set allocations. In Figure 2, we compare the coverage properties of the BCCVP-BR intervals to the Split-Bin and MRVP methods with 2/3, 1/3 and 1/10 observations in the test sets respectively. Among the Split-Bin intervals, the Split-Bin (2/3) gives more serious over-coverage than the Split-Bin (1/3) in Simulations 1, 3 and 4 because a reduction on the learning set size leads to overestimation of the true prediction error. In

Simulation 2, varying the learning set size does not have such an obvious impact since there are no differences between the two classes. The Split-Bin (1/10) is based on very small test sets and is the least stable in all simulations. The Split-Bin (1/3) method clearly works better than Split-Bin (2/3) and Split-Bin (1/10).The MRVP (2/3) methods give more conservative intervals than MRVP (1/3) and MRVP (1/10).

Insert Figure 2 about here.

To provide further insight into the comparison of the methods, we consider another performance measure, how often a method provides upper confidence limits below 0.5. An upper confidence limit exceeding 0.5 is not very useful, because in such a situation we cannot reject the null hypothesis that the prediction is no better than chance. Consider, for example, a method for upper confidence limits that gives 1 with probability 0.95 and 0 with probability 0.05. This gives an exact 95% coverage of the true prediction error but the confidence limits themselves are not useful at all.

In Figure 3, we plot the proportion of simulation runs giving upper confidence intervals smaller than 0.5 against the nominal coverage levels for the three Split-Bin methods and the three MRVP methods. The MRVP (1/3) method outperforms MRVP (2/3) and MRVP (1/10) according to this performance measure. In Simulations 1, 3, and 4 where the genes of the two classes are expressed differently, the MRVP (1/10) and Split-Bin (1/10) both perform poorly, while the BCCVP-BR always performs the best. In Simulation 1, for instance, the 90% intervals using MRVP (1/10) falls below 0.5 less than 10% of the times, while the 90% BCCVP-BR intervals are smaller than 0.5 around 50% of the times; although the MRVP (1/10) gives slightly better coverage than the BCCVP-BR in this example. In other words, compared to BCCVP-BR method, the number of times we reach conclusive inference is much smaller when using MRVP (1/10) or Split-Bin (1/10) intervals. In the “null” situation, an upper confidence interval below 0.5 leads to a false positive conclusion on the prediction model. Since we focus this investigation on the conservative approaches, the panel of Simulation 2 in Figure 3 confirms that the false positive rates are controlled below α for all $100(1-\alpha)\%$ upper intervals.

Insert Figure 3 about here.

3.2. Application to Lymphoma Data

Rosenwald et al. (2002) performed a microarray study using tumors for patients with large-B-cell lymphoma. The study measured 7399 genes on a total of N=240 patients and developed a classifier to distinguish the germinal-center B-cell like subgroup from the other subgroups. We define the germinal-center B-cell-like as class 1, the other subgroups as class 0 accordingly. In the following analysis, we draw a random sample of size n, equally divided between class 0 and class 1; the upper confidence limits are computed as described in Section 2. To evaluate coverage, we calculate the "true" prediction error $\tilde{\theta}_n$ using the N-n patients not selected in the sample as an independent test set. The number of bootstrap repetitions is 100 in the BCCVP, BCCVP-BR, BCCV-BCa methods, the number of random splits is 100 in the MRVP method, 10 features with the largest absolute value t-statistics are repeatedly selected throughout and the results are reported in Table 3.

The analysis illustrates that the BCCVP-BR method gives reasonable confidence intervals for the lymphoma data with small to moderate sample sizes. It is worth noting that the LOOCV-Bin interval does not cover the "true" prediction error $\tilde{\theta}_n$ at either 80% or 90% nominal level in the analysis with n=40. The BCCV-BCa interval barely covers $\tilde{\theta}_n$ at 80% level when n=20 and the Split-Bin method with 1/3 sample in the test set gives rise to upper confidence limits greater than 0.5 when n=20. Performance of the confidence interval methods, however, cannot be judged on a one-time application on a real data example, but it is not possible to evaluate coverage probabilities for most datasets because the total number of specimens is typically much smaller and the true prediction error, $\tilde{\theta}_n$, is not available. Even on a large dataset such as the lymphoma data, repeated random sampling on the complete dataset does produce replicates of confidence intervals; but such random samples are correlated and it is rather uncertain how to analyze the properties of these confidence intervals. Consequently, comparison of the confidence interval procedures has to be based on extensive simulation studies as in the previous subsection.

Insert Table 3 about here.

4. Discussion

Inference for the prediction error is more challenging in the *n*<<*p* scenarios than in the traditional context with *n*>*p*. The prediction error should ideally be assessed on an independent large test set, but this is often impossible because of the relatively small number of specimens in microarray studies. For this reason, inference for prediction error in this context often has to rely on partitioning or resampling the observed data to form learning and test sets. When the number of features exceeds the number of specimens, feature selection is a crucial part of the prediction model and has to be performed prior to the class prediction step on every learning set arising from the data partitioning or resampling.

Because of the large amount of noise and high dimensionality characterizing microarray data, conservative inference is preferable in order to avoid false positive claims on prediction models. When *n*>*p*, the LOOCV-Bin method is a common choice for inference; the BCa method is well-known to correct on the bootstrap percentile intervals. But both methods give anti-conservative confidence intervals when *n*<<*p* in the study in this paper. Both methods are built on strong distributional assumptions that are either invalid or hard to verify.

In a microarray study featuring small to moderate sample sizes, the chances of reaching conclusive inference for the prediction is very low for the 95% or more extreme level confidence intervals. Confidence levels of 80% to 90% are more reasonable choices in the presence of high dimensionality and sparseness of the data.

Additional simulation studies are presented in the supplement, in which we compare the confidence interval procedures in situations with a larger sample size (n=100), a larger number of genes (p=3000), a different signal level separating the classes and a different classifier. With a larger sample size or a stronger signal, we see an increase in the percentages of reaching conclusive confidence intervals (upper confidence limit < 0.5). In most of the simulations, we use the simple DLDA classifier since its performance is better than more sophisticated classifiers (Dudoit, Fridlyand and Speed, 2002). In one of simulations (included in the supplementary materials), we replace the DLDA classifier by the supported vector machines (SVM) classifier (Vapnik, 1998) but it does not change the overall comparison of the confidence interval procedures.

The MRVP and Split-Bin methods both give overly conservative confidence intervals when the learning set size n^{learn} is small, and they are less likely to produce practically meaningful intervals when the test set size n^{test} is small. This is not surprising since both methods make use of split-sample approach, which is known to work well only for problems with large sample sizes (Molinaro et al, 2005). With small to moderate sample sizes, a good allocation of the sample in these methods is to assign roughly one third of the observations in the test sets and two thirds in the learning sets.

In the Split-Bin and LOOCV-Bin methods, we construct direct binomial confidence intervals on account of the true or the assumed test error distribution. It is also possible to employ normal approximation to binomial distribution and compute instead the normal confidence intervals. But with a small number of Bernoulli trials, it is often necessary to carry out continuity corrections on these normal approximations; although it is difficult to specify a correction which is suitable in any application context. A variety of continuity corrections in this effort has been discussed by Martin and Hirschberg (1996).

The bootstrap cross-validation (BCV) resampling of Fu et al., 2005 initiates a bootstrap approach on cross-validation procedure for point estimation on the true prediction error. However, this method is prone to give highly anti-conservative inference on the true prediction error due to the overlaps between the resampled learning and test sets. In the supplementary materials, we explore confidence interval methods based on the original BCV resampling, but the methods lead to substantial under-coverage. The BCCV resampling is developed on the BCV technique; it amends the flaws of the BCV method and extends its use to confidence interval estimation. The BCCVP-BR method is free of distributional assumptions; it gives slightly conservative confidence intervals for the true prediction error, thus effectively avoids overly optimistic claims on the prediction; it performs better than other methods considered in this paper in situations with small to moderate sample sizes and large numbers of features characterizing microarray applications. One drawback of the BCCVP-BR method is that the confidence intervals are not strictly confined in the interval between 0 and 1, although this only occurs sporadically at the extreme confidence levels. In an attempt to circumvent this problem, we fit a beta density on the BCCV estimates $\hat{\theta}^{*,b}$, $b = 1,...B$, and shift the mean of the beta distribution to the LOOCV estimates; the quantiles of the resulting distribution are used as the confidence intervals. Preliminary study suggests this approach provides slight improvements on the BCCVP-BR intervals at the extreme confidence levels.

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.

Dudoit, S., Fridlyand, J. and Speed, T. (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. (1987) Better bootstrap confidence intervals. (with discussion.) *J. Am. Stat. Assoc.*, **82**, 171-200.

Efron, B. and Tibshirani, R. (1998) *An Introduction to the Bootstrap*. Chapman and Hall.

Fu, W. Carroll, R. J. and Wang, S. (2005) Estimating misclassification error with small samples via bootstrap cross-validation. *Bioinformatics,* **21**, 1979-1986.

Ihaka, R. and Gentleman, R. (1996) R: a language for data analysis and graphics. *J. Comput. Graph. Stat.,* **5**, 299-314.

Jiang, W. and Simon, R. (2006) A comparison of bootstrap methods and an adjusted bootstrap approach for estimating prediction error in microarray classification. *Technical Report.* Biometric Research Branch, Division of Cancer Treatment and Diagnosis, NCI. http://linus.nci.nih.gov/~brb/TechReport.htm

Lachenbruch P.A. and Mickey M. R. (1968) Estimation of error rates in discriminant analysis. Technometrics 10:1-11.

Martin JK, Hirschberg DS. (1996). Small sample statistics for classification error rates II: Confidence intervals and significance tests**.** *Technical Report*, ICS-TR-96-22.

Michiels S, Koscielny S and Hill C. (2005) Prediction of cancer outcome with microarrays: a multiple random validation strategy.*Lancet*, **365**, 488-492.

Molinaro, A. M., Simon, R. and Pfeiffer, R. M. (2005) Prediction error estimation: a comparison of resampling methods. *Bioinformatics,* **21**, 3301-3307.

Nadeau C, Bengio Y. (2003) Inference for the generalization error. *Machine Learning*, 52, 239-281.

Radmacher, M. D., McShane, L. M. and Simon, R. (2002) A paradigm for class prediction using gene expression profiles. *J. Comput. Biology*, **9**, 505-511.

Rosenwald, A., Wright, G., Chan, W. C., Connors, J. M., Campo, E., Fisher, R. I., Gascoyne, R. D., Muller-Hermelink, H. K., Smeland, E. B. and Staudt, L. M. (2002) The use of molecular profiling to predict survival after chemotherapy for diffuse large B-cell lymphoma. New Engl. J. Med., **346**, 1937-1947.

Simon, R., Korn, E. L., McShane, L. M. Radmacher, M. D., Wright, G. W. and Zhao, Y. (2003) *Design and Analysis of DNA Microarray Investigations*. Springer.

Simon, R., Radmacher, M. D., Dobbin, K. and McShane, L. M. (2003) Pitfalls in the use of DNA microarray data for diagnostic and prognostic classification. *J. Natl Cancer Inst.* **95**, 14-18.

Vapnik, V. (1998) *Statistical Learning Theory*, 1st ed., New York: John Wiley.

Table 1. Description of Simulations 1 to 4. Diagonal linear discriminant analysis (DLDA) is used in class prediction algorithm. Number of simulation replications is 1000. The quantity $\tilde{\theta}_n$ is the "true" prediction error for each sample evaluated on 1000 independent test data. Average of $\tilde{\theta}_n$ is calculated across all simulation replications.

Simulation	n	p	% differential genes	Average of $\tilde{\theta}_n$
1	40	1000	2%	.264
2	40	1000	0%	.500
3	20	1000	2%	.384
4	40	10	50%	.274

Table 2. Upper Confidence Intervals for Simulations 1-4. Methods include bootstrap case cross-validation percentile interval (BCCVP), BCCVP with bias reduction (BCCVP-BR), bias corrected accelerated interval using BCCV (BCCV-BCa) , binomial interval based on leave-one-out cross-validation (LOOCV-Bin), binomial interval based on split-sample (Split-Bin) and multiple random validation percentile (MRVP). Number of bootstrap repetitions is 100 in BCCVP, BCCVP-BR, BCCV-BCa and number of random splits is 100 in MRVP.

	Simulation 1		Simulation 2		Simulation 3		Simulation 4	
Nominal levels	80%	90%	80%	90%	80%	90%	80%	90%
BCCVP								
Coverage Probability	1	1	.998	1	1	1	.895	.964
Average Confidence Limit	.533	.619	.679	.758	.689	.782	.369	.415
SD of Confidence Limit	.088	.089	.051	.049	.076	.072	.078	.085
BCCVP-BR								
Coverage Probability	.928	.992	.841	.939	.883	.969	.802	.933
Average Confidence Limit	.425	.511	.671	.749	.629	.722	.346	.392
SD of Confidence Limit	.142	.143	.162	.159	.207	.202	.082	.087
BCCV-BCa								
Coverage Probability	.628	.752	.725	.815	.719	.819	.815	.911
Average Confidence Limit	.353	.423	.643	.704	.585	.661	.362	.408
SD of Confidence Limit	.213	.219	.238	.219	.297	.278	.100	.105
LOOCV-Bin								
Coverage Probability	.731	.832	.705	.758	.782	.854	.858	.932
Average Confidence Limit	.346	.378	.585	.617	.530	.574	.351	.384
SD of Confidence Limit	.132	.134	.156	.153	.196	.191	.075	.076
Split-Bin (1/3 in test set)								
Coverage Probability	.951	.985	.878	.934	.933	.980	.901	.950
Average Confidence Limit	.481	.536	.637	.688	.659	.728	.435	.491
SD of Confidence Limit	.139	.137	.123	.117	.177	.160	.124	.124
MRVP (1/3 in test set)								
Coverage Probability	.976	.994	.937	.992	.949	.986	.889	.953
Average Confidence Limit	.433	.485	.600	.651	.573	.654	.365	.410
SD of Confidence Limit	.092	.094	.057	.056	.112	.113	.079	.082

SD: Standard Deviation

Table 3. Upper Confidence Limits for Lymphoma Data. Methods include the bootstrap case cross-validation percentile interval (BCCVP), BCCVP with bias reduction (BCCVP-BR), bias corrected accelerated interval using BCCV (BCCV-BCa) , binomial interval based on leave-one-out cross-validation (LOOCV-Bin), binomial interval based on split-sample (Split-Bin) and multiple random validation percentile (MRVP).

	n=20		n=40		n=100	
Nominal levels	80%	90%	80%	90%	80%	90%
BCCVP	.400	.450	.275	.375	.230	.270
BCCVP-BR	.235	.285	.194	.294	.203	.243
BCCV-BCa	.150	.250	.225	.275	.200	.230
LOOCV-Bin	.202	.245	.162	.190	.199	.217
Split-Bin (1/3 in test set)	.585	.667	.199	.251	.410	.447
MRVP (1/3 in test set)	.333	.333	.214	.214	.235	.265

The “true” prediction errors for n=20, 40 and 100 are .150, .195 and .107, evaluated on test sets of the remaining 240-n patients respectively**.**

Figure Legends

Figure 1. Comparison of confidence interval procedures. Empirical coverage probabilities are plotted against nominal confidence levels of 70%, 80%, 90% and 95% for the bootstrap case cross-validation percentile interval (BCCVP), BCCVP with bias reduction (BCCVP-BR), bias corrected accelerated interval using BCCV (BCCV-BCa) , binomial interval based on leave-one-out cross-validation (LOOCV-Bin), binomial interval based on split-sample (Split-Bin) with 1/3 sample in the test set and multiple random validation percentile interval (MRVP) with 1/3 sample in the test set. The yellow line is the 45° line for reference.

Figure 2. Comparison of coverage properties for the binomial intervals based on split-sample (Split-Bin) and the multiple random validation percentile intervals (MRVP) with 2/3, 1/3 and 1/10 samples in the test sets. Empirical coverage probabilities are plotted against nominal confidence levels of 70%, 80%, 90% and 95%. Also displayed are results using the bootstrap case cross-validation percentile interval with bias reduction (BCCVP-BR) and the 45° yellow line for reference.

Figure 3. Comparison of chances to reach conclusive confidence intervals using the binomial intervals based on split-sample (Split-Bin) and the multiple random validation percentile intervals (MRVP) with 2/3, 1/3 and 1/10 samples in the test sets. Proportions of simulated upper confidence intervals falling below 0.5 are plotted against nominal confidence levels of 70%, 80%, 90% and 95%. Also displayed are results using the bootstrap case cross-validation percentile interval with bias reduction (BCCVP-BR).

Figure 1

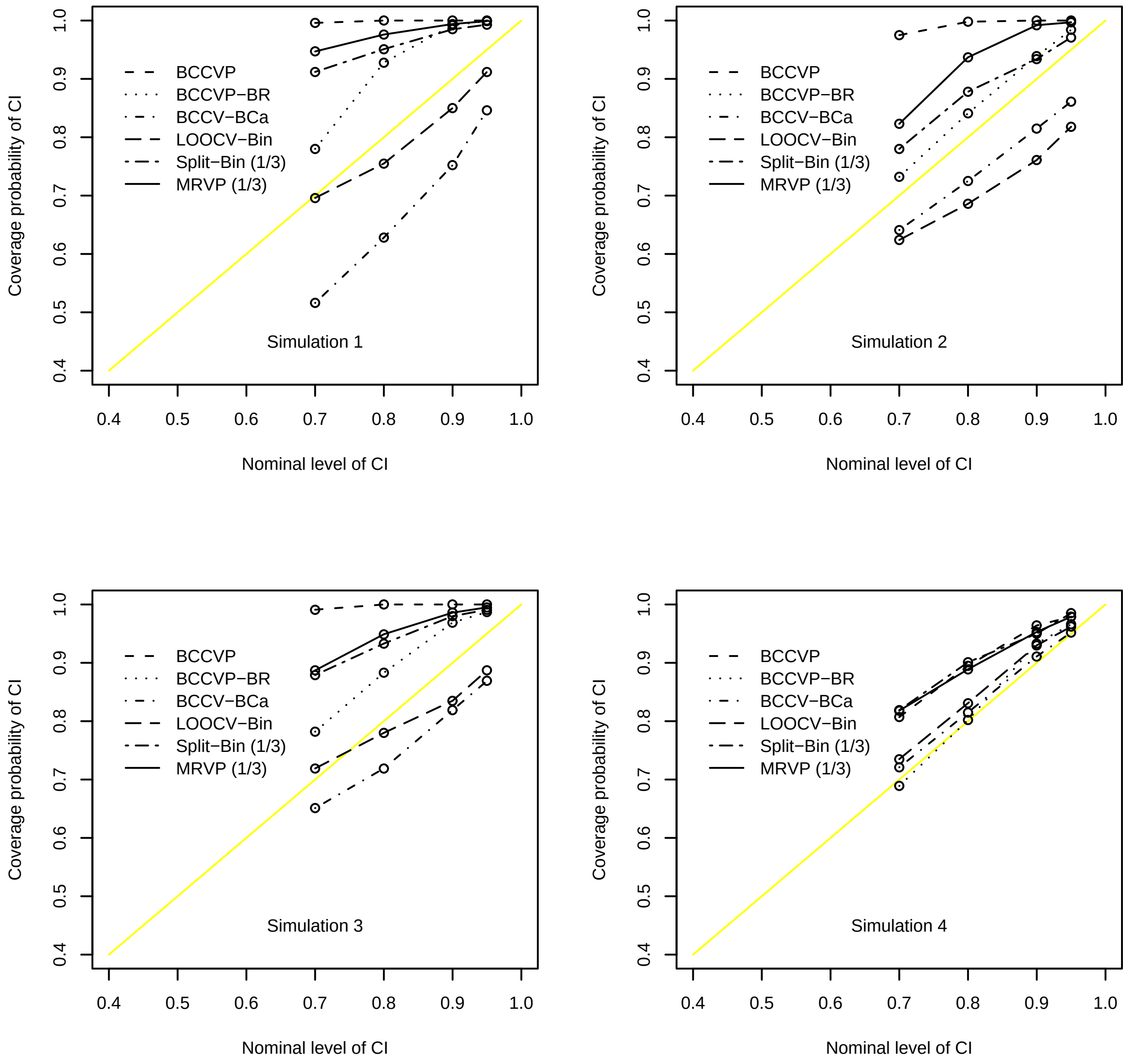

Coverage probability of CI
1.0
0.9
0.8
0.7
0.6
0.5
0.4
BCCVP
BCCVP−BR
BCCV−BCa
LOOCV−Bin
Split−Bin (1/3)
MRVP (1/3)
Simulation 1
0.4
0.5
0.6
0.7
0.8
0.9
1.0
Nominal level of CI
Simulation 2
Simulation 3
Simulation 4

Figure 2

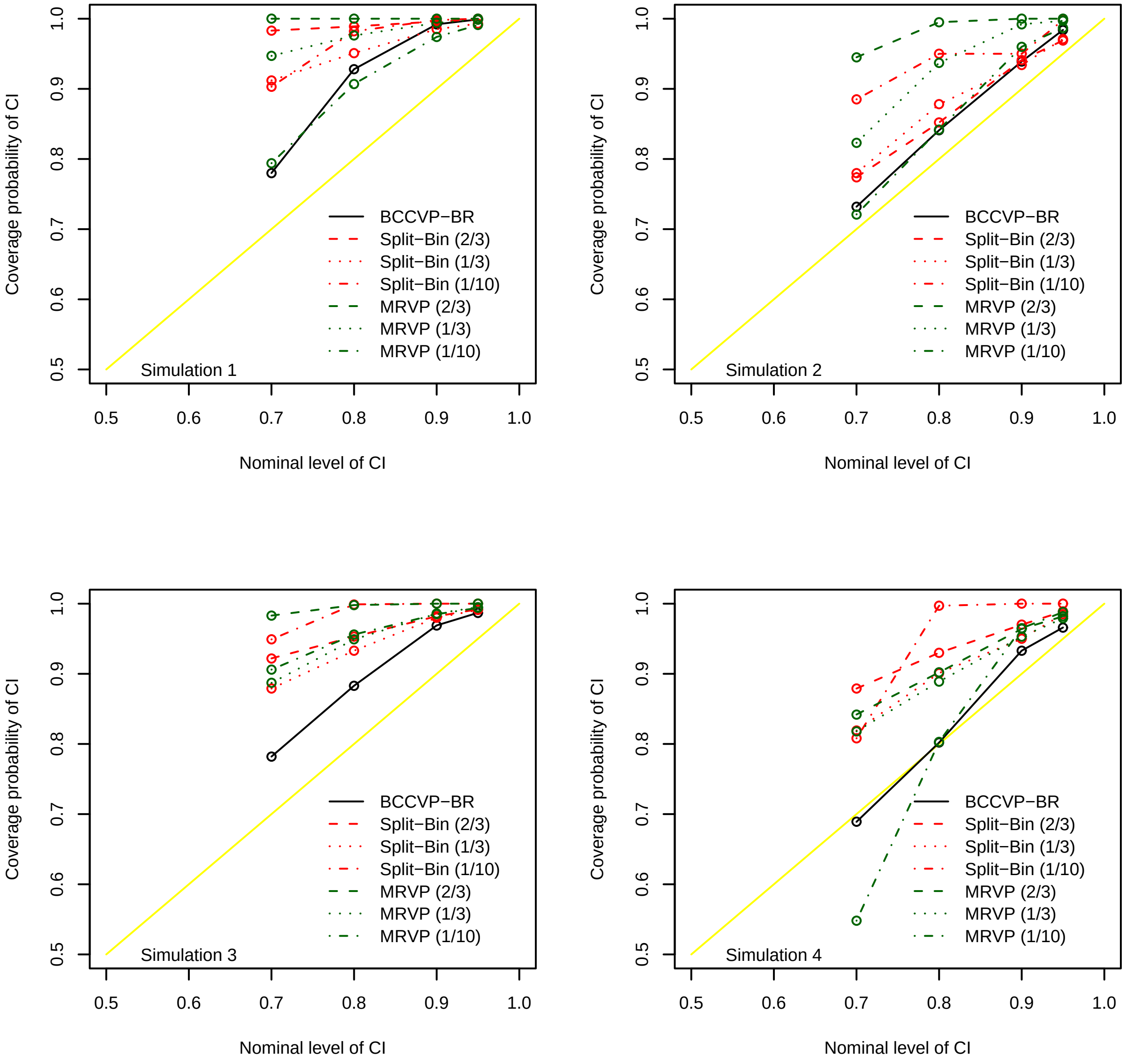
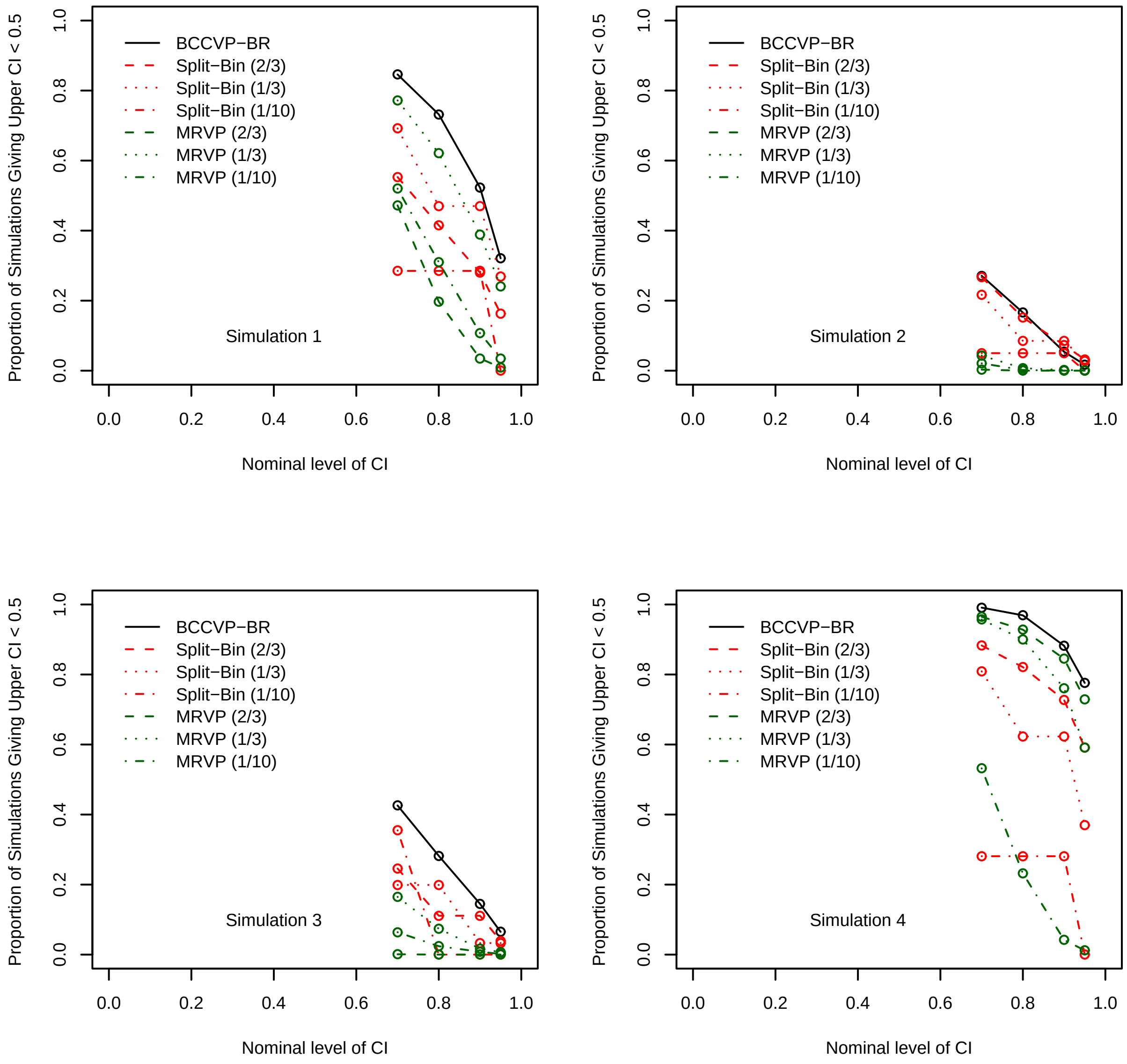

Coverage probability of CI
0.5
0.6
0.7
0.8
0.9
1.0
Nominal level of CI
0.5
0.6
0.7
0.8
0.9
1.0
BCCVP−BR
Split−Bin (2/3)
Split−Bin (1/3)
Split−Bin (1/10)
MRVP (2/3)
MRVP (1/3)
MRVP (1/10)
Simulation 1
Simulation 2
Simulation 3
Simulation 4

Figure 3


Proportion of Simulations Giving Upper CI < 0.5
Nominal level of CI
0.0
0.2
0.4
0.6
0.8
1.0
BCCVP−BR
Split−Bin (2/3)
Split−Bin (1/3)
Split−Bin (1/10)
MRVP (2/3)
MRVP (1/3)
MRVP (1/10)
Simulation 1
Simulation 2
Simulation 3
Simulation 4