BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 15 2005, pages 3301–3307
doi:10.1093/bioinformatics/bti499

*Data and text mining*

# Prediction error estimation: a comparison of resampling methods

Annette M. Molinaro[1,3,*], Richard Simon[2] and Ruth M. Pfeiffer[1]

[1]Biostatistics Branch, Division of Cancer Epidemiology and Genetics and [2]Biometric Research Branch, Division of Cancer Treatment and Diagnostics, NCI, NIH, Rockville, MD 20852 USA and [3]Department of Epidemiology and Public Health, Yale University School of Medicine, New Haven, CT 06520, USA

Received on April 6, 2005; revised on April 28, 2005; accepted on May 12, 2005
Advance Access publication May 19, 2005

**ABSTRACT**

**Motivation:** In genomic studies, thousands of features are collected on relatively few samples. One of the goals of these studies is to build classifiers to predict the outcome of future observations. There are three inherent steps to this process: feature selection, model selection and prediction assessment. With a focus on prediction assessment, we compare several methods for estimating the 'true' prediction error of a prediction model in the presence of feature selection.

**Results:** For small studies where features are selected from thousands of candidates, the resubstitution and simple split-sample estimates are seriously biased. In these small samples, leave-one-out cross-validation (LOOCV), 10-fold cross-validation (CV) and the .632+ bootstrap have the smallest bias for diagonal discriminant analysis, nearest neighbor and classification trees. LOOCV and 10-fold CV have the smallest bias for linear discriminant analysis. Additionally, LOOCV, 5- and 10-fold CV, and the .632+ bootstrap have the lowest mean square error. The .632+ bootstrap is quite biased in small sample sizes with strong signal-to-noise ratios. Differences in performance among resampling methods are reduced as the number of specimens available increase.

**Contact:** annette.molinaro@yale.edu

**Supplementary Information:** A complete compilation of results and R code for simulations and analyses are available in Molinaro *et al*. (2005) (http://linus.nci.nih.gov/brb/TechReport.htm).

## 1 INTRODUCTION

In genomic experiments one frequently encounters high dimensional data and small sample sizes. Microarrays simultaneously monitor expression levels for several thousands of genes. Proteomic profiling studies using SELDI-TOF (surface-enhanced laser desorption and ionization time-of-flight) measure size and charge of proteins and protein fragments by mass spectroscopy, and result in up to 15 000 intensity levels at prespecified mass values for each spectrum. Sample sizes in such experiments are typically $<100$.

In many studies, observations are known to belong to predetermined classes and the task is to build predictors or classifiers for new observations whose class is unknown. Deciding which genes or proteomic measurements to include in the prediction is called *feature selection* and is a crucial step in developing a class predictor. Including too many noisy variables reduces accuracy of the prediction and may lead to over-fitting of data, resulting in promising but often non-reproducible results (Ransohoff, 2004).

Another difficulty is model selection with numerous classification models available. An important step in reporting results is assessing the chosen model's error rate, or generalizability. In the absence of independent validation data, a common approach to estimating predictive accuracy is based on some form of resampling the original data, e.g. cross-validation. These techniques divide the data into a learning set and a test set, and range in complexity from the popular learning-test split to $v$-fold cross-validation, Monte-Carlo $v$-fold cross-validation and bootstrap resampling. Few comparisons of standard resampling methods have been performed to date, and all of them exhibit limitations that make their conclusions inapplicable to most genomic settings. Early comparisons of resampling techniques in the literature are focussed on model selection as opposed to prediction error estimation (Breiman and Spector, 1992; Burman, 1989). In two recent assessments of resampling techniques for error estimation (Braga-Neto and Dougherty, 2004; Efron, 2004), feature selection was not included as part of the resampling procedures, causing the conclusions to be inappropriate for the high-dimensional setting.

We have performed an extensive comparison of resampling methods to estimate prediction error using simulated (large signal-to-noise ratio), microarray (intermediate signal to noise ratio) and proteomic data (low signal-to-noise ratio), encompassing increasing sample sizes with large numbers of features. The impact of feature selection on the performance of various cross-validation methods is highlighted. The results elucidate the 'best' resampling techniques for future research involving high dimensional data to avoid overly optimistic assessment of the performance of a model.

## 2 METHODS

In the prediction problem, one observes $n$ independent and identically distributed (i.i.d.) random variables $O_1, \ldots, O_n$ with unknown distribution $P$. Each observation in $O$ consists of an outcome $Y$ with range $\mathcal{Y}$ and an $l$-vector of measured covariates, or features, $X$ with range $\mathcal{X}$, such that $O_i = (X_i, Y_i)$, $i = 1, \ldots, n$. In microarray experiments $X$ includes gene expression measurements, while in proteomic data, it includes the intensities at the mass over charge ($m/z$) values. $X$ may also contain covariates such as a patient's age and/or histopathologic measurements. The outcome $Y$ may be a continuous measure such as months to disease or a categorical measure such as disease status.

*To whom correspondence should be addressed.

The goal in class prediction is to build a rule implementing the information from $X$ in order to predict $Y$. The intention is that by building this rule, based on the observations $O_1, \ldots, O_n$, a future unobserved outcome $Y_0$ can be predicted based on its corresponding measured features $X_0$. If the outcome is continuous, then the rule, or predictor, $\psi$ is defined as a mapping from the feature space $\mathcal{X}$ onto the real line, i.e. $\psi : \mathcal{X} \to \mathbb{R}$. Consequently, $\hat{y} = \psi(x)$ denotes the predicted outcome based on the observed $X$. Such predictors can be built via regression (linear and non-linear) or recursive binary partitioning such as classification and regression trees (CART) (Breiman *et al*., 1984). If the outcome $Y$ is categorical it assumes one of $K$ values. In this case, the rule $\psi$ partitions the feature space $\mathcal{X}$ into $K$ disjoint and exhaustive groups $G_k$, where $k = 1, \ldots, K$, such that $\hat{y} = k$ if $x \in G_k$. Standard statistical analyses include linear discriminant analysis (LDA) and diagonal discriminant classifiers (DDA), nearest neighbors (NN) and CART, as well as aggregate classifiers.

Thorough discussions of the prediction problem and available algorithms can be found in Breiman *et al*. (1984), McLachlan (1992), Ripley (1996) and Hastie *et al*. (2003).

The rule $\psi$ can be written as $\psi(\cdot|P_n)$, where $P_n$ denotes the empirical distribution of $O$ and reflects the dependence of the built rule on the observed data. Loss functions may be employed to quantify the performance of a given rule. A common loss function for a continuous outcome $Y$ is the squared error loss, $L(Y, \psi) = [Y - \psi(X)]^2$. With a categorical outcome $Y$, a popular choice is the indicator loss function, $L(Y, \psi) = I[Y \neq \psi(X)]$. A loss function could also incorporate differential misclassification costs (Breiman *et al*., 1984).

For either type of outcome, the expected loss, or *risk*, is defined as:

$$\tilde{\theta} = R(\psi, P) = E_P[L(Y, \psi)] = \int L(y, \psi(x))\, \mathrm{d}P(x, y). \tag{1}$$

The rule in Equation (1) is constructed and evaluated upon the distribution $P$, as such, $\tilde{\theta}$ is referred to as the *asymptotic risk*. However, in reality $P$ is unknown, thus, the rule based upon the observations $O_1, \ldots, O_n$ has an expected loss, or *conditional risk* (also known as the generalization error), defined as:

$$\tilde{\theta}_n = R(\psi(\cdot|P_n), P) = \int L(y, \psi(x|P_n))\, \mathrm{d}P(x, y). \tag{2}$$

There are two impetuses for evaluating the conditional risk: model selection and performance assessment. In model selection, the goal is to find the one which minimizes the conditional risk over a collection of potential models. In performance assessment, the goal is to estimate the generalization error for a given model, i.e. assess how well it predicts the outcome of an observation not included in $O$.

In an ideal setting an independent dataset would be available for the purposes of model selection and estimating the generalization error. Typically, however, one must use the observed sample $O$ for model building, selection and performance assessment. The simplest method for estimating the conditional risk is with the resubstitution or apparent error:

$$\hat{\theta}_n^{\mathrm{RS}} = R(\psi(\cdot|P_n), P_n) = \int L(y, \psi(x|P_n))\, \mathrm{d}P_n(x, y). \tag{3}$$

Here each of the $n$ observation is used for constructing, selecting and, subsequently, evaluating the prediction error of $\psi$. Consequently, the resubstitution risk estimate tends to underestimate the generalization error (Efron, 1983; McLachlan, 1992). To alleviate this biased estimation, resampling methods such as cross-validation or bootstrapping can be employed. In the next section, we describe these techniques and their implications in the framework of prediction error.

## 2.1 Resampling methods

In the absence of a large, independent test set, there are numerous techniques for assessing prediction error by implementing some form of partitioning or resampling of the original observed data $O$. Each of these techniques involves dividing the data into a *learning set* and a *test set*. For purposes of model selection, the learning set may further be divided into a training set and a validation set. We will focus solely on the partitioning of the data into learning and test sets for the express purpose of estimating the generalization error.

To enhance a general discussion of resampling methods, we define a binary random $n$-vector, $S_n \in \{0, 1\}^n$, which splits the observations into the desired subsets (Molinaro *et al*., 2004). A realization of $S_n = (S_{n,1}, \ldots, S_{n,n})$ prescribes a particular split of the entire dataset of $n$ observations into a learning set, $\{i \in \{1, \ldots, n\} : S_{n,i} = 0\}$, and a test set, $\{i \in \{1, \ldots, n\} : S_{n,i} = 1\}$. Let $p$ be the proportion of observations in the test set. The empirical distributions of the learning and test sets are denoted by $P^0_{n,S_n}$ and $P^1_{n,S_n}$, respectively. Importantly, $S_n$ is independent of the empirical distribution of the complete dataset of $n$ observations $P_n$ and the particular distribution of $S_n$ defines the type of resampling method. Given $S_n$, the performance of any given estimator $\psi(\cdot|P_n)$ can be assessed via the *resampling conditional risk* estimate

$$\hat{\theta}_{n(1-p)} = E_{S_n} \int L(o, \psi(\cdot|P^0_{n,S_n}))\mathrm{d}P^1_{n,S_n}(o), \tag{4}$$

where $S_n$ refers to binary split vectors for the entire dataset of $n$ observations and $p = \sum_i S_{i,n}/n$ is the proportion of $n$ observations in the test set.

There are several considerations when selecting a resampling method. The first is sample size $n$. For $v$-fold cross-validation and bootstrap, Dudoit and van der Laan (2003) (http://www.bepress.com/ucbbiostat/paper126) have shown that as $n \to \infty$ (and consequently $np \to \infty$) asymptotic optimality is achieved. However, no such results exist for finite samples. Other considerations are on the proportion $p$ of the observations for the test set and the number of times the estimate is calculated. We address these considerations in the following sections and refer the reader to more detailed discussions in McLachlan (1992) and Davison and Hinkley (1997).

*2.1.1 Split sample* This popular resampling method, also known as the learning-test split or holdout method (McLachlan, 1992), entails a single partition of the data into a learning set and a test set based on a predetermined $p$. For example, $p = 1/3$ allots two-thirds of the data to the learning set and one-third to the test set. The distribution of $S_n$ places mass 1/2 on two binary vectors which assign the $n$ observations to the learning and test sets. The advantage of this method is the ease of computation. Also, since the classifier is developed only once, a completely specified algorithm for classifier development need not be available; the development can be more informal and subjective. There are two potential sources of bias inherent in this method: bias introduced by each individual observation contributing only to the learning or test set; and, bias due to a small learning set, whereas both features and classifiers selected depend solely on the learning set. Because the learning set is smaller than the full data set, the test set error for a model built on the training set will tend to over-estimate the unknown generalization error for a model built on the full dataset.

*2.1.2 v-fold cross-validation* This method randomly assigns the $n$ observations to one of $v$ partitions such that the partitions are near-equal size. Subsequently, the learning set contains all but one of the partitions which is labeled the test set. The generalization error is assessed for each of the $v$ test sets and then averaged over $v$. In this method, the distribution of $S_n$ puts mass $1/v$ on the $v$ binary vectors, which assign each of the $n$ observations to one of the $v$ partitions. The proportion $p$ is approximately equal to $1/v$. Both $p$ and the number of averages can adversely or positively affect this estimate of error. For example, a larger $v$ (e.g. $v = 10$) results in a smaller proportion $p$ in the test set; thus, a higher proportion in the learning set decreases the bias. In addition, the number of averages is equivalent to $v$ and thus, may additionally decrease the bias.

*2.1.3 Leave-one-out cross-validation (LOOCV)* This is the most extreme case of $v$-fold cross-validation. In this method each observation is individually assigned to the test set, i.e. $v = n$ and $p = 1/n$ (Lachenbruch and Mickey, 1968; Geisser, 1975; Stone, 1974, 1977). The distribution of $S_n$ places mass $1/n$ on the $n$ binary vectors, which assign each of the $n$ observations to the learning and test sets. LOOCV and the corresponding $p = 1/n$ represent the best example of a bias-variance trade-off. It tends toward a small bias with elevated variance. In model selection, LOOCV has

performed poorly compared to $v$-fold cross-validation (Breiman and Spector, 1992). Due to the computational burden, LOOCV has not been a favored method for large samples, and its behavior in estimating generalization error has not been thoroughly studied.

*2.1.4 Monte Carlo cross-validation (MCCV)* MCCV randomly splits the sample into a learning and test set numerous times (e.g. 20, 50 or 1000 iterations). For each split, $np = n(1/v)$ of the observations are labeled as the test set and $n(1-p) = n(1-1/v)$ as the learning set. For example, in MCCV with $v = 10$ each of 50 iterations allot 10% of the data to the test set and 90% to the learning set. The generalization error is assessed for each of the 50 test sets and subsequently averaged over the 50 iterations. The distribution of $S_n$ puts mass $1/\binom{n}{np}$ on each of the binary vectors representing one split into a learning and test set. As the number of iterations increases the computational burden of MCCV is quite large. However, unless the iterations of random splits approaches infinity, the chance that each observation is included in a learning set and a test set (over all iterations) is small, introducing a similar bias to that of the split sample approach (i.e. when each observation is either in the learning set or test set).

*2.1.5 .632+ Bootstrap* Several variations of the bootstrap have been introduced to estimate the generalization error. The leave-one-out bootstrap ($\hat{\theta}_n^{\mathrm{BS}}$) is based on a random sample drawn with replacement from $n$ observations (Efron, 1983; Efron and Tibshirani, 1993). For each draw, the observations left out ($\approx .368n$) serve as the test set. The learning set has $\approx .632n$ unique observations which leads to an overestimation of the prediction error (i.e. a decrease in the learning set leads to an increase in the bias). To correct for this, two estimators have been suggested: the .632 bootstrap and the .632+ estimator. Both correct by adding the underestimated resubstitution error $\hat{\theta}_n^{\mathrm{RS}}$, $\omega\hat{\theta}_n^{\mathrm{BS}} + (1-\omega)\hat{\theta}_n^{\mathrm{RS}}$. For the .632 bootstrap the weight $\omega$ is constant ($\omega = .632$), whereas for the .632+ bootstrap $\omega$ is determined based on the 'no-information error rate' (Efron and Tibshirani, 1997). We focus on the latter as it is the most used in the literature and the most robust across different algorithms (Efron and Tibshirani, 1997).

## 2.2 Algorithms

Predictions of outcomes based on the observed $X$ can employ parametric or non-parametric algorithms. If the outcome is continuous, predictors can be built using regression models or recursive binary partitioning like CART. If the outcome is categorical, algorithms which partition the feature space $X$ into disjoint and exhaustive groups are used. In this manuscript, we limit our discussion to the classification of binary outcomes, i.e. $Y = 0$ or $Y = 1$, and thus, evaluate methods for the estimation of prediction error in the context of the following classification algorithms.

We calculate the LDA with the `lda` function in the `MASS` library of the statistical package `R` (Venables and Ripley, 1994; Ihaka and Gentlemen, 1996). We use the function `dlda` in the library `supclust` in `R` to implement DDA (Dettling and Maechler, 2005, http://lib.stat.cmu.edu/R/CRAN/src/contrib/Descriptions/supclust.html). The library `supclust` also houses the function `nnr` for NN. CART classification is obtained using the library and function `rpart` in `R` (Breiman *et al.*, 1984; Therneau and Atkinson, 1997).

# 3 ANALYSIS

The goal of this analysis is to ascertain differences between resampling methods in the estimation of generalization error (presently, limited to the classification problem) in the presence of feature selection. We evaluate the influence of sample size, parametric to non-parametric classification methods, and large feature spaces on each resampling method's ability to estimate the resampling conditional risk $\hat{\theta}_{n(1-p)}$ [Equation (4)] compared to that of the 'true' conditional risk $\tilde{\theta}_n$ [Equation (2)]. As such, a range of sample sizes ($n = 40$, 80 and 120), classification algorithms (LDA, DDA, NN and CART), and data sets (simulated, microarray and proteomic; see Sections 3.1–3.3) are utilized. Prior to discussing results, the general strategy for estimating the risks is explained followed by the specifics of each dataset.

Each dataset consists of $N$ observations with $N_1$ cases and $N_0$ controls and $l$ measured features. For $r = 1, \ldots, R$ repetitions, a random sample of size $n$ stratified by case/control status is selected from $N$, such that the number of cases in the subsample ($n/2$) equals the number of controls. The stratification allows for equal representation of both, cases and controls, such that classification algorithms relying on majority consensus are not biased toward either (Quackenbush, 2004). This random sample, or subsample, plays two roles. First, it serves as a sample from which the resampling conditional risk $\hat{\theta}_{n(1-p)}$ can be estimated. This is accomplished by splitting the subsample into a learning and test set corresponding to each of the resampling methods. For each $r$, an estimate of $\hat{\theta}_{n(1-p)}$ is obtained for each resampling method with all four algorithms. In reality, the distribution $P$ of the observed data $O$ is unknown and thus, so is the 'true' conditional risk. In order to estimate $\tilde{\theta}_n$ in Equation (2) we will use the complete observed data. As such, the subsample's second role is to serve as the learning set and the remaining $N - n$ observations as the test set for an approximation of the conditional risk $\tilde{\theta}_n$.

Given the high-dimensional structure of each data set (i.e. large $l$), feature selection is an important task administered before running any of the algorithms. Feature selection must occur based on the learning set within each resampling, otherwise additional bias is introduced (Simon *et al.*, 2003). This correct approach to feature selection within cross-validation has been referred to as honest or complete (Quackenbush, 2004). There are many methods available for feature selection; here $t$-tests are used. Initially components of $X$ with the largest 10 absolute value $t$-test statistics are considered. Subsequently, the largest 20 are discussed.

All simulations and analyses were implemented in `R` (Ihaka and Gentlemen, 1996).

## 3.1 Simulated data

The simulated datasets are generated as described in Bura and Pfeiffer (2003). Each dataset contains $N = 300$ observations with 750 covariates, representing patients and genes, respectively. Half of the observations (i.e. 150) are labeled controls ($Y = 0$) and half cases ($Y = 1$). Of the 750 genes, 8 are associated with disease and the others are non-predictive. The controls are simulated from a multivariate normal distribution with a mean of 0 and covariance matrix $\Sigma$. The cases have 99% non-differentially expressed genes which are generated from the same $N(0, \Sigma)$ as the controls. The 1% of the genes that are differentially expressed are generated from a mixture of two multivariate normals with means $\mu_1$ and $\mu_2$ and covariance structure $\Sigma$. The mixing probability is 0.5. The covariance matrix $\Sigma = (\sigma_{ij})$ has a block structure with $\sigma_{ij} = 0.2$ for $|j - i| \leq 5$ and zero otherwise. Estimates of $\hat{\theta}_{n(1-p)}$ and $\tilde{\theta}_n$ are based on learning samples of size 40, 80 and 120 and test sets of size 260, 220 and 180, respectively.

## 3.2 Lymphoma and lung datasets

The microarray datasets are both publicly available. The first focuses on diffuse large-B-cell lymphoma (Rosenwald *et al.*, 2002). In this study there are 7399 genes on the microarray and 240 patients. For the purposes of this analysis, the outcome-variable represents the lymphoma subtype: activated B-cell for $Y = 0$ and germinal-center

B-cell for $Y = 1$. This is an example of a moderate signal-to-noise ratio dataset, as the subgroups do not separate perfectly based on the microarray observations (Wright *et al*., 2003). Estimates of $\hat{\theta}_{n(1-p)}$ and $\tilde{\theta}_n$ are based on learning samples of size 40, 80 and 120 and test sets of size 200, 160 and 120, respectively. The second study uses oligonucleotide microarrays to measure 12 601 transcript sequences for 186 lung tumor samples (Bhattacharjee *et al*., 2001). For our analysis, the outcome represents the 139 adenocarcinomas as $Y = 0$ and the remaining 47 tumors as $Y = 1$.

### 3.3 Proteomic ovarian dataset

The proteomic dataset consists of 164 SELDI-TOF measurements from NCI/Mayo Clinic serum samples. These data are part of a study designed to validate previously identified proteomic markers for ovarian cancer. The readings are from fraction 4, IMAC30 ProteinChip arrays, read at high and low energy settings in a PCS4000 ProteinChip Reader (Ciphergen Biosystems, Inc., Fremont, CA). The spectra were externally calibrated for mass, internally normalized for intensity using total ion current, and baseline subtracted. Peaks were manually selected and the intensity recorded.

Of the $n = 164$ observations, 45 are ovarian cancer cases and 119 controls. Estimates of $\hat{\theta}_{n(1-p)}$ and $\tilde{\theta}_n$ are based on learning samples of size 40 and 80 and test sets of size 144 and 104, respectively. Given the nature of proteomic data as well as the naive algorithms implemented, this will serve as a low signal-to-noise example.

### 3.4 Results

To compare the resampling methods in Section 2.1, conditional risk estimates for each method are calculated and compared to each other and the truth (i.e. the conditional risk). This evaluation is based on the mean squared error (MSE) and bias, calculated as follows:

$$\text{MSE} = \frac{1}{r}\sum_{r=1}^{R}(\hat{\theta}_{n,r} - \tilde{\theta}_{n,r})^2$$

$$\text{Bias} = \frac{1}{r}\sum_{r=1}^{R}(\hat{\theta}_{n,r} - \tilde{\theta}_{n,r}),$$

where $\hat{\theta}_{n,r}$ is the resampling conditional risk and $\tilde{\theta}_{n,r}$ is the conditional risk for the $r$-th repetition. In all results the total number of repetitions is set at 100, i.e. $R = 100$.

There were several attempts to examine the effect of varying $p$ on those resampling methods which allow user-defined test set proportions (i.e. $v$-fold cross-validation, MCCV and split sample). For $v$-fold cross-validation, 2-, 5- and 10-fold were explored. In MCCV, both $p$ and the number of MCCV repetitions affect the estimation, thus, test set proportions of $p = 0.5$, $p = 0.2$, $p = 0.1$ as well as repetitions of 20, 50 and 1000 were run. In split-sample estimation test set proportions of both $p = 1/3$ and $p = 1/2$ were examined to assess the bias/variance trade-off.

Due to space limitations, all results are discussed but only a limited number of tables can be displayed. The interested reader is referred to Molinaro *et al*. (2005) (http://linus.nci.nih.gov/brb/TechReport.htm) for a comprehensive compilation of results. The MCCV results are not included below, as the only noticeable improvement over $v$-fold CV is a slight decrease in variance. Additionally, the advantage of increasing the MCCV iterations from 20 to 50 to 1000 is minimal.

*Simulation study results* For $n = 40$, LOOCV and 10-fold CV have the smallest MSE and bias, followed by 5-fold CV and then .632+ (Table 1). The largest MSE and bias occur with 2-fold CV and split sample with $p = 1/2$. For $n = 80$ and $n = 120$, the differences among these methods diminish. For $n = 40$ and $n = 80$, .632+ has the smallest SD, followed by 10-fold CV, LOOCV and 5-fold CV. The only exception is for LDA and NN at $n = 80$, when LOOCV and 10-fold CV have the smallest. At $n = 120$, the differences among these methods diminish.

**Table 1.** Prediction error estimates

| Estimator | $p$ | Algorithm | Estimate | SD | Bias | MSE |
|---|---|---|---|---|---|---|
| $\tilde{\theta}_n$ | 0.87 | LDA | 0.078 | 0.093 | | |
| | | DDA | 0.160 | 0.086 | | |
| | | NN | 0.042 | 0.084 | | |
| | | CART | 0.121 | 0.133 | | |
| $v$-fold CV | 0.5 | LDA | 0.357 | 0.126 | 0.279 | 0.097 |
| | | DDA | 0.342 | 0.106 | 0.182 | 0.052 |
| | | NN | 0.277 | 0.135 | 0.235 | 0.077 |
| | | CART | 0.430 | 0.121 | 0.309 | 0.134 |
| | 0.2 | LDA | 0.161 | 0.127 | 0.083 | 0.017 |
| | | DDA | 0.208 | 0.086 | 0.048 | 0.012 |
| | | NN | 0.108 | 0.102 | 0.066 | 0.011 |
| | | CART | 0.284 | 0.117 | 0.163 | 0.055 |
| | 0.1 | LDA | 0.118 | 0.120 | 0.040 | **0.008** |
| | | DDA | 0.177 | 0.087 | 0.017 | **0.007** |
| | | NN | 0.078 | 0.102 | 0.036 | **0.005** |
| | | CART | 0.189 | 0.104 | 0.068 | 0.024 |
| LOOCV | 0.025 | LDA | 0.092 | 0.115 | **0.014** | **0.008** |
| | | DDA | 0.164 | 0.096 | **0.004** | **0.007** |
| | | NN | 0.058 | 0.103 | **0.016** | **0.005** |
| | | CART | 0.146 | 0.125 | **0.025** | **0.018** |
| Split | 0.333 | LDA | 0.205 | 0.184 | 0.127 | 0.053 |
| | | DDA | 0.243 | 0.138 | 0.083 | 0.034 |
| | | NN | 0.145 | 0.169 | 0.103 | 0.044 |
| | | CART | 0.371 | 0.174 | 0.25 | 0.121 |
| | 0.5 | LDA | 0.348 | 0.185 | 0.270 | 0.113 |
| | | DDA | 0.344 | 0.139 | 0.184 | 0.062 |
| | | NN | 0.265 | 0.177 | 0.223 | 0.086 |
| | | CART | 0.438 | 0.155 | 0.317 | 0.147 |
| .632+ 50 repetitions | $\approx$.368 | LDA | 0.274 | **0.084** | 0.196 | 0.047 |
| | | DDA | 0.286 | **0.074** | 0.126 | 0.028 |
| | | NN | 0.200 | **0.070** | 0.158 | 0.032 |
| | | CART | 0.387 | **0.080** | 0.266 | 0.100 |

The estimate $\hat{\theta}_n$ (column 4) and SD (column 5) based on learning sample of size 40. The estimate $\tilde{\theta}_n$ (rows 1–4) and SD based on the remaining 260 observations. Bias (column 6) and MSE (column 7) reported for each resampling technique (column 1) and algorithm (column 3). The ten features with largest $t$-statistics used in algorithms. Minimums in bold.

*Lymphoma and lung study results* In the lymphoma study, for $n = 40$, 80 and 120, .632+, LOOCV, 5- and 10-fold CV have the smallest MSE and bias. The two split-samples and 2-fold CV have the largest MSE and bias. Similar to the simulation study, .632+ has the smallest SD across the algorithms and sample sizes, while both split samples do by far the worst. Partial results are shown in Table 2. The results from the lung study are very similar and thoroughly discussed in Molinaro *et al*. (2005). (http://linus.nci.nih.gov/brb/TechReport.htm).

*Ovarian study results* For $n = 40$ to $n = 80$, LOOCV and .632+ have the smallest MSE, followed by 5- and 10-fold CV. As for bias,

**Table 2.** Lymphoma study results

| Resampling method | $n = 40$ | | | $n = 80$ | | | $n = 120$ | | |
|---|---|---|---|---|---|---|---|---|---|
| | SD | Bias | MSE | SD | Bias | MSE | SD | Bias | MSE |
| 2-fold CV | 0.085 | 0.038 | 0.01 | 0.043 | 0.002 | 0.004 | 0.031 | 0.0 | 0.003 |
| 5-fold CV | 0.07 | 0.004 | 0.007 | 0.045 | −0.008 | 0.005 | 0.032 | −0.006 | 0.003 |
| 10-fold CV | 0.063 | −0.007 | 0.006 | 0.036 | −0.009 | 0.003 | 0.031 | −0.006 | 0.003 |
| LOOCV | 0.072 | −0.019 | 0.008 | 0.04 | −0.013 | 0.004 | 0.033 | −0.004 | 0.003 |
| Split 1/3 | 0.119 | 0.001 | 0.017 | 0.071 | 0.0 | 0.007 | 0.059 | −0.004 | 0.005 |
| Split 1/2 | 0.117 | 0.037 | 0.018 | 0.058 | 0.001 | 0.005 | 0.046 | −0.001 | 0.004 |
| .632+ | 0.049 | −0.006 | 0.004 | 0.025 | −0.02 | 0.003 | 0.018 | −0.015 | 0.002 |

Comparison of resampling method's MSE, bias and SD. Results shown are for the DDA algorithm using the top 10 genes as ranked by $t$-tests.

**Table 3.** Ovarian study results

| Resampling method | $n = 40$ | | | $n = 80$ | | |
|---|---|---|---|---|---|---|
| | SD | Bias | MSE | SD | Bias | MSE |
| 2-fold CV | 0.098 | 0.026 | 0.015 | 0.05 | 0.004 | 0.007 |
| 5-fold CV | 0.082 | 0 0 | 0.012 | 0.039 | −0.005 | 0.006 |
| 10-fold CV | 0.082 | −0.01 | 0.011 | 0.036 | −0.005 | 0.005 |
| LOOCV | 0.079 | −0.004 | 0.011 | 0.037 | −0.004 | 0.006 |
| Split 1/3 | 0.133 | −0.002 | 0.022 | 0.075 | −0.009 | 0.009 |
| Split 1/2 | 0.113 | 0.027 | 0.018 | 0.071 | 0.013 | 0.01 |
| .632+ | 0.075 | −0.006 | 0.011 | 0.028 | −0.014 | 0.005 |

Comparison of resampling method's MSE, bias and SD. Results shown are for the DDA algorithm using the top 10 peaks as ranked by $t$-tests.

10-fold CV, .632+ and LOOCV vie for the smallest. The largest MSE and bias occur with the split samples and 2-fold CV. Again .632+ has the smallest SD across algorithms and sample sizes; however, the discrepancy is much smaller than in the other two studies. The split samples have the largest SDs. Partial results are shown in Table 3.

All analyses were repeated, selecting the 20 features having the largest $t$-test statistics. The ranking of the resampling methods remained the same (Supplementary material).

*Repeated resampling* We examined the effect of repeated resampling on 2-, 5- and 10-fold CV and split sample with $p = 1/3$, for the three samples sizes and four algorithms. Each was repeated 10 and 30 times. Interestingly, there was minimal improvement when increased from 10 to 30 repeats. However, when increasing repeats from 1 to 10 (or 30), all SDs decreased (up to 50%). The MSE either decreased (up to 35%) or stayed similar, which was also true for the bias except in split sample for $n = 40$ and 2-fold CV for $n = 40$ and $n = 80$ (Supplementary material).

*Dimensionality of feature space* In the simulations of Efron and Tibshirani (1997), .632+ outperformed LOOCV and 10-fold CV. For example, in their experiment 22, with 10 variables and 36 patients, the MSE was .040 for .632+ and .058 for LOOCV. However, in our simulations with $n = 40$ (Table 1) .632+ does not fare so well, particularly with regard to bias. To investigate the differences between our simulations and those in Efron and Tibshirani, we decreased the dimensions of the feature space to a total of 10 variables instead of 750. The results are shown in Table 4 for the sample size of 40. With low dimension the large bias of the bootstrap is substantially reduced, and the .632+ does as well or better than LOOCV and 10-fold CV.

**Table 4.** Prediction error estimates without feature selection

| Estimator | $p$ | Algorithm | Estimation | SD | Bias | MSE |
|---|---|---|---|---|---|---|
| $\tilde{\theta}_n$ | 0.87 | LDA | 0.026 | 0.028 | | |
| | | DDA | 0.073 | 0.058 | | |
| | | NN | 0.010 | 0.017 | | |
| | | CART | 0.099 | 0.092 | | |
| $v$-fold CV | 0.5 | LDA | 0.067 | 0.060 | 0.041 | 0.005 |
| | | DDA | 0.106 | 0.079 | 0.033 | 0.009 |
| | | NN | 0.011 | 0.025 | 0.001 | **0** |
| | | CART | 0.304 | 0.088 | 0.205 | 0.063 |
| | 0.2 | LDA | 0.034 | 0.045 | 0.008 | 0.002 |
| | | DDA | 0.085 | 0.049 | 0.012 | 0.003 |
| | | NN | 0.011 | 0.024 | 0.001 | **0** |
| | | CART | 0.158 | 0.072 | 0.059 | 0.012 |
| | 0.1 | LDA | 0.032 | 0.041 | 0.006 | **0.001** |
| | | DDA | 0.074 | 0.048 | **0.001** | **0.002** |
| | | NN | 0.010 | 0.021 | **0** | **0** |
| | | CART | 0.118 | 0.063 | 0.019 | **0.006** |
| LOOCV | 0.025 | LDA | 0.028 | 0.040 | **0.002** | **0.001** |
| | | DDA | 0.072 | 0.049 | **−0.001** | **0.002** |
| | | NN | 0.010 | 0.022 | **0** | **0** |
| | | CART | 0.110 | 0.075 | **0.011** | **0.006** |
| Split | 0.333 | LDA | 0.046 | 0.076 | 0.020 | 0.005 |
| | | DDA | 0.066 | 0.085 | −0.007 | 0.008 |
| | | NN | 0.007 | 0.029 | −0.003 | 0.001 |
| | | CART | 0.265 | 0.116 | 0.166 | 0.047 |
| | 0.5 | LDA | 0.073 | 0.078 | 0.047 | 0.007 |
| | | DDA | 0.093 | 0.099 | 0.020 | 0.013 |
| | | NN | 0.010 | 0.028 | **0** | 0.001 |
| | | CART | 0.308 | 0.114 | 0.209 | 0.071 |
| .632+ | ≈.368 | LDA | 0.037 | **0.036** | 0.011 | **0.001** |
| 50 repetitions | | DDA | 0.085 | **0.036** | 0.012 | 0.003 |
| | | NN | 0.008 | **0.016** | −0.002 | **0** |
| | | CART | 0.160 | **0.034** | 0.061 | 0.010 |

To assess the effect of no feature selection on resampling methods estimation, only 10 features were simulated and all 10 used in estimation. Results based on a learning sample of 40 and a test sample of 260. Absolute minimums in bold.

**Table 5.** Resampling with and without replacement

| $n$ | Algorithm | Leave-one-out bootstrap | | | | 3-fold MCCV | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Estimation | SD | Bias | MSE | Estimation | SD | Bias | MSE |
| | LDA | 0.331 | 0.075 | 0.252 | 0.072 | 0.242 | 0.101 | 0.164 | 0.035 |
| | DDA | 0.337 | 0.075 | 0.177 | 0.044 | 0.270 | 0.072 | 0.110 | 0.022 |
| $n = 40$ | NN | 0.259 | 0.072 | 0.217 | 0.055 | 0.167 | 0.083 | 0.125 | 0.022 |
| | CART | 0.414 | 0.065 | 0.296 | 0.114 | 0.377 | 0.085 | 0.256 | 0.094 |
| | LDA | 0.07 | 0.063 | 0.043 | 0.004 | 0.044 | 0.053 | 0.017 | 0.002 |
| | DDA | 0.146 | 0.058 | 0.074 | 0.008 | 0.104 | 0.058 | 0.033 | 0.003 |
| $n = 80$ | NN | 0.046 | 0.056 | 0.036 | 0.003 | 0.022 | 0.043 | 0.012 | 0.001 |
| | CART | 0.098 | 0.047 | 0.057 | 0.006 | 0.062 | 0.039 | 0.020 | 0.002 |
| | LDA | 0.032 | 0.033 | 0.011 | 0.001 | 0.026 | 0.026 | 0.005 | 0 |
| | DDA | 0.088 | 0.045 | 0.036 | 0.002 | 0.068 | 0.043 | 0.016 | 0.001 |
| $n = 120$ | NN | 0.016 | 0.030 | 0.007 | 0 | 0.012 | 0.023 | 0.003 | 0 |
| | CART | 0.048 | 0.025 | 0.022 | 0.001 | 0.038 | 0.022 | 0.012 | 0.001 |

The leave-one-out bootstrap and 3-fold MCCV estimate, SD, bias, and MSE, over 3 samples sizes and 4 algorithms. Feature selection was used to select the top 10 ranked features by $t$-tests.

*Resampling with and without replacement* To understand the ramification of resampling with replacement as it pertains to the bootstrap estimates, we compared the leave-one-out bootstrap estimate (Section 2.1.5) to the 3-fold MCCV. The 3-fold MCCV randomly selects .666$n$ for the learning set and .333$n$ for the test set. This is repeated numerous times and the estimates averaged. Therefore the 3-fold MCCV is equivalent to the leave-one-out bootstrap, except it employs resampling without replacement. Table 5 displays the simulation study results for the two estimates using 50 iterations for both. Interestingly, the bias and MSE for the leave-one-out bootstrap are roughly double that of 3-fold MCCV. The only two distinct differences between the two methods are the replicate copies in the learning set, inherent in the bootstrap estimate, and the fact that **on average** .632$n$ unique observations are in the learning sample for the leave-one-out bootstrap, whereas there are always .666$n$ in the learning sample for the 3-fold MCCV. Both these factors may contribute to the increase in bias and MSE.

## 4 DISCUSSION

Estimation of prediction error when confronted with a multitude of covariates and small sample sizes is a relatively new problem. Feature selection, sample size and signal-to-noise ratio are important influences on the relative performance of resampling methods. We have evaluated resampling methods for use in high dimensional classification problems using a range of sample sizes, algorithms and signals. Some general conclusions may be summarized as follows:

(1) **With small sample sizes, the split sample method and 2-fold CV perform very poorly**. This poor performance is primarily due to a large positive bias resulting from the use of a reduced training set size, which severely impairs its ability to effectively select features and fit a model. The large bias contributes to a large MSE.

(2) **LOOCV generally performs very well with regard to MSE and bias**. The only exception is when an unstable classifier (e.g. CART) is used in the presence of a weak signal. In this setting, the larger MSE is attributed to LOOCV's increased variance.

(3) **10-fold CV prediction error estimates approximate those of LOOCV in almost all settings**. For computationally burdensome analyses, 10-fold CV may be preferable to LOOCV. Additionally, in the simulated data, repeated resamplings (the average of 10 repeats) reduce the MSE, bias, and variance of 10-fold CV.

(4) **The .632+ prediction error estimate performs best with moderate to weak signal-to-noise ratios**. Previous studies have found the bootstrap variants superior to LOOCV and $v$-fold CV; however, these studies did not include feature selection. As seen in Table 1, honest resampling in small samples with strong signal suggest that LOOCV and 10-fold CV are in fact better than the .632+ bootstrap. This discrepancy fades when feature selection is discarded (Table 4) and when the signal decreases, as seen in the lymphoma and ovarian datasets (Tables 2 and 3). Additional glimpses into the bootstrap estimate (Table 5) indicate that the sampling with replacement increases the MSE and bias substantially over 3-fold MCCV (i.e. resampling without replacement).

(5) MCCV does not decrease the MSE or bias enough to warrant its use over $v$-fold CV.

(6) As the sample size grows, the differences among the resampling methods decrease. Additionally, as the signal decreases from strong in the simulated data to rather weak in the ovarian data the discrepancies between the methods diminish.

In future work we will compare the resampling methods for continuous outcomes and continue to explore the behavior of the bootstrap estimates. Also, the effect of feature selection method may play an important role in prediction and needs further investigation.

## ACKNOWLEDGEMENTS

A.M.M. was supported by the Cancer Prevention Fellowship Program, DCP/NCI/NIH. The authors thank Mark J. van der Laan for fruitful discussions.

*Conflict of Interest:* none declared.

## REFERENCES


Bhattacharjee,A. *et al*. (2001) Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. *Proc. Natl Acad. Sci. USA*, **98**, 13790–13795.

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

Breiman,L. and Spector,P. (1992) Submodel selection and evaluation in regression. The X-random case. *Int. Stat. Rev.*, **60**, 291–319.

Breiman,L., Friedman,J.H., Olshen,R.A. and Stone,C.J. (1984) *Classification and Regression Trees*. Wadsworth and Brooks/Cole, Monterey, CA.

Bura,E. and Pfeiffer,R.M. (2003) Graphical methods for class prediction using dimension reduction techniques on DNA microarray data. *Bioinformatics*, **19**, 1252–1258.

Burman,P. (1989) A comparative study of ordinary cross-validation, *v*-fold cross-validation and the repeated learning-testing methods. *Biometrika*, **76**, 503–514.

Davison,A.C. and Hinkley,D.V. (1997) *Bootstrap Methods and Their Application*. Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge University Press, Cambridge, UK.

Dettling,M. and Maechler,M. (2005) Software R Contributed Package: Supclust: Supervised Clustering of Genes. (http://cran.r-project.org) version 1.0-5.

Dudoit,S. and van der Laan,M.J. (2003) Asymptotics of cross-validated risk estimation in model selection and performance assessment. *Technical Report 126*, U.C. Berkeley Division of Biostatistics Working Paper Series. http://www.bepress.com/ucbbiostat/paper126.

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

Efron,B. (2004) The estimation of prediction error: covariance penalties and cross-validation. *J. Am. Stat. Assoc.*, **99**, 619–642.

Efron,B. and Tibshirani,R.J. (1993) *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability. Chapman and Hall, Vol. 57, NY.

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

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

Hastie,T., Tibshirani,R. and Friedman,J. (2003) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, 1st edn, 3rd print. Springer-Verlag, New York.

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

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

McLachlan,G.J. (1992) *Discriminant Analysis and Statistical Pattern Recognition*. John Wiley & Sons, Inc, New York.

Molinaro,A.M. *et al*. (2004) Tree-based multivariate regression and density estimation with right-censored data. *J. Multivar. Anal.*, **90**, 154–177.

Molinaro,A.M., Simon,R. and Pfeiffer, R.M. (2005) Prediction error estimation: a comparison of resampling methods. *Technical Report Number 30*, Biometrics Research Branch, Division of Cancer Treatment and Diagnosis, NCI.

Quackenbush,J. (2004) Meeting the challenges of functional genomics: from the laboratory to the clinic. *Preclinica*, **2**, 313–316.

Ransohoff,D.F. (2004) Rules of evidence for cancer molecular marker discovery and validation. *Nature Reviews/Cancer*, **4**, 309–313.

Ripley,B.D. (1996) *Pattern Recognition and Neural Networks*. Cambridge University Press, New York.

Rosenwald,A. *et al*. (2002) The use of molecular profiling to predict survival after chemotherapy for diffuse large-B-cell lymphoma. *New Engl. J. Med.*, **346**, 1937–1946.

Simon,R. *et al*. (2003) Pitfalls in the use of DNA microarray data for diagnostic and prognostic classification. *J. Natl Cancer Inst.*, **95**, 14–18.

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

Stone,M. (1977) Asymptotics for and against cross-validation. *Biometrika*, **64**, 29–35.

Therneau,T. and Atkinson,E. (1997) An introduction to recursive partitioning using the RPART routine. *Technical Report 61*, Section of Biostatistics, Mayo Clinic, Rochester.

Venables,W.N. and Ripley,B.D. (1994) *Modern Applied Statistics with S-PLUS*. Springer, New York.

Wright,G. *et al*. (2003) A gene expression-based method to diagnose clinically distinct subgroups of diffuse large B cell lymphoma. *Proc. Natl Acad. Sci. USA*, **100**, 9991–9996.