Machine Learning, 48, 287–297, 2002

Estimating Generalization Error on Two-Class Datasets Using Out-of-Bag Estimates*

TOM BYLANDER bylander@cs.utsa.edu
Division of Computer Science, University of Texas at San Antonio, San Antonio, Texas 78249-0667, USA

Editors: Yoshua Bengio and Dale Schuurmans

Abstract. For two-class datasets, we provide a method for estimating the generalization error of a bag using out-of-bag estimates. In bagging, each predictor (single hypothesis) is learned from a bootstrap sample of the training examples; the output of a bag (a set of predictors) on an example is determined by voting. The out-of-bag estimate is based on recording the votes of each predictor on those training examples omitted from its bootstrap sample. Because no additional predictors are generated, the out-of-bag estimate requires considerably less time than 10-fold cross-validation. We address the question of how to use the out-of-bag estimate to estimate generalization error on two-class datasets. Our experiments on several datasets show that the out-of-bag estimate and 10-fold cross-validation have similar performance, but are both biased. We can eliminate most of the bias in the out-of-bag estimate and increase accuracy by incorporating a correction based on the distribution of the out-of-bag votes.

Keywords: bagging, cross-validation, generalization error

1. Introduction

Supervised learning involves finding a hypothesis to correctly classify examples in a domain. If, for example, we wanted to classify mushrooms as edible or poisonous based on relevant characteristics such as color, smell, habitat, etc., we could learn a hypothesis by using mushrooms whose characteristics and classifications are known.

Much work has been done in supervised learning in developing learning algorithms for decision trees, neural networks, Bayesian networks, and other hypothesis spaces. As an improvement on these learning algorithms, work has recently been done using algorithms that combine several "single hypotheses" (called "predictors" from this point onward) into one "aggregate hypothesis." One such algorithm is bagging (bootstrap aggregating) (Breiman, 1996a). Bagging involves repeated sampling with replacement to form several bootstrap training sets from the original dataset. Bagging should not be viewed as a competitor to other aggregation algorithms (such as boosting) because bagging can use these learning algorithms to generate predictors.

Over many types of predictor algorithms, bagging has been shown to improve on the accuracy of a single predictor (Breiman, 1996a; Dietterich, 2000; Freund & Schapire, 1996; Maclin & Opitz, 1997; Quinlan, 1996). An important issue is determining the generalization error of a bag (a bagging aggregate hypothesis). Usually, generalization error is estimated

*This is a revision of a paper that appeared in AAAI-99.

by k-fold cross-validation over the dataset (Michie, Spiegelhalter, & Taylor, 1994; Weiss & Kulikowski, 1991).

There are two potential problems with the cross-validation estimate. One is the additional computation time. If there are B predictors in the bag, then $10B$ additional predictors must be generated for 10-fold cross-validation. This becomes a serious issue if significant time is needed to generate each predictor, e.g., as in neural networks.

The other is that the cross-validation estimate does not directly evaluate the aggregate hypothesis. None of the $10B$ predictors generated during 10-fold cross-validation become part of the bag (except by coincidence). It is an assumption that the performance of the hypotheses learned from the cross-validation folds will be similar to the performance of the hypothesis learned using the whole dataset (Kearns & Ron, 1997). In fact, previous research (Kohavi, 1995) has shown that 10-fold cross-validation tends to have a pessimistic bias, i.e., the estimated error rate tends to have a higher expected value than the true error rate.[1]

One solution is to use the predictors in the bag to estimate generalization error. Each predictor is generated from a bootstrap sample, which typically omits about $1/e \approx 37\%$ of the examples. The *out-of-bag estimate* (Breiman, 1996b) records the votes of each predictor over the examples omitted from its corresponding bootstrap sample. The aggregation of the votes followed by plurality voting for each example results in an estimate of generalization error. Our experiments show that the out-of-bag estimate slightly overestimates generalization error on average.

We can improve the out-of-bag estimate by incorporating a correction. If there are B predictors in the bag, then there are B votes for each test example compared to about $0.37B$ out-of-bag votes on average for each training example. We propose a model of this process, and a correction to the out-of-bag estimate that takes this model into account.

Our model is based on the voting patterns in the test examples, where a voting pattern is specified by the number of votes for each class, e.g., 29 votes for class A and 21 votes for class B. For a given test example, we can simulate out-of-bag voting by drawing a subsample of the votes on the test examples, i.e., each vote is selected with probability $1/e$. We assume that the accuracy of the simulation on the test examples compared to the out-of-bag estimate on the training examples gives us a "gold standard" for the generalization estimates.

Our correction tries to reverse this process. It uses the out-of-bag voting patterns on the training examples to estimate the distribution of B-vote patterns. Based on this estimated distribution, we compute the expected value of the difference between the out-of-bag estimate and B-vote voting.

This method for determining a generalization estimate differs from previous research. Both Wolpert and Macready (1999) and Tibshirani (1996) compute estimates based on a bias-variance decomposition, while we attempt to develop a more direct estimate. Wolpert and Macready's technique is developed for using bagging on continuous outputs, where the bagging output is the average of the predictors. This does not apply to two-class datasets. Tibshirani analyzes the two-class case, but is not so much concerned with a final estimate of generalization error, but in estimating bias and variance in order to better understand the behavior of the learning algorithm.

We performed experiments on 10 two-class datasets. We used ID3 (Quinlan, 1986) and C4.5 (Quinlan, 1993) to generate predictors. Generalization error is represented by the empirical error of the bag on a separate test set. The out-of-bag estimate slightly overestimated generalization error on average. 10-fold cross-validation had similar behavior, which is consistent with previous research (Kohavi, 1995). Our out-of-bag correction was less biased and produced more accurate estimates than both the out-of-bag estimate and 10-fold cross-validation.

The remainder of this paper is organized as follows. First, we describe the experimental procedure. Next, we provide the results of the experiments and their implications. Finally, we conclude with a summary and future research issues.

2. Experimental procedure

We selected a number of two-class datasets from the UCI repository and the C4.5 distribution. Several of these datasets were used extensively to develop the generalization error estimates. The other datasets (see the Appendix) were used for the experiments presented in this paper.

We used two learning algorithms. One algorithm was C4.5 using default parameters (Quinlan, 1993). We also used the ID3 decision-tree learning algorithm with no pruning (Quinlan, 1986). In our version of ID3, missing values are handled by creating an extra branch from each internal node to represent the case of a missing value. If there are no examples for a leaf node, it is given a classification equal to the most common class of its parent.

For this paper, the following procedure for experimenting with the bagging method was used:

1. The data set is randomly divided in half to create a training set S and a test set T.
2. A bootstrap sample S_1^* is selected from S and a predictor is created from S_1^* using a learning algorithm. This is repeated B times to create B predictors, $h_1, \ldots, h_B$, from the B bootstrap samples $S_1^*, \ldots, S_B^*$.
3. The out-of-bag estimate is determined from the training set S by allowing each predictor h_i to vote only on the examples $S - S_i^*$, i.e., the training examples omitted from the ith bootstrap sample. Then the predicted class of each example is determined by a vote with ties broken in favor of the most common class in S. On average, about 37% of the examples are excluded from each bootstrap sample, so on average, about 37% of the predictors vote on each training example.
4. Test error is determined from the test set T by a vote on each example over the B predictors. Ties are broken in favor of the most common class in S. Test error is considered to be an accurate estimate of generalization error.[2]
5. The above steps 1–4 are repeated 1000 times for each data set, learning algorithm, and value for B (we used $B = 50$ and $B = 100$). Averages and standard deviations for the out-of-bag estimate, test error, and the paired difference were computed. Any substantial difference in the averages ought to become statistically significant after 1000 trials.

2.1. Other generalization error estimates

Besides the out-of-bag estimate, we also evaluated 10-fold cross-validation and two different corrections to the out-of-bag estimate.

2.1.1. 10-Fold cross-validation. For $B = 50$, we computed a 10-fold cross-validation estimate of generalization error. Cross validation has been widely accepted as a reliable method for calculating generalization accuracy (Michie, Spiegelhalter, & Taylor, 1994; Weiss & Kulikowski, 1991), and experiments have shown that cross validation is less biased than bootstrap sampling (Efron & Tibshirani, 1993). However, there is some evidence that 10-fold cross-validation can have high variance (Dietterich, 2000).

In order to compute the cross-validation estimate, a step is inserted between steps 4 and 5 in the procedure described above. In this new step, the training set S is partitioned into 10 cross-validation sets or folds of nearly equal size. Then for each cross-validation fold F_i, the examples $S - F_i$ are used with bagging to form B predictors. The resulting bag is used to classify the examples in F_i and produce an error measure. The cross-validation estimate is the average error over the 10 iterations.

2.1.2. Test error correction. Our "test error correction" is a model of out-of-bag voting, and is used as a gold standard for other estimates to attain. It is not a true generalization estimate because it estimates the out-of-bag estimate from the test examples.

In the out-of-bag estimate, there are about $0.37B$ out-of-bag votes on average for each training example. The test error is determined from B votes for each test example, so it might be expected that the out-of-bag voting would be inaccurate.

Our approach is to model out-of-bag voting as if we were taking a subsample of the B votes on that example. We call this "out-of-bag sampling." We can simulate out-of-bag sampling by choosing each vote with probability $1/e$. This simulation is expected to be a good model of the out-of-bag estimate because test examples and training examples should be interchangeable as far as out-of-bag voting is concerned.

We increment the test error correction if the simulated out-of-bag sample voted for a different class from the B votes. In our experiments, the total test error correction is compared to the out-of-bag estimate on the training examples.

2.1.3. Out-of-bag correction. The test error correction estimates out-of-bag voting patterns from the B-vote patterns of the test examples. Our "out-of-bag correction" attempts to reverse directions by estimating B-vote patterns from the out-of-bag voting patterns of the training examples. The difficulty with making this estimate is that the probability of a B-vote pattern given an out-of-bag voting pattern depends on the prior distribution of the B-vote patterns.

For a given trial with a two-class dataset, designate one class to be the majority class, and let the other class be the minority class. The majority class is determined using the training set. Assume that B predictors are in the bag.

For a given example, let $E_B(x, y)$ be the event of x votes for the majority class and y votes for the minority class, where $x + y = B$. Let $E_O(u, v)$ be the event of u votes for the majority class and v votes for the minority class, where the votes are a subsample of the B

votes, where each vote is independently selected to be in the subsample with probability $1/e$. A probability distribution is specified by assigned priors to $P(E_B(x, y))$. We note that:

$$P(E_O(u, v) \mid E_B(x, y)) = b(u, x, 1/e)\, b(v, y, 1/e) \tag{1}$$

$$P(E_O(u, v)) = \sum_{x=0}^{B} P(E_O(u, v) \mid E_B(x, B - x))\, P(E_B(x, B - x)) \tag{2}$$

where $b(k, n, p)$ is the probability of k successes in n i.i.d. Bernoulli trials, each with probability of success p. That is, $E_O(u, v)$ means that u of the x votes for the majority class were chosen, and v of the y votes for the minority class were chosen.[3]

The out-of-bag correction for a given training example α is based on computing $P(E_B(x_\alpha, y_\alpha) \mid E_O(u_\alpha, v_\alpha))$, where u_α and v_α are the known out-of-bag votes for training example α. The probability distribution of $E_B(x, y)$ needed for this calculation is estimated from the out-of-bag votes of the other training examples with the same label as α. That is, x_α and y_α are expected to be similar to the x and y votes for the other training examples. Leaving out α from the distribution estimate avoids a resubstitution bias. Leaving out the training examples with a different label avoids a mixture of different distributions.

Suppose then that there are n other training examples with the same label as training example α. The distribution estimate is based on computing $P(E_B(x_i, y_i) \mid E_O(u_i, v_i))$ for each training example i, $1 \leq i \leq n$, assuming a uniform distribution for $E_B(x, y)$. With no prior knowledge of how $E_B(x, y)$ is distributed, a uniform distribution leads to a qualitatively reasonable approximation, e.g., the ratio of x_i to y_i is expected to be similar to the ratio of u_i to v_i, but the possibility of dissimilar ratios cannot be excluded. Because the calculation for training example i does not take any other training examples into account, this estimate might be expected to be less effective when there are more training examples, but this effect was not observed in our experiments.

The out-of-bag correction for a training example α is computed as follows. Define a uniform distribution $P_{\mathcal{U}}(E_B(x, y)) = 1/(B + 1)$ for $x \in \{0, 1, \ldots, B\}$ and $y = B - x$. It follows that:

$$P_{\mathcal{U}}(E_B(x, y) \mid E_O(u, v)) = \frac{P_{\mathcal{U}}(E_O(u, v) \mid E_B(x, y))\, P_{\mathcal{U}}(E_B(x, y))}{P_{\mathcal{U}}(E_O(u, v))} \tag{3}$$

Equations (1) and (2) specify the calculations that are needed. Then, over the n other training examples with the same label as α, we define a probability distribution $\mathcal{D}$:

$$P_{\mathcal{D}}(E_B(x, y)) = \frac{\sum_{i=1}^{n} P_{\mathcal{U}}(E_B(x_i, y_i) \mid E_O(u_i, v_i))}{n} \tag{4}$$

Using this probability distribution, we estimate the B-vote predictions for training example α by calculating $P_{\mathcal{D}}(E_B(x_\alpha, y_\alpha) \mid E_O(u_\alpha, v_\alpha))$. We can then determine the probability that B-vote voting favors the majority class:

$$\sum_{x \geq y} P_{\mathcal{D}}(E_B(x_\alpha, y_\alpha) \mid E_O(u_\alpha, v_\alpha)) \tag{5}$$

and the minority class:

$$\sum_{x<y} P_{\mathcal{D}}(E_B(x_\alpha, y_\alpha) \mid E_O(u_\alpha, v_\alpha)) \tag{6}$$

If α's label is the majority/minority class, then the probability for the minority/majority class is added to the out-of-bag correction. In our experiments, the total out-of-bag correction is compared to the number of errors on the test examples.

2.2. Statistical tests

The following statistical tests were employed to compare the results of different experiments over the 1000 trials.

To evaluate the bias of an estimate, we performed a paired difference t test (paired comparison of means). For a given estimate on a given trial, we compute the estimate minus test error. Specifically, this test evaluates the hypothesis that the average estimate of generalization error has the same expected value as the average test error. To pass this test with a 5% significance level, the magnitude of the t value should be no more than 1.962. t is computed by $t = \bar{X}/\sqrt{s^2/n}$, where $\bar{X}$ and s^2 are respectively the sample average and sample variance over $n = 1000$ samples.

To compare the accuracy of two estimates, we performed another paired difference t test. We determine the "accuracy" of an estimate by computing the absolute value of the difference between the estimate and the test error. A more accurate estimate will have values closer to 0. We are interested in whether our out-of-bag correction is more accurate than the other estimates. So for a given estimate on a given trial, we compute the accuracy of the estimate minus the accuracy of the out-of-bag correction. Specifically, this test evaluates the hypothesis that the accuracy of an estimate has the same expected value as the accuracy of the out-of-bag correction.

The need for both statistical tests is because an unbiased test is not necessarily accurate, and vice versa. An unbiased test could have a high variance, which would tend to make it inaccurate. A biased test with a lower variance could be more accurate.

3. Results

3.1. Bias

Table 1 shows some statistics for the bias of the estimates on 10 data sets using $B = 50$ predictors generated by ID3. The first column gives the abbreviations for the datasets (see the first Appendix), and the second column gives the test error percentage. The following columns show the averages and standard deviations of the observed bias, i.e., the difference between each of the estimates and test error. OOB, CV, OOC, and TEC respectively stand for out-of-bag estimate, 10-fold cross-validation, out-of-bag correction, and test error correction.

Table 1. Results for ID3, $B = 50$: Averages and standard deviations of bias.

		vs. Test error			
Data Set	Test Error	OOB	CV	OOC	TEC
BA	27.550	0.883 ± 4.51[a]	0.306 ± 4.87[a]	0.234 ± 4.42	-0.059 ± 4.25
CR	16.241	0.428 ± 3.13[a]	0.266 ± 3.08[a]	0.055 ± 3.08	0.072 ± 3.06
FL	20.518	0.073 ± 2.73	0.146 ± 2.75	-0.020 ± 2.74	0.027 ± 2.70
IO	8.177	0.456 ± 3.14[a]	0.264 ± 3.21[a]	0.036 ± 3.09	0.047 ± 3.13
M1	1.772	0.778 ± 1.89[a]	1.024 ± 2.12[a]	0.344 ± 1.79[a]	0.044 ± 1.89
M2	51.931	-0.899 ± 5.26[a]	-1.390 ± 5.23[a]	-0.497 ± 5.28[a]	-0.096 ± 5.35
M3	0.000	0.004 ± 0.05[a]	0.014 ± 0.12[a]	0.004 ± 0.04[a]	0.003 ± 0.05[a]
PI	24.737	0.712 ± 3.24[a]	0.224 ± 3.31[a]	0.174 ± 3.22	0.216 ± 3.22[a]
PR	17.845	1.704 ± 9.65[a]	1.012 ± 9.62[a]	0.557 ± 9.29	0.002 ± 9.02
SO	22.811	1.054 ± 6.44[a]	0.639 ± 6.31[a]	0.102 ± 6.29	0.097 ± 6.27
Average		0.520 ± 4.00	0.250 ± 4.06	0.099 ± 3.92	0.035 ± 3.89

[a]Statistically significant at the 0.05 level.

Table 2. Number of rejections by the bias test by algorithm and size of bag.

		Rejections by bias test			
Predictor algorithm	Bag size	OOB	CV	OOC	TEC
ID3	50	9	9	3	2
ID3	100	9		2	1
C4.5	50	7	6	2	0
C4.5	100	6		1	0

The table shows that both the out-of-bag estimate and 10-fold cross-validation tend to slightly overestimate test error. On average, the out-of-bag estimate differed from test error by 0.52% on average, and 10-fold cross-validation differed by 0.25%.

The out-of-bag correction and the test error correction do much better. As we expected, the test error correction is relatively unbiased; it differed by only 0.04% on average. As we hoped, the out-of-bag correction is also much less biased than the out-of-bag estimate and 10-fold cross-validation; it differed from the test error by 0.10% on average.

The standard deviations of the estimates are fairly close. The out-of-bag correction and the test error correction have slightly lower standard deviations.

Table 2 shows the results of the bias test using ID3 or C4.5 as the predictor algorithm and using 50 or 100 as the size of the bag. As can be seen, the bias test showed that the out-of-bag estimate and 10-fold cross-validation ($B = 50$ only) was significantly different (0.05 level) from test error more often than the out-of-bag correction. The out-of-bag correction was comparable to the test error correction, which was nearly unbiased.

Table 3. Results for ID3, $B = 50$: Averages and standard deviations of accuracy difference.

	vs. accuracy of OOC		
Data set	OOB	CV	TEC
BA	0.137 ± 0.98[a]	0.384 ± 2.29[a]	-0.136 ± 2.89[a]
CR	0.065 ± 0.55[a]	-0.025 ± 1.15	-0.001 ± 0.99
FL	-0.001 ± 0.30	0.010 ± 0.89	-0.022 ± 0.72
IO	0.085 ± 0.62[a]	0.097 ± 1.22[a]	0.026 ± 1.06
M1	0.221 ± 0.57[a]	0.528 ± 1.27[a]	0.003 ± 0.84
M2	0.048 ± 0.94	0.084 ± 2.37	0.050 ± 2.02
M3	0.000 ± 0.02	0.010 ± 0.11[a]	0.000 ± 0.03
PI	0.075 ± 0.76[a]	0.065 ± 1.38	-0.016 ± 1.23
PR	0.412 ± 1.96[a]	0.221 ± 4.20	-0.290 ± 3.65[a]
SO	0.202 ± 1.44[a]	0.070 ± 2.93	0.032 ± 2.54
Average	0.124 ± 0.81	0.144 ± 1.78	-0.035 ± 1.60

[a]Statistically significant at the 0.05 level.

We conclude that out-of-bag estimate and 10-fold cross-validation have a significant, but small bias. The out-of-bag correction has a much smaller bias. The performance of the test error correction partially validates it as a model of out-of-bag voting.

3.2. Accuracy

Table 3 shows some statistics for the accuracy of the estimates on the 10 data sets using $B = 50$ predictors generated from ID3. The first column gives the abbreviations for the datasets (see the first Appendix). The following columns show the averages and standard deviations of the difference between the accuracy of the out-of-bag correction and the accuracy of the other estimates.

The table shows that our out-of-bag correction had better accuracy than the out-of-bag estimate (0.12% on average) and 10-fold cross-validation (0.14% on average). On the other hand, the out-of-bag correction appeared to have slightly worse accuracy (-0.04% on average) than the test error correction. The standard deviations for the out-of-bag correction vs. the out-of-bag estimate are much lower than the other pairs because these two estimates are closely related.

Table 4 shows the results of the accuracy test using ID3 or C4.5 as the predictor algorithm and using 50 or 100 as the size of the bag. Each entry in the table list two numbers: the number of datasets that the out-of-bag correction was respectively significantly better and significantly worse (0.05 level).

As can be seen, the accuracy test showed that the out-of-bag correction was significantly more accurate than the out-of-bag estimate. The results are mixed comparing the accuracy of the out-of-bag correction to 10-fold cross-validation. When ID3 is the predictor algorithm, the out-of-bag correction was significantly better than 10-fold cross-validation on 4 of

Table 4. Number of rejections by the accuracy test by algorithm and size of bag.

		Rejections by accuracy test		
Predictor algorithm	Bag size	OOC vs. OOB	OOC vs. CV	OOC vs. TEC
ID3	50	7–0	4–0	0–2
ID3	100	5–0		0–1
C4.5	50	7–0	3–2	0–3
C4.5	100	5–0		0–3

the datasets, and significantly worse on none of the datasets. When C4.5 is the predictor algorithm, 3 datasets were significantly better while 2 datasets were significantly worse, and the out-of-bag correction had a better average on 4 of the other 5 datasets. The out-of-bag correction is slightly worse than the test error correction.

We conclude that the out-of-bag correction is more accurate than the out-of-bag estimate. It is not as clear whether the out-of-bag correction is generally more accurate than 10-fold cross-validation, but our results give some indication that the out-of-bag correction is better. As we would expect for our model, the test error correction was more accurate than the other three estimates (some data not shown).

4. Conclusion

With the use of any learning algorithm, it is important to use as many examples as possible for training the hypothesis (or hypotheses) from a dataset. It is also important to determine a good estimate of generalization error so that we can have confidence that a good hypothesis has been learned. Our methodology statistically compares an estimate of generalization error determined from a training set to the empirical error on a separate test set.

10-fold cross-validation is one way of estimating generalization error, while using all of the examples for training, but our experiments and previous experiments have shown that it is biased. When bagging is used, the out-of-bag estimate can be to estimate generalization error, and it also uses all examples that are available. Unfortunately, the out-of-bag estimate is also biased.

We have developed a model, the test error correction, based on the voting patterns on the test examples. Our model empirically provides a nearly unbiased estimate and is more accurate than the out-of-bag estimate and 10-fold cross-validation. However, our model is not practical because it cannot be applied until the bag is evaluated on test examples.

Based on this model, we developed an out-of-bag correction based on the voting patterns on the training examples. This correction attempts to estimate the distribution of votes, and so, must be regarded as heuristic. The out-of-bag correction is relatively unbiased compared to 10-fold cross-validation and the out-of-bag estimate, it is clearly more accurate than the out-of-bag estimate, and it appears to be just as accurate as 10-fold cross-validation, if not more accurate.

We conclude that 10-fold cross-validation and the uncorrected out-of-bag estimate should be cautiously used for generalization error estimates because they are biased. For two-class datasets, the out-of-bag correction is a better estimate. The out-of-bag correction uses all the data, is much less biased, has just as good or better accuracy, and avoids the additional time needed for 10-fold cross-validation. However, the out-of-bag correction does not perform as well as our test error correction model, so there appears to be room for improvement. Further research is needed to improve the out-of-bag correction and develop generalization error estimates for bagging on other types of datasets.

A Appendix: Datasets

For each dataset, we list our abbreviation, the number of examples, the number of attributes, and a brief description. The datasets come from the Irvine dataset (Blake & Merz, 1998) or the C4.5 distribution (Quinlan, 1993). We did not consider larger datasets because of the time required to perform bagging and 10-fold cross-validation multiple times.

BA, 550, 35. The UCI cylinder bands dataset.

CR, 690, 15. The C4.5 credit card applications dataset.

FL, 1066, 10. The UCI solar flare dataset. This was changed to a two-class dataset: any flare activity vs. no flare activity.

IO, 351, 34. The UCI ionosphere dataset.

M1, 432, 6. The C4.5 monk 1 dataset.

M2, 432, 6. The C4.5 monk 2 dataset.

M3, 432, 6. The C4.5 monk 3 dataset. The dataset in the C4.5 distribution has no classification noise.

PI, 768, 8. The UCI Pima Indians diabetes dataset.

PR, 106, 57. The UCI promoter gene sequence dataset.

SO, 208, 60. The UCI sonar signals dataset.

Acknowledgments

This research was funded in part by Texas Higher Education Coordinating Board grant ARP 010115-0225-1997.

Notes

1. 10-fold cross-validation is unbiased for a training set subsample of size $9n/10$, where n is the number of examples in the training set. However, learning from the whole training set tends to result in lower error.
2. This assumes that the examples in the dataset are independently drawn from some probability distribution, and that the probability mass of the training set S is near 0%. These assumptions are not true for at least the monks datasets. In this case, the various generalization error estimates can be treated as estimates of error on examples outside of the training set.
3. There is a straightforward generalization of Eqs. (1) and (2) to multiclass datasets, but it leads to a combinatorial number of $E_B(x_1, \ldots, x_c)$ and $E_O(u_1, \ldots, u_c)$ events, where c is the number of classes. We restricted ourselves to two-class datasets because it simplifies the computation of our corrections and because many two-class datasets are available.

References

Blake, C. L., & Merz, C. J. (1998). UCI repository of machine learning databases. [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, California: Department of Information and Computer Science, University of California.

Breiman, L. (1996a). Bagging predictors. *Machine Learning*, *24:2*, 123–140.

Breiman, L. (1996b). Out-of-bag estimation. [ftp://ftp.stat.berkeley.edu/pub/users/breiman/OOBestimation.ps.Z]. Berkeley, California: Department of Statistics, University of California.

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, *40:2*, 139–157.

Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap*. New York: Chapman and Hall.

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on Machine Learning* (pp. 148–156). Bara, Italy: Morgan Kaufmann.

Kearns, M. J., & Ron, D. (1997). Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. In *Proceedings of the Tenth Annual Conference on Computational Learning Theory* (pp. 152–162). Nashville, Tennessee: ACM Press.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (pp. 1137–1143). Montréal: Morgan Kaufmann.

Maclin, R., & Opitz, D. (1997). An empirical evaluation of bagging and boosting. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence* (pp. 546–551). Providence, Rhode Island: AAAI Press.

Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (1994). *Machine learning, neural and statistical classification*. Englewood Cliffs, New Jersey: Prentice Hall.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, *1:1*, 81–106.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. San Mateo, California: Morgan Kaufmann.

Quinlan, J. R. (1996). Bagging, boosting, and C4.5. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence* (pp. 725–730). Portland, Oregon: AAAI Press.

Tibshirani, R. (1996). Bias, variance and prediction error for classification rules. [http://www-stat.stanford.edu/~tibs/ftp/biasvar.ps]. Toronto: Department of Statistics, University of Toronto.

Weiss, S. M., & Kulikowski, C. A. (1991). *Computer systems that learn: Classification and prediction methods from statistics, neural nets, machine learning, and expert systems*. San Mateo, California: Morgan Kaufmann.

Wolpert, D. H., & Macready, W. G. (1999). An efficient method to estimate bagging's generalization error. *Machine Learning*, *35:1*, 41–55.

Received August 11, 2000
Revised January 12, 2001
Accepted January 12, 2001
Final manuscript January 26, 2001