Small Sample Statistics for Classification Error Rates I: Error Rate Measurements

J. Kent Martin and D. S. Hirschberg
(jmartin@ics.uci.edu) (dan@ics.uci.edu)
Department of Information and Computer Science
University of California, Irvine
Irvine, CA 92697-3425
Technical Report No. 96-21
July 2, 1996

Abstract

Several methods (independent subsamples, leave-one-out, cross-validation, and bootstrapping) have been proposed for estimating the error rates of classifiers. The rationale behind the various estimators and the causes of the sometimes conflicting claims regarding their bias and precision are explored in this paper. The biases and variances of each of the estimators are examined empirically. Cross-validation, 10-fold or greater, seems to be the best approach; the other methods are biased, have poorer precision, or are inconsistent. Though unbiased for linear discriminant classifiers, the 632b bootstrap estimator is biased for nearest neighbors classifiers, more so for single nearest neighbor than for three nearest neighbors. The 632b estimator is also biased for CART-style decision trees. Weiss' LOO* estimator is unbiased and has better precision than cross-validation for discriminant and nearest neighbors classifiers, but its lack of bias and improved precision for those classifiers do not carry over to decision trees for nominal attributes.

1 Introduction

The classification problem is: Given a finite set of classified examples from a population, described by their values for some set of attributes, infer a mechanism for predicting the class of any member of the population given only its values for the attributes. Note that this problem is *ill-posed* (see Wolpert [62] and Buntine [14]) — there are usually many hypotheses that will account for a given set of observations, and for this problem in inductive reasoning we are not given sufficient information to guide us in either hypothesis formation or hypothesis evaluation. Consequently, lacking domain-specific knowledge for the problem at hand, our analysis must be (perhaps implicitly) predicated on an assumption as to what kinds of relationships between attributes and classes we are likely to encounter (see Wolpert [59, 61] for a rigorous treatment of these issues of generalization). Since our knowledge of the universe is faulty, to say the least, and nature is not bound by our assumptions, no single method for inferring a classifier can be shown to be uniformly superior from first principles (see also Schaffer [47, 48], for a discussion of generalization as a zero-sum enterprise, the 'no free lunch principle').

These caveats about generalization and algorithmic learning aside, there is abundant evidence that there are broad classes of similar problems for which particular inference approaches appear to work well [47, 61]. A current interest in machine learning is in characterizing problems so as to match them to an appropriate method [4, 26, 50] and in building hybrid classifiers [11, 35].

Many classifier inference methods have been proposed, most falling into one of the following four families: nearest neighbors, discriminant analyses, decision trees or symbolic concept learners, and neural networks. Regardless of the inference method, there are three immediate questions: (1) given a classifier, how accurate is it? (usually, this can only be estimated), (2) given an estimate of accuracy, how accurate and how precise is the estimate (what are its bias, variance, and confidence interval)?, and (3) how much confidence can be placed in an assertion that one classifier is more accurate than another?

Answering these questions is just as much an ill-posed problem of inductive reasoning as the problem of inferring classifiers, and subject to all the difficulties raised in the articles by Wolpert and Schaffer — no one approach to answering these questions is superior for every combination of problem data set and classifier inference method, *i.e.,* methods that work well for discriminant analysis applied to mixtures of multinormal distributions may fare poorly for a nearest neighbors approach to similar problems [33, 36, 56]. Nontheless, there is abundant empirical evidence that methods such as cross-validation work well for many, perhaps most, of the situations in machine learning and pattern recognition which have been studied to date (see Schaffer [47] and Wolpert [61] for extended analyses of the applicability of cross-validation).

In this paper we deal with the first question above and a portion of the second, with methods for estimating a classifier's accuracy and the bias and variance of the estimates obtained from various methods. A second paper [40] deals with the remainder of the second question, confidence intervals, and with the third, significance tests. The thesis of both papers is that *"...the traditional machinery of statistical processes is wholly unsuited to the needs of practical research ...the elaborate mechanism built on the theory of infinitely large samples is not accurate enough for simple laboratory data. Only by systematically tackling small sample problems on their merits does it seem possible to apply accurate tests to practical data." — R. A. Fisher [28] (1925)*

Given this thesis, it behooves us to provide guidelines as to when a sample is considered small, and when traditional methods will suffice. There is no hard rule here. The 150 instances in the Iris data [5, 27], for instance, seem adequate for inferring an accurate classifier and for estimating

its accuracy and confidence limits by traditional methods. In a more difficult problem (say, one having 16 classes and 100 attributes, contrasted to 3 classes and 4 attributes for the Iris data), a sample of 150 would be very scanty. Schaffer's [46] notion of the sparseness of the data relative to the concept to be learned helps to put this in perspective. Sample size is one component of the equation, complexity of the learned classifier another, and its error rate yet another. The interactions of these factors are discussed in conjunction with experiments in which they arise, and more quantitative guidelines are given in conjunction with specific methods.

Vapnik [53, 54] provides bounds on the error of a learning machine in terms of the ratio of the sample size used in training to a measure (the VC-dimension) of the complexity of the set of functions it is able to implement. The VC-dimension is a non-intuitive simplicity measure similar in concept to Goodman and Smyth's J-measure [30] of the information content of a rule (see Wolpert [60] for a discussion of the practical linkage between abstract and traditional measures of complexity). In many cases, *e.g.,* back-propagation, the VC-dimension must be measured indirectly, by examining rates of convergence. The essential result [54] is that the error rate is only trivially bounded (*i.e.,* $\leq 100\%$) whenever the sample size is less than half the VC-dimension. This criterion (sample size less than half the VC-dimension) could be used to define when a sample is considered small. A more practical heuristic rule takes into consideration the fact that classifiers form a partition of the sample space (*e.g.,* the leaves of a decision tree). Whenever the smallest of these partitions contains fewer than 5 instances, traditional measures such as the χ^2 test of association are very suspect [18, 19] — Vapnik's [55] conditions for uniform convergence of frequencies and Wolpert's [60] assumption of convergence of the mean and mode of a binomial distribution are not satisfied.

There is a substantial body of literature on estimating expected error rates, and a clear consensus that some type of resampling technique is necessary to obtain unbiased estimates. These resampling methods, or *estimators*, fall into four main families: *independent subsamples* for classifier inference and error rate estimation [9, pp. 11-12], *leave-one-out* and *k-fold cross-validation* (subsampling without replacement) methods [9, pp. 12-13], *bootstrap* (subsampling with replacement) methods [25], and *hybrid methods,* such as Efron's [24] 632b bootstrap and Weiss' LOO* [58] method. The cross-validation methods are probably the most widely used, especially when the available samples are small, with the independent subsamples methods being preferred by some when very large samples are available. The bootstrap and hybrid methods are computationally expensive and poorly understood and, hence, not widely used.

There are conflicting reports in the literature as to the bias and precision of the various estimators, as well as to their power for testing differences between classifiers. In addition to a tutorial review of the various methods, this and the companion paper also present new and more extensive empirical studies and a framework for resolving the seemingly contradictory reports.

In Section 2 of the paper, we give a short tutorial on issues relating to error rates, introduce the various methods, and define terminology used in the remainder of the paper. In Section 3 we present results of simulation studies on linear discriminant classifiers for very simple data, which reveal fundamental differences in the bias and precision of the methods. Section 4 presents a brief review of pertinent literature which suggests that the behavior found for linear discriminant classifiers may not generalize to other classifier learning methods for some of the estimators, especially when classifiers are overfitted (*e.g.,* nearest neighbors and decision tree pruning). The results of simulation studies on nearest neighbors classifiers for simple continuous attribute data are presented in Section 5, and Section 6 extends these studies to discrete attribute decision trees.

Significant findings from the various experiments are summarized in Section 7. Only the cross-validation (10-fold or greater) methods appear to exhibit consistent behavior across all of the

learning situations studied here. We caution again that no method can be shown to be superior for all situations, and that great care must be exercised when extrapolating empirical results away from the narrow experiments in which they are obtained. However, we do feel that some methods of statistical inference are more *robust* (trustworthy under departure from assumptions) than others, and seek in this paper to shed light on these issues as they relate to estimating error rates from small samples.

2 Error Rate Terminology and Methods

In this section we provide a tutorial on issues relevant to measuring error rates, and define the terminology used in the remainder of the paper.

For practical purposes, a *population* is defined by a set of members, a set of classes, a set of attributes, and the procedures for measuring or assigning the classification and attribute values. Thus, any measurement errors, naming errors, inconsistencies, or omissions are characteristics of the population, not of an inference method.

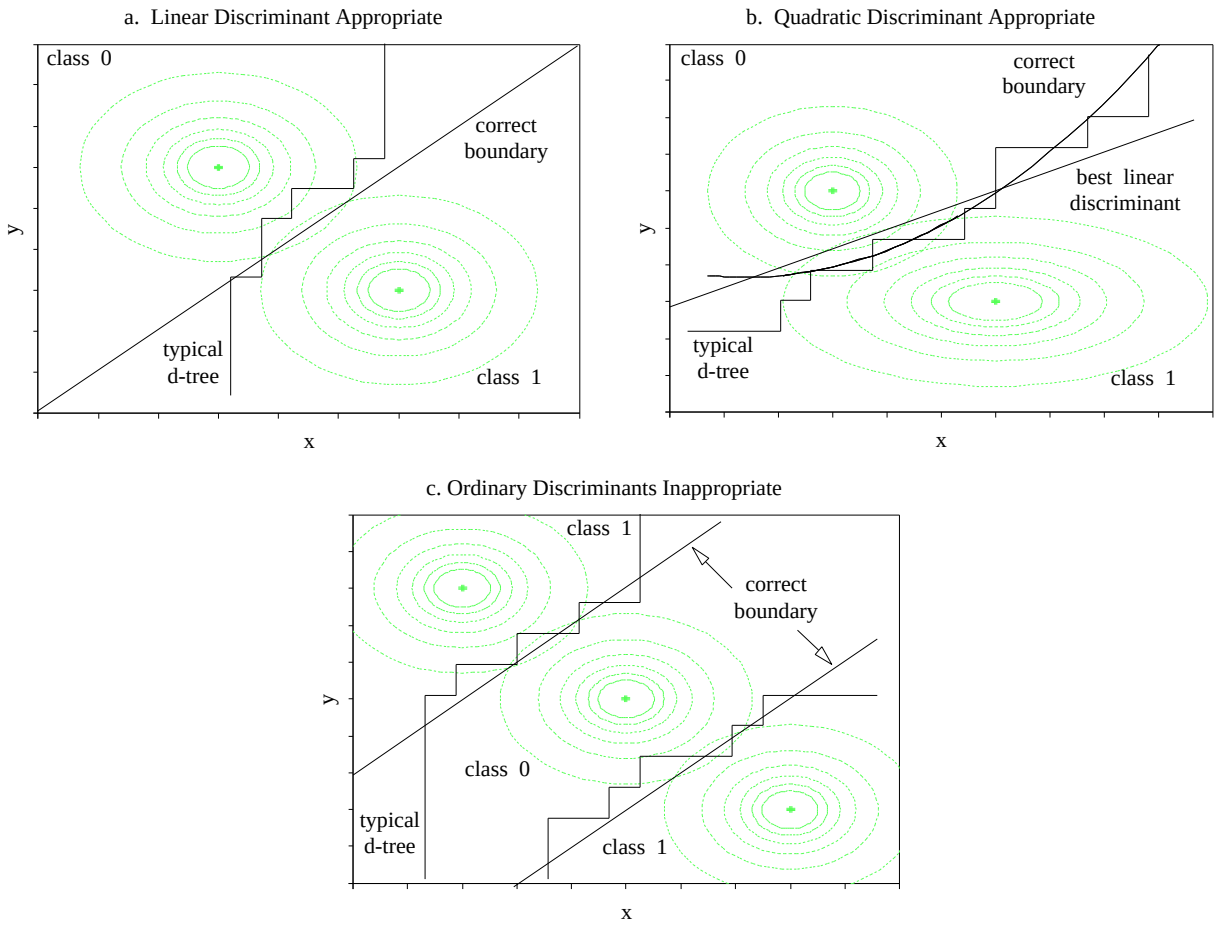
For a given population, there is a hypothetical least-error-rate classifier, known as a *Bayes' Rule* for the population. Its associated *inherent minimum error rate* (inherent error)[1] would ideally be zero, but might well be non-zero because of data errors or because the given attributes are not sufficient to fully separate the classes. This inherent error, also known as the *Bayes' optimal error rate* [9], is a fixed (but unknown) quantity, not a random variable. It is of interest here as a summary value for the population and as a reference target for classifier inference methods.

For a given population and inference method, there is another hypothetical error rate, which is a function of the population, sample size, and representation language (which is often implicitly tied to the inference method). Linear discriminant and single nearest neighbor classifiers, for instance, represent the boundaries between classes by a series of hyperplanes. If the least-error classifier's boundaries are curved surfaces, methods using linear boundaries can only approximate those least-error boundaries. The hypothetical classifier which approximates those boundaries most closely for a given sample size has the *language-intrinsic minimum error rate* (language-intrinsic error), which is greater than the Bayes' Rule inherent error. The language-intrinsic error is a fixed (but unknown) quantity, not a random variable. It is of interest here because it points out one reason that different inference methods can lead to classifiers with very different performance (other causes are the various search strategies and heuristics used).

These distinctions are illustrated in Figure 1. In Figures 1a and 1b, there are two classes labeled 0 and 1, and each class consists of a single multinormal distribution on the real-valued variables x and y, depicted by a set of contour lines of constant probability density. In this situation, the classifier which has least error, *i.e.,* the inherent error, is defined by a curve in the xy plane along which the probability density of class 0 equals that of class 1. When the covariance matrices of x and y for the two classes differ only by a multiplying constant, *i.e.,* when the contours have the same shape and orientation, but not necessarily the same size, this curve is a straight line, as illustrated in Figure 1a, and linear discriminant analysis is appropriate — more generally, the classes differ in the ratio of the x and y variances or in their covariance, and the boundary is a quadratic curve, as illustrated

[1] Throughout this paper, the terms error and error rate (meaning misclassification rate) will be used interchangeably. The term bias (rather than error) is used to refer to a systematic difference between an error rate estimate and the true error rate (non-zero average difference), the term precision is used to refer to the variability of such differences, and the terms variance or standard deviation to refer to the variability of a particular estimate.

Figure 1: Illustration of Language-Intrinsic Error



in Figure 1b. In Figure 1b we also show the linear boundary which has the lowest (the language-intrinsic) error for these data, and for both figures we also show typical CART-style decision tree boundaries. CART-style trees express the boundaries as step functions which can asymptotically[2] approximate the true boundaries here, given a sufficiently large sample and inferring a very complex tree, but cannot exactly express the correct concept with a finite classifier. Linear discriminant analysis has a fixed complexity, and cannot exactly express the correct concept in Figure 1b, nor even approximate it closely, regardless of the sample size. Of course, there are other data sets, especially those featuring nominal attributes, where CART-style trees are more appropriate, and even a quadratic discriminant cannot express those concepts well.

Figure 1c illustrates another case where ordinary discriminant analyses fail. Here, there are 3 distinct subpopulations, but only two classes. The correct boundary in this particular case is a pair of parallel lines (the correct concept here is class=0 if $|y-x| \leq c$, else class=1). This case is superficially similar to that in Figure 1a, but the correct boundary cannot be found or even closely approximated by the usual discriminant analyses because these data violate the fundamental assumption underlying those techniques — namely, that each class is homogeneous, closely approximated by a

[2]Gordon and Olshen [31, 32] showed that nonparametric recursive partitioning (*e.g.,* decision tree) methods asymptotically converge to the Bayes' rule rate. For small samples, these asymptotic results are irrelevant. Also, methods of fixed complexity, such as linear discriminant analysis, cannot be shown to so converge.

single multinormal distribution. The best that a discriminant classifier can do in this case is to set the boundary as a single line perpendicular to the correct boundary lines and outside the range of the data, *i.e.,* to default to the rule of always guessing the more frequent class. Here, even though the decision tree boundaries are a poor approximation, they are a significant improvement over the usual discriminants (*i.e.,* a CART-style tree actually has less language-intrinsic error).

These examples illustrate the fact that, in choosing to use a particular learning algorithm (inference method), we are implicitly making assumptions about the population (the nature and distribution of the attributes and classes) and the language of a correct, minimum error classifier. As in all problems of statistical inference, probably the most crucial step is correctly matching these premises or underlying assumptions to the problem at hand.

References are frequently found in the literature, *e.g.*, in the classic CART text [9, pp. 13-17,269-271], to a Bayes' rule or Bayes' or Bayesian classifier or rate. As defined by CART [9, pp. 13-14], the *Bayes' optimal error rate* is synonymous with the inherent error, in that any other classifier has at least this error rate. This Bayes' nomenclature is confusing, for two reasons: (1) the term Bayes' rule is sometimes used in the context of a particular kind of classifier (*e.g.*, a CART-style decision tree), of a "no data optimal rule" [9, pp. 178,186,190], or of finding a Bayes' optimal classifier for a partition [9, pp. 269-271] — these are references to the language-intrinsic error, not to the inherent error (*i.e.,* the ideal CART-style decision tree is not necessarily the best possible classifier), and (2) these terms are easy to confuse with Bayes' Theorem and Bayesian statistical analysis — they might be misconstrued as any classifier inferred using Bayesian techniques [12, 13, 16, 37, 39], or as only those classifiers.

Given a population, a sample of N items from the population (the *sample* is here defined to be all data currently available for inference and testing), and a classifier inferred from the sample by some means, that classifier has a *true error rate* — the fraction of items that would be misclassified if the entire population could be tested. For any deterministic classifier, the true error rate is a fixed (but unknown) quantity, a function of the population and classifier, and not a random variable.

If only a random subset of Q items from the sample is used to infer the classifier (a *training set*, the unused items forming a *test set*), there is usually a very large number[3] of distinct possible training/test splits. Since the training set is random, the inferred classifier is random — if the splitting is repeated, a different result will probably be obtained due simply to the *random resampling variance*.

Under random resampling, although the true error of the particular classifier is a fixed quantity, it is more appropriate to speak of the *true error rate of the resampling estimator* — the expected (mean) value of the true error rates of these individual splits' inferred classifiers, averaged over all possible splits. The true error of a particular split's classifier, or the average of the true error rates over several splits, is only an estimate of that expected value, and is a random function of the population, sample, and inference method. Since the sample is not the entire population, the true error of any particular classifier can only be estimated from this sample data by some method. When random resampling is used in obtaining the particular classifier(s), this becomes a process of estimating the value of an estimate.

When more than one training/test split is used and estimated errors averaged, a troublesome question arises: to exactly what classifier does this averaged error rate correspond? When the classifiers are decision trees, for example, there is no practical notion of what it would mean to average the classifiers. To answer this question note that (as shown later, see Table 4) for any

[3] $N!\ /\ Q!\ (N-Q)!$ if Q is fixed, otherwise $\mathcal{S}(2,N) = 2^{N-1} - 1$ (a Stirling number of the second kind [1]).

inference method, the classifier whose true error is closest to the language-intrinsic error is usually[4] to be obtained by using the entire sample for classifier inference. Then, the solution is fairly clear: infer a classifier using all available data and some inference method, and estimate that classifier's true error using one or another estimator. One set of criteria for evaluating an estimator are the *bias* and *precision* with which it estimates that whole-sample true error, measured by the average and rms[5] values of (EST − TER) over a wide range of populations and sample sizes (where EST is the estimated and TER the true error).

Note that bias and precision are not fixed properties of an estimator, but depend on the classifier inference method being used, on the characteristics of the population, and on the sample size. For inference methods which allow classifiers of different complexities to be inferred, *e.g.,* by pruning a decision tree, bias and precision may also vary with the complexity of the classifier. Good experimental practice would dictate that the bias and precision be established for the experimental conditions at hand by simulation of known populations with characteristics similar to those believed to obtain for the problem population at hand (this inevitably involves some guesswork, as it is the structure of the problem population that we are trying to uncover in inferring a classifier). A more practical policy would dictate use of error estimators that have been shown by such studies to be robust under fairly general conditions, and to avoid estimators that are known not to be robust.

As noted in the introduction, various estimators have been proposed:

- *Apparent error rate* — The fraction of items misclassified when testing on the same items used to infer the classifier (the training set), sometimes called the resubstitution estimate [9, p. 11]. The apparent error is known to be biased (optimistic). In simple nearest neighbors, for instance, every training item is its own nearest neighbor, resulting in an apparent error of zero if the data are consistent. This problem is sometimes solved by finding the nearest non-identical neighbor, which can be extended to other classifier types as the leave-one-out method (see below).

- *Independent subsamples* — The sample is randomly split into a training set from which the classifier is inferred and a test set from which the estimated error rate is later determined. Typically either one-half, one-third, or one-fourth of the sample is used for the test set. This process can be iterated many times (*random subsampling* [58], *repeated subsampling* [36], or *repeated learning-testing* [17]) and the results averaged to reduce the variance.

- *k-fold cross-validation* — The sample is randomly divided into k approximately equal-size subsets. For each of the subsets, the remaining $k-1$ subsets are combined to form a training set and the resulting classifier's error rate estimated on the reserved subset. A weighted average of the k error rate estimates is used, weighted for the test set size. For $k \ll N$, the entire procedure may be iterated many (typically 100) times and those results averaged. When k equals the sample size, N, the *leave-one-out* (LOO) estimate is obtained. In the statistical literature, the term cross-validation often is used for leave-one-out [23, 24, 62], rather than in a generic sense. Occasionally, leave-one-out is referred to as ordinary [17] or complete cross-validation. Kohavi [36] defines complete k-fold cross-validation more generally as the average over all of the possible training/test splits of the sample for test sets of size N/k (leave-one-out is necessarily complete). Iterating cross-validation approximates a complete k-fold cross-validation, which is frequently not practical.

[4] See Aha [3] and Kohavi [36] for some exceptions to this rule.

[5] Root-mean-squared, $\sqrt{\sum(\text{EST} - \text{TER})^2/N}$

- *Bootstrapping* — A training set of size N is chosen randomly with replacement. Thus, each item in the size N sample may appear 0, 1, or more times in the training set. For large N, an average of $(1-1/e)=63.2\%$ of the items will be used in the training set. Only those items which do not appear in the training set are used for the test set, and only once each. This procedure is iterated many (typically 200) times and the error rates averaged. Efron [24, 25] refers to this estimator as $e0$, distinguishing it from older definitions of the bootstrap which use a different test set.

- *Hybrid methods* — Various combinations of the preceeding estimators have been proposed, such as Efron's [24] 632b bootstrap and Weiss' [58] LOO* method. The principal advantage claimed for the 632b estimator is that, though biased, it has lower variance than the other estimators[6]. 632b is a weighted combination of the $e0$ bootstrapping error (BOOT) and the apparent error (APP), 632b = 0.632 BOOT + 0.368 APP. Weiss' LOO* estimator is

$$\text{LOO*} = \begin{cases} \text{632b}, & \text{if LOO} < \text{632b} \\ \text{2-CV*}, & \text{if 2-CV*} < \text{LOO and 632b} \leq \text{LOO} \\ \text{LOO}, & \text{otherwise} \end{cases}$$

  where 2-CV* is 2-fold cross-validation iterated 100 times.

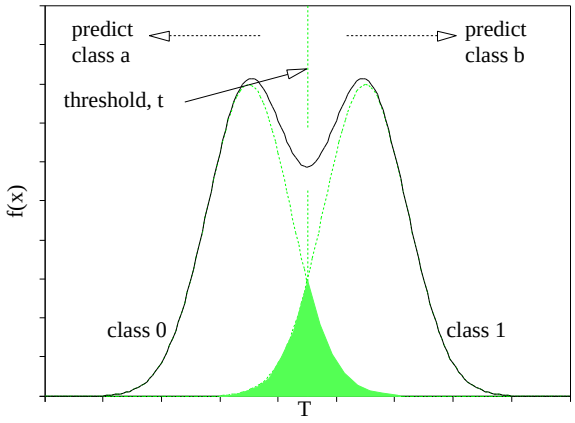
All of these resampling methods would benefit (in terms of reduced variance) from *stratification* [36] — for instance, grouping all the data from the same class and partitioning each class separately so as to keep the class proportions in each partition nearly equal to those in the whole sample. This may increase the sensitivity of certain comparisons, but at the risk of distorting the variance estimate. We typically assume that our whole sample is drawn randomly from the parent population, not in a stratified fashion, and a stratified resampling will tend to underestimate the sample-to-sample variation (the *sampling variance*) of the population. The Iris data [5, 27] can be used to illustrate an interesting point about sampling ...the whole sample is almost certainly stratified (it is doubtful that the 3 species were equally prevalent on the Gaspé peninsula or, if they were, that a random sampling would have produced exactly 50 flowers of each species), and the reported error rates are distorted to the extent that this stratification has distorted the proportions[7]. Stratification is a two-edged sword, and can easily lead to mis-interpreted results.

The iterated, bootstrapped, and hybrid methods are computationally expensive. Efron [23] refers to this as 'thinking the unthinkable' (*i.e.,* that one might be willing to perform 500,000 numeric operations in analyzing a sample of 16 data items), in discussing the impact of computers on statistical theory. Of course, one of the original motivations for using k-fold cross-validation (k-CV) rather than leave-one-out (LOO) was the high computational cost of LOO (for sample sizes greater than 200, even 200 bootstrap iterations may be less expensive than LOO). Even k-CV may be considered expensive at times [9, p. 42], and how large k needs to be to give a close approximation to LOO was an important question for early classifier learning research [9, p. 78]. The intent [24, 58] of iterating k-CV or bootstrapping is to reduce the variance of the estimates in

[6] The precision of a biased estimator (b) is $\sqrt{\text{bias}^2 + \text{variance}(b)}$, while for an unbiased estimator (u) the precision is $\sqrt{\text{variance}(u)}$. If variance($b$) < variance($u$) and bias^2 < variance($u$)−variance($b$) then the biased estimator b is less likely to stray too far from the truth (in the sense of the squared error loss) than is the unbiased estimator u. This is a question of the confidence interval, the interval within which, given the value of the estimate, we expect with high confidence to find the true error rate. These issues are addressed more fully in the companion paper [40]. In these cases, a tradeoff might be made, trading increased cost and perhaps a slight bias to gain improved precision.

[7] Most of the errors occur in distinguishing between two of the species, the other is easily distinguished, and the error rate on future, randomly sampled instances will depend crucially on the true proportions of the species.

Figure 2: A Simple Classifier



order to make comparisons of competing classifiers more reliable, even at the expense of possibly accepting a small bias in the estimates and greatly increasing the computation cost (the hybrid methods aim at correcting for the bias). Whether the bias is acceptable and the extra cost justified can only be decided in the context of a particular problem — sometimes one is willing to pay, sometimes not.

3 Experimental

Some essential properties of these various estimators[8] can be shown using very simple data and a simple kind of classifier, as illustrated in Figure 2. The data population consists of two equally likely classes (labeled 0 and 1), each normally distributed on a single real-valued attribute (x), with different means (μ_0 and μ_1, $\mu_0 \leq \mu_1$) but a common variance σ^2 (assume $\sigma^2 = 1.0$, without loss of generality). From a sample of size N, the inferred classifier is:

$$C(t, a, b) \equiv \text{ if } (x \leq t) \text{ then class} = a \text{ else class} = b$$

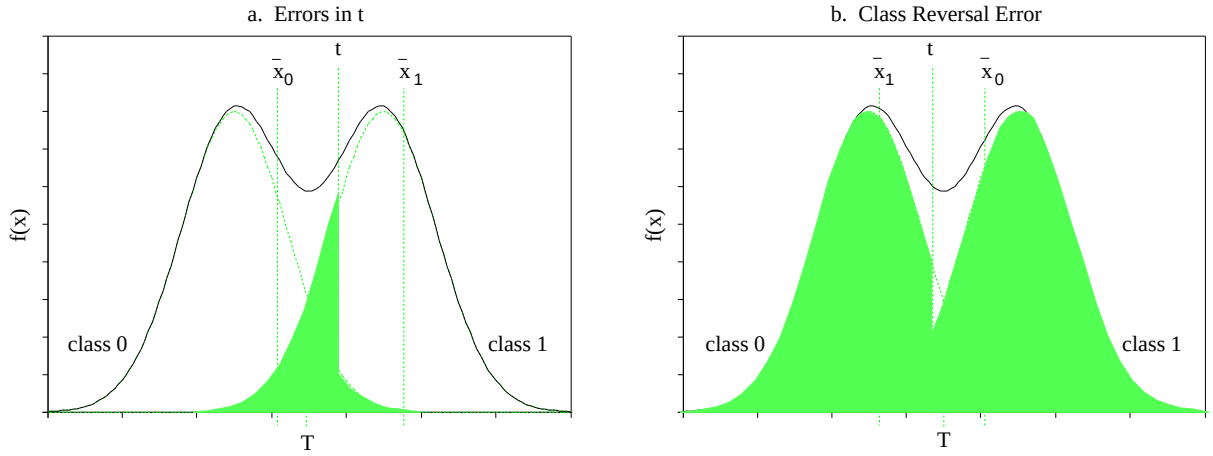
where the threshold, t, and predicted class labels, (a, b), are determined from the training set of size $Q \leq N$ using a simple linear discriminant procedure [34]. The least-error classifier for these data would be $\mathcal{C} \equiv C(T, 0, 1)$, where $T = (\mu_0 + \mu_1)/2$ is the point where the two classes' density functions cross, as shown in Figure 2. The inherent error is equal to the shaded area in Figure 2, provided that the two distribution curves are normalized so that their combined area is unity. For this model, several things can go wrong:

1. The estimated threshold t can differ from T, as shown in Figure 3a, so that the induced classifier $C(t, 0, 1)$ has a true error greater than the inherent error of $\mathcal{C} = C(T, 0, 1)$.

2. The training set's mean values for classes 0 and 1 might be reversed, as shown in Figure 3b. The true error of such a reversed, $C(t, 1, 0)$, classifier is 50% when $\mu_1 = \mu_0$, and increases as $\mu_1 - \mu_0$ increases, asymptotically approaching 100%. However, the likelihood of such a reversal

[8] We caution again that the results presented in this section are illustrative, and may not generalize to other kinds of classifiers or to very complex problem domains (see Section 4 and references [36] and [56]).

Figure 3: Sampling Errors



decreases rapidly as $\mu_1 - \mu_0$ increases. The expected effect of a reversed classifier (the product of these competing effects) peaks, typically in the range $0.25 < \mu_1 - \mu_0 < 0.5$, or 55-60% true error. A detailed analysis of this case in given in the Appendix.

3. All of the items in the training set could be from the same class. For equally frequent classes, this is unlikely (the probability of this happening in a random sample of size Q is 0.5^{Q-1}). If one of the classes is rare, this can be a real problem even for large Q, and special sampling techniques may be needed. In these cases, the classifier always predicts whichever class is observed in the training set, the true error is 50%, and the apparent error is zero.

4. The class means in the training set might be equal. This is rare, but may happen when the x data are rounded with few significant digits. In these cases, the classifier always predicts whichever class is more frequent in the training set, the true error is 50%, and the apparent error is the proportion of the other class in the training set.

Though conditions 2 through 4 are rare, all of these conditions were encountered in these simulations. Let this be a warning that the naive assumption that random sampling somehow guarantees a 'representative' sample may not hold when the sample is small. Vapnik's [55] results make such a guarantee only in the asymptotic case, and then only if certain regularity properties hold. Even a sample of 1,000 instances may under-, over-, or atypically-represent a small (*e.g.*, less than 1%) subpopulation. Stratified sampling would be of great benefit in these cases, but is generally not feasible for observational studies.

Table 4 summarizes the mean bias and precision (averages over 4,000 samples) of the various estimators, and their approximate 95% confidence intervals. The samples represent 100 iterations each of 40 different sample size/inherent error combinations. Five sample sizes (10, 20, 30, 50, 100) and eight inherent error rates (50, 40, 25, 10, 5, 2, 1, and 0.1%) were used for these simulations.

For each sample, the linear discriminant classifier was calculated using the entire sample, and its true error rate was directly computed from our knowledge of the population's normality and its characteristics (μ_0, μ_1, and σ). The various resampling estimators[9] were determined for each

[9]Iterating independent subsamples (ISS) is very similar in spirit and in its results to iterating cross-validation [17], and the ISS methods were not iterated here.

	Estimator		Bias	Precision (rms)
Independent sub-	ISS	k=2	$1.33 \pm .38\%$	$12.41 \pm .27\%$
samples, test set	ISS	k=3	$.64 \pm .40\%$	$12.85 \pm .28\%$
size N/k	ISS	k=4	$.72 \pm .44\%$	$14.37 \pm .32\%$
Apparent error	APP	(k=∞)	$-1.58 \pm .24\%$	$7.94 \pm .17\%$
k-fold cross-	2-CV		$1.10 \pm .29\%$	$9.46 \pm .21\%$
validation	5-CV		$.32 \pm .25\%$	$8.20 \pm .18\%$
k-CV	10-CV		$.31 \pm .25\%$	$7.96 \pm .18\%$
	LOO	(N-CV)	$.21 \pm .25\%$	$8.05 \pm .18\%$
Iterated k-CV	2-CV	$\times$ 100	$1.24 \pm .21\%$	$6.81 \pm .15\%$
and the $e0$	BOOT	$\times$ 200	$.84 \pm .20\%$	$6.55 \pm .14\%$
bootstrap	5-CV	$\times$ 100	$.35 \pm .22\%$	$7.01 \pm .15\%$
	10-CV	$\times$ 100	$.27 \pm .24\%$	$7.59 \pm .17\%$
Hybrids	632b		$-.05 \pm .20\%$	$6.32 \pm .14\%$
	LOO*		$.29 \pm .20\%$	$6.44 \pm .14\%$

Table 4: Overall Results

sample. The bias and precision in Table 4 were calculated from the paired difference between the estimate EST and the true error TER for each sample.

The bias confidence intervals are $\pm 1.96 s/\sqrt{4000}$, where s is the standard deviation of the paired difference (EST $-$ TER) for all 4,000 experiments. The rms precisions' confidence intervals are, for practical purposes, $\pm 2.2\%$ of the rms value[10].

For these experiments, LOO and 632b appear to be unbiased on the whole (the small pessimistic bias for LOO is not significantly different from zero at the 95% level). The other resampling estimators all appear to have a slight positive (pessimistic) bias. The relatively larger rms precision values indicate that all of the estimators may be far from the true error rate in individual cases, with relatively high probability. For these simple discriminant analyses, 632b had the least bias and the best precision of the various estimators.

Two of the bias results (LOO and 632b) are surprising, in light of Efron's findings that 632b had a moderate optimistic bias [24, p322] and LOO was nearly unbiased [24, p318] under similar circumstances. Efron's 632b conclusions [24] were based on only 5 experiments, all at very low sample sizes ($N = 14$ or 20), high inherent error (about 40%, only 0.5σ separation of the classes), and multivariate normal distributions. Recent work by Davison and Hall [21] and by Fitzmaurice, *et al.* [29] showed that the differences in bias and variability emerge strongly only when the populations are very close or the samples very small. Jain, *et al.* [33, p631] reported no consistent difference in bias for either method. CART [9, p77] and Burman [17] have reported that LOO and k-CV should be pessimistically biased (though CART [9, p41] reports 'fairly adequate results' for $k \geq 10$). Burman [17] quantifies the expected bias in k-fold cross-validation as $O(p)/(k-1)N$, where p is the complexity of the classifier (in the present experiments p is fixed). Table 5 gives a closer look at these two estimators as a function of sample size and population inherent error. Corresponding

[10] Note that $\text{rms}^2 - \text{bias}^2 = s^2$, the variance. Confidence intervals for this variance are governed by the F-distribution which, for ν degrees of freedom, gives approximately $s^2(1 \pm 1.96\sqrt{2/\nu})$ as the variance interval for large ν (*i.e.,* $\pm 4.4\%$ for $\nu = 4000$). The bias^2 correction is negligible for all cases in Table 4, giving $(1 \pm .022)$rms as the approximate interval for the rms estimates.

Table 5: A Closer Look at LOO and 632b

Inherent Error %	Sample Size 10		Sample Size 30		Sample Size 100	
	a. (LOO − TER) %					
0.1	.2 ± .4		.1 ± .1		−.0 ± .1	
1	.9 ± 1.0		−.4 ± .4	*	.0 ± .3	
2	1.1 ± 1.3		.4 ± .6		−.0 ± .3	
5	.4 ± 1.6		−.4 ± .7		.4 ± .5	
10	1.0 ± 2.3		−.1 ± 1.1		.2 ± .6	
25	3.2 ± 3.8		−.0 ± 2.0		−.7 ± .9	
40	−.2 ± 3.5		1.4 ± 2.6		.0 ± 1.5	
50	2.0 ± 3.1		1.5 ± 2.3		−.5 ± 1.3	
	b. (632b − TER) %					
0.1	.7 ± .3		.1 ± .1		−.0 ± .0	
1	1.1 ± .8	*	−.4 ± .3	*	.0 ± .1	
2	1.2 ± .9	*	.2 ± .6		−.0 ± .3	
5	.8 ± 1.5		−.3 ± .7		.3 ± .4	
10	1.2 ± 2.2		−.1 ± 1.0		.2 ± .6	
25	.9 ± 2.7		.4 ± 1.6		−.3 ± .8	
40	−2.4 ± 2.6		−.5 ± 1.6		.2 ± 1.0	
50	−1.4 ± 2.5		1.3 ± 1.5		−.4 ± .8	
	c. (LOO − 632b) %					
0.1	−.5 ± .7		−.0 ± .0		.0 ± .0	
1	−.3 ± .4		−.0 ± .1		−.0 ± .0	
2	−.1 ± .7		.1 ± .2		.0 ± .1	
5	−.4 ± .9		−.1 ± .2		.1 ± .1	
10	−.2 ± 1.2		−.0 ± .4		.0 ± .1	
25	2.3 ± 2.0	*	−.4 ± 1.1		−.4 ± .2	*
40	2.4 ± 2.5		1.9 ± 1.6	*	−.2 ± 1.0	
50	3.4 ± 2.1	*	.2 ± 2.0		−.1 ± 1.2	

* Significant at the 95% Confidence Level

d. Correlation Coefficients

Inherent Error %	LOO *vs.* TER, N = 10	30	100	632b *vs.* TER, N = 10	30	100	632b *vs.* LOO, N = 10	30	100
0.1	.07	−.12	−.02	.15	−.11	−.02	.81	.97	.99
1	−.03	−.04	−.08	−.09	−.07	−.04	.92	.95	.98
2	.42	−.04	−.02	.35	−.03	−.02	.87	.96	.96
5	.76	−.09	.02	.69	−.05	.01	.95	.95	.98
10	.65	.05	.16	.59	.07	.19	.92	.94	.98
25	.51	.67	.00	.65	.70	.06	.89	.92	.97
40	.35	.33	.62	.52	.57	.77	.76	.81	.86
50							.74	.51	.48

table entries for each estimator present results for the same 100 simulated samples — the given confidence intervals are $2s/\sqrt{100}$, where s is the standard deviation of the 100 simulated differences.

The bias results in Tables 5a and 5b are very uncertain — the only striking difference being at sample size $N = 10$ and inherent error $\geq 25\%$, where LOO seems to have a positive bias while 632b has a negative bias. The differences in behavior are clearer in Table 5c, paired differences between the two estimators, where the average difference for $N = 10$ is small and negative for inherent error $\leq 10\%$ but large and positive for inherent error $\geq 25\%$. The difference in Table 5c is typically less variable than the bias of either estimator, suggesting that the estimators are more strongly correlated with one another than with the true error. This suggestion is confirmed by the correlation coefficients shown in Table 5d. LOO and 632b (and, in fact, all of the resampling estimators) tend to respond to sample-to-sample differences in the same way, but this is largely independent of the sample-to-sample differences in the true error rates of the inferred classifiers.

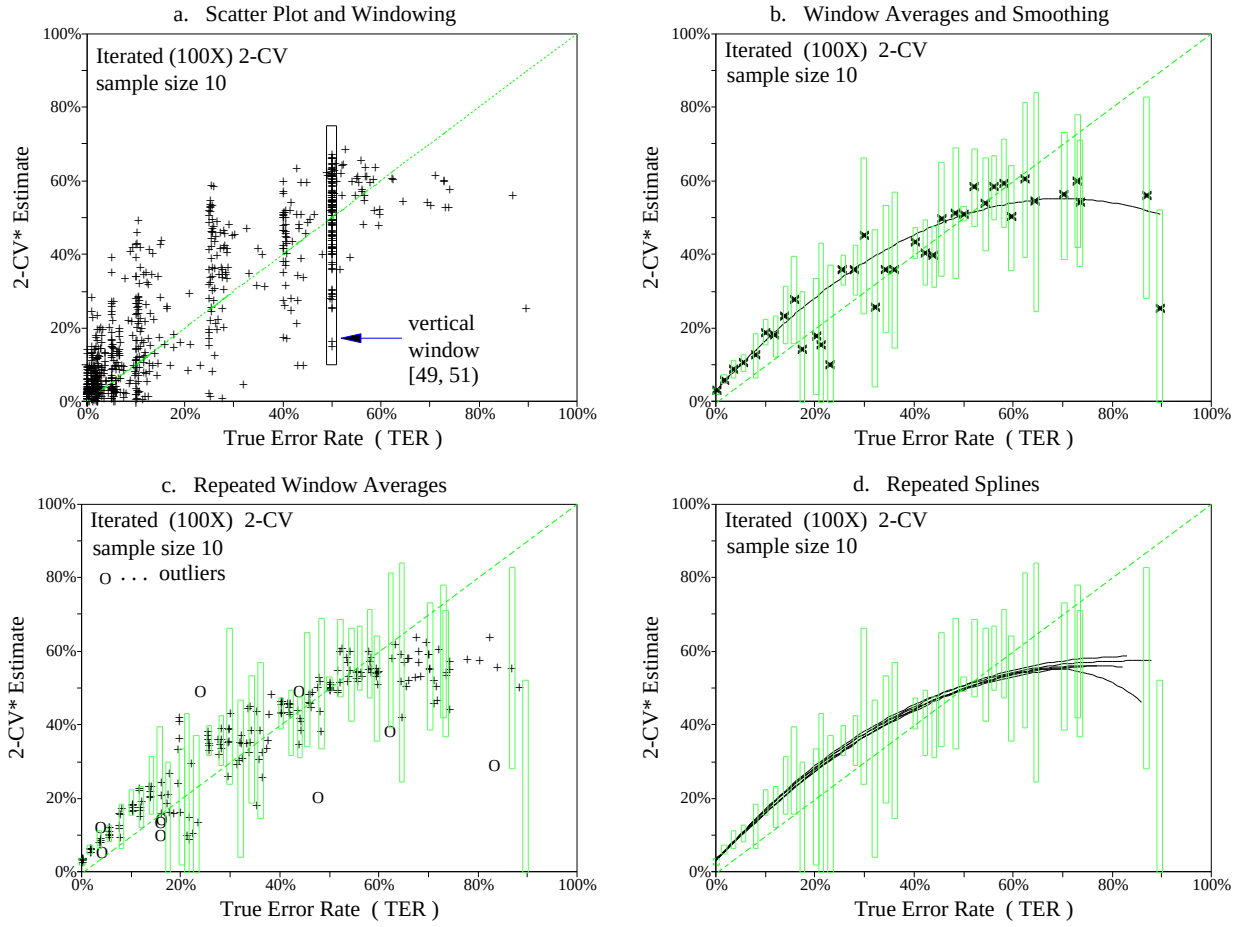
This large variance and lack of predictive correlation from sample-to-sample has important implications for the use of these resampling estimates to compare competing classifiers for a problem population. This will be explored more fully in the second paper, but deserves noting here: different investigators using identical inference methods and estimators but different small random samples from the same population will likely infer slightly different classifiers, with different estimates of their predictive error — for these simple discriminants, the difference between the two estimates has little, if anything, to do with the difference in the true error rates of the two classifiers. Iterating the training/test splits or bootstrapping reduces the sampling variance, as is evident from Tables 5a and 5b; but, apparently, not sufficiently to result in the desired strong correlation between an estimate and the true error. There is currently much research [7, 8, 15, 38, 51, 62, 63] in machine learning and statistics on more robust procedures for selecting a classifier (cross-validating, bootstrapping, 'stacking', or 'bagging' the entire inference procedure).

The methods illustrated in Figure 6 were used to explore the variation of bias with sample size and true error. Figure 6a is a scatter plot of the 2-CV* estimator vs. true error (TER) for samples of size $N = 10$. The 800 data points represent the 100 simulated samples from each of 8 populations (inherent error 0.1, 1, 2, 5, 10, 25, 40, and 50%). The data tend to cluster on the TER axis near the population inherent error rates, but there are some outliers from each experiment.

It is evident in Figure 6a that 2-CV* tends to be optimistically biased for true error rates larger than 60%, while the overall pessimistic bias of 2-CV* is less evident, except perhaps near 25% true error. The points at the extreme right-hand side of Figure 6a bear some explaining. When the two sub-populations coincide (50% inherent error), the true error rate of any partitioning will be 50% — if fraction p of class 0 lies to the right of the threshold (see Figures 2 and 3), fraction $(1-p)$ of class 1 lies to its left, giving a true error of $0.5p + 0.5(1-p) = 50\%$ for all p. Thus, it might seem that a true error higher than 50% is not possible. However, if the two sub-populations are separated only slightly (say 40% inherent error), then the kind of class-reversal error shown previously in Figure 3b can occur, resulting in a very high true error (*i.e.,* it can happen that the mean value of x for class 0 in the sample is higher than that for class 1, which is the reverse of the relationship of the sub-population means[11]).

[11] Such atypical samples are relatively rare (46 of the 800 observations in Figure 6a, or about 6%), and the likelihood decreases rapidly with increasing sample size, but these rare events do occur. It is a mistake to think that a random sampling plan guarantees a representative sample — at best, it merely guarantees an unbiased sample. Stratified sampling plans can ensure that the sample is representative (at least for the characteristics which are controlled), but these approaches assume knowledge of the very same unknown population characteristics which we are trying to infer from the sample.

Figure 6: A Method for Estimating Bias *vs.* True Error Rate



Bias is an average property, which may be obscured by the large y-axis variance of the scatter plot (and 2-CV* has a relatively low variance among the various estimators). To get a better picture of the bias, the data were partitioned by narrow vertical windows, such as that shown in Figure 6a, and the observations within each window were averaged (the windows used were $(2i \pm 1\%)$, $i = 0 \ldots 50$, chosen so as to keep intact the natural clustering of the data).

The locations of the resulting window averages are shown in Figure 6b, with vertical bars representing the approximate 95% confidence intervals for the averages. Obviously, some of the windows are relatively dense, and others quite sparse. The confidence intervals reflect this is the usual way, *i.e.,* that the standard error of the window average is the standard deviation of the individuals in the window divided by the square root of the number of individuals averaged. However, it is not possible to estimate the standard error for windows which contain only one observation, and the estimates obtained for windows containing fewer than 5 observations are very unreliable[12]. To obtain more reliable estimates for the sparse windows, we assumed that each single observation had a standard error equal to that of the 100 observations of this sample size from this same

[12]Though the formula $\sum_{i=1}^{n}(x_i - \bar{x})^2/(n-1)$ is an unbiased estimator, these standard error estimates are highly variable for small n. For $n = 5$, for instance, two estimates differing by a ratio of $2.5 : 1$ are not significantly different at the 95% confidence level (using the F-test).

population[13]. For windows with multiple observations, these assumed standard errors (se_i) were 'pooled' ($se = \text{rms}\{se_i\}/\sqrt{n}$) to obtain the estimated standard error (se) of the window average. The confidence intervals shown in Figure 6b are simply the $\pm 2se$ limits.

The smooth curve shown in Figure 6b is a smoothing least-squares cubic spline fitted using the procedure given by Dierckx [22][14]. In this format, the pessimistic bias of 2-CV* when the true error (TER) is less than 50% is evident, as is the optimistic bias above 50% TER. Examination of similar plots showed that the pessimistic bias below 50% TER decreases with increasing sample size (approximately as $N^{-1/2}$), but the optimism above 50% TER is little affected by sample size, although the frequency of such atypical results and the highest TER observed do decrease. The bias curves for un-iterated 2-CV were essentially the same as for 2-CV*.

We note that there are two levels of smoothing in our experiments: averaging over the window width and the smoothing factor used in fitting the spline. If narrower windows are used, the result is more data points to be fitted, wider confidence limits for each point, and a fitted curve which is less smooth, though generally following the path shown in Figure 6b — the clustering on the x axis constrains all the fitted curves to pass very nearly through the cluster means, due to their higher weights. A lower smoothing factor likewise results in a curve which is less smooth. Given the relatively high variance of all of the error rate estimators, we believe that such short term fluctuations in the estimated bias have neither practical nor statistical significance.

The experiments shown in Figures 6a and 6b were repeated six more times, using a different seed for the random number generator on each repeat. The resulting window averages are shown in Figure 6c, compared to the confidence intervals obtained in Figure 6b (the intervals shown in Figure 6c are identical to those in Figure 6b) — the +'s and O's in Figure 6c show the 6 iterations' equivalent averages for each window, the difference being that the O's are judged to be outside the confidence intervals, while the +'s are judged to be within the intervals[15]. Ten of the 216 window averages (4.6%) fell outside those intervals, suggesting that those intervals are a very good overall approximation to the 95% confidence intervals for these experiments. The resulting smoothing splines are virtually identical, as shown in Figure 6d.

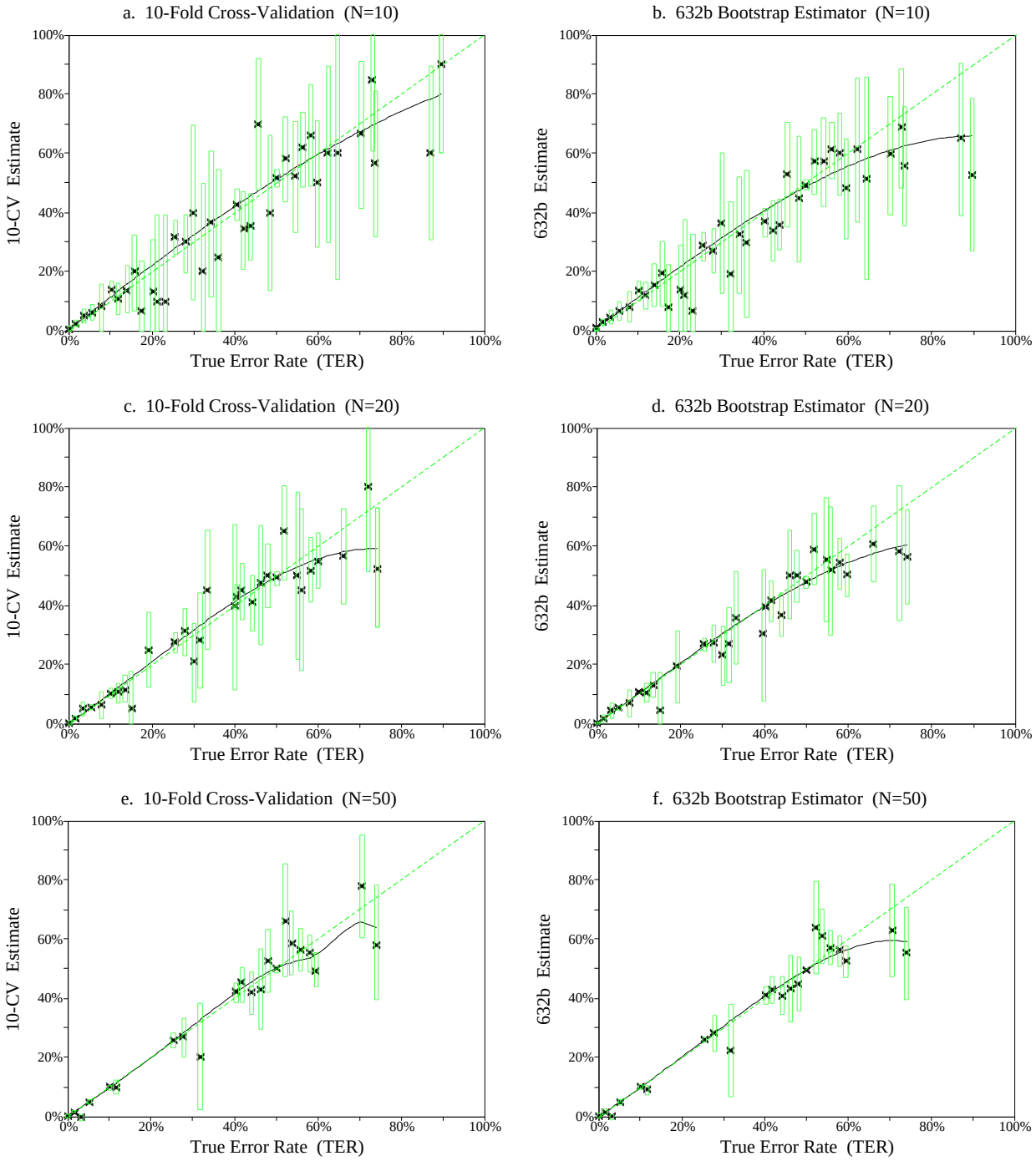
Figure 7 compares the bias of 10-fold cross validation (10-CV) and the 632b bootstrap estimator for sample sizes of 10, 20, and 50. The results for the leave-one-out (LOO) estimator (not shown) are practically the same as those for 10-CV. Below 50% TER, cross-validation has a slight pessimistic bias for sample size $N = 10$ which decreases rapidly with sample size and is negligible for $N \geq 50$. The 632b bootstrap bias in this region is smaller, and is negligible for $N \geq 20$. Above 50% TER, both estimators show a pessimistic bias which may be stronger for 632b than for cross-validation at the smallest sample size. All of the estimators appear to be optimistically biased near or above

[13] It is likely that this assumption slightly understates the variance of the isolated data points in Figure 6a, which are typically outliers for one of the experiments. For the denser windows, the resulting estimates of se were virtually identical to those calculated directly from the observations. Empirical results from further experiments which will be discussed later in connection with Figures 6c and 6d indicate that the assumption leads to a very good overall approximation of the 95% confidence intervals for these experiments.

[14] Each data point (residual deviation from the fitted curve) was weighted by the inverse of its estimated standard error, and the smoothing factor (the targeted weighted sum-of-squares of residuals) was taken to be the number of data points (non-empty windows), m. (Reinsch [45] recommends $m \pm \sqrt{2m}$ when the weights are the inverses of the estimated standard errors, reflecting the fact that such a weighted sum-of-squares has approximately a chi-square distribution, further approximated as a normal distribution with mean m and standard deviation $\sqrt{2m}$.) The choice of a cubic spline is that recommended by Dierckx [22].

[15] In a few cases, there was no corresponding interval from Figure 6b because that experiment had no data in the particular window. In those cases, a datum was judged to be or not be an outlier subjectively, by comparison to other data and intervals at nearby x-axis values.

Figure 7: Estimated Bias of 10-CV and 632b



50% estimated error, but this is of little practical consequence since classifiers with such high error rates are not useful.

For these simple linear discriminant cases, 632b appears to be less biased than 10-CV or LOO, though their bias is small, and also to have lower variance. If this behavior carries over to other classifier types and populations, the improved variance of 632b would offer an advantage in choosing between competing classifiers for a problem. There is, of course, a computational price to be paid for the improved variance. For discriminant analysis and other structurally simple methods, the extra computational time may be mitigated by storing intermediate results and similar methods for rapid updating. For structured classifiers, such as decision trees, the structure itself may be unstable to perturbations of the training set (e.g., the choice of split attribute at the root of the tree may change, which affects every node in the tree and possibly some node additions and deletions) and updating costs may be very high. ITI trees [52] reduce updating costs significantly compared to constructing a new tree, at the expense of perhaps greatly increased storage. Whether the extra cost is justified will be dictated by circumstances and the use to be made of the estimates.

4 Overfitting, Non-independence, and Generality

The apparent error can be made arbitrarily low by considering very complex, *ad hoc* classifiers. This is called *overfitting* [49], which is described by CART [9] as inferring classifiers that are larger than the information in the data warrant, and by ID3 [43] as increasing the classifier's complexity to accomodate a single noise-generated special case.

Weiss' LOO* estimator is motivated by empirical results indicating that the bias and precision relationships for cross-validation and bootstrapping shown in Tables 4 and 5 do not hold for single nearest neighbor (1-NN) classifiers [56], especially for small samples. These difficulties are absent or strongly mitigated in three nearest neighbors (3-NN) classifiers, suggesting that the problems are due to the extreme overfitting which is characteristic of 1-NN. This same degree of overfitting is found in CART-style decision trees when every numeric attribute cut-point is used.

Burman [17] gives the bias of k-CV as $O(p)/(k–1)N$, where p is the number of parameters estimated, as in logistic regression analysis [2]. From this result, we argue that a fairly large bias is likely for cross-validation when classifiers are overfitted (*e.g.*, $p = N$). There is a very strong analogy between 1-NN, decision trees using every numeric cut-point, and fitting a saturated[16] model.

Nearest-neighbor and decision tree methods are non-parametric, and it is difficult to quantify an equivalent of p for these methods. However, we note that saturated and over-saturated models are rote memorizers [48, 61] of their training data, as are 1-NN and unpruned decision trees with many continuous, irrelevant, or noisy redundant attributes. The classifiers are equivalent to look-up tables for the training data, and do not generalize in Schaffer's [48] and Wolpert's [61] strict sense. Kohavi [36] and Jain [33] discuss the breakdown of the 632b bootstrap in these cases — when APP is zero, as for a rote memorizer, the linear trade-off of the optimistic bias of APP and the pessimistic bias of the $\epsilon 0$ bootstrap which is implicit in 632b [24] fails, in that a slightly different classifier might also have zero APP but a higher $\epsilon 0$ value.

[16] In empirical model selection and fitting, a model is said to be saturated [2] with respect to the training data if the number of adjustable components in the model is equal to the number of training cases. Usually, such a model will reproduce its training data exactly, the data are not free to deviate from the model predictions, and the apparent error will be trivially zero. If the data are noisy (contain errors), such a model would certainly be overfitted. We will use the term over-saturated to denote a model with more adjustable components than training instances, *e.g.*, a typical back-propagation network.

Post-pruning strategies (*e.g.,* cost-complexity [9] and reduced-error [44] pruning) begin with an overfitted tree and seek a most accurate and least complex pruned version of that tree. Error rate estimates for the series of candidate trees generated during post-pruning are subject to all the difficulties of 1-NN classifiers, and to the additional difficulty that the trees and their error rates are not independent. Among other purposes, Breiman, *et al.* [9, pp. 79,307-310], adopted the heuristic expression for SE used in their 1-SE rule (in a series of pruned trees, choose the simplest tree whose 10-CV error rate is no more than 1 "standard error" (SE) greater than that of the tree having the lowest 10-CV rate) to deal with the lack of independence. Weiss & Indurkhya [57] recommend a novel form of iterated (10×) 2-fold cross-validation for cost-complexity pruning. Breiman and Spector [10] report that 10-CV is more effective than LOO for selecting a pruned tree and estimating its error rate.

Weiss' [56] results for nearest neighbors classifiers raise serious questions as to whether conclusions drawn from experiments on one type of classifier, as in our and Efron's [24] discriminant analyses, are generally applicable. Similar questions are raised by Kohavi's [36] example of the failure of LOO for a majority inducer on the Iris data[17], Crawford's [20] findings for CART that 632b is strongly biased (in our discriminant analyses, the bias is low), and by Bailey & Elkan's [6] similar findings for the symbolic concept learner FOIL. Though Crawford and Bailey & Elkan show similar results for the bias and variance of 632b, they reach different conclusions — Crawford recommends 632b because of its low variance, while Bailey & Elkan recommend against 632b because its bias is inconsistent (pessimistic when the true error is low, but optimistic when the true error is $> 30\%$) and because it has poor correlation with TER whereas 10-CV seems to correlate well with TER.

Problems such as these are, in a certain sense, inevitable given Schaffer's conservation law for generalization [48] (the 'no free lunch principle', see also [47, 62]). However, Schaffer [47], Wolpert [62], and Breiman & Spector [10] have all found that cross-validation usually performs well for model selection, *i.e.,* is fairly robust, and Schaffer [47] argues that it is fairly safe (can greatly reduce the risk of choosing a poor classifier) and reasonable when we lack problem-specific knowledge.

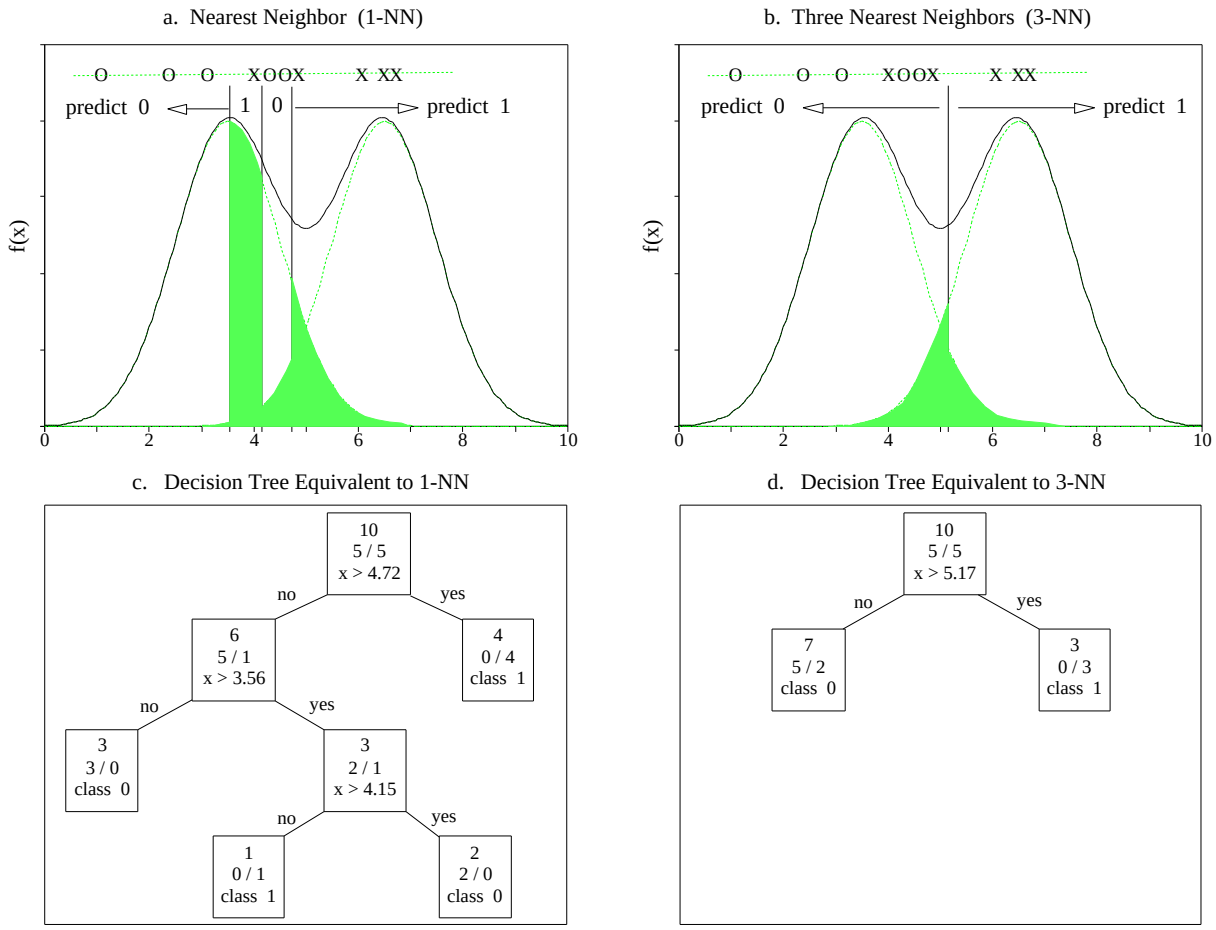
5 Experiments on Nearest-Neighbors Classifiers

In Figure 8 we illustrate nearest neighbors classifiers derived from a small sample from a population similar to those used in our discriminant examples. In Figures 8a and 8b we show the population density function and, above that, the location along the x-axis of the elements of a sample of 10 items from that population ('O' denoting a class 0 and 'X' a class 1 item) — the population and sample are the same in both cases. We also show the class predictions and boundaries, and the true error (shaded area) for two nearest neighbors classifiers, single nearest neighbor (1-NN) in Figure 8a and three nearest neighbors (3-NN) in Figure 8b.

We note the following features in the figures: (1) 3-NN may make different predictions for adjacent sample items which have the same class, as is also the case in discriminant analysis, while 1-NN cannot do this; (2) 1-NN is sensitive to outliers (isolated instances near the extremes of a class distribution), which can lead to a very high error rate for a small sample, while 3-NN smoothes these fluctuations in the sample density; and (3) 1-NN is a rote memorizer (a saturated model), and

[17] The problem generalizes to any maximum-entropy, no-information data set (one where the classes are equally frequent and independent of the attributes). Combining this with our results for TER $\geq 50\%$ in Figures 6 and 7, we conjecture that the problem is common to resampling estimators when the classes are nearly equally frequent and the estimated error near or above that of the majority inducer. As noted earlier, this has little practical significance, since classifiers with such high error rates are not useful.

Figure 8: Nearest Neighbor Classifiers and Equivalent Trees



tends to overfit the sample, *i.e.,* to infer an overly complex classifier, while 3-NN does not (it may overfit, underfit or, as in Figure 8b, be about right, depending on the sample and population).

In Figures 8c and 8d we show decision trees corresponding to the nearest neighbors classifiers of Figures 8a and 8b, respectively. Note that Figure 8d is not merely a pruned version of Figure 8c. Also note that the pair of classifiers depicted in Figures 8a and 8c are entirely equivalent, as are the pair in Figures 8b and 8d. For mutually exclusive classes, any deterministic classifier, in whatever form, can also be expressed as a decision tree or as a set of rules in disjunctive normal form (DNF), provided that the decision nodes may test arbitrary functions of several variables and noting that the translation may be non-trivial. The tree shown in Figure 8c is equivalent to that which would be inferred by CART or ID3 using the usual method of placing cut-points for continuous variables midway between adjacent items having different classes — a slightly different tree would be inferred using C4.5's [44] method of placing cut-points only at one of the values occuring in the sample. The tree shown in Figure 8d is not equivalent to that which would be inferred by CART or ID3 using the usual methods, but could be inferred by these algorithms if 3-NN's method for placing cut-points were substituted.

Thus, we must be careful not to over-generalize conclusions from experiments involving different induction algorithms, as in making assertions that X is true for decision trees, but not for nearest neighbors. As we have illustrated, decision trees and nearest neighbors are not inherently different

kinds of classifiers[18]. We emphasize that observed differences in behavior are the result of differences in the induction algorithms and estimators and interactions between them, and not due to the format in which the classifier is represented. Differences in induction algorithms may express themselves in either or both of two ways:

1. Through differences in the language (not the format) which the algorithm uses to express concepts. CART, for instance, uses DNF where the elements (individual propositions) are assertions about the value of a single attribute, whereas linear discriminant analysis utilizes a single assertion about the value of a linear function of all of the attributes. There is a very significant qualitative difference between $(p-q) < 0.5$ and $(p < 0.5) \vee [(p > 1) \wedge (q < 0.5)]$, and there are concepts which can be expressed correctly in the language used for the first example, but not the language used for the second, and *vice-versa.*

2. Through differences in the search patterns of algorithms when the language is the same. These differences are illustrated in our single real-valued attribute examples. All of these classifiers consist of a set of n cut-points and class predictions $(t_1, p_1) \cdots (t_n, p_n)$ where the prediction rule is: predict class p_i for $t_{i-1} < x \leq t_i$ where $t_0 = -\infty$ and $t_{n+1} = +\infty$. Though the language is identical, the set of potential values of the t_i for a fixed given sample differs from one algorithm to another, both in the number of cut-points allowed and in the permitted values. Linear discriminant analysis allows only one cut-point, 1-NN considers all $(x_j + x_{j-1})/2$ in the sample as potential values for t_i with a maximum $n = N-1$, and 3-NN considers all $(x_j + x_{j-3})/2$ in the sample as potential values for t_i with a maximum $n = N-3$.

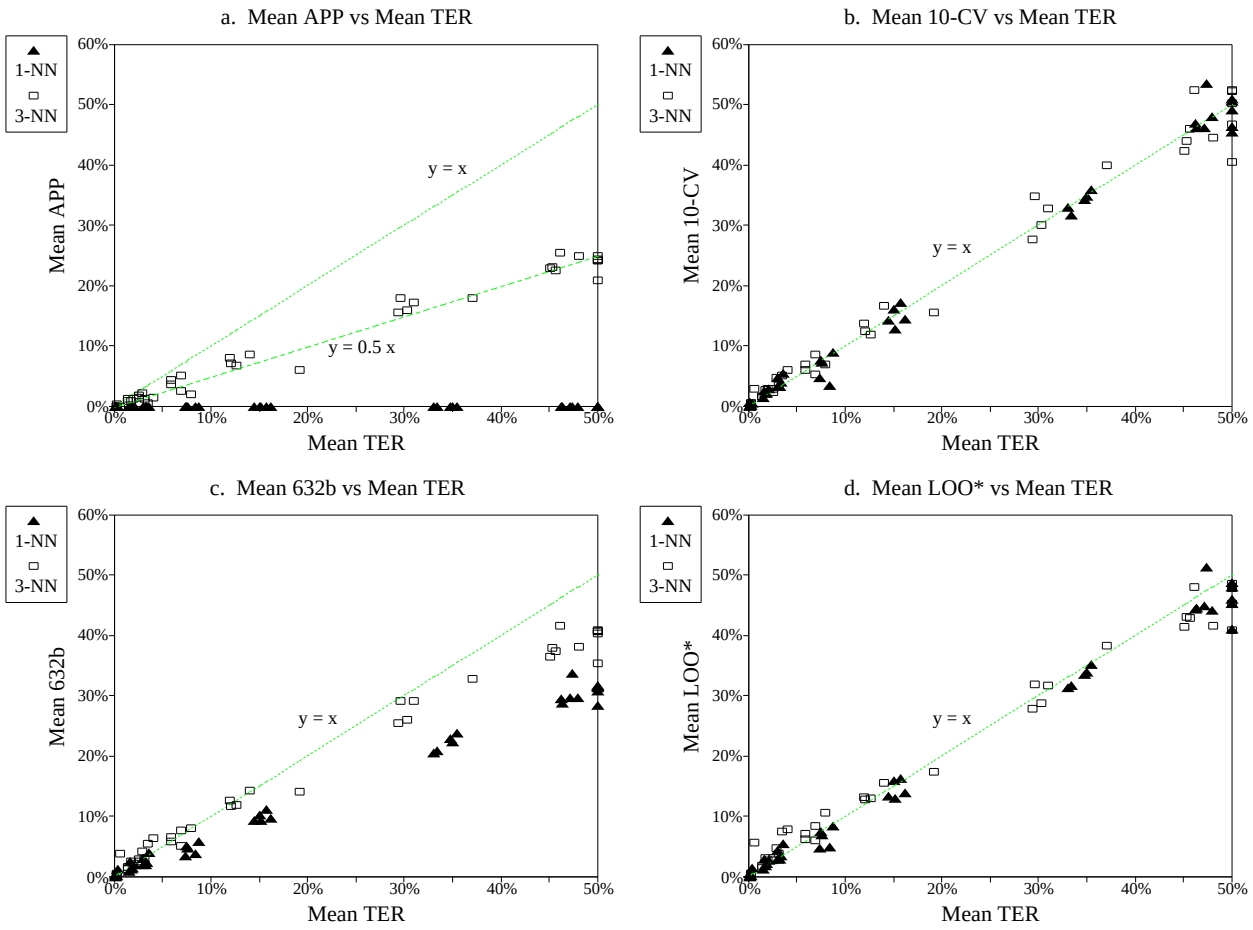
A series of experiments was conducted to explore the behavior of 1-NN and 3-NN classifier error rates for populations similar to that shown in Figure 8 — 20 random samples each of various sizes ($N = 10, 20, 30, 50, 100$) for populations with different inherent error (0.1, 1, 2, 5, 10, 25, 40, 50%). Both a 1-NN and a 3-NN classifier were calculated from each sample, and TER, APP, LOO, 10-CV, 632b, and LOO* error rates calculated for each classifier.

In Figure 9 we show the mean of each estimator plotted *vs.* the mean TER for each of the 40 experiments. LOO and 10-CV give virtually the same results, and only 10-CV is shown — both are unbiased but have high variability for both 1-NN and 3-NN. The APP and 632b results are less variable than LOO or 10-CV, but they are biased and their biases are different for 1-NN than for 3-NN. LOO* is approximately unbiased for these classifiers, but highly variable. The 632b variances are essentially the same for both 1-NN and 3-NN, and lower (by about 40%) than the LOO variances. LOO* has roughly the same precision as LOO overall, but has a lower variance for small samples and high error rates.

Detailed examination of the data shown in Figure 9 confirms Weiss' [56] findings that the lack of bias and improved precision of 632b for linear discriminants do not carry over to nearest neighbors, especially to 1-NN (see also [33, 36]). The LOO and 10-CV results, their lack of bias and relatively high variance, however, apparently do carry over. Weiss' LOO* estimator, developed for nearest neighbors, appears to be approximately unbiased for both discriminant and nearest neighbors classifiers. In our experiments, LOO* had about the same variance as 632b for discriminant classifiers, but LOO* has a higher variance than does 632b for nearest neighbors.

[18] Although decision tree algorithms may be able to deal with nominal attributes for which there are no meaningful *a priori* concepts of order or interval. We can always translate a nearest neighbors classifier into an equivalent decision tree, but the converse is not always true.

Figure 9: Mean Error Rates of 3-NN and 1-NN Classifiers



6 Experiments on Decision Tree Induction

The discriminant and nearest neighbors results were all derived from continuous attributes. To explore the behavior of resampling estimators for non-numeric attributes in other inference environments, a series of experiments was conducted using the contact lens prescription data set [42]. In this artificial problem, patients are classified into 3 categories (hard, soft, none) based on the values of 4 attributes (1 tertiary and 3 binary). The 24 instances given cover all cases and are noise free. Figure 10 shows a correct decision tree[19] for this problem (other correct trees, permuting the order of the splits, are possible — the 9 leaves are necessary and sufficient).

What is the error rate of this tree? Since the 24 sample items are the entire population and the tree classifies all 24 items correctly, the true error rate (TER) is zero. The apparent error is also zero and, thus, is not biased *in this particular case*. Leave-one-out estimates 20.8% error, and is biased in this case. The other resampling estimators are all also biased in this case (summarized by the average and standard deviation of 6 repetitions for each estimator using different train/test splits and random number seeds on each repetition) — 10-CV: 22.2% $\sigma = 3.4$;
632b: 17.6% $\sigma = 0.5$; 2-CV*: 29.5% $\sigma = 0.9$; LOO*: 20.8% $\sigma = 0$.

[19] This and other decision trees in this section were inferred using Quinlan's ID3 algorithm [43], without stopping or pruning the trees.

Figure 10: A Correct Decision Tree for Contact Lens

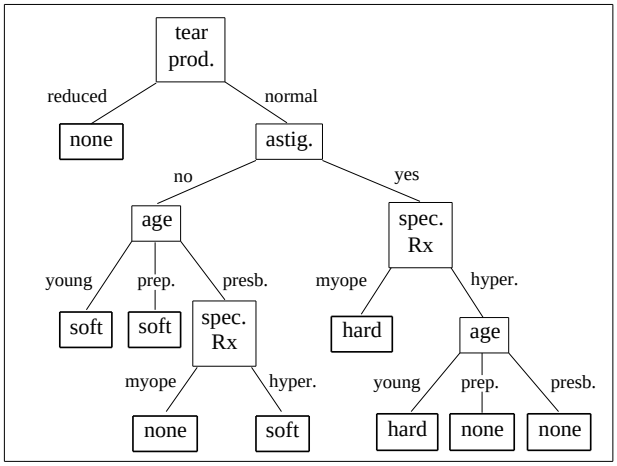


Table 11: Contact Lens Resampling Decision Trees

	Sampling with Replacement Error Rate (%)							
	$N = 24$		$N = 48$		$N = 72$		$N = 96$	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
TER	12.5	8.7	2.8	3.4	0		0	
APP	0		0		0		0	
LOO	5.6	6.8	4.2	2.9	.5	.7	.5	.6
10-CV	5.6	6.3	3.8	2.4	.9	1.7	.5	.6
632b	10.0	3.9	5.3	1.5	2.0	.4	1.1	.4
2-CV*	21.8	6.6	11.1	3.6	5.6	.6	3.2	.4
LOO*	10.6	4.6	6.1	1.4	2.0	.4	1.1	.4

It is interesting that, in this case, the apparent error is correct, while the various resampling estimates all show a strong pessimistic bias. These techniques are not applicable for this data set because two of the key assumptions underlying the methods do not hold; namely, that the data set is a random sample from a large population and that the apparent error is biased. These results underscore the important point that estimation of error rate is a statistical inference, working from a set of observations and premises (*a priori* assumptions, many of which are implicit in the methods but not explicitly stated) — the results of applying a method may be nonsensical if its premises are not satisfied.

But, suppose that the assumption that this is a random sample of 24 items from an infinite population[20] does hold and that, by chance, we happened to draw a minimal complete sample. Table 11 shows the results for random samples of various sizes from this infinite population (*i.e.*, simply sampling the 24 distinct items with replacement). Each entry in Table 11 is the average and standard deviation of 6 repeats of the experiment.

[20]That there are only 24 distinct combinations of the attributes need not limit the population size, it merely dictates that there are many duplicates. If each of the 24 combinations is equally likely, then sampling the infinite population is equivalent to sampling the 24 distinct items with replacement. They need not be equally likely, but we will assume that they are.

In each case, a tree was inferred from the entire simulated sample, and the true error rate (TER) determined by testing the inferred tree on the 24 distinct cases (*i.e.,* on the entire population). The various estimated rates were determined from the training/test splits of each sample. Since there is no noise in the data, the apparent error was zero in every case, which is biased for the smaller sample sizes, but unbiased for larger samples[21]. For samples of $N = 24$ items, 2-CV* was pessimistically biased, while all of the other estimates were optimistic. For $N \geq 48$, all of the resampling estimates were pessimistically biased in these experiments. We emphasize that these data were noise free, and that the variance of the results is purely a consequence of sampling variance.

The dimensionality of the attribute vector space (only 24 distinct items) is artificially small for this illustrative example. It does, however, demonstrate clearly that the behavior of these error rate estimators can change dramatically as a function of sample size (*e.g.,* the switch from optimistic to pessimistic bias), at least when the data are noise free.

Our simple discriminant analysis study (Table 5) showed a relatively poor correlation between estimated error and true error for repeated samples of the same size from the same population. The correlation coefficients for the six repeats in the current experiment were:

	Correlation with TER				
	LOO	10-CV	632b	2-CV*	LOO*
$N = 24$	–.47	–.63	–.76	–.74	–.64
$N = 48$	.35	.35	–.63	–.61	–.33

632b and 2-CV* appear to be more strongly correlated with true error than LOO and 10-CV. However, these correlations are negative, which is undesirable in the sense that the estimators diverge from the true error. Note that the change from optimistic to pessimistic bias for LOO and 10-CV between $N = 24$ and $N = 48$ is accompanied by a change in the sign of the correlation.

One argument for 632b and LOO* is that, though they may have a greater bias than 10-CV or LOO, they have a lower variance and may, therefore, be more powerful for distinguishing between competing classifiers. We address this question more fully in the companion paper [40], but the data in Table 11 raise some interesting points:

1. 632b and LOO* are not always more biased than LOO and 10-CV. Also note that LOO and 10-CV are optimistic for $N = 24$, but pessimistic for $N \geq 48$, while 632b and LOO* are consistently pessimistic.

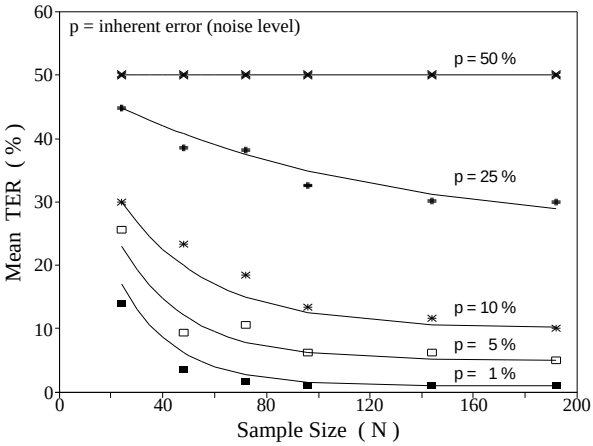
2. Minimal variance alone is not the proper criterion. APP has the least variance of any of the estimators, yet it has no ability to distinguish among the various trees (it predicts a zero error rate for every tree inferred from these data).

3. What we want is an estimator that is correlated with TER, *i.e.,* that a difference in the estimate implies a corresponding difference in TER. On this basis, the limited data in Table 11 suggest that 632b or 2-CV* might be better for smaller samples. The negative sign of the correlation is troubling, however, as this implies that, having concluded that the TER's of two classifiers are different, the one with the higher estimated error rate will actually perform better.

However, we caution that conclusions drawn from the data in Table 11 may not generalize well, since the population for these data is free of attribute or class errors (the data are correct, but

[21] A sample of $N = 24$ will contain instances of about 15 of the 24 distinct items on the average. This increases to 21 for $N = 48$ and 23 for $N = 72$. A correct tree can be inferred for many, but not all, samples of size $N \geq 72$.

Figure 12: True Error Rate *vs.* Sample Size & Noise Level



highly variable because of random resampling variation). Additional experiments were conducted, simulating attribute and class errors in a manner such that the inherent error of the population was controlled — let p be the desired inherent error, then let each of the 24 possible attribute value combinations be equally likely but let the class labels in our infinite population be randomly assigned as follows:

$$\text{class label} = \begin{cases} \text{correct label}, & \text{with probability } 1-p \\ \text{correct label } \textit{modulo } 3 \ + \ 1, & \text{with probability } p \end{cases}$$

Note that this treatment simulates attribute errors as well as class errors, since there is no way to distinguish whether a tuple such as (11111) is the result of an error in the class label or in the values of one or more attributes.

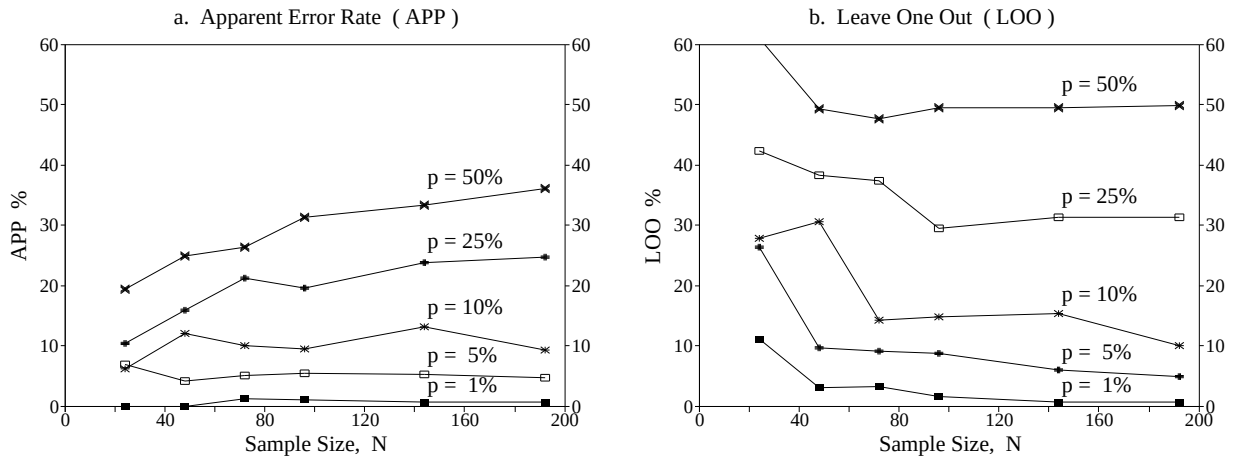
We simulated 6 samples each of several sizes from populations with different inherent error rates. In these experiments, TER is calculated using the 24 base cases, as follows (where class_i is the true class of the i^{th} case and prediction_i is the tree's prediction for the case):

$$\text{TER} = \frac{1}{24} \sum_{i=1}^{24} \begin{cases} p, & \text{if } \text{class}_i = \text{prediction}_i \\ 1-p, & \text{if } \text{class}_i \neq \text{prediction}_i \end{cases}$$

In Figure 12 we show the mean TER's of the various populations and sample sizes. TER approaches the inherent error asymptotically from above. The smooth curves shown in Figure 12 capture a general behavior which has great practical significance: (1) larger training samples tend to yield more accurate classifiers, (2) if the problem is ill-suited for the inference method, we may not be able to infer a good classifier, regardless of the sample size[22], and (3) the larger the intrinsic error of the population and inference method, the more slowly does TER approach its asymptotic value as the sample size increases (the greater the noise level or the more ill-suited the problem and inference method, the greater the sample size required to achieve a near-asymptotic error). Similar curves for the means of APP and LOO are shown in Figure 13. Note that APP approaches its asymptotic level from below, while TER, LOO, and the other resampling estimates approach from above.

[22] TER is bounded below by the inherent error. In most cases, TER is limited by the language-intrinsic error (see Section 2, Figure 1). Here, the inherent and language-intrinsic error rates are the same.

Figure 13: Estimators *vs.* Sample Size & Noise Level



The relationship of the various estimators' means to the mean true error for various noise levels and sample sizes in these experiments is shown in Figure 14. LOO and 10-CV are apparently unbiased and have about the same precision. 632b is pessimistically biased for low ($< 10\%$) error rates, and optimistically biased for higher error rates, as was also reported by Bailey & Elkan [6]. The standard deviation (vertical spread) of 632b is lower than that of LOO or 10-CV. For these decision trees, 2-CV* and LOO* are biased and highly variable.

The charts in Figure 14 show a fairly strong positive correlation between the various estimators' means and the mean TER. However, there is no such correlation of the individual estimates and TER's within a replicated experiment, as shown in Table 15 — though the estimators correlate with one another, they do not correlate well with TER and the weak correlation with TER appears to be negative for small samples and low noise levels. An additional set of experiments was conducted to verify these results by simulating 200 samples each of sizes 24 and 96 from a population with an inherent error of $p = 0.01$ (also summarized in Table 15 as the linear regression coefficients, with r denoting the correlation). The weak correlations of 10-CV *vs.* TER and 632b *vs.* TER are not significant, while the correlation of 632b *vs.* 10-CV is significant.

Thus, it appears that the expected value $\overline{\text{EST}}$ of repeated sampling for any of our estimators is correlated with the expected true error $\overline{\text{TER}}$ for the trees inferred from these samples, *i.e.*, $\overline{\text{EST}} \approx k_0 + k_1 \overline{\text{TER}}$, where $k_0 = 0$ and $k_1 = 1$ would be an unbiased estimator. However, it also appears that, for the i^{th} individual estimate EST_i, the difference $\Delta_i = \text{EST}_i - \overline{\text{EST}}$ is a random variable, and that Δ_i is independent of the random variable $\delta_i = \text{TER}_i - \overline{\text{TER}}$ which is of interest (*i.e.*, $E(\Delta_i \delta_i) = 0$). This means that, for sample i, EST_i might be above average and TER_i below average, while for another sample j of the same size from the same population, EST_j might be below average and TER_j above average. This has important consequences regarding the significance of observed differences between estimates, which are explored in the companion paper [40].

In these experiments, the time T required to infer a tree increased with increasing sample size N or tree complexity η (the number of nodes in the inferred tree), $T \approx k_0 + (k_1 + k_2\eta)N$. The tree complexity increased with increasing sample size or noise level, nearing saturation ($\eta = 46$, or 24 leaves) for the most noisy data and largest sample. In most situations, η is limited by the sample size ($\eta < 2N$, if empty leaves are forbidden), rather than by the saturation level (the number of distinct possible attribute-value vectors) which either grows exponentially with the number of

Figure 14: Mean Estimated Error *vs.* Mean True Error

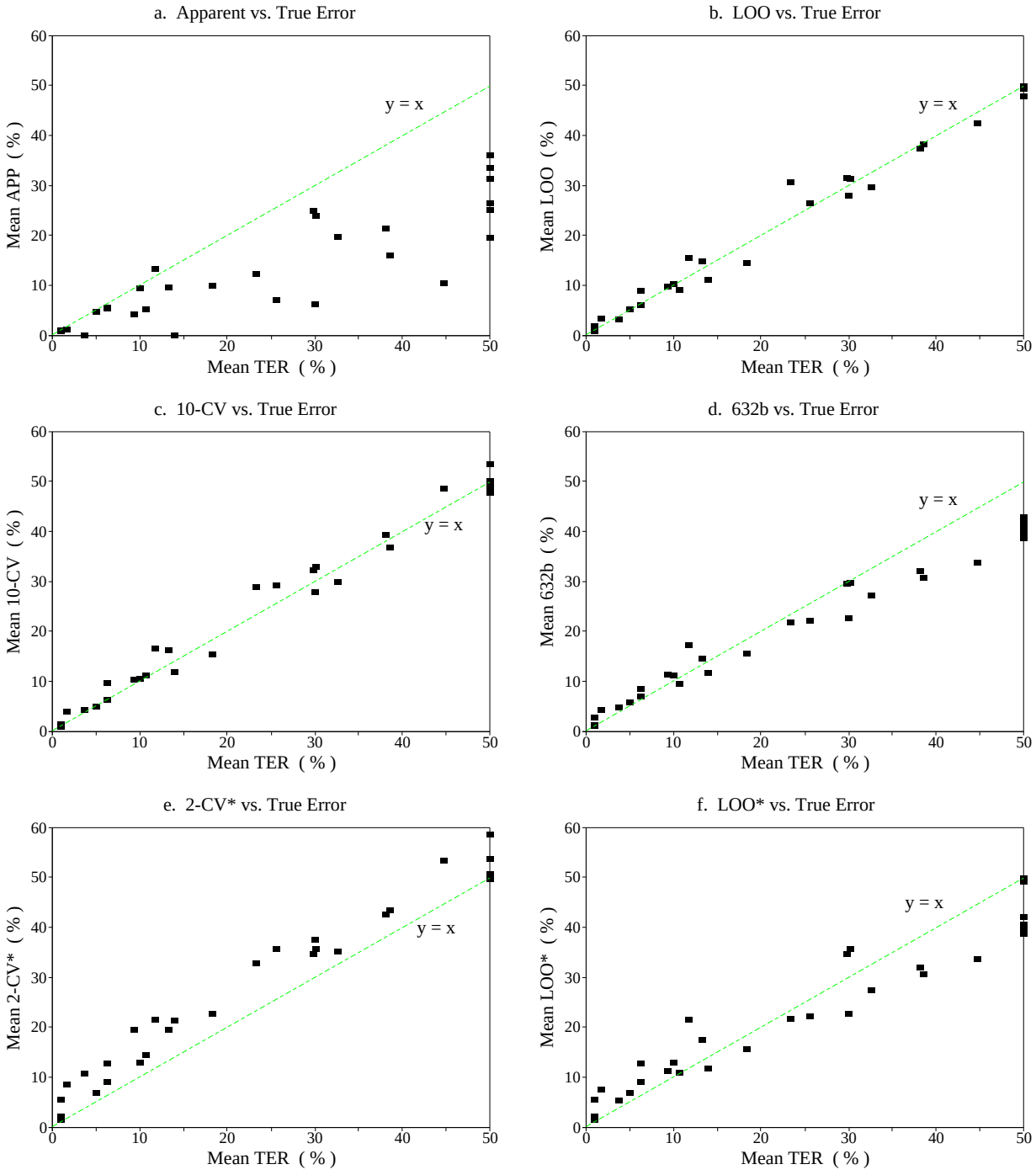


Table 15: Correlation of Repeated Sample Estimates

Six Samples of Each Size and Inherent Error

$p =$	0.01	0.05	0.10	0.25	0.01	0.05	0.10	0.25	0.01	0.05	0.10	0.25
	10-CV				632b				LOO*			
	Correlation with TER											
$N = 24$	−.65	.54	.69	−.06	−.38	.47	.75	.17	−.38	.47	.75	.17
48	−.77	.55	.84	.49	−.68	.45	.89	.45	−.74	.45	.89	.45
72	.88	.35	.83	−.17	.75	.11	.48	.11	.58	.07	.48	.11
96		.44		.21		.55		.40		.58		.37
144		.49	.44	.11		.53	.34	.21		.49	.33	.18
192				.18				.08				.11
	Correlation with 10-CV											
$N = 24$					.70	.99	.92	.93	.70	.99	.92	.93
48					.72	.90	.98	.92	.68	.90	.98	.92
72					.95	.87	.80	.76	.80	.87	.80	.76
96					.54	.83	.98	.87	.53	.80	.93	.85
144					.99	.93	.85	.98	.96	.94	.81	.98
192					.99	.99	.93	.96	.94	.99	.92	.95

p is the population inherent error

Correlation of 200 Samples' Estimates for $p = 0.01$

$N = 24$		$N = 96$	
10-CV = 15.6 − 0.19(TER)	$r = -.19$	10-CV = 2.0 + 0.06(TER)	$r = +.05$
632b = 14.0 − 0.16(TER)	$r = -.28$	632b = 2.7 + 0.03(TER)	$r = +.03$
632b = 6.6 + 0.42(10-CV)	$r = +.78$	632b = 1.4 + 0.66(10-CV)	$r = +.80$

attributes or is in principle[23] unbounded for continuous attributes. This $O(N^2)$ time bound can easily be achieved for noisy data when there are many irrelevant or redundant attributes. For continuous attributes the bound is $O(N^2 \lg N)$ due to sorting costs [44], and it is readily achieved for noisy data when the trees are not pre-pruned (stopped). Since this limit corresponds to a tree with N leaves (a saturated model), such a tree is certainly overfitted if the data are noisy.

This potentially large cost (and the cost of pruning has not been taken into account) may be a matter of concern as problem domains are expanded beyond the current, relatively simple, data sets to large, real-world databases with scores of noisy attributes and thousands of instances, especially for the iterated and bootstrap methods, which must infer several hundred classifiers for each sample. Techniques for updating trees rather than iterating the entire process, such as ITI trees [52], can reduce the time required in such cases, but their increased storage costs may become prohibitive.

7 Conclusions and Recommendations

We remind readers that specific simulation results from very simple populations such as those used in our experiments may not generalize to more complex situations, and that no single method for estimating classifier error rates will perform best in every situation. Those caveats notwithstanding, we conclude and recommend as follows:

1. Leave-one-out (LOO) and 10-fold cross-validation (10-CV) were the only estimators that uniformly had little or no bias in our experiments. 10-CV appears to be the safest method for estimating classifier error rates. Its bias is usually small, and its precision appears to be equivalent to that of LOO, at lower cost. In addition, Breiman and Spector [10] report that 10-CV is more effective than LOO for pruning.

2. The single independent subsamples (ISS) method results in a classifier with poorer expected accuracy and significantly greater variance than 10-CV.

3. Iterating k-fold cross-validation reduces its variance, but the effect is small for $k \geq 10$. Cross-validation is pessimistically biased for $k < 10$, and iteration does not affect the bias.

4. The 632b bootstrap has lower variance than 10-CV (its standard deviation averages 80% that of 10-CV, but at greater computational cost). However, its bias may be different for different learning algorithms, and according to whether a classifier is overfitted. For that reason, 632b may not be suitable for comparing 1-NN and 3-NN classifiers, nor for comparing pruned and unpruned decision trees.

5. LOO* was approximately unbiased for discriminant functions and nearest neighbors, and had lower variance than LOO or 10-CV for these classifiers. This lack of bias and improved precision did not carry over to nominal attribute decision trees, and LOO* is not recommended for those applications.

6. Extreme overfitting, as in the 1-NN classifier and unpruned decision trees, can affect both the bias and precision of cross-validation and bootstrapping. More complex methods may be necessary when classifiers are overfitted.

[23] Real measurements are always discrete (finite resolution) and have a bounded range — the statistician's practical distinction between discrete and continuous attributes is one between variables which have only a few possible values and those which have a great many possible values.

7. The estimated error rates of classifiers inferred from different random samples of the same size from the same population are poorly correlated (and sometimes negatively correlated) with the classifiers' true error rates. This appears to be true for all of the resampling estimators, regardless of the inference method, sample size, or population inherent error. This means that small differences in estimated error rates, such as might be expected when incrementally pruning a decision tree, may have little or nothing to do with the difference in the true error rates of the classifiers being compared. The magnitudes and even the signs of the differences may be different for different samples or even for a different choice of training/test splits of a single sample.

8 Acknowledgement

The authors are indebted to the editors and reviewers of two earlier versions of these papers. Their suggestions have been invaluable in correcting many of the weaknesses and oversights.

References

[1] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions*. Dover, New York, 1972.

[2] A. Agresti. *Categorical Data Analysis*. Wiley, New York, 1990.

[3] D. W. Aha. *A Study of Instance-Based Learning Algorithms for Supervised Learning Tasks: Mathematical, Empirical, and Psychological Evaluations*. PhD thesis, University of California, Irvine, 1990.

[4] D. W. Aha. Generalizing from case studies: A case study. In *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI-92)*, pages 1–10, Cambridge, MA, 1992. MIT Press.

[5] E. Andersen. The Irises of the Gaspé Peninsula. *Bulletin of the American Iris Society*, 59:2–5, 1935. (cited by Morrison [41, p 468]).

[6] T. L. Bailey and C. Elkan. Estimating the accuracy of learned concepts. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI-93)*, volume 2, pages 895–900, San Mateo, CA, 1993. Morgan Kaufmann.

[7] L. Breiman. Bagging predictors. Technical Report 421, Dept. of Statistics, University of California, Berkeley, 1994.

[8] L. Breiman. Heuristics of instability and stabilization in model selection. Technical Report 416, Dept. of Statistics, University of California, Berkeley, 1994.

[9] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth & Brooks, Pacific Grove, CA, 1984.

[10] L. Breiman and P. Spector. Submodel selection and evaluation in regression. The X-random case. *International Statistical Review*, 60:291–319, 1992.

[11] C. E. Brodley. Recursive automatic bias selection for classifier induction. *Machine Learning*, 20:63–94, 1995.

[12] W. Buntine. Decision tree induction systems: A Bayesian analysis. In L. N. Kanal, T. S. Levitt, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence, 3*, pages 109–127, Amsterdam, 1989. North-Holland.

[13] W. Buntine. Learning classification rules using Bayes. In *Proceedings Sixth International Workshop on Machine Learning*, pages 94–98, San Mateo, CA, 1989. Morgan Kauffman.

[14] W. Buntine. Myths and legends in learning classification rules. In *Proceedings of the 8th National Conference on Artificial Intelligence (AAAI-90)*, pages 736–742, Menlo Park, CA, 1990. AAAI Press.

[15] W. Buntine. Classifiers: A theoretical and empirical study. In *Proceedings 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 638–644, San Mateo, CA, 1991. Morgan Kauffman.

[16] W. L. Buntine. *A Theory of Learning Classification Rules*. PhD thesis, University of Technology, Sydney, 1990.

[17] P. Burman. Estimation of optimal transformations using v-fold cross validation and repeated learning-testing methods. *Sankhya: The Indian Journal of Statistics*, 22A:314–345, 1990.

[18] W. G. Cochran. The χ^2 test of goodness of fit. *Annals of Mathematical Statistics*, 23:315–345, 1950.

[19] W. G. Cochran. Some methods of strengthening the common χ^2 tests. *Biometrics*, 10:417–451, 1952.

[20] S. L. Crawford. Extensions to the CART algorithm. *International Journal of Man-Machine Studies*, 31:197–217, 1989.

[21] A. C. Davison and P. Hall. On the bias and variability of bootstrap and cross-validation estimates of error rate in discrimination problems. *Biometrika*, 79:279–284, 1992.

[22] P. Dierckx. *Curve and Surface Fitting with Splines*. Oxford University Press, New York, 1993.

[23] B. Efron. Computers and the theory of statistics: Thinking the unthinkable. *SIAM Review*, 21:460–480, 1979.

[24] B. Efron. Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, 78:316–331, 1983.

[25] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability, Series No. 57. Chapman & Hall, New York, 1993.

[26] C. Feng, R. King, A. Sutherland, S. Muggleton, and R. Henery. Symbolic classifiers: Conditions to have good accuracy performance. In P. Cheeseman and R. W. Oldford, editors, *Artificial Intelligence and Statistics IV: Selecting Models from Data*, volume 89 of *Lecture Notes in Statistics*, pages 371–380. Springer, New York, 1994.

[27] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7, part II:179–188, 1936. (the frequently cited Iris data analysis).

[28] R. A. Fisher. *Statistical Methods for Research Workers*. Oliver & Boyd, Edinburgh, 14th edition, 1970. (the quotation is from the preface to the first (1925) edition).

[29] G. M. Fitzmaurice, W. J. Krzanowski, and D. J. Hand. A Monte Carlo study of the 632 bootstrap estimator of error rate. *Journal of Classification*, 8:239–250, 1991.

[30] R. M. Goodman and P. Smyth. Information-theoretic rule induction. In *Proceedings of the 8th European Conference on Artificial Intelligence (ECAI-88)*, pages 357–362, London, 1988. Pitman.

[31] L. Gordon and R. A. Olshen. Asymptotically efficient solution to the classification problem. *Annals of Statistics*, 6:515–533, 1978.

[32] L. Gordon and R. A. Olshen. Almost surely consistent nonparametric regression from recursive

partitioning schemes. *Journal of Multivariate Analysis*, 15:147–163, 1984.

[33] A. K. Jain, R. C. Dubes, and C. Chen. Bootstrap techniques for error estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9:628–633, 1987.

[34] M. James. *Classification Algorithms*. W. M. Collins & Sons, London, 1985.

[35] M. I. Jordan. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6:181–214, 1994.

[36] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI-95)*, pages 1137–1143, San Mateo, CA, 1995. Morgan Kaufmann.

[37] I. Kononenko. Semi-naive Bayesian classifier. In *Proceedings of the European Working Session on Learning (EWSL-91)*, pages 206–218, Berlin, 1991. Springer.

[38] S. W. Kwok and C. Carter. Multiple decision trees. In R. D. Schacter, T. S. Levitt, L. N. Kanal, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence, 4*, pages 327–335, Amsterdam, 1990. North-Holland.

[39] P. Langley, W. Iba, and K. Thompson. An analysis of Bayesian classifiers. In *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI-92)*, pages 223–228, Cambridge, MA, 1992. MIT Press.

[40] J. K. Martin and D. S. Hirschberg. Small sample statistics for classification error rates, II: confidence intervals and significance tests. Technical Report 96-22, University of California, Irvine, 1996.

[41] D. F. Morrison. *Multivariate Statistical Methods*. McGraw-Hill, New York, 3rd edition, 1980.

[42] P. M. Murphy and D. W. Aha. *UCI Repository of Machine Learning Databases*. Dept. of Information and Computer Science, University of California, Irvine. (machine-readable data depository).

[43] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[44] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.

[45] C. Reinsch. Smoothing by spline functions. *Numerische Mathematik*, 10:177–183, 1967. (cited by Dierckx [22]).

[46] C. Schaffer. Overfitting avoidance as bias. *Machine Learning*, 10:153–178, 1993.

[47] C. Schaffer. Selecting a classification method by cross-validation. *Machine Learning*, 13:135–143, 1993.

[48] C. Schaffer. A conservation law for generalization performance. In *Machine Learning: Proceedings of the 11th International Conference (ML-94)*, pages 259–265, San Francisco, 1994. Morgan Kaufmann.

[49] J. W. Shavlik and T. G. Dietterich, editors. *Readings in Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1990.

[50] J. W. Shavlik, R. J. Mooney, and G. G. Towell. Symbolic and neural learning algorithms: An experimental comparison. *Machine Learning*, 6:111–143, 1991.

[51] R. Tibshirani. Bias, variance and prediction error for classification rules. Technical report, Department of Statistics, University of Toronto, 1996.

[52] P. E. Utgoff. An improved algorithm for incremental induction of decision trees. In *Machine Learning: Proceedings of the 11th International Conference (ML-94)*, pages 318–325,

San Francisco, 1994. Morgan Kaufmann.

[53] V. Vapnik. Principles of risk minimization for learning theory. *Advances In Neural Information Processing Sytems*, 4:831–838, 1992.

[54] V. Vapnik, E. Levin, and Y. LeCun. Measuring the VC-dimension of a learning machine. *Neural Computation*, 6:881–876, 1994.

[55] V. N. Vapnik and A. Y. Chervonekis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16:264–280, 1971.

[56] S. M. Weiss. Small sample error rate estimation for k-nearest neighbor classifers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:285–289, 1991.

[57] S. M. Weiss and N. Indurkhya. Small sample decision tree pruning. In *Proceedings of the 11th International Conference on Machine Learning (ML-94)*, pages 335–342, San Francisco, 1994. Morgan-Kaufman.

[58] S. M. Weiss and C. A. Kulikowski. *Computer Systems that Learn: Classification and Prediction Methods From Statistics, Neural Nets, Machine Learning, and Expert Systems*. Morgan Kaufmann, San Mateo, CA, 1991.

[59] D. H. Wolpert. A mathematical theory of generalization: Part I & Part II. *Complex Systems*, 4:151–249, 1990.

[60] D. H. Wolpert. The relationship between Occam's Razor and convergent guessing. *Complex Systems*, 4:319–368, 1990.

[61] D. H. Wolpert. On the connection between in-sample testing and generalization error. *Complex Systems*, 6:47–94, 1992.

[62] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.

[63] D. H. Wolpert and W. G. Macready. An efficient method to estimate bagging's generalization error. Technical Report SFI-TR-96-03, Santa Fe Institute, Santa Fe, NM, 1996.

A Appendix: Analysis of the Class Reversal Problem

Given that the population is a mixture of two classes labeled 0 and 1, in unknown proportions and that each class is normally distributed on a single real-valued attribute x with possibly different class means (μ_0^* and μ_1^*, respectively) but a common standard deviation, σ^*:

Let p^* denote the fraction of class 0 items in the population and $C(t, a, b)$ a classifier: predict class a if $x \leq t$, else predict class b. Without loss of generality, we assume that $\mu_0^* \leq \mu_1^*$ and $\sigma^* = 1$.

For a size N sample containing n class 0 items, our linear discriminant inference function is:

```
InferClassifier(x, class, N, n)
    if n = 0 then return C(−∞, 0, 1);
    if n = N then return C(+∞, 0, 1);
    p  ←  n / N;
    x̄0  ←  Σ{x[i] : class[i] = 0} / n;
    x̄1  ←  Σ{x[i] : class[i] = 1} / (N−n);
    s²  ←  (Σ x[i]² − n x̄0² − (N−n) x̄1²)  / (N−2);
    t  ←  (x̄0 + x̄1) / 2;
    if x̄0 = x̄1 then
        if N = 2n then return C(t, 0, 1);
        if n < N/2 then return C(−∞, 0, 1);
        else return C(+∞, 0, 1)
    Δt  ←  s² ln[n/(N−n)] / (x̄1 − x̄0);
    t  ←  t + Δt;
    z0  ←  (t − 0.1 − x̄0) / s;        (* evaluate both classes to   *)
    z1  ←  (t − 0.1 − x̄1) / s;        (* the left of the threshold t *)
    if p exp(−z0²/2)  ≥  (1−p) exp(−z1²/2)
    then return C(t, 0, 1);
    else return C(t, 1, 0);                (*** a class reversal ***)
```

The model (probability density function, $f(\cdot)$) we are using here is:

$$f(x, p^*, \mu_0^*, \mu_1^*, \sigma^*) \;=\; p^*\; g(x, \mu_0^*, \sigma^*) \;+\; (1-p^*)\; g(x, \mu_1^*, \sigma^*)$$

where

$$g(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}}\; \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$$

The true values of the parameters ($p^*,\ \mu_0^*,\ \mu_1^*,$ and σ^*) are unknown, and we substitute the empirical estimates ($p^* \approx n/N$, $\mu_0^* \approx \bar{x}_0$, $\mu_1^* \approx \bar{x}_1$, $\sigma^* \approx s$) when they exist, where $\bar{x}_0$ and $\bar{x}_1$ are the sample means for classes 0 and 1, respectively, s is the 'pooled' estimate of σ^*,

$$s^2 = \frac{n\, s_0^2 + (N-n)\, s_1^2}{N-2}$$

and s_0^2 and s_1^2 are the sample variances for classes 0 and 1, respectively.

The true error, $\tau(t,0,1)$, of classifier $C(t,0,1)$ is

$$\tau(t,0,1) \;=\; p^* \int_t^{+\infty} g(x,\mu_0^*,\sigma^*)dx \;+\; (1-p^*)\int_{-\infty}^{t} g(x,\mu_1^*,\sigma^*)dx$$

Equating the derivative $d\tau(t,0,1)/dt$ to zero at $t = T$, the population's least-error threshold T (corresponding to the inherent error) is the solution of $p^*\; g(T,\mu_0^*,\sigma^*) \;=\; (1-p^*)\; g(T,\mu_1^*,\sigma^*)$. The empirical threshold estimate t is found by solving $p\; g(t,\bar{x}_0,s) \;=\; (1-p)\; g(t,\bar{x}_1,s)$.

A class reversal occurs if

$$p\; g(x,\bar{x}_0,s) \;<\; (1-p)\; g(x,\bar{x}_1,s) \quad \text{for } x < t$$

In our inference function on the previous page, we evaluate this condition at $x = t - 0.1$; subject to the limitations of numeric precision, any $x < t$ would do as well as $t - 0.1$ because the bell-shaped curves cross at most once.

The true error, $\tau(t,1,0)$, of the reversed classifier $C(t,1,0)$ is

$$\tau(t,1,0) \;=\; p^* \int_{-\infty}^{t} g(x,\mu_0^*,\sigma^*)dx \;+\; (1-p^*)\int_t^{+\infty} g(x,\mu_1^*,\sigma^*)dx \;=\; 1-\tau(t,0,1)$$

Restricting our attention to the case $p^* = 0.5, \sigma^* = 1$,

$$\begin{aligned}
\tau(t,0,1) &= 0.5\,[1-\Phi(t-\mu_0^*)+\Phi(t-\mu_1^*)] \\
\tau(t,1,0) &= 0.5\,[1-\Phi(t-\mu_1^*)+\Phi(t-\mu_0^*)] \;=\; 1-\tau(t,0,1) \\
\tau(T,0,1) &= \text{inherent error} \;=\; 1-\Phi(\frac{\mu_1^*-\mu_0^*}{2})
\end{aligned}$$

where $T = (\mu_0^* + \mu_1^*)/2$, $\Phi(z) = \int_{-\infty}^{z} g(x,0,1)dx$ is the cumulative standard (zero mean, unity variance) normal distribution, and $\Phi(-z) = 1 - \Phi(z)$.

The threshold, t, is a random variable, a function of the random variables p, $\bar{x}_0$, $\bar{x}_1$, and s:

$$t(N,n,\bar{x}_0,\bar{x}_1,s) = \frac{\bar{x}_0+\bar{x}_1}{2} + \frac{s^2\ln(p/(1-p))}{\bar{x}_1-\bar{x}_0} \tag{1}$$

This function is moderately complex, and its probability distribution is extremely complex.

Since t is random, the error of a reversed classifier, $\tau(t,1,0)$ is also random and, owing to the complexity of $t(\cdot)$ and $\Phi(\cdot)$, the distribution of $\tau(t,1,0)$ is extremely complex. We have not attempted an exact solution, but instead relied on Monte Carlo techniques.

In the experiments in Section 3, we generated N simulated observations for every sample. Here, where we are not resampling, we make use of the following sampling distributions for the empirical parameters to speed the simulations:

- n is binomial$(0.5, N)$.
- For a given n: $\bar{x}_0$ is normal$(\mu_0^*, \sigma^*/\sqrt{n}\,)$, and $\bar{x}_1$ is normal$(\mu_1^*, \sigma^*/\sqrt{N-n}\,)$
- $(N-2)s^2$ is chi-squared$(N-2)$.

Figure 16: Distribution of True Error Rate

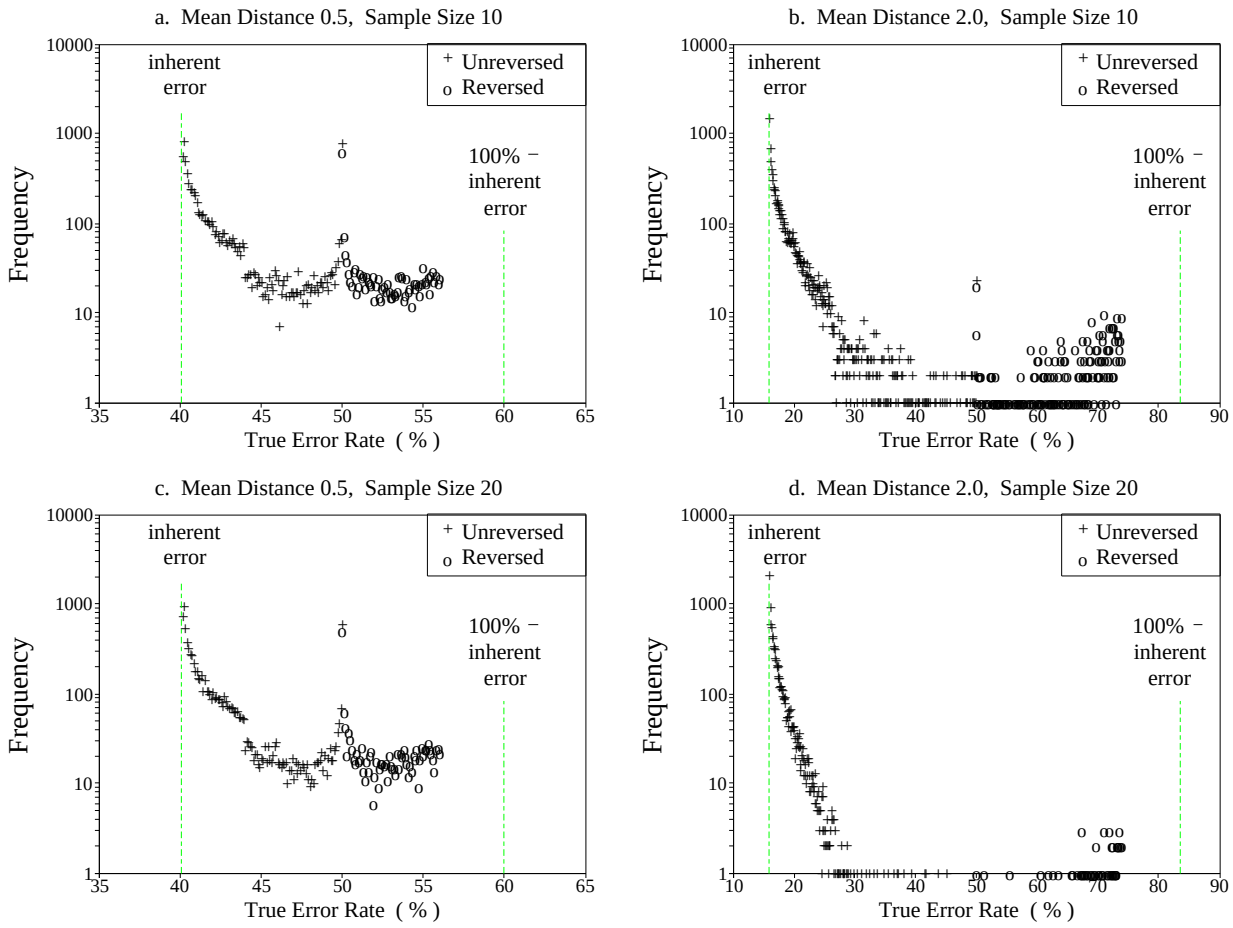
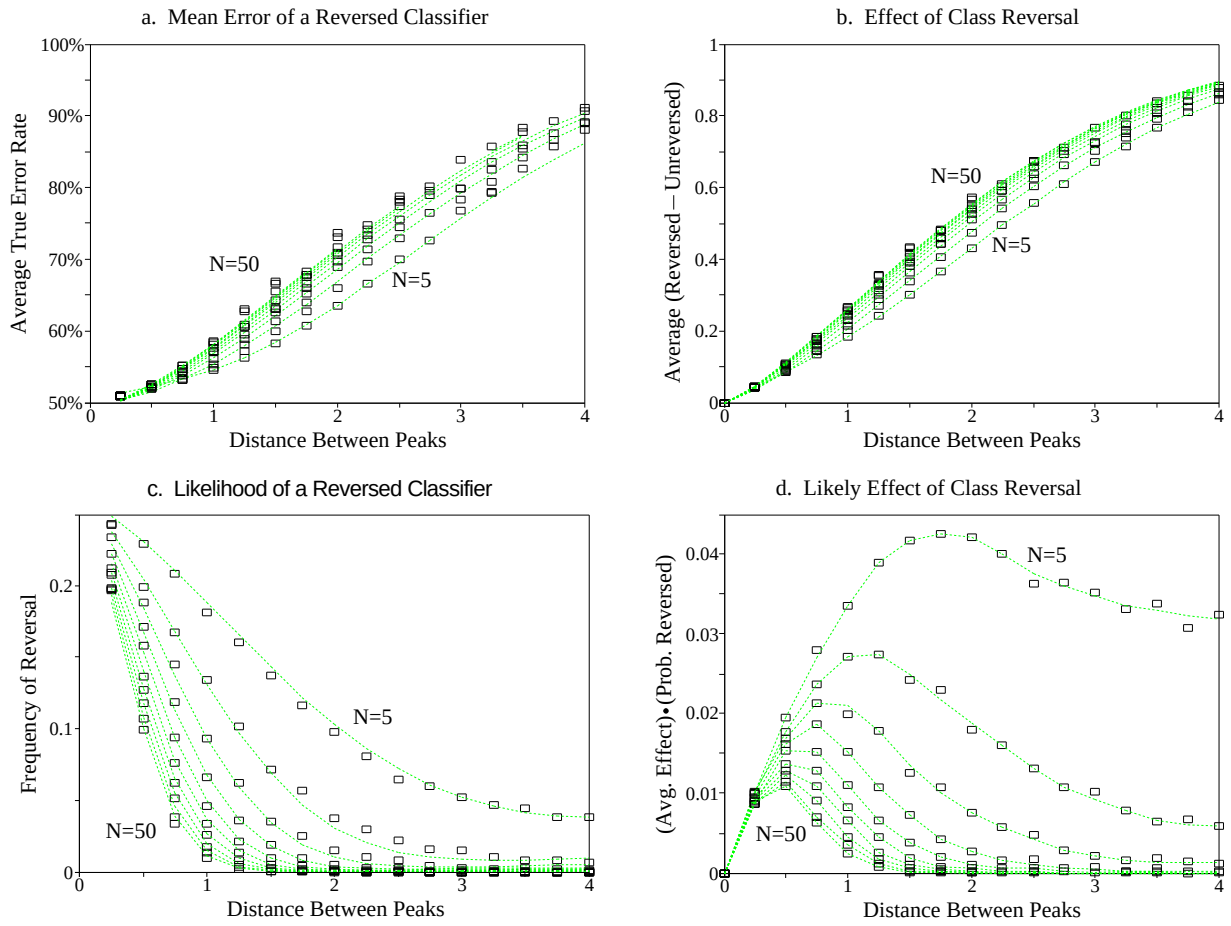


Figure 16 shows the distribution of the true error rates of 10,000 simulated samples for each of two sample sizes ($N = 10, 20$) and two mean distances ($\mu_1 - \mu_0 = 0.5, 2.0$). Only those samples having at least one instance of each class were included. For the unreversed classifiers, the true error tends to cluster slightly above the inherent error rate, with a fairly long tail up to a 'spike' at 50%[24]. For the reversed classifiers, the frequencies are much lower and the distribution is a rough mirror image of the tail of the unreversed distribution. The reversed cases are less frequent as either the sample size or the distance between the class means increases.

Only the tail of the distribution is likely to be reversed. In order for the reversed error rate to be near 100% − inherent error, the threshold t must be near its optimal position $T = (\mu_0^* + \mu_1^*)/2$. From the formula (Equation 1) for t, this would require a relatively unlikely combination of the empirical values for p, $\bar{x}_0$, and $\bar{x}_1$ in addition to the reversal of the means: either (1) $p \approx 0.5$ and $\bar{x}_0$ and $\bar{x}_1$ are approximately equidistant from T, or (2) $s^2 \ln(p/(1-p))/(\bar{x}_1 - \bar{x}_0)$ almost exactly offsets the distance from $(\bar{x}_0 + \bar{x}_1)/2$ to T. The reversal is far more likely when the empirical proportion p is very unbalanced, as this leads to an imprecise estimate of either $\bar{x}_0$ or $\bar{x}_1$. Other things being equal, imbalance in p also corresponds to the threshold t being relatively far from T, *i.e.,* to the tail of the distribution.

[24] The y-axes in Figure 16 are logarithmic, so that the tail is clear; the clustering near the inherent error rate would be even more apparent on a linear scale.

Figure 17: Summary of Reversed Classifiers



In a more thorough experiment, we simulated 10,000 samples each for 10 different sample sizes ($N = 5i, i = 1 \ldots 10$) and 16 distances between the class means ($\mu_1 - \mu_0 = 0.25i, i = 1 \ldots 16$). The results of these experiments are summarized in Figure 17. In Figure 17a, we show the mean true error of the reversed classifiers as a function of sample size and class mean separation. In Figure 17b, we show the mean difference between the true errors of the reversed and unreversed classifiers. In Figure 17c, we show the likelihood that a reversal occurs. And in Figure 17d, we show the expected increase in the overall mean true error due to the reversals (*i.e.*, the product of the curves in Figures 17b and 17c).

The mean true error in Figure 17a is well fit by the model

$$\text{avg. true error} = 1 - I - (k_0 + k_1/N)(0.5 - I)^{k_2 + k_3/N}\sqrt{I}$$

where I is the inherent error of the population, $k_0 \approx 0.6$, $k_1 \approx 6$, $k_2 \approx 0.75$, and $k_3 \approx 2$. That is, the true error of a reversed classifier appears to be $1 - I - O(N^{-1})$.

The mean difference in Figure 17b is well fit by a similar model

$$\text{avg. difference} = 1 - 2I - (k_0 + k_1/N)(0.5 - I)^{k_2 + k_3/N}\sqrt{I}$$

where $k_0 \approx 0.6, k_1 \approx 9.5, k_2 \approx 0.7$, and $k_3 \approx 2$. That is, the difference appears to be $1 - 2I - O(N^{-1})$.

The likelihood in Figure 17c is well fit by the model

$$\text{likelihood} = k_0 \exp(-k_1 z - k_2 z^2) + \frac{k_3}{(N-2)^2} \left(\frac{z}{1+z} \right)^{k_4}$$

where $z = (\mu_1 - \mu_0)\sqrt{N-2}$, $k_0 \approx 0.26$, $k_1 \approx 0.1$, $k_2 \approx 0.05$, $k_3 \approx 2.7$, and $k_4 \approx 18$.

We note that the coefficient k_0 in the likelihood model is the average likelihood when $\mu_1 = \mu_0$, and differs from 0.5 because the sample proportions of the two classes vary. When the sample proportions are 1:1, the likelihood at $\mu_1 = \mu_0$ is 50%, but when the sample proportions are 49:1, the likelihood is near zero. k_0 is the average over the various sample proportions.

The second term in the likelihood model is $O(N^{-2})$, and is insignificant for $N \geq 20$. We attach no particular significance to the $[z/(1+z)]^{k_4}$ form; other increasing, asymptotically flat functions of z fit these data about equally well.

Note that the product in Figure 17d is nearly independent of the sample size at $\mu_1 - \mu_0 = 0.25$. As a consequence, the location of the peak in these plots tends to converge to a point between $\mu_1 - \mu_0 = 0.25$ and $\mu_1 - \mu_0 = 0.5$ (40-45% inherent error) as N increases. The location of the peak and the height of the peak change very slowly for $N > 25$.