The Case Against Accuracy Estimation for Comparing Induction Algorithms

Foster Provost
Bell Atlantic Science and Tech
400 Westchester Avenue
White Plains, NY 10604
foster@basit.com

Tom Fawcett
Bell Atlantic Science and Tech
400 Westchester Avenue
White Plains, NY 10604
fawcett@basit.com

Ron Kohavi
Silicon Graphics Inc. M/S 8U-876
2011 N. Shoreline Blvd.
Mountain View, CA 94043
ronnyk@sgi.com

Abstract

We analyze critically the use of classification accuracy to compare classifiers on natural data sets, providing a thorough investigation using ROC analysis, standard machine learning algorithms, and standard benchmark data sets. The results raise serious concerns about the use of accuracy for comparing classifiers and draw into question the conclusions that can be drawn from such studies. In the course of the presentation, we describe and demonstrate what we believe to be the proper use of ROC analysis for comparative studies in machine learning research. We argue that this methodology is preferable both for making practical choices and for drawing scientific conclusions.

1 INTRODUCTION

Substantial research has been devoted to the development and analysis of algorithms for building classifiers, and a necessary part of this research involves comparing induction algorithms. A common methodology for such evaluations is to perform statistical comparisons of the accuracies of learned classifiers on suites of benchmark data sets. Our purpose is not to question the statistical tests (Dietterich, 1998; Salzberg, 1997), but to question the use of accuracy estimation itself. We believe that since this is one of the primary scientific methodologies of our field, it is important that we (as a scientific community) cast a critical eye upon it.

The two most reasonable justifications for comparing accuracies on natural data sets require empirical verification. We argue that a particular form of ROC analysis is the proper methodology to provide such verification. We then provide a thorough analysis of classifier performance using standard machine learning algorithms and standard benchmark data sets. The results raise serious concerns about the use of accuracy, both for practical comparisons and for drawing scientific conclusions, even when predictive performance is the only concern.

The contribution of this paper is two-fold. We analyze critically a common assumption of machine learning research, provide insights into its applicability, and discuss the implications. In the process, we describe what we believe to be a superior methodology for the evaluation of induction algorithms on natural data sets. Although ROC analysis certainly is not new, for machine learning research it should be applied in a principled manner geared to the specific conclusions machine learning researchers would like to draw. We hope that this work makes significant progress toward that goal.

2 JUSTIFYING ACCURACY COMPARISONS

We consider induction problems for which the intent in applying machine learning algorithms is to build from the existing data a model (a *classifier*) that will be used to classify previously unseen examples. We limit ourselves to predictive performance—which is clearly the intent of most accuracy-based machine learning studies—and do not consider issues such as comprehensibility and computational performance.

We assume that the true distribution of examples to which the classifier will be applied is not known in advance. To make an informed choice, performance must be estimated using the data available. The different methodologies for arriving at these estimations have been described elsewhere (Kohavi, 1995;

Dietterich, 1998). By far, the most commonly used performance metric is classification accuracy.

Why should we care about comparisons of accuracies on benchmark data sets? Theoretically, over the universe of induction algorithms no algorithm will be superior on all possible induction problems (Wolpert, 1994; Schaffer, 1994). The tacit reason for comparing classifiers on natural data sets is that these data sets represent problems that systems might face in the real world, and that superior performance on these benchmarks may translate to superior performance on other real-world tasks. To this end, the field has amassed an admirable collection of data sets from a wide variety of classifier applications (Merz and Murphy, 1998). Countless research results have been published based on comparisons of classifier accuracy over these benchmark data sets. We argue that comparing accuracies on our benchmark data sets says little, if anything, about classifier performance on real-world tasks.

Accuracy maximization is not an appropriate goal for many of the real-world tasks from which our natural data sets were taken. Classification accuracy assumes equal misclassification costs (for false positive and false negative errors). This assumption is problematic, because for most real-world problems one type of classification error is much more expensive than another. This fact is well documented, primarily in other fields (statistics, medical diagnosis, pattern recognition and decision theory). As an example, consider machine learning for fraud detection, where the cost of missing a case of fraud is quite different from the cost of a false alarm (Fawcett and Provost, 1997).

Accuracy maximization also assumes that the class distribution (class priors) is known for the target environment. Unfortunately, for our benchmark data sets, we often do not know whether the existing distribution is the natural distribution, or whether it has been stratified. The iris data set has exactly 50 instances of each class. The splice junction data set (DNA) has 50% donor sites, 25% acceptor sites and 25% nonboundary sites, even though the natural class distribution is very skewed: no more than 6% of DNA actually codes for human genes (Saitta and Neri, 1998). Without knowledge of the target class distribution we cannot even claim that we are indeed maximizing accuracy for the problem from which the data set was drawn.

If accuracy maximization is not appropriate, why would we use accuracy estimates to compare induction algorithms on these data sets? Here are what we believe to be the two best candidate justifications.

1. The *classifier* with the highest accuracy may very well be the classifier that minimizes cost, particularly when the classifier's tradeoff between true positive predictions and false positives can be tuned. Consider a learned model that produces probability estimates; these can be combined with prior probabilities and cost estimates for decision-analytic classifications. If the model has high classification accuracy because it produces very good probability estimates, it will also have low cost for any target scenario.

2. The *induction algorithm* that produces the highest accuracy classifiers may also produce minimum-cost classifiers by training it differently. For example, Breiman et al. (1984) suggest that altering the class distribution will be effective for building cost-sensitive decision trees (see also other work on cost-sensitive classification (Turney, 1996)).

To criticize the practice of comparing machine learning algorithms based on accuracy, it is not sufficient merely to point out that accuracy is not the metric by which real-world performance will be measured. Instead, it is necessary to analyze whether these candidate justifications are well founded.

3 ARE THESE JUSTIFICATIONS REASONABLE?

We first discuss a commonly cited special case of the second justification, arguing that it makes too many untenable assumptions. We then present the results of an empirical study that leads us to conclude that these justifications are questionable at best.

3.1 CAN WE DEFINE AWAY THE PROBLEM?

In principle, for a two-class problem one can reproportion ("stratify") the classes based on the target costs and class distribution. Once this has been done, maximizing accuracy on the transformed data corresponds to minimizing costs on the target data (Breiman *et al.*, 1984). Unfortunately, this strategy is impracticable for conducting empirical research based on our benchmark data sets. First, the transformation is valid only for two-class problems. Whether it can be approximated effectively for multiclass problems is an open question.

Second, we do not know appropriate costs for these data sets and, as noted by many applied researchers (Bradley, 1997; Catlett, 1995; Provost and Fawcett, 1997), assigning these costs precisely is virtually impossible. Third, as described above, generally we do not know whether the class distribution in a natural data set is the "true" target class distribution.

Because of these uncertainties we cannot claim to be able to transform these cost-minimization problems into accuracy-maximization problems. Moreover, in many cases specifying target conditions is not just virtually impossible, it is actually impossible. Often in real-world domains there are no "true" target costs and class distribution. These change from time to time, place to place, and situation to situation (Fawcett and Provost, 1997).

Therefore the ability to transform cost minimization into accuracy maximization does not, by itself, justify limiting our comparisons to classification accuracy on the given class distribution. However, it may be that comparisons based on classification accuracy are useful because they are indicative of a broader notion of "better" performance.

3.2 ROC ANALYSIS AND DOMINATING MODELS

We now investigate whether an algorithm that generates high-accuracy classifiers is generally better because it also produces low-cost classifiers for the target cost scenario. Without target cost and class distribution information, in order to conclude that the classifier with higher accuracy is the better classifier, one must show that it performs better for any reasonable assumptions. We limit our investigation to two-class problems because the analysis is straightforward.

The evaluation framework we choose is Receiver Operating Characteristic (ROC) analysis, a classic methodology from signal detection theory that is now common in medical diagnosis and has recently begun to be used more generally in AI (Swets, 1988; Provost and Fawcett, 1997).

We briefly review some of the basics of ROC analysis. *ROC space* denotes the coordinate system used for visualizing classifier performance. In ROC space, typically the true positive rate, TP, is plotted on the Y axis and the false positive rate, FP, is plotted on the X axis. Each classifier is represented by the point in ROC space corresponding to its (FP,TP) pair. For models that produce a continuous output (*e.g.*, an estimate of the posterior probability of an instance's class membership), these statistics vary together as a threshold on the output is varied between its extremes, with each threshold value defining a classifier. The resulting curve, called the *ROC curve*, illustrates the error tradeoffs available with a given model. ROC curves describe the predictive behavior of a classifier *independent of class distributions or error costs*, so they decouple classification performance from these factors.

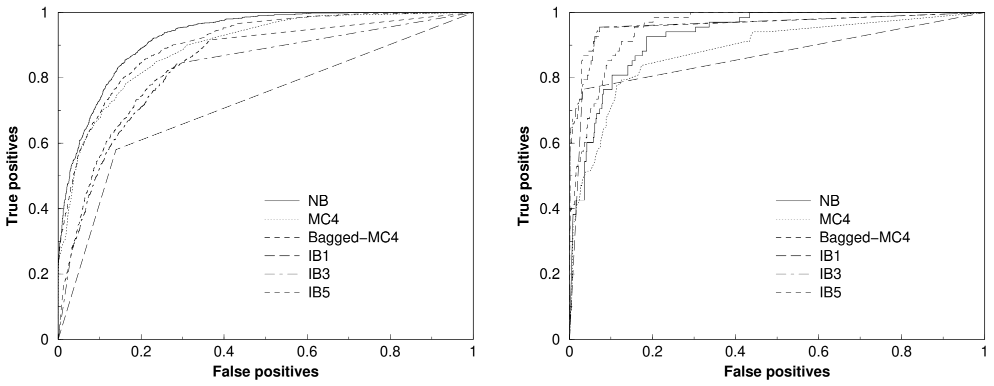
For our purposes, a crucial notion is whether one model *dominates* in ROC space, meaning that all other ROC curves are beneath it or equal to it. A dominating model (*e.g.*, model NB in Figure 1a) is at least as good as all other models for all possible cost and class distributions. Therefore, if a dominating model exists, it can be considered to be the "best" model in terms of predictive performance. If a dominating model does not exist (as in Figure 1b), then none of the models represented is best under all target scenarios; in such cases, there exist scenarios for which the model that maximizes accuracy (or any other single-number metric) does not have minimum cost.

Figure 1 shows test-set ROC curves on two of the UCI domains from the study described below. Note the "bumpiness" of the ROC curves in Figure 1b (these were two of the largest domains with the least bumpy ROC curves). This bumpiness is typical of induction studies using ROC curves generated from a hold-out test set. As with accuracy estimates based on a single hold-out set, these ROC curves may be misleading because we cannot tell how much of the observed variation is due to the particular training/test partition. Thus it is difficult to draw strong conclusions about the expected behavior of the learned models. We would like to conduct ROC analysis using cross-validation.

Bradley (1997) produced ROC curves from 10-fold cross validation, but they are similarly bumpy. Bradley generated the curves using a technique known as *pooling*. In pooling, the ith points making up each raw ROC curve are averaged. Unfortunately, as discussed by Swets and Pickett (1982), pooling assumes that the ith points from all the curves are actually estimating the same point in ROC space, which is doubtful given Bradley's method of generating curves.[1] For our study it is important to have a good approximation of the expected ROC curve.

We generate results from 10-fold cross-validation using a different methodology, called *averaging*. Rather than using the averaging procedure recommended by Swets

[1] Bradley acknowledges this fact, and it is not germane to his study. However, it is problematic for us.



(a) Adult (b) Satimage

Figure 1: Raw (un-averaged) ROC curves from two UCI database domains

and Pickett, which assumes normal-fitted ROC curves in a binormal ROC space, we average the ROC curves in the following manner. For k-fold cross-validation, the ROC curve from each of the k folds is treated as a function, R_i, such that $TP = R_i(FP)$. This is done with linear interpolations between points in ROC space[2] (if there are multiple points with the same FP, the one with the maximum TP is chosen). The averaged ROC curve is the function $\hat{R}(FP) = mean(R_i(FP))$. To plot averaged ROC curves we sample from $\hat{R}$ at 100 points regularly spaced along the FP-axis. We compute confidence intervals of the mean of TP using the common assumption of a binomial distribution.

3.3 DO STANDARD METHODS PRODUCE DOMINATING MODELS?

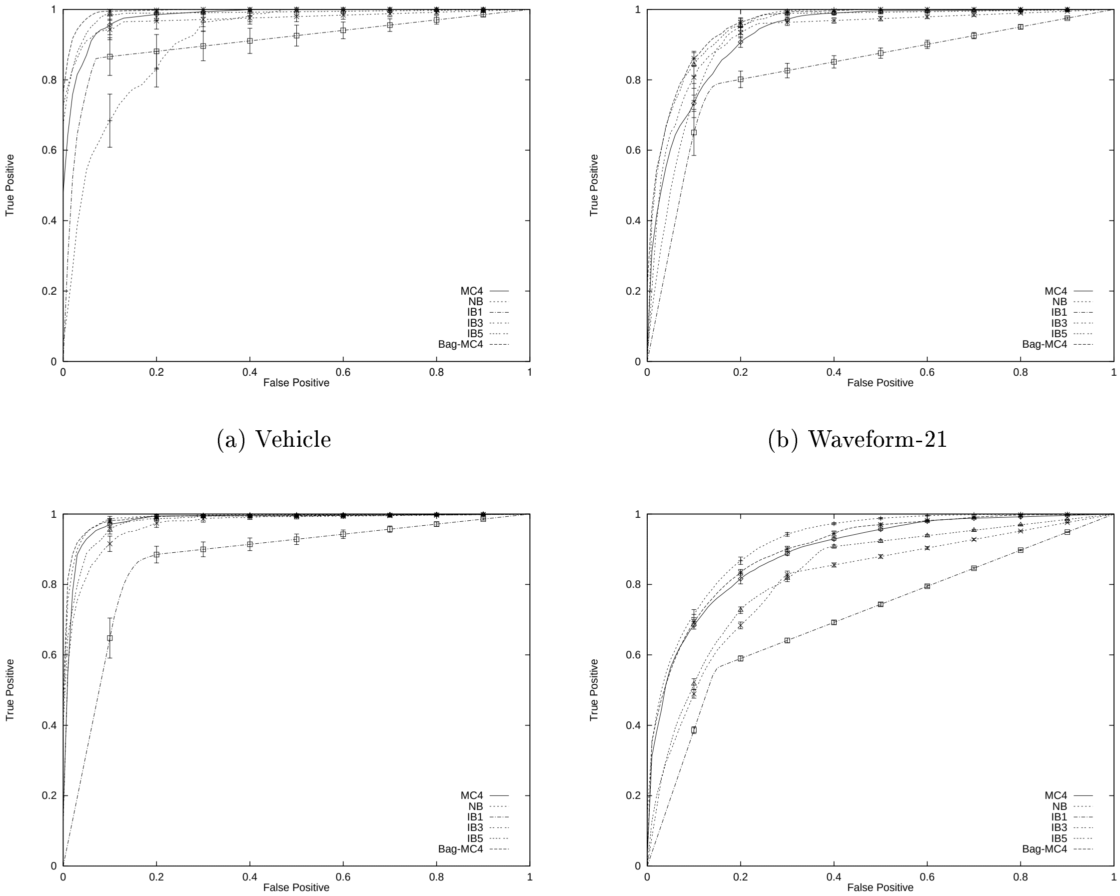
We can now state precisely a basic hypothesis to be investigated: *Our standard learning algorithms produce dominating models for our standard benchmark data sets.* If this hypothesis is true (generally), we might conclude that the algorithm with higher accuracy is generally better, regardless of target costs or priors.[3] If the hypothesis is not true, then such a conclusion will have to rely on a different justification. We now provide an experimental study of this hypothesis, designed as follows.

From the UCI repository we chose ten datasets that contained at least 250 instances, but for which the accuracy for decision trees was less than 95% (because the ROC curves are difficult to read at very high accuracies). For each domain, we induced classifiers for the minority class (for Road we chose the class Grass). We selected several inducers from $\mathcal{MLC}$++ (Kohavi *et al.*, 1997): a decision tree learner (MC4), Naive Bayes with discretization (NB), k-nearest neighbor for several k values (IBk), and Bagged-MC4 (Breiman, 1996). MC4 is similar to C4.5 (Quinlan, 1993); probabilistic predictions are made by using a Laplace correction at the leaves. NB discretizes the data based on entropy minimization (Dougherty *et al.*, 1995) and then builds the Naive-Bayes model (Domingos and Pazzani, 1997). IBk votes the closest k neighbors; each neighbor votes with a weight equal to one over its distance from the test instance.

The averaged ROC curves are shown in Figures 2 and 3. For *only one* (Vehicle) of these ten domains was there an absolute dominator. In general, very few of the 100 runs we performed (10 data sets, 10 cross-validation folds each) had dominating classifiers. Some cases are very close, for example Adult and Waveform-21. In other cases a curve that dominates in one area of ROC space is dominated in another. Therefore, we can refute the hypothesis that our algorithms produce

[2] Note that classification performance anywhere along a line segment connecting two ROC points can be achieved by randomly selecting classifications (weighted by the interpolation proportion) from the classifiers defining the endpoints.

[3] However, even this conclusion has problems. Accuracy comparisons may select a non-dominating classifier because it is indistinguishable at the point of comparison—yet it may be much worse elsewhere.



(a) Vehicle

(b) Waveform-21

(c) DNA

(d) Adult

Figure 2: Smoothed ROC curves from UCI database domains

(statistically significantly) dominating classifiers.

This draws into question claims of "algorithm A is better than algorithm B" based on accuracy comparison. In order to draw such a conclusion in the absence of target costs and class distributions, the ROC curve for algorithm A would have to be a significant dominator of algorithm B. This has obvious implications for machine learning research.

In practical situations, often a weaker claim is sufficient: Algorithm A is a good choice because it is at least as good as Algorithm B (*i.e.*, their accuracies are not significantly different). It is clear that this type of conclusion also is not justified. In many domains, curves that are statistically indistinguishable from dominators in one area of the space are significantly dominated in another. Moreover, in practical situations typically comparisons are not made with the wealth of classifiers we are considering. More often only a few classifiers are compared. Considering general pairwise comparisons of algorithms, there are many cases where each model in a pair is clearly much better than the other in different regions of ROC space. This clearly draws into question the use of single number metrics for practical algorithm comparison, unless these metrics are based on precise target cost and class distribution information.

3.4 CAN STANDARD METHODS BE COERCED TO YIELD DOMINATING ROC CURVES?

The second justification for using accuracy to compare algorithms is subtly different from the first. Specifically, it allows for the possibility of coercing algorithms to produce different behaviors under different scenarios (such as in cost-sensitive learning). If this can be done well, accuracy comparisons are justified by arguing that for a given domain, the algorithm with higher accuracy will also be the algorithm with lower cost for all reasonable costs and class distributions.

Confirming or refuting this justification completely is beyond the scope of this paper, because how best to coerce algorithms for different environmental conditions is an open question. Even the straightforward method of stratifying samples has not been evaluated satisfactorily. We argue that the ROC framework outlined so far, with a minor modification, can be used to evaluate this question as well. We then offer some disconfirming evidence.

For algorithms that may produce different models under different cost and class distributions, the ROC methodology as stated above is not quite adequate. We must be able to evaluate the performance of the *algorithm*, not an individual model. However, one can characterize an algorithm's performance for ROC analysis by producing a composite curve for a set of generated models. This can be done using pooling, or by using the convex hull of the ROC curves produced by the set of models, as described in detail by Provost and Fawcett (1997; 1998).

We can now form a hypothesis for our second potential justification: *Our standard learning algorithms produce dominating ROC curves for our standard benchmark data sets.* Confirming this hypothesis would be an important step in justifying the common practice of ignoring target costs and class distributions in classfier comparisons on natural data. Unfortunately, we know of no confirming evidence.

On the other hand, there is disconfirming evidence. First, consider the results presented above. Naive Bayes is robust with respect to changes in costs—it will produce the same ROC curve regardless of the target costs and class distribution. Furthermore, it has been shown that decision trees are surprisingly robust if the probability estimates are generated with the Laplace estimate (Bradford *et al.*, 1998). If this result holds generally, the results in the previous section would disconfirm the present hypothesis as well.

Second, Bradley's (1997) results provide disconfirming evidence. Specifically, he studied six real-world medical data sets (four from the UCI repository and two from other sources). Bradley plotted the ROC curves of six classifier learning algorithms, consisting of two neural nets, two decision trees and two statistical techniques. Bradley uses composite ROC curves formed by training models differently for different cost distributions. We have previously criticized the design of his study for the purpose of answering our question. However, if the results can be replicated under the current methodology, they would make a strong statement. *Not one* of the six data sets had a dominating classifier. This implies that for each domain there exist disjoint sets of conditions for which different induction algorithms are preferable.

4 RECOMMENDATIONS AND LIMITATIONS

When designing comparative studies, researchers should be clear about the conclusions they want to be able to draw from the results. We have argued that comparisons of algorithms based on accuracy are unsatisfactory when there is no dominating classifier. However, presenting the case against the use of accuracy is only one of our goals. We also want to show how precise comparisons still can be made, even when the target cost and class distributions are not known.

If there is no dominator, conclusions must be qualified. No single number metric can be used to make very strong conclusions without domain-specific information. However, it is possible to look at ranges of costs and class distributions for which each classifier dominates. The problems of cost-sensitive classification and learning with skewed class distributions can be analyzed precisely.

Even without knowledge of target conditions, a precise, concise, robust specification of classifier performance can be made. As described in detail by Provost and Fawcett (1997), the slopes of the lines tangent to the ROC convex hull determine the ranges of costs and class distributions for which particular classifiers minimize cost. For specific target conditions, the corresponding slope is the cost ratio times the reciprocal of the class ratio. For our ten domains, the optimal classifiers for different target conditions are given in Table 1. For example, in the Road domain (see Figure 3 and Table 1), Naive Bayes is the best classifier for any target conditions corresponding to a slope less than 0.38, and Bagged-MC4 is best for slopes greater

Table 1: Locally dominating classifiers for ten UCI domains

Domain	Slope range	Dominator
Adult	[0, 7.72]	NB
	[7.72, 21.6]	Bagged-MC4
	[21.6, ∞)	NB
Breast cancer	[0, 0.37]	NB
	[0.37, 0.5]	IB3
	[0.5, 1.34]	IB5
	[1.34, 2.38]	IB3
	[2.38, ∞)	Bagged-MC4
CRX	[0, 0.03]	Bagged-MC4
	[0.03, 0.06]	NB
	[0.06, 2.06]	Bagged-MC4
	[2.06, ∞)	NB
German	[0, 0.21]	NB
	[0.21, 0.47]	Bagged-MC4
	[0.47, 3.08]	NB
	[3.08, ∞)	IB5
Road (Grass)	[0, 0.38]	NB
	[0.38, ∞)	Bagged-MC4
DNA	[0, 1.06]	NB
	[1.06, ∞)	Bagged-MC4
Pima	[0, 0.06]	NB
	[0.06, 0.11]	Bagged-MC4
	[0.11, 0.30]	NB
	[0.30, 0.82]	Bagged-MC4
	[0.82, 1.13]	NB
	[1.13, 4.79]	Bagged-MC4
	[4.79, ∞)	NB
Satimage	[0, 0.05]	NB
	[0.05, 0.22]	Bagged-MC4
	[0.22, 2.60]	IB5
	[2.60, 3.11]	IB3
	[3.11, 7.54]	IB5
	[7.54, 31.14]	IB3
	[31.14, ∞)	Bagged-MC4
Waveform 21	[0, 0.25]	NB
	[0.25, 4.51]	Bagged-MC4
	[4.51, 6.12]	IB5
	[6.12, ∞)	Bagged-MC4
Vehicle	[0, ∞)	Bagged-MC4

than 0.38. They perform equally well at 0.38. We admit that this is not as elegant as a single-number comparison, but we believe it to be much more useful, both for research and in practice.

In summary, if a dominating classifier does not exist and cost and class distribution information is unavailable, no strong statement about classifier superiority can be made. However, one might be able to make precise statements of superiority for specific regions of ROC space. For example, if all you know is that few false positive errors can be tolerated, you may be able to find a particular algorithm that is superior at the "far left" edge of ROC space.

We limited our investigation to two classes. This does not affect our conclusions since our results are negative. However, since we are also recommending an analytical framework, we note that extending our work to multiple dimensions is an interesting open problem.

Finally, we are not completely satisfied with our method of generating confidence intervals. The present intervals are appropriate for the Neyman-Pearson observer (Egan, 1975), which wants to maximize TP for a given FP. However, their appropriateness is questionable for evaluating minimum expected cost, for which a given set of costs contours ROC space with lines of a particular slope. Although this is an area of future work, it is not a fundamental drawback to the methodology.

5 CONCLUSIONS

We have offered for debate the justification for the use of accuracy estimation as the primary metric for comparing algorithms on our benchmark data sets. We have elucidated what we believe to be the top candidates for such a justification, and have shown that either they are not realistic because we cannot specify cost and class distributions precisely, or they are not supported by experimental evidence.

We draw two conclusions from this work. First, the justifications for using accuracy to compare classifiers are questionable at best. Second, we have described what we believe to be the proper use of ROC analysis as applied to comparative studies in machine learning research. ROC analysis is not as simple as comparing with a single-number metric. However, we believe that the additional power it delivers is well worth the effort. In certain situations, ROC analysis allows very strong, general conclusions to be made—both positive and negative. In situations where strong, general conclusions can not be made, ROC analysis allows very precise analysis to be conducted.

Although ROC analysis is not new, in machine learning research it has not been applied in a principled manner, geared to the specific conclusions machine learning researchers would like to draw. We hope that this work makes significant progress toward that goal.

Acknowledgements

We thank the many with whom we have discussed the justifications for accuracy-based comparisons and ROC analysis as applied to classifier learning. Rob Holte provided very helpful comments on a draft of this paper.

References

J. Bradford, C. Kunz, R. Kohavi, C. Brunk, and C. Brodley. (1998) Pruning decision trees with misclassification costs. In *Proceedings of ECML-98*, pages 131–136.

A. P. Bradley. (1997) The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159.

L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. (1984) *Classification and Regression Trees.* Wadsworth International Group.

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

J. Catlett. (1995) Tailoring rulesets to misclassificatioin costs. In *Proceedings of the 1995 Conference on AI and Statistics*, pages 88–94.

T. G. Dietterich. (1998) Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation.* To appear.

P. Domingos and M. Pazzani. (1997) Beyond independence: Conditions for the optimality of the simple Bayesian classifier. *Machine Learning*, 29:103–130.

J. Dougherty, R. Kohavi, and M. Sahami. (1995) Supervised and unsupervised discretization of continuous features. In A. Prieditis and S. Russell, (eds.), *Proceedings of ICML-95*, pages 194–202. Morgan Kaufmann.

J. P. Egan. (1975) *Signal Detection Theory and ROC Analysis.* Series in Cognitition and Perception. Academic Press, New York.

T. Fawcett and F. Provost. (1997) Adaptive fraud detection. *Data Mining and Knowledge Discovery*, 1(3). Available: `http://www.croftj.net/~fawcett/DMKD-97.ps.gz`.

R. Kohavi, D. Sommerfield, and J. Dougherty. (1997) Data mining using $\mathcal{MLC}$++: A machine learning library in C++. *International Journal on Artificial Intelligence Tools*, 6(4):537–566. Available: `http://www.sgi.com/Technology/mlc`.

R. Kohavi. (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In C. S. Mellish, (ed.), *Proceedings of IJCAI-95*, pages 1137–1143. Morgan Kaufmann. Available: `http://robotics.stanford.edu/~ronnyk`.

C. Merz and P. Murphy. (1998) UCI repository of machine learning databases. Available: `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

F. Provost and T. Fawcett. (1997) Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In *Proceedings of KDD-97*, pages 43–48. AAAI Press.

F. Provost and T. Fawcett. (1998) Robust classification systems for imprecise environments. In *Proceedings of AAAI-98.* AAAI Press. To appear. Available: `http://www.croftj.net/~fawcett/papers/aaai98-dist.ps.gz`.

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

L. Saitta and F. Neri. (1998) Learning in the "Real World". *Machine Learning*, 30:133–163.

S. L. Salzberg. (1997) On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery*, 1:317–328.

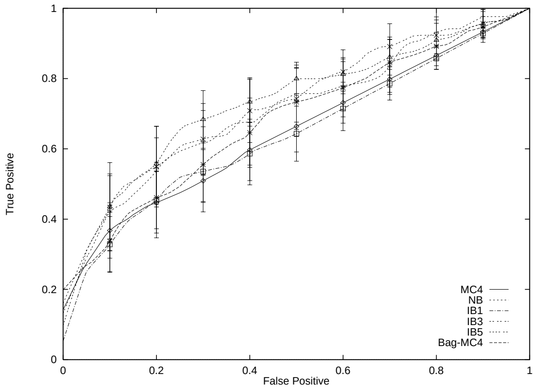
C. Schaffer. (1994) A conservation law for generalization performance. In *ICML-94*, pages 259–265. Morgan Kaufmann.

J. A. Swets and R. M. Pickett. (1982) *Evaluation of Diagnostic Systems: Methods from Signal Detection Theory.* New York: Academic Press.

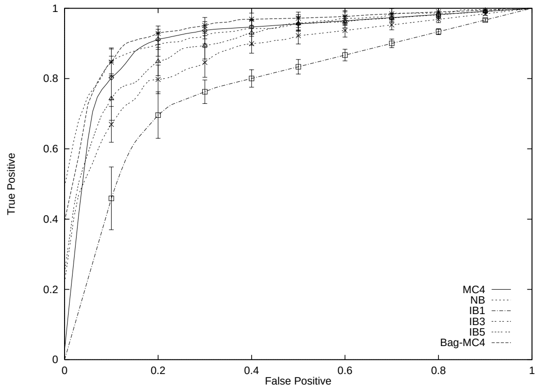
J. Swets. (1988) Measuring the accuracy of diagnostic systems. *Science*, 240:1285–1293.

P. Turney. (1996) Cost sensitive learning bibliography. Available: `http://ai.iit.nrc.ca/bibliographies/cost-sensitive.html`.

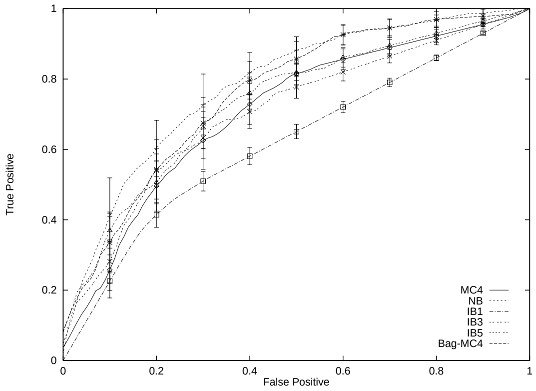
D. H. Wolpert. (1994) The relationship between PAC, the statistical physics framework, the Bayesian framework, and the VC framework. In D. H. Wolpert, (ed.), *The Mathematics of Generalization.* Addison Wesley.



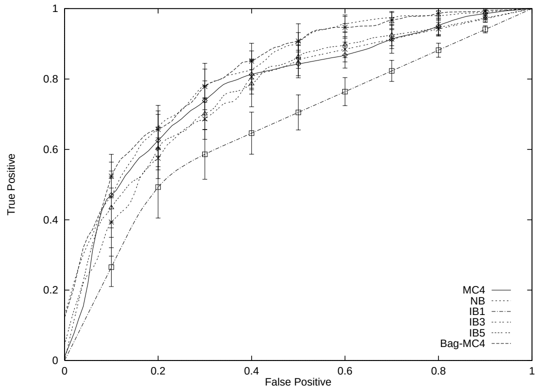
(a) Breast cancer



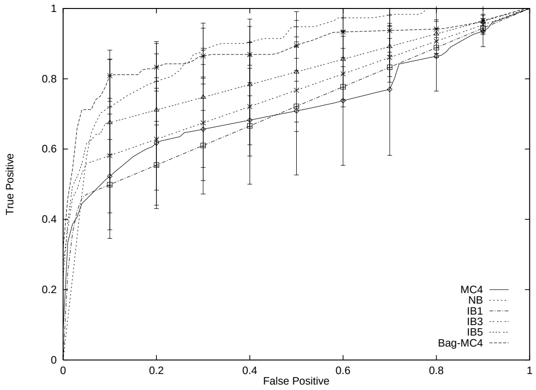
(b) CRX



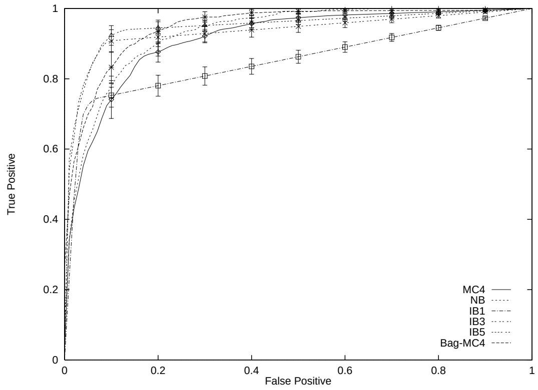
(c) German



(d) Pima



(e) RoadGrass



(f) Satimage

Figure 3: Smoothed ROC curves from UCI database domains, cont'd