

# Instance weighting versus threshold adjusting for cost-sensitive classification

Huimin Zhao

Received: 10 May 2006 / Revised: 22 February 2007 / Accepted: 3 March 2007  
Published online: 3 April 2007  
© Springer-Verlag London Limited 2007

**Abstract** In real-world classification problems, different types of misclassification errors often have asymmetric costs, thus demanding cost-sensitive learning methods that attempt to minimize average misclassification cost rather than plain error rate. Instance weighting and post hoc threshold adjusting are two major approaches to cost-sensitive classifier learning. This paper compares the effects of these two approaches on several standard, off-the-shelf classification methods. The comparison indicates that the two approaches lead to similar results for some classification methods, such as Naïve Bayes, logistic regression, and back-propagation neural network, but very different results for other methods, such as decision tree, decision table, and decision rule learners. The findings from this research have important implications on the selection of the cost-sensitive classifier learning approach as well as on the interpretation of a recently published finding about the relative performance of Naïve Bayes and decision trees.

**Keywords** Classification · Cost-sensitive learning · Instance weighting · Threshold adjusting · ROC curve

## 1 Introduction

Classification is an important type of predictive data mining problem, where the objective is to predict the value of a categorical-dependent variable based on the values of several independent variables. Real-world classification problems are often *cost-sensitive*, meaning that different types of misclassification errors should incur different costs. For example, in bank loan application approval, approving a bad loan is much more costly than rejecting a potentially good loan [33], while in cancer diagnosis [7], failing to diagnose a cancer case has dire consequences and is far more costly than misidentifying an actually negative case.

---

H. Zhao (✉)  
Sheldon B. Lubar School of Business, University of Wisconsin—Milwaukee,  
P. O. Box 742, Milwaukee, WI 53201, USA  
e-mail: hzhao@uwm.edu

While most basic classification methods have not been designed to solve cost-sensitive problems, two major approaches, *instance weighting* [8,26,31,34] and post hoc *threshold adjusting* [1,30], have been proposed for converting these methods into cost-sensitive classifier learners. In instance weighting, different types of instances in the training dataset are weighted according to the misclassification costs during classifier learning, such that the classifier strives to make fewer errors of the more costly type, resulting in lower overall cost. In post hoc threshold adjusting, a classifier is first learned as usual without tempering the training data and then adjusted to account for the costs. Apparently, instance weighting is computationally more involved than post hoc threshold adjusting. The difference is especially considerable when the decision maker cannot precisely pinpoint the costs of different types of misclassification errors and has to consider a range of possible cost settings. To threshold adjusting, changing the cost setting simply involves an adjustment of the decision threshold and does not require retraining the classifier. To instance weighting, however, the classifier needs to be retrained from scratch whenever a new cost setting needs to be considered.

The objective of this research is to compare the effects of the two cost-sensitive learning approaches on various standard, off-the-shelf base classification methods. Findings from such a comparative study will have important implications on the selection of the cost-sensitive learning approach with regard to a particular base classification method. If a base classification method is insensitive to the cost ratio when instance weighting is applied, the computationally cheaper post hoc threshold adjusting approach is appropriate. However, if a base classification method is very sensitive to the cost ratio when instance weighting is used, it becomes very important to incorporate the misclassification costs during training. This study will also help in interpreting a recently published finding about the relative performance of Naïve Bayes and decision trees [19]: while the two methods often achieve similar cost-insensitive performance (i.e., classification accuracy), Naïve Bayes tends to significantly outperform decision trees in terms of another aggregate performance measure, AUC (area under the ROC curve) [3]. We will further investigate whether decision tree learning is indeed inferior to Naïve Bayes in dealing with cost-sensitive problems.

The remainder of the paper is organized as follows. We first briefly review some related work in the next section. We then discuss cost-sensitive learning for several standard, off-the-shelf classification methods in Sect. 3. We then report on some empirical results in Sect. 4. Finally, we conclude the paper and discuss future research directions in Sect. 5.

## 2 Related work

Cost-sensitive classification is an active research area in data mining. Without considering costs, classification accuracy (the probability of making a correct prediction) is a reasonable measure for assessing the performance of a classifier. In cost-sensitive problems, the performance of a classifier should be assessed by the expected misclassification cost rather than accuracy [12,27]. A closely related problem is the classification of imbalanced datasets, where different classes are not approximately equally represented [5,20]. Real-world imbalanced, cost-sensitive classification problems that have been studied in the literature include bank loan approval [30], financial distress prediction [25], credit card fraud detection [4], and unsolicited bulk email categorization [17], just to name a few.

Several methods have been proposed for dealing with cost-sensitive classification problems. Some methods extend particular cost-insensitive classification methods. Drummond and Holte [10] proposed methods for incorporating misclassification costs into decision tree learning. The Iterative Bayes method [15] turns Naïve Bayes cost-sensitive

by taking misclassification costs into account. The AdaCost algorithm [13] modifies the AdaBoost algorithm into a cost-sensitive boosting method.

While the above-mentioned methods extend particular classification methods, instance weighting [8, 26, 31, 34] and post hoc threshold adjusting [1, 30] are generic approaches to converting any classification method into a cost-sensitive one. Instance weighting has similar effects as instance re-sampling (over-sampling or under-sampling) [5]. Post hoc threshold adjusting mainly relies on the posterior class probabilities estimated by the underlying classification method. This research compares these two approaches in relation to several base classification methods.

### 3 Cost-sensitive classifier learning

To simplify the discussion, we consider *binary* classification problems, where the objective is to predict the value of a binary-dependent variable  $\mathbf{y}$ , referred to as the class, based on a vector of independent variables (also called attributes or features)  $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m)$ , where  $m$  is the number of attributes and  $x_i$  is a scalar. A problem instance is considered positive (negative) if its class  $\mathbf{y} = 1(0)$ . A classification algorithm learns a prediction model, known as a classifier, from a training dataset consisting of previously solved problem instances, whose class values are known. The classifier is then applied to make classification decisions on other problem instances. A classification problem is cost-sensitive if the costs of the two types of misclassification errors (false positive and false negative) are different.

#### 3.1 Naïve Bayes

Without considering costs, a rational classifier should estimate the odds ratio,  $\frac{P(\mathbf{y}=1|\mathbf{X}=\mathbf{x})}{P(\mathbf{y}=0|\mathbf{X}=\mathbf{x})}$ , and make a positive (negative) prediction if the odds ratio is above (below) one [32]. Assuming conditional independence of the attributes given a class value, Naïve Bayes [9, 32, 35] estimates the odds ratio as

$$O(\mathbf{x} = x) = \frac{P(\mathbf{y} = 1|\mathbf{x} = x)}{P(\mathbf{y} = 0|\mathbf{x} = x)} = \frac{p_1 P(\mathbf{x} = x|\mathbf{y} = 1)}{p_0 P(\mathbf{x} = x|\mathbf{y} = 0)} = \frac{p_1 \prod_{i=1}^m P(\mathbf{x}_i = x_i|\mathbf{y} = 1)}{p_0 \prod_{i=1}^m P(\mathbf{x}_i = x_i|\mathbf{y} = 0)}, \quad (1)$$

where  $p_1(p_0)$  is the prior probability that a problem instance is positive (negative). This leads to the following condition for making a positive prediction:

$$\frac{\prod_{i=1}^m P(\mathbf{x}_i = x_i|\mathbf{y} = 1)}{\prod_{i=1}^m P(\mathbf{x}_i = x_i|\mathbf{y} = 0)} > \frac{p_0}{p_1}. \quad (2)$$

When costs are considered, the optimal condition minimizing the expected misclassification cost has the right-hand side of (2) adjusted to  $\frac{p_0 C_{10}}{p_1 C_{01}}$ , where  $C_{10}(C_{01})$  is the cost of a false positive (negative) error. A typical instance weighting scheme is to modify the weight of an instance proportional to the cost of misclassifying the actual class of the instance [31]. If the (prior and conditional) probabilities in the Naïve Bayes classifier are directly estimated as sample proportions based on the training data set, it can be easily shown that such instance weighting using the cost-insensitive classification condition (2) leads to exactly the same condition as threshold adjusting does. The estimates of the conditional probabilities  $P(\mathbf{x}_i = x_i|\mathbf{y} = j)(i = 1, 2, \dots, m; j = 0, 1)$  are invariant under instance weighting. Let  $n_1(n_0)$  denote the number of positive (negative) instances in the training dataset, the

right-hand side of (2) becomes

$$\frac{p'_0}{p'_1} = \frac{n_0 C_{10} / (n_0 C_{10} + n_1 C_{01})}{n_1 C_{01} / (n_0 C_{10} + n_1 C_{01})} = \frac{C_{10} n_0 / (n_0 + n_1)}{C_{01} n_1 / (n_0 + n_1)} = \frac{p_0 C_{10}}{p_1 C_{01}}. \tag{3}$$

### 3.2 Discriminant analysis, logistic regression, and neural network

It has also been a common practice in applications of statistical multivariate analysis techniques, such as linear discriminant analysis (LDA) and logistic regression [1,24], to adjust classification models post hoc based on prior probabilities and costs. For example, LDA adjusts the decision threshold on the linear discriminant function by  $\ln \frac{p_0 C_{10}}{p_1 C_{01}}$  [1].

Assuming that the independent variables follow a multivariate normal distribution for each of the two classes and that the two classes have equal dispersion and covariance structures, LDA computes a linear discriminate function  $\mathbf{Z} = \sum_{i=1}^m \beta_i \mathbf{x}_i$  and makes a positive prediction if

$$\mathbf{Z} > \mathbf{C} = \frac{\bar{\mathbf{Z}}_1 + \bar{\mathbf{Z}}_0}{2}, \tag{4}$$

where  $\bar{\mathbf{Z}}_1(\bar{\mathbf{Z}}_0)$  is the average value of  $\mathbf{Z}$  in the positive (negative) class. This procedure minimizes the classification error rate, if the assumptions hold. The posterior probability that a problem instance belongs to the positive class is

$$P(\mathbf{y} = 1 | \mathbf{x} = x) = \frac{1}{1 + e^{(-\mathbf{Z} + \mathbf{C})}}. \tag{5}$$

The odds ratio is

$$O(\mathbf{x} = x) = \frac{P(\mathbf{y} = 1 | \mathbf{x} = x)}{P(\mathbf{y} = 0 | \mathbf{x} = x)} = \frac{1}{e^{(-\mathbf{Z} + \mathbf{C})}}. \tag{6}$$

The optimal condition for making a positive prediction in (4) is equivalent to

$$O(\mathbf{x} = x) > 1. \tag{7}$$

When asymmetric prior probabilities and costs are considered, the dividing point  $\mathbf{C}$  should be chosen to minimize the expected misclassification cost and becomes

$$\mathbf{C} = \frac{\bar{\mathbf{Z}}_1 + \bar{\mathbf{Z}}_0}{2} + \ln \frac{p_0 C_{10}}{p_1 C_{01}}. \tag{8}$$

This is equivalent to adjusting the threshold on the odds ratio in (7) by  $\frac{p_0 C_{10}}{p_1 C_{01}}$ .

Logistic regression models can be adjusted in a similar manner. We also posit that instance weighting has relatively small effects on these as well as other methods, such as neural networks [29], which produce smooth continuous probability estimates and thus allow fine adjusting of the threshold on the odds ratio. A common characteristic of these methods is that the learned classifier is relatively insensitive to the cost ratio. The structure of a classifier learned by these methods is fixed and only the parameters (mainly the threshold on the odds ratio) may change when the cost ratio varies.

### 3.3 Symbolic classifiers

However, symbolic classifiers, such as decision trees [28], decision rules [6], and decision tables [23], are very different. The structure of a symbolic classifier may experience substantial

changes, with different sets of attributes being selected, when the cost ratio varies. For example, the heuristic goodness measure for selecting the branching attribute at an intermediate node of decision trees (e.g., information gain and gain ratio used in C4.5 [28]) does not preserve the ranking among the attributes—thus different attributes may be selected—under different cost ratios. Previous experiments have shown that very different decision trees are learned via instance weighting under different cost ratios [31] (see examples later). In addition, since only a small number (proportional to the number of tree leaves, rules, or selected attributes in a decision table) of thresholds are possible, threshold adjusting on symbolic classifiers can not be precise. We therefore expect that instance weighting has relatively large effects on symbolic classifiers.

#### 4 Visualization using ROC curves

The performance of a binary classifier can be visualized using an ROC (receiver operating characteristics) curve [14] or an alternative, such as cost curve [11]. Based on the odds ratio estimated by a classifier, a binary classification decision can be made by choosing a threshold on the odds ratio. Two quantities, *sensitivity* and *specificity*, can then be estimated. Sensitivity (specificity) is defined as the proportion of positive (negative) instances that are correctly classified by the classifier. The ROC curve plots sensitivity against  $1 - \text{specificity}$  on a series of possible thresholds (it is only a convention to use  $1 - \text{specificity}$ , rather than specificity, as the horizontal axis). The threshold adjusting approach can be viewed as selecting an appropriate point on the ROC curve that pertains to a cost ratio of 1. The area under the ROC curve, or simply AUC, is an aggregate performance measure over all possible thresholds, or equivalently, points on the ROC curve [3].

We have empirically evaluated the effects of instance weighting on the ROC curves produced by various classification methods. We used 13 binary classification datasets collected in the University of California, Irvine machine learning repository [2], including Pima Indians diabetes, ionosphere, hepatitis, horse colic, breast cancer, Wisconsin breast cancer, chess king-rook vs. king-pawn, credit card application approval, congressional voting records, labor relations, sonar, German credit (Statlog project), and heart disease (Statlog project).

We used the Weka machine learning toolkit [33] for the evaluation purpose. We used the CostSensitiveClassifier meta learner [33], which converts a base learner into a cost-sensitive learner by weighting training instances according to a given cost matrix, reflecting the ratio between the costs of the two types of misclassification errors. The base learners we used include Naïve Bayes [9], logistic regression [18], backpropagation neural network [29], C4.5 decision tree learner (named J4.8 in Weka) [28], Ripper decision rule learner (named JRip in Weka) [6], and a decision table learner [23]. We kept the default parameter values for all methods. We did not use LDA, as it relies on the usually unrealistic assumption of multivariate normality and is recommended for use only when logistic regression software is not available, and then only in preliminary analyzes [18]. We evaluated the classification methods under three cost ratios, 1:5, 1:1, and 5:1.

We used tenfold cross-validation [22] for estimating the ROC curves. Cross-validation randomly divides a dataset into several approximately equal-sized, stratified subsets, called folds, and repeatedly uses each fold for performance testing while the other folds are used for training a classifier. To obtain even smoother ROC curves, we repeated tenfold cross-validation twenty times and pooled the probability estimations on the 200 testing folds.

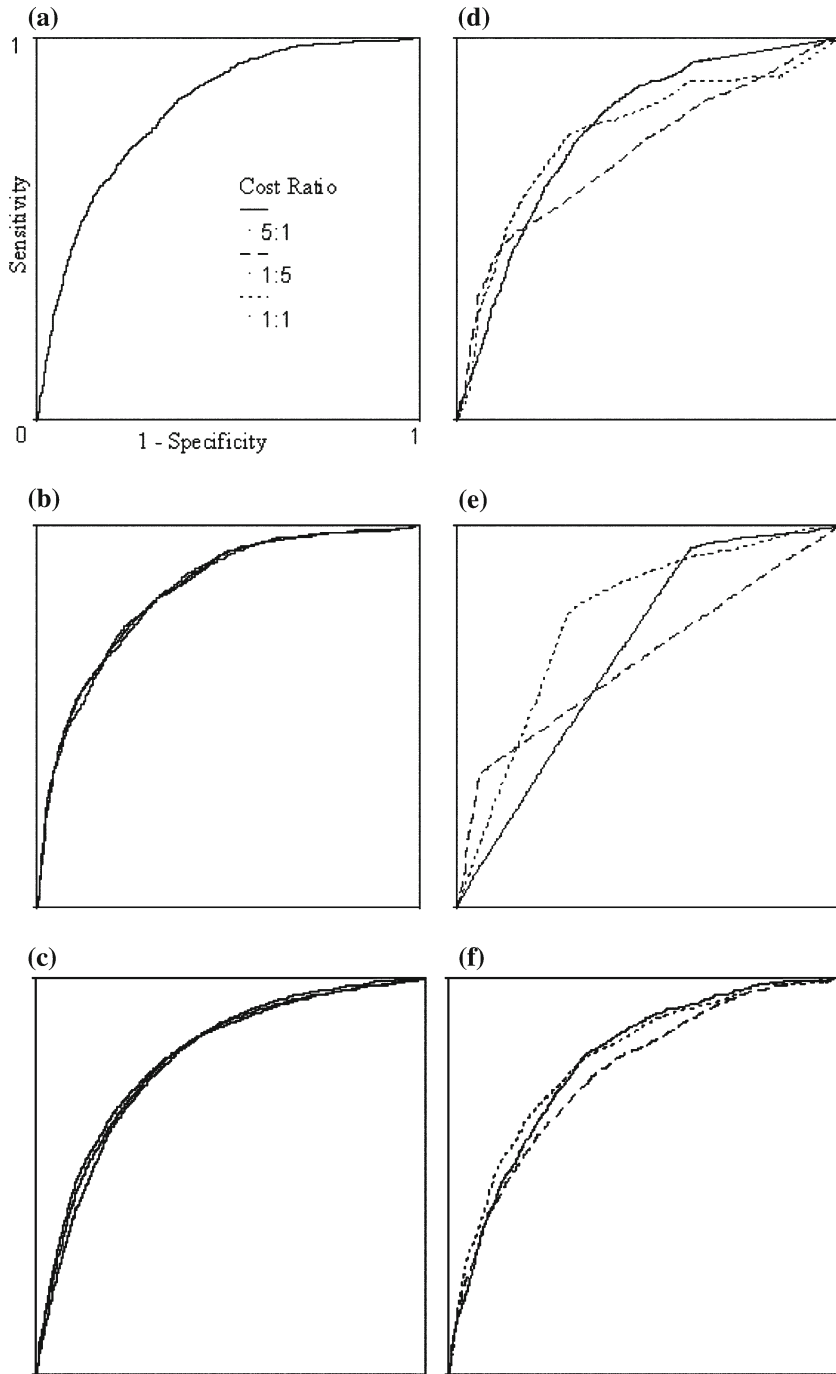
Figures 1, 2, 3, 4, 5 show the ROC curves produced by the six base classification methods under the three cost ratios for five datasets: Pima Indians diabetes, ionosphere, hepatitis, horse colic, and heart disease (Statlog project). The results on other datasets (available from the author) show similar patterns and hence are not reported, for the interest of space. We observe several interesting patterns in the ROC curves for the five datasets. First, the ROC curves produced by Naïve Bayes remain almost the same under different cost ratios, indicating that instance weighting has little effect beyond threshold adjusting. Second, the ROC curves produced by logistic regression and neural network change only slightly across different cost ratios. The benefit of instance weighting beyond that of threshold adjusting is therefore relatively small. Third, decision tree, decision rule, and decision table learners produce very different ROC curves under different cost ratios. For a given classifier/dataset, none of the ROC curves dominates the other two and each curve is superior in a particular section. This implies that different classifiers with substantially different structures are learned to fit different cost settings. Each classifier may excel in a particular range around the cost ratio under which it is trained. It is therefore important to weight the training instances according to the cost ratio when training decision trees, rules, and tables, as simply adjusting the threshold on the odds ratio (equivalent to selecting a particular point on the ROC curve) may result in sub-optimal performance. Figure 6 shows the decision trees learned under the three cost ratios for the Pima Indians diabetes dataset. The structures of these decision trees are very different. In summary, the post hoc threshold adjusting approach, which is computationally cheaper than instance weighting, is appropriate for Naïve Bayes, logistic regression, and backpropagation neural network, but not for decision tree, decision table, and decision rule learners.

A recent study has found that Naïve Bayes significantly outperforms decision trees in terms of AUC [19]. However, this should not be interpreted as that the former has better capability than the later in cost-sensitive learning. When the cost ratio is given, a cost-sensitive decision tree trained using instance weighting may outperform or compete well against Naïve Bayes. AUC is an aggregate measure over all possible cost ratios. It adequately reflects the capability of Naïve Bayes, whose ROC curve does not change across cost ratios, but not that of decision trees. It is indeed possible to use combinations of classifiers to operate on the convex hull of a set of ROC curves generated under various cost ratios [33]. The full capacity of decision trees is defined by the area under this convex hull, rather than the area under a single ROC curve.

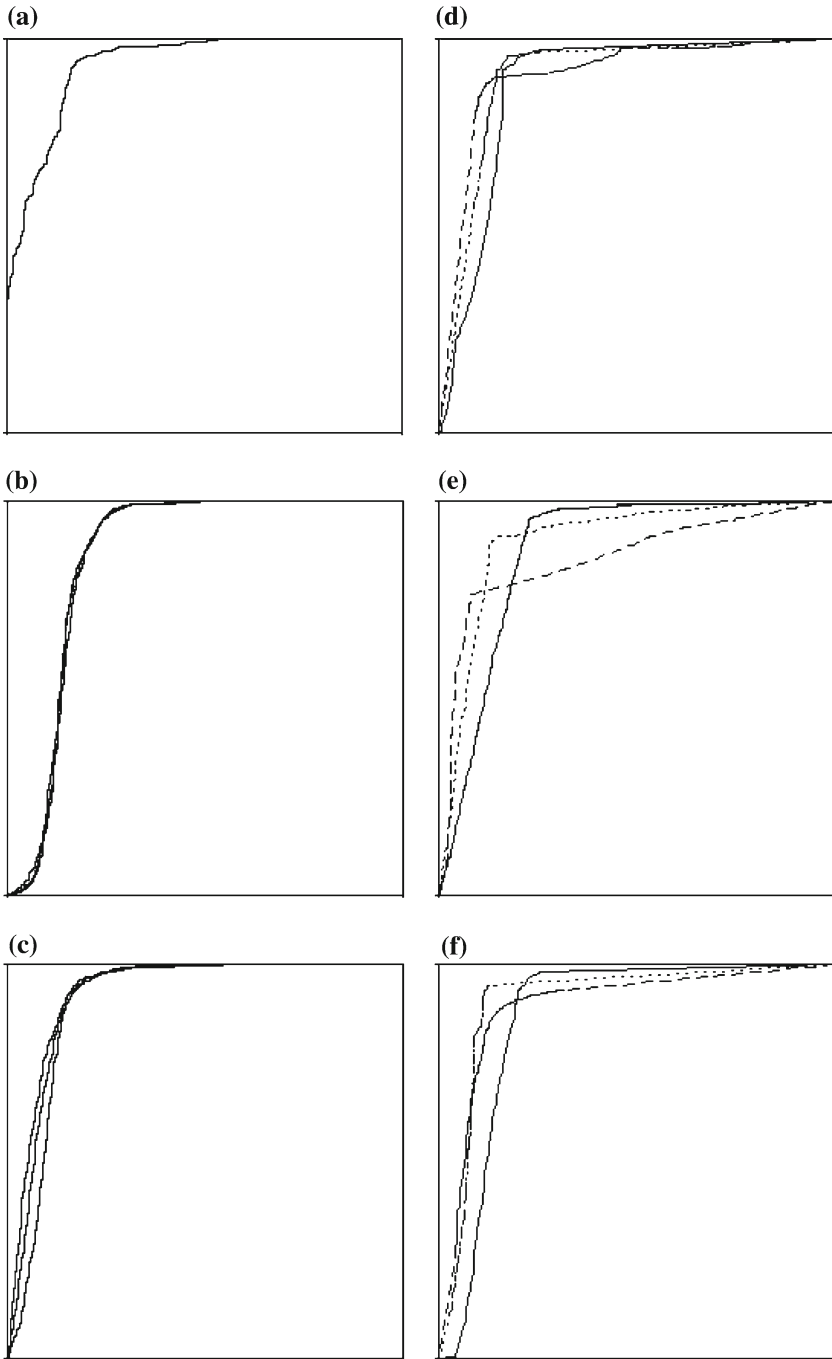
## 5 Conclusion

We have compared two major approaches to cost-sensitive classifier learning in this paper. Our analysis resulted in some interesting findings. Some classification methods that produce continuous probability estimates, such as Naïve Bayes, logistic regression, and backpropagation neural network, are relatively insensitive to the cost ratio when instance weighting is applied. On the other hand, symbolic classification methods, such as decision tree, decision rule, and decision table learners, are very sensitive to the cost ratio, meaning that the structures of the learned classifiers may change substantially as the cost ratio changes. This information helps in selecting the appropriate cost-sensitive classifier learning approach with regard to a base classification method.

There are several limitations in this paper, which can be addressed in future research. First, while we have evaluated six classification methods using 13 UCI datasets under three cost ratios, more extensive evaluations using more methods, larger real-world datasets, and more

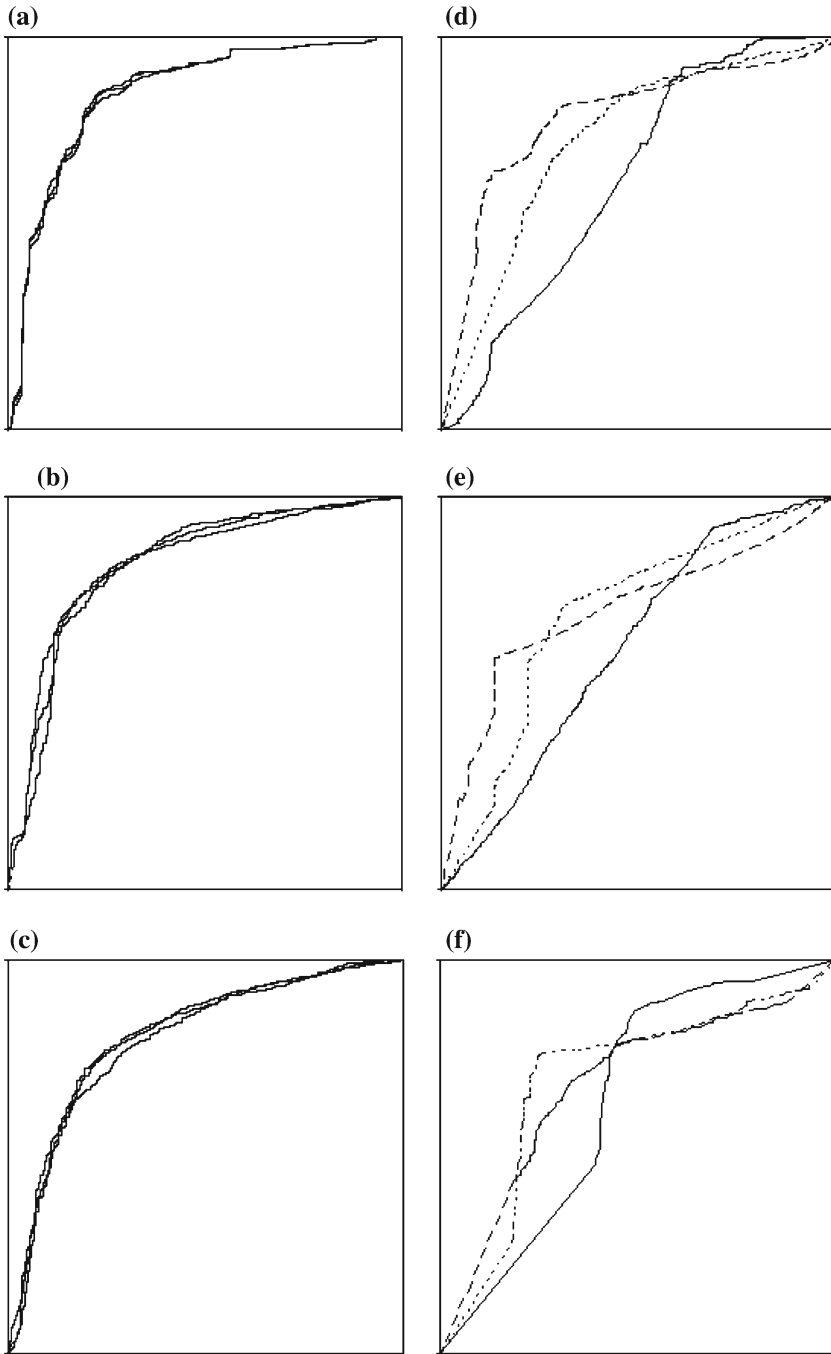


**Fig. 1** ROC curves for the Pima Indians diabetes dataset. **a** Naïve Bayes. **b** Logistic regression. **c** Backpropagation neural network. **d** C4.5 decision tree. **e** Ripper decision rule. **f** Decision table

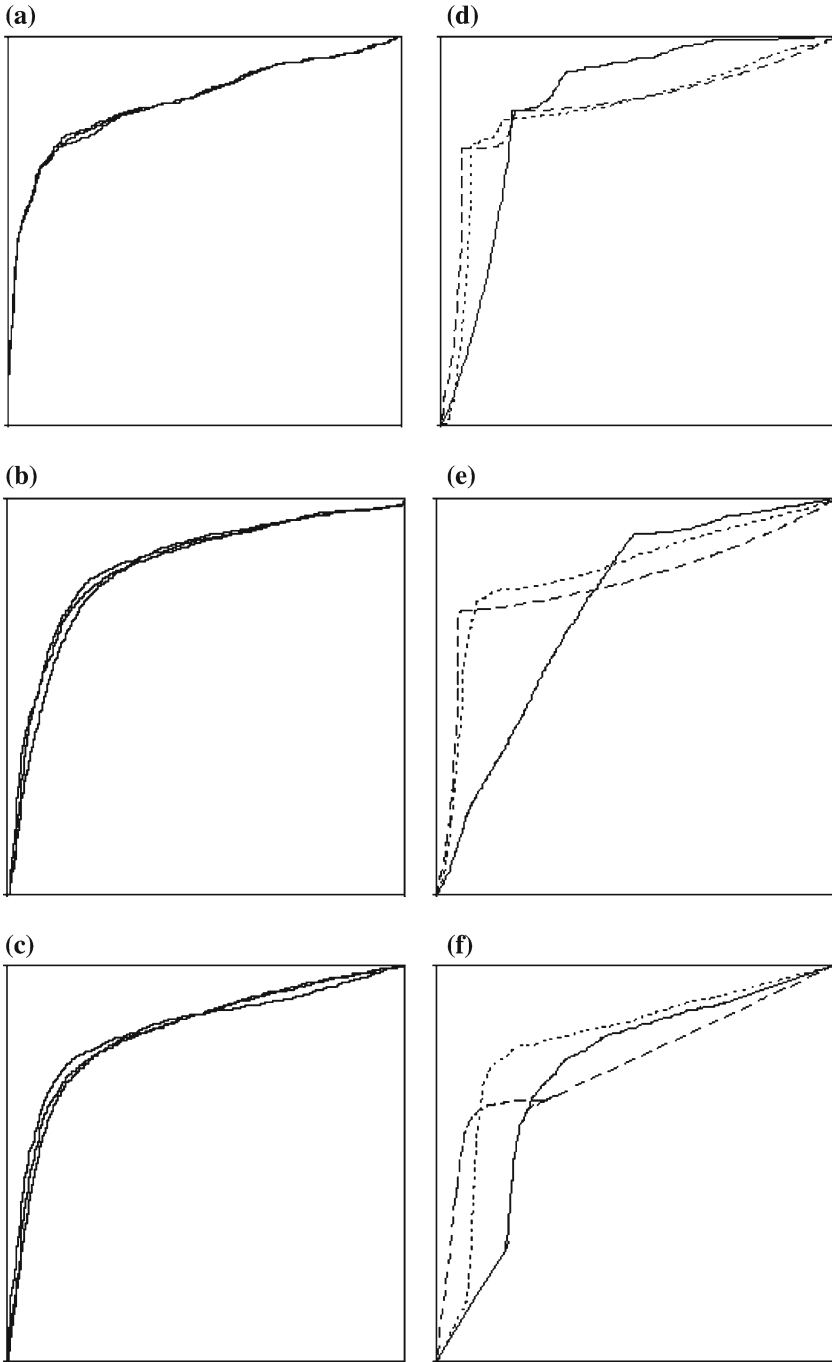


**Fig. 2** ROC curves for the ionosphere dataset. **a** Naïve Bayes. **b** Logistic regression. **c** Backpropagation neural network. **d** C4.5 decision tree. **e** Ripper decision rule. **f** Decision table

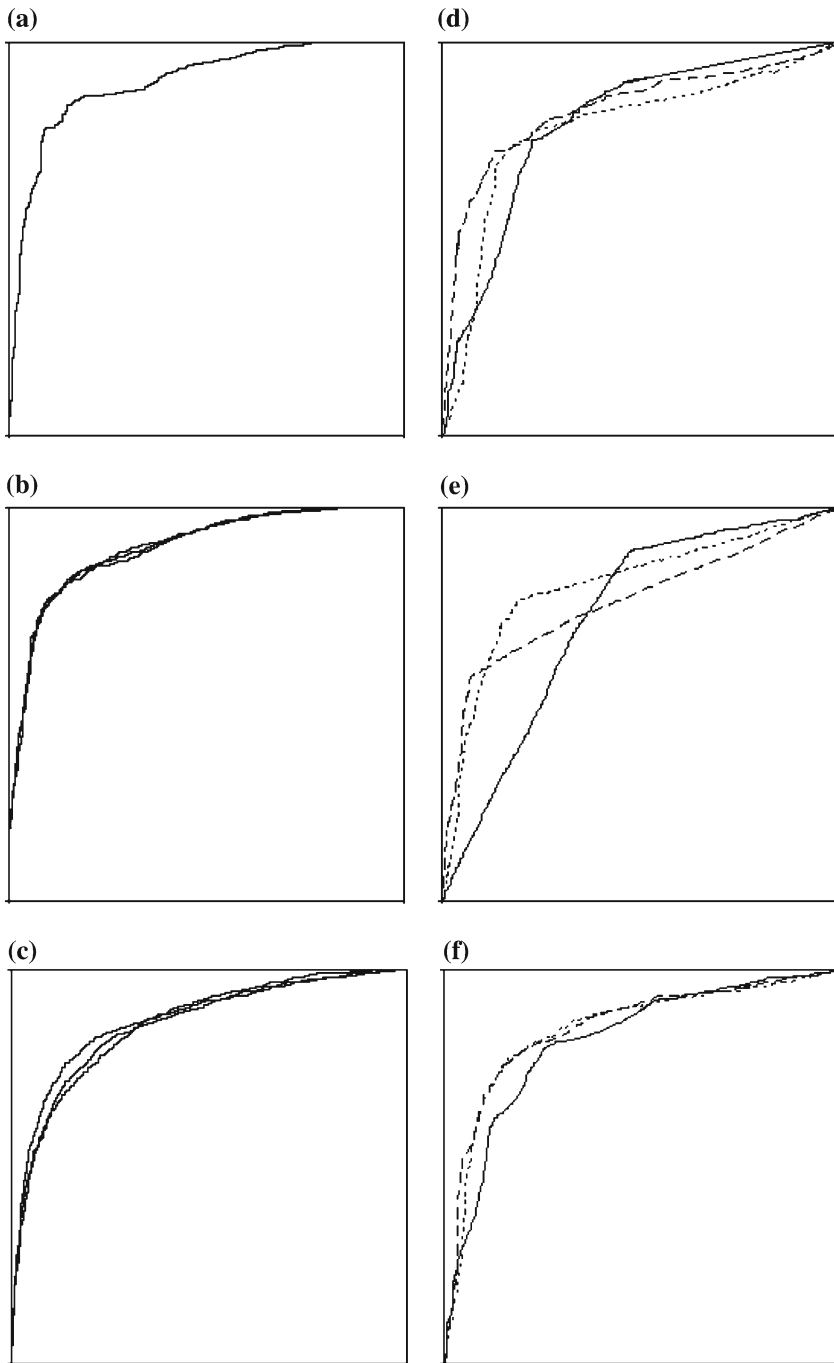




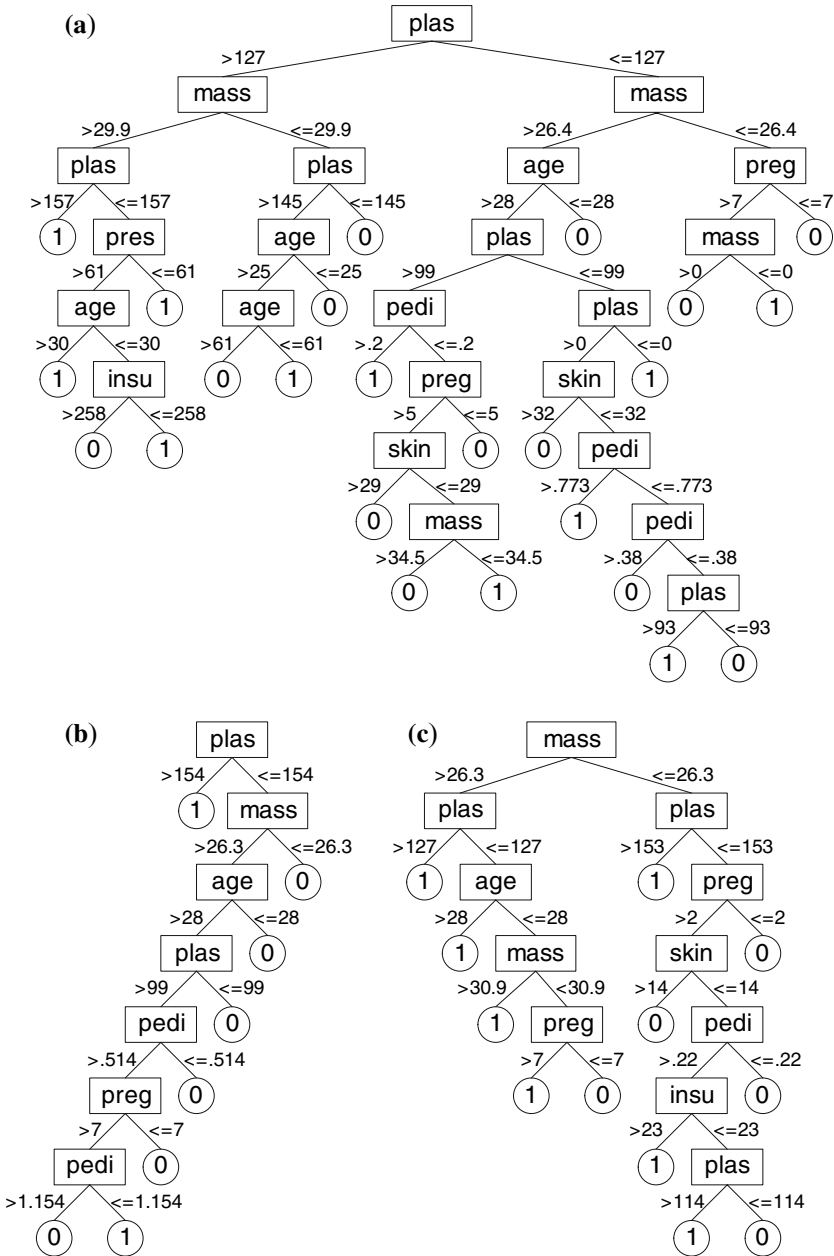
**Fig. 3** ROC curves for the hepatitis dataset. **a** Naïve Bayes. **b** Logistic regression **c** Backpropagation neural network. **d** C4.5 decision tree. **e** Ripper decision rule. **f** Decision table



**Fig. 4** ROC curves for the horse colic dataset. **a** Naïve Bayes. **b** Logistic regression. **c** Backpropagation neural network. **d** C4.5 decision tree. **e** Ripper decision rule. **f** Decision table



**Fig. 5** ROC curves for the heart disease dataset (Statlog Project). **a** Naive Bayes. **b** Logistic regression. **c** Backpropagation neural network. **d** C4.5 decision tree. **e** Ripper decision rule. **f** Decision table



**Fig. 6** Decision trees for the Pima Indians diabetes dataset **a** Cost ratio 1:1. **b** Cost ratio 1:5. **c** Cost ratio 5:1

extreme cost ratios can be conducted to validate the generalizability of our findings. Future research may also carry out a more thorough analytical explanation for the difference in the effects of the two cost-sensitive learning approaches in relation to base classification methods. Second, while we have only investigated binary classification problems, the evaluation needs to be generalized to multiple-class problems [16]. Third, while we have compared several

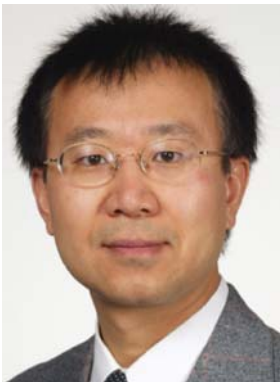
base classification methods, future research may further compare ensemble classification methods that combine multiple base classification methods via bagging, boosting, stacking, and cascading [21,33,36–38].

## References

1. Afifi AA, Clark V (1996) Computer-aided multivariate analysis, 3rd edn. Chapman & Hall, London
2. Blake CL, Merz CJ (1998) UCI repository of machine learning databases. <http://www.ics.uci.edu/~mllearn/MLRepository.html>
3. Bradley AP (1997) The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognit* 30(7): 1145–1159
4. Chan P, Stolfo S (1998) Towards scalable learning with non-uniform class and cost distributions: a case study in credit card fraud detection. In: Proceedings of the 4th international conference on knowledge discovery and data mining (KDD), New York, pp 164–168
5. Chawla N, Bowyer K, Hall L, Kegelmeyer W (2002) SMOTE: Synthetic minority over-sampling technique. *J Artif Intell Res* 16: 321–357
6. Cohen WW (1995) Fast effective rule induction. In: Proceedings of the 12th international conference on Machine Learning (ICML), Lake Tahoe, CA, pp 115–123
7. De Falco I, Della Cioppa A, Iazzetta A, Tarantino E (2005) An evolutionary approach for automatically extracting intelligible classification rules. *Knowl Inf Syst* 7(2): 179–201
8. Domingos P (1999) MetaCost: A general method for making classifiers cost sensitive. In: Proceedings of the 5th ACM SIGKDD international conference on knowledge discovery and data mining (KDD), San Diego, CA, pp 155–164
9. Domingos P, Pazzani M (1996) Beyond independence: conditions for the optimality of the simple Bayesian classifier. In: Proceedings of the 13th international conference on machine learning (ICML), Bari, Italy, pp 105–112
10. Drummond C, Holte R (2000) Exploiting the cost (in)sensitivity of decision tree splitting criteria. In: Proceedings of the 17th international conference on machine learning (ICML), Stanford, CA, pp 239–249
11. Drummond C, Holte R (2006) Cost curves: an improved method for visualizing classifier performance. *Mach Learn* 65(1): 95–130
12. Elkan C (2001) The foundations of cost-sensitive learning. In: Proceedings of the 17th international joint conference on artificial intelligence (IJCAI), Seattle, WA, pp 973–978
13. Fan W, Stolfo SJ, Zhang J, Chan PK (1999) AdaCost: Misclassification cost-sensitive boosting. In: Proceedings of the 16th international conference on machine learning (ICML), Bled, Slovenia, pp 97–105
14. Fawcett T (2003) ROC graphs: Notes and practical considerations for data mining researchers. HPL-2003-4, Intelligent Enterprise Technologies Lab Hewlett-Packard, PaloAlto
15. Gama J (2000) Iterative Bayes. *Intell Data Anal* 4(6): 475–488
16. Hand DJ, Till RJ (2001) A simple generalization of the area under the ROC curve to multiple class classification problems. *Mach Learn* 45(2): 171–186
17. Hidalgo JMG (2002) Evaluating cost-sensitive unsolicited bulk email categorization. In: Proceedings of the 2002 ACM symposium on applied computing (SAC), Madrid, Spain, pp 615–620
18. Hosmer DW, Lemeshow S (2000) Applied Logistic Regression, 2nd edn. Wiley, New York
19. Huang J, Ling CX (2005) Using AUC and accuracy in evaluating learning algorithms. *IEEE Trans Knowl Data Eng* 17(3): 299–310
20. Japkowicz N, Stephen S (2002) The class imbalance problem: a systematic study. *Intell Data Anal* 6(5): 429–449
21. Kim Y, Kim J (2004) Convex hull ensemble machine for regression and classification. *Knowl Inf Syst* 6(6): 645–663
22. Kohavi R (1995a) A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proceedings of the 14th international joint conference on artificial intelligence (IJCAI), Montreal, Quebec, Canada, pp 1137–1143
23. Kohavi R (1995b) The power of decision tables. In: Proceedings of the 8th European conference on machine learning (ECML), Heraclion, Crete, Greece, pp 174–189
24. Li T, Zhu S, Ogihara M (2006) Using discriminant analysis for multi-class classification: an experimental investigation. *Knowl Inf Syst* 10(4): 453–472
25. Lin FY, McClean S (2000) The prediction of financial distress using a cost sensitive approach and prior probability. In: Proceedings of the 17th international conference on machine learning (ICML) workshop on cost-sensitive learning, Stanford, CA, pp 84–88

26. Margineantu D (2002) Class probability estimation and cost-sensitive classification decisions. In: Proceedings of the 13th European conference on machine learning (ECML), Helsinki, Finland, pp 270–281
27. Provost F, Fawcett T, Kohavi R (1998) The case against accuracy estimation for comparing induction algorithms. In: Proceedings of the 15th international conference on machine learning (ICML), Madison, WI, pp 445–453
28. Quinlan JR (1993) C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo
29. Rumelhart DE, Hinton GE, William RJ (1986) Learning representations by back-propagating errors. *Nature* 323: 533–536
30. Sinha AP, May JH (2005) Evaluating and tuning predictive data mining models using receiver operating characteristic curves. *J Manage Inf Syst* 21(3): 249–280
31. Ting KM (2002) An instance-weighting method to induce cost-sensitive trees. *IEEE Trans Knowl Data Eng* 14(3): 659–665
32. Weiss SM, Kulikowski CA (1991) Computer systems that learn—classification and prediction methods from statistics, neural nets, machine learning, and expert system. Morgan Kaufmann, Palo Alto
33. Witten IH, Frank E (2005) Data mining: practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco
34. Zadrozny B, Langford J, Abe N (2003) Cost-sensitive learning by cost-proportionate example weighting. In: Proceedings of the 3rd IEEE international conference on data mining (ICDM), Melbourne, Florida, pp 435–442
35. Zhang J, Kang DK, Silvescu A, Honavar V (2006) Learning accurate and concise naïve Bayes classifiers from attribute value taxonomies and data. *Knowl Inf Syst* 9(2): 157–179
36. Zhao H, Ram S (2004) Constrained cascade generalization of decision trees. *IEEE Trans Knowl Data Eng* 16(6): 727–739
37. Zhao H, Sinha AP (2005) An efficient algorithm for generating generalized decision forests. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Syst Hum* 35(5): 754–762
38. Zhao H, Sinha AP, Ram S (2006) Elitist and ensemble strategies for cascade generalization. *J Database Manage* 17(3): 92–107

### Author's biography



**Huimin Zhao** is an Assistant Professor of Management Information Systems at the Sheldon B. Lubar School of Business, University of Wisconsin-Milwaukee. He earned his Ph.D. in Management Information Systems from the University of Arizona in 2002. His current research interests are in the areas of data mining and data integration. His research has been published in several journals, including *Communications of the ACM*, *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Systems, Man, and Cybernetics*, *Information Systems*, *Data and Knowledge Engineering*, *Journal of Management Information Systems*, and *Decision Support Systems*.