Many Are Better Than One: Improving Probabilistic Estimates from Decision Trees

Nitesh V. Chawla

Department of Computer Science and Engineering,
University of Notre Dame, Notre Dame, IN 46556
nchawla@cse.nd.edu

Abstract. Decision trees, a popular choice for classification, have their limitation in providing probability estimates, requiring smoothing at the leaves. Typically, smoothing methods such as Laplace or m-estimate are applied at the decision tree leaves to overcome the systematic bias introduced by the frequency-based estimates. In this work, we show that an ensemble of decision trees significantly improves the quality of the probability estimates produced at the decision tree leaves. The ensemble overcomes the myopia of the leaf frequency based estimates. We show the effectiveness of the probabilistic decision trees as a part of the Predictive Uncertainty Challenge. We also include three additional highly imbalanced datasets in our study. We show that the ensemble methods significantly improve not only the quality of the probability estimates but also the AUC for the imbalanced datasets.

1 Introduction

Inductive learning identifies relationships between the attributes values of training examples and the class of the examples, thus establishing a learned function. Decision trees [BFOS84, Qui87, Qui92] are a popular classifier for inductive inference. Decision trees are trained on examples comprised of a finite number of predicting attributes with class labels and a learned model is established based on tests on these attributes. This learning mechanism approximates discrete valued functions as the target attribute. The type of training examples applicable to decision trees are diverse, and could range from the consumer credit records to medical diagnostics.

Decision trees typically produce crisp classifications, that is the leaves carry decisions for individual classes. However, that is not sufficient for various applications. One can require a score output from a supervised learning method to rank order the instances. For instance, consider the classification of pixels in mammogram images as possibly cancerous [WDB^{+}93]. A typical mammography dataset might contain 98% normal pixels and 2% abnormal pixels. A simple default strategy of guessing the majority class would give a predictive accuracy of 98%. Ideally, a fairly high rate of correct cancerous predictions is required, while allowing for a small to moderate error rate in the majority class. It is more costly to predict a cancerous case as non-cancerous, than otherwise. Thus, a

J. Quiñonero-Candela et al. (Eds.): MLCW 2005, LNAI 3944, pp. 41–55, 2006.

probabilistic estimate or ranking of cancerous cases can be decisive for the practitioner. The cost of further tests can be decreased by thresholding the patients at a particular rank. Secondly, probabilistic estimates can allow one to threshold ranking for class membership at values < 0.5. Thus, the classes assigned at the leaves of the decision trees have to be appropriately converted to reliable probabilistic estimates. However, the leaf frequencies can require smoothing to improve the "quality" of the estimates [PD03, PMM$^+$94, SGF95, Bra97]. A classifier is considered to be well-calibrated if the predicted probability approaches the empirical probability as the number of predictions goes to infinity [GF83]. The quality of probability estimates, resulting from decision trees, has not been measured as is proposed in the PASCAL Challenge on Evaluating Predictive Uncertainty.

In this Chapter, we report on our experience in the NIPS 2004 Evaluating Predictive Uncertainty Challenge [QC05].

2 Probabilistic Decision Trees with C4.5

A decision tree is essentially in a disjunctive-conjunctive form, wherein each path is a conjunction of the attributes-values and the tree by itself is a disjunction of all these conjunctions. An instance arriving at the root node, takes the branch it matches based on the attribute-value test and moves down the tree following that branch. This continues until a path is established to a leaf node, providing the classification of the instance. If the target attribute is true for the instance, it is called a "true example"; otherwise it is called a 'negative example'. The decision tree learning aims to make "pure" leaves, that is leaves in which all the examples belong to one particular class. This growing procedure of the decision tree becomes its potential weakness for constructing probability estimates.

The leaf estimates, which are a natural calculation from the frequencies at the leaves, can be systematically skewed towards 0 and 1, as the leaves are essentially dominated by one class. For notational purposes, let us consider the confusion matrix given in Figure 1. TP is the number of true positives at the leaf, FP is the number of false positives, and C is the number of classes in the dataset. Typically, the probabilistic (frequency-based) estimate at a decision tree leaf is:

$$P(c|x) = TP/(TP + FP) \tag{1}$$

However, simply using the frequency derived from the correct counts of classes at a leaf might not give sound probabilistic estimates [PD03, ZE01]. A small leaf can potentially give optimistic estimates for classification purposes. For instance, the frequency based estimate will give the same weights to leaves with the following (TP, FP) distributions: $(5, 0)$ and $(50, 0)$. The relative coverage of the leaves and the original class distribution is not taken into consideration. Given the evidence, a probabilistic estimate of 1 for the $(5, 0)$ leaf is not very sound. Smoothing the frequency-based estimates can mitigate the aforementioned problem [PD03].

	Predicted Negative	Predicted Positive
Actual Negative	TN	FP
Actual Positive	FN	TP

Fig. 1. Confusion matrix

Aiming to perfectly classify the given set of training examples, a decision tree may overfit the training set. Overfitting is typically circumvented by deploying various pruning methodologies. Pruning involves evaluating every node of the decision tree and the subtree it may root, as a potential candidate for removal. A node is converted into a leaf node by assigning the most common classification associated at that node. But pruning deploys methods that typically maximize accuracies. Pruning is equivalent to coalescing different decision regions obtained by thresholding at feature values. This can result in coarser probability estimates at the leaves. While pruning improves the decision tree generalization, it can give poorer estimates as all the examples belonging to a decision tree leaves are given the same estimate. We used C4.5 decision trees for our experiments [Qui92].

2.1 Improving Probabilistic Estimates at Leaves

One way of improving the probability estimates given by an unpruned decision tree is to smooth them to make them less extreme. One can smooth these estimated probabilities by using the Laplace estimate [PD03], which can be written as follows:

$$P(c|x)_{Laplace} = (TP + 1)/(TP + FP + C) \tag{2}$$

Laplace estimate introduces a prior probability of $1/C$ for each class. Again considering the two pathological cases of $TP = 5$ and $TP = 50$, the Laplace estimates are 0.86 and 0.98, respectively, which are more reliable given the evidence.

However, Laplace estimates might not be very appropriate for highly imbalanced datasets [ZE01]. In that scenario, it could be useful to incorporate the prior of positive class to smooth the probabilities so that the estimates are shifted towards the minority class base rate (b). The m-estimate [Cus93] can be used as follows [ZE01]:

$$P(c|x)_m = (TP + bm)/(TP + FP + m) \tag{3}$$

where b is the base rate or the prior of positive class, and m is the parameter for controlling the shift towards b. Zadrozny and Elkan (2001) suggest using m, given b, such that $bm = 10$.

However, these smoothing estimates also cannot completely mitigate the effect of overfit and overgrown trees. We use ensemble methods to further "smooth" out the probability estimates at the leaves. Each leaf will potentially have a different $P(c|x)$ due to different training set composition. Averaging these estimates will improve the quality of the estimates, as it overcomes the bias introduced by the systematic error caused by having axis-parallel splits. The overfitting will also be countered as the variance component will be reduced by voting or averaging.

3 Ensemble Methods

> "To solve really hard problems, we'll have to use several different representations..... It is time to stop arguing over which type of pattern-classification technique is best..... Instead we should work at a higher level of organization and discover how to build managerial systems to exploit the different virtues and evade the different limitations of each of these ways of comparing things. [Min91]"

3.1 Random Subspaces

The random subspace method, introduced by Ho [Ho98], randomly selects different feature dimensions and constructs multiple smaller subsets. A classifier is then constructed on each of those subsets, and a combination rule is applied in the end for prediction on the testing set. For each random subspace a decision tree classifier is constructed. The random subspaces are particularly useful when there is a high redundancy in the feature space and for sparse datasets with small sample sizes.

The feature vector, where m is the number of features, can be represented as $X = (x_1, x_2, ..., x_{m-1})$. Then, multiple random subspaces of size $m \times p$ are selected, k times, where p is the size of the randomly selected subspace, $X_p^k\{(x_1, x_2, ..., x_p)|p < (m-1)\}$.

The hyperplanes constructed for each tree will be different, as each tree is essentially constructed from a randomly selected subset of features. The classification can either be done by taking the most popular class attached to the test example or by aggregating the probability estimate computed from each of the subspaces. Each tree has a different representation of the training set (different $P(x|c)$), thus resulting in a different function for $P(c|x)$ at each leaf. The classification assigned by the individual decision trees is effectively invariant for test examples that are different from the training examples in the unselected dimensions. The random subspaces are similar to the uncertainty analyses framework that simulates the distribution of an objective by sampling from the distribution of model inputs, and re-evaluating the objective for each selected set of model inputs.

We let the trees grow fully to get precise estimates, as the averaging would then reduce the overall variance in the estimates. Let $L_j(x)$ indicate the leaf that an example x falls into; let $P(c|L_j(x))$ indicate the probability that an example x belongs to class c at leaf L_j; let the number of trees in the ensemble be K.

$$\hat{P(c|L_j(x))} = \frac{P(c, L_j(x))}{\sum_{k=1}^{n_c} P(c_k, L_j(x))} \tag{4}$$

$$g_c(x) = \frac{1}{K} \sum_{i=1}^{K} \hat{P(c|L_j(x))} \tag{5}$$

$g_c(x)$ averages over probabilities conditioned on reaching a particular leaf (L). Each leaf is, in essence, defining its own region of probability distribution. Since, the trees are constructed from random subspaces, the regions can be of different shapes and sizes.

The random subspace method can be outlined as follows:

1. For each *k=1,2,..K*
   (a) Select a p dimensional random subspace, X_p^k, from X.
   (b) Construct the decision classifier, C_p^k using C4.5.
   (c) Smooth the leaf frequencies by Laplace or m-estimate.
2. Aggregate the probability estimates by each of the C^k classifiers. Output $g_c(x)$.

The individual classifiers can be weaker than the aggregate or even the global classifier. Moreover, the subspaces are sampled independently of each other. An aggregation of the same can lead to a reduction in the variance component of the error term, thereby reducing the overall error [DB98, Bre96]. There is a popular argument that diversity among the weak classifiers in an ensemble contributes to the success of the ensemble [KW03, Die00]. Classifiers are considered diverse if they disagree on the kind of errors they make. Diversity is an important aspect of the ensemble techniques — bagging, boosting, and randomization [Die00]. Diversity, thus, is a property of a group of classifiers. The classifiers might be reporting similar accuracies, but be disagreeing on their errors. One can, for example, construct a correlation measure among the rank-orders provided by each of the individual classifiers to get an estimate of diversity. In addition, the random subspace technique also counters the sparsity in the data, as the subspace dimensionality gets smaller but the training set size remains the same.

3.2 Bagging

Bagging, [Bre96], has been shown to improve classifier accuracy. Bagging basically aggregates predictions (by voting or averaging) from classifiers learned on multiple bootstraps of data. According to Breiman, bagging exploits the instability in the classifiers [Qui96], since perturbing the training set can produce different classifiers using the same learning algorithm, where the difference is in the resulting predictions on the testing set and the structure of the classifier. For instance, the decision trees learned from bootstraps of data will not only have different representations but can also have disagreements in their predictions. It is desired that the classifiers disagree or be diverse as the averaging or voting their predictions will lead ot a reduction in variance resulting in improved performance.

[Dom97] empirically tested two alternative theories supporting bagging: (1) bagging works because it approximates Bayesian model averaging or (2) it works because it shifts the priors to a more appropriate region in the decision space. The empirical results showed that bagging worked possibly because it counter-acts the inherent simplicity bias of the decision trees. That is, with M different bags, M different classifiers are learned, and together their output is more complex than that of the single learner. Bagging has been shown to aid improvement in the probability estimates [PD03, BK99]. The bagging procedure can be outlined as follows:

1. For each *k=1,2,..K*
   (a) Randomly select with replacement 100% of the examples from the training set X, to form a subsample X^k.
   (b) Construct a decision tree classifier, C^k, from X^k.
   (c) Smooth the leaf frequencies for each of the C^k classifiers by Laplace or m-estimate.
2. Aggregate the probability estimates by each of the C^k classifiers. Output $g_c(x)$.

4 Challenge Entry

The characteristics of the datasets prompted us to look at different stages of modeling. The high dimensionality introduced the sparsity in the feature space. Thus, we wanted to have feature selection as the first stage. As we will see in the subsequent sections, feature selection significantly reduced the number of relevant features to be used for modeling. Moreover, feature selection also curtailed the data sparsity issue. For feature selection we used information gain using entropy based discretization [FK93]. We then selected all the features with information gain greater than 0. We then generated ensembles with bagging and random subspaces using probabilistic decision trees as the base classifiers. Thus, our challenge entry comprised of the following steps. Note that our final best submission was with random subspaces.

1. Feature Selection
2. C4.5 Decision Trees
   – Fully grown and Laplace correction at the leaves
3. Ensembles
   – Random subspaces
   – Bagging

Challenge Datasets. The following classification datasets were provided for the Challenge.

1. Catalysis has 617 features and 1,173 examples in the final training set. The testing set has 300 examples.
2. Gatineau has 1,092 features and 5,176 examples in the final training set. It is a highly unbalanced dataset with the positive class comprising only 8.67% of the entire dataset. The testing set has 3000 examples.

4.1 Feature Selection Results

As we mentioned, our first step with both the datasets involved feature selection. We noted in our validation study that feature selection significantly improved the performance. Moreover, feature selection was particularly amenable for random subspaces as the feature relevance was now (approximately) uniformly spread. We, thus, selected the following number of features for both the datasets:

- 312 features for catalysis. Thus, almost 50% reduction in the total number of features. Figure 2 shows the information gain of the selected features for the catalysis dataset. There are not very high ranking features. The average information gain is 0.029 with a standard deviation of 0.0120.

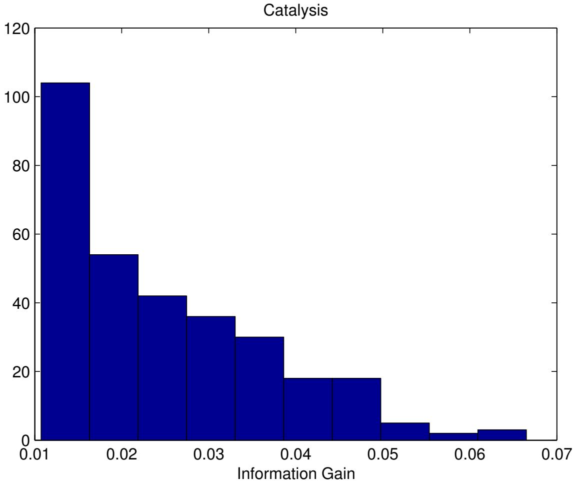


Fig. 2. Information Gain of the selected features for the catalysis dataset

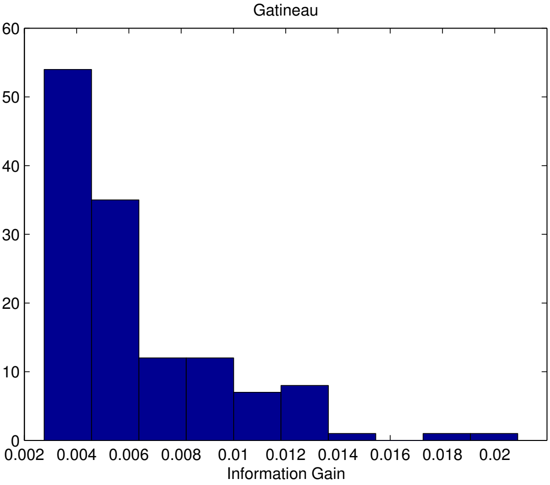


Fig. 3. Information Gain of the selected features for the gatineau dataset

- 131 features for gatineau. Thus, only 11% of the total number of features was retained. Figure 3 shows the information gain of the selected features for the gatineau dataset. The information gain of features is even lower for the gatineau dataset. The average information gain is 0.0061 and standard deviation is 0.0034.

4.2 Ensembles

After feature selection, we implemented random subspaces by setting $p = 50$ and $p = 25$ for catalysis and gatineau, respectively. We used $K = 30$ decision tree classifiers for both the datasets. We also learned 30 bags each for both the datasets. However, bagging did not improve the performance as much as the random subspaces. We attribute that to the sparsity of the dataset. Tables 1 and 2 show the final results of our entries. It is evident from the Tables that feature selection formed an important element of our entry.

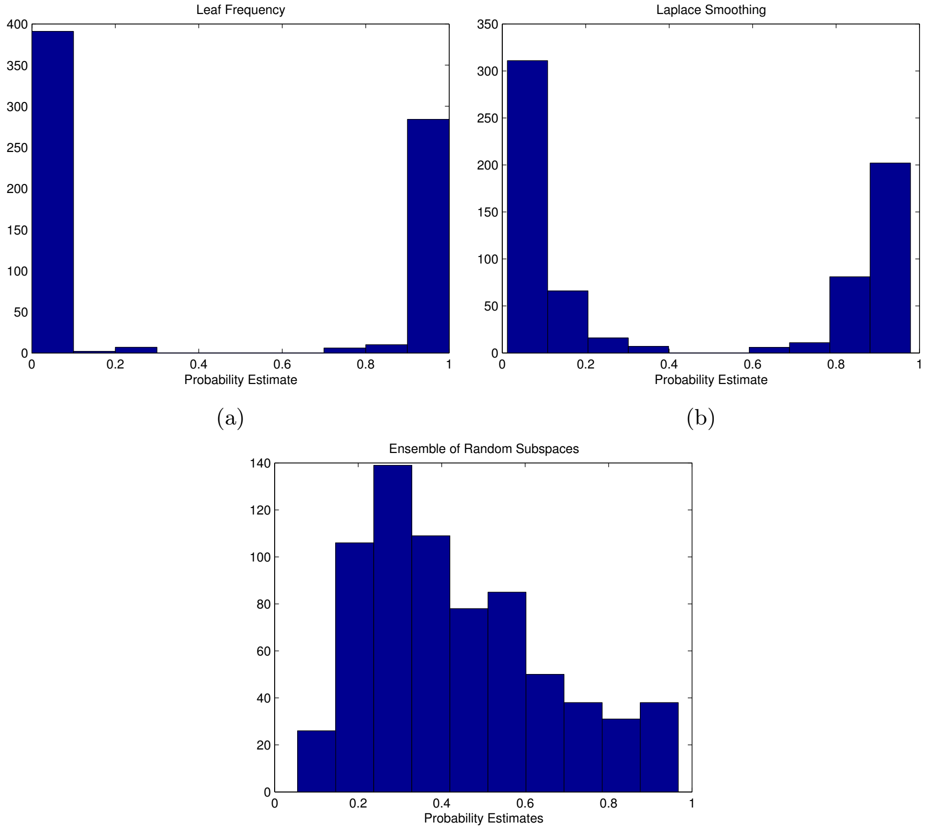
Figures 4 and 5 show the probability distributions achieved after applying the ensemble methods. It is evident from the Figures that the ensemble methods improved the probability mass, and overcame the limitations of skewed probability estimates from the decision tree leaves. The ensembles successfully overcame the bias and variance limitations associated with the decision tree leaves estimates. As expected, the gatineau dataset shows an interesting trend with Laplace smoothing. Gatineau is highly imbalanced, thus the Laplace estimate, which is trying to correct the probability estimate by adding $\frac{1}{C}$ is still being biased towards the minority class. Thus, there is no significant difference between the probability estimates from leaf frequencies and the ones generated from from applying Laplace smoothing. However, the ensembles improve the quality of the

Table 1. Challenge Results for the Catalysis dataset. FS: Feature Selection; RS: Random Subspaces. This entry was ranked Fourth at the time of Challenge termination in December, 2004.

Method	NLP	OIL	LIFT
FS + RS	2.41e-1	2.714e-1	2.371e-1
RS	2.485e-1	2.843e-1	2.534e-1
FS + Bagging	2.49e-1	3e-1	2.565e-1
Bagging	2.51e-1	2.971e-1	2.649e-1

Table 2. Challenge Results for the Gatineau dataset. FS: Feature Selection; RS: Random Subspaces. This entry was ranked First at the time of Challenge termination in December, 2004.

Method	NLP	OIL	LIFT
FS + RS	1.192e-1	8.7e-2	7.408e-1
RS	1.228e-1	8.7e-2	7.555e-1
FS + Bagging	1.193e-1	8.867e-2	7.311e-1
Bagging	1.229e-1	8.7e-2	7.506e-1



(a)

(b)

(c)

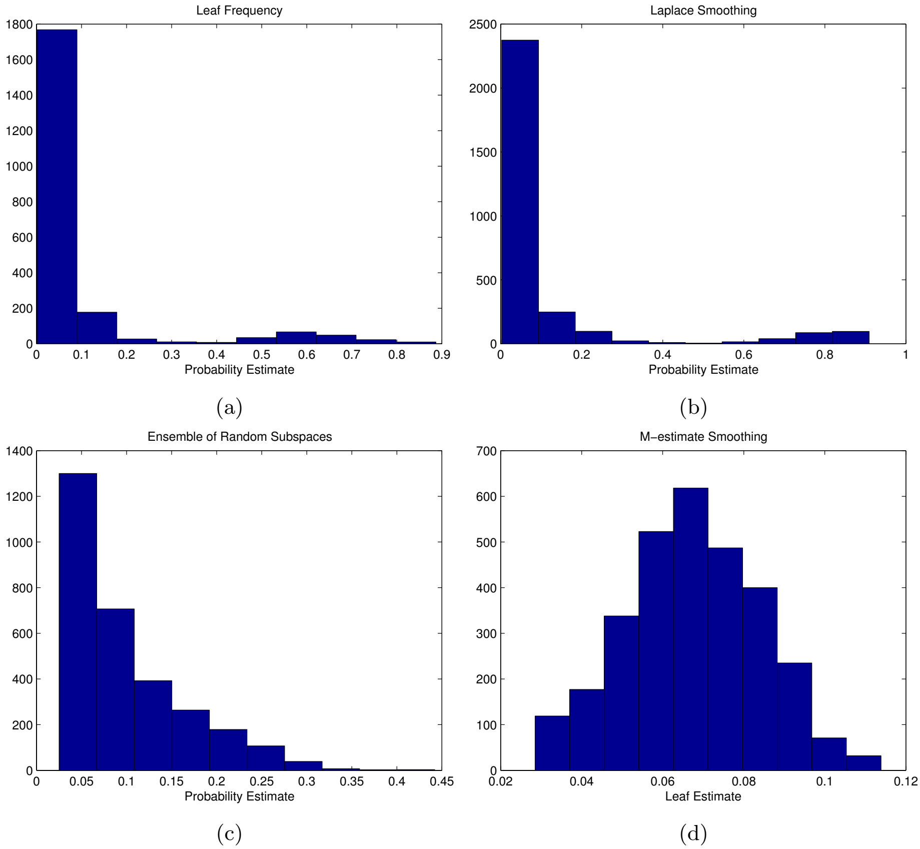
Fig. 4. a) Probability Distribution using the leaf frequencies as estimates. b) Probability distribution by smooting leaf frequencies using Laplace estimates. c) Probability Distribution using random subspaces as ensemble methods. The probabilities are $g_c(x)$ that are averaged from the smoothed leaf estimates.

Table 3. Post-Challenge Entry. These are our best results so far.

Dataset	Method	NLP	OIL	LIFT
Catalysis	FS + RS	2.4076e-1	2.7e-1	2.2874e-1
Gatineau	FS + RS	1.2475e-1	0.87e-1	7.4835e-1

estimates. Then, we applied m-estimate smoothing by setting the base rate to compensate for the high class imbalance. As one can see, the resulting probability estimates follow a much better distribution.

As the Post-challenge participation, we increased the ensemble size and implemented $m - estimate$ for smoothing the decision tree leaves. This further improved our performance on the gatineau dataset, while maintaining similar performance (marginally better) on the catalysis dataset. Table 3 contains those results.



(a) (b)

(c) (d)

Fig. 5. a) Probability distribution using the leaf frequencies as estimates. b) Probability distribution by smooting leaf frequencies using Laplace estimates. c) Probability distribution using random subspaces as ensemble methods. The probabilities are $g_c(x)$ that are averaged from the smoothed leaf estimates. d) Probability Distribution using random subspaces as ensemble methods. The probabilities are $g_c(x)$ that are averaged from the m-estimate smoothed leaf probability estimates.

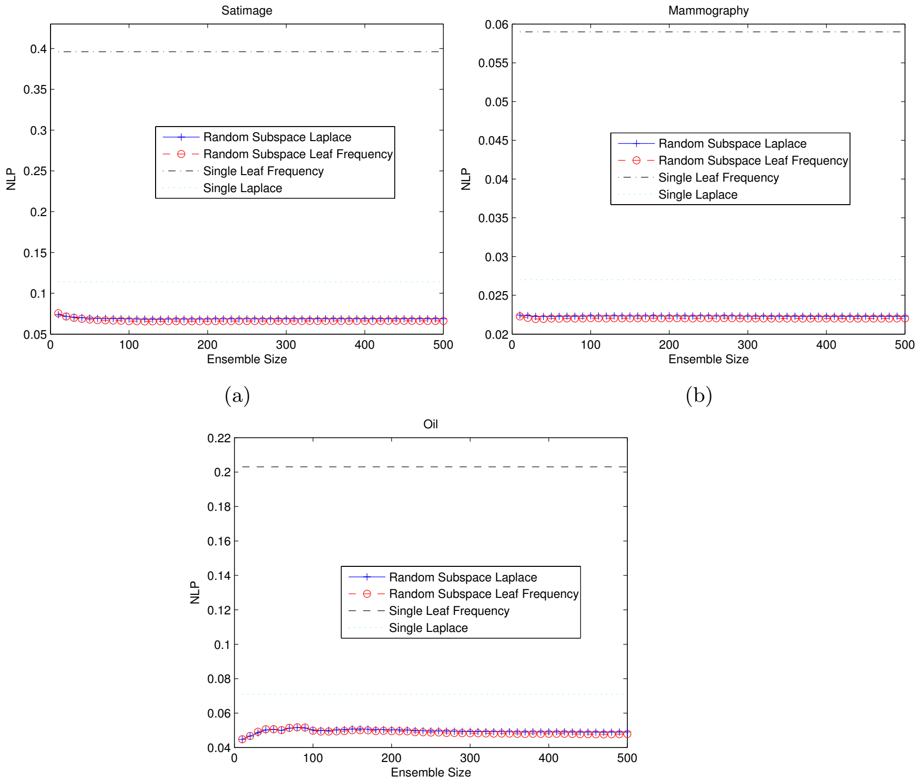
5 Experiments with Imbalanced Datasets

A dataset is imbalanced if the classes are not approximately equally represented [CHKK02, JS02]. There have been attempts to deal with imbalanced datasets in domains such as fraudulent telephone calls [FP96], telecommunications management [ESN96], text classification [LR94, DPHS98, MG99, Coh95] and detection of oil spills in satellite images [KHM98].

Distribution/cost sensitive applications can require a ranking or a probabilistic estimate of the instances. For instance, revisiting our mammography data example, a probabilistic estimate or ranking of cancerous cases can be decisive for the practitioner. The cost of further tests can be decreased by thresholding

Table 4. Dataset distribution

Dataset	Majority Class	Minority Class
Satimage	5809	626
Mammography	10923	260
Oil	896	41



(a)

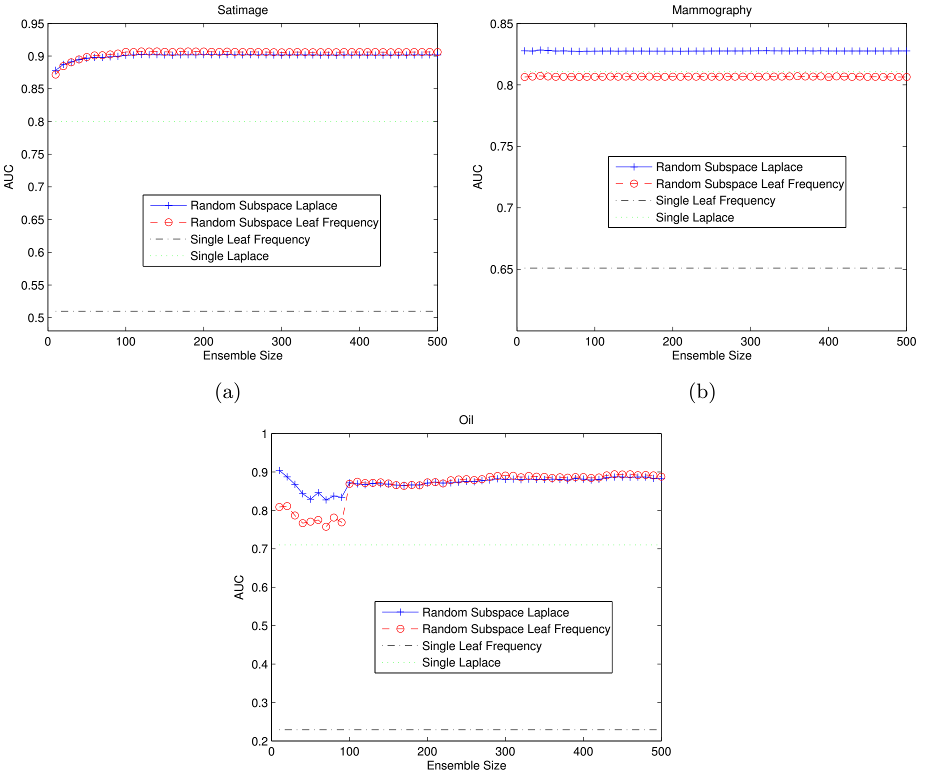
(b)

(c)

Fig. 6. a) NLP as the ensemble size varies for the Satimage dataset. b) NLP as the ensemble size varies for the Mammography dataset. c) NLP as the ensemble size varies for the Oil dataset.

the patients at a particular rank. Secondly, probabilistic estimates can allow one to threshold ranking for class membership at values < 0.5. Hence, the classes assigned at the leaves of the decision trees have to be appropriately converted to probabilistic estimates [PD03]. This brings us to another question: *What is the right probabilistic estimate for imbalanced datasets?*

We added the following three imbalanced datasets to our study for empirically evaluating the effect of the smoothing parameters and ensembles. These datasets



(a)

(b)

(c)

Fig. 7. a) AUC as the ensemble size varies for the Satimage dataset. b) AUC as the ensemble size varies for the Mammography dataset. c) AUC as the ensemble size varies for the Oil dataset.

vary extensively in their size and class proportions, thus offering different. Table 4 shows the class distribution.

1. The Satimage dataset [BM98] has 6 classes originally. We chose the smallest class as the minority class and collapsed the rest of the classes into one as was done in [PFK98]. This gave us a skewed 2-class dataset, with 5809 majority class samples and 626 minority class samples.
2. The Oil dataset was provided by Robert Holte [KHM98]. This dataset has 41 oil slick samples and 896 non-oil slick samples.
3. The Mammography dataset [WDB^{+}93] has 11,183 samples with 260 calcifications. If we look at predictive accuracy as a measure of goodness of the classifier for this case, the default accuracy would be 97.68% when every sample is labeled non-calcification. But, it is desirable for the classifier to predict most of the calcifications correctly.

We report the same NLP loss estimate as used in the Challenge. In addition, we also report the Area Under the ROC Curve. The purpose of the AUC is to see if ranking of the exemplars is affected by improving the quality of the probability estimates. Figure 6 shows the trend of NLP as the ensemble size varies. The single tree estimates are much weaker when no smoothing is applied, as one might expect. This is particularly more critical for the imbalanced datasets, when the positive class is more relevant. However, the ensembles are sufficient in overcoming the bias and variance at the leaves without using the Laplace estimate. The same trend is observed for all the three imbalanced datasets.

Figure 7 shows the AUC's for the three datasets. Again, the single tree AUC using leaf frequencies as the probability estimates is very low, which is also affirmed by the high NLP. And the AUC's significantly improve with ensembles and Laplace smoothing. This results show that there is a relationship between the AUC's and the quality of the probability estimates as established by the NLP. Improving the quality of the estimates not only provides a better spread of probabilities but also improves the ranking of exemplars, thus impacting the AUC.

6 Summary

We summarized our entry for the NIPS 2004 Evaluating Predictive Uncertainty Challenge. We show that ensembles of decision trees, particularly random subspaces, can generate good probability estimates by smoothing over the leaf frequencies. The ensembles overcome the bias and variance arising from the homogeneous and small decision tree leaves. We show that decision trees are a viable strategy for probability estimates and rank among the best methods reported in the challenge. The ensembles are able to overcome the bias in estimates arising from the axis-parallel splits of decision trees, resulting in smoother estimates. We also saw that the prior smoothing at the leaves using Laplace estimate did not offer much gain with ensembles. However, Laplace smoothing did provide significant improvements over just using leaf frequencies.

We also added three highly imbalanced datasets to our study. We show that the rank-order of the exemplars and the resulting AUC is related to the quality of the probability estimates. For most of the applications requiring imbalanced datasets, the resulting rank-order of examples or $P(Xp > Xn)$ can be very important, where Xp is the positive class example. Thus, having reliable probability estimates is important for an improved rank-ordering.

As a part of ongoing work, we are investigating evolutionary techniques to carefully prune away members of the ensemble that don't contribute to the qualaity of the final probability estimation [SC05]. It is important for the classifiers in an ensemble to assist each other and cancel out their errors, resulting in higher accuracy. If all the classifiers are in complete agreement, then the averaging will not result in any changes in the probability estimates (each estimate will be the same). Thus, we would like to identify the more "collaborative" members of the ensemble, and assign higher weights to their predictions. We can, for example, select those classifiers from the ensemble that particularly optimize on the NLP loss function.

References

[BFOS84] L. Breiman, J.H. Friedman, R.A. Olshen, and P.J. Stone. *Classification and Regression Trees.* Wadsworth International Group, Belmont, CA, 1984.

[BK99] E. Bauer and R. Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting and variants. *Machine Learning*, 36(1,2), 1999.

[BM98] C.L. Blake and C.J. Merz. UCI Repository of Machine Learning Databases http://www.ics.uci.edu/∼mlearn/∼MLRepository.html. Department of Information and Computer Sciences, University of California, Irvine, 1998.

[Bra97] A. P. Bradley. The Use of the Area Under the ROC Curve in the Evaluation of Machine Learning Algorithms. *Pattern Recognition*, 30(6):1145–1159, 1997.

[Bre96] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[CHKK02] N.V. Chawla, L.O. Hall, Bowyer K.W., and W.P. Kegelmeyer. SMOTE: Synthetic Minority Oversampling TEchnique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[Coh95] W. Cohen. Learning to Classify English Text with ILP Methods. In *Proceedings of the 5th International Workshop on Inductive Logic Programming*, pages 3–24. Department of Computer Science, Katholieke Universiteit Leuven, 1995.

[Cus93] J. Cussents. Bayes and pseudo-bayes estimates of conditional probabilities and their reliabilities. In *Proceedings of European Conference on Machine Learning*, 1993.

[DB98] B. Draper and K. Baek. Bagging in computer vision. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages 144–149, 1998.

[Die00] T. Dietterich. An empirical comparison of three methods for constructing ensembles of decision trees: bagging, boosting and randomization. *Machine Learning*, 40(2):139 – 157, 2000.

[Dom97] P. Domingos. Why does bagging work? a bayesian account and its implications. In *Proceedings of Third International Conference Knowledge Discovery and Data Mining*, pages 155–158, 1997.

[DPHS98] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive Learning Algorithms and Representations for Text Categorization. In *Proceedings of the Seventh International Conference on Information and Knowledge Management.*, pages 148–155, 1998.

[ESN96] J. Ezawa, K., M. Singh, and W. Norton, S. Learning Goal Oriented Bayesian Networks for Telecommunications Risk Management. In *Proceedings of the International Conference on Machine Learning, ICML-96*, pages 139–147, Bari, Italy, 1996. Morgan Kauffman.

[FK93] U. Fayyad and R. Kohavi. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of 13th International Joint Conference on Artificial Intelligence*, pages 1022–1027, 1993.

[FP96] T. Fawcett and F. Provost. Combining Data Mining and Machine Learning for Effective User Profile. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, pages 8–13, Portland, OR, 1996. AAAI.

[GF83] M. De Groot and S. Fienberg. The Comparison and Evaluation of Forecasters. *Statistician*, 32:12 – 22, 1983.

[Ho98] T. K. Ho. The random subspace method for constructing decision trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998.

[JS02] N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5), 2002.

[KHM98] M. Kubat, R. Holte, and S. Matwin. Machine Learning for the Detection of Oil Spills in Satellite Radar Images. *Machine Learning*, 30:195–215, 1998.

[KW03] L. Kuncheva and C. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51:181–207, 2003.

[LR94] D. Lewis and M. Ringuette. A Comparison of Two Learning Algorithms for Text Categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 81–93, 1994.

[MG99] D. Mladenić and M. Grobelnik. Feature Selection for Unbalanced Class Distribution and Naive Bayes. In *Proceedings of the 16th International Conference on Machine Learning.*, pages 258–267. Morgan Kaufmann, 1999.

[Min91] M. Minsky. Logical versus analogical, symbolic versus connectionist, neat versus scruffy. *AI Magazine*, 12, 1991.

[PD03] F. Provost and P. Domingos. Tree induction for probability-based rankings. *Machine Learning*, 52(3), 2003.

[PFK98] F. Provost, T. Fawcett, and R. Kohavi. The Case Against Accuracy Estimation for Comparing Induction Algorithms. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 445–453, Madison, WI, 1998. Morgan Kauffmann.

[PMM$^+$94] M. Pazzani, C. Merz, P. Murphy, K. Ali, T. Hume, and C. Brunk. Reducing misclassification costs. In *Proceedings of the Eleventh International Conference on Machine Learning*, pages 217–215, 1994.

[QC05] Joaquin Quiñonero-Candela. Evaluating Predictive Uncertainty Challenge, 2005. `predict.kyb.tuebingen.mpg.de`.

[Qui87] J.R. Quinlan. Simplifying decision trees. *International Journal of Man Machine Studies, V.27*, pages 227–248, 1987.

[Qui92] J.R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, CA, 1992.

[Qui96] J. R. Quinlan. Bagging, boosting, and C4.5. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 725–730, 1996.

[SC05] J. Sylvester and N. V. Chawla. Evolutionary ensembles: Combining learning agents using genetic algorithms. In *AAAI Workshop on Multiagent Learning*, pages 46–51, 2005.

[SGF95] P. Smyth, A. Gray, and U. Fayyad. Retrofitting decision tree classifiers using kernel density estimation. In *Proceedings of the Twelth International Conference on Machine Learning*, pages 506–514, 1995.

[WDB$^+$93] K. Woods, C. Doss, K. Bowyer, J. Solka, C. Priebe, and P. Kegelmeyer. Comparative Evaluation of Pattern Recognition Techniques for Detection of Microcalcifications in Mammography. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(6):1417–1436, 1993.

[ZE01] B. Zadrozny and C. Elkan. Learning and making decisions when costs and probabilities are both unknown. In *Proceedings of the Seventh International Conference on Knowledge Discovery and Data Mining*, 2001.