Investigation and Reduction of Discretization Variance in Decision Tree Induction

Pierre Geurts and Louis Wehenkel

University of Liège, Department of Electrical and Computer Engineering
Institut Montefiore, Sart-Tilman B28, B4000 Liège, Belgium

Abstract. This paper focuses on the variance introduced by the discretization techniques used to handle continuous attributes in decision tree induction. Different discretization procedures are first studied empirically, then means to reduce the discretization variance are proposed. The experiment shows that discretization variance is large and that it is possible to reduce it significantly without notable computational costs. The resulting variance reduction mainly improves interpretability and stability of decision trees, and marginally their accuracy.

1 Variance in Decision Tree Induction

Decision trees ([1], [2]) can be viewed as models of conditional class probability distributions. Top down tree induction recursively splits the input space into non overlapping subsets, estimating class probabilities by frequency counts based on learning samples belonging to each subset. Tree variance is the variability of its structure and parameters resulting from the randomness of the learning set; it translates into prediction variance yielding classification errors.

In regression models, prediction variance can be easily separated from bias, using the well-known bias/variance decomposition of the expected square error. Unfortunately, there is no such decomposition for the expected error rates of classification rules (e.g. see [3, 4]). Hence, we will look at decision trees as multidimensional regression models for the conditional class probability distributions and evaluate their variance by the regression variance resulting from the estimation of these probabilities. Denoting by $\hat{P}_N(C_i|x)$ the conditional class probability estimates given by a tree built from a random learning set of size N at a point x of the input space, we can write this variance (for one class C_i) :

$$Var(\hat{P}_N(C_i|.)) = E_X\{E_{LS}\{(\hat{P}_N(C_i|x) - E_{LS}\{\hat{P}_N(C_i|x)\})^2\}\}, \qquad (1)$$

where the innermost expectations are taken over the set of all learning sets of size N and the outermost expectation is taken over the whole input space. Friedman [4] has studied the impact of this variance on classification error rates, concluding to the greater importance of this term as compared to bias.

Sources of Tree Variance. A first (important) variance source is related to the need for discretizing continuous attributes by choosing thresholds. In

R.López de Mántaras, E.Plaza (Eds.): ECML 2000, LNAI 1810, pp. 162-170, 2000.

local discretization, such thresholds are determined on the subset of learning samples which reach a particular test node. Since many test nodes correspond to small sample sizes (say, less than 200), we may expect high threshold variance unless particular care is taken. We will show that classical discretization methods actually lead to very high threshold variance, even for large sample sizes.

Another variance source is the variability of tree structure, i.e. the chosen attribute at a particular node, which also depends strongly on the learning set. For example, for the OMIB database (see appendix), 50 out of 50 trees built from randomly selected learning sets of size 500 agreed on the choice of the root attribute, but only 27 at the left successor and only 22 at the right successor.

A last variance source relates to the estimation of class probabilities, but this effect turns out to be negligible (for pruned trees). Indeed, fixing tree structure and propagating different random learning sets to re-estimate class probabilities and determine the variance, yields with the OMIB database a variance of 0.004, which has to be compared to a total variance of 0.05 (see Table 2).

To sum up, tree variance is important and mainly related to the local node splitting technique which determines the tree structure. The consequences are : (i) questionable interpretability (we can not really trust the choice of attributes and thresholds); (ii) poor estimates of conditional class probabilities; (iii) sub-optimality in terms of classification accuracy, but we have still to prove this.

Reduction of Tree Variance. In the literature, two approaches have been proposed : pruning and averaging. Pruning is computationally inexpensive, reduces complexity significantly and variance to some extent, but also increases bias. Thus, it improves only slightly interpretability and accuracy. Averaging reduces variance and indirectly bias, and hence leads in some problems to spectacular improvements in accuracy. Unfortunately, it destroys the main attractive features of decision trees, i.e. computational efficiency and interpretability.

It is therefore relevant to investigate whether it is possible to reduce decision tree variance without jeopardizing efficiency and interpretability. In what follows, we will focus on the local discretization technique used to determine thresholds for continuous attributes and investigate its variance and ways to reduce it. We show that this variance may be very large, even for reasonable sample sizes, and may be reduced significantly without notable computational costs.

In the next section we will study empirically the threshold variance of three different discretization techniques, then propose a modification of the classical method in order to reduce threshold variance significantly. In the following section we will assess the resulting impact in terms of global tree performance, comparing our results with those obtained with tree bagging [5].

2 Evaluating and Reducing Threshold Variance

Classical Local Discretization Algorithm. In the case of numerical attributes, the first stage of node splitting consists in selecting a discretization threshold for each attribute. Denoting by a an attribute and by $a(o)$ its value for a given sample o, this amounts to selecting a threshold value a_{th} in order to split

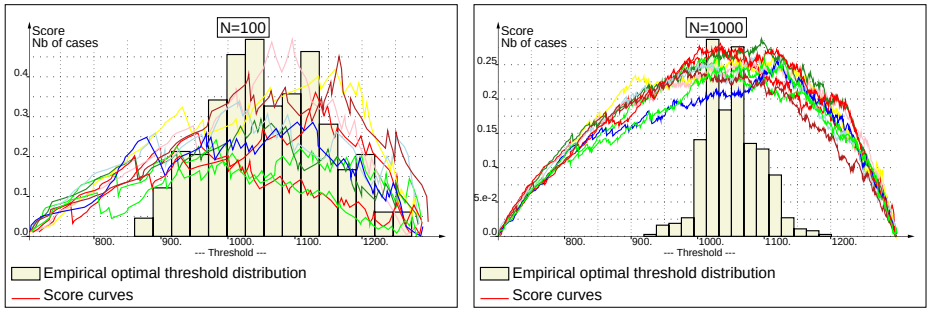

Fig. 1. 10 score curves and empirical optimal threshold distribution for learning sets of size 100 (left) and 1000 (right). OMIB database, attribute Pu.

the node according to the test $T(o) \equiv [a(o) < a_{\text{th}}]$. To determine a_{th}, normally a search procedure is used so as to maximize a score measure evaluated using the subset $ls = \{o_1, o_2, ..., o_n\}$ of learning samples which reach the node to split. Supposing that the ls is already sorted by increasing values of a, most discretization techniques exhaustively enumerate all thresholds $\frac{a(o_i)+a(o_{i+1})}{2} (i = 1...n-1)$. Denoting the observed classes by $C(o_i), (i = 1, \ldots, n)$, the score measures how well the test $T(o)$ correlates with the class $C(o)$ on the sample ls. In the literature, many different score measures have been proposed. In our experiments we use the following normalization of Shannon information (see [6, 7] for a discussion)

$$C_C^T = \frac{2I_C^T}{H_C + H_T}, \tag{2}$$

where H_C denotes class entropy, H_T test entropy (also called split information by Quinlan), and I_C^T their mutual information.

Figure 1 represents the relationship between C_C^T and the discretization threshold, for the OMIB database (see appendix). Each curve shows the variation of score in terms of discretization threshold for a given sample. The histograms beneath the curves correspond to the sampling distribution of the global maxima of these curves (i.e. the threshold selected by the classical method). One observes that even for large sample sizes (right hand curves), the variance of the "optimal" threshold determined by the classical method remains rather high.

Figure 2 shows results for sample sizes $N \in [50; 2500]$ obtained on the GAUSSIAN database according to the following procedure : (i) for each value of N, 100 samples $ls_1, \ldots, ls_{100}$ of size N are drawn; (ii) for each ls_i the threshold a_{th}^i maximizing $\hat{C}_C^T(ls_i)$ is computed, as well as left and right hand estimates of conditional class probabilities. The graphs of Figure 2 plot the averages ($\pm$ standard deviation) of these 100 numbers as a function of N; it highlights clearly how slowly threshold variance decreases with sample size.

Alternative Discretization Criteria. To assess whether the information theoretic measure is responsible for the threshold variance, we have compared it with two alternative criteria : (i) **Kolmogorov-Smirnov measure** (see [8]); (ii) **Median**, a naïve method discretizing at the (local) sample median.

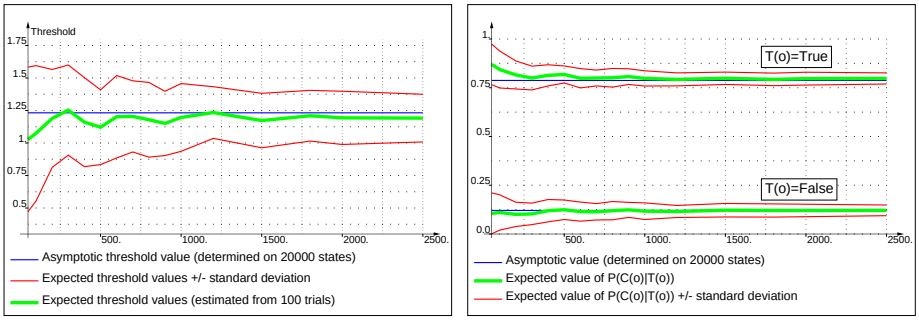

Fig. 2. Expected threshold values and standard deviation (left); Class probability estimates and standard deviation (right). Attribute a_1 of GAUSSIAN database.

Table 1. OMIB database, asymptotic value of a_{th}=1057, $\sigma\ attribute = 170$

method	$N = 50$			$N = 500$			$N = 2000$		
	$\sigma_{a_{th}}$	$b(a_{th})$	$Var(\hat{P})$	$\sigma_{a_{th}}$	$b(a_{th})$	$Var(\hat{P})$	$\sigma_{a_{th}}$	$b(a_{th})$	$Var(\hat{P})$
classic	91.0	-15.6	0.01335	55.4	-1.5	0.00383	36.8	-8.6	0.00138
Kolmogorov	59.3	-13.8	0.00900	26.6	-13.5	0.00126	18.7	-18.6	0.00042
median	38.2	-55.9	0.00772	13.1	-59.2	0.00095	6.1	-58.8	0.00016
averaging	34.6	-49.3	0.00945	20.3	-20.0	0.00115	14.3	-13.0	0.00035
bootstrap	56.0	22.4	0.00834	37.0	2.8	0.00194	25.9	-8.5	0.00071
smoothing	96.6	-1.7	0.01485	51.6	-1.0	0.00317	33.2	-8.8	0.00108

The upper part of Table 1 shows results obtained for one of the test databases (using the same experimental setup as above). It provides, for different sample sizes, threshold standard deviations ($\sigma_{a_{th}}$) and bias ($b(a_{th})$, the average difference with the asymptotic threshold determined by the classical method and using the whole database), and standard deviations of class probability estimates (average of the two successor subsets, denoted $Var(\hat{P})$). Note that the results for the other two databases described in the appendix are very similar to those shown in Table 1. They confirm the high variance of thresholds and probability estimates determined by the classical technique, independently of the considered database. On the other hand the "median" and to a lesser extent the "Kolmogorov-Smirnov measure" reduce variance very strongly, but lead to a significant bias with respect to the classical information theoretic measure. Note that median is not a very sensible choice for decision tree discretization, since it neglects the distribution of classes along the attribute values.

Improvements of the Classical Method. The very chaotic nature of the curves of Figure 1 obviously is responsible of the high threshold variance. We have thus investigated different techniques to "smoothen" these curves before determining the optimal threshold, of which we report the three following :
Smoothing : a moving-average filter of a fixed window size is applied to the score curve before selecting its maximum (window size was fixed to $ws = 21$).
Averaging : (i) the score curve and the optimal threshold are first computed, yielding test T^* as well as the score estimate $\hat{C}_C^{T^*}$ and its standard deviation

estimate $\hat{\sigma}_{C_C^{T^*}}$ (see [9]); (ii) a second pass through the score curve determines the smallest and largest threshold values $\underline{a}_{\text{th}}$ and $\overline{a}_{\text{th}}$ yielding a score larger than $\hat{C}_C^{T^*} - \lambda\hat{\sigma}_{C_C^{T^*}}$, where λ is a tunable parameter set to 2.5 in our experiments; (iii) finally the discretization threshold is computed as $\overline{a}^*_{\text{th}} = (\underline{a}_{\text{th}} + \overline{a}_{\text{th}})/2$.
Bootstrap : the procedure is as follows : (i) draw by bootstrap (i.e. with replacement) 10 learning sets from the original local learning subset; (ii) use the classical procedure on each subsample to determine 10 threshold values; (iii) determine discretization threshold as the average of these latter.

These variants of the classical method where evaluated using the same experimental setup as before. Results are shown in the lower part of Table 1; they show that "averaging" and "bootstrap" allow to reduce the threshold variance significantly, while only the former increases (slightly) bias. The same holds in terms of reductions of probability estimate variance. Hence averaging is the most interesting, since it does not increase significantly computing times.

3 Global Effect on Decision Trees

To evaluate the various discretization techniques in terms of global performance of decision trees, we carried out further experiments. The databases are first split into three disjoint parts : a set used to pick random samples for tree growing (LS), a set used for cross-validation during tree pruning (PS), a set used for testing the pruned trees (TS) (the divisions for each database are shown in Table 3, in the appendix). Then, for a given sample size N, 50 random subsets are drawn without replacement from the pool LS, yielding LS_1,LS_2,...,LS_{50}, and for each method the following procedure is carried out

- A tree is grown from each LS_i and for each discretization method.
- These trees are pruned (see [10] for a description of the method), yielding the trees $\mathcal{T}_i, (i = 1, \ldots, 50)$.
- Average test set error rate $\overline{P}_e$ and complexity $\overline{C}$ of the 50 trees are recorded.
- To evaluate variance, the quantity (1) is estimated using the test sample, providing $\hat{Var}(\hat{P}_{\mathcal{T}_i}(C|.))$

Table 2 shows results obtained on the three databases for a learning sample size of $N = 1000$; note that similar result were obtained for smaller and larger learning sets but are not reproduced here due to space limitations (for more details please refer to [11]). The last line of the table provides, as a ground for comparison, the results obtained by tree bagging, implemented using 10 bootstrap samples and aggregation of class-probability estimates of pruned trees, reporting the sum of the complexities of the 10 trees. One observes that all the methods succeed in decreasing the variance of the probability estimates on the three databases, the most effective being the median, followed by averaging and Kolmogorov-Smirnov. But, comparing the reduction in variance with the one obtained in the previous section, we note that the decrease is less impressive here. The main reason for this is that tree pruning, as it adapts the tree complexity to the method, has the side effect of increased complexity of the trees

Table 2. Results on three databases (global tree performances for $N = 1000$)

method	Gaussian ($P_e^B = 11.8\%$)			Omib ($P_e^B = 0\%$)			Waveform ($P_e^B = 14\%$)		
	$\overline{P_e}$	$\overline{C}$	$\hat{var}$	$\overline{P_e}$	$\overline{C}$	$\hat{var}$	$\overline{P_e}$	$\overline{C}$	$\hat{var}$
classic	12.56	10.32	0.0147	11.20	67.6	0.0572	27.30	45.96	0.0434
Kolmogorov	12.85	9.92	0.0109	10.41	73.6	0.0493	27.57	54.12	0.0432
median	12.17	14.28	0.0083	10.39	103.92	0.0383	27.30	66.04	0.0382
averaging	12.21	17.32	0.0105	10.69	98.68	0.0493	27.56	55.64	0.0386
bootstrap	12.49	12.28	0.0133	11.59	74.6	0.0500	27.39	49.48	0.0402
smoothing	12.56	9.88	0.0137	10.89	77.4	0.0532	27.23	47.68	0.0396
tree bagging	12.07	92.3	0.0047	8.29	468.6	0.0133	20.83	367.3	0.0100

obtained with the variance reduction techniques. This balances to some extent the local variance reduction effect. From the tables it is quite clear that median and averaging reduce variance locally most effectively, but also lead to the highest increase in tree complexity. The error rates are mostly unaffected by the procedure; they decrease slightly on the GAUSSIAN and OMIB databases while they remain unchanged on the WAVEFORM database.

Unsurprisingly, tree bagging gives very impressive results in terms of variance reduction and error rates improvement on all the databases, and especially on the WAVEFORM. Of course, we have to keep in mind that this improvement comes with a loss of interpretability and a much higher computational cost.

4 Discussion and Related Work

In this paper, we have investigated the reduction of variance of top down induction of decision trees due to the discretization of continuous attributes, considering its impact on both *local* and *global* tree characteristics (interpretability, complexity, variance, error rates). In this, our work is complementary to most existing work on discretization which has been devoted exclusively to the improvement of *global* characteristics of trees (complexity and predictive accuracy), neglecting the question of threshold variance and interpretability.

On the other hand, several authors have proposed tree averaging as a means to decrease the important variance of the decision tree induction methods, focusing again on global accuracy improvements. This has led to variations on the mechanism used to generate alternative trees and on the schemes used to aggregate their predictions. The first well known work in this context concerns the Bayesian option trees proposed by Buntine [12], where several trees are maintained in a compact data structure, and a Bayesian scheme is used to determine a posteriori probabilities in order to weight the predictions of these trees. More recently, so-called tree bagging and boosting methods were proposed respectively by Breiman [5] and Freund and Schapire [13]. In addition to the spectacular accuracy improvement provided by these latter techniques, they are attractive because of their generic and non-parametric nature. From our investigations it is clear that these approaches are much more effective in improving global accuracy than local variance reduction techniques such as those proposed in this

paper. However, the price to pay is a definite shift towards black-box models and a significant increase in computational costs. Our intuitive feeling (see also the discussion in Friedman [4]) is that tree averaging leads to local models, closer in behavior to nearest-neighbor techniques than classical trees. In terms of predictive accuracy, we may thus expect it to outperform classical trees in problems where the kNN method outperforms them (as a confirmation of this, we notice that kNN actually outperforms tree bagging significantly on the WAVEFORM dataset).

Another recent class of proposals more related to our local approach and similar in spirit to the early work of Carter and Catlett [14], consists in using continuous transition regions instead of crisp thresholds. This leads to overlapping subsets at the successor nodes and weighted propagation mechanisms. For example, in a fuzzy decision tree, fuzzy logic is used in order to build hierarchies of fuzzy subsets. Wehenkel ([9]) showed that in the context of numerical attributes this type of fuzzy partitioning allows indeed to reduce variance significantly. In [4], Friedman proposes a technique to split the learning subset into overlapping subsets and uses again voting schemes to aggregate competing predictions. Along the same ideas, we believe that a Bayesian approach to discretization ([9]) or probabilistic trees (such as those proposed in [15]) would allow to reduce variance. The main advantage of this type of approach with respect to global model averaging is to preserve (possibly to improve) the interpretability of the resulting models. The main disadvantage is a possibly significant increase in computational complexity at the tree growing stage.

With respect to all the intensive research, we believe that the contribution of this paper is to propose low computational cost techniques which improve interpretability by stabilizing the discretization thresholds and by reducing the variance of the resulting predictions. In the problems where decision trees are competitive, these techniques also improve predictive accuracy. We also believe that our study sheds some light on features of decision tree induction and may serve as a starting point to improve our understanding of its weaknesses and strengths and eventually yield further improvements.

Although we have focused here on local (node by node) discretization philosophies, it is clear from our results that global discretization must show similar variance problems and that some of the ideas and methodology discussed in this paper could be successfully applied to global discretization as well. More broadly, all machine learning methods which need to discretize continuous attributes in some way, could take advantage of our improvements.

In spite of the positive conclusions, our results show also the limitations of what can be done by further improving decision tree induction without relaxing its intrinsic representation bias. A further significant step would need a relaxation of this representation bias. However, if we want to continue to use the resulting techniques for data exploration and data mining of large datasets, this must be achieved in a cautious way without jeopardizing interpretability and scalability. We believe that fuzzy decision trees and Bayesian discretization techniques are promising directions for future work in this respect.

References

1. L. Breiman, J.H. Friedman, R.A. Olsen, and C.J. Stone. *Classification and Regression Trees.* Wadsworth International (California), 1984.
2. J.R. Quinlan. *C4.5: Programs for machine learning.* Morgan Kaufmann (San Mateo), 1986.
3. R. Kohavi and D. H. Wolpert. Bias plus variance decomposition for zero-one loss functions. In *Proc. of the Thirteenth International Conference on Machine Learning*, 1996.
4. J. H. Friedman. Local learning based on recursive covering. Technical report, Department of Statistics, Standford University, August 1996.
5. L. Breiman. Bagging predictors. Technical report, University of California, Department of Statistics, September 1994.
6. R.L. De Mantaras. A distance-based attribute selection measure for decision tree induction. *Machine Learning*, 6:81–92, 1991.
7. L. Wehenkel. On uncertainty measures used for decision tree induction. In *Proc. of Info. Proc. and Manag. Of Uncertainty*, pages 413–418, 1996.
8. J. H. Friedman. A recursive partitioning decision rule for nonparametric classifier. *IEEE Transactions on Computers*, C-26:404–408, 1977.
9. L. Wehenkel. Discretization of continuous attributes for supervised learning : Variance evaluation and variance reduction. In *Proc. of The Int. Fuzzy Systems Assoc. World Congress (IFSA'97)*, pages 381–388, 1997.
10. L. Wehenkel. *Automatic learning techniques in power systems.* Kluwer Academic, Boston, 1998.
11. P. Geurts. Discretization variance in decision tree induction. Technical report, University of Liège, Dept. of Electrical and Computer Engineering, Jan. 2000. (`http://www.montefiore.ulg.ac.be/~geurts/`)
12. W. Buntine. Learning classification trees. *Statistics and Computing*, 2:63–73, 1992.
13. Y. Freund and R.E. Schapire. A decision theoretic generalization of on-line learning and an application to boosting. In *Proc. of the 2nd European Conference on Computational Learning Theory*, pages 23–27. Springer Verlag, 1995.
14. C. Carter and J. Catlett. Assessing credit card applications using machine learning. *IEEE Expert*, Fall:71–79, 1987.
15. M. I. Jordan. A statistical approach to decision tree modeling. In *Proc. of the 7th Annual ACM Conference on Computational Learning Theory.* ACM Press, 1994.

A Databases

Table 3 describes the datasets (last column is the Bayes error rate) used in the empirical studies. They provide large enough samples and present different features : GAUSSIAN corresponds to two bidimensional Gaussian distributions; OMIB is related to electric power system stability assessment [10]; WAVEFORM denotes Breiman's database [1].

Table 3. Datasets (request from `geurts@montefiore.ulg.ac.be`)

Dataset	#Variables	#Classes	#Samples	#LS	#PS	#TS	P_e^{Bayes}
GAUSSIAN	2	2	20000	16000	2000	2000	11.8
OMIB	6	2	20000	16000	2000	2000	0.0
WAVEFORM	21	3	5000	3000	1000	1000	14.0