A Comparative Study of Reliable Error Estimators for Pruning Regression Trees

Luís Torgo

LIACC/FEP – University of Porto
R. Campo Alegre, 823, 2º - 4150 PORTO - PORTUGAL
Phone : (+351) 2 607 8830 Fax : (+351) 2 600 3654
email : ltorgo@ncc.up.pt WWW : http://www.ncc.up.pt/~ltorgo

Abstract. This paper presents a comparative study of several methods for estimating the true error of tree-structured regression models. We evaluate these methods in the context of regression tree pruning. Pruning is considered a key issue for obtaining reliable tree-structured models in a real world scenario. The major step of a pruning process consists of obtaining accurate estimates of the error of alternative tree models. We evaluate experimentally four methods for obtaining these estimates in twelve domains. The goal of this evaluation was to characterise the performance of the methods in the task of selecting the best possible tree among the set of trees considered during pruning. The results of the comparison show that certain estimators lead to poor decisions in some domains. The Cross Validation variant that we have proposed achieved the best results on the set-ups we have considered.

Keywords : Machine Learning, Regression Trees, Pruning methods.

1 Introduction

This paper describes an experimental comparison of several alternative methods for obtaining *reliable* error estimates of tree-based regression models. These methods are evaluated in the context of pruning regression trees which is considered a key factor for obtaining these models (Breiman *et al.*,1984).

Tree-based models are obtained using a recursive partitioning algorithm that rapidly ends up with very small samples lacking statistical support. Moreover, real world domains are noisy, which leads to over-specialized trees. These facts originate unreliable decisions in lower branches of tree-structured models. The standard approach followed to overcome this difficulty consists of growing a very large tree and then prune it back to the “right size”. This pruning step is guided by better estimates of the “true” error of the pruned trees. Several methodologies exist to obtain unbiased estimates of an unknown population parameter based on samples of this population. Resampling techniques use separate samples to obtain estimates that are "independent" of the sample used to grow the models. Examples of this technique are Cross Validation or the Houldout method used in CART (Breiman et al., 1984). Other approaches use the sampling properties of the distribution of the parameter being estimated to make "corrections" to the estimates obtained with the training sample. C4.5 (Quinlan, 1993) for instance, uses a binomial correction to the distribution of the error rate. Bayesian methods combine prior knowledge of the parameter with the observed value to obtain a posterior estimate of the target parameter. *M*-estimates (Cestnik, 1990) are an example of such techniques and have been used in the context of pruning regression trees (Karalic and Cestnik, 1991).

In this paper we empirically compare several alternative methods for error estimation in the context of pruning regression trees. We describe three new variants of existing methods. Previous comparative studies on tree pruning (Mingers, 1989; Esposito *et al.*, 1993, 1995) have concentrated on classification trees. Moreover, they have compared full pruning algorithms instead of error estimators as we do here.

2 Inducing Regression Trees

In this section we present a brief overview of the methods used for growing a regression tree. This recursive process involves three main decisions :

- Deciding which split test to include on each inner node of the tree.
- When to stop the growth of the tree.
- Which model to use in the leaves of the tree.

The usual method followed consists of using a partitioning algorithm that keeps splitting the given sample into smaller and smaller sub-sets until the stopping criteria are fulfilled. A classical example of such procedure is used in the CART system (Breiman et. al., 1984).

This recursive partitioning algorithm very quickly ends-up with a small number of cases. The splits selected on the basis of such small samples are extremely unreliable, hardly generalizing over unseen cases. This may lead to poor predictive performance of the obtained regression model. The usual strategy of overcoming this problem was proposed by Breiman et al. (1984) and consists of post-pruning the overly large regression tree obtained using the methods outlined above. Breiman and his colleagues described the pruning task as a three step process :

- Generate a set of "interesting" pruned trees.

- Obtain reliable estimates of the error of these trees.
- Choose one of these trees according to the estimates.

To solve the first issue of this list two types of methods exist. Nested sequences of trees are obtained by iteratively choosing a node to prune from the previous tree in the sequence. We start with the unpruned tree until a tree with a unique leaf is reached. Several methods exist to choose the node to prune at each step. An alternative to nested sequences is to try to find a sequence of trees with size decreasing in one, such that for each size *i* we obtain the tree with lowest error among all possible sub-trees with that size. These methods are computationally more complex than the former although efficient backward dynamic algorithms exist (see for instance Bohanec & Bratko, 1994 or Almuallim, 1996).

The key issue of the pruning process is how to obtain reliable estimates of the error of the pruned trees. We require that the estimates perform a correct ranking of the candidate trees. This ensures the selection of the best possible tree from the set of candidate pruned trees. As mentioned by Weiss & Indurkhya (1994) this is basically an estimation problem. In the context of regression tree pruning, more important than the precision (bias) of these estimates, is the correct ranking of the trees in the sequence.

3 The Estimation Methods

3.1 Resampling Methods

In our study we have used two variations of existing resampling estimators. The first variant is based on the *Holdout* method. The use of this method in the context of regression trees can be described as follows. Given a learning sample we randomly divide it into a training and a pruning set (the holdout). A large tree is grown without seeing the holdout. A sequence of pruned trees is obtained and the pruning set is used to obtain reliable estimates of the error of these trees. The key question of this method is which proportion of cases should we leave for the holdout. Ideally one wants to have a pruning set as large as possible to ensure good estimates. However, this may lead to a shortage of cases for growing the tree, which will damage the overall accuracy of the final tree. We propose an heuristic variant based on extensive experimentation, which consists of using 30% of the data for pruning set, limited to a maximum of 1000 cases, i.e.

$$\#\{\text{PruningSet}\} = \min\left(0.3 \times \#\{LearningSample\}, 1000\right) \quad (1)$$

The reasoning behind this limit is that we have observed that it is a sufficient amount to ensure reliable estimates (similar observation was made by Weiss & Kulikowski, 1991). Exceeding this size will bring little advantage in terms of estimates accuracy, whilst decreasing the size of the training set.

N-fold Cross Validation (Stone, 1974) can also be used to obtain reliable estimates for selecting a pruned tree (Breiman *et al.*,1984). These authors divided the learning sample in N folds. For each fold a tree is grown using the remaining N-1 folds. A sequence of trees is generated and the fold not used for learning the tree is used to obtain reliable estimates for the trees in the sequence. Their goal is to estimate the *optimal* value of a complexity parameter α. This parameter is obtained as a weighted average of tree error and complexity (size). After this estimation phase a tree is grown on all learning sample and a sequence of pruned trees is generated. Based on the *optimal* α value obtained by Cross Validation a tree is selected from the sequence. This selection is based on a heuristic assumption about the equivalence of trees with similar α values. There is also a potential source of bias on the fact that the estimated α value is obtained on training sets with smaller size. Moreover, this method is strongly linked to the method of generating the sequence of trees. In effect, the trees are generated by pruning at each step the node that is the *weakest link* in terms of α.

We propose a Cross Validation (CV) method that can be applied whatever the algorithm used to generate the pruned trees. As mentioned above, the main problem to be solved for using CV estimates is the problem of tree matching. In effect, we have several sequences of pruned trees (one for each fold), plus the final sequence obtained using all learning sample. Our goal is to know which is the best tree in this final sequence. We estimate the error of these trees based on the estimates obtained in the "folded" sequences. These later estimates are reliable because they are obtained using a separate set of data. To obtain the estimate for a tree in the final sequence we should use the reliable estimates of the *most similar* trees in the folded sequences. This tree-matching problem is "solved" in CART using the α values. Our alternative proposal is the following. Given a sequence of pruned trees T_0, T_1, …, $T_{\max}$, where T_0 is the unpruned tree, we know that the error in the training data of these trees decreases with the increasing size of the trees, i.e. $Err_{tr}(T_{\max}) \geq Err_{tr}(T_{\max\text{-}1}) \geq \ldots \geq Err_{tr}(T_0)$. We calculate a score for each tree in the sequence as its decrease in error over the maximal decrease in the sequence,

$$Score(T_i) = \frac{Err_{tr}(T_{\max}) - Err_{tr}(T_i)}{Err_{tr}(T_{\max}) - Err_{tr}(T_0)} \qquad (2)$$

The values of this function range from 0 ($Score(T_{\max})$) to 1 ($Score(T_0)$). We obtain these scores for all trees in all sequences. We estimate the error of a tree in the final sequence by averaging over the reliable errors of the trees in each folded sequence, which have most similar score. For instance, the error of the unpruned tree in the final sequence is estimated by averaging over the reliable estimates of all unpruned trees of each folded sequence. Compared to the α-based method used in CART, our method is independent of the algorithm used to generate sequences of pruned trees.

3.2 Bayesian Methods

Bayesian methods estimate a population parameter by a combination between prior and observed knowledge. Cestnik (1990) introduced *m*-estimates in the context of machine learning. Later Karalic and Cestnik (1991) used them within regression trees. Due to the difficulty of obtaining priors for the variance of the target variable of the domain under consideration, the usual approach followed within tree-based models is to take the estimate on the entire given sample as our prior estimate. The *m*-estimate of the variance based on a sample of size *n* (for instance in a leaf of the tree), given that the size of all available data is *N*, uses the *m*-estimate of the mean and is given by

$$m - \mathrm{Est}(\mu_Y) = \frac{1}{n+m}\sum_{i=1}^{n} y_i + \frac{m}{N(n+m)}\sum_{i=1}^{N} y_i$$

$$m - \mathrm{Est}(\sigma_Y^2) = \frac{1}{n+m}\sum_{i=1}^{n} y_i^2 \quad + \quad \frac{m}{N(n+m)}\sum_{i=1}^{N} y_i^2 \quad - \quad (m - \mathrm{Est}(\mu))^2 \qquad (3)$$

Several values for the *m* parameter were tried in the context of our experimental comparisons. The best results were obtained with the value 2.

3.3 Methods based on Sampling Distribution Properties

Least squares regression trees use an error criterion that relies on the estimates of variance in the leaves of the trees. Estimation theory tells us that the sampling distribution of the variance follows a χ^2 distribution (Bhattacharyya and Johnson, 1977). A 100×(1-α)% confidence interval for the population variance based on a sample of size *n*, is given by

$$\left(s_Y^2 \frac{n-1}{\chi^2_{\alpha/2}},\ s_Y^2 \frac{n-1}{\chi^2_{(1-\alpha/2)}} \right) \qquad (4)$$

where, s_Y is the sample variance (in our case obtained on each tree leaf).

This formulation relies on a strong assumption regarding the normality of the distribution of the variable *Y*. In most real-world domains we can not guarantee *a priori* that this assumption holds. Failure on this assumption may lead to unreliably narrow intervals for the location of the true population variance. However, in the context of our work we are not particularly interested in the precision of the estimates, but in guaranteeing that the estimate obtains a correct ranking of the pruned trees. Being so, we have decided to use this method adopting a kind of heuristic (and pessimistic) estimate of the variance by choosing as our estimate the highest value of the interval given by Equation 4.

4 The Experiments

In our experiments we have used 12 data sets whose main characteristics are describe in Table 1. The goal of our experiments is two-folded. First, we want to assert the selection performance of each method when given a set of candidate pruned trees. Secondly, we want to compare the trees selected by each method in terms of size and accuracy in an independent test set.

Data Set	*Training Pool;Test Set*	*Data Set*	*Training Pool;Test Set*
Abalone	3133;1044	Kinematics	4500;3692
Pole	5000;4065	Fried1	30000;5000
Elevators	8752;7847	Census16H	17000;5784
Ailerons	7154;6596	Census8L	17000;5784
CompActiv.	4500;3692	2Dplanes	20000;5000
CompActiv(s)	4500;3692	Mv1	20000;5000

Table 1. The used Data Sets showing the available number of cases.

We have randomly divided the original data set in a large independent test set and a training pool. Using this training pool we have randomly obtained samples of different sizes. For each size we have grown a regression tree and obtained a sequence of pruned trees. Each of the estimation methods was used to select one of these trees and the accuracy of these choices was tested on the independent test set. Using this test set we have also observed what would be the best possible selection from the available trees. The results we present are averages of 20 repetitions for each of the tried sample sizes (300, 600, 1000 and 2000 cases).

The first experiments address the question whether the estimation methods are able to select the tree that would perform better in the test set. We first compare the size of this best tree to the size of the selected tree. Figure 1 shows the average percentage difference over the 20 repetitions for each of the combinations of methods and data sets. Due to lack of space we only show the graphs for training samples of 300 and 2000 cases. However, the general pattern of results is similar for the other sizes. The results in the figure were truncated to a maximum 150% increase over the best tree.

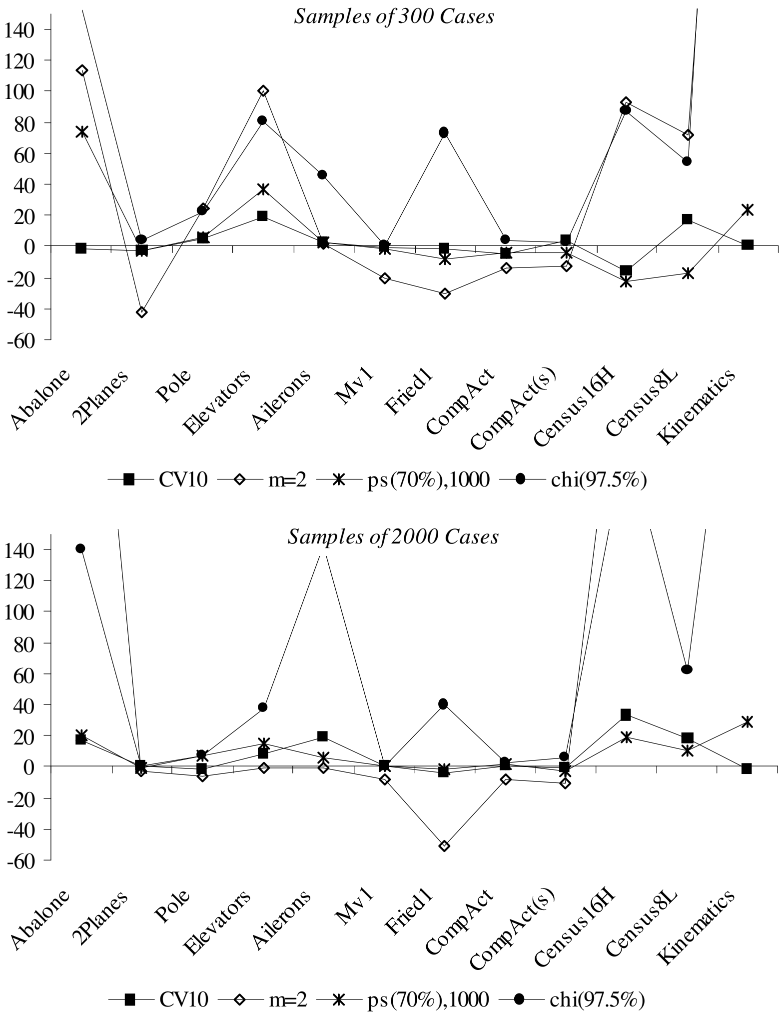

Figure 1. Average percentage size difference between selected and best trees.

From the results presented in Figure 1 we can conclude that χ^2 estimators have a general tendency to select trees much larger than the best tree available in the sequence. *M*-estimators vary quite wildly from data set to data set. They make very poor selections in some domains for all tried sizes. However, with larger samples they are able to make quite good selections in some domains. Both CV and Holdout estimators exhibit a quite stable behavior over all domains and sizes. They seldom make poor selections and they frequently choose exactly the best tree in the sequence.

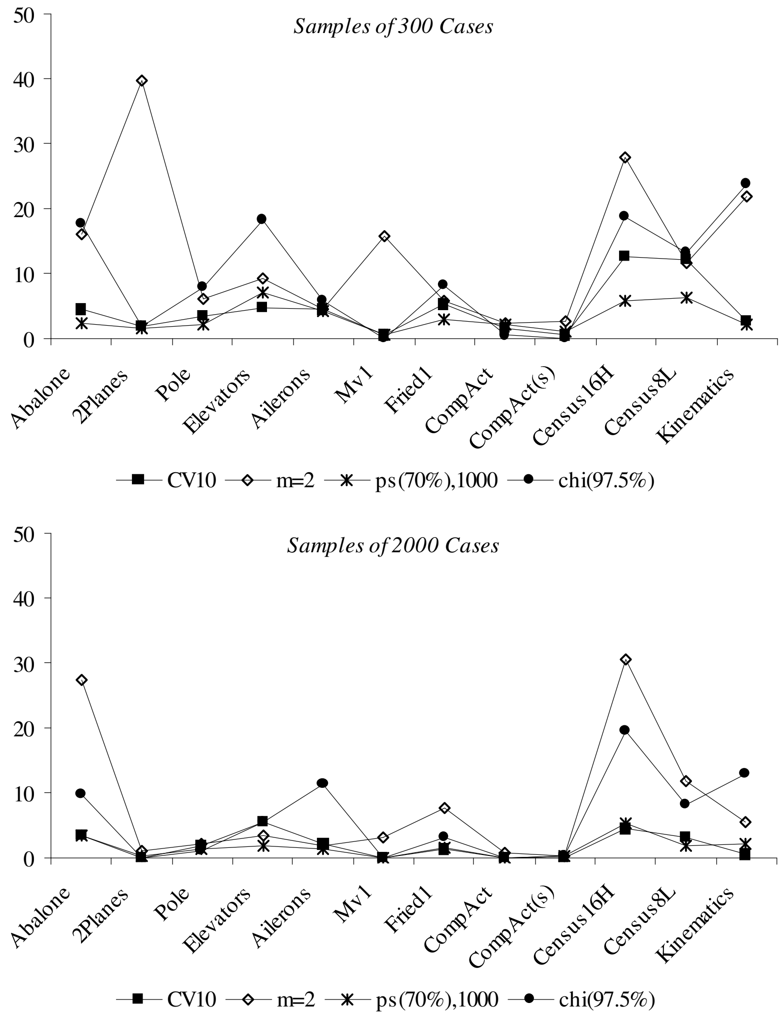

Figure 2. Percentage error increase with respect to the best tree on the test set.

Selecting larger trees has an undesirable effect on interpretability, which is one of the advantages of these models. However, as Breiman *et al.*(1984) mentioned, the accuracy of the trees in the sequence is quite similar for a large variety of sizes around the best tree. This means that although a method may consistently choose less interpretable trees, this does not necessarily entails a large loss in accuracy. We try to understand this relation in Figure 2. This figure shows the percentage error increase with respect to the best tree corresponding to the choices shown in Figure 1.

The results of Figure 2 confirm that although there are huge differences in tree sizes the corresponding loss in accuracy is not so high. Still, there are relevant accuracy losses entailed by the selections of some methods, particularly in small samples. Once again, both CV and the Holdout are clearly the best tree selectors.

The second goal of our experimental comparisons was to find out if there is a clearly best estimation method. For this purpose we compared the trees selected by each method in terms of size and accuracy on an independent test set. This is a different comparison from the previous one where we were comparing the selected trees to the best possible selection. The results we will now describe assume particular relevance for the holdout method. In effect, this method is selecting trees from a sequence that is based on a tree learned with less data.

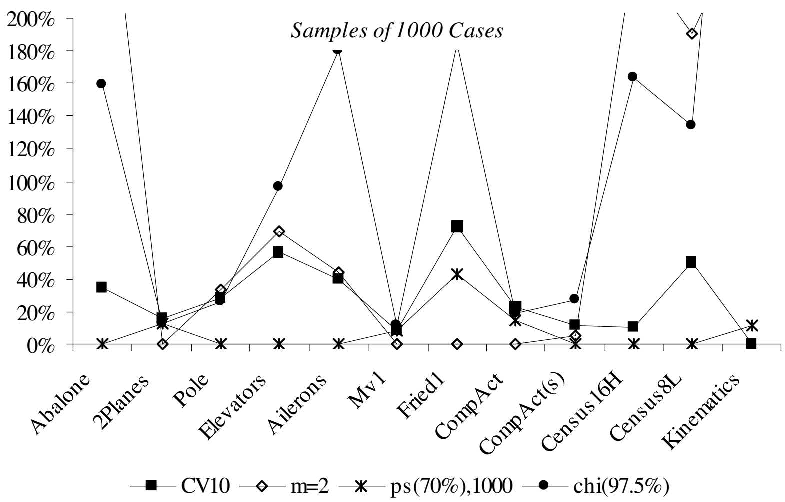

Figure 3. The comparison of the sizes of the trees selected by each method.

Figure 3 shows the results of the size comparisons for samples of 1000 cases. We omit the graphs for other sizes because the overall pattern is similar. The figure shows a percentage loss in size of the tree selected by each method when compared to the best score (i.e. the best method in each data set as a value 0). We can see that the holdout method usually selects smaller trees than the others. *M*-estimators also score particularly well in some domains, but again high instability is observed. This may indicate that this method needs tuning of the *m* parameter for each domain. CV estimates also achieve reasonable results over all domains, while χ^2 are quite bad in terms of the interpretability of the selected regression models.

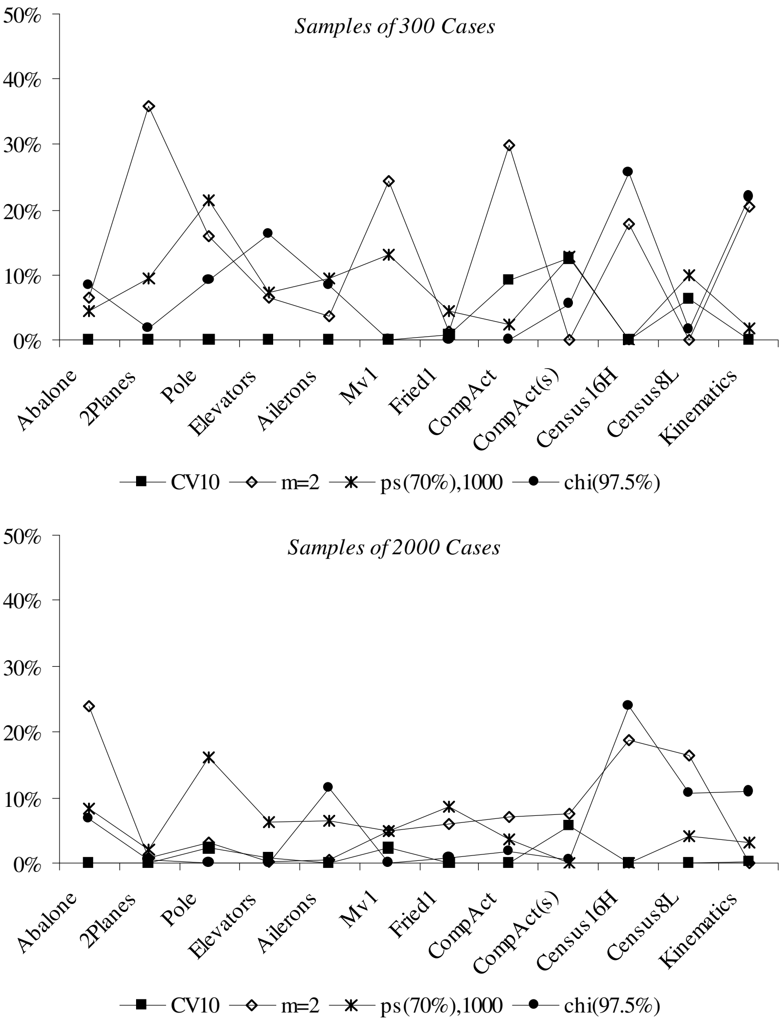

Figure 4. The accuracy comparisons between the trees slected by each method.

We know present the results concerning the accuracy comparison measured on an independent test set. Figure 4 presents the percentage accuracy loss over the score of the best method.

The first conclusion we can draw from these graphs is that there is a penalty to pay for using a separate holdout. This is more evident for smaller samples as expected. However, even with samples of 2000 cases, we observed a consistent loss over methods like CV estimators. Still, there is a tendency of decrease in this loss as the size of the sample increase. This may be a good indication of the applicability of the holdout with larger samples. This assumes particular relevance because the results of Figure 3 show that this method usually leads to more interpretable trees. In extremely large domains like the ones faced in data mining, this may be a strong advantage. Moreover, we have to recall that with the holdout we learn one tree, while with CV we need to induce N+1 trees. In effect, this is the main drawback of CV estimators. On the other hand they select trees with excellent accuracy and with reasonable size. Both *m*-estimators as χ^2 have the advantage of growing only one tree and not "wasting" data in a separate holdout. However, in our experiments these methods were not able to capitalize on these advantages. Their results are quite unstable over the different domains. This may be an indication that the parameters of these methods (*m* and confidence level) need specific tuning for each domain. However, this can only be achieved with resampling making them loose the mentioned efficiency advantages.

5 Conclusions

Tree-based regression is based on an efficient recursive partitioning algorithm. However, this same algorithm causes one of its well-known problems, namely the unreliability of lower levels of the trees. Post-pruning of these trees is considered an essential step to overcome this drawback. Reliable estimates of the true error of the trees are the key issue for successful pruning. In this paper we have presented a comparative study of four alternative methods of estimating the error of trees. This comparison was carried out in twelve domains for different sample sizes. Our comparisons confirmed the importance of this pruning stage. In effect, significant differences in terms of accuracy and tree size were observed by using different error estimation methods.

We have presented a new estimation method based on the sampling distribution properties of the variance, and two new variants of existing resampling methods. The main conclusions of our comparative study can be summarised as follows. Concerning the problem of selecting the best possible tree from a sequence of pruned trees, both CV and Holdout estimates achieve the best results. The results of the χ^2 and *m*-estimates vary a lot from domain to domain. When comparing the trees selected by each method we have observed that the Holdout chooses more interpretable models. However, this method has lower accuracy because of the use of less data for inducing the trees. This negative effect has a tendency to disappear with larger samples. Still, for the set-ups that we have explored our proposed CV estimator is clearly the overall winner. The overhead in computation of this method can be considered irrelevant for these sample sizes.

Summarising, for these set-ups our recommendation is clearly CV estimates. For larger samples one may consider the use of the Holdout method due to its lower computational complexity and smaller selected trees.

Acknowledgements : I would like to thank PRAXIS XXI and FEDER for their financial support. Thanks also to my supervisor Pavel Brazdil and my colleagues.

References

Almuallim,H. (1996) : An efficient algorithm for optimal pruning of decision trees, Artificial Intelligence, 82 (2), Elsevier.

Bohanec,M., Bratko,I. (1994) : Trading Accuracy for Simplicity in Decision Trees, *Machine Learning*, **15** (3). Kluwer Academic Publishers.

Breiman, L. (1996) : Bagging predictors. In *Machine Learning*, **24** (2). Kluwer Academic Publishers, 1996.

Breiman,L. , Friedman,J.H., Olshen,R.A. and Stone,C.J. (1984) : *Classification and Regression Trees*, Wadsworth Int. Group, Belmont, California, USA, 1984.

Cestnik,B. (1990) : Estimating probabilities : A crucial task in Machine Learning. In proceeding of the 9th European Conference on Artificial Intelligence (ECAI-90), Pitman Publishers.

Esposito,F., Malerba,D., Semerano,G. (1993) : Decision Tree Pruning as a Search in the State Space. In Proceedings of the European Conference on Machine Learning (ECML-93), Brazdil,P. (ed.). LNAI-667, Springer Verlag

Esposito,F., Malerba,D., Semerano,G. (1995) : Simplifying Decision Trees by Pruning and Grafting: New Results. In Proceedings of the European Conference on Machine Learning (ECML-95), Lavrac,N. and Wrobel,S. (eds.). LNAI-912, Springer Verlag

Karalic,A. (1992) : Employing Linear Regression in Regression Tree Leaves. In Proceedings of the European Conference on Artificial Intelligence (ECAI-92). Wiley & Sons.

Karalic,A., Cestnik,B. (1991) : The bayesian approach to tree-structured regression. In proceedings of the ITI-91.

Mingers,J. (1989) : An Empirical Comparison of Pruning Methods for Decision Tree Induction, in *Machine Learning*, **4** (2), p.227-243. Kluwer Academic Publishers.

Quinlan,J.R. (1992) : Learning with continuous classes. In Proceedings of AI'92, Adams & Sterling (eds.). World Scientific. 1992

Quinlan,J.R. (1993) : *C4.5: programs for machine learning*. Morgan Kaufmann Publishers,1993.

Stone, M. (1974) : Cross-validatory choice and assessment of statistical predictions, *Journal of the Royal Statistical Society*. B **36**, 111-147, 1974.

Torgo,L. (1997) : Functional Models for Regression Tree Leaves. In Proceedings of the International Conference on Machine Learning (ICML-97), Fisher,D.(ed.). Morgan Kaufmann Publishers, 1997.

Weiss,S., Indurkhya,N. (1994) : Decision Tree Pruning : Biased or Optimal ?. In Proceedings of the AAAI-94.

Weiss,S., Kulikowski,C. (1991) : *Computer Systems that Learn*. Morgan Kaufmann Publishers, 1991.