Technical Updates: Decision Tree Pruning: Biased or Optimal?

Sholom M. Weiss
Department of Computer Science,
Rutgers University
New Jersey 08903, USA

Nitin Indurkhya
Department of Computer Science,
University of Sydney,
NSW 2006, AUSTRALIA

1 Technical Updates

In this section we will see how we can add some technical enhancements and how we can update the existing methods in the paper in its Machine Learning Related components. This will help us to improve and can be a possible upgrade for future version of this paper. The main aim of the paper was tree pruning and comparing tree pruning performance for 3 different error estimators on various datasets. Some technical updates which we can implement are:

- In the paper in CV we used 10 fold cross validation. When estimating error rates, cross-validation will train on less than the full sample .Paper used 90% of samples for training, so the error rate estimates are for 90% not for 100% which is a small weakness when the true answer is near the unpruned tree. So what we can do is for large datasets we can have 95-5 split instead of 90-10, so that training is improved and error estimates are also closer to optimal since we will be using 95% for training.When we increase the training data, model learns more generalized results and performs better on unseen or test data.

- In order to improve the decision tree accuracy we can also use Principal Component Analysis(PCA) for dimensionality reduction. In PCA we reduce the original feature space dimension to relevant no. of dimensions which give us a great correlation among the data. We calculate the eigenvalue and higher values correspond to the larger spread along that respective eigenvector. So, an eigenvector with high eigenvalues becomes our principal component and helps to improve the accuracy by removing features which have very less contribution in classification. Hence we keep top ‘n’ dimensions that contain the most variance and it also helps to contain most of the information. By using PCA one can also get rid of the problem caused by outliers or noise. It sort of helps in removing noise. And as we saw in the paper tha NP estimator performs badly for noisy data and after doing PCA and using reduced and suitable features we can get the true measure of NP estimator performance.

- The slight modification in the induction program that ties were broken in the favour of larger trees can be avoided and instead we could have broken ties in favour of smaller trees so that we can get accurate results and also a simpler tree. Larger trees with more layers of decision nodes may lead to overfitting of the data and due to overfitting, the results and accuracy on test data or unseen data will be pretty bad. So what we can do is, we can also use along with test data , a validation set to keep an eye on overfitting. As soon as we reach a point where test error decreases and validation increases we can stop at that point and we can guess that overfitting has started.

- Unlike in the paper where we are only searching for minimum error pruning, we should include a criteria for the simplicity of the tree.We can add a parameter along with error value, which adds a penalty according to size of tree. So when we have we have great accuracy tree size will be large and penalty will also be large and if penalty is less that is simple tree then the error will be more.This will help us to have a good trade off. And then this will help us to select a tree with minimum error and small size, so that our tree can be less error-pruned but at the same time simple also, because in the real world simpler trees are preferred more and a little inaccuracy can be tolerated for that. Larger trees means more no.of splits and more layers which may be a kind of overfitting. Large trees may provide better accuracy but at the same time make it a bit computationally extensive and may lead to overfitting. Whereas when choosing a small tree, we achieve simplicity but we have to compromise a bit on accuracy.

- We know that NP is a highly optimistic strategy and performs well only when bias is close to the true answer. NP leads to drastically degraded performance in case of noisy data. Although real world data is noisy, in order to compare its efficiency with others before using the data we can do data cleaning so that any outlier and noise from data can be reduced and then we can compare true NP's performance to other estimators.

- As an upgrade/update, we can, instead of taking one decision tree finally what we can take the best 3 trees one simpler tree, one minimum error tree and last one can be any basic decision tree. So, we can use random forest to get multiple trees and then combine their results in order to get more accurate predictions. Random forest as the name suggests is an ensemble of many decision trees. Random forest helps the model to learn in a better manner due to the wide diversity of trees being used by Random forest.Having many trees i.e a Random Forest helps us to get a better and well suited result.

Although the paper was well written and presented, we can still use the above listed updates and can expect to have a better performance than presented in the paper.