Decision Trees

Tommy Baird tbaird7@jhu.edu
Johns Hopkins University
Baltimore, MD 21205, USA

Editor: Tommy Baird

Abstract

In this paper, we will discuss the application of a decision tree machine learning algorithm to predict the outcomes of both classification and continuous targets. This application will be tested on 6 different data sets of varying length and complexity. Each tree will be pruned to improve generalization as we apply it to new data points outside of the training data. We will then compare the performance of the full tree and pruned tree to test the general effectiveness of the algorithm.

Keywords: Machine Leaning, Decision Trees

1 Introduction

In this experiment, we will be creating decision trees for 6 different data sets to test the algorithm's effectiveness to predict target values. In general, we believe that data sets will perform better once pruned than they will as fully grown trees. Furthermore, we believe that the continuous targets see more improvement from pruning than the categorical targets. Within those subsets, we believe that data sets that rely on categorical features will perform better than those who rely on continuous features. As we learned through our lectures, the largest danger of a decision tree is over-fitting with an overly specific full tree. With a finite number of outcomes in categorical targets and features, we believe that we will hit earlier stopping points as we have a higher likelihood of hitting a tree leaf node where there is only one possible outcome left. In terms of continuous values, there are theoretically infinite target and feature values, so we will be at risk of over-splitting as we continue to search for a stopping point. Thus, pruning our continuous values will see more overall benefit to pruning that, frankly, should have more pruning to do to reach a more generalized model state.

1.1 Categorical Targets Hypotheses

Turning our attention to the specific categorical target data sets, we believe that our car evaluation and congressional voting will perform best in that they have a categorical features and targets. congressional voting should be especially successful given that there are only two target possibilities, there are at most three but usually two feature value possibilities (voting for or against with small population of abstain), and, thankfully for our experiment, politicians generally vote along their party lines. This should provide us clean bucketing as we try and reach our intended classifications. Car evaluation has some of the same benefits,

but, as we will show, the distribution is not very balanced to provide proper representation of each class. The Breast Cancer data set should also perform well given its binary target values of 2 (benign) and 4 (malignant), but its continuous feature values present concerns of an overgrown tree that over-fits the training data. Our hope is that pruning will clean up lack of generality.

1.2 Continuous Targets Hypotheses

As we flip over to the continuous targets, our concerns for over-fitting are at their highest with the abalone and forest fire data sets as they have not only their continuous targets but also at least a few continuous features. Pruning should show significant performance improvement over the fully grown trees. Computer Hardware is a tough data set to get a read on because the integer feature values may provide some reprieve with a higher chance of value overlap than we see with continuous values. However, it is a smaller data set, so we do wonder how much room there is for generalization with such a wide range of targets.

2 Experimental Approach

We created our decision tree creation, pruning, and testing with the help of the pandas and numpy libraries within python. Our trees were stored as data frames with pointers from parent to children and children back to parent as needed. When traversing our tree during creation and testing, decisions were made based on previous filters provided by ancestor nodes. Our initial tree grew out to the leaves until we had no further decisions to make either due to a single possible target remaining or to no further splits possible. These splits were done either by categorical values or by a binary split for continuous values based on the mean of the test data subset at that specific node. We picked each subsequent node during the growth phase either by the max of the gain ratio for classification trees or the min of mean squared error for regression trees. When pruning, we looked at the performance of all the parent nodes of the existing leaves and pruned where performance was worse.

2.1 Data Assumptions Made

As one would expect, there does not exist a one size fits all data assumption that fit all of our features within each data set. In most cases, we could organize and split features as defined in the problem experimental approach: split and match categorical features while continuous features were split based on the mean of the test data subset. When a categorical feature did not have an exact match, we had two approaches to picking the proper path of traversal. If we had ordinal or relative features such as the car evaluation descriptions from low to very high or the forest fires day of the week, we performed a distance calculation and picked our closest available option. If there was no logical order of the feature possibilities as we saw with the sex of an abalone or an abstaining vote within our congressional sample, we deferred to a random choice as we did not want to introduce any unnecessary bias

2.2 Pruning

When pruning, we looked to clip at the parent nodes that led to the worst child node performance. This will allow us to focus on the over-fitted portions of the fully grown tree and generalize where performance has the most room for improvement.

Before we delve into the results of our testing, we would like to highlight the difference between 'post-pruned' and 'extra-pruned' test results. As we constructed our tests, we saw that not much pruning was taking place. To test for degradation of performance from pruning, we did a simple comparison of if the performance from further pruning decreased relative to the tree before. To encourage more pruning, we provided an extra set of results labeled 'Extra-Pruned' that forced pruning by not allowing the pruning to end until 20% of the prune-able nodes (non-leaf nodes) had been cut. This seemed to improve performance significantly as we will show below. As such, we will use the extra-pruned figure when referencing prune performance improvement percentage.

As an extra note, we also have early stop if the parent chosen to prune is the root node. We admit that this is not the most ideal of stopping case assumptions as we could potentially guide the pruning to other portions of the tree, but the point of this 'extra' prune was to ensure that we could show substantial pruning as a comparison point to the fully grown trees rather than needing to get to this exact arbitrary number of 20%.

If given more time, we would have liked to delve into a more intelligent degradation measure where we could look at a reversal of a pre-defined moving average range rather than just simple comparison to most recent example. This would hopefully would encourage more pruning without having to hard code a pre-specified amount.

3 Experiment Performance

Our performance review will be broken into categorical and regression trees. This will hopefully help the reader digest our different metrics of basic success percentage and a mean squared error separately.

3.1 Categorical Tree Performance

Our categorical performance was strong depending on which test best fit each data set, but interesting to see each have varying reaction to pruning. Referring to table 1, we can see that breast cancer benefited most from pruning as it steadily improved with more pruning. Car evaluation on the other hand showed deterioration on performance from initial and extra pruning. Finally, congress voting showed no real difference in its ability to predict target values between a fully grown tree and a pruned tree. We will delve deeper into each to see the same performance breakdown but one level deeper to see class by class improvement or degradation.

Table 1: Categorical Target Success Rate

Data Set	Pre-Pruned	Post-Pruned	Extra-Pruned	Prune Improve %
Breast Cancer	0.813	0.871	0.934	14.917
Car Eval	0.899	0.894	0.834	-7.210
Congress Voting	0.931	0.944	0.947	1.667

3.1.1 Breast Cancer

Within the breast cancer set, we can see in table 2 that class 2 benefited most from pruning which we believe is due to the fact that it is the larger class by proportion. As we generalize the tree, it improves performance overall, but it will have a higher likelihood of being picked in a fringe/toss-up case. Once we get to the extra-pruned cases, we have flipped our more successful class to 2, but both show steady improvement.

Table 2: Breast Cancer Class Breakdown

Class	Percent of Sample	Pre-Pruned	Post-Pruned	Extra-Pruned	Prune Improve %
2	0.650	0.799	0.865	0.957	19.732
4	0.350	0.838	0.882	0.891	6.375

3.1.2 Car Evaluation

As we move to the car evaluation data set, we can see that pruning had a detrimental performance as shown in table 1 of -7%. Looking deeper at each class in table 3, this worsened performance trickled down into each with the exception of the acceptable class. We can see though that this is the largest class by far in terms of percentage of the sample, so any generalization of the tree would push those toss-up cases in a majority vote to this larger set of available choices. Given the unbalanced distribution of data points, this data set would be an important example of where less generalization benefited our less represented class values.

Table 3: Car Evaluation Class Breakdown

Class	Percent of Sample	Pre-Pruned	Post-Pruned	Extra-Pruned	Prune Improve %
unacc	0.040	0.956	0.949	0.878	-8.128
acc	0.700	0.818	0.833	0.890	8.758
good	0.222	0.513	0.480	0.215	-58.156
vgood	0.038	0.735	0.669	0.350	-52.356

3.1.3 Congressional Voting

As we saw in table 1 and now in table 4, the congressional voting data set performed well regardless of being tested on a fully grown or pruned tree. Some generalization helped our smaller republican sample size, but these mostly binary splits (the occasional abstain vote being the exception) fit well with classifying our congress members. We benefit here from

members of each party generally voting along party lines. Even if they cross the aisle at times, eventually their voting record will guide them into their respective bucket.

Table 4: Congressional Voting Class Breakdown

Class	Percent of Sample	Pre-Pruned	Post-Pruned	Extra-Pruned	Prune Improve %
democrat	0.615	0.942	0.941	0.942	0.000
republican	0.385	0.906	0.948	0.954	5.272

3.2 Regression Tree Performance

We will then move our attention to the regression tree performance. Referring to table 5, performance improved with pruning as generalization definitely helped the predictive power of the model. Note that as hypothesized, extra pruning helped the data sets that had continuous features as well as continuous targets. The other, computer hardware, may have benefited in the integer feature values to prevent over-splitting, and therefore over-fitting, so pruning may have had detrimental effects on the predictive power of the model.

Table 5: Continuous Target Success Rate

Data Set	Pre-Pruned	Post-Pruned	Extra-Pruned	Prune Improve %
Abalone	12.402	10.882	7.980	35.656
Computer Hardware	14009.979	14107.127	19515.064	-39.294
Forest Fires	4.237	3.693	2.726	35.649

3.2.1 Abalone

As we dive deeper into the effects of pruning on the continuous target data sets, please note that we have switched from looking at a class by class breakdown, to one that involves ranges of values. As an example, our abalone data set in table 6 looks at ranges of .99 to 8, 8 to 9, 9 to 11, and 11 to 29. These are not perfect splits, but they hopefully help the reader see which data points performed best with fully grown and pruned trees.

Speaking of table 6, we can see that performance improvement was randomly scattered through each range. We saw significant improvement on the oldest abalones in the 11 to 29 bucket which makes us think that the older marine molluscs have defined features to help identify them. The same could be said for the youngest range of 1 to 8 and a middle range of 9 to 11 that each showed steady improvement throughout the pruning process. Oddly, the 8 to 9 bucket did not benefit from pruning and performed worst on a relative and absolute basis.

Table 6: Abalone Target Breakdown

Range	% Sample	Pre-Pruned	Post-Pruned	Extra-Pruned	Prune Improve %
(0.999, 8.0]	0.339	9.739	9.253	7.818	19.727
(8.0, 9.0]	0.269	10.559	19.568	16.567	-56.905
(9.0, 11.0]	0.229	9.129	8.599	4.974	45.508
(11.0, 29.0]	0.163	21.559	6.908	2.679	87.575

3.2.2 Computer Hardware

Moving on to computer hardware in table 7, we can see that most ranges improved with pruning with the exception being our lowest range which degraded in performance significantly. We cannot say for sure as to why this is because the last bucket would inherently have a higher risk of a larger error given the range it looks at. However even with a range of 50x its size, the last and largest range has a small fraction of the MSE of its smallest counterpart. This leads us to believe that even with less density within the higher range, there is more predictive power in the features to decipher the processing power of higher performing computers than with that of the lesser performing ones.

Table 7: Computer Hardware Target Breakdown

Range	% Sample	Pre-Pruned	Post-Pruned	Extra-Pruned	Prune Improve %
(5.999, 27.0]	0.261	717.662	49921.398	58446.825	-8044.063
(27.0, 50.0]	0.253	1676.650	1795.398	5692.969	-239.544
(50.0, 114.0]	0.248	3087.133	814.018	814.018	73.632
(114.0, 1150.0]	0.238	51048.583	4435.013	13988.749	72.597

3.2.3 Forest Fires

For our final forest fires data set in table 8, we saw steady improvement throughout our three ranges of values. The largest values struggled the most of the bunch, but this may be due to some data point feature that we are not considering that would cause the larger output value. Interesting as well when compared to computer hardware, the least dense and highest range of values performed worst on absolute and relative prune improvement perspective. This would compound further our concern that there is less predictive power with the features provided.

Table 8: Forest Fire Target Breakdown

Range	% Sample	Pre-Pruned	Post-Pruned	Extra-Pruned	Prune Improve %
(-0.001, 0.385]	0.500	3.514	3.022	1.942	44.725
(0.385, 2.029]	0.251	2.846	1.843	0.861	69.746
(2.029, 6.996]	0.249	7.093	6.906	6.184	12.817

4 Conclusions

When looking at our categorical results, our hypothesis seemed to have some validity to it. The car evaluation tree and congress voting tree performed well which we attribute to their categorical features and targets. Congress Voting stayed steady through pruning which we believe might have to do with its binary and well distributed target structure. Any future consideration of model performance will have to include concerns over generalization and small target populations as we saw with the unacc and vgood classes within the car evaluation data set.

Pivoting over to our breast cancer set, we were correct in thinking that pruning would improve performance. We had concerns of over-fitting due to the binary splits within the feature node decisions, but those were cleaned up in both stages of pruning.

With the regression trees, we can see some similarities in performance to our categorical sets. As predicted, our abalone and forest fires data sets benefited most from pruning given their continuous features and targets which we assumed would have the highest danger of over-fitting. On the other hand, computer hardware showed performance degradation when pruning was introduced. This could have been due to the fact that it had less over-growth given the integer features in place of continuous ones as we mentioned before, but we also may want to consider the effects of smaller data sets (209 for computer hardware versus 4000+ for abalone as example) when it comes to algorithms that cause us to subset the data significantly before making a decision as we did here.