ID3 and CART Decision Trees

Brian Loughran BLOUGHRAN618@GMAIL.COM
Department of Machine Learning
Johns Hopkins University
Baltimore, MA 21218, USA

Editor: Brian Loughran

Abstract

This document describes the ID3 and CART algorithms as a method of creating a model to predict real-world data sets. Included in this paper is a problem statement summarizing assumptions made for the algorithm and projections on how the algorithms are expected to perform against a variety of different data sets. Also discussed is a description of the experimental approach and any processing that is done on input data to fit the decision tree model and ways to decrease the size of the model. Results for the algorithm are presented for all data sets for both pruned and unpruned models. Finally, conclusions are drawn on algorithm accuracy.

1 Problem statement

Both ID3 and CART are supervised learning algorithms that can be used to solve classification and regression problems, respectively. The algorithms work by greedily splitting the data into groups and creating rules for each split to predict the result class. 3 data sets are used to evaluate the performance of ID3 classification, and those are breast-cancer-wisconsin, car and machine. Another 3 data sets are used to evaluate the performance of CART regression, and those sets are abalone, forestfires and segmentation. Predictions are made based on the trained trees, with loss functions being classification error for ID3 and mean squared error for CART.

For evaluating algorithm performance we use a 5-fold cross validation. 10% of the data is taken out before the 5-fold cross validation for optimizing hyperparameters. The remainder of the data is used for the 5-fold cross validation, and performance over each of the folds is considered to get the performance over the entire set.

There are a variety of factors that come into play when considering how ID3 and CART will perform. Factors such as the amount of data present in the data model will effect performance. Increasing the data set size will increase model accuracy in general, but will have the effect of diminishing returns as the data set increases, as well as increase compute time for a given point. Overlap between similar data points will also play a role in how well the decision trees perform. Sets where different classes are clearly separated will have an easier time learning the rules, while sets with classes that overlap greatly will be more of a challenge. Sets with a lot of data may run into the problem of overfitting the data, and different pruning methods will be used to reduce the effects of overfitting.

Using the items discussed above, we can compare and contrast data sets to predict which data sets will perform well for ID3. Breast-cancer-wisconsin is expected to perform fairly well, and the decision tree produced should be very interesting in terms of which features it considers to be most important as well as threshold values for those features. Car is expected to perform very well for similar reasons as breast-cancer-wisconsin, however the model should be slightly easier to learn than the breast cancer model. Finally, segmentation is expected to perform well but not great, ID3 is not expected to have a bias that matches this problem particularly well.

We can have the same discussion for the CART regression data sets. Abalone is expected to perform very well given the simplicity of the data set, large amount of data and the good bias match of the decision tree algorithm. Forestfires is expected to perform ok, no issues with the bias or amount of data, but forestfires has a very variable data set. Finally, machine is expected to perform well given a good amount of data and good bias match from the CART algorithm.

One issue with larger data sets is that decision trees have a tendency to overfit the data. This can lead to memorization of the data rather than generalization of the model which can lead to problems with predictions. There are methods to prune parts of trees that grow to overfit, however. One way for ID3 is to perform reduced error pruning. The idea in reduced error pruning is to begin at the root and recursively move down nodes, removing branches and checking if performance on a separate validation set improves. If performance improves or stays the same, that branch is pruned. Another method implemented in the CART algorithm is early stopping. For a branch, if there are less than a threshold value of examples in the training set, then the CART algorithm will not proceed further down the branch, and instead return a mean of the remaining data points as the prediction. Both of these methods work to reduce the size of decision trees, improving generalization capability and simplifying the model.

One other prediction to be made is that the pruned versions of both implementation (ID3 and CART) will perform slightly better than each of their unpruned counterparts. The performance enhancement is expected to be mild (<~5%).

2 Experimental Approach

ID3 and CART are well known algorithms, thus the algorithms will not be discussed in detail in this section. Some assumptions are made to streamline the algorithms which are discussed in this section, including reading in data sets and assumptions made in the computations as well as implementation details.

An important element for each of the algorithms implemented is the splitting criteria used to create the branches. Each algorithm can have a variety of splitting criteria, thus it is important to specify which criteria is used in building the tree. ID3 utilizes an gain ratio criteria to determine the splits. CART chooses sp
lits which minimize mean squared error against the data set.

Also important to each algorithm is which splits are considered. It can be computationally expensive to split the data between each point, thus some considerations can be made to choose “good” places to split. For the ID3 algorithm, for categorical data the data is split by category, however for numeric data we sort on the feature and consider possible binary splits at midpoints between adjacent data points where the result class changes. This produces a reduced set of splits which improves the computational performance of the problem while not likely effecting the performance of the model. For CART, there was very little computational issues when considering all possible splits, thus all possible splits were considered.

Both the ID3 and CART algorithms are flexible in the sense that they can handle both numeric and categorical attributes. As discussed above, for numeric attributes ID3 essentially computes a binary split of the data and then proceeds as if there are two classification groups, one above the binary split and one below. One item that the algorithms are less flexible handling are unique identifiers. Items like id# or name will likely cause the trees to overfit the data on that attribute, even when considering gain ratio criteria for ID3. To combat this, values that were deemed to potentially overfit the data were excluded from the input sets, specifically both model and PRP were removed from the machine dataset to combat overfitting on these attributes. Similar to other implementations, all index values were removed from data sets.

Another consideration for classification data sets and the ID3 algorithm is ensuring each of the sets generated (tuning set and 5 validation groups) has a representative sample from each class. The way that this was done was to start by ordering the data set as a whole based on the result. The tuning set then took every 10th data point from the set, thus resulting in a representative sample for the tuning set. Each of the validation groups took every 5th data point from the remaining set resulting in each of the validation sets also having a representative sample. Splitting the data sets in this way ensured that the tuning set and each validation fold had representative data to train and test on.

Discussed above is the dangers of overfitting the data with decision trees. The ID3 algorithm has a way to prune the tree to reduce overfitting and the method used is reduced error pruning. Reduced error pruning works in a top-down fashion, recursively checking each node to see if it can be compressed. Performance on a separate validation set is done for each cross validation fold to determine if the node can be pruned. If the performance on the validation set is better or the same as the unpruned performance, the node is pruned and the size of the tree is reduced. The same validation set is used for each of the 5 cross validation folds.

A different method is used for pruning CART trees, and that is early stopping. Early stopping sets a threshold where if there are less than the threshold number of data points that meet the criteria then the node is a leaf node. This can greatly reduce the size of a tree, and ensure that a single data point cannot do too much damage to the prediction model. The threshold is considered to be a hyperparameter, and the threshold for each of the cross validation folds is tuned using a separate validation set to evaluate performance. The threshold with the best performance is then used as the model for the test data.

Tree building is often best done using a recursive algorithm to build nodes and children. Both the ID3 and CART implementations use a recursive approach. However, for very large data sets it is possible to have stack overflow issues for the recursive algorithms. When the recursion depth reaches close to the stack overflow limit, the data set is considered to be a node similar to the early stopping criteria. This is an edge case that only occurs with the largest data sets.

One other experimental consideration is in determining which attributes are categorical and which are numeric. Different methods can be utilized, however the method used for both ID3 and CART is to scan the entire feature value range and determine whether each feature value is numeric. Only if every feature value is numeric is the feature considered to be numeric, otherwise the feature is categorical. It is important to check each feature value in the whole dataset since some features have a mix of numeric and non-numeric data, which can cause some subsets to appear numeric when they are actually categorical.

3 Results

ID3 is a classification algorithm and was run on 3 classification sets (breast-cancer-wisconsin, car, and segmentation). CART is a regression algorithm and was run on 3 regression data sets (abalone, forestfires and machine). The name of the output file is of the format set.output.txt, thus the output for cars would be cars.output.txt. For ID3, if the decision tree is to be pruned, the output file format will be set-prune.output.txt. For CART, if the decision tree should use early stopping the output file format will be set-early.output.txt. Since each of the 6 data sets have 2 conditions (pruned/not pruned) there are a total of 12 output files.

There is a wealth of information included in the output files, and referencing those files should give greater insight into low-level items not included in this report. What is included in the *output.txt files for ID3 is the following:

- The preprocessed data, as well as the splits of the data into the validation set and the 5 cross validation sets.
- The decision tree (pruned or unpruned)
- Sample outputs for all test sets for ID3 classification
- The computation of gain ratio and information gain

- Demonstration of a decision to prune a branch using reduced error pruning
- Demonstration of classification using the decision tree and the result of classification
- The performance of each fold and the cross validation fold classification error
- The average performance over each of the 5 cross validation folds

Similarly, there is a great amount of information stored in the output files for the CART algorithm. The *output.txt files for CART include the following information

- The preprocessed data, as well as the splits of the data into the validation set and the 5 cross validation sets.
- The decision tree (pruned or unpruned)
- Sample outputs for all test sets for CART classification
- The computation of mean squared error
- Demonstration of a decision to do early stopping if there are not enough data points to meet threshold
- Demonstration of regression using the decision tree and the result of regression
- The performance of each fold and the cross validation mean squared error
- The average performance over each of the 5 cross validation folds
- The average error in regression

We can summarize the results of each of the classification and regression and their pruned and unpruned versions by their loss functions. Classification uses classification error as its loss function, while regression uses mean squared error for its loss function. A summary of performance for classification is shown below:

Classification		
Set	Unpruned Accuracy	Pruned Accuracy
breast-cancer-wisconsin	65.96%	68.15%
car	94.66%	95.12%
segmentation	14.29%	18.33%

Table 1: Summary of unpruned and pruned classification accuracy for ID3 implementation

As expected, classification accuracy improves slightly for each of the data sets on the pruned version of the ID3 tree. . This is as expected, since pruning the tree improves the generalization abilities of the tree and helps to prevent overfitting.

Similarly, regression is grouped below comparing pruned and unpruned error for each of the sets.

Regression		
Set	Unpruned MSE	Pruned MSE
abalone	45.18	36.77
forestfires	50057.49	43981.85

machine	12348.1	9717

Table 2: Summary of unpruned and pruned MSE for CART implementation

Reporting mean squared error is a useful way to quickly see that the pruned version of the CART decision tree improved performance across the test set. As stated for ID3, this is expected as pruning should improve generalization capability of the tree. MSE is a bit difficult to parse in terms of how well the regression performed; another measure that can be taken is the average error for each of the predictions. This information is summarized below:

Regression		
Set	Unpruned Avg. Error	Pruned Avg. Error
abalone	2.07	1.92
forestfires	24.02	23.37
machine	15.43	14.88

Table 3: Summary of unpruned and pruned average error for CART implementation

The information contained in the average error is a bit easier to understand than in the mean squared error. The average error shows the same trend of reducing error for the pruned tree in all cases. The average error also gives us helpful information about the accuracy of the model as well. For example, for abalone, the decision tree model is accurate on average to about 2 years within the actual age. The decision tree can predict how large a forest fire will be to within 25 square acres most of the time, and can predict within $15 of machine sell price typically. This is pretty good regression for each of the sets.

4 Algorithm Behavior

It is interesting to compare and contrast the behavior of the different pruning algorithms, namely reduced error pruning for ID3 and early stopping for CART. Reduced error pruning is a top-down algorithm which is performed after the tree has been trained. Thus the space below the prune is explored in the tree building phase, and then the information is removed after the pruning phase. This is in contrast to early stopping. Early stopping is done during tree building, resulting in the possible splits after the early stopping criteria never being explored. Both are valid ways to prune a tree, and both produce more accurate performance on the data sets.

Also interesting to observe is the runtime of each algorithm. It is observed on the data sets provided that ID3 takes longer than CART to run. This can be implementation specific, however this result was interesting considering that ID3 evaluated less splits due to the binary split criteria than CART which considered all possible splits. It is likely that the reason that ID3 takes longer to compute is the fact that the gain ratio takes much longer to compute than the MSE that is computed as the splitting criteria for CART.

Another observation on algorithm behavior is on the accuracy of the tree as the tree grows. The tree begins with very poor accuracy when there are very few nodes. The accuracy then improves as the tree grows, however the improvement can only continue to a certain point. After this inflection point, the performance actually begins to get worse. This is in contrast to other discussed algorithms like k-nearest neighbor and naïve bayes, which are not likely to hit an inflection point where performance begins to degrade. Because of this, pruning is clearly important in decision tree implementations to try to get performance closer toward the inflection point where classification and regression perform at their best.

5 Conclusion

In the problem statement there were predictions for how well each algorithm would perform on each data set. While performance in relation to each data set was predicted, there were some surprises. ID3 performed very well against car as predicted, the car price model may very well fit the bias of the decision tree learning algorithm. ID3 performed less well against breast-cancer-wisconsin, lower performance was expected in relation to car, however the performance was poorer than expected. And ID3 performance was very poor against segmentation, the bias of the decision tree algorithm likely did not match the segmentation data set well. While the pruned trees worked marginally better for each of the classification data sets, the difference was not too significant.

The regression sets that were classified with the CART algorithm performed much better. While the mean squared error was difficult to parse, the average error for each of the data sets was illuminating. Predictions within an average of 2 years of age for the abalone data set is pretty good for a machine algorithm. Likewise predicting the forest fires within 25 acres and the machine price within $15 is an impressive feat for the regression CART algorithm.

It was also interesting to see the feature importance assigned to the decision trees created by both algorithms. For example, CART emphasized summer months for forest fires, and maximum memory for the machine cost data set, while ID3 emphasized cell size uniformity for breast cancer and safety for car. The feature importance determined by each of the algorithms can help determine the most important independent variables for each of the problem spaces.