Deterministic & Random Learners

Jessica Buzzelli

***Abstract*—**"Overfitting" describes the extent to which a model is likely to perform better on data it has seen (in-sample data) in comparison to data it has never seen (out-sample data). An overfit model's prediction quality depends on how extensible the patterns in the training data are to that of the testing data, but often that extensibility is hindered by parameters or prediction methodologies that were optimized for predicting values that the model had already been exposed to during training. This study explores overfitting tendencies of Tree Learner algorithms as they relate to models' learning methodologies and parameters.

1 INTRODUCTION

Models' predictions are subject to bias and randomness from the training dataset, but researchers can reduce the likelihood of overfitting the training data through introducing randomness, tuning model parameters, and training submodels on different subsets of the training set.

As a first line of defence, parameter tuning offers a straightforward way to alter how training data is aggregated or weighted to form a modeling baseline. To explore the impacts of parameter tuning, Experiment 1 manipulates instances of Decision Tree learners at varying leaf sizes to understand how the parameter influences in-sample and out-sample errors. Since leaf size determines how many datapoints are used in a point prediction, I expect that the effects of overfitting diminish as leaf size increases.

Further, an ensemble modeling technique that aggregates outcomes of multiple child models may be used to counteract the effects of training bias. As such, Experiment 2 employs ensemble modeling in the form of "bagged" Decision Trees to demonstrate how bagging can reduce overfitting in comparison to a singular instance of a child model. Since this model will be polling multiple predictions from models with different subsets of the training dataset, I expect the bagging to result in less overfitting than experienced by a singular model.

Beyond tuning input values or training subsets, models' methodologies may also be reconstructed to incorporate randomness in hopes of encouraging the model to occasionally deviate outside of its trained biases. In Experiment 3, I explore

the relative performances of Decision Tree and Random Tree learners on the same sample data and parameter values. As randomness will cause divergence from the trained outcomes, I expect the Random Tree model to result in less overfitting irrespective to leaf size.

2 METHODS

All experiments herein utilize a dataset describing national stock exchanges' performances across the 537 trading days between January 5th, 2009 and February 22, 2011. In this study, the dependent variable in question was the MSCI Emerging Markets Index (EM), which was to be predicted in relation to the Istanbul Stock Exchange Index in Turkish Lira (ISE-TL), Istanbul Stock Exchange Index in U.S. Dollars (ISE-USD), the S&P 500 Index (SP), the Deutscher Aktien Index (DAX), FTSE 100 Index (FTSE), Nikkei Index (NIKKEI), Bovespa Index (BOVESPA), and the MSCI Europe Index (EU) for a total of eight independent variables.

The date of the data observation was excluded from this study as timeseries data introduces complexities for Decision Tree and Random Decision Tree learners beyond the scope of this assignment. Further, the order of the data is made less relevant by allowing the model to view all of the other exchanges' indicies prior to predicting a value of the Emerging Markets Index for the same day.

Before beginning the experiments, the dataset was split into model training and testing subsets by randomly shuffling the order of the set's rows and selecting the first 321 rows for training (60%) and the following 215 rows for testing (40%). The same training and testing subsets were used in all models.

3 DISCUSSION

3.1 Experiment 1

A Decision Tree learner is deterministic in all aspects except leaf size, which was used to tune the models in order to determine their overfitting tendencies. Shown below in Figure 1, the models' root mean squared errors (RMSEs) for the in-sample tended to be higher than that of the out-sample data once leaf size surpassed an inflection point around 20, indicating that overfitting is likely up until that point:

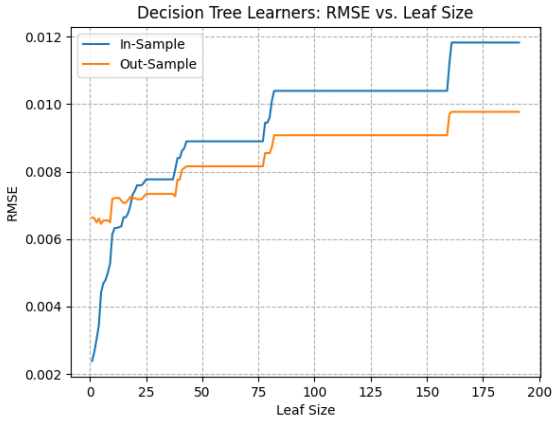

Figure 1—Decision Tree learners using the same training and testing data at various leaf sizes exhibited overfitting behavior at leaf sizes smaller than 20.

Since smaller leaf sizes cause Decision Trees to aggregate fewer values in its point predictions, it logically follows that leaf sizes smaller than 20 would favor the in-sample dataset over the out-sample. As leaf size increases beyond 20, both of the samples see an overall increase in RMSE, but the in-sample accuracy (approximated as inverse of RMSE) remains lower than that of the out-sample due to training bias. Judging from Figure 1, an appropriate leaf size for a Decision Tree learner using the Istanbul market comparison dataset would be >25, possibly up to 75 or 150 depending on the use case of the model's predictions.

3.2 Experiment 2

Bagging is a valuable tool in the fight against overfitting since there are multiple child models acting with different information subsets. Whereas a solitary Decision Tree may get misled by randomness in the training dataset, Bagged Decision Tree learners sample the training dataset – with replacement – n times to form n models, all of which have a say in the ensemble's final predictions. However, if there are too many bags, the learner may become overfit on certain subsets of the training data.

As shown in Figure 2, a fixed bag size of 5 leads to much more varied results than the single Decision Tree models used in Figure 1. That said, the same general patterns held; at a leaf size around 20, the model began to perform better than the out-sample than the in-sample, indicating less overfitting potential

among larger leaf sizes.

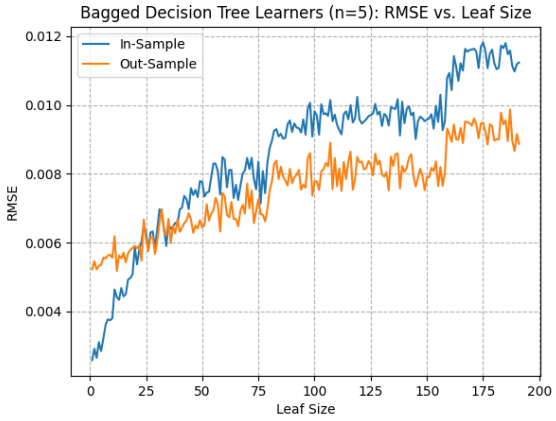

Figure 2—Bagged Decision Tree learners (n=5) using the same training and testing data at various leaf sizes exhibited overfitting behavior at leaf sizes smaller than 20.

If each bag is an "expert" on its sample of the training data it would follow that a group of more experts have a higher likelihood of predicting the correct outcomes, but small leaf sizes may still lead to overfitting on certain subsets of the training data. Overfitting is further illustrated below in a follow-up experiment where bag size was increased to 10; whereas the overall pattern from bag sizes of 1 and 5 persisted, there were fewer large jumps in RMSE at larger leaf sizes (e.g. 75, 150) due to dampened random effects:

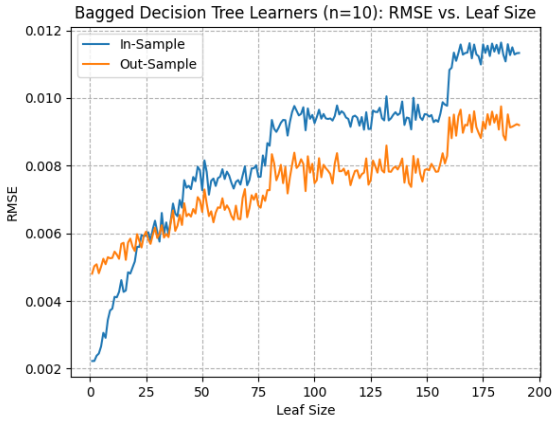

Figure 3—Bagged Decision Tree learners (n=10) using the same training and testing data at various leaf sizes exhibited overfitting behavior at leaf sizes smaller than 20.

All in all, these experiments demonstrated that bagging can reduce overfitting, but cannot overcome the effects entirely at smaller leaf sizes.

3.3 Experiment 3

Since Random Trees incorporate chance into which feature is used to split the data, my initial hypothesis was that they would surpass the predictive quality of a similar Decision Tree model at larger leaf sizes where overfitting doesn't lend as much of an advantage.

What I uncovered was that while Random Trees offered more coverage from overfitting, their results were much more sporadic across various leaf sizes than a "standard" Decision Tree learner. However, as shown in Figure 4, the Random Tree learner's out-sample mean absolute error (MAE) tended to surpass that of the Decision Tree across the board. Where one conclusion from this experiment may be that the Random Tree model is simply less accurate, the differences in MAEs across leaf sizes is typically within 0.001, or 9.86% of the mean absolute value of the dependent variable (0.01016). From this experiment, I would argue that a upper bound of testing inaccuracy at 10% is a fair and healthy trade for the Random Tree model's higher likelihood of accuracy on future unseen testing samples.

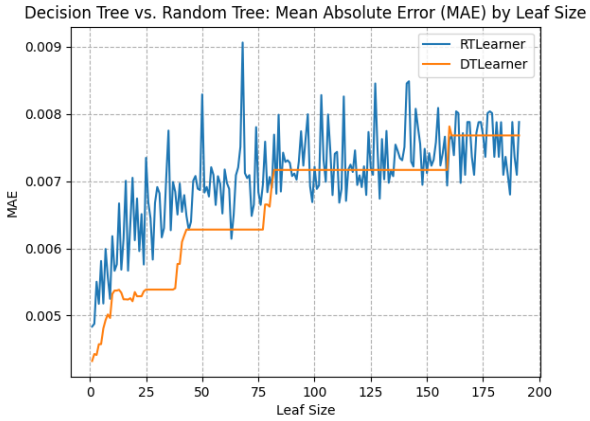

Figure 4—Decision Tree and Random Tree learners using the same training and testing data resulted in very different absolute error patterns across the tested leaf sizes.

Assuming the Random Tree is the superior modeling algorithm in this case on the grounds of accuracy, I then sought out to determine if there was a significant computational component that favored one model over the other. To conduct this subexperiment, I used Python v3.6's built-in timeit.default_timer to clock the learners' add_evidence method.[1]

As shown in Figure 5, the computational time to train each model at varying leaf sizes favored the Random Tree model up until the two models' times converged at leaf sizes of roughly 60% of the training dataset size. Assuming that differences between my RTLearner and DTLearner implementations were solely related to implementation requirements and did not misrepresent the time required to run each model, it would follow that the Random Tree learner took less time to build its decision tree since feature correlations did not need to be recalculated for each node.

1 The computer used to run the experimentation scripts is a 2020 Macbook Pro running MacOS v11.6 on a 2 GHz Quad-Core Intel Core i5 processor with Intel Iris Plus integrated graphics and 3733 MHz memory.

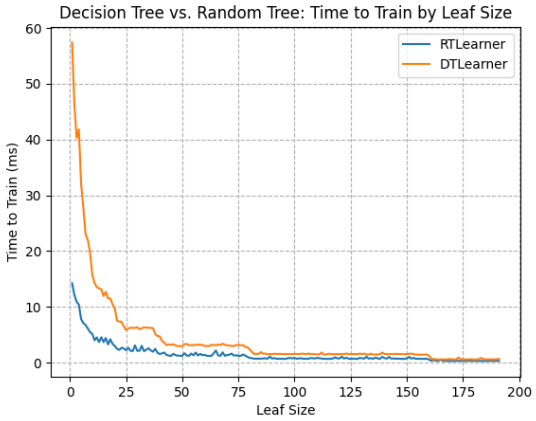

Figure 5—Decision Tree learners using the same training and testing data at various leaf sizes exhibited overfitting behavior at leaf sizes smaller than 20.

Even despite its lower relative accuracy as approximated by MAE, I'd argue that the Random Tree had superior performance among the two metrics considered. Of course, no one model will ever be a clear winner in all cases; depending on researchers' expected use cases (i.e. future test data), risk tolerance, and computational ability, one could gain ground on the other. That said, however, for the majority of cases I would argue in favor of the Random Tree model given its versatility and speed.

4 SUMMARY

Among the Decision Tree learners explored, leaf size of roughly 20 observations seemed to be the magic number where overfitting effects began to wane; however, in a setting in which the model's predictions lead to consequences, a more conservative leaf size of 50-100 would likely be more appropriate. Conversely, though there was not a truly apples-to-apples comparison of the Random Tree's resulting RMSE on in-sample and out-sample groups, Experiment 3's MAE intuition suggests that there would be a slight guardrail against overfitting when employing the RTLearner even on small leaf sizes.

Surprisingly, the bag learners did not deviate from the singular Decision Tree models' results as much as I had originally expected. In hindsight, I believe this is due to overfitting within bags that occurs at small leaf sizes that can magnify the effects of random or poorly representative training data.

As for the experiment's validity, overfitting is especially prevalent on datasets with outliers or poor extensibility, which are abound in time series stock data; however, more upfront data analysis should have been conducted to monitor possible effects on models' outcomes from the dataset. Likewise, if I were to expand upon these experiments, I would be interested in exploring the accuracy vs. runtime tradeoff of an ensemble learner that had multiple leaf sizes across its bags to better control for over/under-training. Additionally, I would want to better understand the memory requirements of the studied learners, particularly in the case of comparing various ensemble configurations of Decision Trees and Random Trees.