Francis L. Bermillo
bermillf@oregonstate.edu
Implementation Assignment 4: Decision Tree and Random Forest
CS534 Machine Learning

Introduction

For this assignment, we implemented decision trees and random forests, which were created from a provided dataset and then examined the accuracies of each algorithm and were tasked analyze the performance of different models with given hyperparameters. For the decision tree section, 11 depth parameters (dmax) were used to create the trees to analyze overfitting. In random forests, the decision tree algorithm was used to create individual trees for the random forest. The random forests were created with the given hyperparameters for tree depth (dmax), feature sub-sample (m), and ensemble tree size (T). The results and explanations for each section are shown below:

Part 1 (55 pts) : Decision Tree

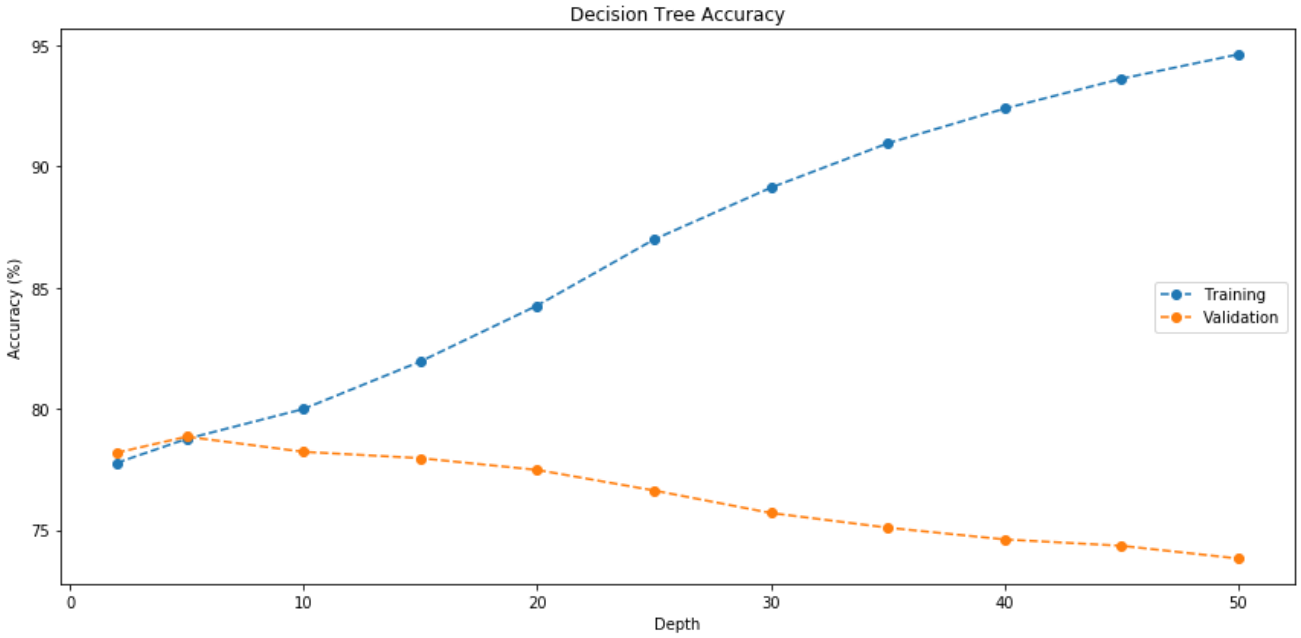
(a) What are the first three splits selected by your algorithm? This is for the root, and the two splits immediately beneath the root. What are their respective information gains?

- **Previously_Insured** (information gain = 0.308)
    - **Vehicle_Damage** (information gain = 0.035)
        - **Age_9** (information gain = 0.042)
        - **Age_0** (information gain = 0.018)
    - **Vehicle_Damage** (information gain = 0.004)
        - **Region_Code_45** (information gain = 0.002)
        - **Age_4** (information gain = 0.027)

(b) Evaluate and plot the training and validation accuracies of your trees as a function of dmax. When do you see your tree start overfitting?

- It can be observed from the plot below that the tree starts overfitting the training data for dmax > 5. This makes sense because as the decision tree grows and gets deeper, the grouped examples become smaller and more specific to the data being trained on, resulting to overfitting.

Francis L. Bermillo
bermillf@oregonstate.edu
Implementation Assignment 4: Decision Tree and Random Forest
CS534 Machine Learning

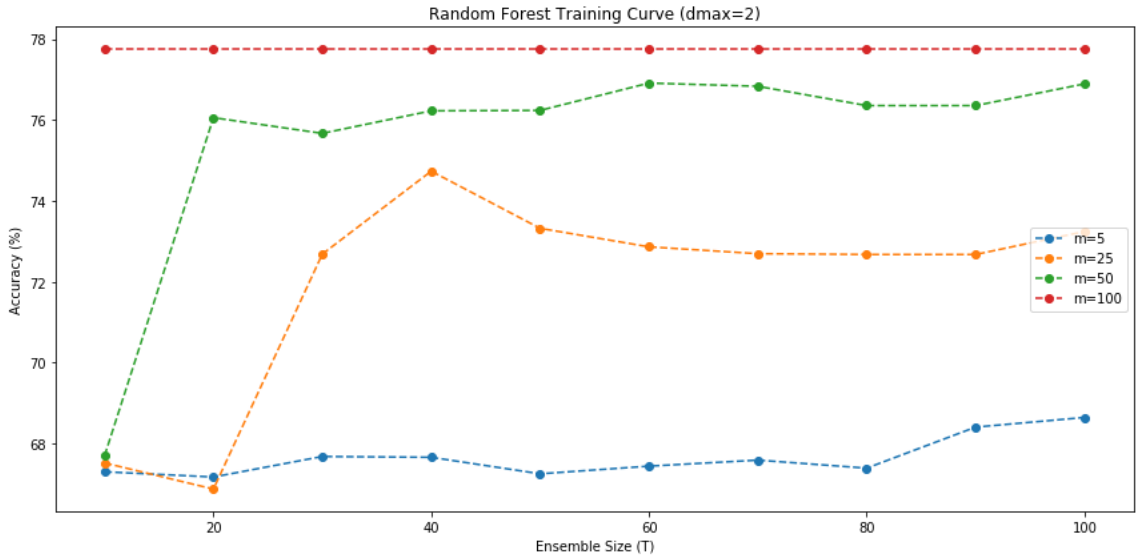


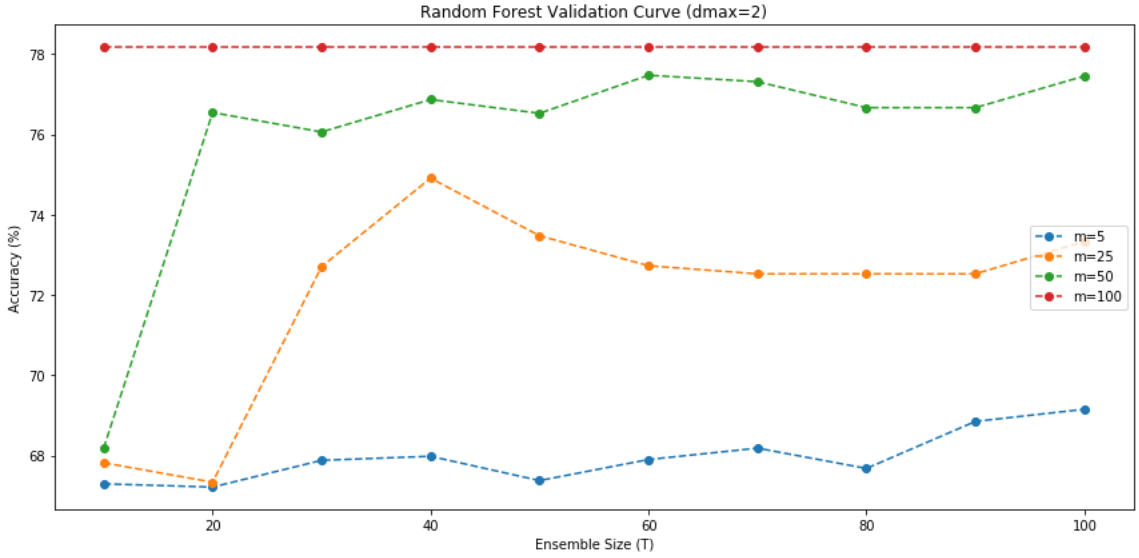
Part 2 (35 pts) : Random Forest

(a) For each dmax value, create two figures, one for training accuracy and one for validation accuracy. For the training accuracy figure, it will contain four curves, each showing the train accuracy of your random forest with a particular m value as we increase the ensemble size T = 10, 20, ..., 100. That is, plot the training accuracy (y axis) as a function of the ensemble size T (x-axis), for each m value. Be sure to use different colors/lines to indicate which curve corresponds to which m value and include a clear legend for your figure to help the readability. Repeat the same process for validation accuracy. Compare your training curves with the validation curves, do you think your model is overfitting or underfitting for particular parameter combinations? And why?

- With dmax = 2, the models seem to perform poorly compared to the other random forests in these experiments and both curves seem to have almost the same accuracies. These are signs of the models underfitting the data, which is because of the shallow trees that were created are simple models with high bias and low variance. These models do not sufficiently capture the relationship between the features and the labels leading to a high bias and poor performance even with larger ensemble size.

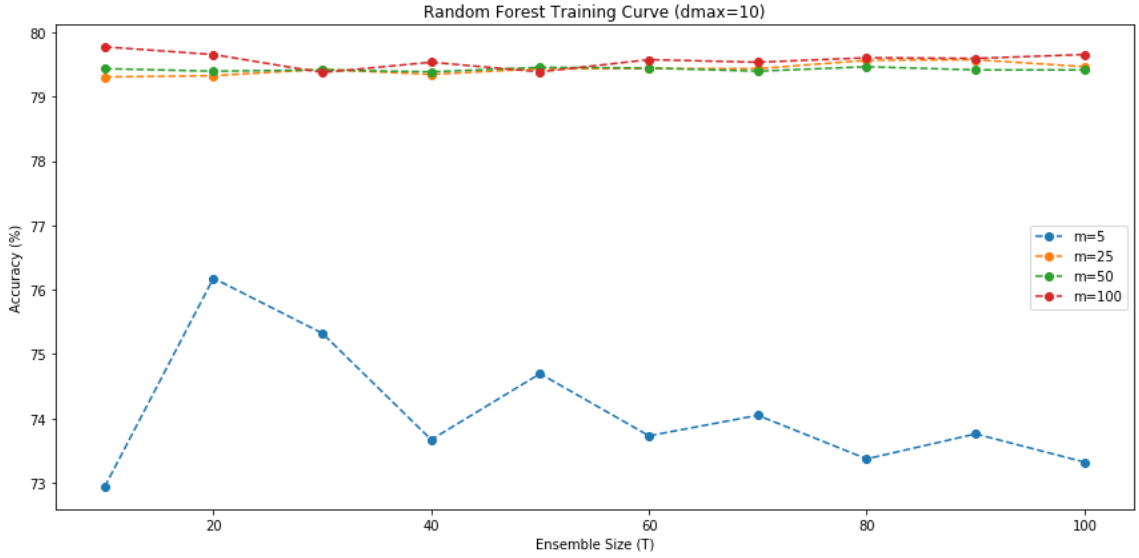
Francis L. Bermillo
bermillf@oregonstate.edu
Implementation Assignment 4: Decision Tree and Random Forest
CS534 Machine Learning

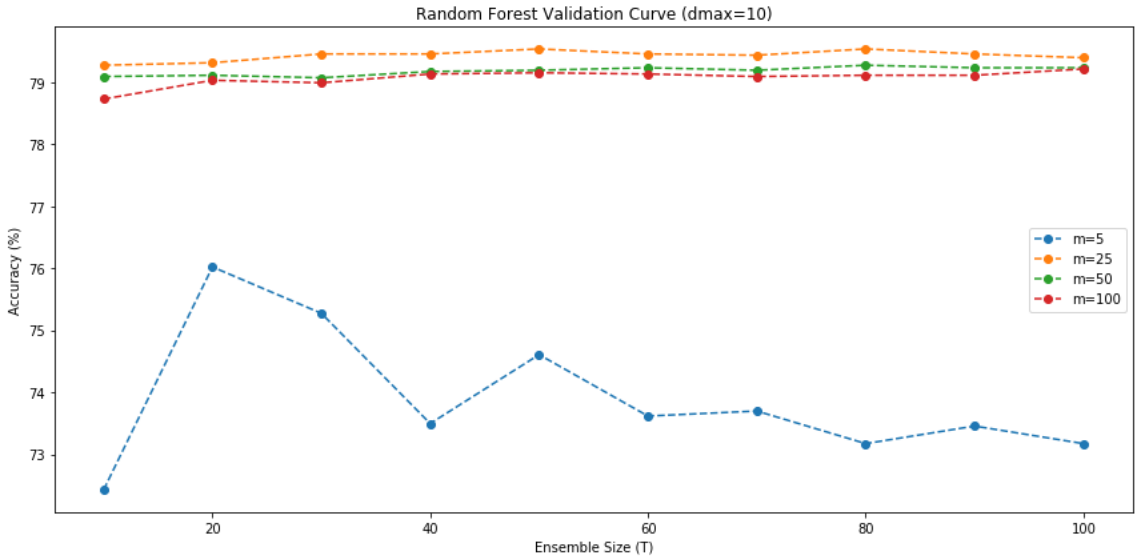




Francis L. Bermillo
bermillf@oregonstate.edu
Implementation Assignment 4: Decision Tree and Random Forest
CS534 Machine Learning

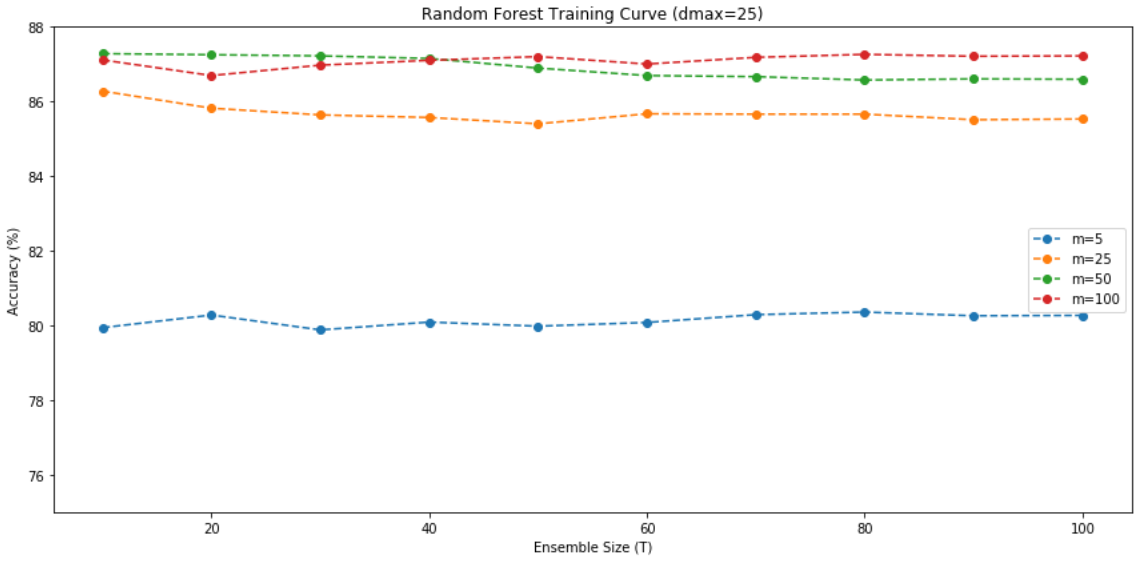
- With dmax = 10, all models are overfitting the data with the training performance slightly doing better than validation, though the performance compared to dmax=2 is much higher.
- Compared to dmax=2, all the models here are more complex, and almost fitting the data correctly, however, as the size of the feature sub-sample grows (m), these models overfit possibly due to having more features to choose from which increases variance.

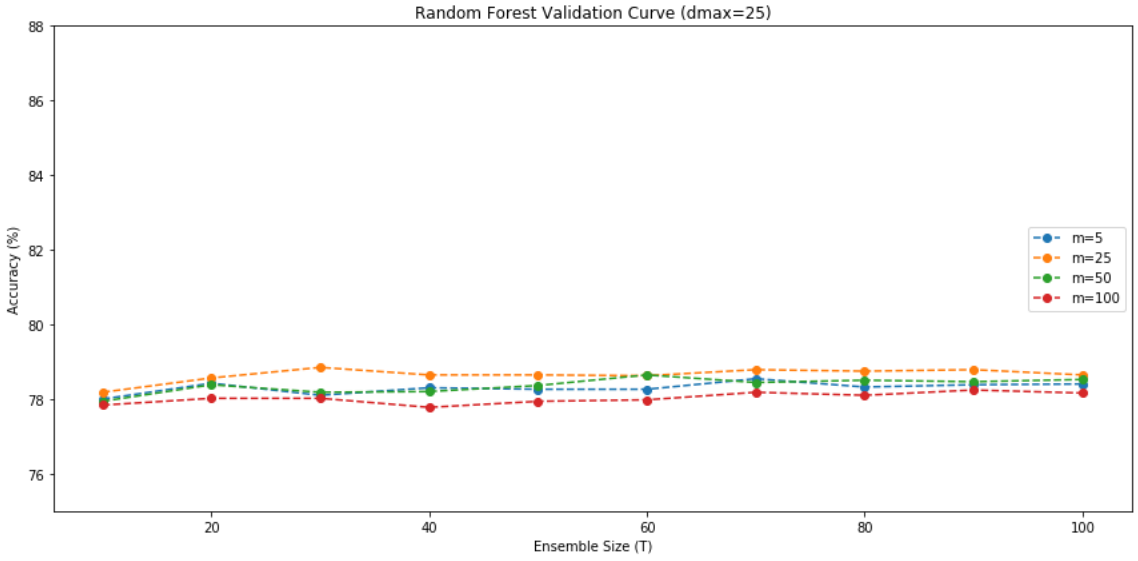




Francis L. Bermillo
bermillf@oregonstate.edu
Implementation Assignment 4: Decision Tree and Random Forest
CS534 Machine Learning

- With dmax = 25, it can be observed all the models have severely overfitted the training data regardless of the size of the ensemble or the feature sub-sample. This can be attributed to the higher complexity of the models which have low bias but high variance due to the depth of the trees.
- However, it can also be observed that as the feature sub-sample size (m) grows, the model generally overfits the data more. This again can be due to having more features to choose from every split, increasing variance even more with every tree which in turn increases the correlation between any pair of trees in the ensemble.





Francis L. Bermillo
bermillf@oregonstate.edu
Implementation Assignment 4: Decision Tree and Random Forest
CS534 Machine Learning

(b) For each dmax value, discuss what you believe is the dominating factor in the performance loss based on the concept of bias-variance decomposition. Can you suggest some alternative configurations of random forest that might lead to better performance for this data? Why do you believe so? Are there any issues inherent with the data you can find that make the performance increase difficult?

- For dmax=2, because the random forests here are composed of simpler models, the dominating factor to its performance comes from the bias. The size of the feature sub-sample (m) is inversely correlated to the bias and it can be observed from the plots that as the size of **m** grew, the performance of the random forests increased, meaning the bias became lower as the trees became more complex. Although we also see an increase in performance as **T** grew, which is correlated to lowering the variance, this parameter is overshadowed by how much the bias dominates the performance.

- For dmax=10, variance seemed to be the dominating factor because of the deeper and more complex trees. It was observed that the training performance achieved higher results than the validation which is a sign of overfitting and another observation of variance dominating was seen as the size of **m** grew, making the trees even more complex, overfitting became more obvious.

- For dmax=25, the variance significantly dominated the bias-variance composition, again because of much deeper trees and more complex models. Similar reasoning as in dmax=10, the training performance significantly did better compared to the validation performance, clearly demonstrating overfitting and high variance. And as the size of **m** grew, the performance with the training data achieved even higher performance.

- From the above analysis of the random forest configurations, by setting the dmax between 2 and 10, closer to 10, while keeping **T** large to reduce variance and **m** around 25, would reduce overfitting and may lead increase overall performance. The trees' depth and the number of feature sub-samples (**m**) seems to be the biggest determining factor for achieving better performance which can be observed from the plot above. This makes sense because these parameters contribute to the model's complexity which can control how well the random forest trees bias and keeping a large ensemble size **T** will reduce the variance.

- The number of relevant variables from the data could be making it difficult for the random forests to achieve better performances. With a large number of features variables but only a handful of them are relevant would most likely constrain the random forests achieving better performance especially with a smaller **m** size. This is because at each split, the probability that the feature variable selected is smaller.