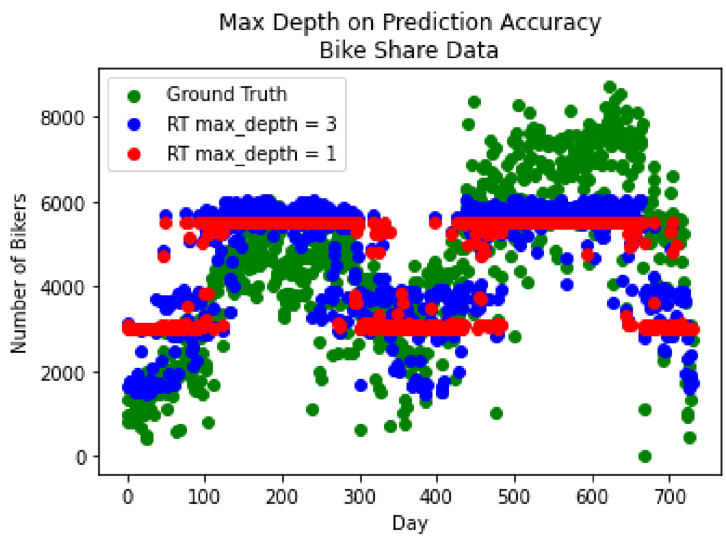
Madison Chamberlain

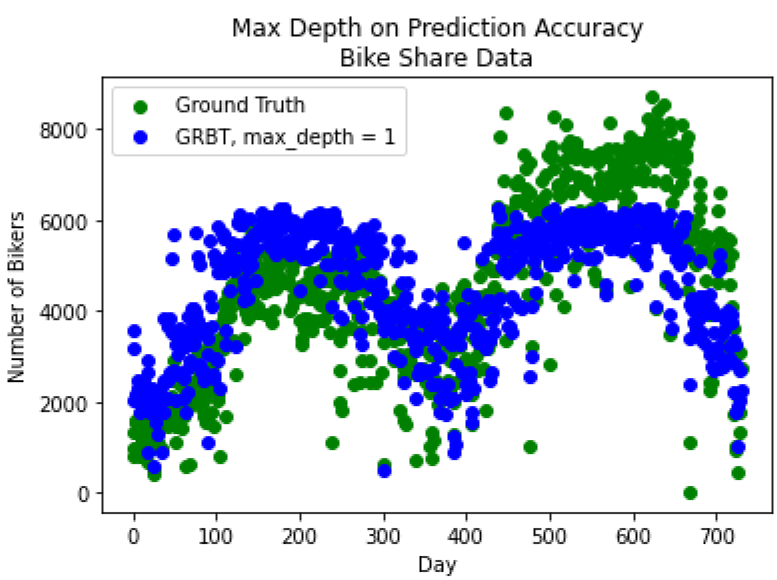
Question 5

Graph 1:



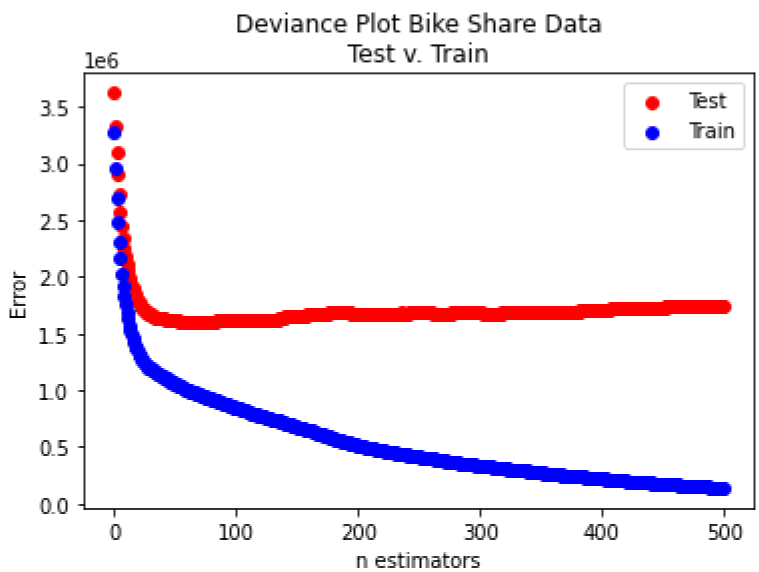
This is a graph showing day versus the number of bikers from the bike share data. The green points are the real x,y points from the dataset. The Red points are the predictions for the number of bikers on a given day using a random forest decision tree that has a maximum depth of one. The blue points are the predictions for the number of bikers given a random forest decision tree with a maximum depth of three. As you can see, the blue points mimic the pattern of the green points more than the red points which demonstrates how the variance is reduced by increasing the tree depth. The bias is also increased with the blue dots compared to the red dots, because the blue dots specifically fit the green ones while the red ones would be more likely to fit any dataset.

Graph 2:



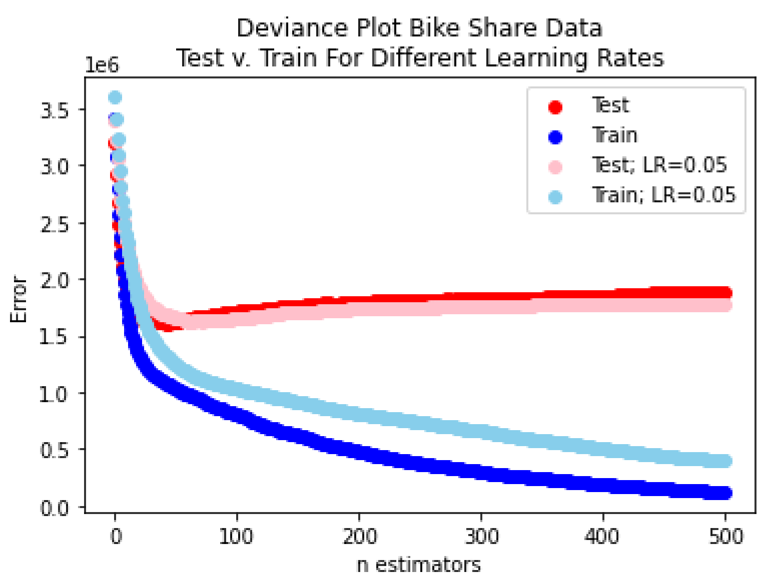
This graph shows the difference between the true data from the bike share data and predictions generated with a gradient boosting regressor tree with a max depth of one. As you can see in this graph, using a gradient boosting regression tree with a depth of one fits the points much better than a random forest with a depth of three. This means that the gradient boosting method has less variance than the random forest method since it generates more trees. However, since the predictions fit the data so well, there is a chance that overfitting took place, and that this method may not generate accurate points for another dataset. This explains why bias is also increased when using the gradient boosting regression method.

Graph 3:



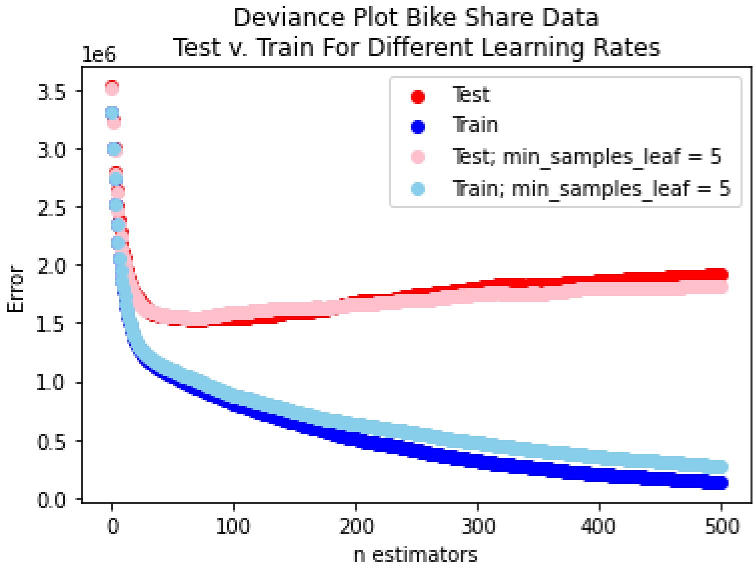
This plot shows how prediction error for the bike share data is affected by the number of trees used to generate a prediction. As you can see, when we are fitting to the training set, the more trees we use, the less error we have. On the other hand, for the test data, there is an optimized number of trees which generates the least error which lies somewhere between 0-50 trees. After this optimal number, the error begins to increase because the predictions are being overfit to the train set.

Graph 4:



This plot shows how prediction error for the bike share data is affected by the number of trees used to generate a prediction given different learning rates. The original test and train predictions are generated with the default learning rate of 0.1, while the other two lines, the light pink and light blue, have a lower learning rate of 0.05. As you can see, using a lower learning rate requires more trees to reach its minimum, but for the test set, after it hits its minimum, the predictions generated with a lower learning rate generate less error for the rest of the number of trees. This means we have controlled for overfitting slightly by using a lower learning rate.

Graph 5:



This plot shows how prediction error for the bike share data is affected by the number of trees used to generate a prediction given numbers of minimum samples per leaf. The default minimum number of samples per leaf is 1, and this is seen with the red and dark blue points, while the pink and light blue points have minimum samples per leaf of 5. Since the light blue line generated slightly more error than the dark blue line, and the pink line generates less error than the red line (after a certain point at around 200 trees), we can see that increasing the minimum number of samples per leaf has helped control for overfitting when using decision trees to generate predictions for the bike share data.