Cai Yunfeng

Project 1: Predicting Boston Housing Prices

Project 1: Predicting Boston Housing Prices

1) Statistical Analysis and Data Exploration

- Number of data points (houses) : 506
- Number of features: 13
- Minimum housing prices: 5
- Maximum housing prices: 50
- Mean of Boston housing prices: 22.5328063241
- Median of Boston housing prices: 21.2
- Standard deviation: 9.188

2) Evaluating Model Performance

- Which measure of model performance is best to use for predicting Boston housing data and analyzing the errors? Why do you think this measurement most appropriate? Why might the other measurements not be appropriate here?

Ans: Mean-square-error is probably the best for the errors-analysis of this regression model. Although the Mean-absolute-error will also works in this case, the property of Mean-square-error penalising the large errors more than the small errors and MSE also being differentiable which makes us choose Mean-square-errors as our scoring metrics.

To predict the housing price is a regression problem, which concerns the continuous data. While unsupervised or classification aims at a sort of 'True/False' prediction, which dealing with discrete data. Such that metrics of accuracy, recall and F1-score are not fit for building a regression model to predict a continuous value.

- Why is it important to split the Boston housing data into training and testing data? What happens if you do not do this?

Ans: In order to test the performance of our model, it is the direct way to use the data we have got. That leads us to split the data into two sets, one for training the model, the other to test the trained model. If we use all data to train the model, it is impossible to ensure the predictive effectiveness and the error band of the model.

- What does grid search do and why might you want to use it?

Ans: Grid search is a really handy way to make the model having the best performance through tuning multiple combinations of parameters we gave. As long as we give a range of parameters that would work well, the grid search is able to automatically compute to get the best param.

- Why is cross validation useful and why might we use it with grid search?

Ans: Cross validation splits a data set into a training set and a testing set, which makes training&testing based on the same data set which avoid the high variance(overfitting) of the trained model. With different partition of the data set, it averages the results of multiple rounds of performing cross validation. That makes all the data being used for training, and gives us a more accurate trained model for prediction.

Grid search helps select the best performance model from a group of models with different parameters. So it automatically search a parameter space we gave for the best of Cross validation, which leads to a better performance model with more appropriate parameter(s).

3) Analyzing Model Performance

- Look at all learning curve graphs provided. What is the general trend of training and testing error as training size increases?

Ans: At a given depth, the training errors increase as the training size gets larger since the more data we have, the more difficult to fit the model better. Meanwhile, the testing errors decrease from an asymptote and begin perturbing as the training size increases. On the other hand, when the max depth becomes larger, the training errors reduce since the more complicated model is able to fit the given training data nicer. However, the testing errors would be higher due to the overfitting happened by the model's increasing complexity.

- Look at the learning curves for the decision tree regressor with max depth 1 and 10 (first and last learning curve graphs). When the model is fully trained does it suffer from either high bias/underfitting or high variance/overfitting?

Ans: At max depth 1, the model suffers from high bias(underfitting) due to a large testing errors and training errors. The tree is too shallow to make use of the training data to get a better model for prediction.

Obviously, the model suffers from overfitting(high variance) when it arrives at the max depth 10 or even more depth, where the training errors go down and close to 0, the testing errors hold forward trend which seems constant. From that moment, a huge gap between test errors and training errors forms. The model perfectly fits the training data, but make a bad perdition for the testing data.

- Look at the model complexity graph. How do the training and test error relate to increasing model complexity? Based on this relationship, which model (max depth) best generalizes the dataset and why?

Ans: As the increasing of model complexity, the training error with the max depth 10 tends to 0, which causes overfitting. It is clearly at certain depth, the training errors begin to move downward and depart from the testing errors.

At max depth 5, it gives us a optimal model that can be better generalise the data set. Since at the depth 5, it balances the training error and the testing error very well. The training error keeps decreasing and far away from the overfitting at the depth 10. And the testing errors arrive at a steady error band with slight fluctuation.

4) Model Prediction

- Model makes predicted housing price with detailed model parameters (max depth) reported using grid search. Note due to the small randomization of the code it is recommended to run the program several times to identify the most common/reasonable price/model complexity.

- Compare prediction to earlier statistics and make a case if you think it is a valid model.

Ans: The prediction of the housing price with the features [11.95, 0.00, 18.100, 0, 0.6590, 5.6090, 90.00, 1.385, 24, 680.0, 20.20, 332.09, 12.13] is 20.96776316 at max depth 5, which is well in the range of mean(22.5328063241) and standard deviation(9.188). It also close to the median 21.2.