Project-1 Report

1) Statistical Analysis and Data Exploration
   - Number of data points = 506
   - Number of features = 13
   - Minimum house price = 5.0; Maximum house price = 50.0
   - Mean of prices = 22.53; Median of prices = 21.2
   - Standard deviation of prices = 9.19

2) Evaluating Model Performance

   - Best measure of model performance:

     For regression models the best measures of performance are mean absolute error or mean squared error. These measurements are appropriate as they represent the collective difference between the original and predicted values and this difference or error value can be used to fine tune the model

     I would choose **mean squared error** as the measure of model performance as more emphasis has to be given to larger errors rather than smaller errors while evaluating the performance in this scenario.

     The other popular measures are accuracy, precision, recall, F1 score which give information about how properly model is classified but not about the prediction. To be short accuracy, precision, recall and F1 would work properly when predicting targets that are in discrete form but not targets that are in continuous form.

   - Importance of splitting data:

     Splitting the available data into two sets helps us to train the data on one set and then measure the performance of the trained model on the second set. This gives us the measure of how well the model has performed when exposed to data that it has never seen before. If we don't split the data, we can never know how good the model is predicting on new data and so we cannot validate if the trained model is performing well or not.

   - Grid Search:

     In order to use a machine learning model, some initial parameters have to be set before fitting the data, these parameters are called hyper parameters. There could be many hyper parameters that can be set for a model. For example, in SVM classifier we can use different kernel functions. Choosing right set of hyper parameters is as important as training the model.

Grid Search algorithm is one of the Hyper Parameter tuning algorithms. This algorithm takes different sets of hyper parameters that can be used to configure a model and trains the model multiple times using different elements (set of hyper parameters) from the Cartesian product of the sets of hyper parameters and compares the performance of each model to select the best set of hyper parameters. Generally, the performance metric is measured by performing cross validation on the training set.

In this project the hyper parameter we are trying to tune is max depth of the decision tree regressor. Max depth parameter takes in any integer value. So while configuring the GridSearchCV we provide list of different max depths we want the algorithm to consider while deciding the best model. And mean squared error is the performance metric used to evaluate the performance.

- Cross Validation

Cross validation is a way of evaluating the performance of model during the training phase itself using an independent dataset which the model has never seen before. In cross validation the training data is partitioned into two or more sets and then the model is trained on all the partitioned sets except one and the trained model's performance is evaluated using the left out set. This can be repeated over all the sets by leaving one set during training phase and using it to validating the performance. Finally, the average performance gives over all model performance.

Here are some scenarios in which cross validation is advantageous:

1) To perform hyper parameter tuning we need to train and pick the model without touching the test dataset. In this scenario cross validation can be used to partition the training set into different partitions for training and validating the model to reduce the chance of over fitting.
2) When training a model with limited data, cross validation helps in optimized usage of available training data.

3) Analyzing Model Performance

- As the training size increases the training error gradually increases and the testing error gradually decreases until they both reach a plateau. And the difference between training and testing error reduces until it reaches a plateau.

- If a learning curve shows high error on both training and testing data, then the model is suffering from high bias and similarly if the learning curve shows large gap between training and testing sets then the model is suffering from high variance.

Figure 1 shows the Learning curve graphs for models with max depth of 1 and 10. From (a) We can observe that the training and the testing error are high and they don't improve with adding new data so this model is highly biased.

From (b) we can observe that the learning curve for the model with max depth 10, the magnitude of training and testing errors are low but the difference between them is very high, so this model has high variance.

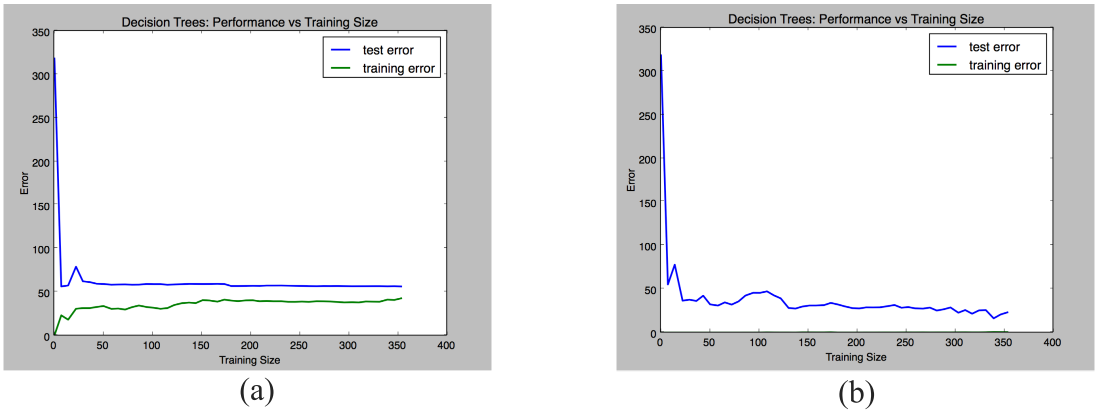

Figure 1: Max depth 1 and 10 Learning curve graphs; (a) : Graph with depth 1 ; (b) : Graph with depth 10

- We can observe that the graph shows how the model performs as the its model complexity increases. Several observations can be made, here are two if them

If we compare the training and the testing curves individually, as the model complexity is increasing the training error has decreased eventually reaching a plateau and the testing error has almost stabilized at a certain magnitude.

Here is comparison if we compared the training and testing error together. Initially when the model is less complex the difference between the training and test errors is low but the magnitudes of the errors itself is very high, this explains the high bias and low variance condition of the model when its complexity is low.

If we compare the training and test errors when the complexity is high then we can observe that magnitudes of both the errors are far apart from each other, which explains the high variance and low bias condition of the model.

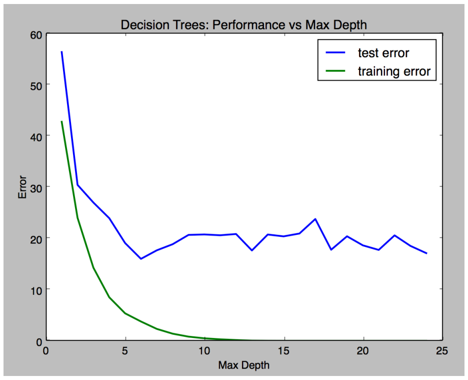

Figure 2: Performance vs Max Depth graph

- At the depth of 6 the training error is 3.77 and testing error: 16.00. The model with depth of 6 is the optimal model as after this depth the test error is plateauing and the training error is decreasing rapidly to almost zero error where it is over fitting. This is the point where there is proper trade off between bias and variance.

4) Model Prediction
   - The optimal model from the parameter tuning is the model with max depth of 6. The predicted price of the house is 20.77
   - The predicted price is -0.19 standard deviations away from the mean so it seems reasonable enough.

References:

1) http://scikit-learn.org/stable/documentation.html
2) http://datascience.stackexchange.com/questions/361/when-is-a-model-underfitted
3) Udacity Machine Learning Nano Degree Course content
4) Udacity Introduction to Machine Learning by Sabastian Thrun https://www.udacity.com/course/intro-to-machine-learning--ud120
5) https://en.wikipedia.org/wiki/Cross-validation_(statistics)
6) https://en.wikipedia.org/wiki/Hyperparameter_optimization