Questions and Report Structure

1) Statistical Analysis and Data Exploration

- Number of data points (houses)? 506
- Number of features? 13
- Minimum and maximum housing prices? $5-50
- Mean and median Boston housing prices? Mean $22.533, Median 21.1
- 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?

  Mean Squared Error provides the best measure of model performance. Using mean squared error inherently accounts for positive or negative differences in predicted and actual value. Using squared error instead of absolute error allows one to find maximums and minimums through differentiation instead of a guess-check method. The use of the mean rather than median makes our measurement of performance more sensitive to the overall error than using a median value would.

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

  Splitting the dataset allows us to train or build the model on part of the data and evaluate its performance using the other part. Building the best model by using all the data leaves us with no way to verify if our model will generalize enough with new data and will likely result in overfitting. Not utilizing enough of the dataset for model development will likely result in our model not accurately reflecting the dataset and not performing well in predictions either.

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

  Grid search is an automated method to search over a range of model parameters to find the best fit for a model and given dataset. This greatly simplifies the necessary coding requirements to tune a model to a dataset.

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

  Cross validation is a method of model parameter tuning in which a dataset is divided into uniform groups. All except one of these groups is then used to train the model and the final group is used for testing. This process is repeated until each group has been used as a test group which effectively allows the model to train over the entire dataset but not overfit the model to the data. Utilizing cross validation methods with grid search enables multiple parameter values to be tested for each iteration of cross validation which increases the efficiency of model creation and testing.

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?

  In general, testing error decreases as the training size increases and training error increases as training size increases.

- 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?

  At max depth of 1 the model appears to suffer from underfitting as the training and test error are both quite high. This is expected as the model complexity is very low and would not accurately generalize the dataset. At max depth of 10, the model appears to suffer from overfitting as the training error is very low while the test error does not get as low. This results from the high complexity too closely matching the training data and not generalizing as well with 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?

  Training and test error drop rapidly over the initial increases in model complexity. Over several runs, the best parameter found is 4. The model minimizes error on training and testing, indicating this depth best generalizes the dataset without overfitting as increasing the max depth increases testing error while training error decreases to nearly zero.

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.

  The best parameter is most often reported as a max depth of 4

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

  The model gives a prediction of approximately 20. Given the mean and median are 22 and 21, this prediction falls well within the standard deviation of 9 where we would expect most prices. Given these results, the model appears to be valid.