Statistical Analysis and Data Exploration

Number of data points (houses)	**=** 506
Number of features	**=** 13
Minimum housing price	= 5.0
Maximum housing price	= 50.0
Mean Boston housing price	= 22.53
Median Boston housing price	= 21.20
Standard deviation	= 9.19

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?

The Mean Absolute Error (MAE) is best for predicting Boston housing data since it is the more robust option for dealing with outliers in our dataset. The Mean Squared Error (MSE) will not be appropriate in this situation because it gives more weight to outliers in our dataset.

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 data into training and testing data provides a way to assess the performance of our model. Since the training dataset is the raw material used to construct our model a separate test dataset is needed to determine how well our model makes predictions. Without a separate test dataset it would be impossible to tell if our model is overfitting the data.

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

Grid search is used to optimize the hyperparameters of a learning algorithm by performing an exhaustive search of all possible combinations of a specified set of hyperparameters to find the combination that optimizes the performance of the learning algorithm on an independent data set.

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

In cross validation, the random splitting of our data into training and test sets is done multiple times and the performance of our model is evaluated over each split. The model error is computed by averaging the error across all splits.

Using grid search we can find the best performing set of hyperparameters for our model by evaluating each set using cross validation. By creating random splits for training and testing, cross validation helps us eliminate biases that may occur when we are relying on one particular split to train and test our model. Also, when our data set is small and partitioning our data set into training and testing means there's insufficient data to train the model on, using a cross validation technique called leave-one-out (LOO) can allows us to us all data for training.

Analyzing Model Performance

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

As training size increases the training error increases whereas the test error decreases. The two slowly converge.

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?

The first learning curve graph shows a model which suffers from high bias/underfitting because both training and testing errors are high when the model is fully trained.

The last learning curve graph shows a model suffering from high variance/overfitting because there is a large gap between the training curve and the testing curve once the model is fully trained.

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?

As the model complexity increases the training and test errors decrease and begin to diverge. The training error decreases to zero while the test error decreases to about 5 and then begins to plateaus. Based on this relationship the model with max depth 5 best generalizes the dataset.

Model Prediction

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

For a house with the following 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] the model predicts the price to be 21.63. This is within 1 standard deviation of our mean boston house price of 22.53.

Also comparing this to a house from the Boston data with similar features: [11.95110 0.00 18.100 0 0.6590 5.6080 100.00 1.2852 24 666.0 20.20 332.09 12.13] and an actual price of 27.90 we can see that our model's prediction is not too far off.

This is a useful model.