1) Statistical Analysis and Data Exploration

- Number of data points (houses)? **506**
- Number of features? **13**
- Minimum and maximum housing prices? **minimum 5.0, maximum 50.0**
- Mean and median Boston housing prices? **Mean 22.533, Median 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? **Mean squared error, because we are making a regression, and we want to make a prediction about the price of a house, and it's a continuous value, we care about how close the prediction is. Because with mean squared error we got all the errors on positives, it doesn't matter if we went too high or too low, we know that it was an error, also it makes more notorious the largest errors instead of smaller ones, we don't care if the price of the house was wrong by 10 dollars, we need it to empathize in larger amounts of money.**
- Why is it important to split the Boston housing data into training and testing data? What happens if you do not do this? **If we don't divide the data we can't know for sure how accurate our model is, because it will see all the data available and we can't see how it will perform with an independent dataset, also we can check if it's overfitting on the data and making a particular set and not generalizing which is our goal**
- What does grid search do and why might you want to use it? **It makes an exhaustive search by making randomly generated sets and using each one of them in different iterations as a test set, and making the rest of that iteration a training set, so we can get an average of the accuracy. We want to use it, because it gives us a better exploitation of the data available by making all our available sets into a test one, and**

giving us a more complete idea of how our model will behave with real data, because we are testing it with all the data we have at our disposition

- Why is cross validation useful and why might we use it with grid search? **Because with cross validation we don't know exactly how we should split the data, with grid search, it searches the best way to split the data so we get the best results**

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? **They become closer and closer, and even in some cases like depth 1, they reach the same value in some cases**
- 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? **With max depth equal to 1 we have a high bias, because we missed relevant information on the features of the houses by only having one level of depth to get, this explains the high level of error in both sets. With max depth equal to 10, we have a case of overfitting, because for the training set we make it perfect, making the error really low, making the model adjust so well to the training set, that houses outside this set will get a high chance of error, because they weren't on the original training set**
- 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? **The training set will always get better when we increase the model complexity, because we are looking at more details from the training set and thus, overfitting it, making the error really low, the test set on the other hand, will, at some point won't get any better, because we are looking for particular features that not everyone outside the training set will get right. The best generalization is with max depth 5, because the error on the test sets is the lowest on the graph.**

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. **After running the code several times, the best prediction for the hose price was around 21.5, and the best parameter was a max depth of 4**
- Compare prediction to earlier statistics and make a case if you think it is a valid model. **Yes, the price is always less than one standard deviation away from the mean and the median, also, when comparing our house features, and getting the nearest neighbours in terms of features, the average of the prices is really close from our prediction (+/-0.1)**