1. Statistical Analysis and Data Exploration

1.1. For the detail how to calculate, refer to the attached source code step 1.

1.2. Number of data points (houses)?
Answer: 506

1.3. Number of features?
Answer: 13

1.4. Minimum and maximum housing prices?
Minimum: 5.0
Maximum: 50.0

1.5. Mean and median Boston housing prices?
Mean: 22.5
Median: 21.2

1.6. Standard deviation?
Standard deviation: 9.188012

2. Evaluating Model Performance

2.1. 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?

SKlearn provides 5 metric functions for regression model.
Bellows are the name of metrics definition and explanation whether appropriate for the model.
$\hat{y}_i$ is the predicted value of the i-th sample and y_i is the corresponding true value.
As a conclusion, the most appropriate metric function of 5 is r2_score.

- mean_squared_error(MSE)

$$\text{MSE}(y, \hat{y}) = \frac{1}{n_\text{samples}} \sum_{i=0}^{n_\text{samples}-1} (y_i - \hat{y}_i)^2.$$

This is good metrics for predicting continuous value. This metrics put weight on the distance between predict and true value. Better score means the model can predict value close to true value.

- mean_absolute_error (MAE)

$$\text{MAE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} |y_i - \hat{y}_i| .$$

This metric put weight the difference between predict and true value. But MSE is better metrics since it put weight on distance.

Following is the example case.

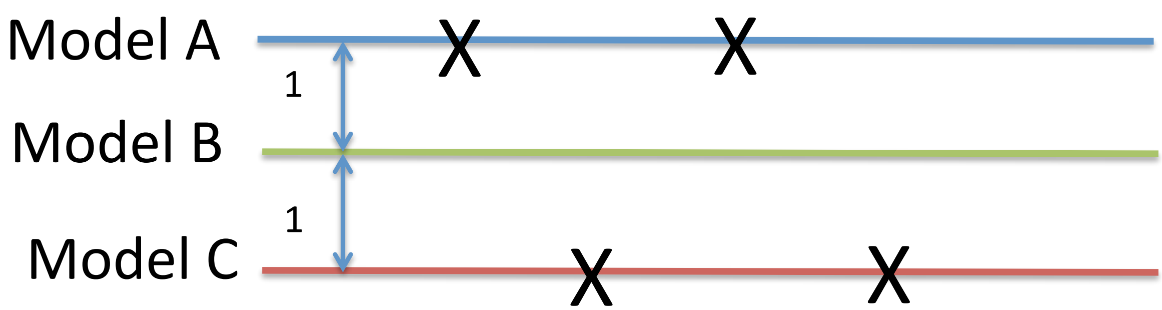

Fig. 1 example case of 3 predicting model and true value.

	Model A	Model B	Model C
MAE score	4	4	4
MSE score	8	4	8

X means true value, and lines are candidates of regression model.
In this case, Model B is the best one since it is closer to each true value than other models.
MAE is not possible to find the best model since all models have same score 4. On the other hand, MSE can find the best model B.

- explained_variance_score

$$\texttt{explained\_variance}(y, \hat{y}) = 1 - \frac{Var\{y - \hat{y}\}}{Var\{y\}}$$

This metrics is to evaluate of Variance of difference between predict and true value. It is not appropriate for the predict price.

For example, following test result gets the best score 1.0 in explained_variance_score even though the difference between true and predict value is big.

Index	True value y_i	Predict Value $\hat{y}_i$
1	10	20
2	20	30
3	30	40
4	40	50

- median_absolute_error

$$\text{MedAE}(y, \hat{y}) = \text{median}(\mid y_1 - \hat{y}_1 \mid, \ldots, \mid y_n - \hat{y}_n \mid).$$

This metrics is not appropriate. Since it uses only one value for the score. It cannot fit the model over space of sample.

- r2_score

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n_{\text{samples}}-1}(y_i - \hat{y}_i)^2}{\sum_{i=0}^{n_{\text{samples}}-1}(y_i - \bar{y})^2}$$

r2_score is the most appropriate for predicting house price in this 5 functions. It put weight on the distance between prediction and true value like MSE.
This is better metric than MSE since it is scaled with variance of true values then we can compare the results without affection from variance of test dataset.
In this report, r2_score is used as a model performance.

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

If the model knows the answer of test data, the model can get good result in the test even though it cannot predict other data not included in training.

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

Grid search is faster way to find better parameters of function than humans do.
Grid search execute training and test with all combinations of parameters. Then it tells the best parameter according to test result. Generally it is

difficult to find good parameter over all combinations of multiple continuous parameters for humans since it takes too much time for testing.

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

If the test is executed only once, there is a case model is evaluated with biased training and test data set at random. If data sets are biased to specific group, the model fits on specific cases and its test result gets worse.
By execute evaluation in multiple times, that specific case affection on result gets stochastic low.

3. Analyzing Model Performance

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

When training data increases, training error also increases, but testing error decreases.
When the number of max depth is small, after number of training data exceeds a certain number, both errors converge on near value.
When the number of max depth is big, training error converge on 0 but testing error converge on bigger value than 0.

3.2. 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?

Refer to the Fig 2, 3.
Compare test errors on both curves. The test error with max depth 1 converges on about 0.4. And the one with max depth 10 converges on 1.0. Since test errors one with max depth 1 is lower than one with max depth 10, the one with max depth 1 suffer from bias/underfitting.

Next compare the difference between training and test errors in Fig 3. Even though training error is almost on 1.0 which is the best score in r2_score, the test error stay around 0.7. It means the one with max depth 10 suffers from variance/overfitting.

Fig. 2 learning curve (Depth = 1)

Fig. 3 learning curve (Depth = 10)

3.3. 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?

Refer to the Fig. 4.

Until max depth 3, the both error curves simply increase. After that, training error continue to converges on 1.0 but test error stay between 0.7 and 0.8.
Since difference between both error curves get bigger, the model seems to suffer from variance/overfitting after max depth 4.
Thus the best model is one with max depth 3.

Fig. 4 Complexity graph

4. Model Prediction

4.1. 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.

Following is 10 results of the prediction of house which has feature [[11.95, 0.00, 18.100, 0, 0.6590, 5.6090, 90.00, 1.385, 24, 680.0, 20.20, 332.09, 12.13]] from a model tuned with same model parameters.

No.	Predict value	Max depth
1	22.9052	3
2	22.9052	3
3	22.9052	3

4	22.9052	3
5	22.9052	3
6	22.9052	3
7	22.9052	3
8	18.8167	8
9	22.9052	3
10	22.9052	3

We can see the randomization on results. This is because of that the result of grid search CV function has randomization for selecting training and test dataset.
I choose prediction value 22.9052 (max depth3) since it appears the most frequently.

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

The model is valid since its prediction value 22.9052 is enough close to the mean value 22.5 and median value 21.2 of Boston housing price dataset. Considering standard deviation 9.19, the prediction is not too high price and not too low price in the distribution of Boston house price.