Project I: Predicting Boston House Pricing

1 Statistical Analysis and Data Exploration

The data exploration gives the following results:

- Data size (number of houses): 506
- Number of features: 13
- Minimum price: 5.0
- Maximum price: 50.0
- Mean price: 22.532
- Median price: 21.200
- Standard deviation: 9.188

The median and mean prices are not much different – this is an indication that the data not contains extreme outliers

2 Evaluating Model Performance

2.1 Performance Metric

The model is a regression model; therefore, we have to select a regression metric. The most used regression metric is mean squared error but I choose the median absolute error, because this metric is not as sensitive to outliers as the mean squared error metric.

2.2 Testing/Training Split

It is necessary to split the data into a training and a test set. The training set used to find the optimal parameters of the regression model. The test set is used to evaluate the performance of the regression model with data the regression model has never seen during training. Therefore, we can test if the regression model predicts well. If we use the whole data for training, we have no data to evaluate the performance of the regression model with unknown data.

2.3 Grid Search

Grid search is used to find the optimal depth of the decision tree. It works as follows:

- Set up a parameter set for the grid search. Here we use the parameter set parameters = {'max_depth':(1,2,3,4,5,6,7,8,9,10)}.
  This means we use the parameter max_depth in the range from 1 to 10.
  I use cross validation with cv03
- For each parameter entry in the parameter set do:
  - Split the trainings data set into 3 parts randomly. Use 2 parts for training, one part for testing the model. Find the best model for the selected parameter value. Do this 3 times and calculate the mean over all test error for a specific parameter of the parameter set. This is called 3fold cross validation. Store this mean test error.
- Select the model with the lowest mean test error. Use this model to predict the price for the data provided.

It is important not to use any data of the test set for grid search.

Cross validation is very useful for the following reasons:

1. It is necessary to split the trainings set into a set for training the model with a specific parameter and use one part of the trainings set to evaluate the model.
2. Cross validation allows it to use the full trainings set to do it, because we do several runs with other random splits.

3 Analyzing Model Performance

3.1 Analyis of the learning curves

The following graphs shows the learning curves for max_depth=1 to 10.

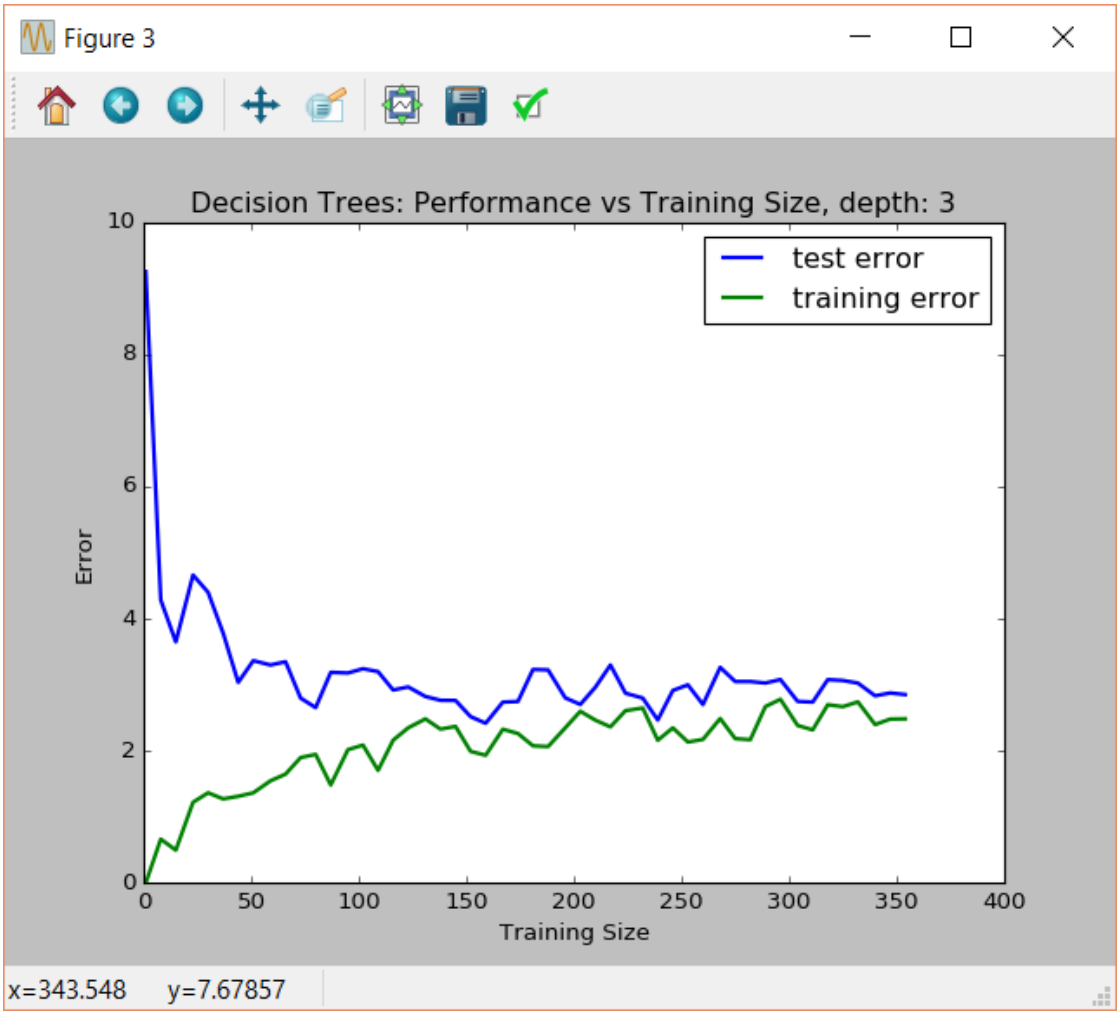
Figure 3
Decision Trees: Performance vs Training Size, depth: 3
test error
training error
Error
Training Size
0
2
4
6
8
10
0
50
100
150
200
250
300
350
400
x=343.548 y=7.67857


Figure 4
Decision Trees: Performance vs Training Size, depth: 4
test error
training error
Error
Training Size
0
2
4
6
8
10
0
50
100
150
200
250
300
350
400

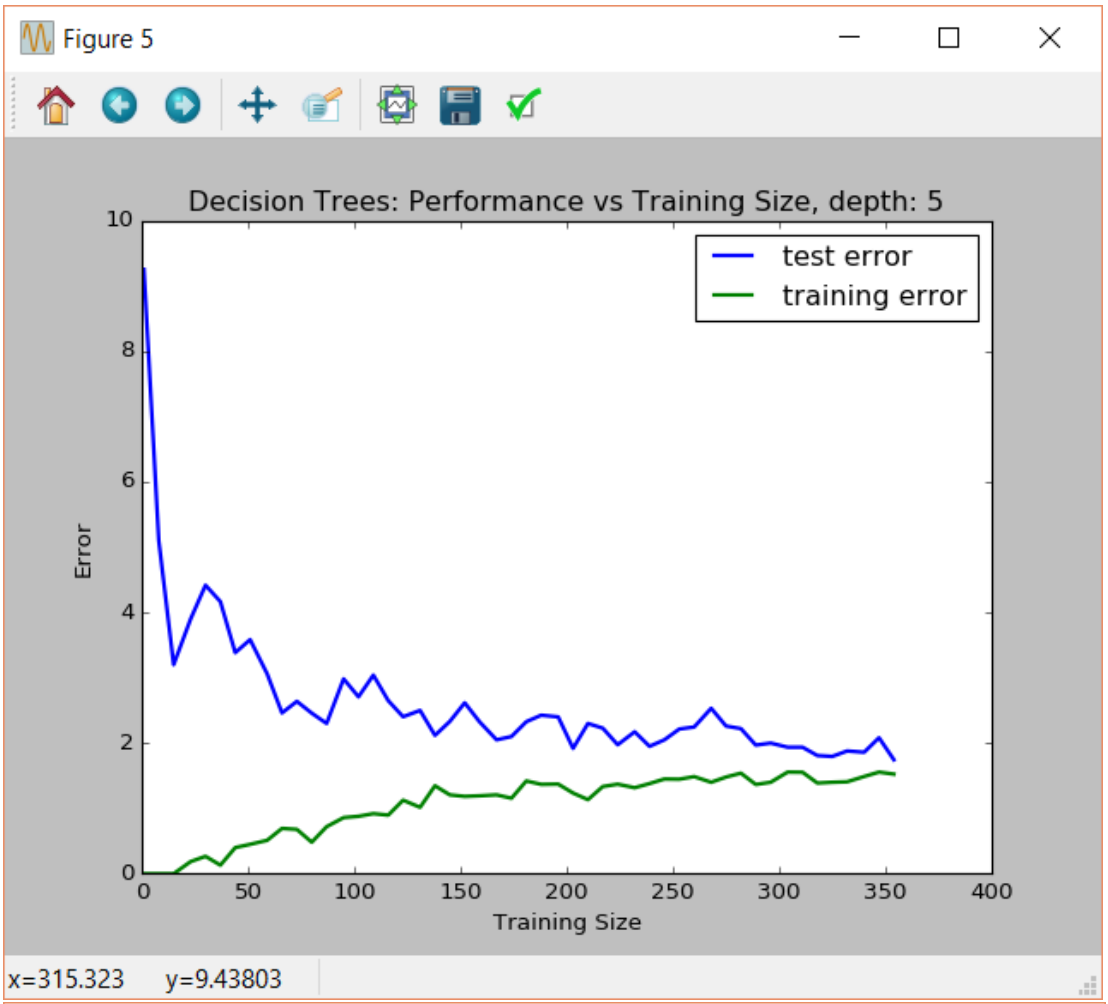
Figure 5
Decision Trees: Performance vs Training Size, depth: 5
test error
training error
Error
Training Size
0
2
4
6
8
10
0
50
100
150
200
250
300
350
400
x=315.323 y=9.43803

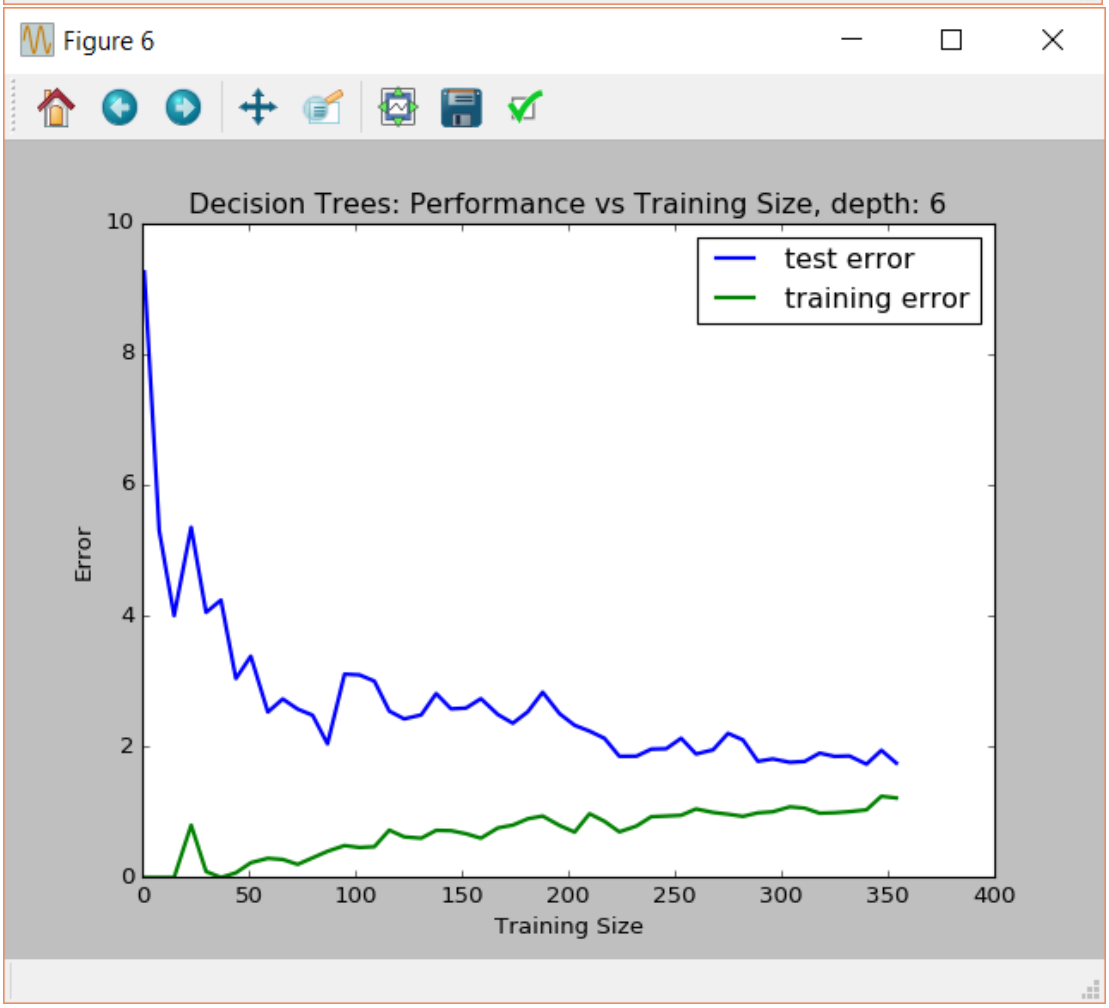
Figure 6
Decision Trees: Performance vs Training Size, depth: 6
test error
training error
Error
Training Size
0
2
4
6
8
10
0
50
100
150
200
250
300
350
400

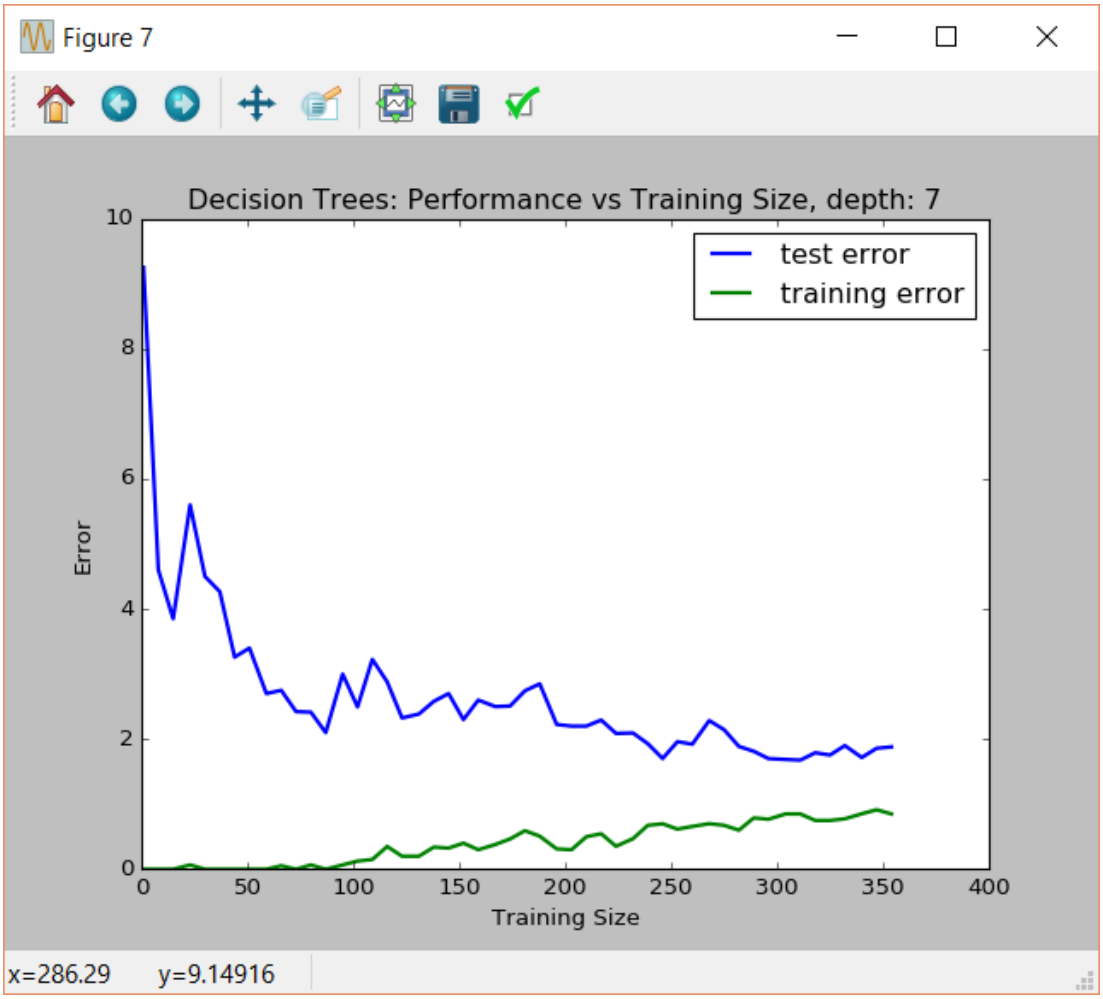
Figure 7
Decision Trees: Performance vs Training Size, depth: 7
test error
training error
Error
Training Size
0
2
4
6
8
10
0
50
100
150
200
250
300
350
400
x=286.29
y=9.14916

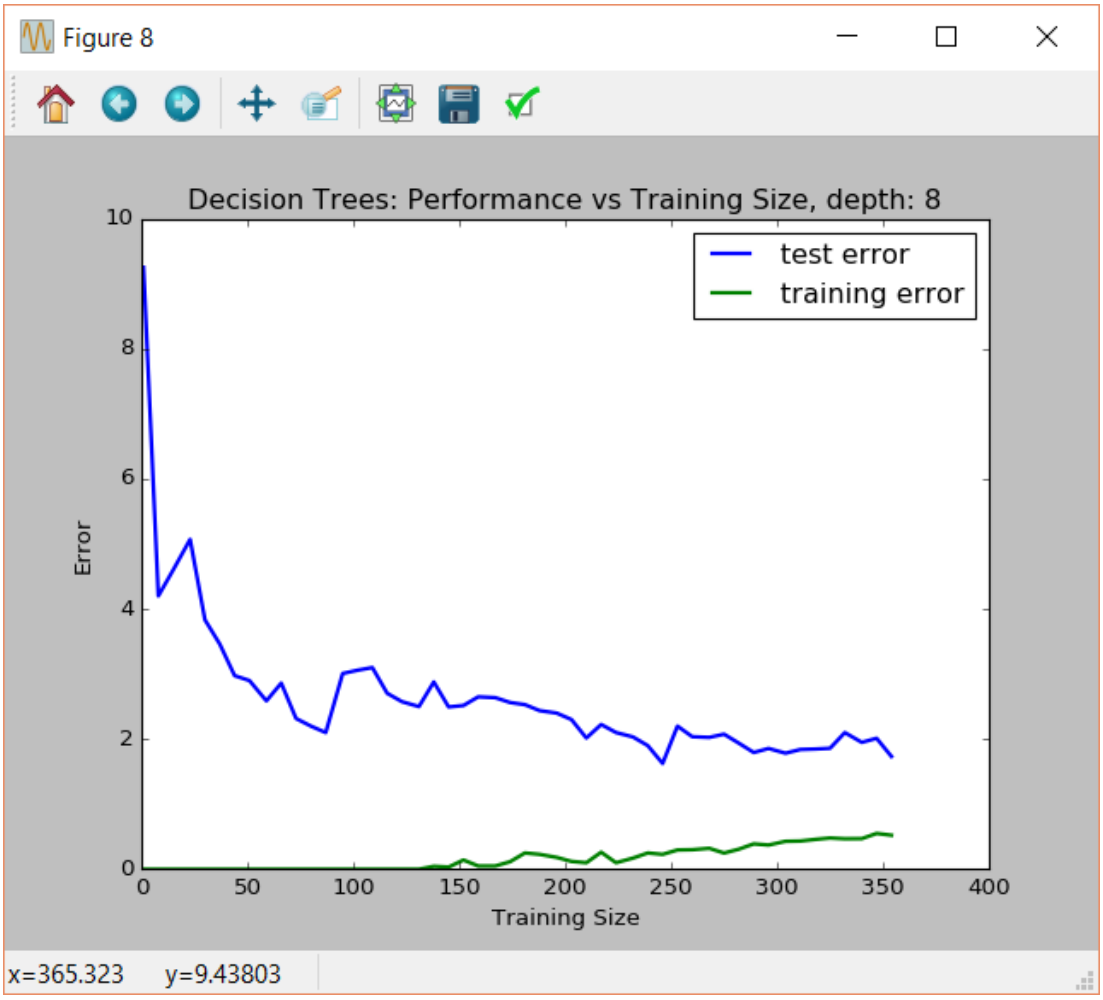
Figure 8
Decision Trees: Performance vs Training Size, depth: 8
test error
training error
Error
Training Size
0
2
4
6
8
10
0
50
100
150
200
250
300
350
400
x=365.323
y=9.43803


Figure 9
Decision Trees: Performance vs Training Size, depth: 9
test error
training error
Error
Training Size
0
2
4
6
8
10
0
50
100
150
200
250
300
350
400


Figure 10
Decision Trees: Performance vs Training Size, depth: 10
test error
training error
Error
Training Size
0
2
4
6
8
10
0
50
100
150
200
250
300
350
400

The general trend is: with increasing training sizes the training error increases and the test error decreases.

3.2 Analysis for max depth 1 and 10

For max depth = 1 the test error and training error is rather high (about 4) and not changes much with increasing training size. This is an example for high bias/underfitting. The model generalize quite well but the error is quite high – an indication for underfitting.

For max depth =10 the training error is very low - about zero. But the test error is quite high – about 2. This is a clear indication for overfitting, because the model can't predict new data well. It only performs well on the data of the training set. This is an example for high variance/overfitting.

3.3 Error Curves and model complexity and picking the optimal model

The graph shows that for increasing max depth the training error goes down to zero and the test error decreases with increasing max depth too for max depth between 1 and 5, but higher max depth not decreases test error further. This shows that for increasing model complexity an optimal max depth exists. If we add more model complexity, we will not get a better prediction. From the graph a best max depth value =5.

3.4 Model Prediction

Grid search selects an optimal max depth of 5. The predicted house price is: 20.967 This price is reasonable because the data exploration (part 1) shows:

- Mean price: 22.532
- Median price: 21.200

- Standard deviation: 9.188

The predicted house price fits well in the price range of the mean price +/- standard deviation.