Machine Learning Nanodegree
P1: Boston Housing Prediction
Jose Augusto Montiel

Statistical Analysis and Data Exploration

- Number of data points (houses)?
    - 506 Houses
- Number of features?
    - 13 Features
- Minimum and maximum housing prices?
    - Minimum price: USD 5,000
    - Maximum price: USD 50,000
- Mean and median Boston housing prices?
    - Mean price: USD 22,532
    - Median price: USD 21,200
- Standard deviation?
    - 9.188

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 (MSE) was chosen as measurement of model performance due to its ability of quantifying how much of the data does not fit into the generated model (the mean of original values minus the model's values squared). MSE was selected over other model performance measurements, like Mean Absolute Error (MEA), due to MSE's sensibility to outliers; it puts more weight toward the lower and upper bounds of the data (thanks to its "squared" nature). This allows us to take into consideration overinflated house prices due to unknown features, for example, a

house where a celebrity lived or with historic value would alter our prediction for houses with similar features since “historic value” is not being taken into account by the model.

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

Splitting data into training and testing sets allows us to experiment with the generated model in order to quantify its performance; by not allocating a portion of the data to a testing set we might be overfitting it, ending up with a high-variance model and wrong predictions for unknown data.

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

Grid search implements an iterative method for calculating the inherent parameters of a given estimator, this allows us to come up with a model better fitted to our data.

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

Cross validation along grid search allows us to divide the data set into k-number of bins, one to be used as testing data while the rest is used as training data, this process is repeated switching the training bin; The performance of the model is calculated by averaging the errors of each iteration; by doing this we can test our model without “sacrificing” any data by using it only for testing.

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 the training size increases, the error for the testing curve decreases while the error for the training curve increases, trying to converge along a horizontal asymptote.

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?

For a max depth of 1 we can notice the model suffers of a high bias due to its high error value. However, for max depth 10 high variance could be argued, although when the error value is

still “decent”, the trend from max depth 6 and up is for the error of the testing curve to keep increasing (shifting upward).

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?

Both training and test error decrease until the optimal max depth for the data given is reached, beyond that point error plateaus or increases due to overfitting of the model. For the given data a max depth of 4 is optimal; a higher depth overfits the model to outliers, increasing the error of our predictions.

Model Prediction

Prediction: USD 21,630

Hyperparameter:

Max Depth = 4

The NearestNeighbors class was implemented to retrieve the 10 most similar houses to the one being predicted from the given data, these had a mean price of USD 21,520, ~ 0.5% difference from our predicted value; this way we can assure that the model generated is a good fit for our purpose.