Predicting Boston Housing Prices

Project Description

You want to be the best real estate agent out there. In order to compete with other agents in your area, you decide to use machine learning. You are going to use various statistical analysis tools to build the best model to predict the value of a given house. Your task is to find the best price your client can sell their house at. The best guess from a model is one that best generalizes the data.

For this assignment your client has a house with the following feature set: [11.95, 0.00, 18.100, 0, 0.6590, 5.6090, 90.00, 1.385, 24, 680.0, 20.20, 332.09, 12.13]. To get started, use the example scikit implementation. You will have to modify the code slightly to get the file up and running.

Statistical Analysis and Data Exploration

Statistic	Value
No of data points	506
No of features	13
Min housing price	5.0
Max housing price	50.0
Mean housing price	22.5
Median housing price	21.2
Std dev of housing prices	9.2

Evaluating Model Performance

Which measure of model performance is best to use for regression and predicting Boston housing data? Why is this measurement most appropriate? Why might the other measurements not be appropriate here?

> The mean squared error (MSE) is the best metric for this Boston housing regression model as it is a measurement of consistency in the predictions. MSE is an average of the squares of the difference between the actual observations and those predicted. In the case of a regression model where predictions are extrapolated or interpolated, the recurring proximity of the predicted value to the true value is much more appropriate than the accuracy, precision and recall of values. Hence, means squared error is a good measurement. Alternatively, mean absolute error or r-squared could be used as well.

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

> It is important to split the dataset into training and testing sets so that we can build a model from the training data and evaluate and optimize the model from the testing

> data. The model and all its intricacies are created from the training data only. Without a testing partition, it would be difficult to ensure that the model generalizes well enough to predict accurate or consistent results.

Which cross validation technique do you think is most appropriate and why?

> A fairly simple train-test-split technique is used, which can randomly split the Boston dataset into training and test sets quickly. It is needed to ensure the model is not overfitted on the same training data, and enables generalization of the model for better predictions.

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

> Grid search facilitates a machine learning algorithm to be exhaustively searched over numerous specified parameter values. It is invaluable in fine-tuning and optimizing a model to it best.

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 values decreased.

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?

> When the model is fully trained, both the test and training plateau and converge at a low error value. This suggests the model has learned as much as it can from the data is ideal. It is neither underfitted or overfitted.

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 error appears to steadily decrease. The test error seems to plateau at about a max depth of four, and the training error decays inversely to the error where it drops sharply at a max depth of

four and plateaus at about a max depth of fifteen. Based on this observation, a model of max depth of four is ideal as it generalizes the data well enough without having to incur greater computational costs.

Model Prediction

From the grid search using max depth as the parameter of interest and mean squared error as the scoring method, the following algorithm was given as the best estimator for the dataset.

```
DecisionTreeRegressor(criterion='mse', max_depth=4, max_features=None,
max_leaf_nodes=None, min_samples_leaf=1, min_samples_split=2,
min_weight_fraction_leaf=0.0, random_state=None,
splitter='best')
```

```
House: [11.95, 0.0, 18.1, 0, 0.659, 5.609, 90.0, 1.385, 24, 680.0, 20.2, 332.09,
12.13]
Prediction: 21.62974359
```

Inputting the given house feature data into the model, the predicted price is 21.6. This value is reasonable as it is just below the average house price of 22.5 and within an order of standard deviation of 9.2. Also it is within the range of house prices in Boston, from 5.0 to 50.0.

Resources

- What's the bottom line? How to compare models
- Why Data Scientists Split Data into Train and Test