Boston house pricing

Shmuel Naaman

Questions and Report Structure

1) Statistical Analysis and Data Exploration

- Number of data points (houses)? 506
- Number of features? 13
- Minimum and maximum housing prices? 5 and 50
- Mean and median Boston housing prices? 22.532 and 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?

I used the “coefficient of determination”. I am not sure if there is the BEST metric in this case. There are few metrics that are reasonable. To select the appropriate metric I consider the following.

- The data is continues, therefore one of the metrics that work with regression should work.

- Looking at the histogram, we cannot claim the prices distribute normally, but considering standard deviation we can claim that the mean and medians are quite close therefore there is no reason to preffer a method that consider the median or the mean.

- Other metric that might fit are the mean or “median absolute error regression loss”, “Mean squared error regression loss” and “explained variance regression score function”, are all summarize performance in ways that disregard the direction of over- or under- prediction. And indeed, in this case we want the price to be closer as possible to the real price and penalize equally for higher or lower price. The reason that I choose the “coefficient of determination” is that the distance (X^2) behave better than the absolute.

In addition there are other metrics that are not in the category of regression. For example classification metric, will suit for a model that deal with classification. A multilabel metric will suit for a model that deal with more

than one labels. Clustering metric will suit for models that deal with information content. Therefore all these metric are not suitable in this case.

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

When we are building a prediction model, there is always a risk of over fitting from one side and under fitting on the other side. The reason for that is that in most practical cases, you can increase the number of the model parameters (or degree of freedom) up to the point that the model will perfectly fit the data. However, such a model will not be practical and in particular will be useless when it comes to prediction a new set of data. Therefore it is important to train the model on one set of data and test the model on another set of data. When doing so we balance the model between over fitting and under fitting.

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

Exhaustive search over specified parameter values for an estimator. Gridsearch is a function that allows us to test the performance of a given predictive model while using cross validation. Practically the function changing one parameter while measuring the performance. The output of this function, and that what make it so useful is the optimal parameter for the given model.

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

Cross validation is a technique that allows us to estimate the performance of the model in the general case. Since the grid search is an optimization method, we want to avoid the situation of over fitting. Therefore using different set for training and testing (cross validation) will make sure we are not over fitting the model while optimizing the parameters.

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?

Training error increase together with the size of the training sample (in the chart it seems as decrease since we are looking at R squared).

Testing error decrease together with the size of the training sample (in the chart it seems as increase since we are looking at R squared) .

- 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 the case of depth 1, when the model is fully trained, R squared for the testing set and the training set is around 0.4, therefore we can consider that as under fitting.

For the case of depth 10, when the model is fully trained, R squared for the testing set is around 0.8 which is a reasonable good value. However for the training set R squared equal to 1 which is a good indication for overfiting.

- 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?

From the model complexity graph we can see that as the max depth increase, training error decrease till it reach 0, while the testing error decrease up to a point and then stabilize around this value. From this chart we can estimate that a model with max depth of ~ 5 will best generalize the data.

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.

Max depth = 4

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

Prediction: [21.62974359]. The value that we obtained is reasonable; most of the houses in the data set will be in this price range

Reference

www.Wikipedia.com

http://scikit-learn.org/

http://stackoverflow.com/