Project-1: Predicting Boston Housing Prices – Udacity Machine Learning Nanodegree program

Boston Housing Prices prediction using Decision Tree Regression Model

Marimuthu Ananthavelu

Udacity- Machine Learning Nanodegree Program-Project 1.

The questions and answers are inserted within the code, are reported here.

Abstract

The following are the Questions and answers for the Project-1 for Boston Housing prediction which is a part of Machine Learning Nanodegree curriculum.

Keywords: Nanodegree, Machine learning, Udacity, Project-1, Boston, Housing prices

Boston Housing Prices prediction using Decision Tree Regression Model

##Question 1

Of the features available for each data point, choose three that you feel are significant and give a brief description for each of what they measure.

The following three Attributes, i believe to have a strong influence in predicting the Housing prices closely and thus I chose the three as following;

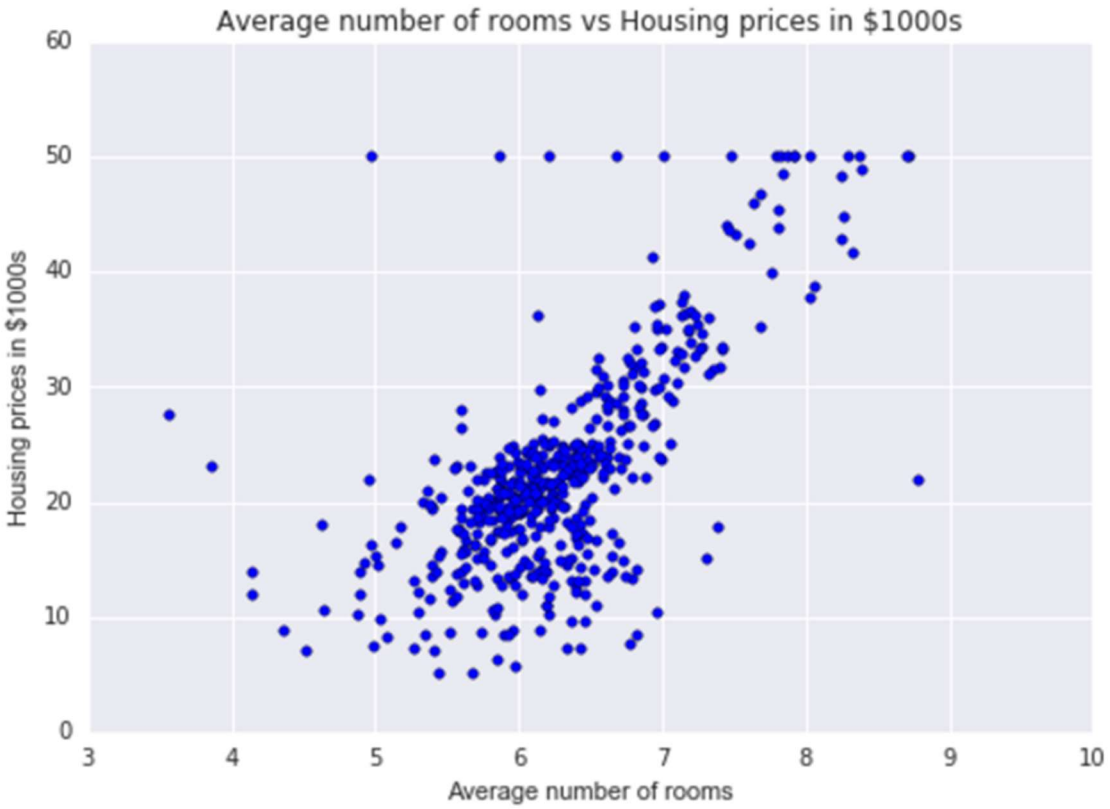
6 . RM: average number of rooms per dwelling

Intuition: One of the most important criteria in choosing the Home and its prices, depends upon the Number of Rooms it contains. I believe, total number of rooms most likely to give the following information which are closely associated with House pricing;

a) Average number of rooms (RM) make good approximation about the 'AREA' of House, which is a hidden information, but inferred approximately using RM-average number of rooms per dwelling.

b) This also gives indirectly an information about an average number of inhabitants in individuals home likely to be which in turn getting to know an average expenditure in buying a new home with respect to number of inhabitants (Though, not very compelling view)

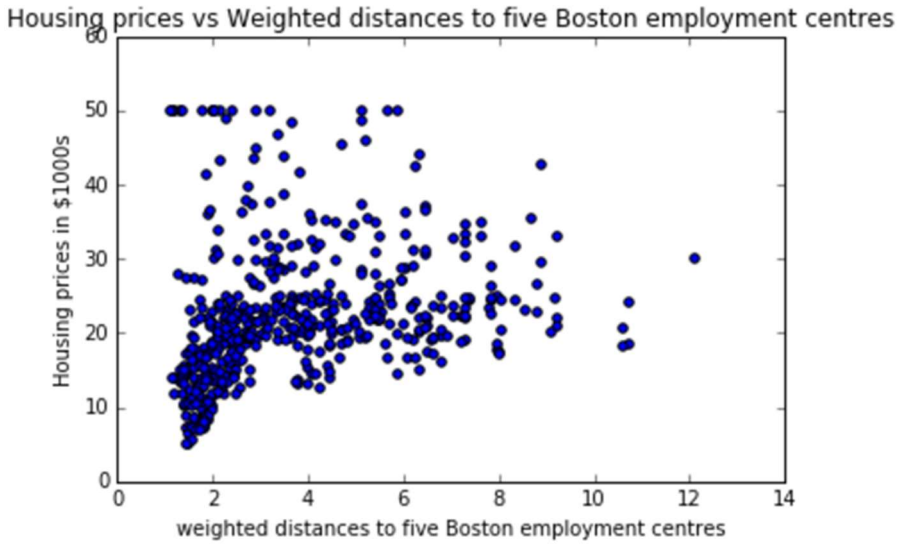
Statistical view: By trying to fit this feature 'RM' and 'Housing prices', there is a very much linear fit between the two (Feature and Target value.i.e. housing prices). This is in comparison to other available features in the dataset as we could see MOST of the data points of all the houses feature (in this case 'RM') goes linearly with the housing prices. So it is believed to be having the major influence in predicting the housing prices.



8. DIS: weighted distances to five Boston employment centres

Intuition: The House prices are most likely to be influenced when they are in place where there is a great accessibility to individual's workplace for daily commuting. Here i imagine a 'Demand' tag for the houses which has less distances from the 'Five Boston employment centres' which makes the Inhabitants ready to pay more for one of their comfort (Commuting for work).

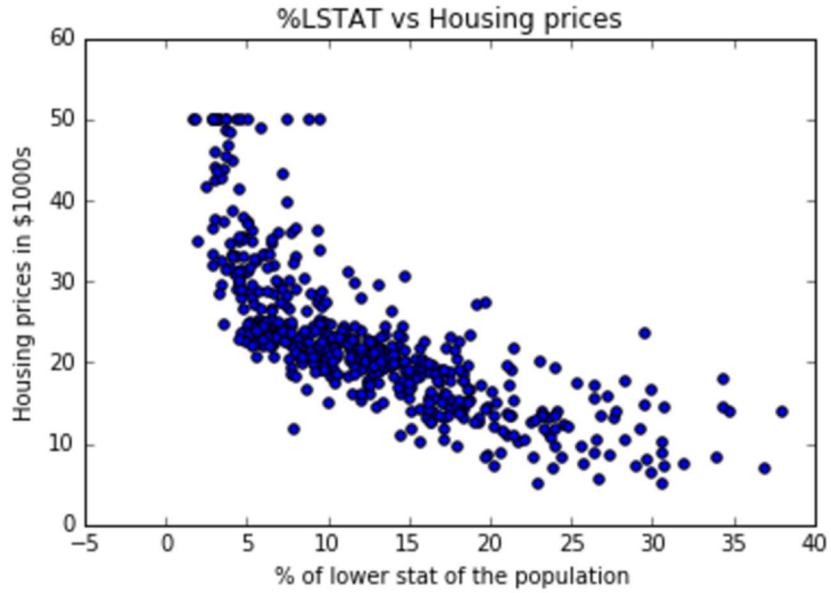
Statistical view: By trying to fit a line with linear/polynomial view, there is a better spread-out can be seen with respect to prices. Most of the data points can be drawn as second order polynomial along the line. Thus i felt that, this feature can play an important role in predicting the housing prices which is unknown.



9. %LSTAT: lower status of the population

Intuition: Impact is being very close to 'DIS: weighted distances to five Boston employment centres' and at the same time, the accessibility to highways give an edge in rise of prices due to its well connectivity.

Statistical view: By trying to fit in the similar way,i felt that the most of the data points (feature=%LSTAT) can be fit as a linear. So when doing that, there is an inference that this particular feature may play an important role as well in predicting the Housing prices.



##Question 2 ***Using your client's feature set*** `CLIENT_FEATURES`***, which values correspond with the features you've chosen above?***

RM :5.609

DIS : 1.385

LSTAT : 12.13

##Question 3 *Why do we split the data into training and testing subsets for our model? *

We need to split the data into Training and Test for the following main reasons;

1. To ensure that the 'model' has learned to predict the target values for unknown features over the different possible scenarios at all the times (Gives an estimate of performance on an independent datasets)

2. To ensure that the 'model' has not only to known to predict for given features but getting tested with different sets for the chosen problem, gives an advantage for another sets of features. (This helps in avoiding 'overfitting' while training the model.).

##Question 4*Which performance metric below did you find was most appropriate for predicting housing prices and analyzing the total error. Why? *

- *Accuracy*

- *Precision*

- *Recall*

- *F1 Score*

- *Mean Squared Error (MSE)*

- *Mean Absolute Error (MAE)*

The following performance metrics were considered for predicting housing prices and specific reasons are provided for individual metric saying why or why not the metric is considered.

- *Accuracy*

Accuracy is described as whether the number of samples are classified or labeled as 'correct' or 'not. Our problem is associated with 'Prediction of continuous value output' , here in this case as Housing price. There will be a specific number as an outcome in our Model. So this metric will not be applicable to see how well the model is performing. 'Accuracy' is seen good for classification problems of which, the outcome is discrete.i.e whether the model predicts 'accurately' or 'not'.

- *Precision* & *Recall*

There 2 performance metrics deal with the specific approximation and probability of choosing the labels for the given dataset. Represented as a 'Discrete' outcome which are pertaining to 'Classification' problems. Since, our model is dealing with predicting the specific number, which is here the most important thing is to see how well the predicted value is close to the data set (opposite to 'Discrete)

- *F1 Score*

F1 score is a weighted average of Precision and Recall, which is as below;

F1 = 2 * (precision * recall) / (precision + recall)

In this case of F1 score too, the error is expected to perform for 'Discrete' outcome. This is another way of to know the performance of model similar to 'Precision' and 'Recall'

- *Mean Absolute Error (MAE)*

Mean Absolute Error is one of the USABLE performance metric for our Problem-Housing prices prediction. It implies;

1. Calculating the difference between each data point to the mean as an absolute value.

[Making absolute is due to the fact that the deviation is same when the data point is above or below the mean]

This one was very closely considered to be used as a performance metric.

- *Mean Squared Error (MSE)*

The chosen performance metric is 'Mean Squared Error'. The following are the reasons for the same;

1. It gives an 'Error' in absolute values, i.e. Positive always due to 'Squaring' the differences between the 'y_true' and 'y_predict' values, which is easy to quantify easily.(As same as MSE)

By squaring the differences, it gives more weightage to higher deviations when calculating the Mean Squared Error, which impacts the overall performance of the model. Thus Making the 'Mean Squared Error (MSE) preferred over 'Mean Absolute Error'

2. Choosing the 'Mean Squared Error' provides a good estimate to compare with how different datasets are spread over from mean value (i.e. Standard deviation when dividing by total number of data points).

##Question 5*What is the grid search algorithm and when is it applicable? *

Grid search algorithm is an algorithm which is used to fine tune the parameters used in the predictive model.

Grid search algorithm is applicable when there is a need to fine tune the parameters to best fit the model without much work in comparison to other methods (Guess/Try methods).

It fits the data for different parameters using the scoring function taking note of given range of parameters for a particular method of prediction (DecisionTreeRegression in this case).

##Question 6 ***What is cross-validation, and how is it performed on a model? Why would cross-validation be helpful when using grid search?***

Cross validation is a method for estimating the predictive accuracy of the Model and a way to ensure that the model learns fair in all the sets of given data. This is done by splitting the dataset by training and testing data in successive combinations keeping testing data as different each time.

For example:

Step 1 : Splitting the total data set into 10 (most times uniform) sets. Let's say K=10

Step 2 : Consider 1 set as testing set with remaining (K-1) sets for training the model

Step 3 : Choose a different set now and keeping the remaining sets for training the model

Step 4 : Step 2 and 3 makes all the sets of data (different samples in the population) to train the model.

The above Cross validation procedure ensures the model is fair for any unknown new dataset.

Grid search algorithm is useful when the parameters are not learned within the estimators, so that it can be set by Grid search by looking into the given range of parameters.

##Question 7*Choose one of the learning curve graphs that are created above. What is the max depth for the chosen model? As the size of the training set increases, what happens to the training error? What happens to the testing error? *

The max depth for the chosen model is :1

As the size of the training set increases, i see the training error is increasing. (High Bias)

As the size of the training set increases, i see the testing error is decreasing.

##Question 8*Look at the learning curve graphs for the model with a max depth of 1 and a max depth of 10. When the model is using the full training set, does it suffer from high bias or high variance when the max depth is 1? What about when the max depth is 10? *

For the model with a max depth of 1:

When the model is using the full training set, the model is suffered by high Bias. This is because of the following reasons;

1. Increase in training set increases the Training error. The error for 'Training' and 'Testing' sets remain high at the end of training set where both the errors converge into. Thus the model is not performing well in both the Training and Testing sets.

2. Providing more data to the model does not help much in this case for better predictive model.

For the model with a max depth of 10:

When the model is using the full training set, it suffers by high Variance. This is because of the following reason;

1.The model exactly memorizes the 'training set' whereas virtually the 'Training set' error as NIL, but when it comes to 'Testing set' the testing error remain high. Thus it cannot predict and generalize well for an unseen new dataset.

Reference : http://scott.fortmann-roe.com/docs/BiasVariance.html

##Question 9*From the model complexity graph above, describe the training and testing errors as the max depth increases. Based on your interpretation of the graph, which max depth results in a model that best generalizes the dataset? Why?*

The Testing error decreases and at a certain point it starts to increase again whereas training error continue to decrease from the beginning.

The region where the testing error touches the minimum and starts to increase again, is the spot where the optimum performance of the model. Thus with 'Maximum depth' of 4, the model is likely to perform better.

Reference : http://scott.fortmann-roe.com/docs/docs/BiasVariance/biasvariance.png

***##Question 10*Using grid search on the entire dataset, what is the optimal `max_depth` parameter for your model? How does this result compare to your intial intuition? ***

The optimal 'max_depth' for the model using grid search algorithm is 4. I understand that the performance of the model is measured on the testing set.

The indication in the testing error to have an optimum model is, the testing error is decreases and at certain point ,it starts to increase when looking at the model complexity.

My initial intuition was expecting the value somewhere in the middle but not definitely on the start or end of defined parameters range.

When i study through the DecisionTreeRegressor, i will evaluate my thoughts as and often..

##Question 11*With your parameter-tuned model, what is the best selling price for your client's home? How does this selling price compare to the basic statistics you calculated on the dataset?

* **Answer: **

The best selling price for my client’s home shall be (in $1000s): 21.630.

This price is closer to our original dataset's mean and median housing price

Mean house price: 22.533,

Median house price: 21.2.

##Question 12 (Final Question):*In a few sentences, discuss whether you would use this model or not to predict the selling price of future clients' homes in the Greater Boston area.*

The following are the reasons to conclude that this model fairly predicts the Housing price for an unknown dataset;

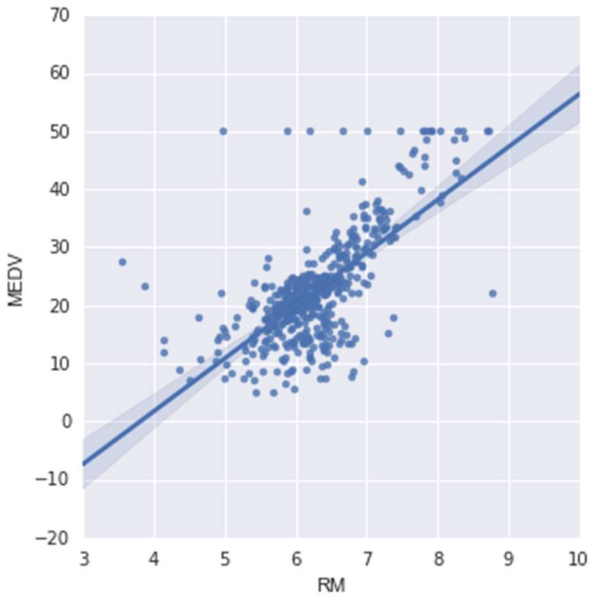
1.Looking at the three most important features listed above for clients features, i.e.

- 'RM' - average number of rooms per dwelling [Client feature :5.609]
- 'DIS' - weighted distances to five Boston employment centers Intuition [Client feature :1.385]
- '%LSTAT' - lower status of the population Intuition [Client feature :24]

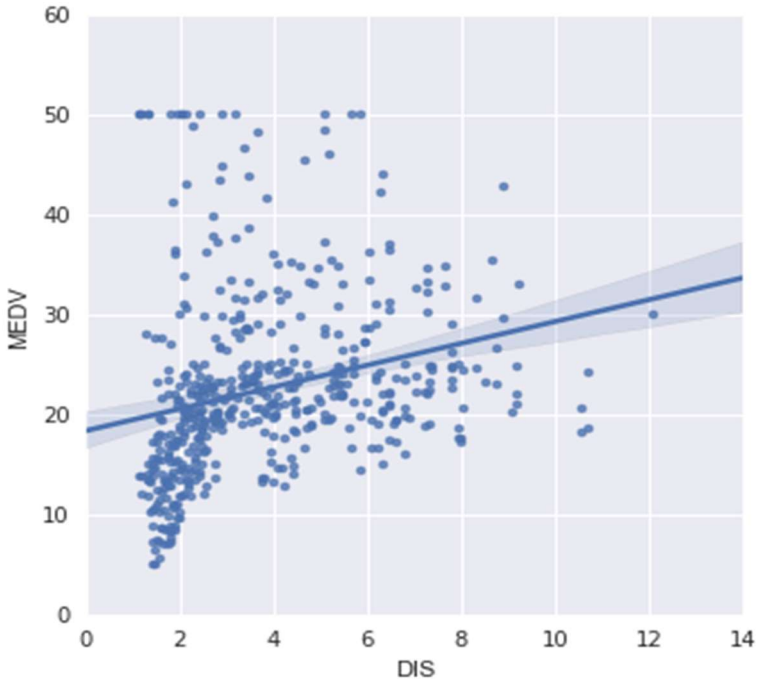
I tried to visualize the Housing price individually with respect to above 3 important features.It looks the data point in all the three plots fall very close to the predicted price by the model.

The single variable linear fit for the above features separately are as below;

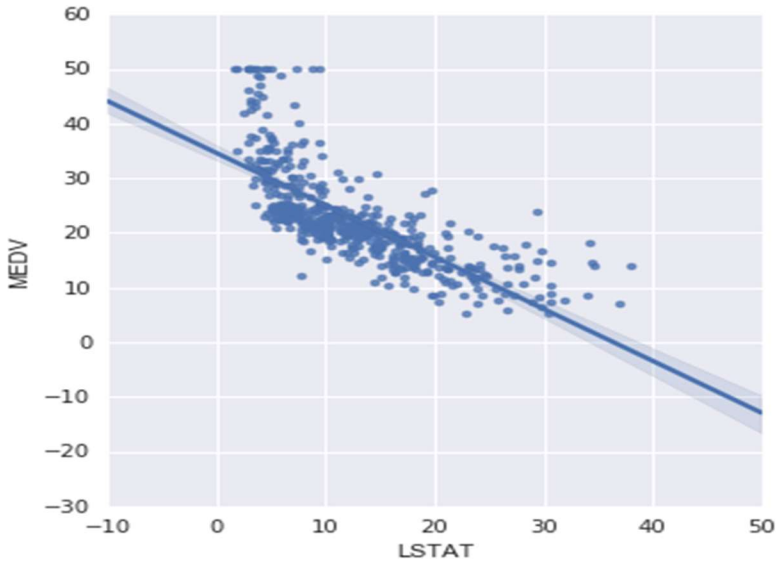
RM vs MEDV



DIS vs MEDV



LSTAT vs MEDV



2. The complexity curve indicated the close associate between Training and Testing Error with the chosen 'Max_depth' parameter.(‘Max_depth’ as 4)

Considering the above facts, it is fairly to believe that the chosen model will predict the good price for an unknown dataset in Boston. Here in this case, I would ask the client to sell @ (1000s) : $ 21.630