Boston Housing Data Analysis Report

Amirhossein Goudarzi

Executive Summary

Boston housing data included 14 sets of attributes each having 506 data. Main attribute which is the main point of our interest is the MEDV (median value of owner occupied homes in $1000). The main purpose of this paper is proposing the best model which predicts the movements of MEDV in different areas of Boston.

After analyzing the data, I realized that there are some attributes that need a higher attention such as the ones with higher correlation or a more similar whisker graph. Designing the model, I have used regression analysis to make predictions. I have avoided other tools as I was going to use only a supervised training which results in predictions and not categorization.

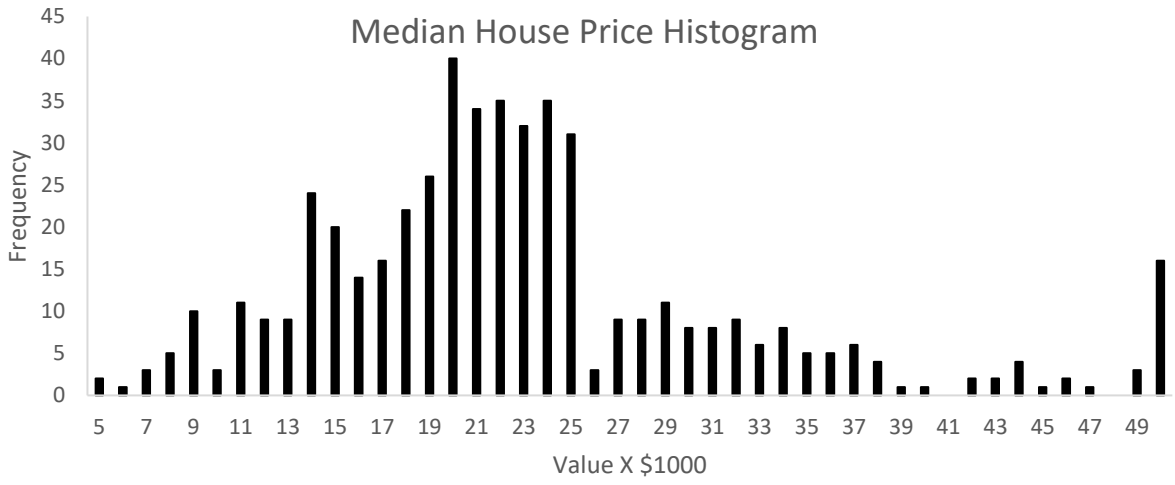
In this model I have used two performance analysis measures which has given me more room to analyze different outputs and choose the most optimum one. Finally I have applied a fit model using Grid Search to calculate the optimum depth for the decision tree regression analysis which was matching my prior analysis in the modelling section. I concluded that a decision tree regression with a depth of 5 or 6 while using a 70:30 sampling was the ideal combination for this dataset.

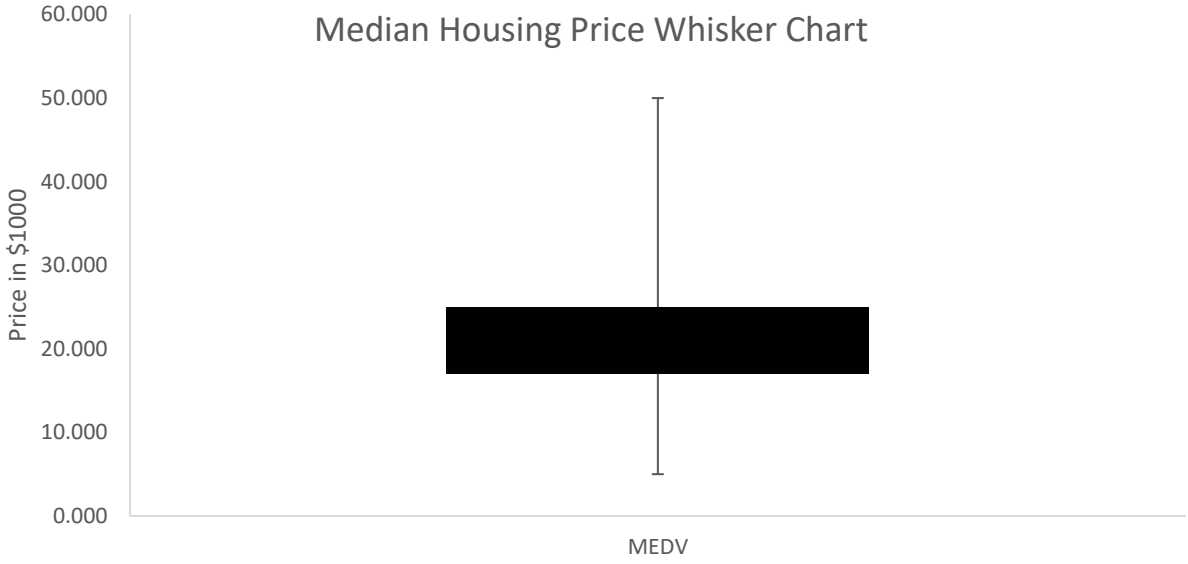
I believe this was a great practice to see the effect of regression model on such data set and feel the power of machine learning. However in the real world we need more realistic data in order to make an accurate prediction for the house prices.

Data Exploration

Boston housing data included 14 sets of attributes each having 506 data. Main attribute which is the main point of our interest is the MEDV (median value of owner occupied homes in $1000). The main purpose of this paper is proposing the best model which predicts the movements of MEDV in different areas of Boston.

Initially I analyzed the MEDV and then compared it with other 13 attributes to find the most useful attributes and analyze their effects on MEDV. As it is visible in the histogram below, MEDV has a median of $21,000 but it is scattered and there are large frequencies happening in higher values. MEDV has almost matching mean and median but it has a standard deviation of $9,000 which is comparably large. Whicker chart of the MEDV also shows a large deviation between the median, MIN and MAX values.





I have also compared the whicker chart of different attributes to find more about the behavior of each attribute regardless of the MEDV[1]. Among all of the attributes, TAX has one of the widest distribution while B, CRIM, ZN, AGE, DIS and LSTAT show a much deviated result from the median. A correlation factor can show a relation between these irregularities and the changes in MEDV.

Correlation chart shows a high correlation with MEDV in almost all of the attributes except DIS and CHAS which have lower correlation comparing to other attributes. This means MEDV is less correlated with being located near the Charles River comparing to other factors. Moreover, when checking near zero variance to find indifferent attributes to the changes of MEDV, we find that all of the attributes are non-zero variance therefore we cannot eliminate them through this process.

Also correlation factors between other attributes[2] shows that while INDUS has a low correlation with MEDV but there is a high correlation between INDUS and other attributes such as NOX, AGE, DIS, TAX and LSTAT. This shows that some attributes may indirectly affect MEDV while they don't show up in the correlation analysis. AGE and DIS, RAD and TAX are also highly correlated.

Attribute	Correlation
RM	0.695
ZN	0.36
B	0.333
DIS	0.25
CHAS	0.175
AGE	-0.377
RAD	-0.382
CRIM	-0.388
NOX	-0.427
TAX	-0.469
INDUS	-0.484
PTRATIO	-0.508
LSTAT	-0.738

Running a simple regression analysis[3], we can see that there is a high T value for the CRIM, ZEN, INDUS, CHAS, AGE, TAX and B while the P value of INDUS and AGE are considerably high. These values and relations will be effected in the modeling stage and it will help creating a more relevant model with higher R square.

[1] Appendix I,II,III
[2] Appendix IV
[3] Appendix V

Modelling & Performance

Boston housing problem is a regression problem as we do not have to make categories in a supervised or unsupervised manners. Therefore using a regression technique or mixing it with machine learning can give us the required prediction model. Moreover, most of the data is continuous numerical data so it is the best to choose regression for solution.

In this problem I have used (*train_test_split and ShuffleSplit from sklearn.cross_validation)* to split the data into test and training sets. Initially I started with 80:20 ratio but after running several samples I realized 70:30 had better results so I used this ratio as the main reference.

Initially I have chosen a simple linear regression model (*exploredata.py*) to explore the data and find the R^2 and other measurements[4]. However, linear regression is too simple and this is a prediction problem in which regression trees do the best. The decision trees is used to fit a sine curve with addition noisy observation. As a result, it learns local linear regressions approximating the sine curve. So I decided to choose Decision Tree Regression in order to capture the best of this prediction.

I had to choose between a boosted tree training such as Ada Boost or a bootstrap aggregated decision tree which runs on an ensemble learning. Because of the short training time of the Decision Tree I chose to apply this framework from Scikit. When using decision tree, we can see that if the maximum depth of the tree (*controlled by the max_depth parameter*) is set too high, the decision trees learn too fine details of the training data and learn from the noise, and become overfit.

In order to control the performance and find the best depth for this model I have used two different performance analysis techniques and compared them to find the optimum training depth and avoid overfitting. First I have used Mean Squared Error (*MSE*)[5]. MSE (*mean_squared_error from sklearn.metrics)* measures the standard deviation of the residuals. This is a measure of the noise on the system so a lower one is better. Then I have used R Squared measurements[6]. R square measures (*r2_score from sklearn.metrics)* the proportion of variability in Y explained by the regression model. It is the square of correlation r. WE prefer higher R Square because we want more variability.

Initially in the MSE measurements[7] we can see the training error is very high for the first four depth analysis, but it starts to be smooth and similar from the 5th depth analysis. From the depth 5 analysis, we can confirm that trainings which have more than 250 training size have similar result. Therefore we do not need a deeper training and we can avoid further overfitting. Also by looking at the R Squared measurements we can see a sharp negative test results for the first 5 depth analysis. As we confirmed in the MSE, R square is also stable from the 5th and 6th depth analysis. Therefore I recommend a depth analysis of 5 or 6 in order to achieve the best fitting results.

[4] Appendix V
[5] Appendix VI
[6] Appendix VII
[7] Appendix VI

Performance Evaluation

In order to confirm the performance of the model, I have applied two different performance measurement systems and chose the most optimum depth for the model. Moreover I have used a final fit model calculating the most optimum depth using the scoring and grid search libraries from Scikit.

This data set has 14 attributes which don't equally effect the model. When using Machine learning models we usually have some parameters *(hyperparameters)* that won't be learned by the model and, therefore, a human on the backend must be tuning and editing them to adjust the model and improve the performance and the precision of the model. In order to reduce this time consuming process and also make a more precise model we can use computer automated systems which are based on a grid by applying the human modification and taking advantage of the fast analysis of the computer.

One of the most basic solutions are using exhaustive search in all parameters to find the optimum combination. However this solution has a problem of running for long times as it has to analyze every point in the grid. Since we do not have many in this problem, I have used this to receive a recommendation on the most optimum combination for the system.

According to the fit model, the most optimum depth is in the range of 5 to 6 which is consistent with my analysis in the previous section. However sometimes I received a value of 7 which is due to the random sampling process in the cross validation. Therefore, we can trust this result and apply a depth of 5 or 6 to our sample.

Conclusion

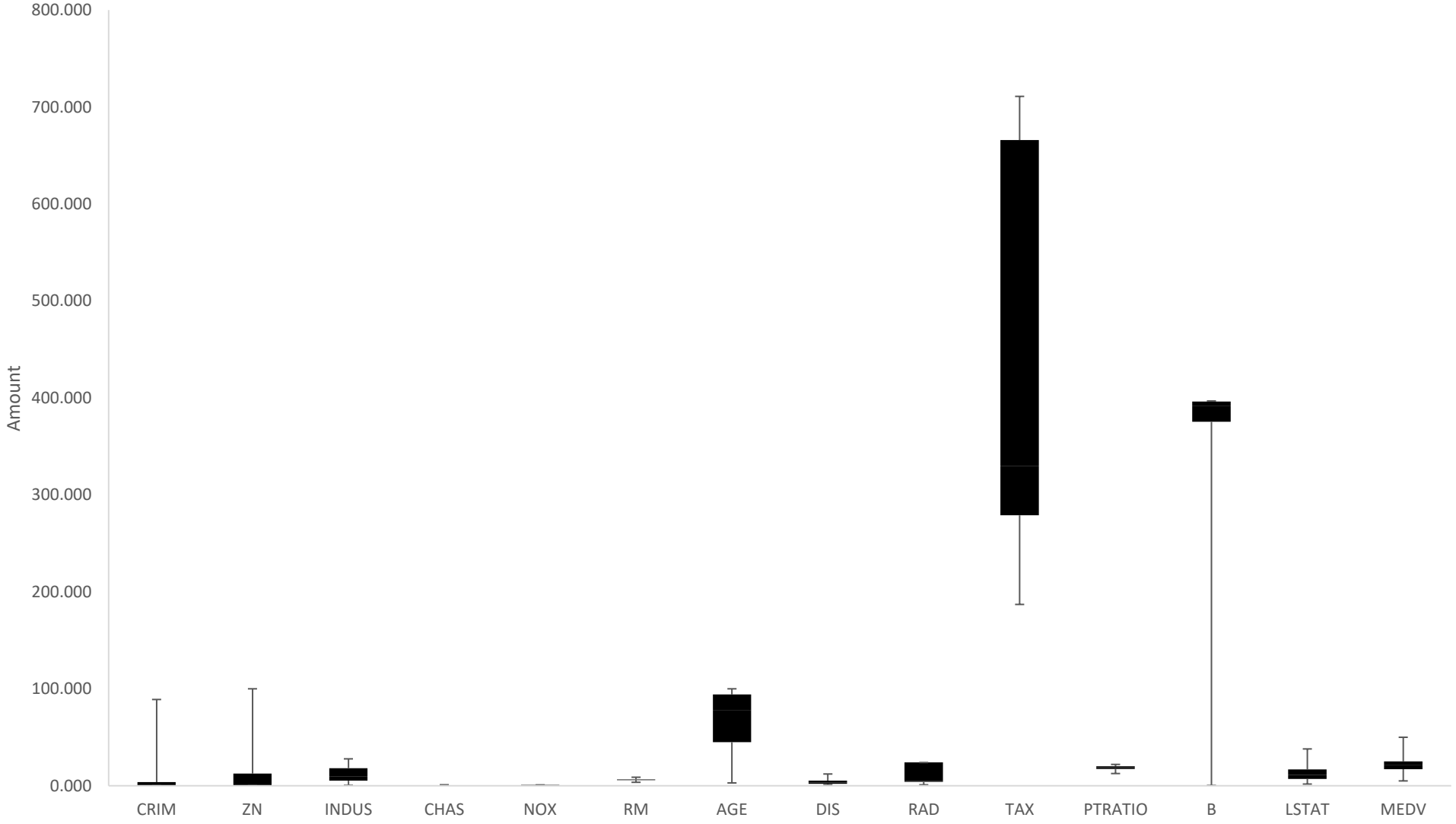
There are many limitations on this dataset which makes it almost impossible to be used in the real world. Such factors include the crime rate in neighborhood, age of the house and many other factors that can be important in one city but not in another one. Also in a real world, the way someone makes a decision to buy a house may be completely different and not depending only on these attributes.

Also the size of our data is only 506 which is very limiting and it is a very old data which can't be referred to any more. In order to make a better data set we need more information about the consumer behavior in real estate industry in each city and also decision making ranking on how people choose their houses. Finally man of these factors change by time so this can affect the model and we should exclude them from the training system in the proper time.

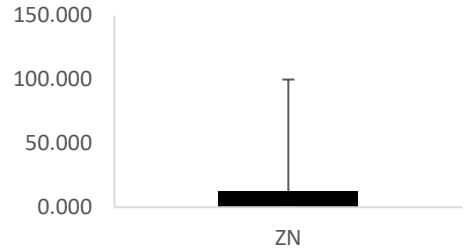
Finally this was a great practice to learn more about the regression analysis, and I hope to have a chance to solve more of these problems in the real world.

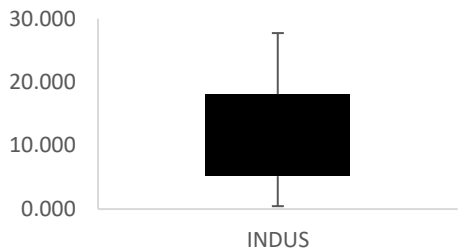
Appendices

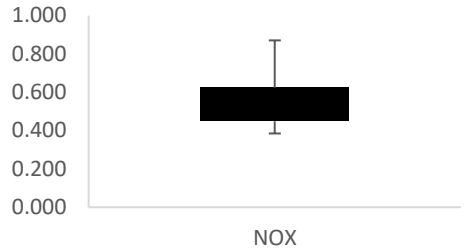
Appendix I <Whisker Chart Comparison All In One>

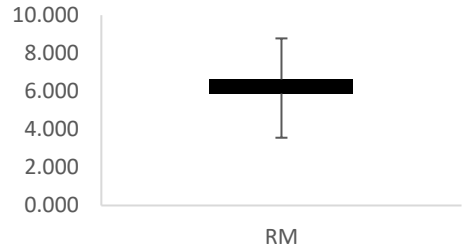


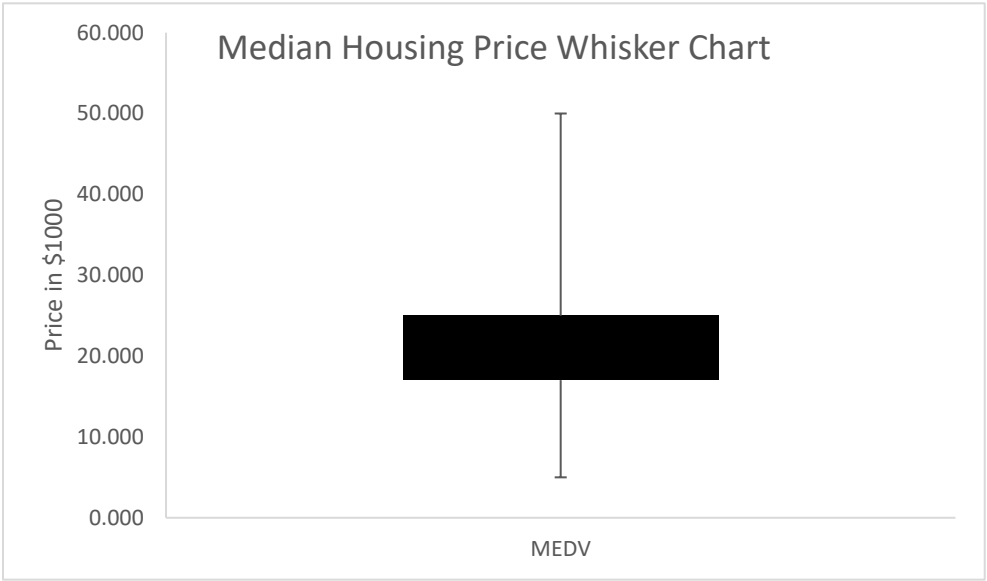
Appendix II <Whisker Chart Comparison of all Attributes>

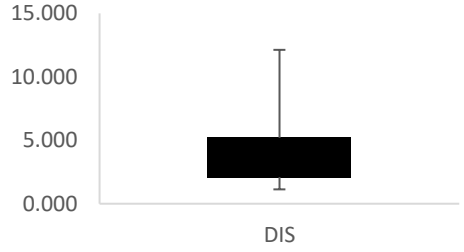


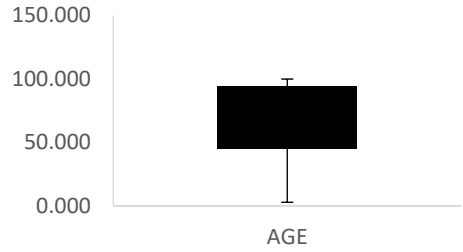


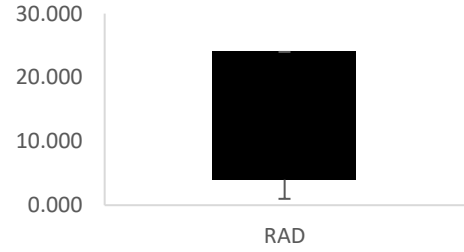


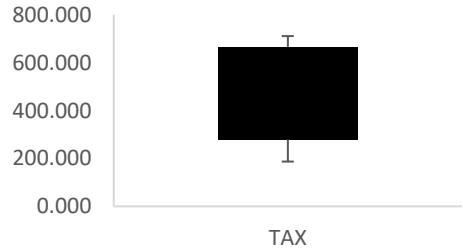


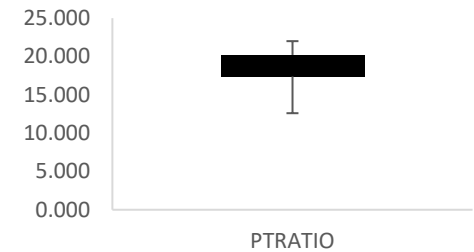


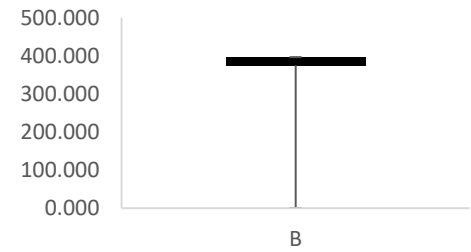


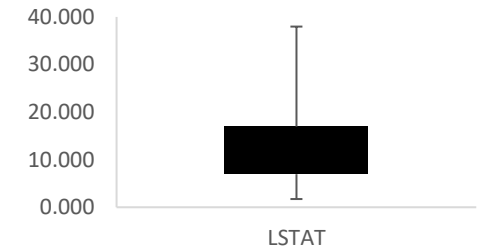












Appendix III <Attributes Comparison Data>

	CRIM	ZN	INDUS	CHAS[8]	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
MIN	0.006	0.000	0.460	0	0.385	3.561	2.900	1.130	1.000	187.000	12.600	0.320	1.730	5.000
First Quartile	0.082	0.000	5.190	0	0.449	5.886	45.025	2.100	4.000	279.000	17.400	375.378	6.950	17.025
Median	0.257	0.000	9.690	0	0.538	6.209	77.500	3.207	5.000	330.000	19.050	391.440	11.360	21.200
Third Quartile	3.677	12.500	18.100	0	0.624	6.624	94.075	5.188	24.000	666.000	20.200	396.225	16.955	25.000
Max	88.976	100.000	27.740	1	0.871	8.780	100.000	12.127	24.000	711.000	22.000	396.900	37.970	50.000
STD	8.602	23.322	6.860	0.254	0.116	0.703	28.149	2.106	8.707	168.537	2.165	91.295	7.141	9.197
Average	3.614	11.364	11.137	0.069	0.555	6.285	68.575	3.795	9.549	408.237	18.456	356.674	12.653	22.533

Appendix IV <Correlation Factor between Attributes>

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
CRIM	1.000													
ZN	-0.200	1.000												
INDUS	0.407	-0.534	1.000											
CHAS	-0.056	-0.043	0.063	1.000										
NOX	0.421	-0.517	0.764	0.091	1.000									
RM	-0.219	0.312	-0.392	0.091	-0.302	1.000								
AGE	0.353	-0.570	0.645	0.087	0.731	-0.240	1.000							
DIS	-0.380	0.664	-0.708	-0.099	-0.769	0.205	-0.748	1.000						
RAD	0.626	-0.312	0.595	-0.007	0.611	-0.210	0.456	-0.495	1.000					
TAX	0.583	-0.315	0.721	-0.036	0.668	-0.292	0.506	-0.534	0.910	1.000				
PTRATIO	0.290	-0.392	0.383	-0.122	0.189	-0.356	0.262	-0.232	0.465	0.461	1.000			
B	-0.385	0.176	-0.357	0.049	-0.380	0.128	-0.274	0.292	-0.444	-0.442	-0.177	1.000		
LSTAT	0.456	-0.413	0.604	-0.054	0.591	-0.614	0.602	-0.497	0.489	0.544	0.374	-0.366	1.000	
MEDV	-0.388	0.360	-0.484	0.175	-0.427	0.695	-0.377	0.250	-0.382	-0.469	-0.508	0.333	-0.738	1.000

[8] CHAS has a binary value, therefore it was not suitable for a histogram demonstration. But its effects will be reviewed in the modelling part.

Appendix V <Regression Analysis of All Attributes>

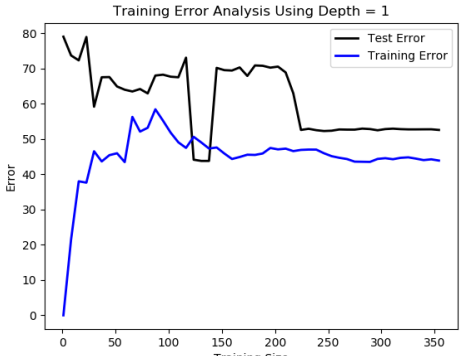
Regression Statistics	
Multiple R	0.860605987
R Square	0.740642664
Adjusted R Square	0.733789726
Standard Error	4.745298182
Observations	506

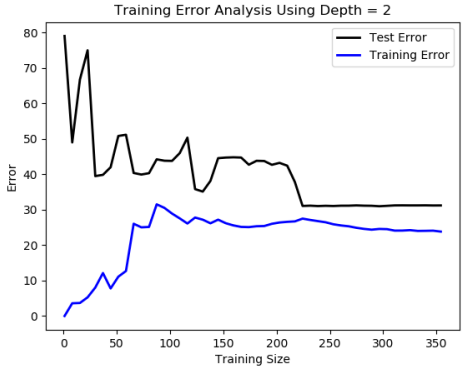
ANOVA

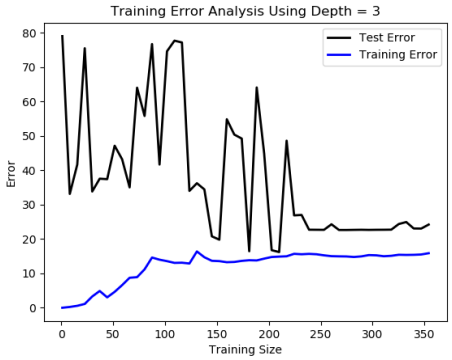
	df	*SS*	*MS*	*F*	*Significance F*
Regression	13	31637.51084	2433.65468	108.0766662	6.7222E-135
Residual	492	11078.78458	22.51785483		
Total	505	42716.29542			

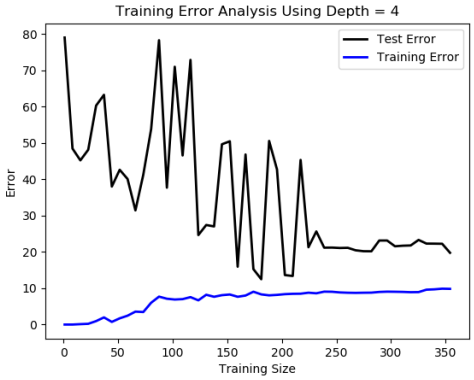
	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
Intercept	36.45948839	5.103458811	7.144074193	3.28344E-12	26.43222601	46.48675076	26.43222601	46.48675076
CRIM	-0.108011358	0.032864994	-3.286516871	0.00108681	-0.172584412	-0.043438304	-0.172584412	-0.043438304
ZN	0.046420458	0.013727462	3.381576282	0.00077811	0.019448778	0.073392139	0.019448778	0.073392139
INDUS	0.020558626	0.061495689	0.334310042	0.738288071	-0.100267941	0.141385193	-0.100267941	0.141385193
CHAS	2.686733819	0.861579756	3.118380858	0.00192503	0.993904193	4.379563446	0.993904193	4.379563446
NOX	-17.76661123	3.819743707	-4.651257411	4.24564E-06	-25.27163356	-10.26158889	-25.27163356	-10.26158889
RM	3.809865207	0.417925254	9.1161402	1.97944E-18	2.988726773	4.63100364	2.988726773	4.63100364
AGE	0.000692225	0.013209782	0.052402427	0.958229309	-0.02526232	0.026646769	-0.02526232	0.026646769
DIS	-1.475566846	0.199454735	-7.398003603	6.01349E-13	-1.867454981	-1.08367871	-1.867454981	-1.08367871
RAD	0.306049479	0.06634644	4.612899768	5.07053E-06	0.175692169	0.436406789	0.175692169	0.436406789
TAX	-0.012334594	0.003760536	-3.28000914	0.001111637	-0.019723286	-0.004945902	-0.019723286	-0.004945902
PTRATIO	-0.952747232	0.130826756	-7.282510564	1.30884E-12	-1.209795296	-0.695699168	-1.209795296	-0.695699168
B	0.009311683	0.002685965	3.466792558	0.000572859	0.004034306	0.01458906	0.004034306	0.01458906
LSTAT	-0.524758378	0.050715278	-10.3471458	7.77691E-23	-0.624403622	-0.425113133	-0.624403622	-0.425113133

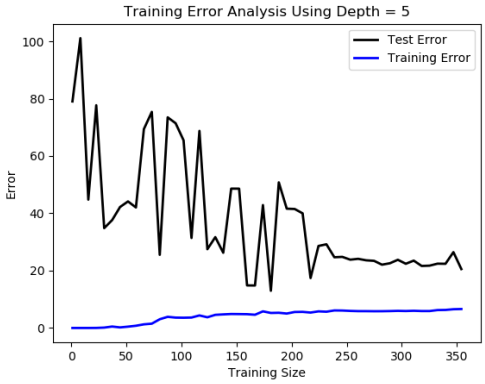
Appendix VI <Decision Tree Regression Analysis Using Mean Squad Error>

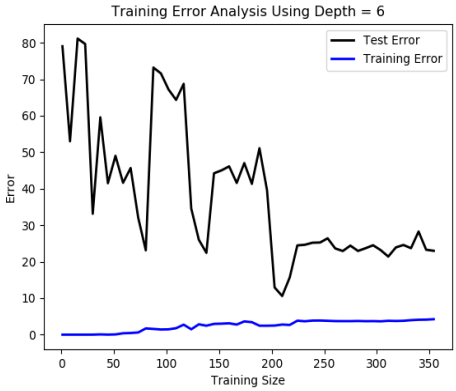


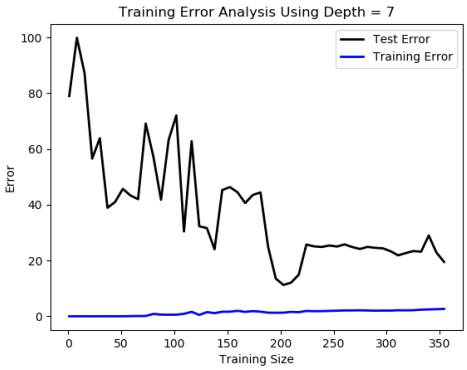


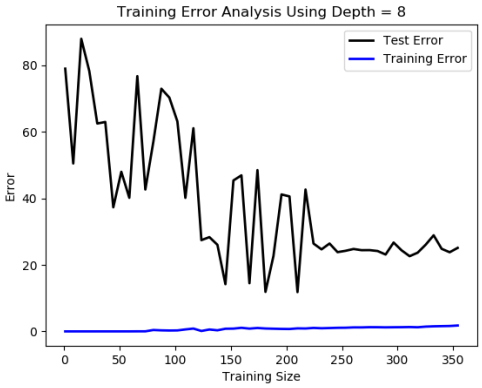


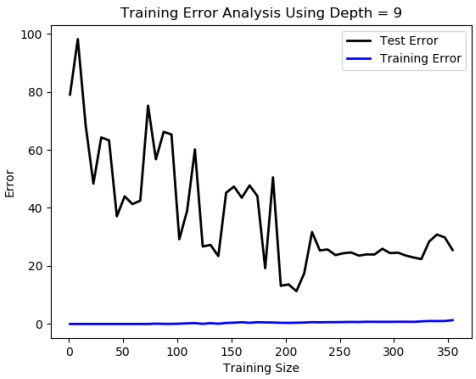


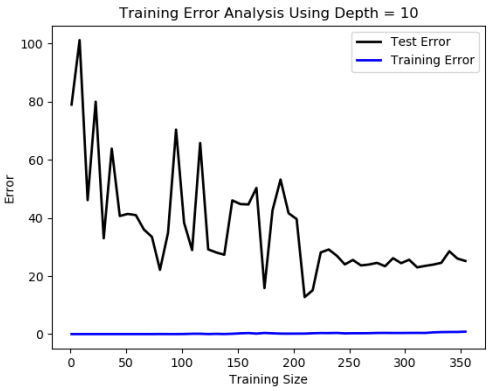




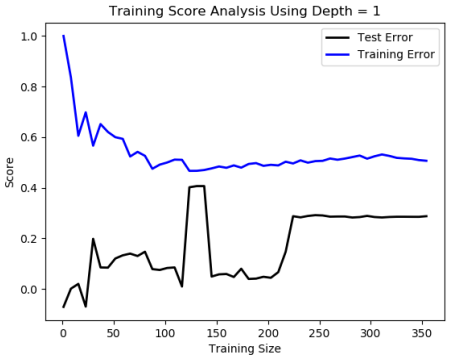


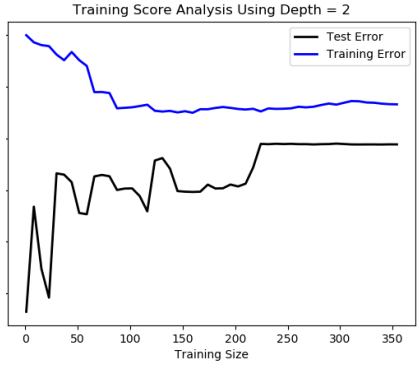


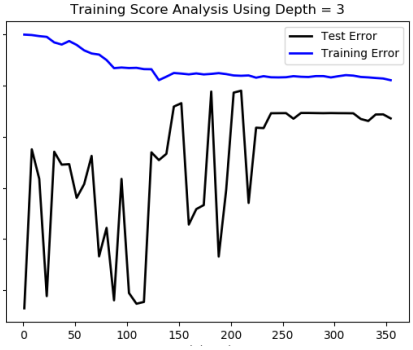


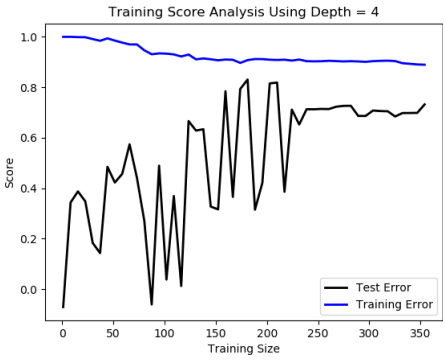


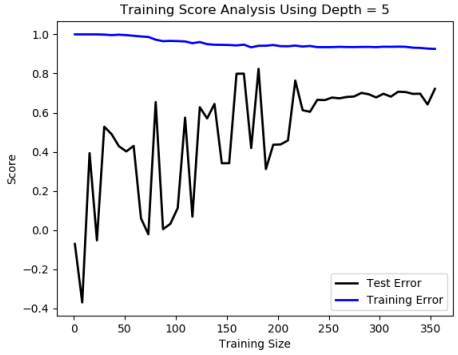
Appendix VII <Decision Tree Regression Analysis Using R Square>

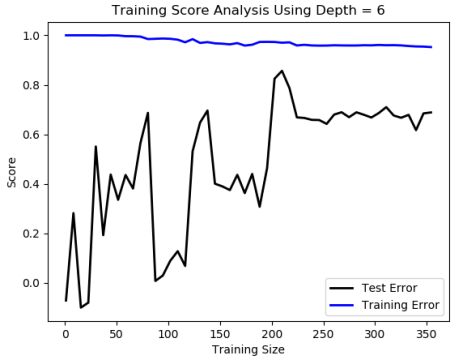


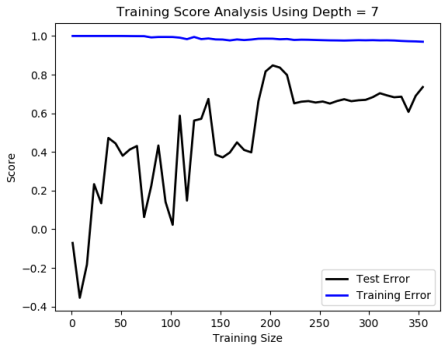


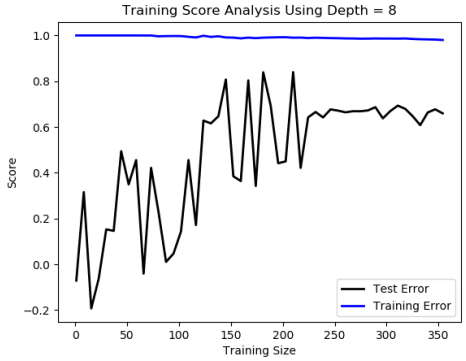


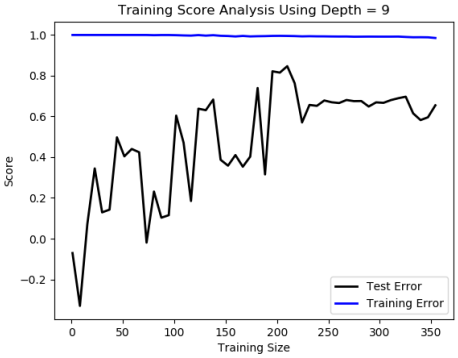


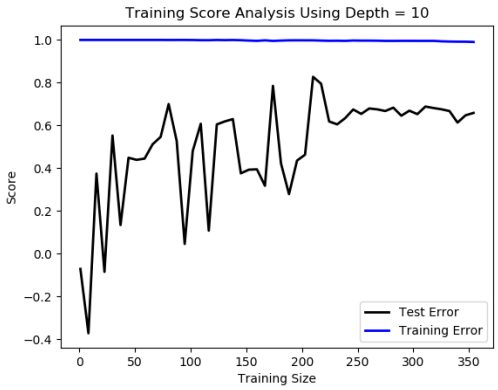












References

1. Decision Tree Repressor
   http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html Hjjgi
2. Near Zero Variance
   https://www.r-bloggers.com/near-zero-variance-predictors-should-we-remove-them/
3. GridSearch
   http://scikit-learn.org/stable/modules/grid_search.html
4. Boosted Decision Tree
   http://scikit-learn.org/stable/auto_examples/ensemble/plot_adaboost_regression.html