# 239AS - Special Topics in Signals and Systems
# Project 1

Mansee Jadhav - 204567818
Mauli Shah - 004567942
Ronak Sumbaly - 604591897

January 31, 2016

## INTRODUCTION

In this project we have done data analysis on two data sets - **Network Backup dataset and Boston Housing dataset**. The datasets are analyzed by using the data mining approach of regression. Regression is an approach for modeling the relationship between a scalar dependent variable and one or more explanatory variables / features. We have studied the concepts of cross-validation, regularization, random forest and neural network regression in order to predict a dependent variable present in the datasets.

The Network-Backup data set has captured simulated traffic data on a backup system and contains information of the size of the data moved to the destination as well as the time it took for backup. Our task was to predict the *backup size* of the traffic depending on the file-name, day/time of backup. Prediction models have been created using Linear, Random Forest, Neural Network and Polynomial Regression.

The Boston-Housing data set that contains the housing values of suburbs. We have employed Linear and Polynomial Regression along with other algorithms to overcome overfitting to create a predictive model that is used to *estimate the value of owner-occupied homes.*

# NETWORK BACKUP

For details about the dataset look at Appendix A: Network Backup Dataset.

Since the model comprised of categorical attribute one-hot encoding and enumeration techniques were applied to change the attributes to numeric datatype for the purposes of Regression model creation.

## Ques 1. Types of Relationship in the Dataset

In order to understand the relationships in the Network-Backup Dataset, for each workflow the actual copy sizes of all files on a time period of 20 days was plotted and analyzed.

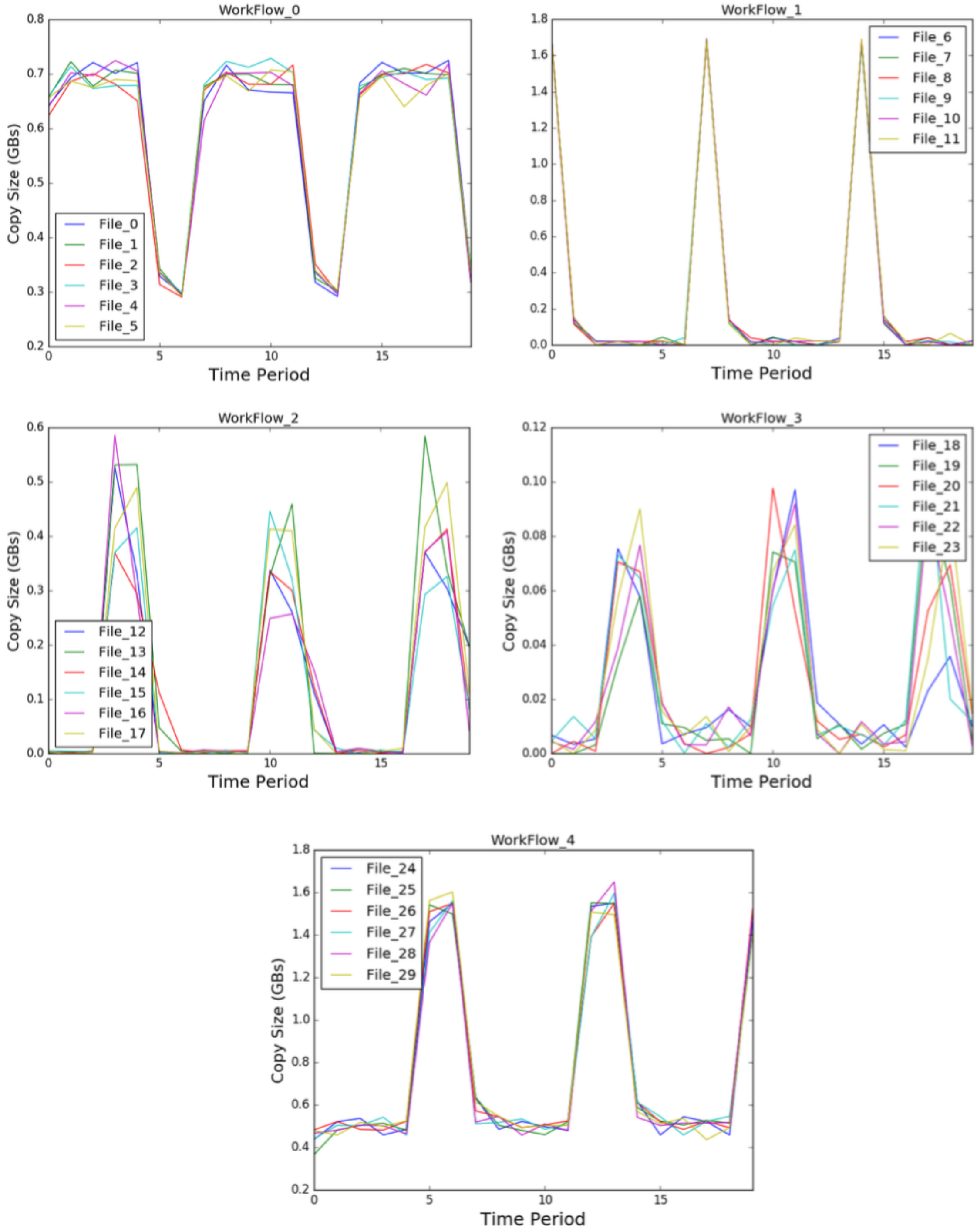


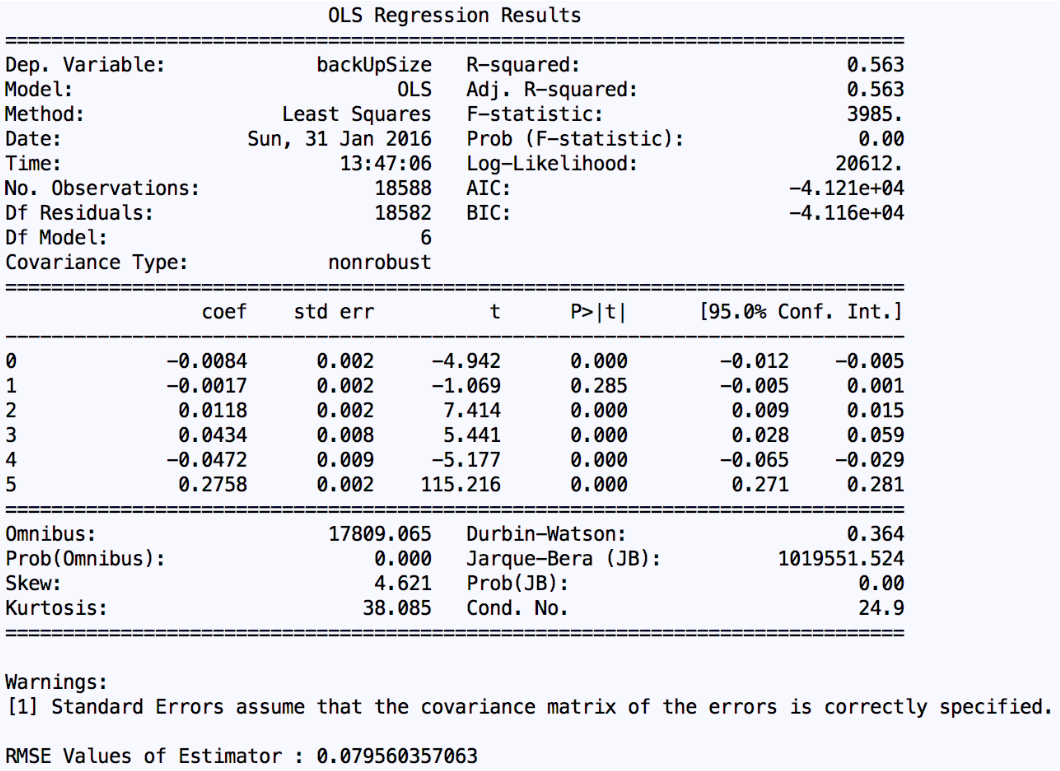
Figure 1: Copy Size vs Time Period for each WorkFlow

On careful analysis of the graphs for each workflow, the following observations/relationships were made with respect to the back-up size of files.

1. **WorkFlow_0** : A dip in copy-sizes of files can be seen during the weekends (non-office days) indicating Files (0 to 5) are mostly used to log content during the weekdays (office days).
2. **WorkFlow_1** : The back up data copy size reaches a peak (maximum) only on Mondays, and then drops down to 0 on rest of the days. This indicates Files (6 to 11) are only worked upon on Mondays hence there is a rise in back-up size and then cleared for the rest of the week.
3. **WorkFlow_2** : The copy size for Files (12 to 17) is shown to vary only during the time period Wednesday to Friday i.e the copy size is high on these days, while for the rest of days the copy size is relatively very low indicating no contents is stored in the files during those days.
4. **WorkFlow_3** : There is uneven distribution of copy size of data for Files (18 to 23) from Monday to Wednesday after which we can see the rise in copy size upto Friday.
5. **WorkFlow_4** : Data back-up for Files (24 to 29) is constant from Monday to Thursday indicating that the files contains headers that are required for operation every-time a task is performed on these files, and then from Thursday to Sunday the copy size shoots up showing that the files are worked on during the week-end and refreshed during start of every week.

## Ques 2a. Linear Regression

In order to predict the copy size of a file, given all the other attributes, a Linear Regression model was built and tested using 10 folds cross validation.*OLS library for Pandas* and *Linear_Models from Scikit-Learn* Libraries were used in order to create the model. The results obtained can be visualized in the figure below.



```
                            OLS Regression Results
==============================================================================
Dep. Variable:             backUpSize   R-squared:                       0.563
Model:                            OLS   Adj. R-squared:                  0.563
Method:                 Least Squares   F-statistic:                     3985.
Date:                Sun, 31 Jan 2016   Prob (F-statistic):               0.00
Time:                        13:47:06   Log-Likelihood:                 20612.
No. Observations:               18588   AIC:                        -4.121e+04
Df Residuals:                   18582   BIC:                        -4.116e+04
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
0             -0.0084      0.002     -4.942      0.000        -0.012    -0.005
1             -0.0017      0.002     -1.069      0.285        -0.005     0.001
2              0.0118      0.002      7.414      0.000         0.009     0.015
3              0.0434      0.008      5.441      0.000         0.028     0.059
4             -0.0472      0.009     -5.177      0.000        -0.065    -0.029
5              0.2758      0.002    115.216      0.000         0.271     0.281
==============================================================================
Omnibus:                    17809.065   Durbin-Watson:                   0.364
Prob(Omnibus):                  0.000   Jarque-Bera (JB):          1019551.524
Skew:                           4.621   Prob(JB):                         0.00
Kurtosis:                      38.085   Cond. No.                         24.9
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

RMSE Values of Estimator : 0.079560357063
```

Figure 2: Linear Regression Results - Network Dataset

The **RMSE** obtained after 10-fold Cross-Validation was **0.07956**

*Note.* For details about the statistics used refer Appendix B. Regression Statistics

In the output above, we can see that,

1. The predictor variables of [0, 2, 3, 4, 5] i.e [weekIndex, backupStartTime, workFlow, fileName, backUpTime] are **significant** because their p-values are 0.000.

2. However, the p-value for [1] i.e weekDay (0.285) is greater than the common alpha level of 0.05, which indicates that it is **not statistically significant**.

**Fitted vs. Actual Values**

The graph below shows the **Fitted Values** vs. **Actual Values** of Copy Size of the network back-up for the Linear Regression model. As seen from this plot, most of the values predicted have minimum deviation/error from the actual values. We can also notice some outliers in the scatter plot which has a high deviation from the model predicted by the system.

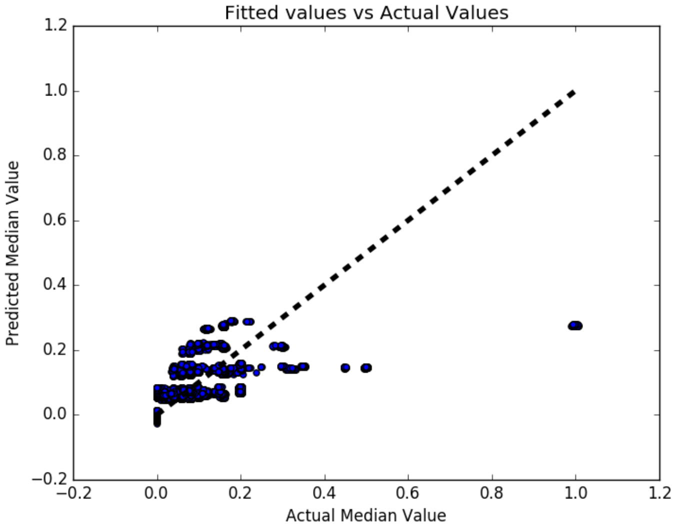


Figure 3: Linear Regression - Fitted Values vs. Actual Values of Copy Size

**Residuals vs. Actual Values**

The graph below shows the **Residual Values (Actual Values-Fitted Values)** vs. **Actual Values** of Copy Size of the network back-up for the Linear Regression model. . As seen from this graph, the residuals error is minimum and concentrated on the zero boundary indicating a good fit. Some outliers can be seen which have a high residual value.

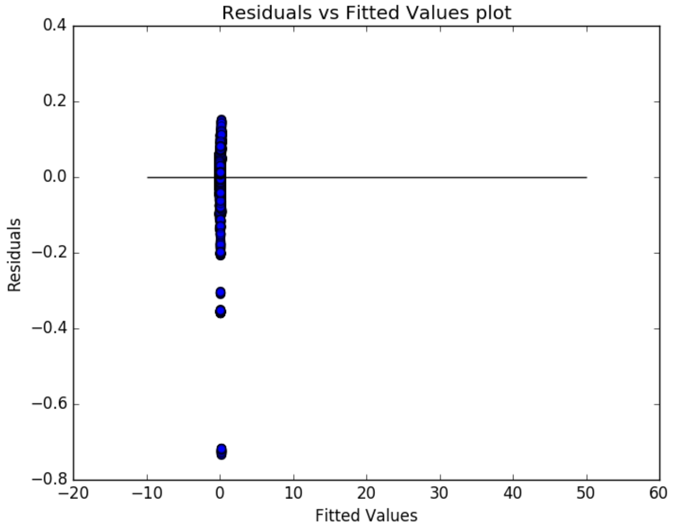


Figure 4: Linear Regression - Fitted Values vs. Actual Values of Copy Size

Based on the above two graphs we can conclude that the model constructed does not handle outliers properly and is not a good fit for prediction of backup sizes.

## Ques 2b. Random Forest Regression

Random Forest operates by constructing a multitude of decision trees of certain depth and certain number at training time and outputting the model. It depends on the notion of - each classifier, individually, is a weak learner, while all the classifiers taken together are a strong learner. Also, since the trees are generated randomly the results may vary with each execution.

The model was created using the scikit-learn library. In order to get the best of the model, parameter tuning was performed on **Maximum Number of Trees in the model and Maximum Depth of each Tree**. Results are presented below.

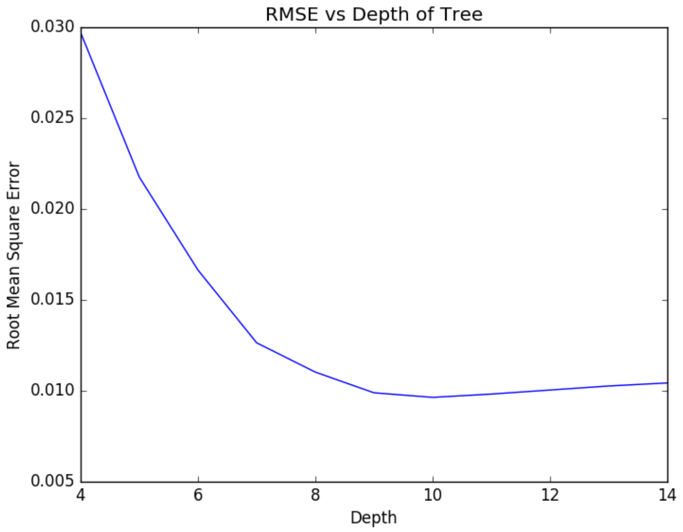


Figure 5: Random Forest - RMSE vs Maximum Depth of Tree

Parameter tuning was done to get the best depth of tree in the model. A fixed number of trees (20 trees) was taken and the model was created for multiple values of depth. As seen from the graph, at certain depth of 10, the RMSE is minimum and then remains sort-of constant from there on.

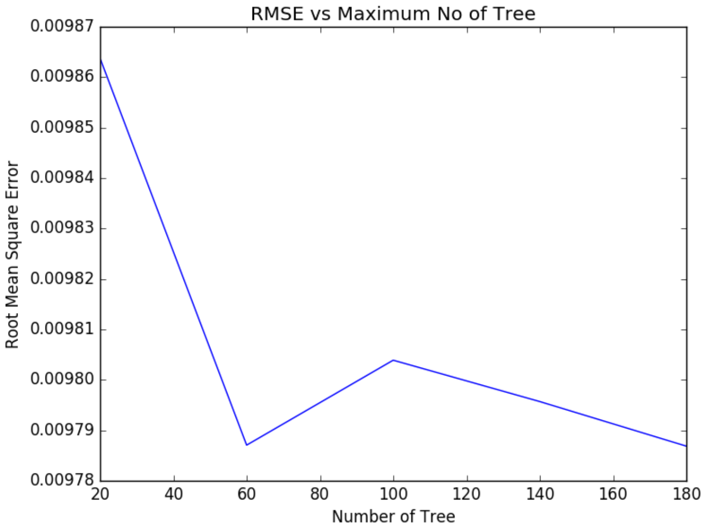


Figure 6: Random Forest - RMSE vs Maximum Number of Tree

Taking **best depth to be 10** we performed parameter tuning for **maximum number of tree**s and as seen from the above graph the best value obtained was **60**. Taking these values, we performed 10 folds cross-validation on the model and to get **RMSE = 0.009784**

**Fitted vs. Actual Values**

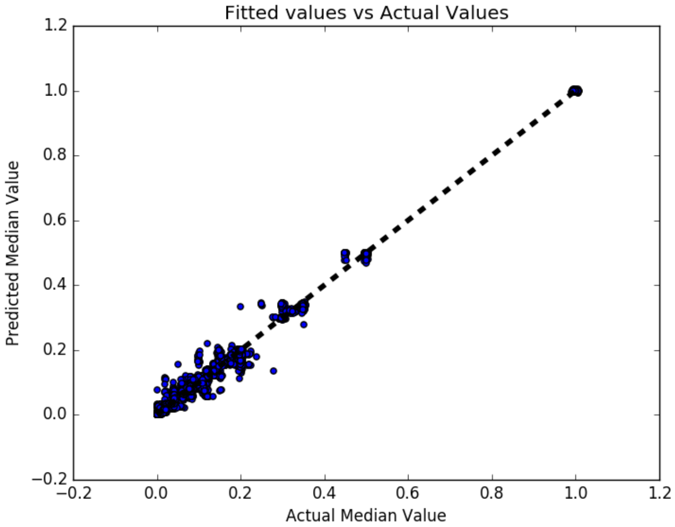


Figure 7: Random Forest Regression - Fitted Values vs. Actual Values of Copy Size

**Residuals vs. Actual Values**

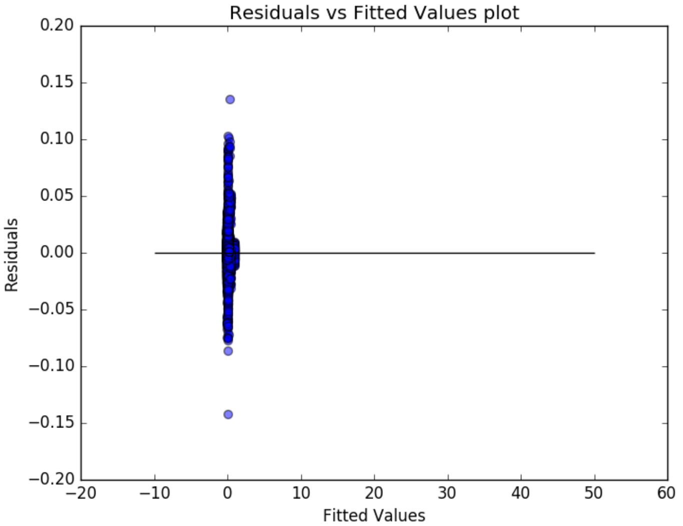


Figure 8: Random Forest Regression - Fitted Values vs. Actual Values of Copy Size

**Relationships between Linear and Random Forest Regression**

The RMSE value obtained from **Random Forest** = **0.009784** and that of **Linear Regression** = **0.07956**. Based on just the statistics of the Regression models we can say that Random Forest provides a better model of prediction than that of Linear Regression. However it must be noted that in Random Forest Regression, the start node is chosen at random, hence we may observe certain uncertainties in RMSE values.

With a large number of predictors, the eligible predictor set will be quite different from node to node. The greater the inter-tree correlation, the greater the random forest error rate, so one pressure on the model is to have the trees as uncorrelated as possible. As our network-backup dataset has less correlated predictors, the RMSE value obtained is good.

As shown in Figure 7 and Figure 8 we can see that the outliers have been removed from the Random Forest model and like the Linear Regression model the points lie close to regressed line. Hence the better RMSE.

## Ques 2c. Neural Network Regression

A Neural Network Regression was built using the PyBrain library. In order to train the model we performed parameter tuning with respect to the number of epochs (one full training cycle on the training set) that the model runs for and the hidden size (the number of hidden nodes for the hidden layer) of the neural network. The results are as follows,

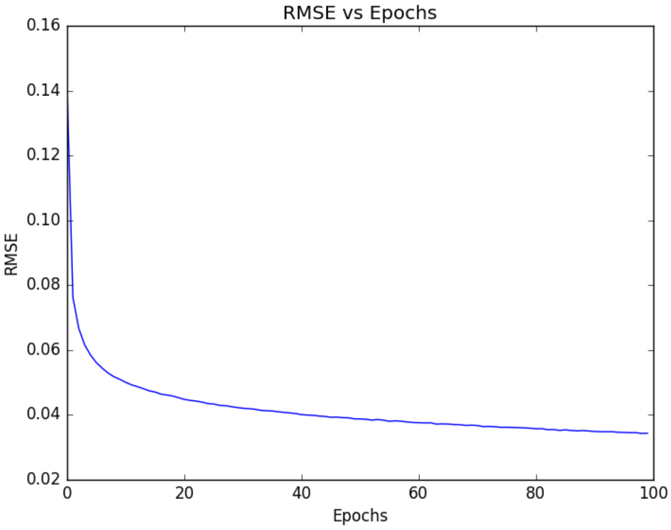


Figure 9: Neural networks - RMSE vs Epochs

The Neural Network was modeled to get the best parameters for number of epochs and hidden size. We identified that the model works best for both parameter values = 100. The **RMSE** obtained 0.042939.

### How Parameters Affect the RMSE

1. Since we can't be sure if 10 epochs or 1000 are enough for convergence since it is dependent on the kind of dataset we feed to the network. Hence a threshold value can be used to stop training when the error goes below a certain threshold. This is crucial as it can also help prevent overfitting of data.

2. Since error may fluctuate, we can also set a parameter( continueEpoch) to set the number of epochs until which it has to wait. This will get the best score or it can stop once it reaches this value. This will help in fine tuning of the epochs. The best RMSE value is obtained when epochs for this dataset is set to 10 and maxEpochs is 100.

3. The number of nodes in hidden layer is dataset dependent. We ran the program for different values of hidden node size starting from 10. The hidden node size which minimizes the RMSE for network dataset is 100.

## Ques 3a. Linear Regression on WorkFlows

We performed linear regression separately on each of the workflows present in the dataset and obtained following results.

**WorkFlow_0**
**Estimated intercept coefficient** : 0.04397
**RMSE Values of Estimator** : 0.02941

| Features | EstimatedCoefficient |
|---|---|
| 0 | -9.709492e-04 |
| 1 | -5.917107e-02 |
| 2 | 8.847221e-02 |
| 3 | -1.073190e-20 |
| 4 | 6.451885e-04 |
| 5 | 1.059369e-01 |

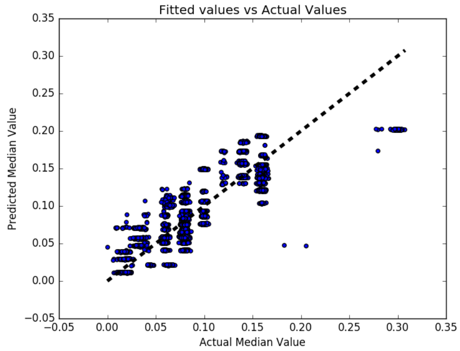


Figure 10: WF 0. Fitted Values vs Actual Values

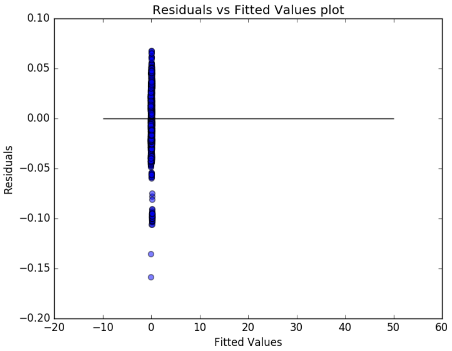


Figure 11: WF 0. Residual Values vs Fitted Values

The RMSE value for the all the workflows together as seen in Question 1 is 0.07956, and for Workflow-0 alone the RMSE value is improved to 0.02941. As seen from the graph of *Residual Vs Fitted values* the Residual goes upto 0.1 and from the graph of *Fitted values versus Actual values* we can see that there is very less deviation between the actual and predicted values. The fit has improved for Workflow-0 compared to the overall one.

**WorkFlow_1**
**Estimated intercept coefficient** : -0.03971
**RMSE Values of Estimator** : 0.10267

| Features | EstimatedCoefficient |
|---|---|
| 0 | 3.923594e-03 |
| 1 | -1.138390e-02 |
| 2 | 6.863092e-02 |
| 3 | -2.625512e-17 |
| 4 | 8.452505e-04 |
| 5 | 4.950399e-01 |

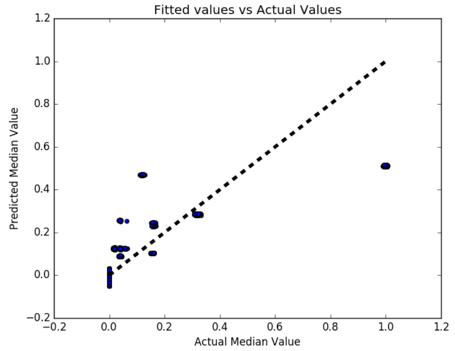


Figure 12: WF 1. Fitted Values vs Actual Values

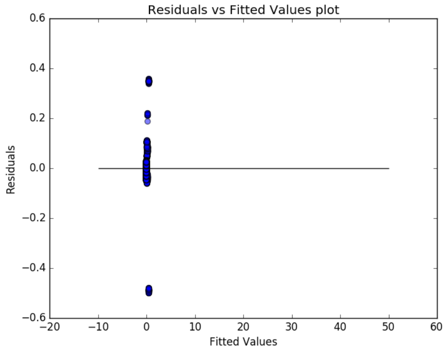


Figure 13: WF 1. Residual Values vs Fitted Values

For Workflow-1 alone the RMSE value is to 0.10267, which is higher than 0.07956. The fit has not improved for Workflow-1 compared to the overall model. As seen from the graph of *Residual Vs Fitted values* the Residual values are high and hence RMSE value is also high.

**WorkFlow_2**
**Estimated intercept coefficient** : -0.008344
**RMSE Values of Estimator** : 0.02545

| Features | EstimatedCoefficient |
|---|---|
| 0 | 2.232091e-03 |
| 1 | 4.597047e-03 |
| 2 | 2.038946e-03 |
| 3 | -1.176710e-17 |
| 4 | 5.315170e-04 |
| 5 | 1.735517e-01 |

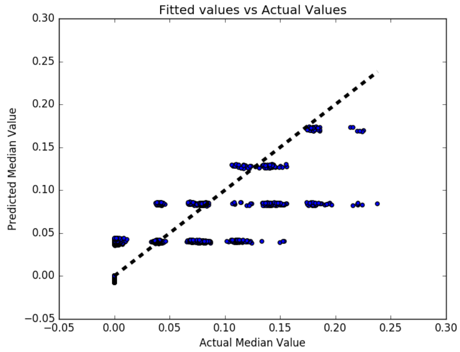


Figure 14: WF 2. Fitted Values vs Actual Values

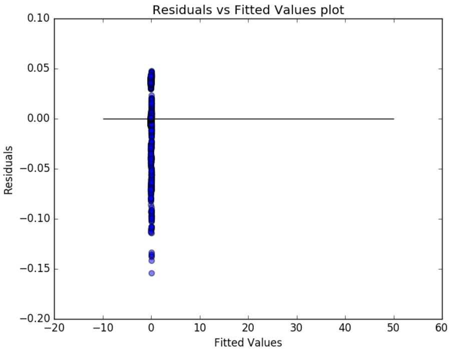


Figure 15: WF 2. Residual Values vs Fitted Values

For Workflow 2 alone the RMSE value is to 0.02545. The fit has improved for Workflow-2 compared to the overall one. As seen from the graph of *Residual Vs Fitted values* the residual values obtained deviates to the range of -0.15 to 0.05. From the graph of *Fitted values versus Actual values* we can see that the values are scattered giving a high standard deviation.

**WorkFlow_3**
**Estimated intercept coefficient** : -0.001411
**RMSE Values of Estimator** : 0.005859

| Features | EstimatedCoefficient |
|---|---|
| 0 | 4.979755e-04 |
| 1 | 1.488333e-03 |
| 2 | 9.546763e-04 |
| 3 | 1.310738e-18 |
| 4 | -2.074723e-04 |
| 5 | 1.681548e-02 |

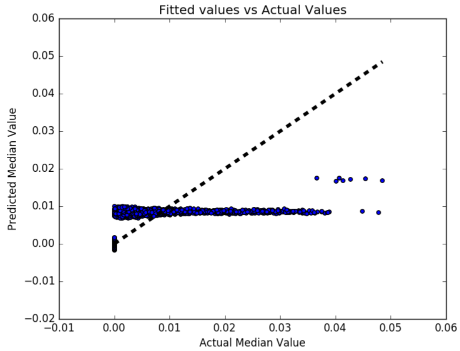


Figure 16: WF 3. Fitted Values vs Actual Values

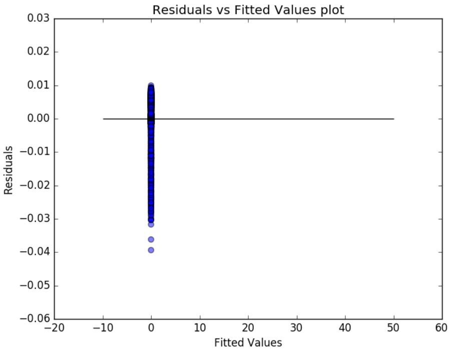


Figure 17: WF 3. Residual Values vs Fitted Values

For Workflow-3 alone the RMSE value is to 0.005859. This workflow records the best RMSE. The fit has improved for Workflow-3 compared to the overall one. As seen from the graph of *Residual Vs Fitted values* the residual values obtained deviates to the range of -0.04 to 0.01. From the graph of *Fitted values versus Actual values* we can see a constant prediction of 0.1. This is mainly due to the size of files being small.

**WorkFlow_4**
**Estimated intercept coefficient** : 0.03364
**RMSE Values of Estimator** : 0.08357

| Features | EstimatedCoefficient |
|---|---|
| 0 | 1.222092e-03 |
| 1 | 1.722148e-01 |
| 2 | -3.977649e-03 |
| 3 | 1.820625e-16 |
| 4 | 1.470688e-03 |
| 5 | 5.385042e-02 |

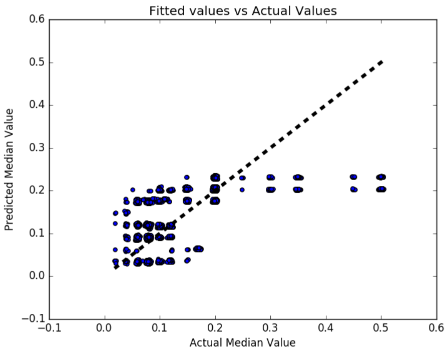


Figure 18: WF 4. Fitted Values vs Actual Values

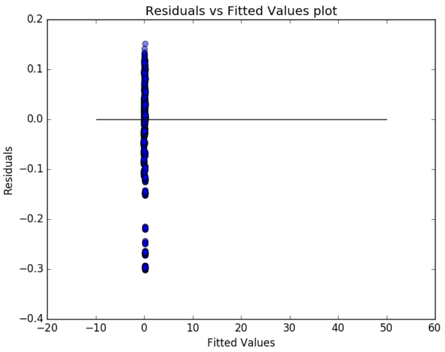


Figure 19: WF 4. Residual Values vs Fitted Values

For Workflow 4 alone the RMSE value is to 0.08357 which is higher than 0.07956.The fit has not improved for Workflow-4 compared to the overall one. As seen from the graph of *Residual Vs Fitted values* the residual values obtained is 0.3 which is high.

Hence for individual workflows we can see the RMSE has improved for all, except for the two workflows - 1 and 4.

## Ques 3b. Polynomial Regression

In order to test our prediction model, we tried fitting a polynomial function to our variables. We tried and tested the model, by fitting the polynomial function upto power of 10. For polynomial regression we plotted **RMSE vs polynomial degree** and found that for **degree 5** the RMSE value is minimum and then we get constant RMSE value as shown in the plot below.

We used this obtained degree on our entire dataset by splitting the dataset as 90 percent training and 10 percent testing. The input attributes of the dataset were transformed to the degree of 5 and the model was created and tested. We obtained an **RMSE Value = 0.16**.

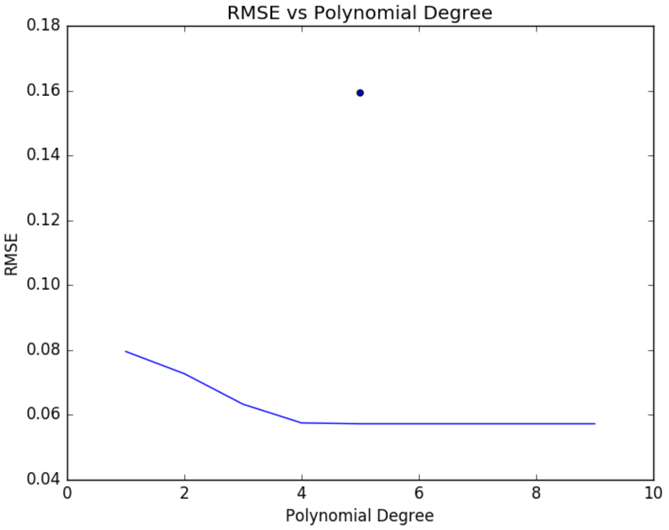


Figure 20: Polynomial Regression - RMSE vs Degree of Polynomial

### How Cross-Validation Helps Control Complexity of the Model

1. Cross validation has been used as a way to measure predictive performance of a statistical model. Since we are using statistics of model fitting we might not always get accurate results. Getting a high R-square values is not a definitive way of defining a good model as it can be easily over-fitted by using high degrees of freedom. Thus a model made to fit best on training data using polynomial regression may not work well on actual real world data.

2. In CV, test data is not used in model estimation. Thus the MSE calculated on this data will give a more predictive accuracy. This is done over multiple folds to get a more conservative result. Thus, this method provides a nearly unbiased measure of MSE on new observations.

3. Thus, CV separates training data which is used for model selection and testing data which is used to validate the model. We can use this approach with different methods of training. The method which gives minimum cross validation error can be chosen.

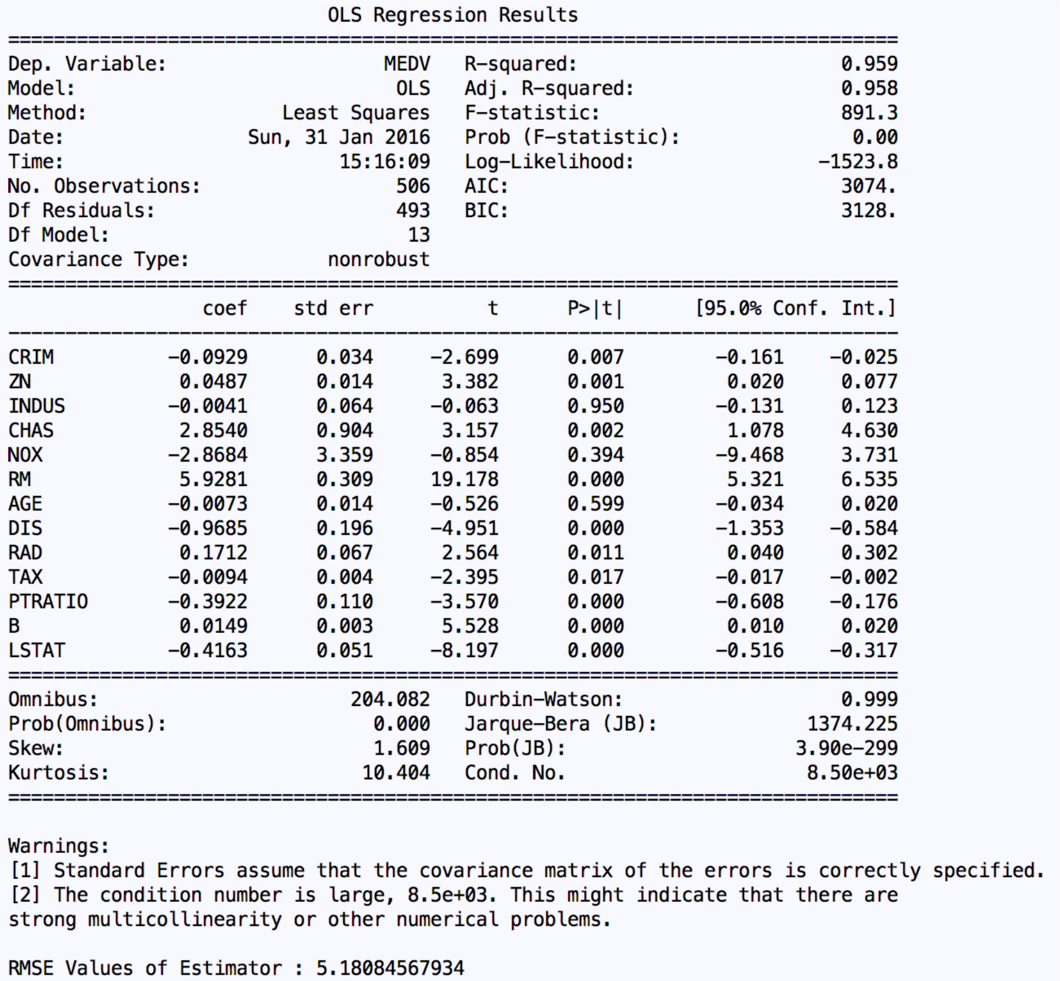
All these factors can thus help in controlling the complexity of the model.

## BOSTON HOUSING

For details about the dataset look at Appendix A: Boston Housing Dataset.

### Ques 4a. Linear Regression

We first performed linear regression with MEDV as the target variable and the other attributes as the features and ordinary least square as the penalty function. The same method of 10 folds Cross-Validation was used to create and test the model as used before for Network-Backup dataset. The results obtained were as follows.



```
                            OLS Regression Results
==============================================================================
Dep. Variable:                   MEDV   R-squared:                       0.959
Model:                            OLS   Adj. R-squared:                  0.958
Method:                 Least Squares   F-statistic:                     891.3
Date:                Sun, 31 Jan 2016   Prob (F-statistic):               0.00
Time:                        15:16:09   Log-Likelihood:                -1523.8
No. Observations:                 506   AIC:                             3074.
Df Residuals:                     493   BIC:                             3128.
Df Model:                          13
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
CRIM          -0.0929      0.034     -2.699      0.007        -0.161    -0.025
ZN             0.0487      0.014      3.382      0.001         0.020     0.077
INDUS         -0.0041      0.064     -0.063      0.950        -0.131     0.123
CHAS           2.8540      0.904      3.157      0.002         1.078     4.630
NOX           -2.8684      3.359     -0.854      0.394        -9.468     3.731
RM             5.9281      0.309     19.178      0.000         5.321     6.535
AGE           -0.0073      0.014     -0.526      0.599        -0.034     0.020
DIS           -0.9685      0.196     -4.951      0.000        -1.353    -0.584
RAD            0.1712      0.067      2.564      0.011         0.040     0.302
TAX           -0.0094      0.004     -2.395      0.017        -0.017    -0.002
PTRATIO       -0.3922      0.110     -3.570      0.000        -0.608    -0.176
B              0.0149      0.003      5.528      0.000         0.010     0.020
LSTAT         -0.4163      0.051     -8.197      0.000        -0.516    -0.317
==============================================================================
Omnibus:                      204.082   Durbin-Watson:                   0.999
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1374.225
Skew:                           1.609   Prob(JB):                    3.90e-299
Kurtosis:                      10.404   Cond. No.                     8.50e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 8.5e+03. This might indicate that there are
strong multicollinearity or other numerical problems.

RMSE Values of Estimator : 5.18084567934
```

Figure 21: Linear Regression Results - Housing Dataset

Here MEDV is the dependent variable and the remaining variables are independent variables.

1. The overall regression accuracy can be seen from R-square and adjusted R-square.
2. The R-square of 0.959 shows that the variance in MEDV is dependent by 95% on the remaining features. Thus the model is a good fit.
3. In the above table, we can see that except INDUS, NOX and AGE, all other variables are significant due to their p-values.
4. RMSE Value of the estimator is 5.1808 which is due to the presence of outliers in the data as indicated in Figure 11.

### Fitted vs. Actual Values

In the below plot of **Fitted Values** vs. **Actual Values** of MEDV, we can see that most of the points are close to the regressed diagonal line. There's a strong correlation between the model's predictions and its actual results.

Hence the actual values and the predicted values are close to each other with small deviation. This validates the high R square value obtained for this model. A high R square indicates that the goodness of fit is strong and thus the above plot should have majority points close to the diagonal. For a actual median value of 50, we see that the predicted values are underestimated by the model. Thus, overall we can see a linear relationship between actual values and the predicted values.

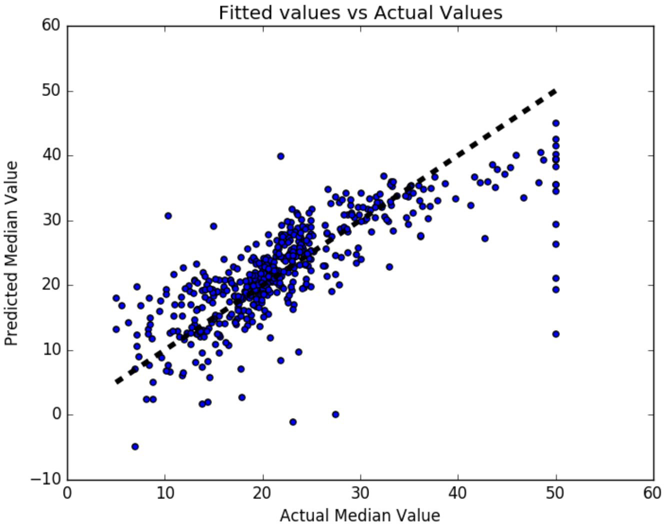


Figure 22: Linear Regression - Fitted Values vs. Actual Values of MEDV

### Residuals vs. Actual Values

Since the data points don't fit the regression line exactly, the coefficients calculated are actually point estimates which are mean values from distribution of possible coefficient values. Hence to check the statistical significance of the model coefficients, residuals are analyzed.

From the plot of **Residual Values** vs. **Fitted Values** we can see that the residuals are close to the fitted line and a cluster can be seen in the middle of the plot. Also, there are no clear patterns visible which makes this plot an ideal one. However, there are some outliers visible which can be seen from the high absolute value of the residual for some data points.

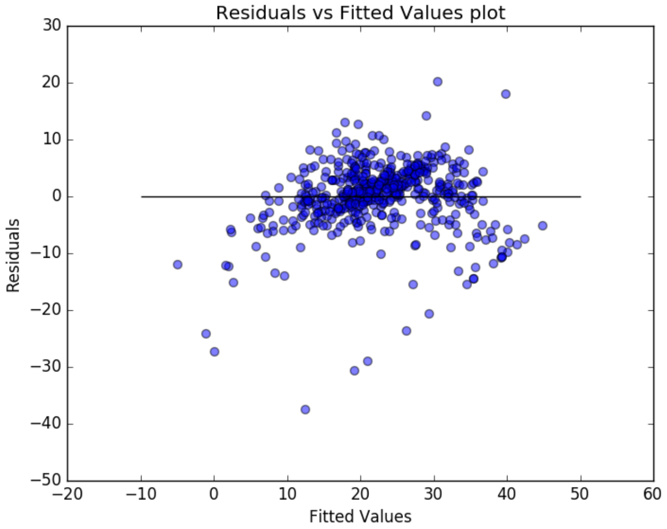


Figure 23: Linear Regression - Residual Values vs. Fitted Values of MEDV

## Ques 4b. Polynomial Regression

To understand the need of polynomial regression, we plotted scatter plots of target variable with individual independent variables. Here are the scatter plots

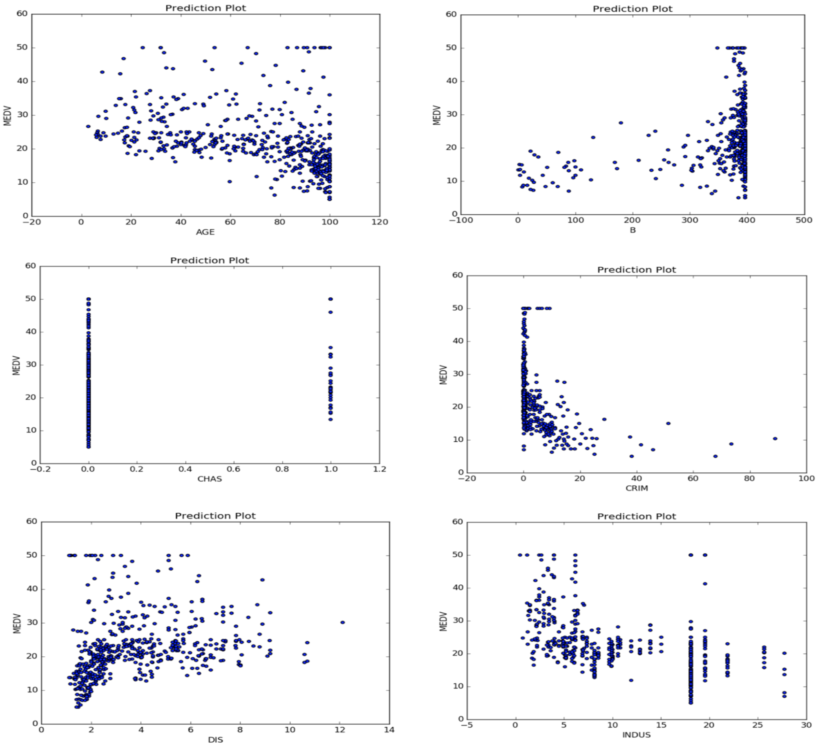


Figure 24: MEDV Relationship with AGE, B, CHAS, CRIM, DIS, INDUS

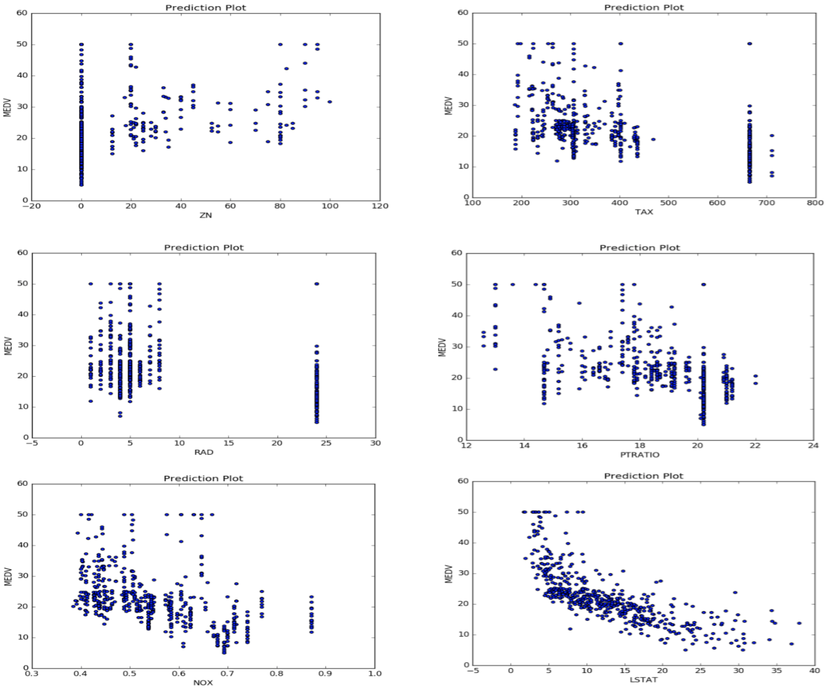


Figure 25: MEDV Relationship with ZN, TAX, RAD, PTRATIO, NOX, LSTAT

From these scatter plots we can see some features like **LSAT, CRIM** have a **non-linear relationship** with MDEV. Hence using Linear Regression only will not necessarily give the best fit. Hence we use polynomial regression in these scenarios to improve the fit. On applying polynomial regression, we get a plot as seen below.

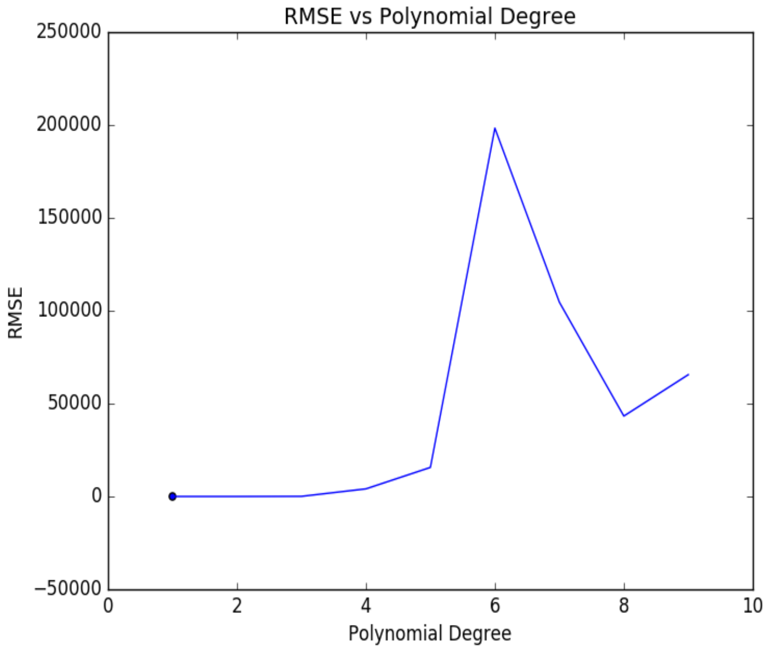


Figure 26: Polynomial Regression - RMSE vs. Degree of Polynomial

Here we see that the best fit is observed when the **Polynomial Degree is from 1 to 3**. Beyond this the RMSE values goes on increasing. We can hence state that **the threshold degree is 3**. Beyond this point, the generalization error of the model gets worse. The RMSE obtained for this model was **1.5832**

## Ques 5. Ridge Regression and Lasso Regularization

### RIDGE REGRESSION

1. Ridge regression is similar to least squares but shrinks the estimated coefficients towards zero.
2. RidgeCV implements ridge regression with built-in cross-validation of the alpha parameter.
3. RidgeCV generates the best alpha after performance tuning on the given dataset.

**Best Alpha value for Ridge Regression** : 0.1
**Best RMSE for corresponding Alpha** = 4.6795

### LASSO REGULARIZATION

1. The Lasso is a linear model that estimates sparse coefficients. It is useful in some contexts due to its tendency to prefer solutions with fewer parameter values, effectively reducing the number of variables upon which the given solution is dependent. For this reason, the Lasso and its variants are fundamental to the field of compressed sensing.
2. The library used for performing lasso regression is LassoCV. After performance tuning, the alpha obtained is given below.

**Best Alpha value for Lasso Regularization** : 0.01
**Best RMSE for corresponding Alpha** = 4.8658

# APPENDIX A : Dataset Description

## NETWORK-BACKUP DATASET

The Network-Backup Dataset has information of files maintained in destination machine and it monitors and copies their changes in four hours cycle. The features captured in data set are as follows.

1. **Week index**
2. **Day of the week**: at which the file is backed up starts
3. **Backup start time - Hour of the day**: the exact time that the backup process is completed
4. **Workflow ID**
5. **File name**
6. **Backup size**: the size of the file that is backed up in that cycle in GB
7. **Backup time**: the duration of the backup procedure in hour

## BOSTON-HOUSING DATASET

The Boston-Housing Dataset has information of housing values in the suburbs of the greater Boston area. The features captured in data set are as follows.

1. **CRIM**: per capita crime rate by town
2. **ZN**: proportion of residential land zoned for lots over 25,000 sq. ft.
3. **INDUS**: proportion of non-retail business acres per town
4. **CHAS**: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
5. **NOX**: nitric oxides concentration (parts per 10 million)
6. **RM**: average number of rooms per dwelling
7. **AGE**: proportion of owner-occupied units built prior to 1940
8. **DIS**: weighted distances to five Boston employment centers
9. **RAD**: index of accessibility to radial highways
10. **TAX**: full-value property-tax rate per $10,000
11. **PTRATIO**: pupil-teacher ratio by town
12. **B**: $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town
13. **LSTAT**: % lower status of the population
14. **MEDV**: Median value of owner-occupied homes in $1000’s

# APPENDIX B : REGRESSION STATISTICS

The description for the statistics used through the report can be found below.

1. **p-value** :

   (a) A low p-value ($< 0.05$) indicates that you can reject the null hypothesis. In other words, a predictor that has a low p-value is likely to be a meaningful addition to your model because changes in the predictor's value are related to changes in the response variable.

   (b) Conversely, a larger (insignificant) p-value suggests that changes in the predictor are not associated with changes in the response.

2. **coefficient** :

   (a) Size of the coefficient for each independent variable gives you the size of the effect that variable is having on your dependent variable, and the sign on the coefficient (positive or negative) gives you the direction of the effect.

   (b) It tells us how much the dependent variable is expected to increase (if the coefficient is positive) or decrease (if the coefficient is negative) when that independent variable increases by one.

   (c) In regression with multiple independent variables, the coefficient tells you how much the dependent variable is expected to increase when that independent variable increases by one, holding all the other independent variables constant.