EAS 506 Statistical Data Mining — Jaideep Reddy Kommera

Homework 2 — Class no. 28

Question 1:

Introduction: The given dataset is 'College' dataset from ISLR package. It has 777 instances and 18 variables. The dataset contains different variables necessary during College Application process. The target variable here is Number of Applications.

Part a)

Train-Test Split: The dataset has been split into train and test. 75% of the data has been allotted to training and 25% has been allotted to testing. set.seed() has been used before splitting data.

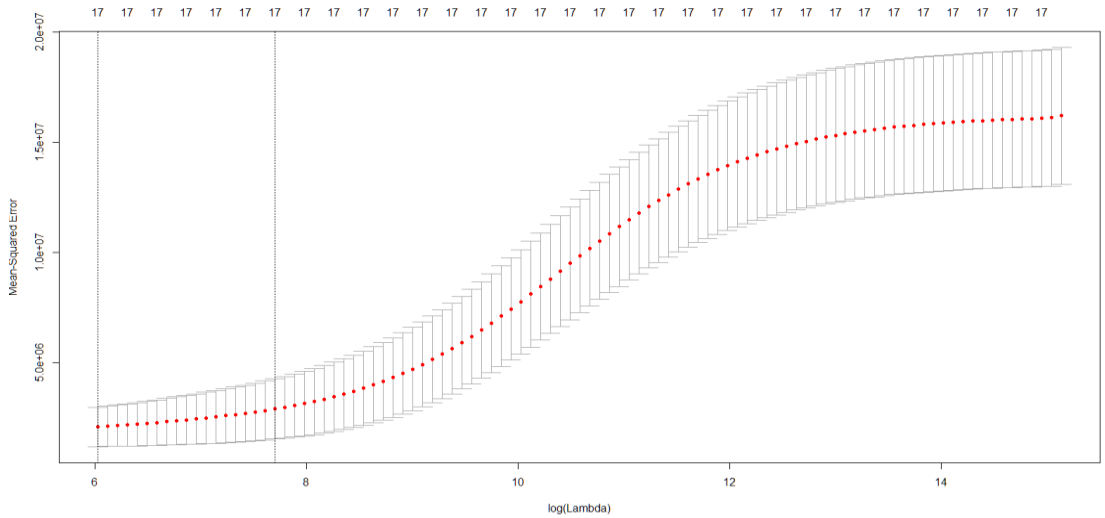
Linear Model: Building Linear Regression model on train data and then using the model on test-data to predict the number of applications.

```
> lm_error
[1] 562112.4
```

The Mean squared error of the test predictions by linear regression is as above.

Part b)

Ridge Regression: Building a ridge regression model with λ chosen by cross validation. The function cv.glmnet() from library 'glmnet' has been used to perform cross validation and choosing the right λ. α has been set to '0' for ridge regression and 10 fold validation has been considered.

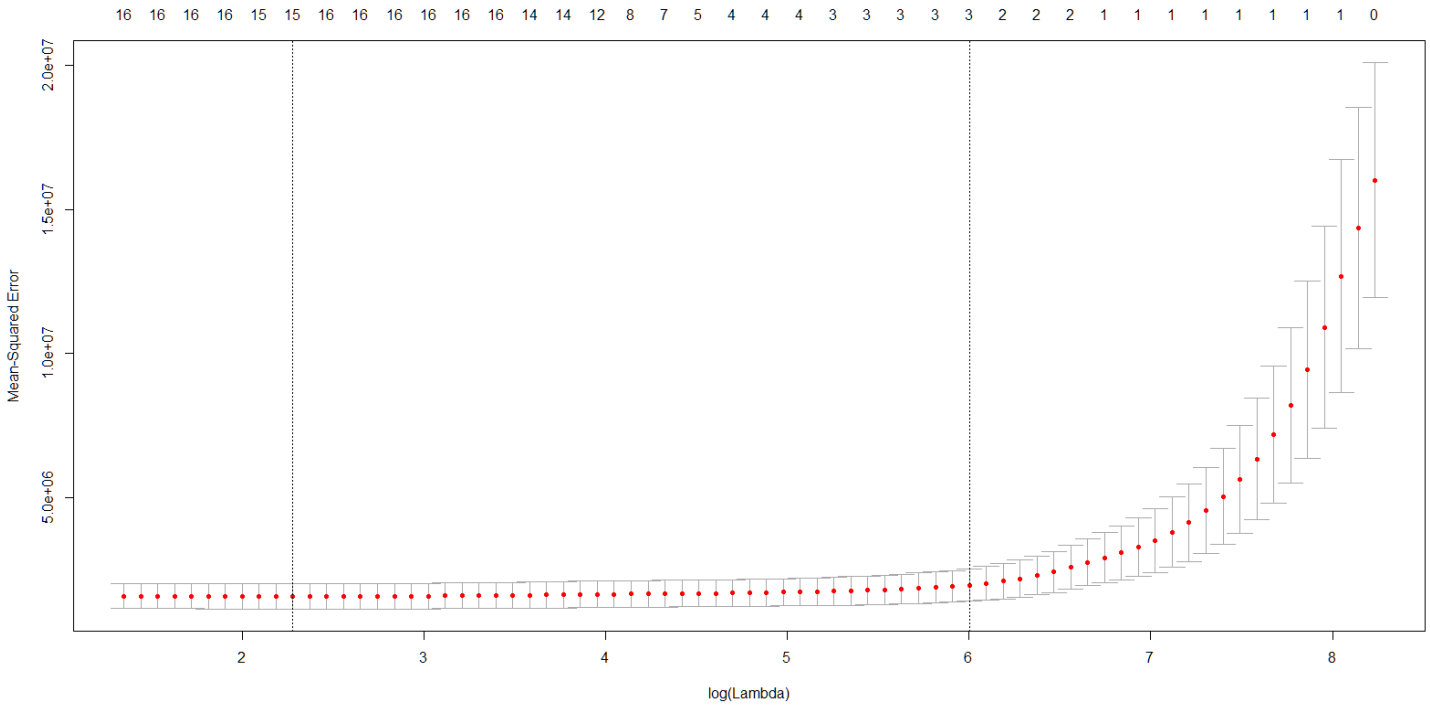


After finding the minimum value of λ by cross-validation the model is used to predict on the test data using that value of λ. The resultant mean squared error from ridge regression on the test data is as below

```
> ridge_error
[1] 647759.7
```

Part c)

Lasso Regression: Building a Lasso regression model with λ chosen by cross validation. The function cv.glmnet() from library 'glmnet' has been used to perform cross validation and choosing the right λ. α has been set to '1' for ridge regression and 10 fold validation has been considered.



After finding the minimum value of λ by cross-validation the model is used to predict on the test data using that value of λ. The resultant mean squared error from Lasso regression on the test data is as below

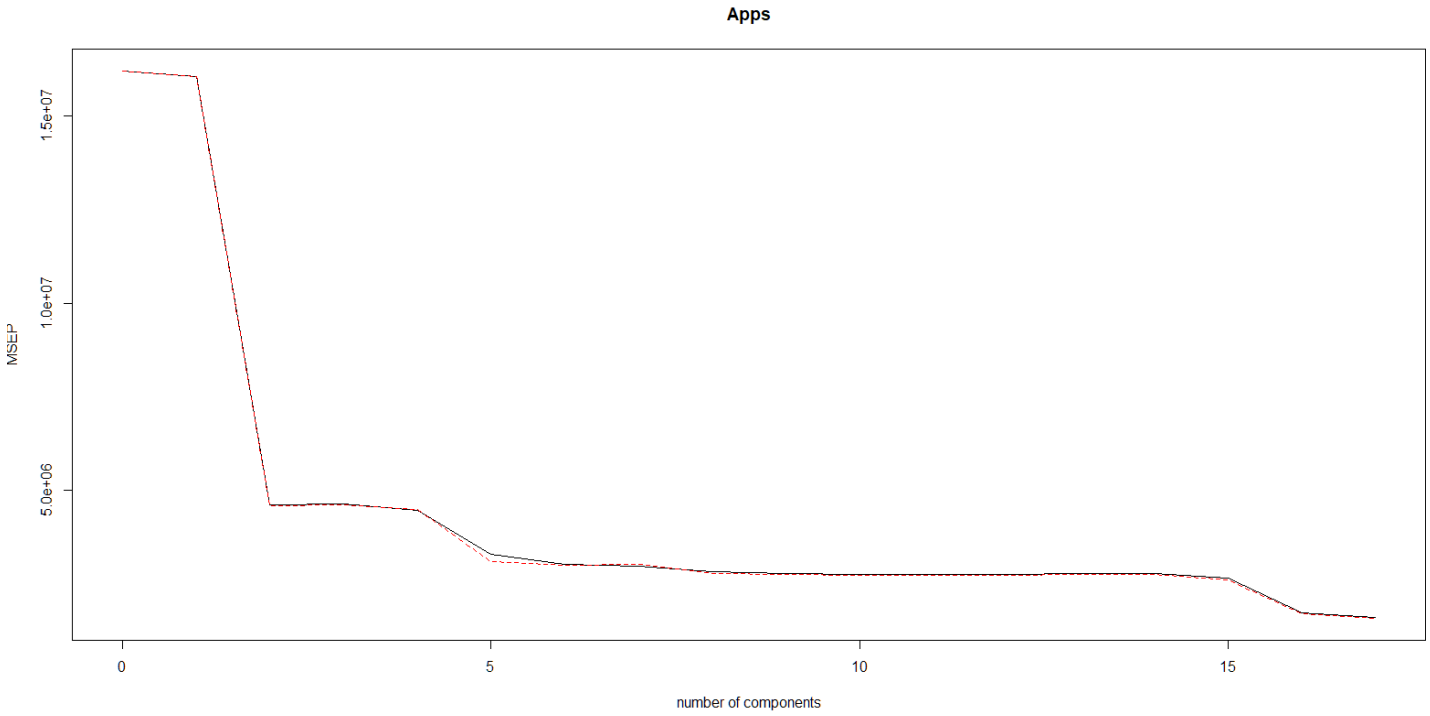
```
> lasso_error
[1] 539928.7
```

Part d)

Principal Component Regression: The principal component regression model is built using the 'pcr()' function from the library 'pls'. The hyper parameter validation is assigned as 'CV' to perform cross validation.

```
TRAINING: % variance explained
      1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps  9 comps  10 comps  11 comps  12 comps
X      32.041    57.52    64.53    70.21    75.43    80.42    83.95    87.44    90.57     92.93     94.94     96.79
Apps    1.642    72.92    73.05    74.09    82.40    83.24    83.29    84.55    84.61     84.78     84.81     84.91
      13 comps  14 comps  15 comps  16 comps  17 comps
X        97.88     98.74     99.35     99.84    100.00
Apps     84.92     84.95     89.27     91.99     92.37
```

The above summary shows the percentage of variance explained



From the above Validation plot we can see that the there is a flat line where the MSE doesn't change much. So, selecting the number of components at the start of the flat line should give us the same results as selecting more number of components anywhere else on the line.

So, based on the percentage of variance explained and the plot I have chosen a K value of 8.

```
> pcr_error
[1] 1112551
```

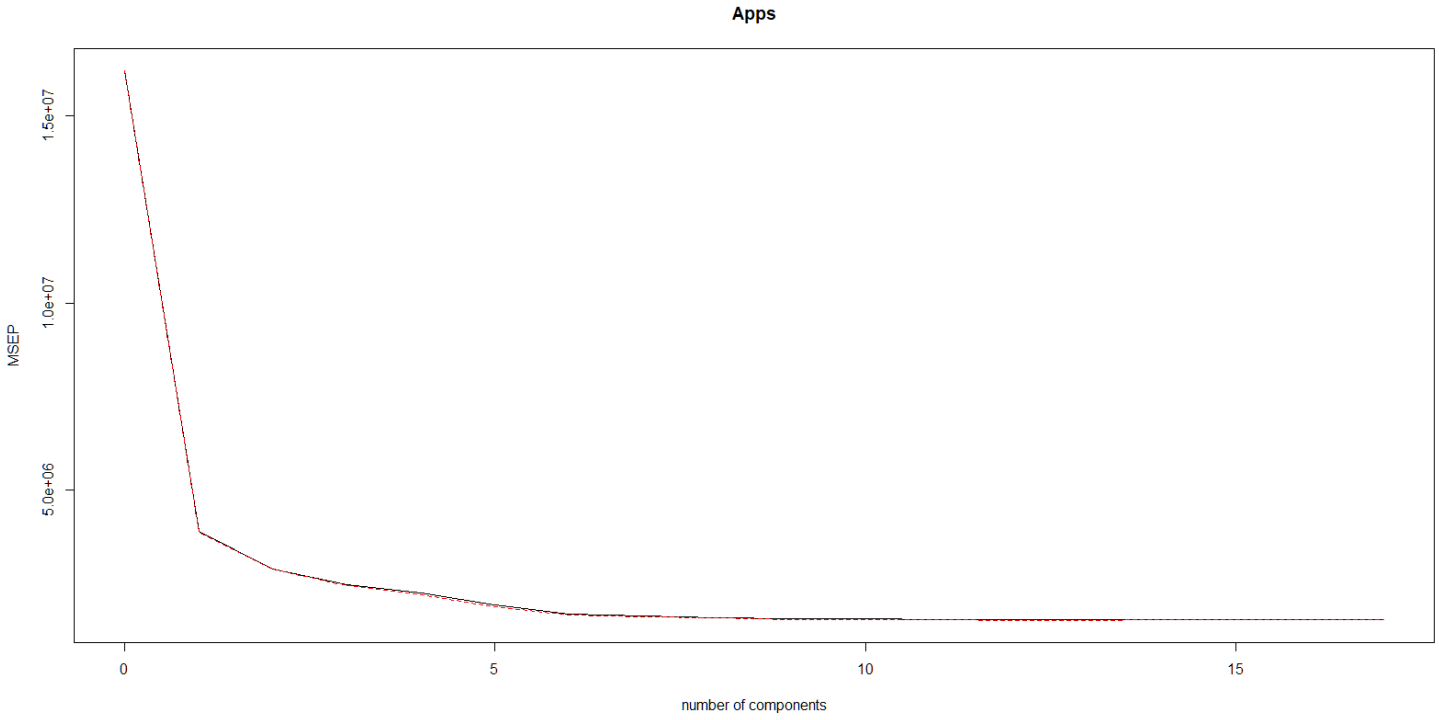
Therefore, the MSE using PCR is as above.

Part e)

Partial Least Squares: The Partial least squares model is built using the 'pslr' function from the library 'pls'. For cross-validation the parameter validation is assigned as 'cv'.

```
TRAINING: % variance explained
      1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps  9 comps  10 comps  11 comps  12 comps
X      32.041    57.52    64.53    70.21    75.43    80.42    83.95    87.44    90.57     92.93     94.94     96.79
Apps    1.642    72.92    73.05    74.09    82.40    83.24    83.29    84.55    84.61     84.78     84.81     84.91
      13 comps  14 comps  15 comps  16 comps  17 comps
X        97.88     98.74     99.35     99.84    100.00
Apps     84.92     84.95     89.27     91.99     92.37
```

The above summary shows the amount of variance explained



So, based on the percentage of variance explained and the plot I have chosen a K value of 6.

```
> plsr_error
[1] 567045.8
```

The MSE for Partial Least squares model is as above.

Part f)

PCR has given the highest mean square error. The other models namely Linear, Ridge, Lasso and PLS have performed similarly to each other. They all performed better than PCR. Any of those better performing models can be used to predict the college applications.

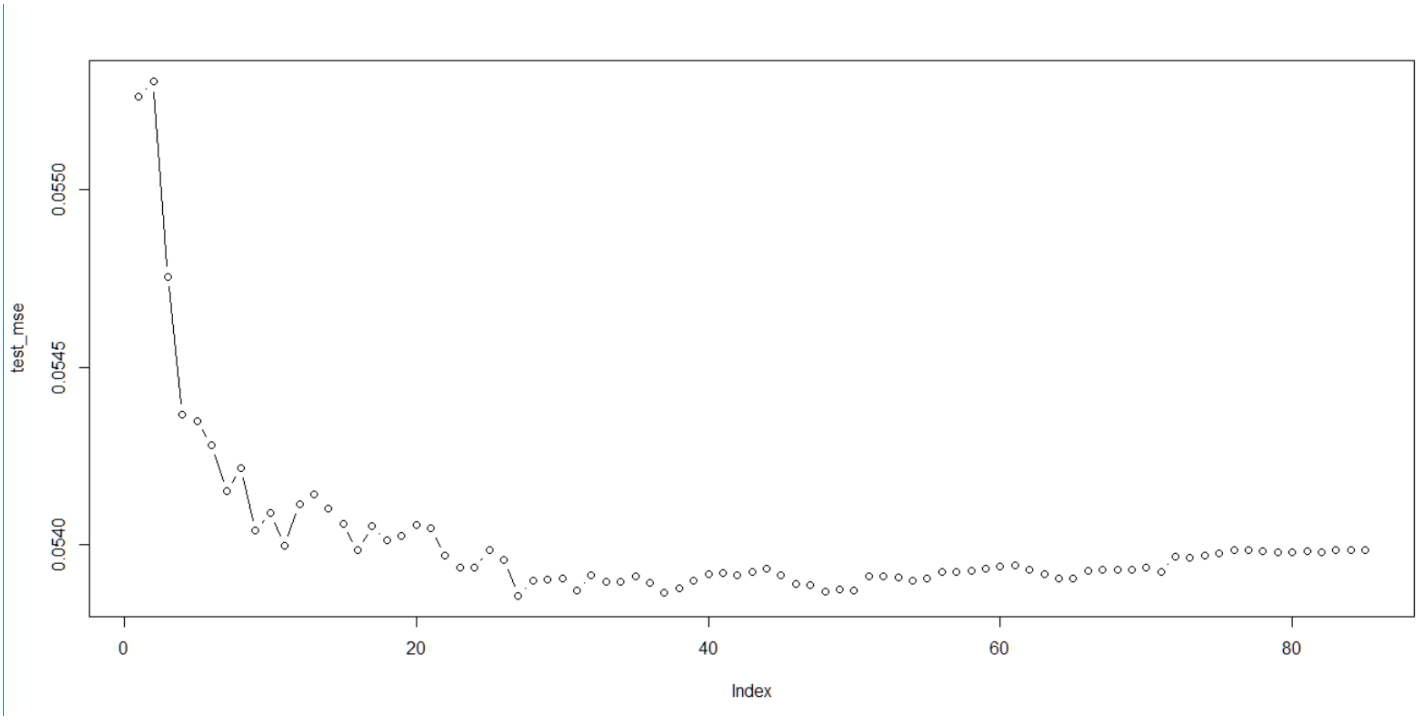
Question 2:

Introduction: The given dataset is insurance company benchmark data set. The dataset contains 86 variables. The data has already been split into train and test. The train data has 5822 instances and the test data has 4000 instances. The target is to predict who will be interested in buying caravan insurance policy.

Linear Model: Building a linear model with the V86 (has insurance policy, yes or no) as the target variable. The Mean Squared error obtained from

```
> lm_error
[1] 0.053985
```

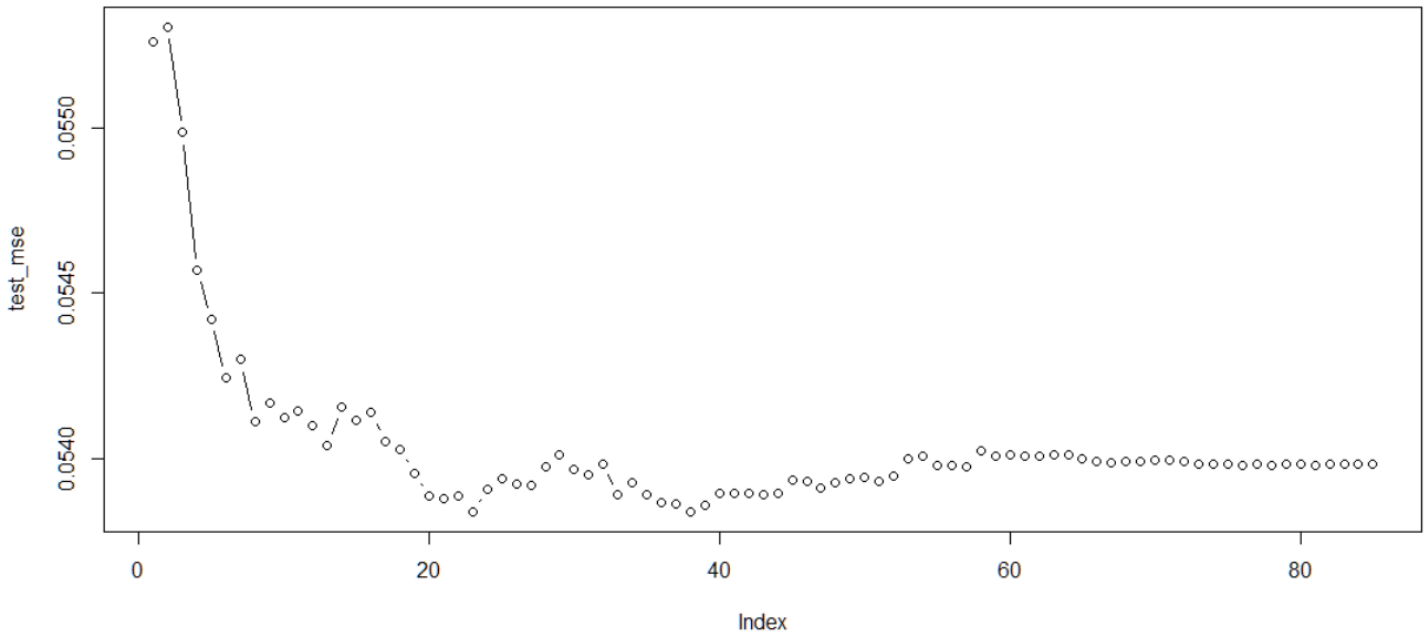
Forward Selection: Forward subset selection is done using 'regsubsets' function from the "leaps" library.



By calculating the Mean squared error on the test data using different sizes of subset we have found out that the least MSE on test data is when the subset has 27 components.

```
> which(test_mse==min(test_mse))
[1] 27
```

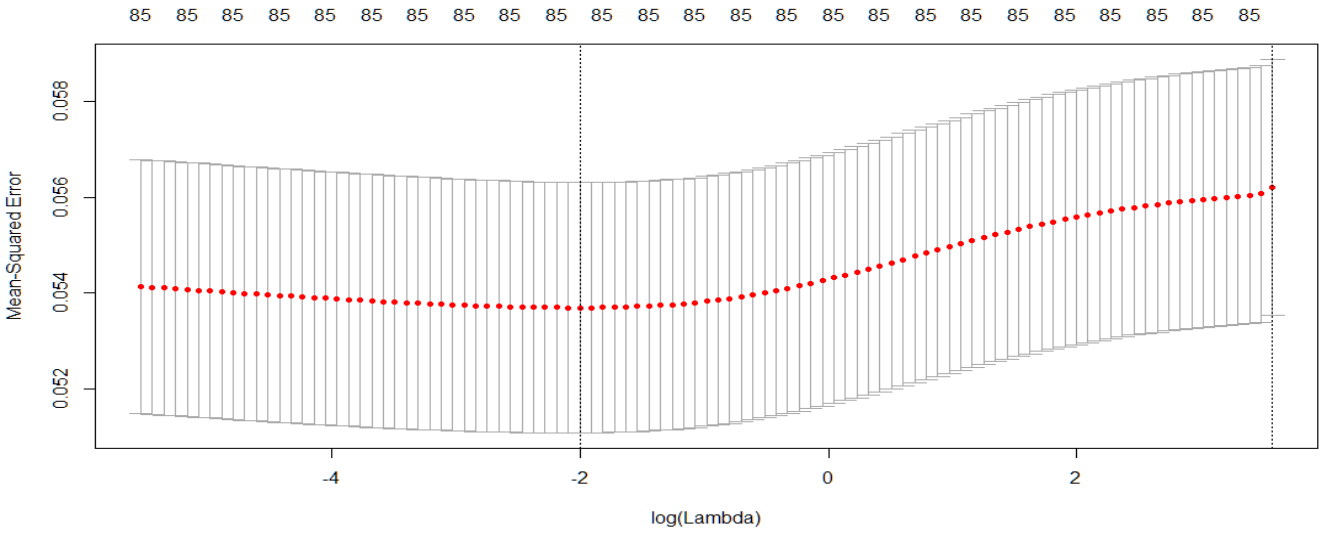
Backward Selection: Back subset selection is done using ‘regsubsets’ function from the “leaps” library.



From the above plot by calculating the Mean squared error on the test data using different sizes of subset we have found out that the least MSE on test data is when the subset has 38 components.

```
> which(test_mse==min(test_mse))
[1] 38
```

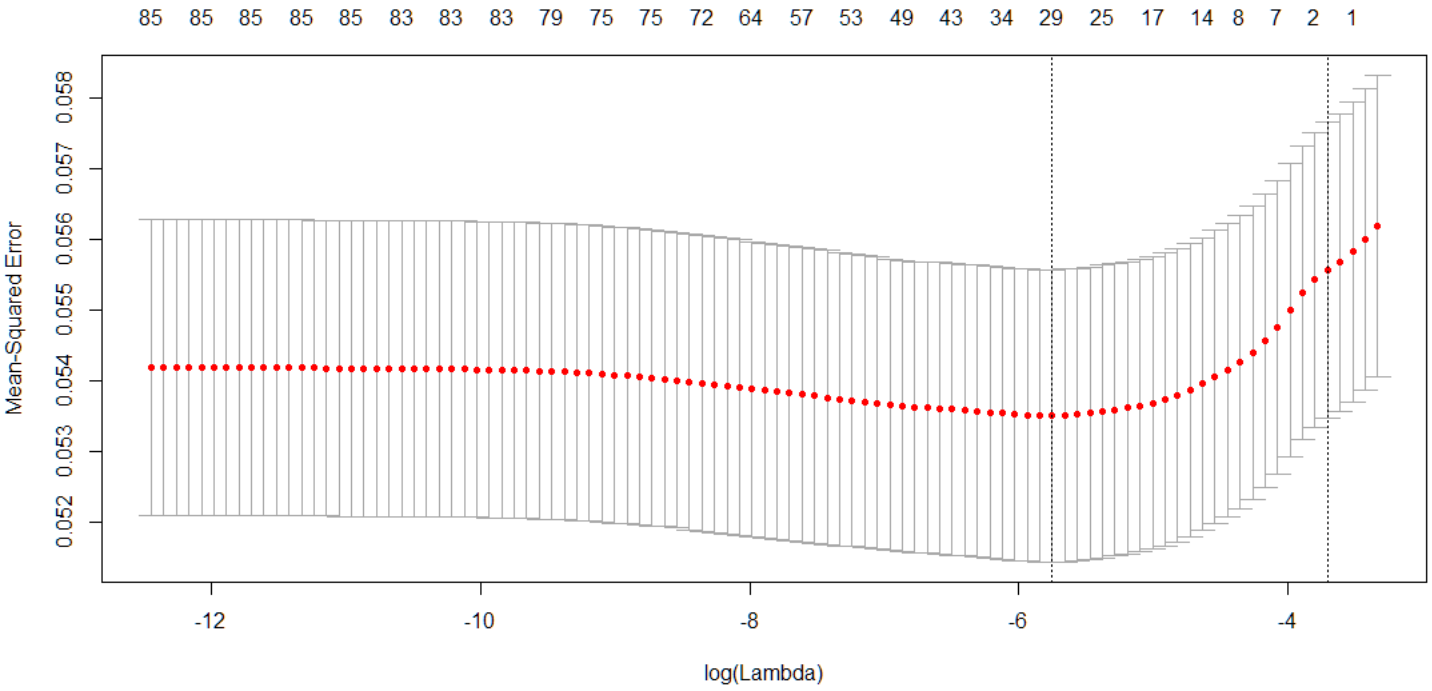
Ridge Regression: Building a ridge regression model with λ chosen by cross validation. The function cv.glmnet() from library ‘glmnet’ has been used to perform cross validation and choosing the right λ. α has been set to ‘0’ for ridge regression and 10 fold validation has been considered.



After finding the minimum value of λ by cross-validation the model is used to predict on the test data using that value of λ. The minimum λ is as below

```
> cv_ridge$lambda.min
[1] 0.162238
```

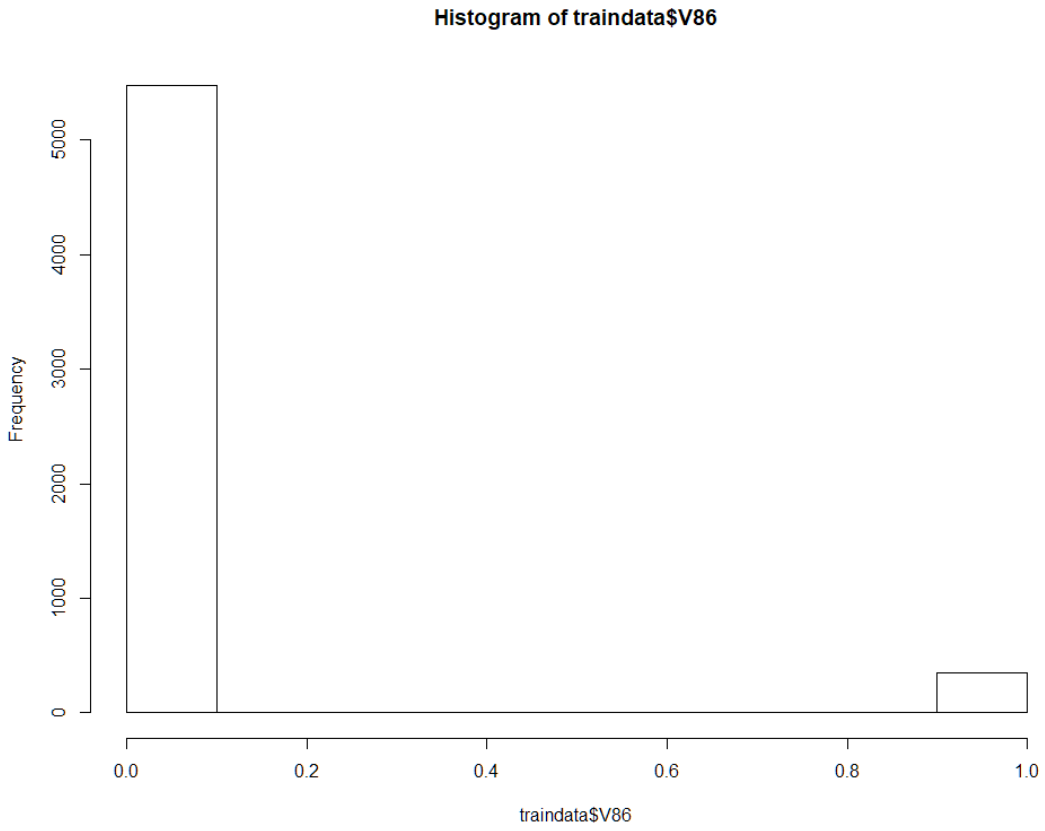
Lasso Regression: Building a Lasso regression model with λ chosen by cross validation. The function cv.glmnet() from library 'glmnet' has been used to perform cross validation and choosing the right λ. α has been set to '1' for ridge regression and 10-fold validation has been considered.



After finding the minimum value of λ by cross-validation the model is used to predict on the test data using that value of λ. The minimum λ is as below

```
> cv_lasso$lambda.min
[1] 0.003184799
```

Inference: The target variable has 2 levels (0 or 1) and the train data has more instances with way more instances with 0's than 1's. This will cause the model to be biased towards predicting more 0's than usual and therefore increases the error and reduces accuracy.



Question 3:

Introduction: The dataset is a simulated dataset. It has 1000 instances and 20 variables. The target variable is generated using the equation $Y=X\beta + \varepsilon$.

Generating Data: Each column has data generated from random normal distribution i.e. each variable has data generated from one normal distribution and the same has been iterated 20 times for all 20 variables.

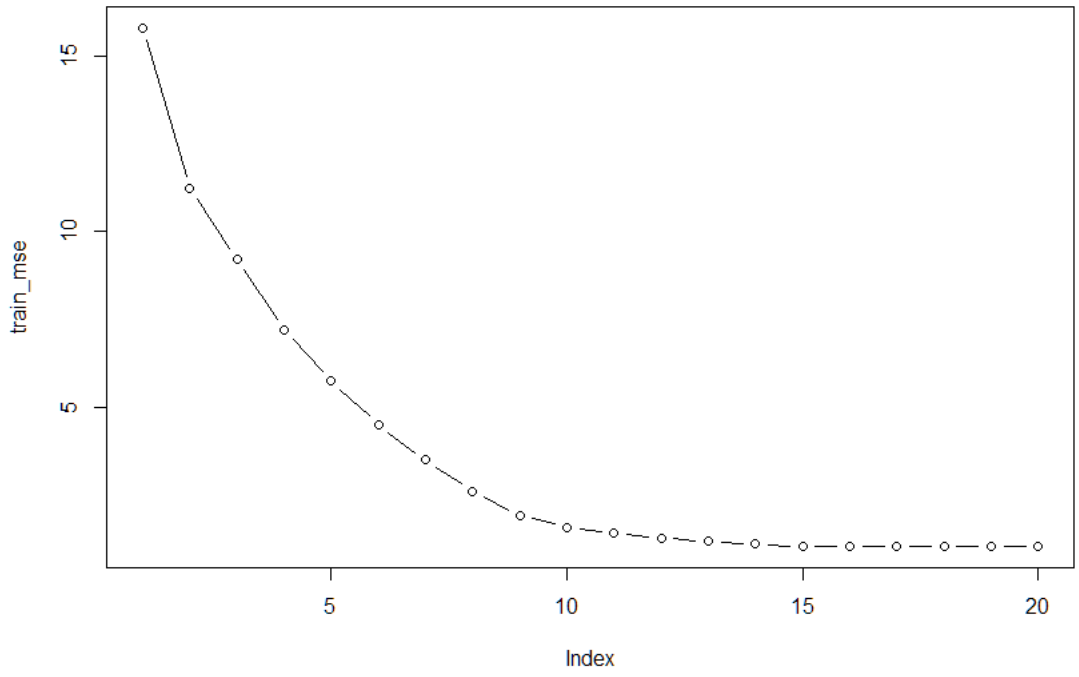
The target variable Y has been generated from the data using the equation $Y=X\beta + \varepsilon$. The coefficients β have also been generated from a normal distribution. ε has been assumed to be a single constant.

5 values of β have been sampled out and made as zeroes.

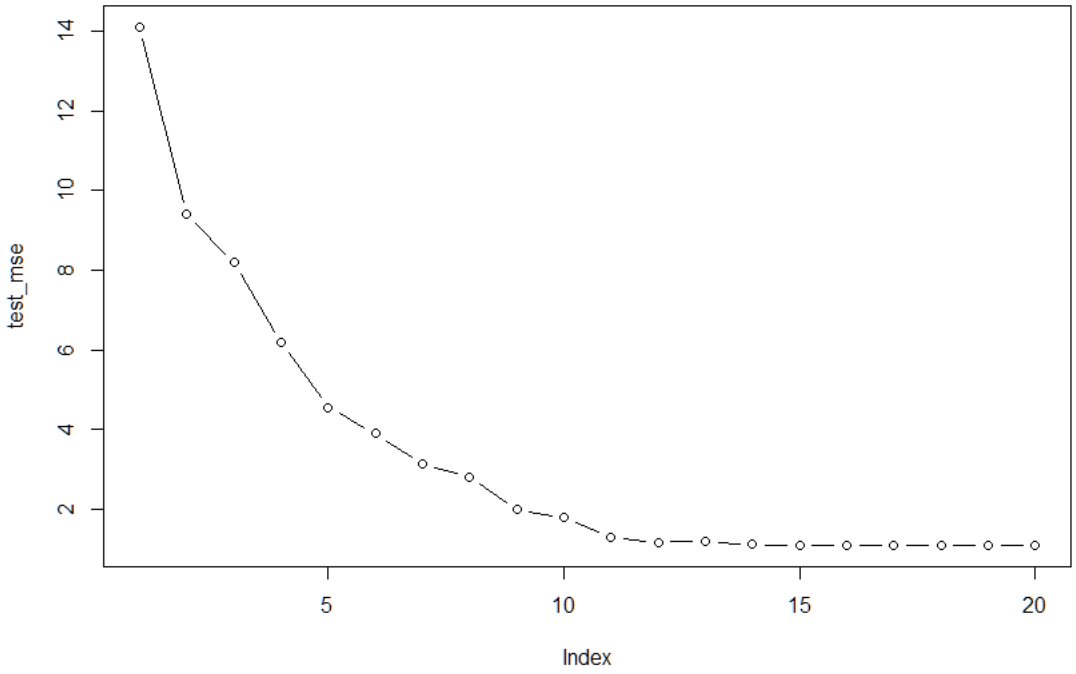
The simulated data has been split into Train data and Test data. The train data has 90% of the total data and the test has been allotted the rest 10%

Best Subset Selection: The best subset selection has been performed on the train and test data.

The plot of MSE with different sizes of subsets on train dataset is as below



The plot of MSE with different sizes of subsets on train dataset is as below



From the above plot we can notice that the best subset for the least MSE on test data is when the number of components is 15.

```
> which(test_mse==min(test_mse))
[1] 15
```

```
> train_mse
 [1] 14.119669 10.531561  8.723991  7.200538  5.647060  4.676032  3.620465  2.774625  2.084405  1.637784  1.373981
[12]  1.194645  1.141604  1.089616  1.081150  1.079043  1.078107  1.077622  1.077536  1.077511
> test_mse
 [1] 14.112546  9.402246  8.184763  6.187838  4.555098  3.889074  3.131890  2.803205  1.978582  1.775292  1.286844
[12]  1.169215  1.180043  1.106381  1.085561  1.095313  1.100052  1.094860  1.098600  1.099506
```

Inference:

After number of components is greater than 15 we can see that the test MSE is increasing and this is due to the randomness induced in the data due to epsilon being random to emulate a natural dataset rather than an ideal one.