P1) CollegeDataSet-HW2

Jeetendra Gan | jgan2 | 50325023

College data set analysis

The data set has 17 predictors and 777 observations.

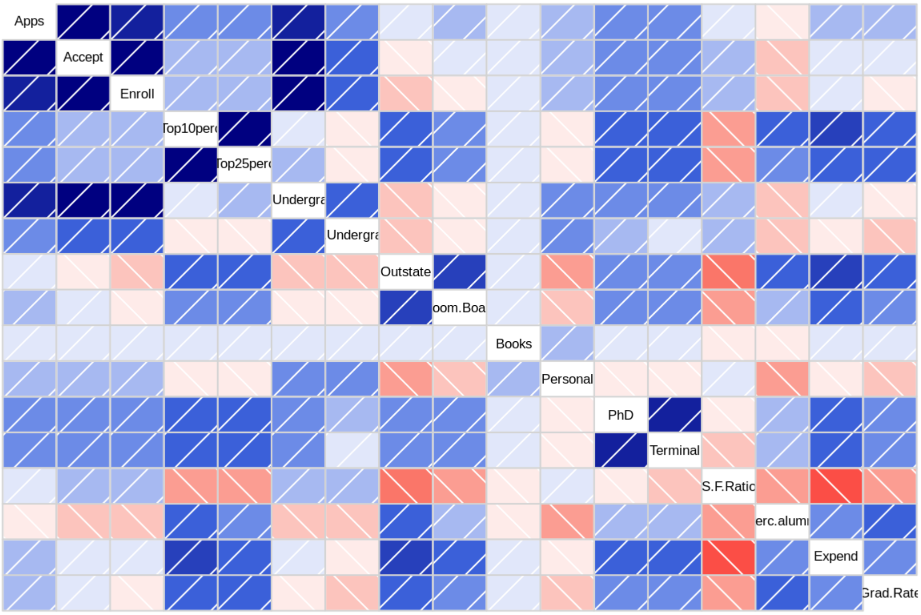
```
##  [1] "Private"     "Apps"        "Accept"      "Enroll"      "Top10perc"
##  [6] "Top25perc"   "F.Undergrad" "P.Undergrad" "Outstate"    "Room.Board"
## [11] "Books"       "Personal"    "PhD"         "Terminal"    "S.F.Ratio"
## [16] "perc.alumni" "Expend"      "Grad.Rate"
```

Following is the summary of each feature

```
##  Private        Apps           Accept          Enroll       Top10perc
##  No :212   Min.   :   81   Min.   :   72   Min.   :  35   Min.   : 1.00
##  Yes:565   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242   1st Qu.:15.00
##            Median : 1558   Median : 1110   Median : 434   Median :23.00
##            Mean   : 3002   Mean   : 2019   Mean   : 780   Mean   :27.56
##            3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902   3rd Qu.:35.00
##            Max.   :48094   Max.   :26330   Max.   :6392   Max.   :96.00
##    Top25perc      F.Undergrad     P.Undergrad         Outstate
##  Min.   :  9.0   Min.   :  139   Min.   :    1.0   Min.   : 2340
##  1st Qu.: 41.0   1st Qu.:  992   1st Qu.:   95.0   1st Qu.: 7320
##  Median : 54.0   Median : 1707   Median :  353.0   Median : 9990
##  Mean   : 55.8   Mean   : 3700   Mean   :  855.3   Mean   :10441
##  3rd Qu.: 69.0   3rd Qu.: 4005   3rd Qu.:  967.0   3rd Qu.:12925
##  Max.   :100.0   Max.   :31643   Max.   :21836.0   Max.   :21700
##    Room.Board       Books           Personal         PhD
##  Min.   :1780   Min.   :  96.0   Min.   : 250   Min.   :  8.00
##  1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850   1st Qu.: 62.00
##  Median :4200   Median : 500.0   Median :1200   Median : 75.00
##  Mean   :4358   Mean   : 549.4   Mean   :1341   Mean   : 72.66
##  3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700   3rd Qu.: 85.00
##  Max.   :8124   Max.   :2340.0   Max.   :6800   Max.   :103.00
##     Terminal       S.F.Ratio      perc.alumni        Expend
##  Min.   : 24.0   Min.   : 2.50   Min.   : 0.00   Min.   : 3186
##  1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00   1st Qu.: 6751
##  Median : 82.0   Median :13.60   Median :21.00   Median : 8377
##  Mean   : 79.7   Mean   :14.09   Mean   :22.74   Mean   : 9660
##  3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00   3rd Qu.:10830
##  Max.   :100.0   Max.   :39.80   Max.   :64.00   Max.   :56233
##    Grad.Rate
##  Min.   : 10.00
##  1st Qu.: 53.00
##  Median : 65.00
##  Mean   : 65.46
##  3rd Qu.: 78.00
##  Max.   :118.00
```

Only the “Private” feature is categorical otherwise everything else is quantitative. I have changed the “Private” variable to a quantitative value. It will be 1 if the college is private, else it will be 0.

Here is the covariance matrix of each of the variables



There is a positive correlation between a few coefficients, which may be eliminated by the techniques used below.

a) Validation set approach

I have divided the data into 70% training set and 30% test set.

Here is the summary of the linear fit on the training data.

```
##
## Call:
## lm(formula = Apps ~ ., data = trainingData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5102.2  -418.9   -16.1   294.2  7283.8
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -337.13510  502.72576  -0.671 0.502761
## Private     -537.56375  168.19849  -3.196 0.001477 **
## Accept         1.62548    0.04498  36.135  < 2e-16 ***
## Enroll        -0.86023    0.21146  -4.068 5.47e-05 ***
## Top10perc     49.83396    6.90784   7.214 1.90e-12 ***
## Top25perc    -14.12674    5.47667  -2.579 0.010166 *
## F.Undergrad    0.03989    0.03895   1.024 0.306211
## P.Undergrad   -0.02430    0.05141  -0.473 0.636712
## Outstate      -0.08299    0.02320  -3.576 0.000381 ***
## Room.Board     0.18024    0.05908   3.051 0.002398 **
## Books          0.00927    0.32057   0.029 0.976942
## Personal       0.06032    0.07900   0.764 0.445458
## PhD           -5.71653    5.97250  -0.957 0.338936
## Terminal      -6.57137    6.68524  -0.983 0.326076
## S.F.Ratio     14.01420   16.19025   0.866 0.387107
## perc.alumni   -1.39267    5.07959  -0.274 0.784062
## Expend         0.07133    0.01516   4.704 3.26e-06 ***
## Grad.Rate      7.42083    3.73181   1.989 0.047270 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1067 on 526 degrees of freedom
## Multiple R-squared:  0.9357, Adjusted R-squared:  0.9336
## F-statistic: 450.2 on 17 and 526 DF,  p-value: < 2.2e-16
```

The R-Squared value is 0.929, which indicates that the amount of variance explained by the model, is pretty decent. It may also be an indicator that the model does not overfit the data as R-squared is not too large. Important variables indicated by the model are

1. **PrivateYes**
2. **Accept**
3. **Enroll**
4. **Top10perc**
5. Top25perc
6. **Outstate**
7. Room.Board
8. **Expend**
9. Grad.Rate

The highlighted features have smaller p values indicating their importance in the model.

The train RSS is:

```
## [1] 599230767
```

The train RMSE is:

```
## [1] 1049.54
```

Test error

The test RSS is

```
## [1] 233128960
```

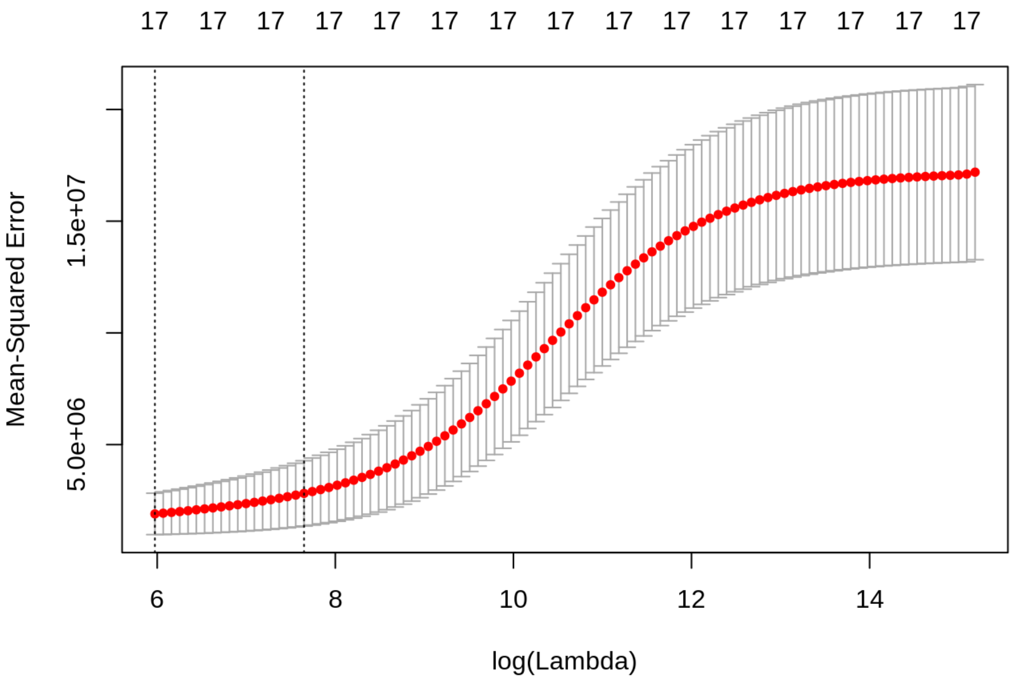
The test RMSE is

```
## [1] 1000.28
```

The train root mean square error is slghtly more than test root mean square error. This indicates that the model generalizes well.

b) Ridge regression

The Mean Squared error for Ridge regression is as shown below. The error increases with increase in penalization term due to addition of bias.



The best lambda is:

```
## [1] 393.17
```

The coefficients associated with the lambda are as follows:

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                         1
## (Intercept) -1.505352e+03
## Private     -5.290337e+02
## Accept       9.835225e-01
## Enroll       4.594292e-01
## Top10perc    2.514429e+01
## Top25perc    9.524357e-01
## F.Undergrad  7.579784e-02
## P.Undergrad  2.440732e-02
## Outstate    -2.192204e-02
## Room.Board   1.995874e-01
## Books        1.338967e-01
## Personal    -8.798046e-03
## PhD         -3.823988e+00
## Terminal    -4.733621e+00
## S.F.Ratio    1.286811e+01
## perc.alumni -8.772719e+00
## Expend       7.540216e-02
## Grad.Rate    1.134616e+01
```

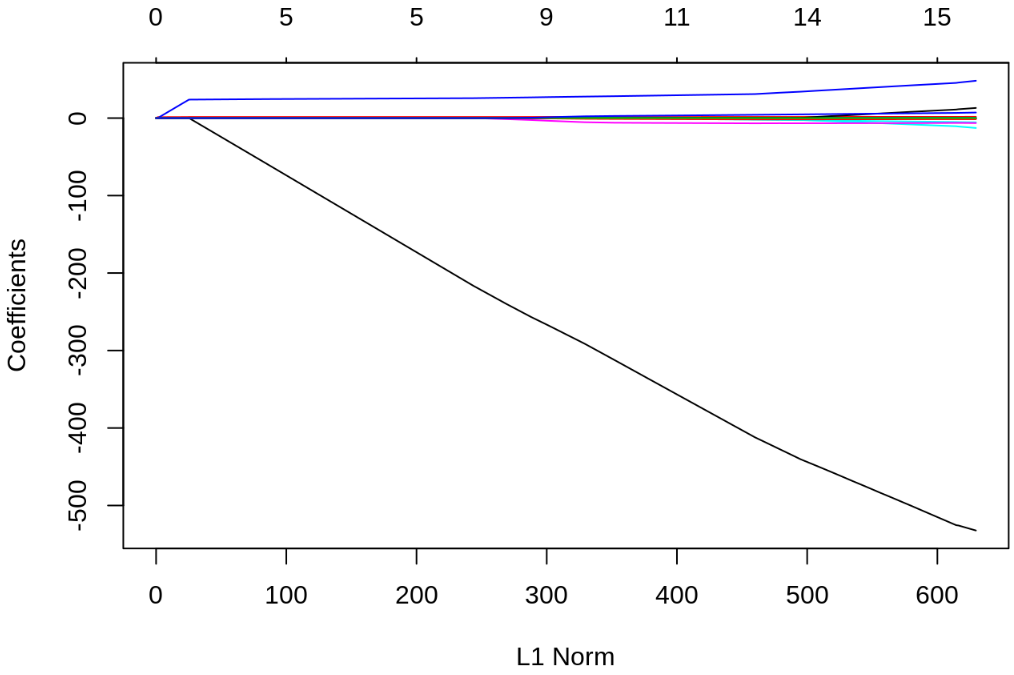
```
## [1] 230524509
```

The test RMSE for ridge regression is:

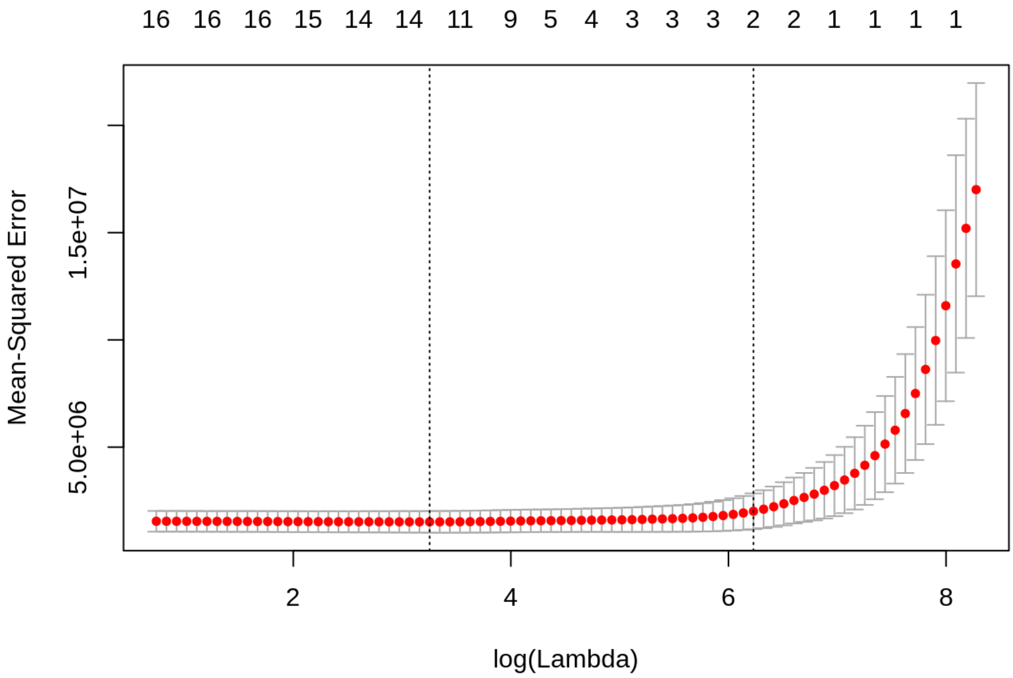
```
## [1] 994.67
```

c) The LASSO

Here is a plot that shows the elimination of coefficients in the LASSO.



Graph below shows how the model performs(MSE) on the training data with increase in lambda.



The best lambda for lasso is:

```
## [1] 25.87
```

The coefficients for the best lambda are:

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                         1
## (Intercept) -578.80716997
## Private     -411.93785668
## Accept         1.49251513
## Enroll        -0.26679903
## Top10perc     30.98948953
## Top25perc     -0.06033137
## F.Undergrad    .
## P.Undergrad    .
## Outstate      -0.05098141
## Room.Board     0.14127719
## Books          .
## Personal       .
## PhD           -2.07607487
## Terminal      -6.75788347
## S.F.Ratio      .
## perc.alumni   -1.72123995
## Expend         0.06062484
## Grad.Rate      4.24623450
```

The lasso model with the best lambda has 12 predictors.

Lasso RMSE for test data is-

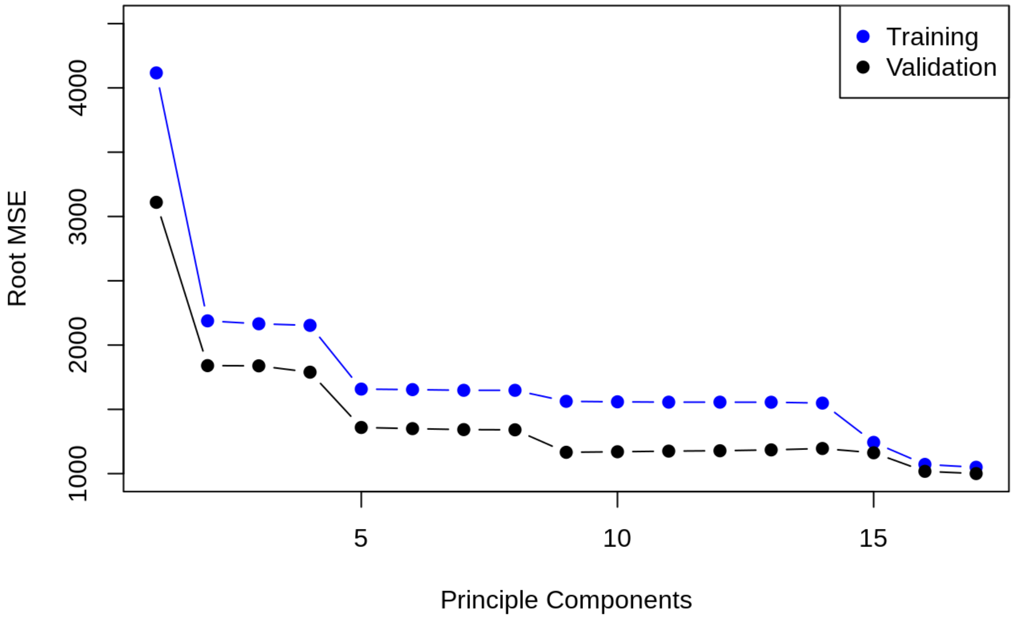
```
## [1] 1002.883
```

e) Principle component analysis

Here are the principle components.

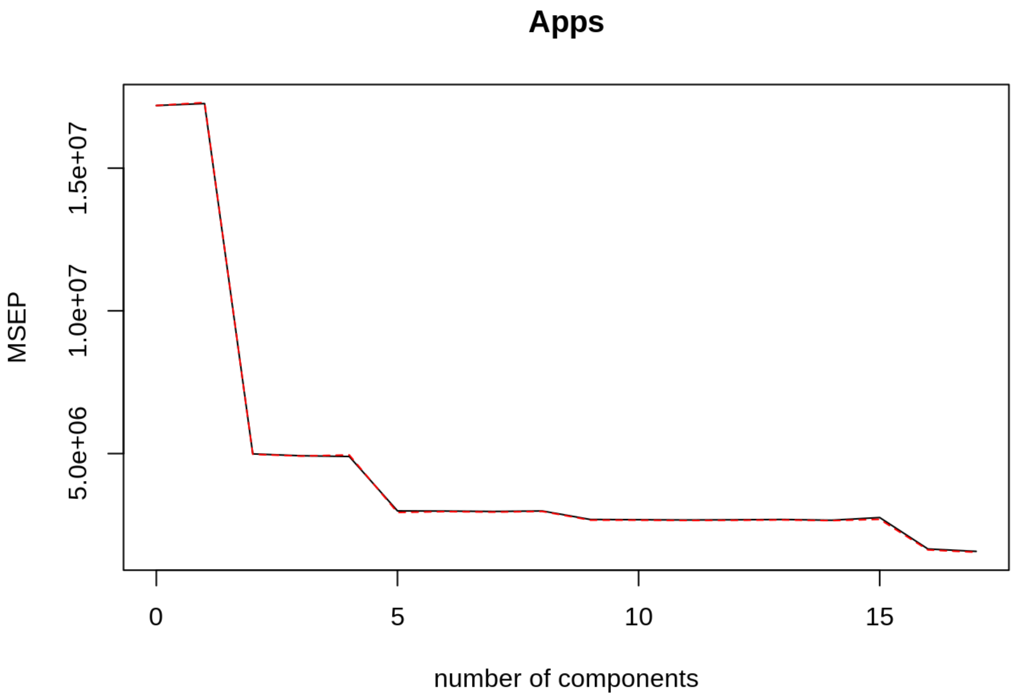
```
## Data:    X dimension: 544 17 
##  Y dimension: 544 1
## Fit method: svdpc
## Number of components considered: 17
## TRAINING: % variance explained
##       1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X      31.019    57.66    64.53    70.39    75.87    80.85    84.34
## Apps    1.072    72.05    72.63    72.94    83.96    84.04    84.14
##       8 comps  9 comps  10 comps  11 comps  12 comps  13 comps  14 comps
## X       87.72    90.51     92.89     94.95     96.77     97.89     98.72
## Apps    84.15    85.75     85.82     85.87     85.87     85.88     86.00
##       15 comps  16 comps  17 comps
## X        99.36     99.84    100.00
## Apps     90.98     93.30     93.57
```

The graph below shows the Root RMSE (both train and test) for model with each subset of principle components.



The model performs better on the test data set. The lowest error observed is for the the model with 17 components. But it is still comparable to a model with 9 principle components. It is better to choose a model with 9 principle components to allow model to generalize better.

Using cross validation to select the best PCs The plot below shows MESP as a function of number of principle components.



As seen in the graph, the fit gets better as we add more components. The graph is simular to the one without cross validation except that both are on a different scale, but their trajectory is the same. In this case too, a model with more than 9 PCs is not performing too better than the one with 9 PCs.

The test RMSE obtained using 9 components is:

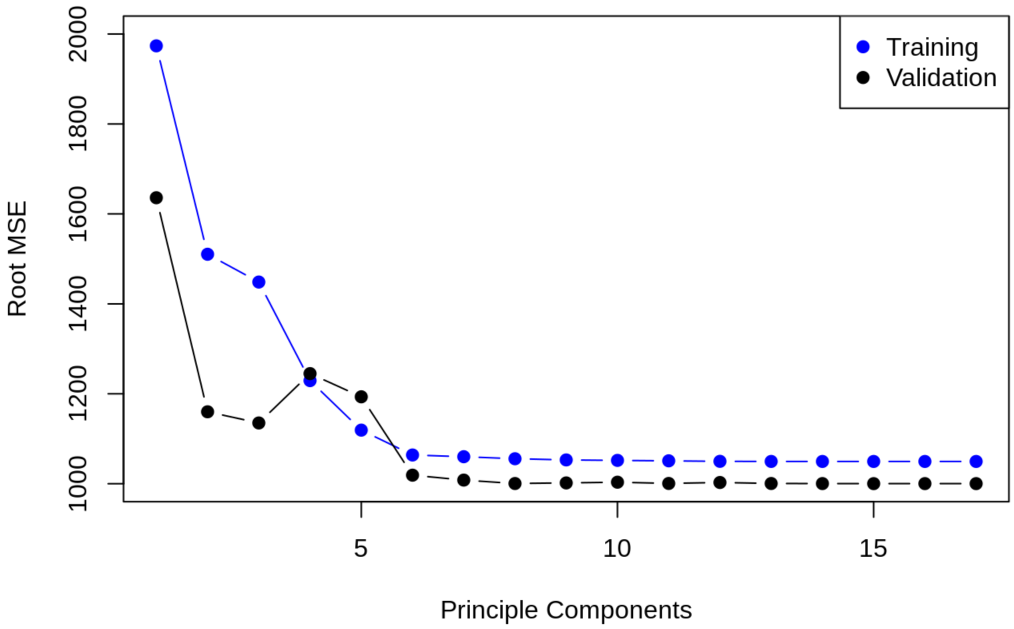
```
## [1] 1165.84
```

f) Partial Least Squares

Here are all the least square coefficients

```
## Data:     X dimension: 544 17 
##  Y dimension: 544 1
## Fit method: kernelpls
## Number of components considered: 17
## TRAINING: % variance explained
##       1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X       26.57    34.70    62.73    65.63    70.07    73.43    77.14
## Apps    77.26    86.68    87.75    91.18    92.69    93.39    93.44
##       8 comps  9 comps  10 comps  11 comps  12 comps  13 comps  14 comps
## X       80.13    83.01     86.22     89.26     90.75     92.20     93.87
## Apps    93.50    93.53     93.54     93.55     93.57     93.57     93.57
##       15 comps  16 comps  17 comps
## X        96.32     98.23    100.00
## Apps     93.57     93.57     93.57
```

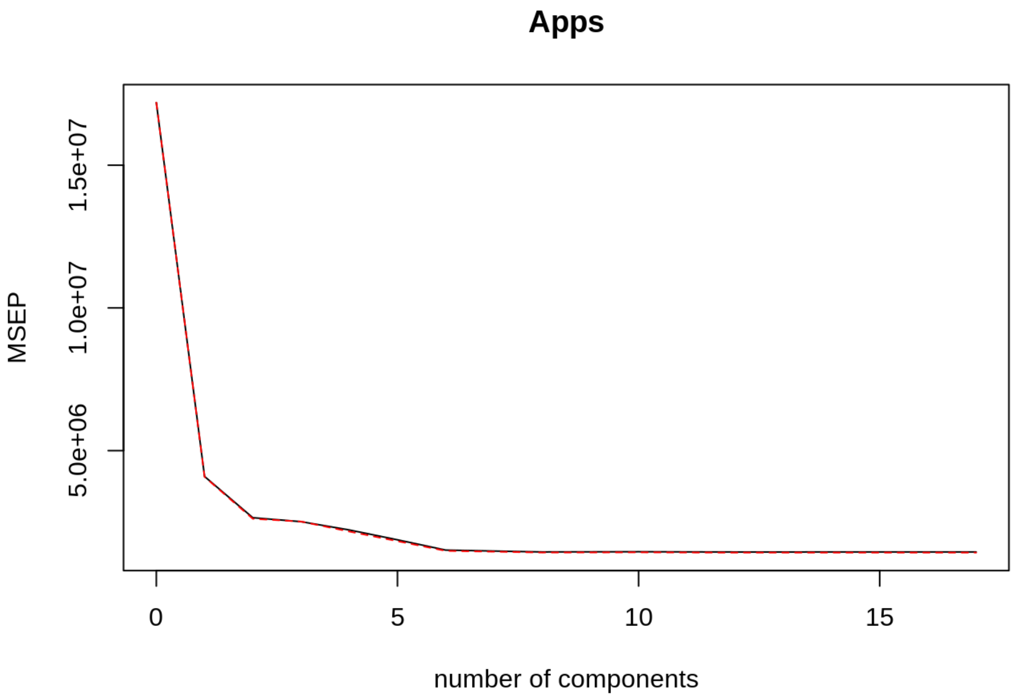
Graph below shows the test and train RMSE for each subset of the principle components.



The minimum RMSE obtained for the training data set is 1000.23, which is for the model with 15 components. The minimum does not do much better than the model with 6 principle components. The RMSE for the model with 6 PC is 1019.06. So, for PLS, these 6 principle components will be selected in the final model.

Selecting the number of principle components with cross validation

```
## Data:    X dimension: 544 17
##  Y dimension: 544 1
## Fit method: kernelpls
## Number of components considered: 17
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV            4146     2024     1628     1585     1491     1370     1231
## adjCV         4146     2020     1617     1586     1472     1353     1220
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV        1218     1204     1206      1208      1205      1204      1204
## adjCV     1208     1196     1197      1199      1196      1195      1195
##        14 comps  15 comps  16 comps  17 comps
## CV         1204      1204      1204      1204
## adjCV      1195      1195      1195      1195
##
## TRAINING: % variance explained
##       1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X       26.57    34.70    62.73    65.63    70.07    73.43    77.14
## Apps    77.26    86.68    87.75    91.18    92.69    93.39    93.44
##       8 comps  9 comps  10 comps  11 comps  12 comps  13 comps  14 comps
## X       80.13    83.01     86.22     89.26     90.75     92.20     93.87
## Apps    93.50    93.53     93.54     93.55     93.57     93.57     93.57
##       15 comps  16 comps  17 comps
## X        96.32     98.23    100.00
## Apps     93.57     93.57     93.57
```



Again the tragectory for the

PLS fit with and without cross validation is the same. The fit starts getting better after 5, at around 6. In the non-cross validation approach the value of 6 was selected as the best. After cross validation, the value of 6 seems to be the best as adding more components is not showing much improvement.

The test RMSE obtained using PLS with 6 components is:

```
## [1] 1063.94
```

Commment on the results

Following table shows the RMSEs obtained for each of the models.

	ModelNames	TestErrors
1	LS	1000.2800
2	Ridge	994.6736
3	Lasso	1002.8831
4	PCR	1165.8400
5	PLS	1063.9400

Out of all the models, ridge regression performs slightly better.

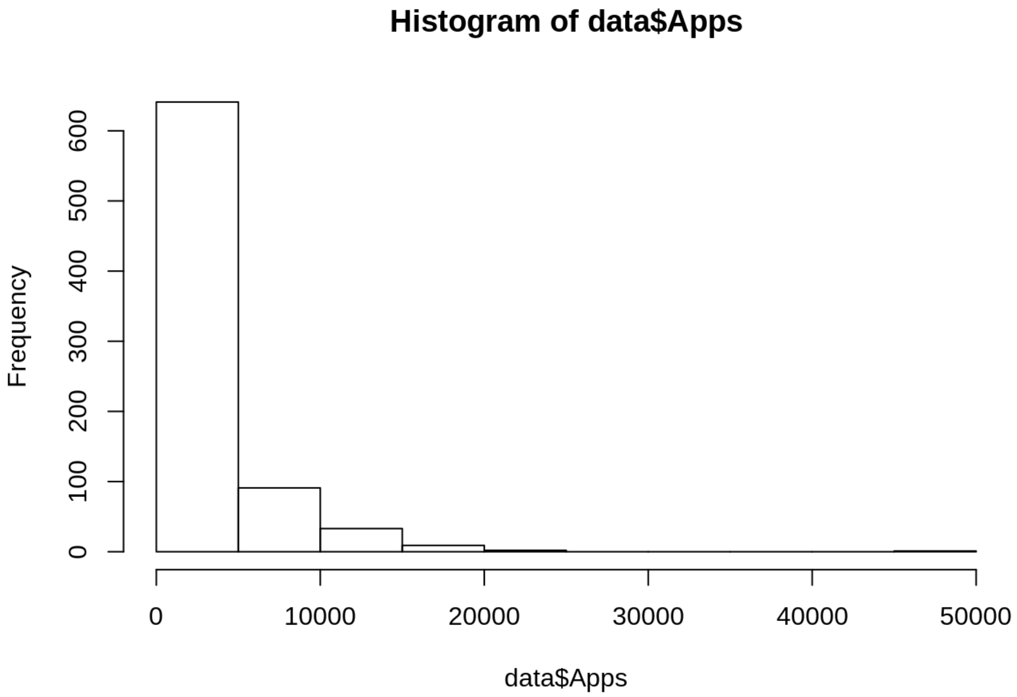
Is there much difference between the test erros obtained?

There is not much difference in the errors obtained. They all lie closely.

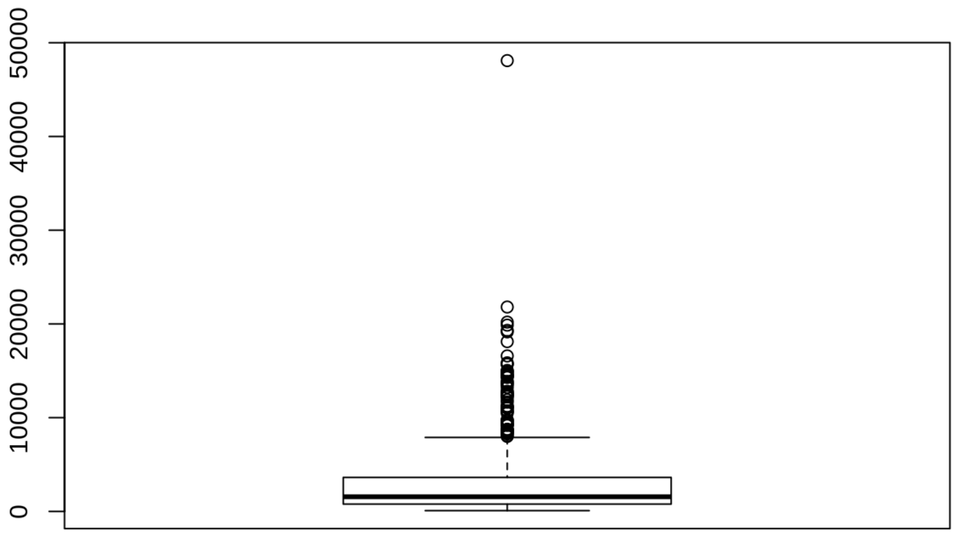
How accurately can we predict the number of college applications received?

If we select the best model of all, the Ridge it produces an error of 994. i.e it may underestimate or overestimate the number of applications received to a university. We need to figure out how significant this estimation can be in terms of the current data that we have.

Below is a histogram of the number of college applications received in the complete data. Most of the values are clustered below 10000.



Below is a box plot for the number of applications.



```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##      81     776    1558    3002    3624   48094
```

The data is very closly squished below < ~4000. Which amounts for around 75% of the data. If we assume our best, the ridge, our estimate is going to be around **25%** off from the actual value. Which may not be large for 4000, but increases as we go towards smaller value.

At 50 percentile, out estimate is going to be off by around 30%. Which again might not be too much.

In general, out model is off by around 1000 applications all the time. For universities receiving larger applications, it may be a very small value, but for universities with less number of applications, the error is large.

We could be off on both the sides. Consider an example when we have 2000 applications, the model may estimate 1000 or may even estimate 3000. The value range is 2000, which is as large as the true number of applications received!

I would say that the model is decently accurate. Especially if we are dealing with an college application predictions where lesser accuracy might be acceptabe.