Statistical Data Mining I

Homework 2

Name: Priya Murthy

UB person number: 50248887, Roll no: 53

Q1.

(a) Split the data set into a training set and a test set. Fit a linear model using least squares on the training set, and report the test error obtained.

Ans. The dataset has 777 observations of 18 variables, which is split into half, i.e Training data with 388 observations and rest as test.

****
Sample code
****

```
train = sample(1:nrow(data_set), round(nrow(data_set)/2))
test<- -train
train_data <- College[train, ]
test_data <- College[test, ]
```

After fitting the linear model on the above dataset, the test error obtained is:

```
> lm_error
[1] 1612931
```

The figure below displays the summary of the result:

```
> print(summary(result))

Call:
lm(formula = train_data$Apps ~ ., data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-2476.3  -383.9   -41.9   312.5  6055.4 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)   49.70791  463.77037   0.107  0.91470    
PrivateYes  -749.68488  157.01512  -4.775 2.60e-06 ***
Accept         1.36383    0.06504  20.968  < 2e-16 ***
Enroll        -0.10981    0.24001  -0.458  0.64755    
Top10perc     43.25219    6.42622   6.731 6.45e-11 ***
Top25perc    -11.15419    5.34626  -2.086  0.03763 *  
F.Undergrad    0.01909    0.04253   0.449  0.65386    
P.Undergrad   -0.07091    0.05791  -1.224  0.22155    
Outstate      -0.04468    0.02240  -1.994  0.04684 *  
Room.Board     0.16579    0.05707   2.905  0.00389 ** 
Books         -0.19765    0.23390  -0.845  0.39865    
Personal       0.03859    0.07713   0.500  0.61708    
PhD           -6.36068    5.13199  -1.239  0.21598    
Terminal      -4.55328    5.85725  -0.777  0.43743    
S.F.Ratio     -4.16665   14.45613  -0.288  0.77333    
perc.alumni   -6.02118    4.75232  -1.267  0.20595    
Expend         0.03902    0.01683   2.318  0.02102 *  
Grad.Rate     10.07348    3.41860   2.947  0.00342 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 855.6 on 370 degrees of freedom
Multiple R-squared:  0.9349,	Adjusted R-squared:  0.9319 
F-statistic: 312.8 on 17 and 370 DF,  p-value: < 2.2e-16
```

(b) Fit a ridge regression model on the training set, with λ chosen by cross- validation. Report the test error obtained.

```
> print(test_error)
[1] 1664412
> lambda.best
[1] 14.17474
```

- The best lambda (i.e the minimum lambda) obtained by cross validation is: 14.17474
- The test mean squared error for Ridge regression is: 1664412
- This is higher than that obtained by least squares.

(d) Fit a lasso model on the training set, with λ chosen by cross validation. Report the test error obtained, along with the number of non-zero coefficient estimates.

```
> lambda.best.lasso
[1] 5.547756
> test_error_lasso
[1] 1624526
```

- The best lambda (i.e the minimum lambda) obtained by cross validation is: 5.547756
- The test mean squared error for Ridge regression is: 1624526
- This is lower than that obtained by ridge regression.

```
> coef(lasso.mod, s =lambda.best.lasso)
19 x 1 sparse Matrix of class "dgCMatrix"
                        1
(Intercept) -910.56008627
(Intercept)    .
PrivateYes  -338.20151706
Accept         1.31398715
Enroll         .
Top10perc     41.12724127
Top25perc    -12.69964773
F.Undergrad    0.01398002
P.Undergrad    0.01948985
Outstate      -0.06120774
Room.Board     0.13266814
Books          .
Personal       .
PhD           -8.80627766
Terminal      -0.98794223
S.F.Ratio     29.97265903
perc.alumni   -1.97335727
Expend         0.10919768
Grad.Rate      5.60711000
>
```

The figure shows the coefficient estimates obtained from the lasso model.

The variables Enroll, Books and personal have zero estimates. Thus, the other predictors are the non-zero estimates obtained from the lasso model.

(e) Fit a PCR model on the training set, with k chosen by cross-validation. Report the test error obtained, along with the value of k selected by cross-validation.

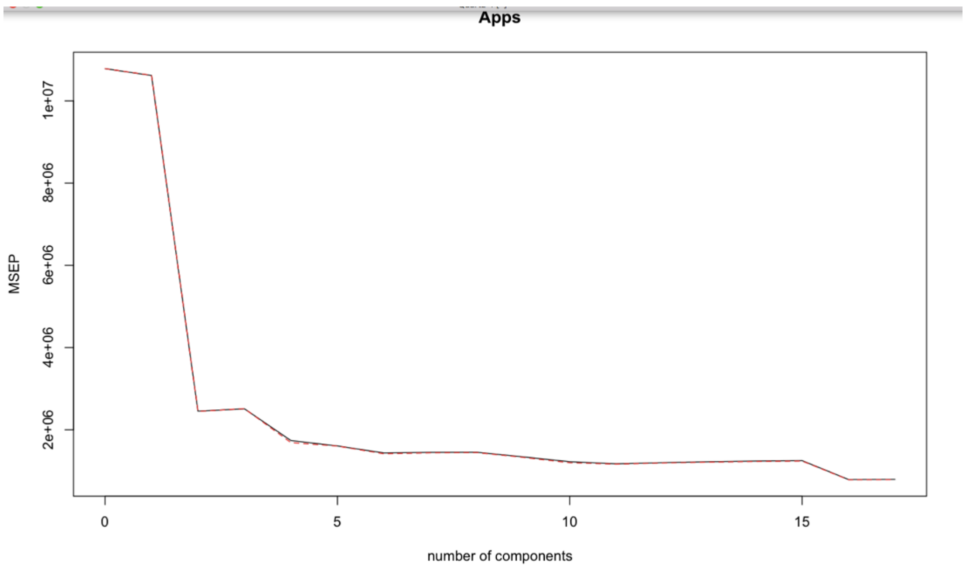


Fig: Graph of Number of components vs Mean squared error of prediction.

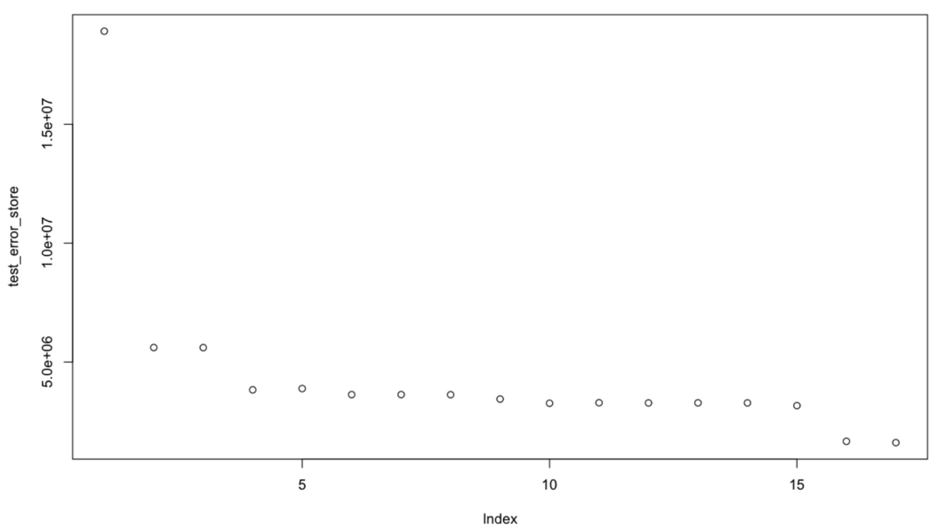


Fig: The figure shows test errors obtained n case of PCR for different values of components

The test error for different component values are:

```
> test_error_store
 [1] 18910523  5610322  5607886  3832887  3886300  3631678  3632303  3628669  3446071  3268560
[11]  3291247  3282773  3286844  3283413  3168913  1667885  1612931
```

The minimum test error obtained is 1612931 where number of components = 17

This error is same as that for least squares

(f)Fit a PLS model on the training set, with k chosen by crossvalidation. Report the test error obtained, along with the value of k selected by cross-validation.

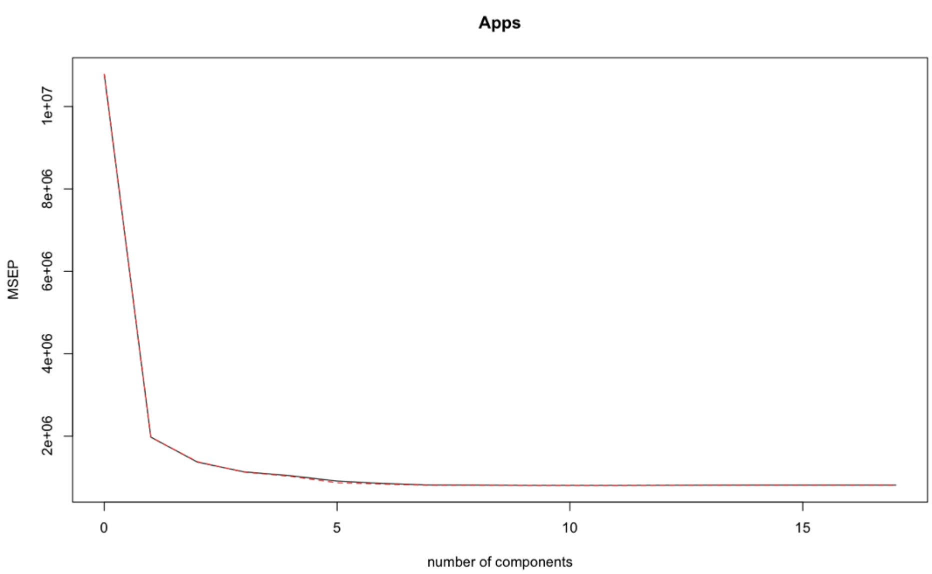


Fig: Graph of Number of components vs Mean squared error of prediction for PLS.

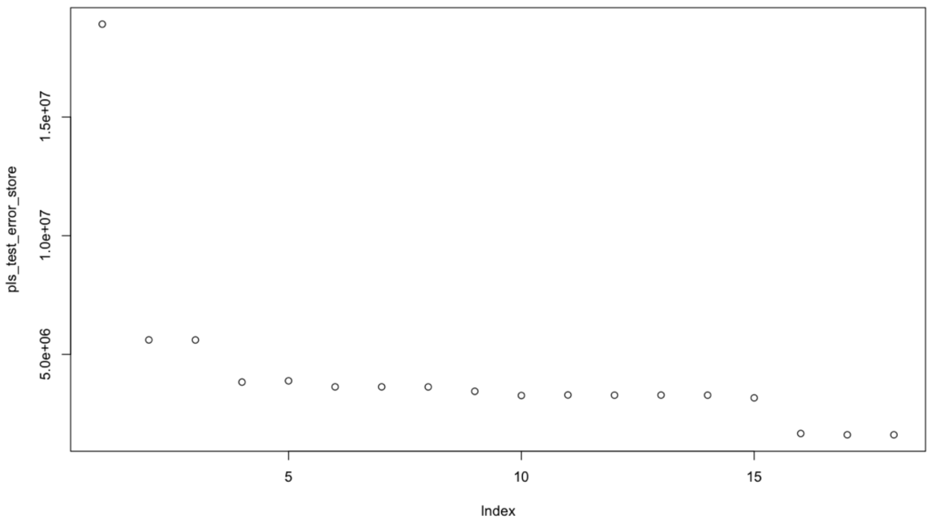


Fig: The figure shows test errors obtained n case of PCR for different values of components

```
> pls_test_error_store
 [1] 18910523  5610322  5607886  3832887  3886300  3631678  3632303  3628669  3446071  3268560
[11]  3291247  3282773  3286844  3283413  3168913  1667885  1612931  1612931
```

(g) Comment on the results obtained. How accurately can we predict the number of college applications received? Is there much difference among the test errors resulting from these five approaches?

Looking at the test errors of all the above models, we can say that even though all of the above models have a high value of test error, but Lasso and Ridge regression methods have a slightly greater error rate as compared to the others.

On computing the R2, we get:

```
> lm_r2
[1] 0.9070907
> ridge_r2
[1] 0.9029556
> lasso_r2
[1] 0.904504
> pcr_r2
[1] 0.9070907
> pls_r2
[1] 0.9070907
```

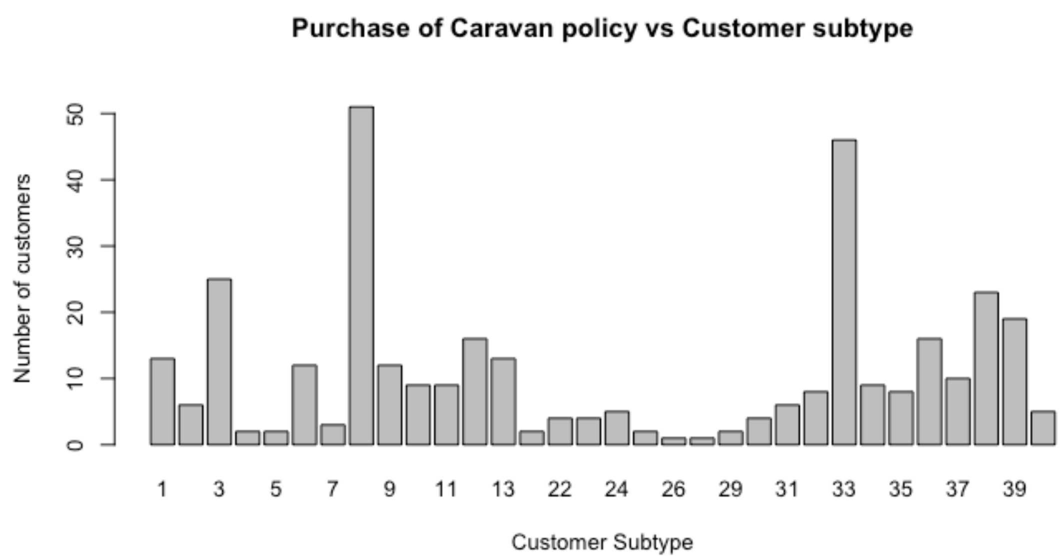
Fig: R2 values for all models.

Comparing the above values, we see that the R2 for Ridge and Lasso models is slightly less than Linear regression, Partial least squares, Principal component regression. Thus, other models have a slightly better accuracy than Lasso and Ridge regression and Ridge gives the least accuracy.

Q2.

Load the test and train data.

a) Can you predict who will be interested in buying a caravan insurance policy and give an explanation why?



Looking at the above barplot, we can say that the customers who belong to subtype 8(Middle class families) & 33(lower class with large families) have maximum number of customers who have purchased the Caravan policy.

We can also compare the number of customer who have purchased Caravan policy with the number of customers who have purchased other policies instead of caravan policy. Let’s look at the OLS estimate of the training data to compare the relationship between the response variable and other variables.

Let’s compare the OLS estimates and compare:

```
Residuals:
     Min       1Q   Median       3Q      Max 
-0.67293 -0.08720 -0.04593 -0.00639  1.04628 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)  
(Intercept)  0.7685381  0.4298406   1.788 0.073835 .
V1           0.0035209  0.0022512   1.564 0.117866  
V2          -0.0072642  0.0076739  -0.947 0.343875  
V3          -0.0012739  0.0071737  -0.178 0.859055  
V4           0.0107473  0.0049596   2.167 0.030279 *
V5          -0.0154869  0.0101044  -1.533 0.125405  
V6          -0.0056016  0.0056016  -1.000 0.317353  
V7          -0.0002069  0.0060664  -0.034 0.972795  
V8           0.0003569  0.0054592   0.065 0.947874  
V9          -0.0030237  0.0058038  -0.521 0.602399  
V10          0.0086829  0.0075479   1.150 0.250036  
V11          0.0020367  0.0072008   0.283 0.777310  
V12          0.0055682  0.0076295   0.730 0.465526  
V13         -0.0038250  0.0065474  -0.584 0.559107  
V14         -0.0050625  0.0066861  -0.757 0.448980  
V15         -0.0026253  0.0069795  -0.376 0.706824  
V16          0.0021357  0.0068161   0.313 0.754038  
V17         -0.0048456  0.0071396  -0.679 0.497358  
V18         -0.0113977  0.0073004  -1.561 0.118525  
V19          0.0021884  0.0045182   0.484 0.628153  
V20         -0.0004665  0.0052201  -0.089 0.928796  
V21         -0.0050974  0.0050426  -1.011 0.312122  
V22          0.0041254  0.0044806   0.921 0.357228  
V23         -0.0006060  0.0044709  -0.136 0.892190  
V24          0.0019733  0.0044532   0.443 0.657690  
V25         -0.0013674  0.0051653  -0.265 0.791225  
V26         -0.0031701  0.0050198  -0.632 0.527724  
V27         -0.0012603  0.0044827  -0.281 0.778603  
V28          0.0024879  0.0049115   0.507 0.612502  
V29         -0.0008866  0.0047145  -0.188 0.850832  
V30         -0.0454201  0.0376622  -1.206 0.227872  
V31         -0.0432242  0.0376290  -1.149 0.250730  
V32          0.0085964  0.0075592   1.137 0.255502  
V33          0.0077871  0.0068554   1.136 0.256038  
```

V34	0.0047215	0.0072646	0.650	0.515762	
V35	-0.0561024	0.0444643	-1.262	0.207094	
V36	-0.0593733	0.0443897	-1.338	0.181097	
V37	0.0070879	0.0051150	1.386	0.165884	
V38	0.0069414	0.0049276	1.409	0.158986	
V39	0.0049679	0.0050144	0.991	0.321862	
V40	0.0059267	0.0052728	1.124	0.261053	
V41	-0.0098939	0.0069270	-1.428	0.153258	
V42	0.0063044	0.0045645	1.381	0.167277	
V43	0.0029097	0.0022664	1.284	0.199250	
V44	0.0284931	0.0166017	1.716	0.086166	.
V45	-0.0101533	0.0205121	-0.495	0.620625	
V46	-0.0201220	0.0390424	-0.515	0.606301	
V47	0.0102787	0.0026346	3.901	9.67e-05	**
V48	0.0014405	0.0148574	0.097	0.922765	
V49	-0.0061279	0.0079415	-0.772	0.440364	
V50	-0.0249190	0.0415892	-0.599	0.549083	
V51	0.0588044	0.0557610	1.055	0.291662	
V52	0.0121481	0.0142358	0.853	0.393504	
V53	-0.0062440	0.0370186	-0.169	0.866060	
V54	0.0078683	0.0152793	0.515	0.606598	
V55	-0.0155397	0.0064753	-2.400	0.016433	*
V56	0.0098926	0.0335157	0.295	0.767880	
V57	0.1937254	0.0793370	2.442	0.014644	*
V58	0.0647933	0.0256913	2.522	0.011696	*
V59	0.0132643	0.0035906	3.694	0.000223	**
V60	-0.1917507	0.1439848	-1.332	0.182998	
V61	-0.0299076	0.0269224	-1.111	0.266666	
V62	-0.0107777	0.0549693	-0.196	0.844564	
V63	-0.0441620	0.0307404	-1.437	0.150883	
V64	-0.0184858	0.0288890	-0.640	0.522269	
V65	-0.0377952	0.0323794	-1.167	0.243154	
V66	0.0185448	0.0529740	0.350	0.726296	
V67	0.0180904	0.1374585	0.132	0.895300	
V68	0.0002821	0.0127496	0.022	0.982347	
V69	-0.0214816	0.0652955	-0.329	0.742175	
V70	0.0203252	0.0310683	0.654	0.513004	
V71	0.0563675	0.1589388	0.355	0.722866	
V72	-0.0804238	0.0944352	-0.852	0.394455	
V73	-0.0395651	0.0353795	-1.118	0.263484	

```
V73         -0.0395651  0.0353795  -1.118 0.263484
V74         -0.0010526  0.0728240  -0.014 0.988468
V75         -0.0236462  0.0467611  -0.506 0.613101
V76          0.0372344  0.0154024   2.417 0.015661 *
V77         -0.0464279  0.0954471  -0.486 0.626684
V78         -0.4050642  0.1898715  -2.133 0.032938 *
V79         -0.2304561  0.1243310  -1.854 0.063852 .
V80         -0.0211374  0.0116048  -1.821 0.068593 .
V81          0.4958051  0.2815591   1.761 0.078304 .
V82          0.3633887  0.0885318   4.105 4.11e-05 ***
V83          0.0416061  0.0408644   1.018 0.308650
V84          0.0959436  0.0699079   1.372 0.169983
V85          0.1312250  0.0983836   1.334 0.182319
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.23 on 5736 degrees of freedom
Multiple R-squared:  0.0729,    Adjusted R-squared:  0.05916
F-statistic: 5.306 on 85 and 5736 DF,  p-value: < 2.2e-16
```

Looking the at OLS estimates of all 86 variables, we can say that the predictors

V82(APLEZIER Number of boat policies)

V46 PWALAND Contribution third party insurane (agriculture)

V59 PBRAND Contribution fire policies

have the highest impact on the value of the response (i.e purchase of caravan policy) and the variables

V76 ALEVEN Number of life insurances

V78 AGEZONG Number of family accidents insurance policies

V55 PLEVEN Contribution life insurances

V57 PGEZONG Contribution family accidents insurance policies

V58 PWAOREG Contribution disability insurance policies

V4 MGEMLEEF Avg age see L1

have a good impact on the output response.

Thus, if we deduce the relationship between these variables and the output response. We can predict who will be interested in buying a caravan insurance policy.

Let’s take an example:

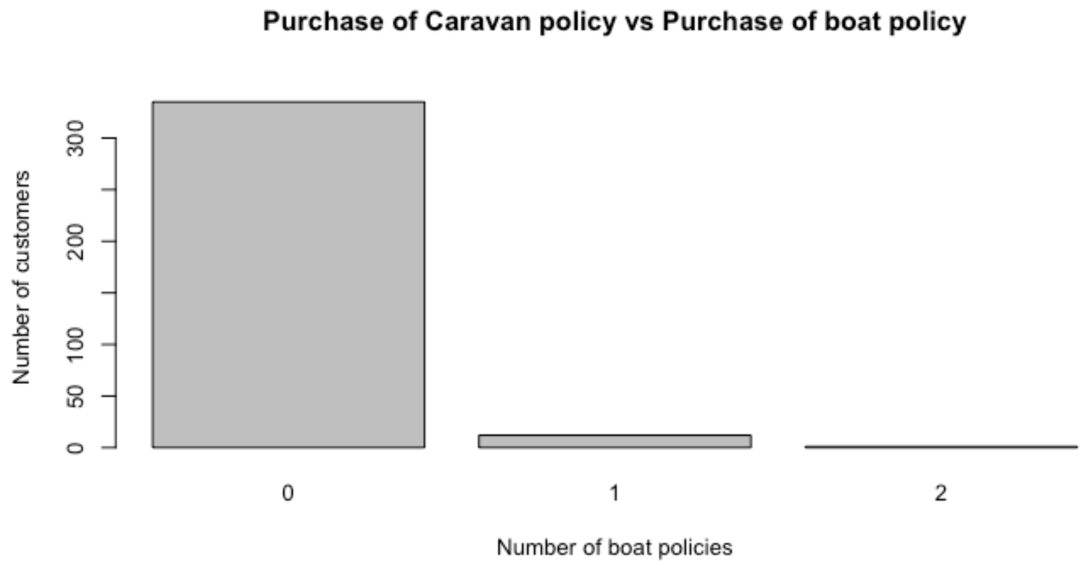


Fig: Shows relationship between customers who purchased caravan policy vs customers who purchased boat policy.

From this we can conclude that customers who purchased boat policy did not purchase caravan policy. Thus, using similar relationships between the predictors we can predict who will be interested in buying a caravan insurance policy.

Linear Regression:

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

The error for least squares estimate is 0.053985

For **Forward Selection:**

```
> which.min(err_vals_test)
[1] 27
> min.err_vals_test <- sort(err_vals_test)[1]
> print(min.err_vals_test)
[1] 0.05385551
> coef(regfit.fwd, which.min(err_vals_test))
 (Intercept)           V4           V7          V10          V16          V18          V21
 0.558970194  0.011027882  0.002804163  0.004425560  0.006334498 -0.005726973 -0.007010769
         V22          V28          V30          V35          V36          V41          V42
 0.002920793  0.003268096 -0.001756812 -0.070961501 -0.073968707 -0.014300904  0.005234995
         V43          V44          V46          V47          V57          V58          V59
 0.002746762  0.010505371 -0.015755346  0.010334210  0.193254633  0.063028861  0.012577249
         V78          V79          V80          V81          V82          V83          V85
-0.409800846 -0.224589385 -0.020748342  0.175792387  0.278146326  0.037369540  0.070002221
```

The coefficients of the best model are shown in the figure above and the minimum error value for a model with 27 coefficients.

The MSE for forward selection is 0.05385551

For Backward selection:

```
> which.min(err_vals_test_bwd)
[1] 38
> min.err_vals_test_bwd <- sort(err_vals_test_bwd)[1]
> print(min.err_vals_test_bwd)
[1] 0.05383966
> coef(regfit.bwd, which.min(err_vals_test_bwd))
 (Intercept)           V1           V4           V5           V6           V9          V10
 0.604372535  0.003346395  0.011740595 -0.014676301 -0.005077877 -0.002333848  0.004979876
         V14          V17          V18          V21          V22          V28          V30
-0.002376406 -0.006476407 -0.012747058 -0.006230546  0.002907774  0.003371689 -0.001896290
         V35          V36          V41          V42          V43          V44          V46
-0.066139439 -0.068717745 -0.012874284  0.005580509  0.003222598  0.029403224 -0.016117545
         V47          V55          V57          V58          V59          V60          V63
 0.010389136 -0.016792246  0.195801656  0.063077120  0.012805158 -0.183106675 -0.042933275
         V65          V69          V76          V78          V79          V80          V81
-0.039215986 -0.026621172  0.039467026 -0.412869084 -0.226110389 -0.020237344  0.464696488
         V82          V83          V84          V85
 0.275011459  0.034877493  0.092711041  0.071796086
```

The coefficients of the best model are shown in the figure above and the minimum error value for a model with 38 coefficients.

The MSE for forward selection is 0.05383966

For Ridge Regression:

```
> mean((y_hat - y_true)^2)
[1] 0.05369642
> test_error
[1] 214.7857
```

The test error is 214.7857 and the mean square of the predicted minus the true value of response is 0.05369642

For Lasso Regression:

```
> mean((y_hat_lasso - y_true)^2)
[1] 0.05374687
> test_error_lasso
[1] 214.9875
```

The test error is 214.9875 and the mean square of the predicted minus the true value of response is 0.05374687.

Lasso: 0.05374687, Ridge 0.05369642, Backward 0.05383966, Forward: 0.05385551, Linear Regression: 0.053985.

Looking at the above mean values: we can see that even though all of them have similar error values, Ridge regression has the least difference between predicted and expected output values.

The output from OLS estimates are almost similar to those obtained by other models, like output of forward subset selection also has no. of boat policies with high rate.

Q3.

a) To generate dataset and split into training and test data:

- In this problem, we first generate the dataset and the response vector
- The dataset has 20 features and 1000 observations
- Then, the dataset is split into training and test data

b) (i) Perform best subset selection on the training set.

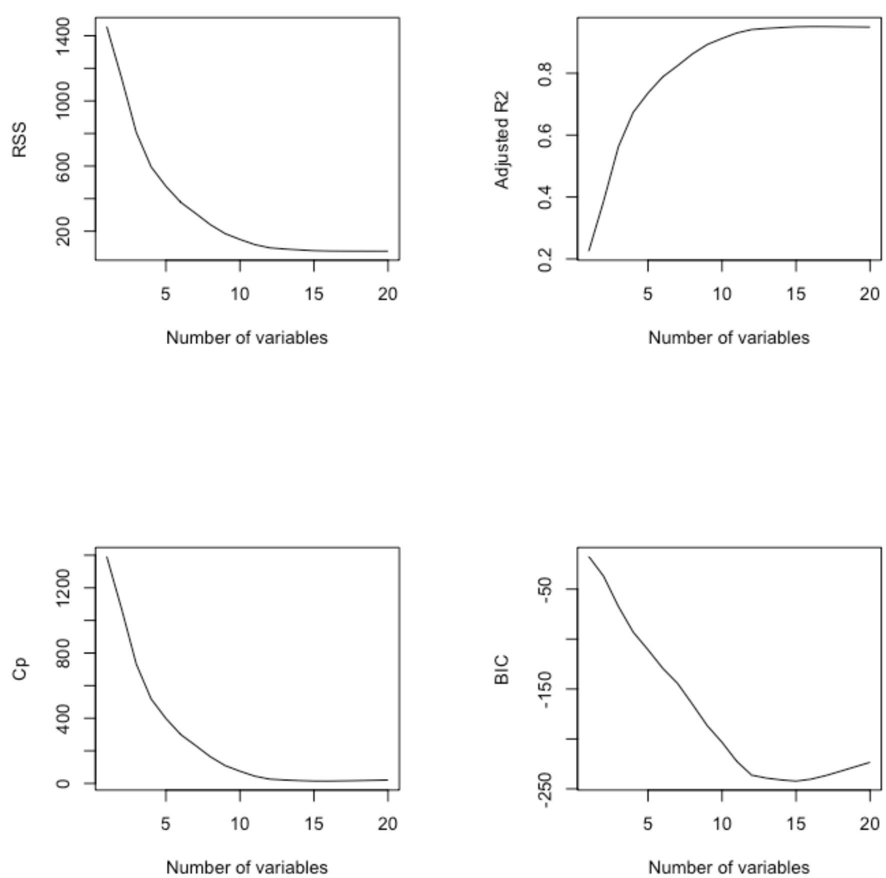


FIG:

The figure above represents the relationship between number of variables vs (RSS, Adjusted R2, Cp and BIC) obtained from subset selection

We can see that as the number of variables increase, there is a decrease in the values of RSS, Cp, BIC and the Ajdusted R2 increases with the increase in number of variables.

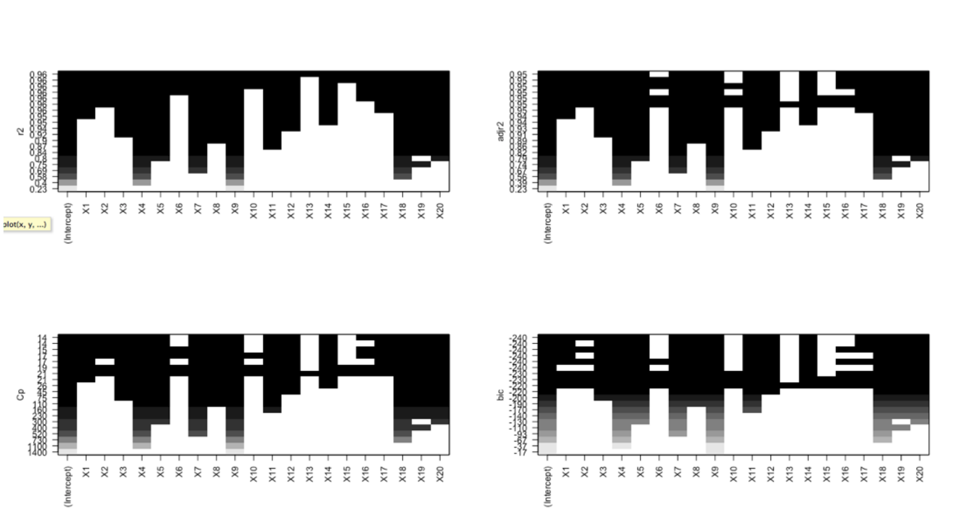


Fig:

This figure represents the values of output variables obtained from subset selection with respect to the features.

```
> summary((my_sum)$outmat)
 X1      X2      X3      X4      X5      X6      X7      X8      X9      X10     X11     X12     X13
  :12     :14     : 9     : 1     : 5     :16     : 3     : 8    *:20     :17     : 7     :10     :19
 *: 8    *: 6    *:11    *:19    *:15    *: 4    *:17    *:12            *: 3    *:13    *:10    *: 1
 X14     X15     X16     X17     X18     X19     X20
  :11     :18     :15     :13     : 2     : 5     : 5
 *: 9    *: 2    *: 5    *: 7    *:18    *:15    *:15
```

Fig: Displays the summary of the output matrix obtained from subset selection

(ii) Plot the training set MSE associated with the best model of each size

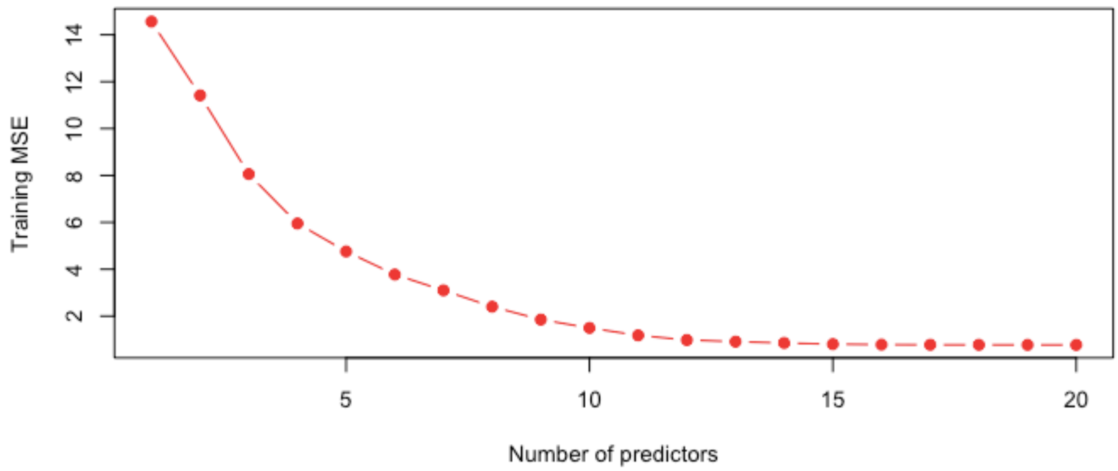


Fig: Number of predictors vs Training MSE

The Training error is minimum for a model with all 20 predictors.

```
> which.min(err_vals)
[1] 20
> coef(regfit.full, which.min(err_vals))
(Intercept)          X1          X2          X3          X4          X5          X6
 0.10637120  0.33980878  0.21080792 -0.64522317 -2.15774487  0.96154034  0.10893179
         X7          X8          X9         X10         X11         X12         X13
-1.40066885  0.69505684  2.14323749  0.03874903  0.93292274  0.66923064  0.04223371
        X14         X15         X16         X17         X18         X19         X20
-0.48606666 -0.04169437  0.16833862  0.29427798  1.84979217  1.06035555 -1.19398469
```

Fig: Coefficients of model with minimum training error.

c)

Plot the test set MSE associated with the best model of each size. For which model size does the test set MSE take on its minimum value?

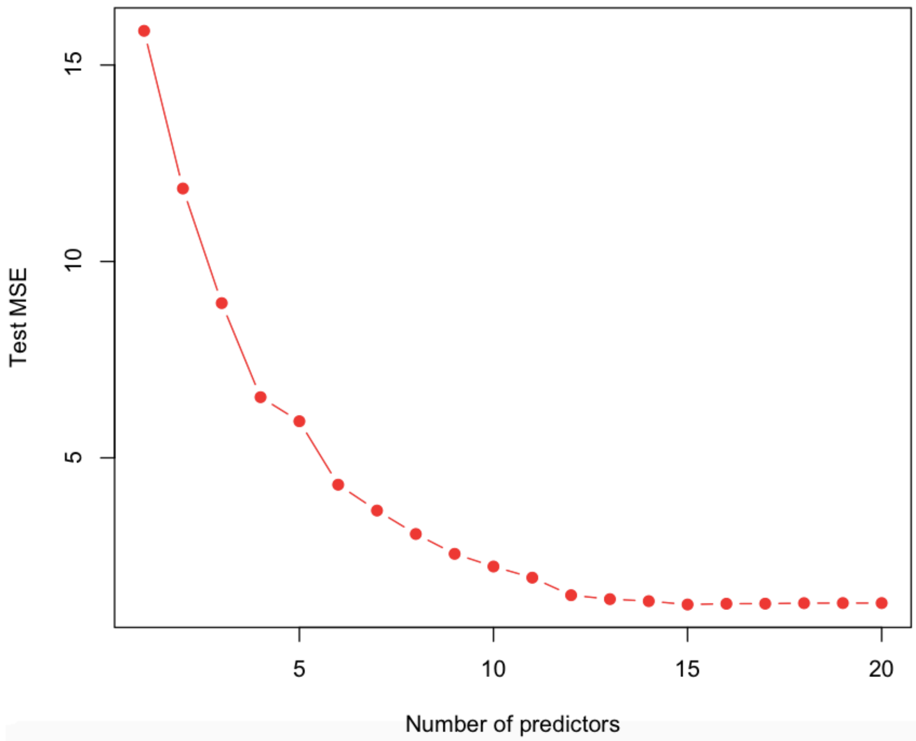


Fig: Number of predictors vs Test MSE

The Test error is minimum for a model with 15 predictors.

```
> which.min(err_vals_test)
[1] 15
> coef(regfit.full, which.min(err_vals_test))
(Intercept)          X1          X2          X3          X4          X5          X7
  0.1109564   0.3402539   0.2184820  -0.6665373  -2.1758910   0.9714736  -1.4065895
         X8          X9         X11         X12         X14         X17         X18
  0.6988052   2.1317413   0.9588559   0.6674565  -0.4614519   0.3022537   1.8609001
        X19         X20
  1.0768225  -1.1648164
```

Fig: Coefficients of model with minimum test error.

d) How does the model at which the test set MSE is minimized compare to the true model used to generate the data?

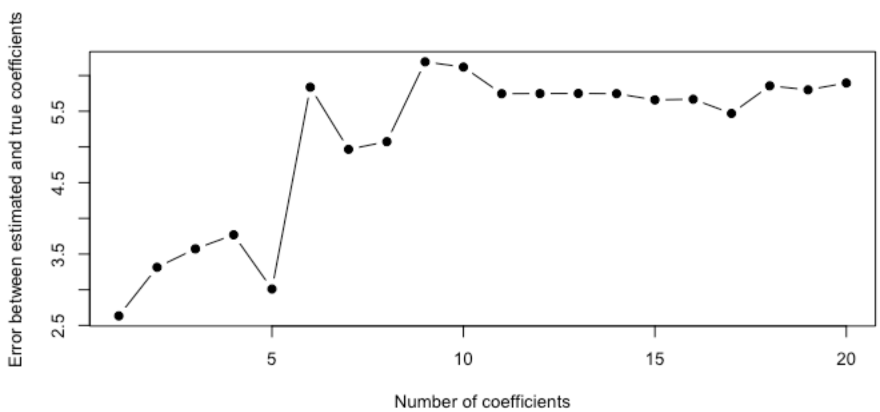


Fig: Shows the relationship between number of coefficients and the error between estimated and true coefficients.

```
> val.errors
 [1] 2.633729 3.314138 3.573010 3.769028 3.010304 5.837190 4.966600 5.074600 6.192329 6.119005
[11] 5.746335 5.748438 5.749724 5.746680 5.659219 5.668330 5.469528 5.856108 5.800627 5.896304
```

From the above fig and errors, we can see that the models with 1-5 variables minimize the error between true and estimated coefficients. But the model with 15 predictors has minimum test error. Thus, low value of test MSE doesn't mean that the error between true and estimated coefficients will be less.