# Statistical Data Mining Project 2

## Question 1

a) In this problem we had to predict the number of applications received i.e. the response variable, using the other variables in the college data set in the ISLR package. First pre-processing the data was performed and the data was checked for empty data values. As well, from reading the dataset we can see there the private predictor was a yes and no and thus was converted to categorical predictor. After the data set was split into a training set and a test set, using 30% of data for testing and 70% for training.

b) A linear model using least squares was fitted on the training set and the summary of the fitted model is shown in Table 1.

**Table 1**: Summary of the linear model fitted on the training college data set

```
lm(formula = Apps ~ ., data = collegetrain)

Residuals:
    Min      1Q  Median      3Q     Max 
-2829.4  -425.8   -56.7   296.0  6608.2 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 238.57242  545.66406   0.437 0.662135    
Private    -775.33261  164.72802  -4.707 3.22e-06 ***
Accept        1.21883    0.05741  21.230  < 2e-16 ***
Enroll       -0.23479    0.21711  -1.081 0.279998    
Top10perc    42.54908    6.45766   6.589 1.08e-10 ***
Top25perc   -10.69497    5.15890  -2.073 0.038649 *  
F.Undergrad   0.08097    0.03561   2.274 0.023382 *  
P.Undergrad   0.02698    0.03699   0.730 0.465966    
Outstate     -0.05333    0.02317  -2.301 0.021760 *  
Room.Board    0.16495    0.05644   2.922 0.003623 ** 
Books         0.06047    0.26387   0.229 0.818825    
Personal      0.06421    0.07936   0.809 0.418861    
PhD          -3.13442    5.22584  -0.600 0.548903    
Terminal    -10.01200    5.70339  -1.755 0.079765 .  
S.F.Ratio     2.62252   14.65469   0.179 0.858043    
perc.alumni  -5.71431    4.85971  -1.176 0.240185    
Expend        0.09903    0.01589   6.231 9.53e-10 ***
Grad.Rate    12.36077    3.35370   3.686 0.000252 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 998.7 on 526 degrees of freedom
Multiple R-squared:  0.9216,	Adjusted R-squared:  0.9191 
F-statistic: 363.7 on 17 and 526 DF,  p-value: < 2.2e-16
```

The mean squared error for the test data was calculated to be 1740793.

c) A ridge regression model was fitted on the training set, with λ chosen by cross-validation and summary is shown in table 2.

**Table 2**: Summary of the ridge regression model fitted on the training college data set

| | Length | Class | Mode |
|---|---|---|---|
| a0 | 100 | -none- | numeric |
| beta | 1700 | dgCMatrix | S4 |
| df | 100 | -none- | numeric |
| dim | 2 | -none- | numeric |
| lambda | 100 | -none- | numeric |
| dev.ratio | 100 | -none- | numeric |
| nulldev | 1 | -none- | numeric |
| npasses | 1 | -none- | numeric |
| jerr | 1 | -none- | numeric |
| offset | 1 | -none- | logical |
| call | 4 | -none- | call |
| nobs | 1 | -none- | numeric |

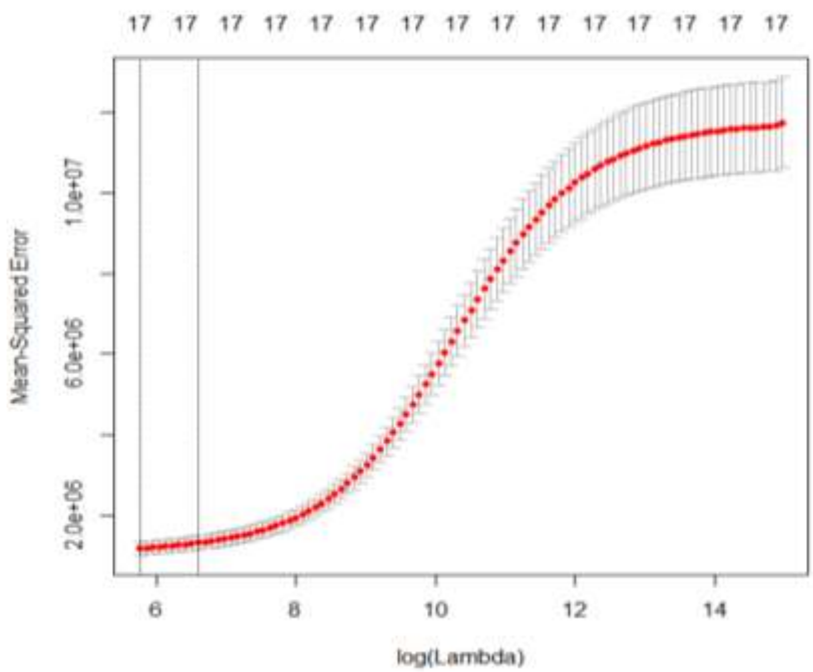


**Figure 1:** A CV output for Ridge Regression model fitted on the training college data

The best lamda selected for the cross validation was 327.
The mean squared error for the test data was calculated to be 3041114, which is higher than the linear fitted model.

d) The lasso model was fitted on the training set, with λ chosen by cross validation and summary was produced as shown in table 3.

**Table 3**: Summary of the lasso model fitted on the training college data set

| | Length | Class | Mode |
|---|---|---|---|
| a0 | 81 | -none- | numeric |
| beta | 1377 | dgCMatrix | S4 |
| df | 81 | -none- | numeric |
| dim | 2 | -none- | numeric |
| lambda | 81 | -none- | numeric |
| dev.ratio | 81 | -none- | numeric |
| nulldev | 1 | -none- | numeric |
| npasses | 1 | -none- | numeric |
| jerr | 1 | -none- | numeric |
| offset | 1 | -none- | logical |
| call | 4 | -none- | call |
| nobs | 1 | -none- | numeric |

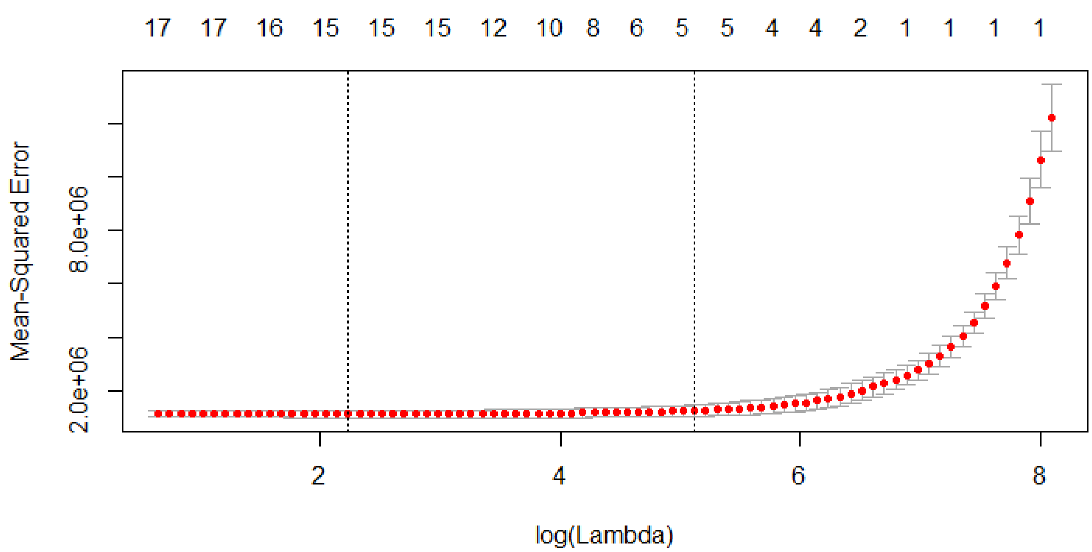


**Figure 2:** A CV output plot for the lasso model fitted on the training college data

The best lamda selected for the cross validation was 9.31755.

The mean squared error for the test data was calculated with MSE=1816608, which is higher than the linear fitted model but lower than the ridge regression model fit.

The number of non-zero coefficient estimates for the lasso model was 10, which can be seen in Table 4.

**Table 4:** The non-zero coefficient estimates for the lasso model

| (Intercept) | Private | Accept | Top10perc | Top25perc | F.Undergrad | P.Undergrad | Outstate |
|---|---|---|---|---|---|---|---|
| 140.36717355 | -756.36873095 | 1.17746233 | 36.09349819 | -5.65992984 | 0.05580209 | 0.01948605 | -0.04070817 |
| Room.Board | Books | Personal | | | | | |
| 0.14778773 | 0.04126430 | 0.05188711 | | | | | |

e) A PCR model was fitted on the training set, with k chosen by cross-validation and a summary was produced as shown in table 5, and a validation plot as shown in figure 3.

**Table 5**: Summary of the PCR model fitted on the training college data set

```
Data:   X dimension: 544 17
        Y dimension: 544 1
Fit method: svdpc
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  7 comps  8 comps  9 comps  10 comps
CV            3514     3513     1721     1708     1390     1299     1270     1234     1209     1195      1197
adjCV         3514     3513     1718     1713     1386     1283     1266     1233     1205     1192      1194
       11 comps  12 comps  13 comps  14 comps  15 comps  16 comps  17 comps
CV         1188      1188      1193      1196      1208      1051      1040
adjCV      1186      1186      1190      1193      1207      1048      1037

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
X     31.8106    57.40    64.08    69.93    75.19    80.22    84.02    87.55    90.67     93.11     95.14
Apps   0.5794    76.64    77.24    85.07    87.46    87.56    88.33    88.85    89.18     89.22     89.41
      12 comps  13 comps  14 comps  15 comps  16 comps  17 comps
X        96.95     98.00     98.86     99.39     99.83    100.00
Apps     89.46     89.47     89.51     89.54     91.82     92.16
```

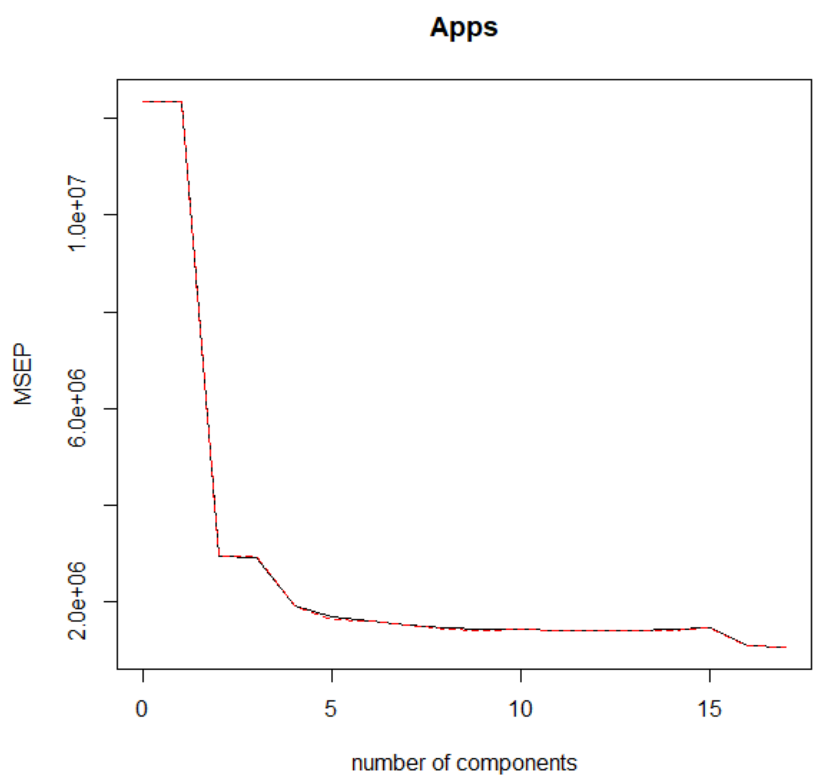


**Figure 3:** A validation plot for PCR model showing the MSEP for different number of components

From table 5 and figure 3 we can see that the global minima is the full model with 17 components, however, the local minima is 12 components.

The mean squared error for the test data was calculated to be 1740793, which is approximately similar to the linear fitted model.

f) A PLS model was fitted on the training set, with k chosen by cross-validation and a summary was produced as shown in table 6, and a validation plot as shown in figure 4.

**Table 6**: Summary of the PLS model fitted on the training college data set

```
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  7 comps  8 comps  9 comps  10 comps
CV            3514     1537     1199     1163     1142     1102     1068     1052     1052     1050      1047
adjCV         3514     1535     1193     1161     1137     1094     1063     1048     1048     1047      1044
       11 comps  12 comps  13 comps  14 comps  15 comps  16 comps  17 comps
CV         1051      1050      1050      1050      1050      1050      1050
adjCV      1048      1046      1046      1046      1046      1046      1046

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
X       25.51    34.63    62.83    66.57    69.80    73.56    77.21    81.02    83.27     85.14     87.90
Apps    81.56    88.94    89.77    90.64    91.51    91.96    92.06    92.08    92.11     92.14     92.15
      12 comps  13 comps  14 comps  15 comps  16 comps  17 comps
X        90.76     93.56     96.06     97.63     99.18    100.00
Apps     92.16     92.16     92.16     92.16     92.16     92.16
```

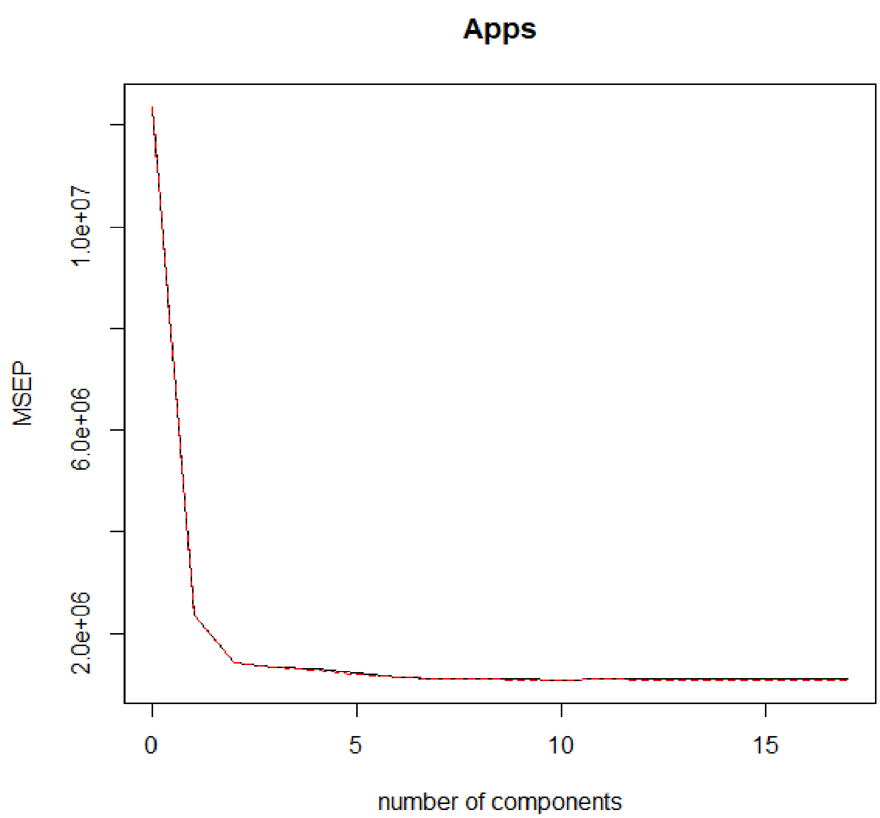


**Figure 4:** A validation plot for PLS model showing the MSEP for different number of components

From table 6 and figure 4 we can see that the minima is 10 and thus we should keep 10 components.
The mean squared error for the test data was calculated to be 1750013.

g) There is a difference among the test errors resulting from these five approaches. The linear model and PCR model have the lowest test error, followed by PLS, LASSO, and the Ridge model. To determine if we can accurately predict the number of college applications received the R-squared error for each model was determined as shown in figure 5. Based on the graph we can see that all models have a high R-squared value, this means the models fit our observations well. With the OLS and PCR having the largest R^2 of 0.917, indicating around 91.7% of the data is close to the fitted regression line. Thus, we can use the model to predict the number of applications received.

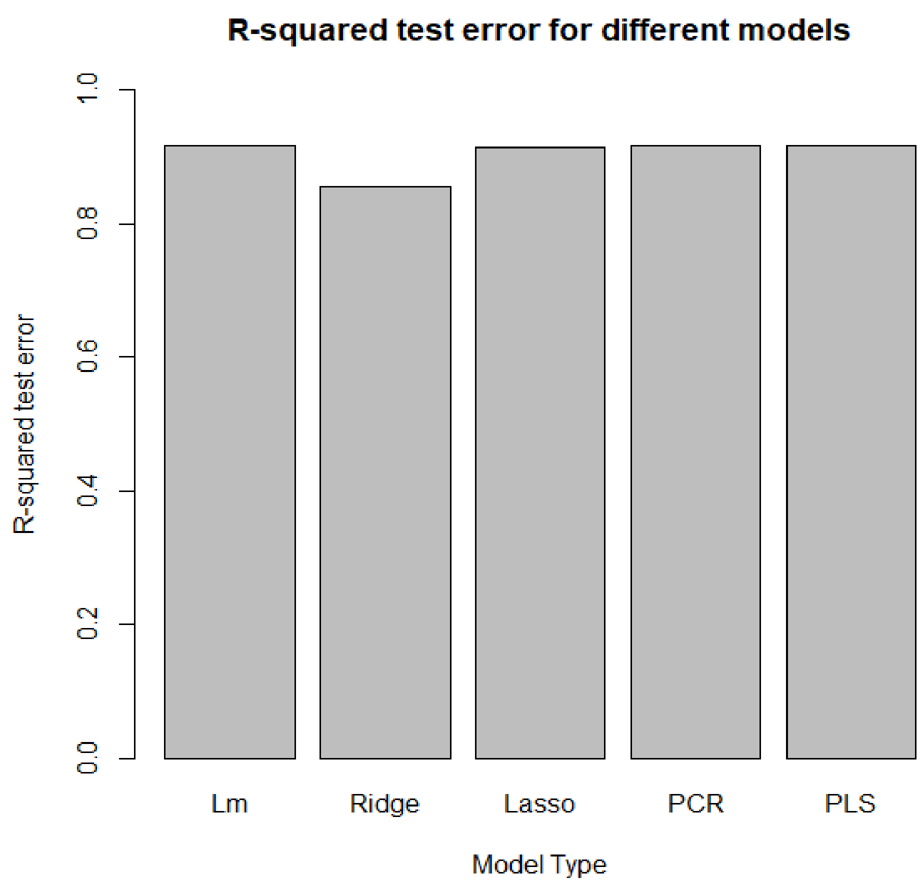


**Figure 5:** R-squared value for the test error of different model type

# Question 2

In this problem we had to predict whether a customer is interested in a caravan insurance policy using the CARAVAN dataset. It consists of consists of 86 variables/ predictors and includes product usage data and socio-demographic data derived from zip area codes. The data provided was split with 5,822 customers in the training set and another 4,000 in the test set. There was 2 testing data set, one with the response variable and one with the 86 predictors. Both were appended to create the complete testing dataset. Also, predictor and response variable column names were created to both the training and testing data to better understand the data.

### Linear Regression model

To predict whether a customer is interested in a caravan insurance policy, the OLS model was first fitted on the training data set and a summary of the fit is shown in table 7.

**Table 7**: Summary of the OLS model fitted on the training data set

```
Call:
lm(formula = CARAVAN ~ ., data = traindata)
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 .
MOSTYPE      0.0035209  0.0022512   1.564 0.117866
MAANTHUI    -0.0072642  0.0076739  -0.947 0.343875
MGEMOMV     -0.0012739  0.0071737  -0.178 0.859055
MGEMLEEF     0.0107473  0.0049596   2.167 0.030279 *
MOSHOOFD    -0.0154869  0.0101044  -1.533 0.125405
MGODRK      -0.0056016  0.0056016  -1.000 0.317353
MGODPR      -0.0002069  0.0060664  -0.034 0.972795
MGODOV       0.0003569  0.0054592   0.065 0.947874
MGODGE      -0.0030237  0.0058038  -0.521 0.602399
MRELGE       0.0086829  0.0075479   1.150 0.250036
MRELSA       0.0020367  0.0072008   0.283 0.777310
MRELOV       0.0055682  0.0076295   0.730 0.465526
MFALLEEN    -0.0038250  0.0065474  -0.584 0.559107
MFGEKIND    -0.0050625  0.0066861  -0.757 0.448980
MFWEKIND    -0.0026253  0.0069795  -0.376 0.706824
MOPLHOOG     0.0021357  0.0068161   0.313 0.754038
MOPLMIDD    -0.0048456  0.0071396  -0.679 0.497358
MOPLLAAG    -0.0113977  0.0073004  -1.561 0.118525
MBERHOOG     0.0021884  0.0045182   0.484 0.628153
MBERZELF    -0.0004665  0.0052201  -0.089 0.928796
MBERBOER    -0.0050974  0.0050426  -1.011 0.312122
MBERMIDD     0.0041254  0.0044806   0.921 0.357228
MBERARBG    -0.0006060  0.0044709  -0.136 0.892190
MBERARBO     0.0019733  0.0044532   0.443 0.657690
MSKA        -0.0013674  0.0051653  -0.265 0.791225
MSKB1       -0.0031701  0.0050198  -0.632 0.527724
MSKB2       -0.0012603  0.0044827  -0.281 0.778603
MSKC         0.0024879  0.0049115   0.507 0.612502
MSKD        -0.0008866  0.0047145  -0.188 0.850832
MHHUUR      -0.0454201  0.0376622  -1.206 0.227872
MHKOOP      -0.0432242  0.0376290  -1.149 0.250730
MAUT1        0.0085964  0.0075592   1.137 0.255502
MAUT2        0.0077871  0.0068554   1.136 0.256038
MAUT0        0.0047215  0.0072646   0.650 0.515762
MZFONDS     -0.0561024  0.0444643  -1.262 0.207094
MZPART      -0.0593733  0.0443897  -1.338 0.181097
MINKM30      0.0070879  0.0051150   1.386 0.165884
```

```
MINK3045     0.0069414  0.0049276   1.409 0.158986
MINK45455    0.0049679  0.0050144   0.991 0.321862
MINK45512    0.0059267  0.0052728   1.124 0.261053
MINK123M    -0.0098939  0.0069270  -1.428 0.153258
MINKGEM      0.0063044  0.0045645   1.381 0.167277
MKOOPKLA     0.0029097  0.0022664   1.284 0.199250
PWAPART      0.0284931  0.0166017   1.716 0.086166 .
PWABEDR     -0.0101533  0.0205121  -0.495 0.620625
PWALAND     -0.0201220  0.0390424  -0.515 0.606301
PPERSAUT     0.0102787  0.0026346   3.901 9.67e-05 ***
PBESAUT      0.0014405  0.0148574   0.097 0.922765
PMOTSCO     -0.0061279  0.0079415  -0.772 0.440364
PVRAAUT     -0.0249190  0.0415892  -0.599 0.549083
PAANHANG     0.0588044  0.0557610   1.055 0.291662
PTRACTOR     0.0121481  0.0142358   0.853 0.393504
PWERKT      -0.0062440  0.0370186  -0.169 0.866060
PBROM        0.0078683  0.0152793   0.515 0.606598
PLEVEN      -0.0155397  0.0064753  -2.400 0.016433 *
PPERSONG     0.0098926  0.0335157   0.295 0.767880
PGEZONG      0.1937254  0.0793370   2.442 0.014644 *
PWAOREG      0.0647933  0.0256913   2.522 0.011696 *
PBRAND       0.0132643  0.0035906   3.694 0.000223 ***
PZEILPL     -0.1917507  0.1439848  -1.332 0.182998
PPLEZIER    -0.0299076  0.0269224  -1.111 0.266666
PFIETS      -0.0107777  0.0549693  -0.196 0.844564
PINBOED     -0.0441620  0.0307404  -1.437 0.150883
PBYSTAND    -0.0184858  0.0288890  -0.640 0.522269
AWAPART     -0.0377952  0.0323794  -1.167 0.243154
AWABEDR      0.0185448  0.0529740   0.350 0.726296
AWALAND      0.0180904  0.1374585   0.132 0.895300
APERSAUT     0.0002821  0.0127496   0.022 0.982347
ABESAUT     -0.0214816  0.0652955  -0.329 0.742175
AMOTSCO      0.0203252  0.0310683   0.654 0.513004
AVRAAUT      0.0563675  0.1589388   0.355 0.722866
AAANHANG    -0.0804238  0.0944352  -0.852 0.394455
ATRACTOR    -0.0395651  0.0353795  -1.118 0.263484
AWERKT      -0.0010526  0.0728240  -0.014 0.988468
ABROM       -0.0236462  0.0467611  -0.506 0.613101
ALEVEN       0.0372344  0.0154024   2.417 0.015661 *
APERSONG    -0.0464279  0.0954471  -0.486 0.626684
AGEZONG     -0.4050642  0.1898715  -2.133 0.032938 *
AWAOREG     -0.2304561  0.1243310  -1.854 0.063852 .
ABRAND      -0.0211374  0.0116048  -1.821 0.068593 .
AZEILPL      0.4958051  0.2815591   1.761 0.078304 .
APLEZIER     0.3633887  0.0885318   4.105 4.11e-05 ***
AFIETS       0.0416061  0.0408644   1.018 0.308650
AINBOED      0.0959436  0.0699079   1.372 0.169983
ABYSTAND     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
```

From table 7 we can see the APLEZIER (number of boat policies), PBRAND (contribution fire policies), PPERSAUT (contribution car policies) are the predictors which appear to have a significant relationship to the response, since they have three significance stars in the last column indicating p<0.001, which means these predictors are very significant. Also, AGEZONG (number of private accident insurance policies), ALEVEN (Number of life insurances), PGEZONG(contribution to family accidents insurance policies), PWAOREG( contribution disability insurance policies), MGEMLEEF (average age) have a significant relationship to the response variable since they have 1 star thus p<0.05.

After fitting the OLS model, the MSE test error and train error for the OLS model was determine d to be 0.05210329 and 0.053985, respectively, which are low errors.

After fitting the OLS model to determine the prediction, a confusion matrix was constructed to determine the output of a model to examine all possible outcomes of the predictions.
First, the predicted probabilities were cut at a 25% threshold to turn probabilities into class predictions and determine a contingency table. After the confusion matrix was calculated and the associated statistics, as shown in table 8.
From these statistics the precision, recall and F1 score were determined. The precision is the ratio of correctly predicted positive observations to the total predicted positive observations, the recall (sensitivity) is the ratio of correctly predicted positive observations to the all observations in the actual class, and the F1 score the weighted average of precision and recall.

**Table 7**: A confusion matrix of who purchased the caravan policy and the associated statistics for OLS model

```
               Reference
Prediction      Not Purchased Purchased
  Not Purchased          3734       230
  Purchased                28         8

               Accuracy : 0.9355
                 95% CI : (0.9274, 0.9429)
    No Information Rate : 0.9405
    P-Value [Acc > NIR] : 0.9134

                  Kappa : 0.0434

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.03361
            Specificity : 0.99256
         Pos Pred Value : 0.22222
         Neg Pred Value : 0.94198
             Prevalence : 0.05950
         Detection Rate : 0.00200
   Detection Prevalence : 0.00900
      Balanced Accuracy : 0.51309

       'Positive' Class : Purchased

Recall (Sensitivity): 0.2222222
Precision: 0.03361345
F1 score: 0.05839416
```

Using the F1 score which is the weighted average of precision and recall we can see for the OLS model F1 is close to 1 indicating the test accuracy i.e. correctly predicted positive observations of this model is good

**Forward model**

To predict whether a customer is interested in a caravan insurance policy, the forward model w as fitted on the training data set. After fitting the forward selection model, MSE test error and MSE train error were found as shown in figure 6.

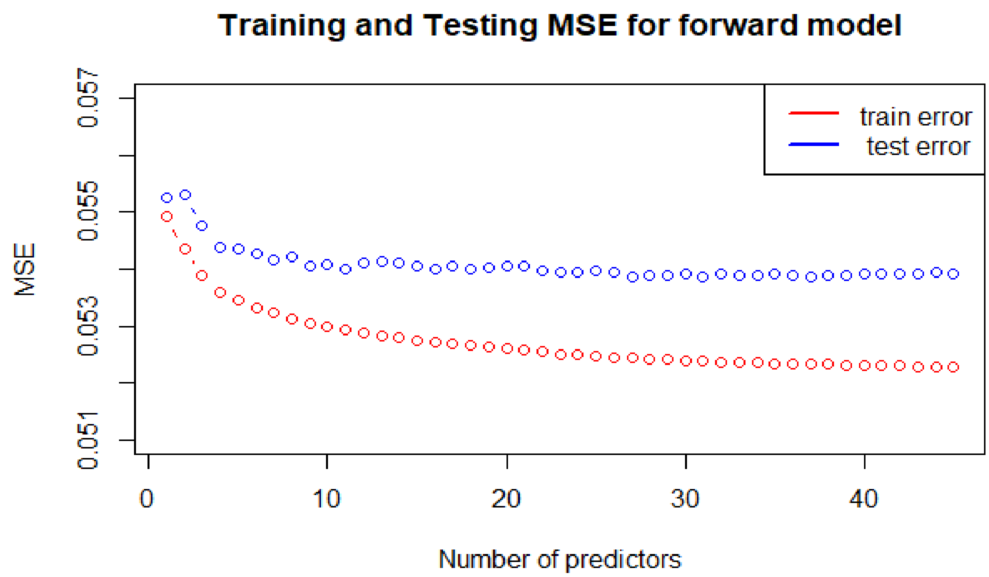


**Figure 6:** MSE test error and train error for different number of predictors using the forward selection model.

From figure 6 we can see that the predication training error decreases as the number of predictors/variable increase and we get a better fitter fit. It increases flexibility of model and will closely fit the observations. However, the testing error stays roughly constant as we add more predictors.

A confusion matrix was also constructed to determine the output of a model to examine all possible outcomes of the predictions.

The predicted probabilities were cut at a 25% threshold to turn probabilities into class predictions and determine a contingency table. The confusion matrix was calculated and the associated statistics. From these statistics the precision, recall and F1 score were determined, as shown in figure 7, 8, and 9 respectively.

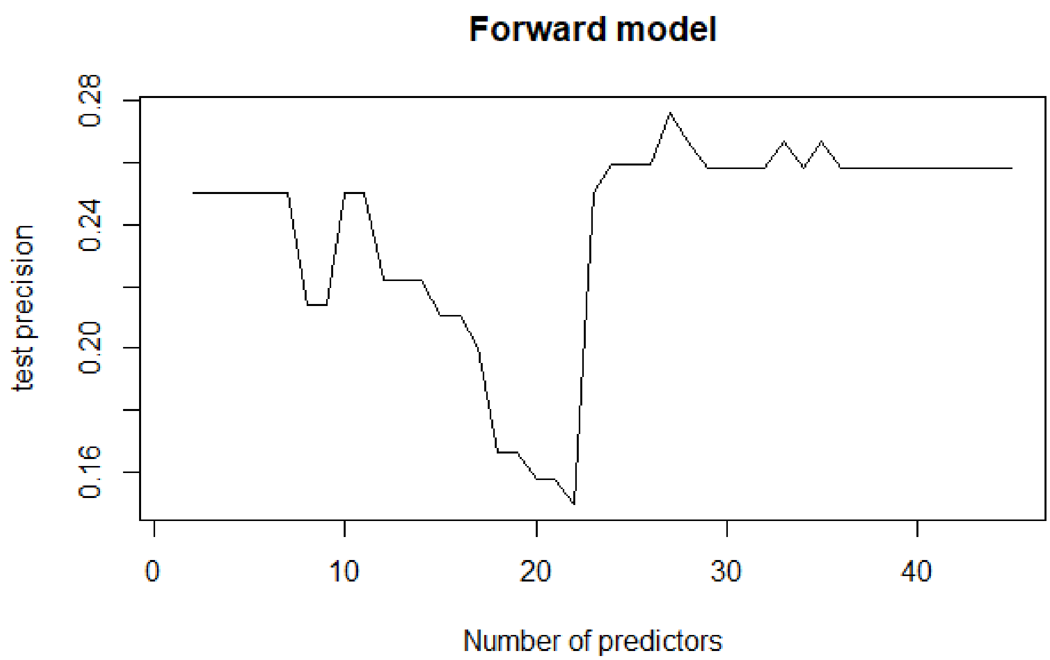


**Figure 7:** A plot of the precision (i.e. the predicted positive observations to the total predicted positive observations) on the test data for different number of predictors using the forward selection model

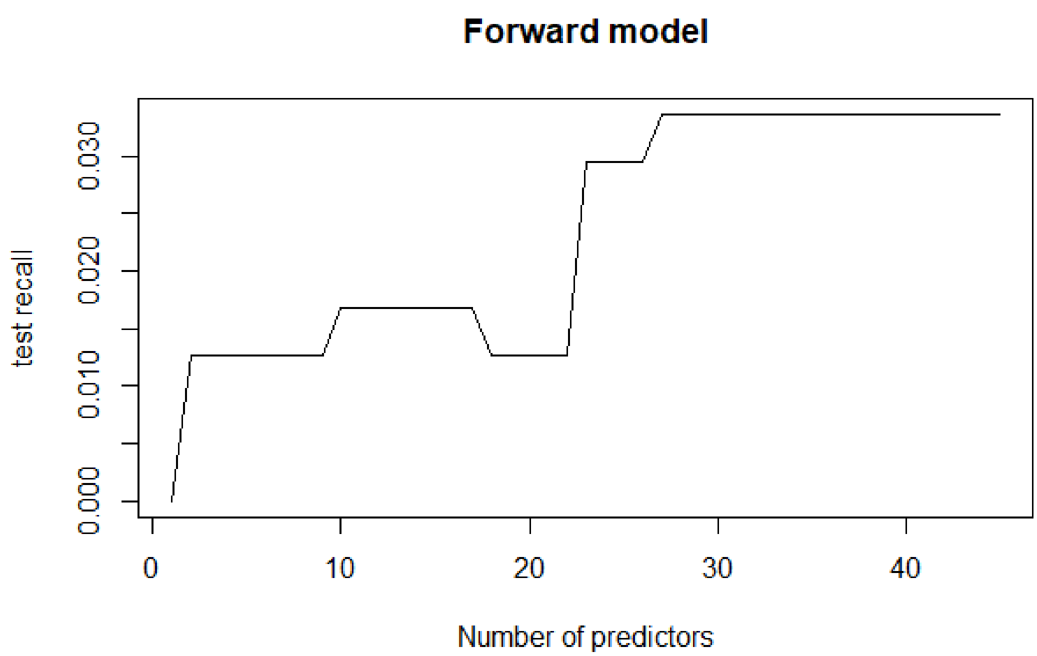


**Figure 8:** A plot of the recall (sensitivity) on the test data for different number of predictors using the forward selection model

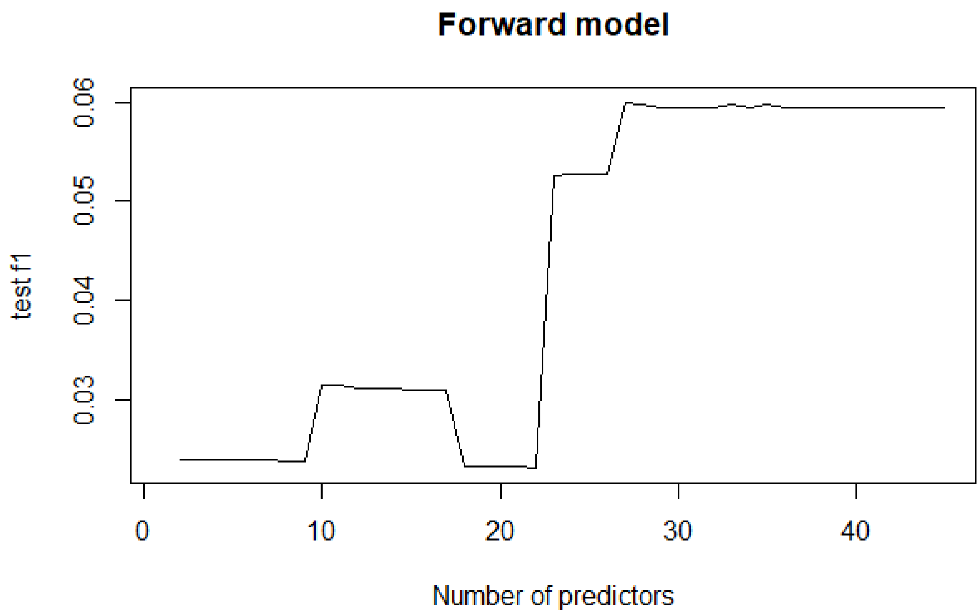


**Figure 9:** A plot of the F1 score (i.e. weighted average of precision and recall. on the test data for different number of predictors using the forward selection model

Using the F1 score, which is the weighted average of precision and recall for the forward model, we can see that the higher the number of predictors gives a better test accuracy for the model since it gets closer to 1.

**Backward selection model**
To predict whether a customer is interested in a caravan insurance policy, the backward model was fitted on the training data set. After fitting the backward selection model, MSE test error and MSE train error were found as shown in figure 10.

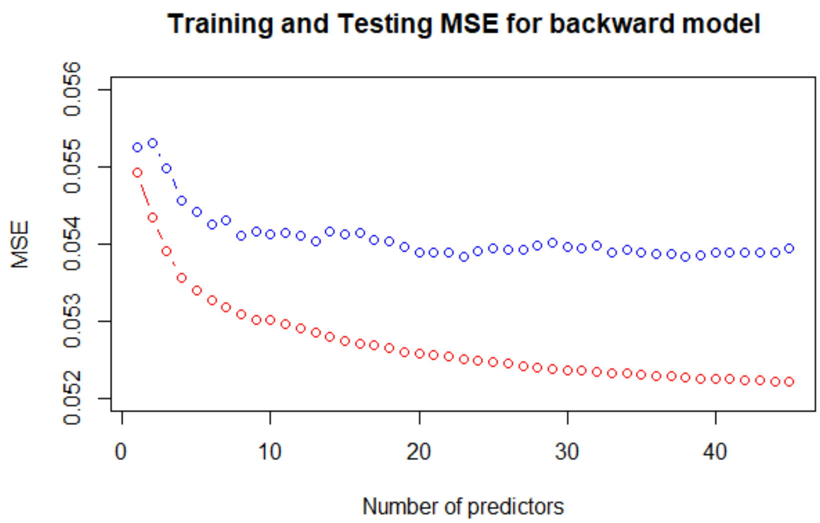


**Figure 10:** MSE test error and train error for different number of predictors using the backward selection model.

From figure 10 we can see that the predication training error decreases as the number of predictors/variable increase and we get a better fitter fit. It increases flexibility of model, becomes less bias and will closely fit the observations. However, the testing error stays roughly constant as we add more predictors.

A confusion matrix was also constructed to determine the output of a model to examine all possible outcomes of the predictions of the model. The predicted probabilities were cut at a 25% threshold to turn probabilities into class predictions and determine a contingency table. The confusion matrix was calculated and the associated statistics. From these statistics the precision, recall and F1 score were determined, as shown in figure 11, 12, and 13 respectively.

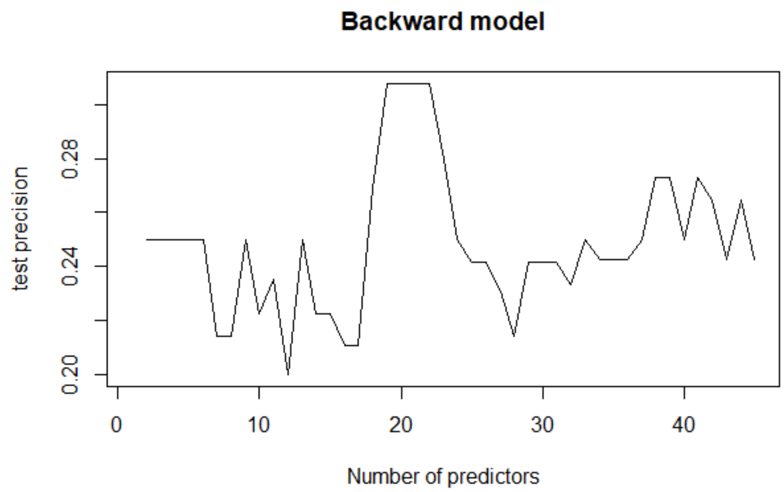


**Figure 11:** A plot of the precision (i.e. the predicted positive observations to the total predicted positive observations) on the test data for different number of predictors using the backward selection model

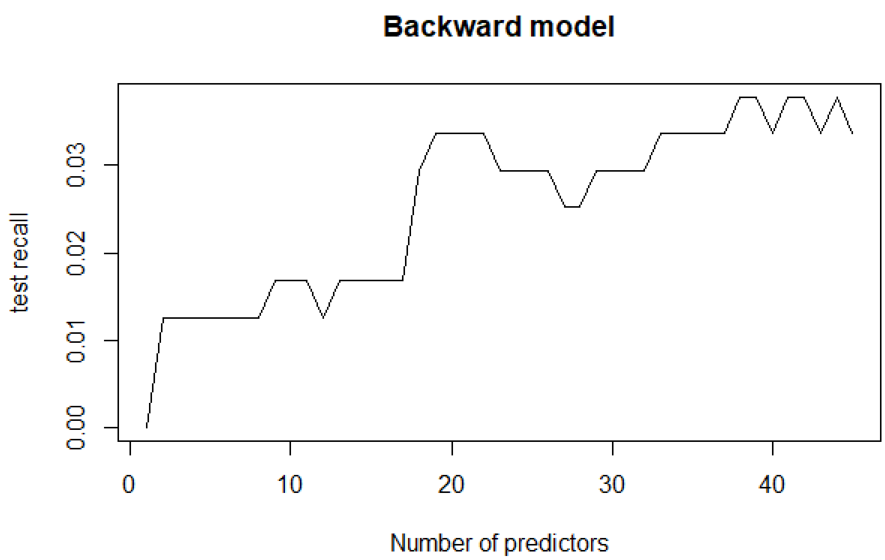


**Figure 12:** A plot of the recall (sensitivity) on the test data for different number of predictors using the backward selection model

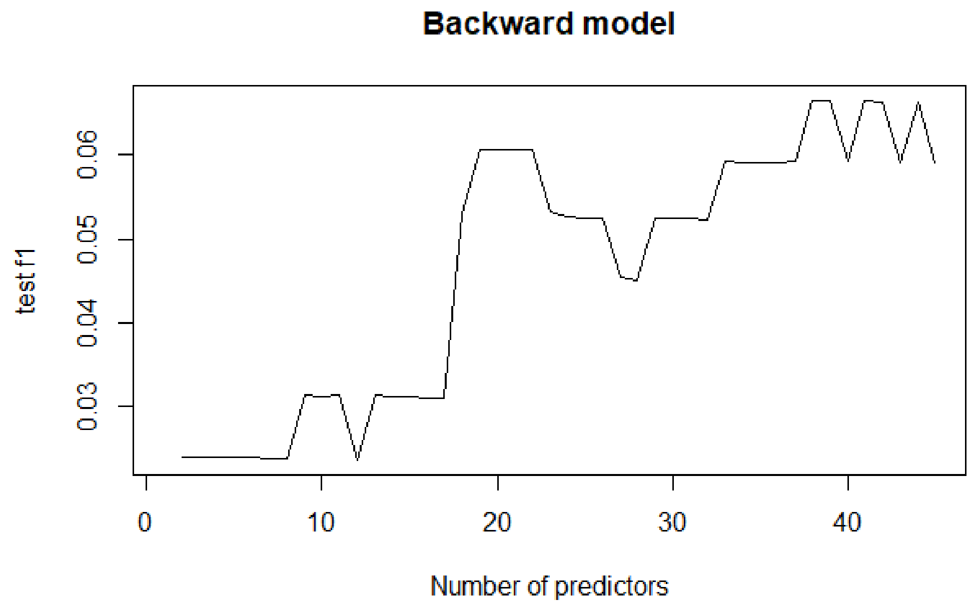


**Figure 13:** A plot of the F1 score (i.e. weighted average of precision and recall. on the test data for different number of predictors using the backward selection model

From the F1 score, which is the weighted average of precision and recall of the backward model, we can see that as the number of predictors increase the F1 score increases. We get a better test accuracy for the model as the F1 score get closer to one, indicating better prediction accuracy i.e. correctly predicted positive observations.

**Ridge regression model**

A ridge regression model was fitted on the training set, with λ chosen by cross-validation to predict whether a customer is interested in a caravan insurance policy. After fitting the model, MSE test error was found.

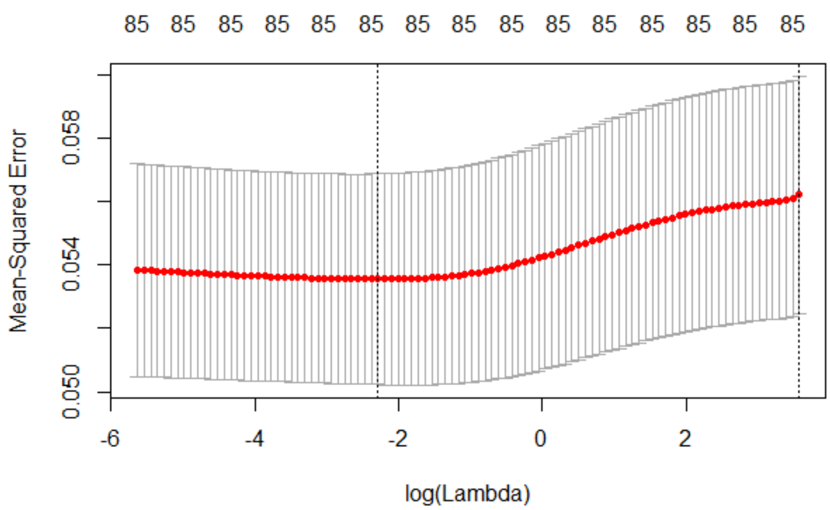


**Figure 14:** A CV output for Ridge Regression model fitted on the training data

The best lamda selected for the cross validation was 0.101.
The mean squared error for the test data was calculated to be 0.05369624, which is low error.

A confusion matrix was also constructed to determine the output of a model to examine all possible outcomes of the predictions of the model. The predicted probabilities were cut at a 25% threshold to turn probabilities into class predictions and determine a contingency table. The confusion matrix was calculated and the associated statistics, as shown in table 8.
From these statistics the precision, recall and F1 score were determined.

**Table 8**: A confusion matrix of who purchased the caravan policy and the associated statistics fo r Ridge Regression model

```
                Reference
Prediction      Not Purchased Purchased
  Not Purchased          3754       234
  Purchased                 8         4

               Accuracy : 0.9395
                 95% CI : (0.9317, 0.9467)
    No Information Rate : 0.9405
    P-Value [Acc > NIR] : 0.6216

                  Kappa : 0.0264

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.01681
            Specificity : 0.99787
         Pos Pred Value : 0.33333
         Neg Pred Value : 0.94132
             Prevalence : 0.05950
         Detection Rate : 0.00100
   Detection Prevalence : 0.00300
      Balanced Accuracy : 0.50734

       'Positive' Class : Purchased
```

```
Precision= 0.3333333
Recall= 0.01680672
F1 score= 0.032
```

From the F1 score which is the weighted average of precision and recall of the Ridge regression model, we can see it is low indicating poor test accuracy for the model as the F1 score is close to zero.

**LASSO model**

The lasso model was fitted on the training set, with λ chosen by cross validation to predict whether a customer is interested in a caravan insurance policy. After fitting the backward selection model, MSE test error was found.

The best lamda selected for the cross validation was 0.003184799
The mean squared error for the test data was calculated to be 0.05376028, which is a low error.

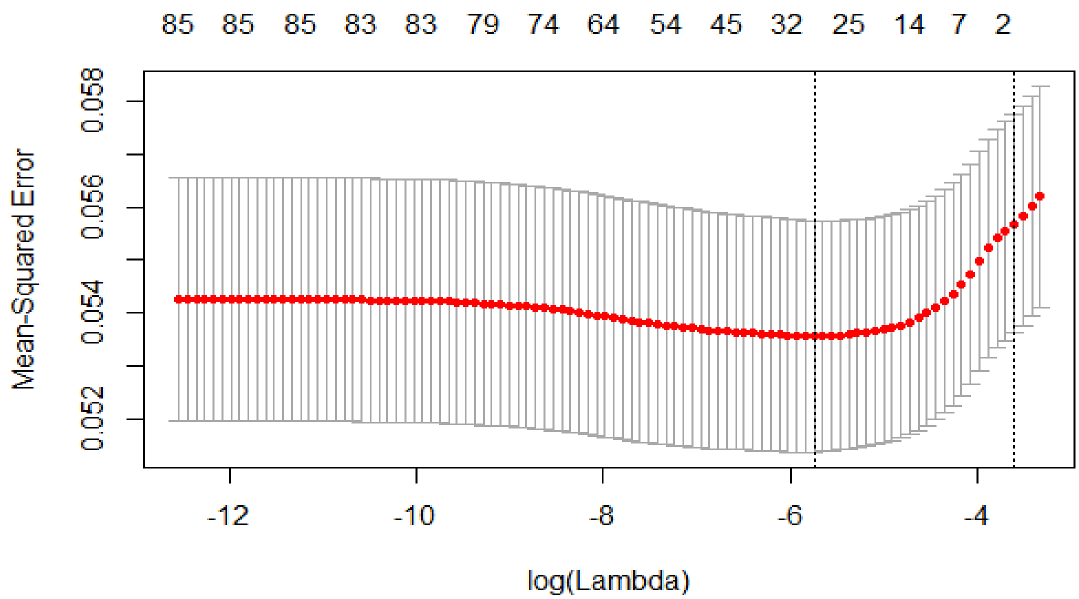


**Figure 15:** A CV output plot for the lasso model fitted on the training data

A confusion matrix was also constructed. The predicted probabilities were cut at a 25% threshold to turn probabilities into class predictions and determine a contingency table. The confusion matrix was calculated and the associated statistics, as shown in table 8.
From these statistics the precision, recall and F1 score were determined.

**Table 8:** A confusion matrix and the associated statistics for LASSO model

```
               Reference
Prediction      Not Purchased Purchased
  Not Purchased          3753       235
  Purchased                 9         3

               Accuracy : 0.939
                 95% CI : (0.9311, 0.9462)
    No Information Rate : 0.9405
    P-Value [Acc > NIR] : 0.6709

                  Kappa : 0.0184

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.01261
            Specificity : 0.99761
         Pos Pred Value : 0.25000
         Neg Pred Value : 0.94107
             Prevalence : 0.05950
         Detection Rate : 0.00075
   Detection Prevalence : 0.00300
      Balanced Accuracy : 0.50511

       'Positive' Class : Purchased
```

```
Precision= 0.25
Recall= 0.0126
F1 score= 0.024
```

Looking at the F1 score, which is the weighted average of precision and recall of the LASSO model, we can see it is low indicating poor test accuracy for the model as the F1 score is close to zero.

Overall, we can predict we can buy the caravan policy, using the OLS model. From the OLS model we can see that number of boat policies, contribution fire policies, contribution car policies are the predictors which appear to have a significant relationship to the response. Also, number of private accident insurance policies, number of life insurances, contribution to family accidents insurance policies contribution disability insurance policies, and average age are statistically significant predictors. Furthermore the F1 score for the OLS was 0.58 closer to 1 indicating the better test predictor, where the forward and backward testing accuracy fluctuate with the number of predictors.

## Question 3

In this problem we had to generate a data set with p = 20 features, n = 1, 000 observations, and an associated quantitative response vector according to the model $Y = X\beta + \varepsilon$ where β had some elements that are exactly equal to zero. The data was generated using a random number generator such as a normal distribution. After appending the response variable to the table of 20 predictors with 1,000 observations, the data set was split into a training set containing 100 observations and a test set containing 900 observations.

Afterwards a best subset selection using the exhaustive method was fitted on the training set, and the training set MSE associated with the best model of each size was plotted, as well as the test set MSE associated with the best model of each size, as shown in figure 16.

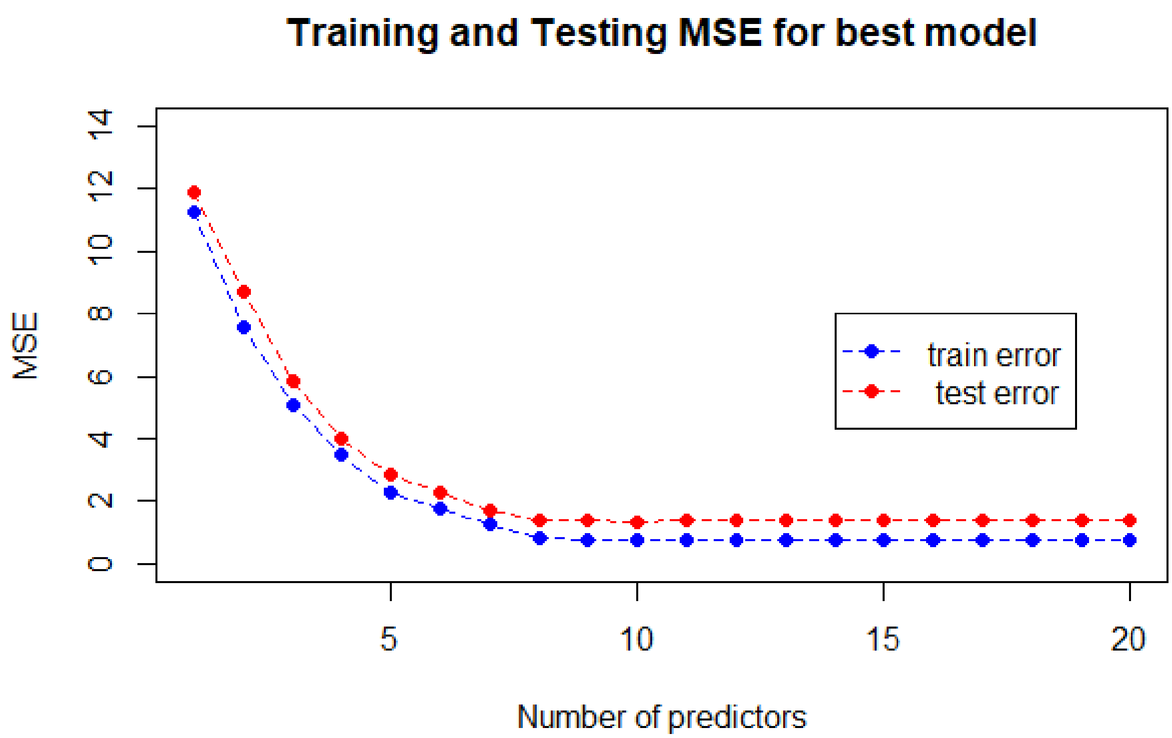


**Figure 16:** A plot of the training and testing MSE for the best model (exhaustive model)

The model size that the test set MSE takes on its minimum value is 10 predictors/variables, as shown in figure 16.

From the figure we can see as the number of predictors increase the training error and testing error decrease. The test error settles at a minimum test error at 10 predictors and then stops decreasing.

The MSE error for the training is lower because the model becomes more flexible and as we add more predictors, the observations will closely fit the model, leading to lower train RSS.

The model at which the test set MSE is minimized uses 10 predictors as opposed to 20 predictors in the true model, because the additional predictors can lead to overfitting of the model, which increase the test RSS.

The coefficients for the best subset model, for which the test MSE is minimum was determined as shown in table 9. From table 9 we can see that the best model (exhaustive model) was able to identify all the zeroed out β value coefficients in the true model and removed from the model. The β variables which were set equal to zero where 2, 3, 4, 7, 8, 11, 12, and 18.

**Table 9:** The best model coefficients where the test MSE is minimum and the zeroed β values not included.

```
(Intercept)          X1          X5          X6         X10         X13         X15         X16         X17         X19 
  0.1312413   0.7465284   0.9091897   1.1185736  -1.8602531   0.1119230   1.5159887  -1.5596310  -1.7591222  -0.8059523 
        X20 
 -0.1574736 
```