 # Assignment: ASSIGNMENT 9.2.1

 # Name: Rapuru, Supraja

 # Date: 2021-08-04

 # Analysis of Thoracic Surgery Binary Dataset

```
## Load the foreign package
library(foreign)
library(caTools)
## Set the working directory to the root of your DSC 520 directory
setwd("/Users/Supraja/dsc520")
thoraric_surgery_df <- read.arff("data/ThoraricSurgery.arff")
str(thoraric_surgery_df)
'data.frame':	470 obs. of  17 variables:
 $ DGN   : Factor w/ 7 levels "DGN1","DGN2",..: 2 3 3 3 3 3 3 2 3 3 ...
 $ PRE4  : num  2.88 3.4 2.76 3.68 2.44 2.48 4.36 3.19 3.16 2.32 ...
 $ PRE5  : num  2.16 1.88 2.08 3.04 0.96 1.88 3.28 2.5 2.64 2.16 ...
 $ PRE6  : Factor w/ 3 levels "PRZ0","PRZ1",..: 2 1 2 1 3 2 2 2 3 2 ...
 $ PRE7  : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...
 $ PRE8  : Factor w/ 2 levels "F","T": 1 1 1 1 2 1 1 1 1 1 ...
 $ PRE9  : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...
 $ PRE10 : Factor w/ 2 levels "F","T": 2 1 2 1 2 2 2 2 2 2 ...
 $ PRE11 : Factor w/ 2 levels "F","T": 2 1 1 1 2 1 1 1 2 1 ...
 $ PRE14 : Factor w/ 4 levels "OC11","OC12",..: 4 2 1 1 1 1 2 1 1 1 ...
 $ PRE17 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 2 1 1 1 ...
 $ PRE19 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...
 $ PRE25 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 2 1 1 ...
 $ PRE30 : Factor w/ 2 levels "F","T": 2 2 2 1 2 1 2 2 2 2 ...
 $ PRE32 : Factor w/ 2 levels "F","T": 1 1 1 1 1 1 1 1 1 1 ...
 $ AGE   : num  60 51 59 54 73 51 59 66 68 54 ...
```

```
 $ Risk1Yr: Factor w/ 2 levels "F","T": 1 1 1 1 2 1 2 2 1 1 ...
```

head(thoraric_surgery_df)

```
   DGN PRE4 PRE5 PRE6 PRE7 PRE8 PRE9 PRE10 PRE11 PRE14 PRE17 PRE19 PRE25
1 DGN2 2.88 2.16 PRZ1    F    F    F     T     T  OC14     F     F     F
2 DGN3 3.40 1.88 PRZ0    F    F    F     F     F  OC12     F     F     F
3 DGN3 2.76 2.08 PRZ1    F    F    F     T     F  OC11     F     F     F
4 DGN3 3.68 3.04 PRZ0    F    F    F     F     F  OC11     F     F     F
5 DGN3 2.44 0.96 PRZ2    F    T    F     T     T  OC11     F     F     F
6 DGN3 2.48 1.88 PRZ1    F    F    F     T     F  OC11     F     F     F
  PRE30 PRE32 AGE Risk1Yr
1     T     F  60       F
2     T     F  51       F
3     T     F  59       F
4     F     F  54       F
5     T     F  73       T
6     F     F  51       F
```

i.Fit a binary logistic regression model to the data set that predicts whether or not the patient survived for one year (the Risk1Y variable) after the surgery. Use the glm() function to perform the logistic regression.

See Generalized Linear Models for an example. Include a summary using the summary() function in your results.

#Fit the binary logistic regression model to the data set

mymodel <-glm(Risk1Yr ~ .,data = thoraric_surgery_df, family = 'binomial')

summary(mymodel)

```
Call:
glm(formula = Risk1Yr ~ ., family = "binomial", data = thoraric_surgery_df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6084  -0.5439  -0.4199  -0.2762   2.4929
```

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.655e+01  2.400e+03  -0.007  0.99450
DGNDGN2      1.474e+01  2.400e+03   0.006  0.99510
DGNDGN3      1.418e+01  2.400e+03   0.006  0.99528
DGNDGN4      1.461e+01  2.400e+03   0.006  0.99514
DGNDGN5      1.638e+01  2.400e+03   0.007  0.99455
DGNDGN6      4.089e-01  2.673e+03   0.000  0.99988
DGNDGN8      1.803e+01  2.400e+03   0.008  0.99400
PRE4        -2.272e-01  1.849e-01  -1.229  0.21909
PRE5        -3.030e-02  1.786e-02  -1.697  0.08971 .
PRE6PRZ1    -4.427e-01  5.199e-01  -0.852  0.39448
PRE6PRZ2    -2.937e-01  7.907e-01  -0.371  0.71030
PRE7T        7.153e-01  5.556e-01   1.288  0.19788
PRE8T        1.743e-01  3.892e-01   0.448  0.65419
PRE9T        1.368e+00  4.868e-01   2.811  0.00494 **
PRE10T       5.770e-01  4.826e-01   1.196  0.23185
PRE11T       5.162e-01  3.965e-01   1.302  0.19295
PRE14OC12    4.394e-01  3.301e-01   1.331  0.18318
PRE14OC13    1.179e+00  6.165e-01   1.913  0.05580 .
PRE14OC14    1.653e+00  6.094e-01   2.713  0.00668 **
PRE17T       9.266e-01  4.445e-01   2.085  0.03709 *
PRE19T      -1.466e+01  1.654e+03  -0.009  0.99293
PRE25T      -9.789e-02  1.003e+00  -0.098  0.92227
PRE30T       1.084e+00  4.990e-01   2.172  0.02984 *
PRE32T      -1.398e+01  1.645e+03  -0.008  0.99322
AGE         -9.506e-03  1.810e-02  -0.525  0.59944
---
```

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 395.61 on 469 degrees of freedom

Residual deviance: 341.19 on 445 degrees of freedom

AIC: 391.19

Number of Fisher Scoring iterations: 15

ii. According to the summary, which variables had the greatest effect on the survival rate?

As all the below variables have less p-value, it looks like below are the good predictors for the whether or not the patient Risk1Y variable) after the surgery.

 # PRE5,PRE9T,PRE14OC13,PRE14OC14,PRE17T,PRE30T

iii. To compute the accuracy of your model, use the dataset to predict the outcome variable. The percent of correct predictions is the accuracy of your model.

What is the accuracy of your model?

#Split the data into test and train datasets

split <- sample.split(thoraric_surgery_df,SplitRatio = 0.8)

split

```
 [1]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE
[13] FALSE FALSE  TRUE  TRUE  TRUE
```

train<- subset(thoraric_surgery_df,split=="TRUE")

test<- subset(thoraric_surgery_df,split=="FALSE")

#run the test data through model

res<- predict(mymodel,test,type="response")

res

```
           7           10           13           14           24
1.918605e-01 9.458663e-02 1.154378e-01 4.908434e-01 5.824619e-02
          27           30           31           41           44
7.223499e-02 5.945905e-08 3.730799e-01 3.831235e-01 6.839303e-01
          47           48           58           61           64
8.354285e-02 1.128335e-01 3.868351e-01 1.882038e-01 5.221406e-02
          65           75           78           81           82
2.068899e-01 5.622961e-02 1.088240e-01 1.007965e-01 3.642241e-01
          92           95           98           99          109
7.598004e-02 2.064928e-01 8.663401e-08 5.044656e-02 2.236160e-02
         112          115          116          126          129
2.098142e-01 1.245632e-01 2.922307e-01 1.099803e-01 4.130719e-01
         132          133          143          146          149
1.221660e-01 1.801905e-01 1.029460e-02 9.334413e-02 8.884902e-02
         150          160          163          166          167
6.588596e-02 9.485986e-02 2.214874e-01 3.826184e-01 1.813499e-01
         177          180          183          184          194
3.419036e-01 2.023070e-01 7.236749e-02 1.208968e-01 8.899037e-02
         197          200          201          211          214
1.467324e-01 1.827940e-01 1.353227e-01 4.821277e-02 2.562132e-01
         217          218          228          231          234
1.778609e-01 7.094838e-02 1.206525e-01 1.897757e-01 1.282731e-01
         235          245          248          251          252
1.317057e-01 3.259522e-08 1.397018e-01 9.038743e-02 1.235385e-01
         262          265          268          269          279
1.358705e-01 8.239156e-02 3.207561e-01 4.979178e-01 1.913885e-02
         282          285          286          296          299
2.915087e-02 8.066292e-02 7.923320e-02 1.088983e-01 1.081833e-01
```

```
       302          303          313          316          319
3.501333e-02 1.976446e-01 2.067867e-01 2.022857e-01 8.579839e-02
       320          330          333          336          337
1.157016e-02 5.928026e-02 7.606859e-02 8.617946e-02 1.576282e-01
       347          350          353          354          364
1.104491e-01 5.654319e-03 1.349788e-02 5.923665e-02 1.613167e-01
       367          370          371          381          384
8.388680e-02 8.565278e-02 1.063537e-01 1.138013e-01 3.412311e-02
       387          388          398          401          404
2.795678e-01 1.164616e-01 8.124317e-02 2.757069e-02 1.132803e-01
       405          415          418          421          422
2.694429e-01 1.205709e-01 4.364347e-02 3.111636e-01 3.420630e-01
       432          435          438          439          449
1.122630e-01 7.843992e-02 1.073693e-01 1.186243e-01 9.585091e-02
       452          455          456          466          469
1.667358e-01 5.883086e-02 1.580380e-01 2.763209e-01 1.908312e-01
```

```
#run the train data through model
res<- predict(mymodel,train,type="response")
res
```

```
           1            2            3            4            5
5.699656e-01 1.031988e-01 8.287068e-02 2.160824e-02 1.692634e-01
           6            8            9           11           12
3.415054e-02 1.068699e-01 1.265083e-01 8.295347e-02 4.978455e-02
          15           16           17           18           19
8.528088e-02 7.638833e-02 2.298384e-01 1.686594e-01 1.170482e-01
          20           21           22           23           25
6.346676e-02 7.899455e-02 1.358877e-01 1.166706e-01 4.628603e-01
          26           28           29           32           33
```

2.759707e-01 1.044741e-01 1.225337e-01 3.210049e-02 5.401980e-01

34 35 36 37 38

1.222741e-01 4.321161e-02 8.141605e-02 1.247959e-01 1.985475e-01

39 40 42 43 45

5.379752e-02 5.736768e-02 1.723143e-01 1.022412e-01 1.886592e-01

46 49 50 51 52

7.698128e-02 1.528144e-01 2.634907e-02 3.990471e-02 5.705188e-02

53 54 55 56 57

5.605594e-01 1.268064e-01 9.604222e-02 1.518051e-01 1.040492e-01

59 60 62 63 66

9.091183e-02 8.436518e-02 1.775659e-01 4.497232e-02 4.547291e-02

67 68 69 70 71

3.426478e-02 2.306748e-01 1.215150e-01 1.235686e-01 1.769600e-02

72 73 74 76 77

2.044482e-01 5.872367e-02 1.854511e-02 3.214431e-01 1.517401e-01

79 80 83 84 85

1.454896e-01 3.573413e-02 1.092554e-01 6.808071e-02 8.282431e-02

86 87 88 89 90

9.959463e-02 1.516943e-01 2.220150e-01 6.230735e-01 1.389749e-01

91 93 94 96 97

1.475171e-01 1.018244e-01 3.580610e-02 5.670370e-02 1.650967e-01

100 101 102 103 104

3.001414e-01 6.405787e-02 3.957982e-01 1.102611e-01 2.874635e-08

105 106 107 108 110

3.097683e-02 1.314217e-01 1.343593e-01 1.068128e-01 2.980639e-01

111 113 114 117 118

1.234449e-01 1.482006e-02 4.971735e-02 2.340033e-01 2.686309e-01

119 120 121 122 123

6.225151e-02 1.764599e-01 3.945990e-02 9.033179e-02 6.199320e-01

124 125 127 128 130

8.917611e-02 1.457683e-01 5.418171e-02 3.286049e-01 8.031190e-02

131 134 135 136 137

6.957820e-02 8.439071e-02 7.935226e-02 7.695837e-02 2.933734e-01

138 139 140 141 142

3.812039e-01 1.332096e-01 2.572193e-02 1.500561e-01 9.231166e-02

144 145 147 148 151

1.677159e-01 1.824691e-01 2.010585e-02 1.100579e-01 4.217588e-02

152 153 154 155 156

7.084935e-02 4.472309e-02 1.399897e-01 1.027427e-01 9.794784e-02

157 158 159 161 162

4.854969e-01 1.019523e-07 1.867933e-01 3.309436e-02 7.273292e-02

164 165 168 169 170

7.306653e-02 4.378233e-01 1.147794e-01 1.863320e-01 3.319553e-01

171 172 173 174 175

8.981011e-02 3.371654e-01 4.754743e-01 8.801868e-02 1.701133e-01

176 178 179 181 182

3.810037e-01 1.155253e-01 1.691160e-01 1.555587e-01 7.226418e-02

185 186 187 188 189

2.770187e-02 4.974416e-01 7.037954e-02 1.081729e-01 8.370741e-02

190 191 192 193 195

9.786972e-02 1.071501e-07 7.315314e-02 5.107552e-02 6.161650e-02

196 198 199 202 203

1.414413e-01 4.208491e-02 3.568805e-02 7.811592e-02 3.490320e-01

204 205 206 207 208

1.466339e-01 3.045425e-02 1.172731e-01 5.645845e-02 8.096561e-02

209 210 212 213 215

7.137263e-02 3.416674e-01 1.035481e-01 3.447902e-01 7.482114e-02

216 219 220 221 222

1.935358e-01	5.571797e-02	6.582535e-02	7.270148e-01	1.194467e-01
223	224	225	226	227
2.586989e-01	5.110705e-02	8.371578e-02	3.768849e-01	1.733864e-01
229	230	232	233	236
2.726272e-02	2.558265e-01	5.557867e-01	8.326085e-02	8.638962e-02
237	238	239	240	241
1.567634e-01	1.013461e-01	4.082054e-01	1.033867e-01	4.409613e-02
242	243	244	246	247
6.391354e-02	4.370160e-01	3.604740e-02	7.021216e-02	7.865337e-02
249	250	253	254	255
1.168226e-01	1.146856e-01	9.386811e-02	9.485861e-02	7.640224e-02
256	257	258	259	260
3.947346e-02	8.482854e-02	7.348739e-02	8.010688e-02	9.248713e-02
261	263	264	266	267
1.134974e-01	1.392593e-01	3.270853e-02	1.027026e-01	8.726133e-02
270	271	272	273	274
1.011537e-01	1.828671e-01	3.733253e-01	4.705393e-02	3.399052e-01
275	276	277	278	280
1.567863e-01	1.394679e-01	1.087993e-01	2.164656e-01	6.634443e-02
281	283	284	287	288
9.474987e-02	7.344261e-02	2.368618e-01	1.148553e-01	1.138796e-01
289	290	291	292	293
4.295451e-01	9.208997e-02	1.361976e-01	2.422470e-01	6.389221e-08
294	295	297	298	300
7.516974e-02	2.834210e-01	1.352075e-01	4.421943e-01	9.709489e-02
301	304	305	306	307
1.561671e-01	1.532303e-01	6.402083e-02	1.129776e-01	6.260657e-01
308	309	310	311	312
1.232557e-01	8.953267e-02	7.994164e-02	3.219110e-02	9.183286e-02

```
         314          315          317          318          321 
1.165480e-01 1.848784e-01 3.778067e-02 3.285881e-01 2.226277e-01 
         322          323          324          325          326 
6.807046e-02 7.937344e-02 3.651378e-01 4.155550e-02 7.208965e-03 
         327          328          329          331          332 
1.526670e-01 1.666427e-01 1.462120e-01 3.731696e-02 5.786913e-02 
         334          335          338          339          340 
4.020393e-02 1.420674e-01 1.472018e-01 5.226116e-02 1.184043e-01 
         341          342          343          344          345 
5.243980e-02 8.247275e-02 1.308726e-01 1.241559e-01 9.590097e-02 
         346          348          349          351          352 
5.656586e-01 2.955094e-01 1.098571e-01 1.324475e-01 7.237318e-02 
         355          356          357          358          359 
5.718804e-02 1.025151e-01 3.593093e-01 1.182733e-01 1.279055e-01 
         360          361          362          363          365 
5.614757e-02 1.310811e-01 8.812173e-02 3.602838e-01 1.680713e-01 
         366          368          369          372          373 
1.219306e-01 7.446550e-01 9.387401e-08 4.586356e-02 8.895595e-02 
         374          375          376          377          378 
7.256814e-01 1.212894e-01 6.274914e-02 6.161964e-02 1.197857e-01 
         379          380          382          383          385 
7.570812e-02 1.073616e-01 4.627649e-02 1.229746e-01 5.307208e-02 
         386          389          390          391          392 
2.491018e-01 2.464913e-01 4.146143e-01 1.034826e-01 2.719705e-01 
         393          394          395          396          397 
2.534894e-01 9.711942e-02 1.678380e-01 2.298356e-01 5.616655e-02 
         399          400          402          403          406 
1.166192e-01 8.003204e-02 2.984281e-02 1.238295e-01 2.519493e-08 
         407          408          409          410          411 
```

```
7.206242e-02 1.665778e-01 2.468327e-01 7.494754e-02 2.054893e-01
         412          413          414          416          417
2.746506e-01 2.333291e-02 1.471190e-01 2.156125e-02 2.147515e-01
         419          420          423          424          425
1.413123e-01 2.844515e-01 1.008647e-01 4.699953e-02 1.966650e-01
         426          427          428          429          430
1.228541e-01 2.471998e-01 5.189285e-02 1.736524e-01 4.688095e-01
         431          433          434          436          437
8.261827e-02 6.454238e-02 1.250300e-01 8.168373e-02 2.592223e-01
         440          441          442          443          444
1.379159e-01 1.720875e-01 4.374357e-02 1.902351e-01 3.464447e-02
         445          446          447          448          450
1.492523e-02 7.192786e-02 5.371397e-01 2.229532e-01 1.278963e-01
         451          453          454          457          458
5.352113e-02 3.479825e-01 1.344147e-01 1.317175e-01 8.141729e-02
         459          460          461          462          463
2.703658e-02 4.519309e-02 4.462500e-02 1.132793e-01 1.270542e-01
         464          465          467          468          470
4.422608e-01 2.741168e-01 5.646663e-02 9.063997e-02 7.494837e-02
```

#Validate the model - confusion Matrix

```
confmatrix <- table(Actual_Value=train$Risk1Yr,Predicted_Value = res 0.5)
confmatrix
```

```
           Predicted_Value
Actual_Value FALSE TRUE
           F   303   10
           T    45    2
```

#Accuracy of the model

(confmatrix[[1,1]] + confmatrix[[2,2]]) / sum(confmatrix)

[1] 0.8472222

#The accuracy of the model is 84.7%