Breast Cancer Analysis and Predictor

2024-04-28

```r
##First we will import the Breast Cancer table from Kaggle to perform testing##
library(readxl)
BRCA <- read_excel("/Users/gwengorman/Downloads/BRCA 2.xlsx",
    col_types = c("text", "numeric", "text",
       "numeric", "numeric", "numeric",
       "numeric", "text", "text", "text",
      "text", "text", "text", "date", "date",
     "text"))
View(BRCA)

#remove any observations with null data#
breast_cancer <- na.omit(BRCA)

summary(breast_cancer)
```

```
##   Patient_ID             Age           Gender             Protein1
##  Length:317         Min.   :29.00   Length:317         Min.   :-2.144600
##  Class :character   1st Qu.:49.00   Class :character   1st Qu.:-0.350600
##  Mode  :character   Median :58.00   Mode  :character   Median : 0.005649
##                     Mean   :58.73                      Mean   :-0.027232
##                     3rd Qu.:67.00                      3rd Qu.: 0.336260
##                     Max.   :90.00                      Max.   : 1.593600
##     Protein2          Protein3          Protein4          Tumour_Stage
##  Min.   :-0.9787   Min.   :-1.6274   Min.   :-2.025500   Length:317
##  1st Qu.: 0.3688   1st Qu.:-0.5314   1st Qu.:-0.382240   Class :character
##  Median : 0.9971   Median :-0.1930   Median : 0.038522   Mode  :character
##  Mean   : 0.9496   Mean   :-0.0951   Mean   : 0.006713
##  3rd Qu.: 1.6120   3rd Qu.: 0.2512   3rd Qu.: 0.436250
##  Max.   : 3.4022   Max.   : 2.1934   Max.   : 1.629900
##   Histology          ER status          PR status          HER2 status
##  Length:317         Length:317         Length:317         Length:317
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  Surgery_type       Date_of_Surgery
##  Length:317         Min.   :2017-01-15 00:00:00.0
##  Class :character   1st Qu.:2018-03-11 00:00:00.0
##  Mode  :character   Median :2018-09-27 00:00:00.0
##                     Mean   :2018-09-04 07:56:58.2
##                     3rd Qu.:2019-03-26 00:00:00.0
##                     Max.   :2019-11-21 00:00:00.0
##  Date_of_Last_Visit               Patient_Status
##  Min.   :2017-04-05 00:00:00.00   Length:317
##  1st Qu.:2019-01-29 00:00:00.00   Class :character
##  Median :2019-12-28 00:00:00.00   Mode  :character
##  Mean   :2019-11-26 02:34:26.88
##  3rd Qu.:2020-08-27 00:00:00.00
##  Max.   :2026-09-24 00:00:00.00
```

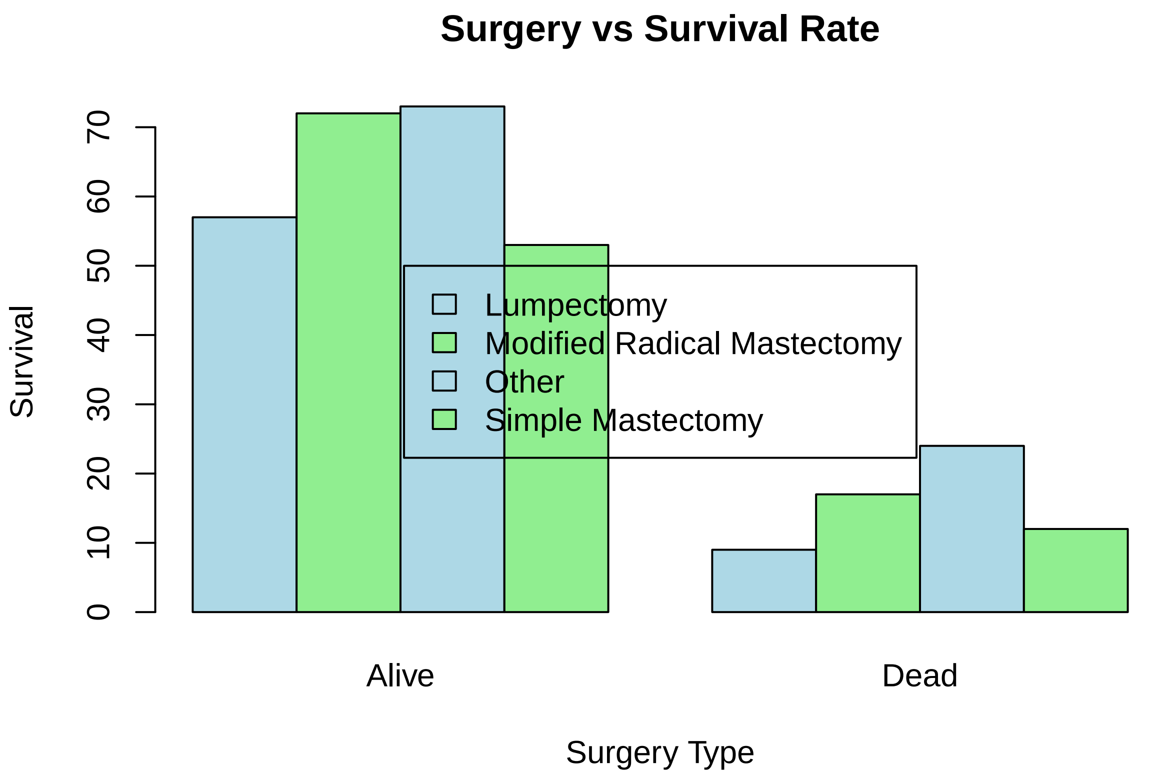
```
###We will perform bivariate analysis of the given variables
#to test whether the different variables are independent of survival rate
#at a .05 significance level.To do this, we will need to perform a
#chi-squared test given the response variable is categorical###

#SURGERY vs. SURVIVAL
cancer_survival = data.frame(breast_cancer$Surgery_type,breast_cancer$Patient_Status)

#contingency table
cancer_survival = table(breast_cancer$Surgery_type,breast_cancer$Patient_Status)

print(cancer_survival)
```

```
##
##                                  Alive Dead
##    Lumpectomy                       57    9
##    Modified Radical Mastectomy     72   17
##    Other                           73   24
##    Simple Mastectomy               53   12
```

```
#Chi-square test
print(chisq.test(cancer_survival))
```

```
##
##  Pearson's Chi-squared test
##
## data:  cancer_survival
## X-squared = 3.1895, df = 3, p-value = 0.3633
```

```
#Bar Plot
barplot(cancer_survival, beside = TRUE, col = c("lightblue", "lightgreen"),
        main = "Surgery vs Survival Rate",
        xlab = "Surgery Type", ylab = "Survival")

legend("center", legend = rownames(cancer_survival), fill = c("lightblue", "lightgreen"))
```



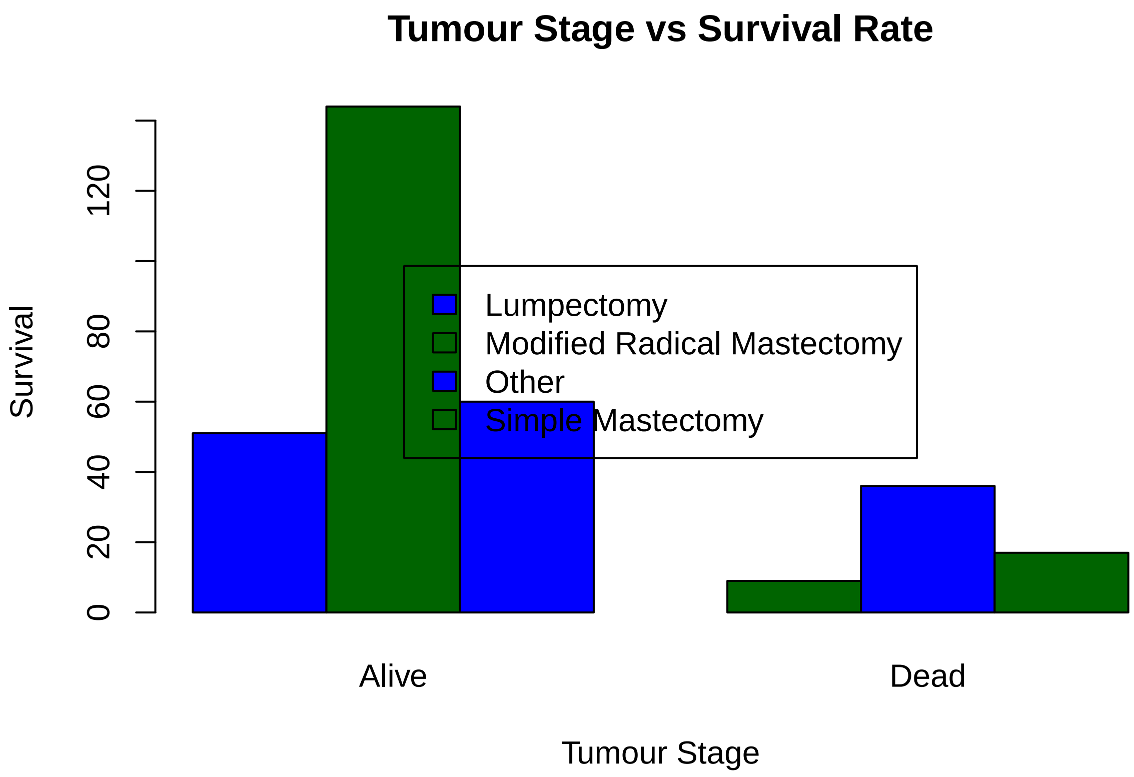
```
##With a p-value of .36, we conclude that surgery is independent of survival rate
#and thus there is weak correlation between the two variables##

#TUMOR STAGE vs. SURVIVAL
tumour_survival = data.frame(breast_cancer$Tumour_Stage,breast_cancer$Patient_Status)

tumour_survival = table(breast_cancer$Tumour_Stage,breast_cancer$Patient_Status)

print(tumour_survival)
```

```
##
##         Alive Dead
##   I        51    9
##   II      144   36
##   III      60   17
```

```
print(chisq.test(tumour_survival))
```

```
##
##  Pearson's Chi-squared test
##
## data:  tumour_survival
## X-squared = 1.1254, df = 2, p-value = 0.5697
```

```
barplot(tumour_survival, beside = TRUE, col = c("blue", "darkgreen"),
        main = "Tumour Stage vs Survival Rate",
        xlab = "Tumour Stage", ylab = "Survival")

legend("center", legend = rownames(cancer_survival), fill = c("blue", "darkgreen"))
```



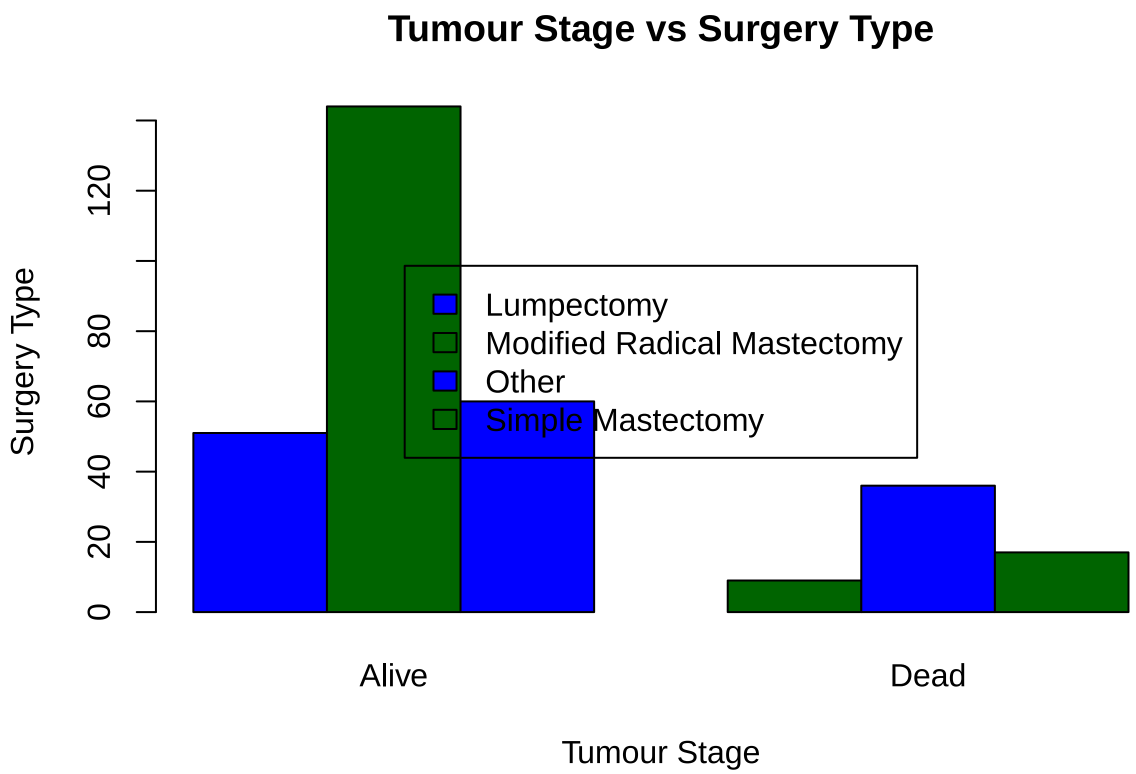
```
#with a p-value of 0.57 we conclude that tumour stage is independent of survival rate
#and thus, there is weak correlation between the two variables.##

#TUMOUR VS SURGERY TYPE#

tumour_surgery = data.frame(breast_cancer$Surgery_type,breast_cancer$Tumour_Stage)

tumour_surgery = table(breast_cancer$Surgery_type,breast_cancer$Tumour_Stage)

print(tumour_surgery)
```

```
##
##                                  I II III
##   Lumpectomy                    22 36   8
##   Modified Radical Mastectomy    7 45  37
##   Other                         18 56  23
##   Simple Mastectomy             13 43   9
```

```
print(chisq.test(tumour_surgery))
```

```
##
##  Pearson's Chi-squared test
##
## data:  tumour_surgery
## X-squared = 32.623, df = 6, p-value = 1.239e-05
```

```
barplot(tumour_survival, beside = TRUE, col = c("blue", "darkgreen"),
        main = "Tumour Stage vs Surgery Type",
        xlab = "Tumour Stage", ylab = "Surgery Type")

legend("center", legend = rownames(cancer_survival), fill = c("blue", "darkgreen"))
```



```
#with a p-value of 0.000012 we conclude that there is a strong correlation between the two variables.##

#GENDER vs. SURVIVAL
gender_survival = data.frame(breast_cancer$Gender,breast_cancer$Patient_Status)

gender_survival = table(breast_cancer$Gender,breast_cancer$Patient_Status)

print(gender_survival)
```

```
##
##          Alive Dead
##   FEMALE   252   61
##   MALE       3    1
```

```
print(chisq.test(gender_survival))
```

```
## Warning in chisq.test(gender_survival): Chi-squared approximation may be
## incorrect
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  gender_survival
## X-squared = 7.8344e-30, df = 1, p-value = 1
```

```
barplot(gender_survival, beside = TRUE, col = c("lightpink", "lightyellow"),
        main = "Gender vs Survival Rate",
        xlab = "Gender", ylab = "Survival")

legend("center", legend = rownames(gender_survival), fill = c("lightpink", "lightyellow"))
```

```
#The results are uninformative considering how small the male sample is. In general, this data set is a

#First, male patients and null values have been removed in a SQL query. Also we will remove irrelevant

#Oversample "Dead" Patients
library(conflicted)
library(dplyr)
BRCA_updated <- read.csv("/Users/gwengorman/Downloads/BRCA_updated - BRCA 2.csv (2).csv", header = TRUE)
View(BRCA_updated)

BRCA_updated$Patient_Status<-as.factor(BRCA_updated$Patient_Status)
prop.table(table(BRCA_updated$Patient_Status))
```

```
## 
##     Alive      Dead 
## 0.7949527 0.2050473
```

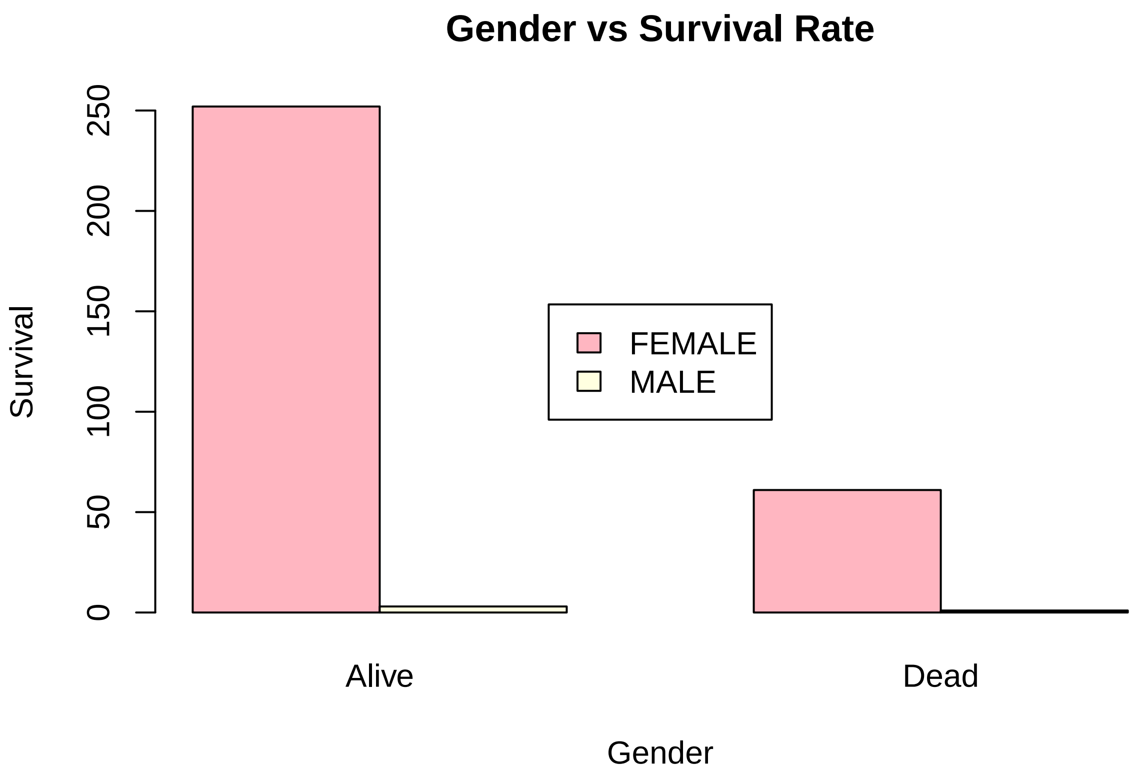
```
summary(BRCA_updated)
```

```
##       Age           Protein1            Protein2          Protein3       
##  Min.   :29.00   Min.   :-2.144600   Min.   :-0.9787   Min.   :-1.62740  
##  1st Qu.:49.00   1st Qu.:-0.361770   1st Qu.: 0.3599   1st Qu.:-0.53136  
##  Median :58.00   Median : 0.003977   Median : 1.0003   Median :-0.17720  
##  Mean   :58.84   Mean   :-0.036148   Mean   : 0.9546   Mean   :-0.09213  
##  3rd Qu.:68.00   3rd Qu.: 0.331860   3rd Qu.: 1.6332   3rd Qu.: 0.28149  
##  Max.   :90.00   Max.   : 1.593600   Max.   : 3.4022   Max.   : 2.19340  
##    Protein4          Tumour_Stage        Histology          HER2.status       
##  Min.   :-2.025500   Length:317         Length:317         Length:317        
##  1st Qu.:-0.382240   Class :character   Class :character   Class :character  
##  Median : 0.040511   Mode  :character   Mode  :character   Mode  :character  
##  Mean   : 0.009829                                                           
##  3rd Qu.: 0.436250                                                           
##  Max.   : 1.629900                                                           
##  Surgery_type       Patient_Status
##  Length:317         Alive:252     
##  Class :character   Dead : 65     
##  Mode  :character                 
##                                   
##                                   
## 
```

```
# To perform machine learning predictions, we will break our sample into a test set and training set.
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```



```
set.seed(317)
index<-createDataPartition(BRCA_updated$Patient_Status,p=0.8,list=FALSE)
train<-BRCA_updated[index,]
test<-BRCA_updated[-index,]

#We will upsample the training set to increase the dead patients
set.seed(317)
trainup<-upSample(x=train[,-ncol(train)],
                  y=train$Patient_Status)

summary(trainup)
```

```
##       Age          Protein1           Protein2          Protein3
##  Min.   :29.00   Min.   :-2.14460   Min.   :-0.9787   Min.   :-1.30710
##  1st Qu.:49.00   1st Qu.:-0.44469   1st Qu.: 0.2875   1st Qu.:-0.49571
##  Median :59.00   Median :-0.04614   Median : 0.9432   Median :-0.17628
##  Mean   :59.37   Mean   :-0.06906   Mean   : 0.9698   Mean   :-0.07038
##  3rd Qu.:68.00   3rd Qu.: 0.32776   3rd Qu.: 1.6806   3rd Qu.: 0.33389
##  Max.   :89.00   Max.   : 1.59360   Max.   : 3.4022   Max.   : 2.19340
##     Protein4        Tumour_Stage        Histology          HER2.status
##  Min.   :-1.76840   Length:404         Length:404         Length:404
##  1st Qu.:-0.26623   Class :character   Class :character   Class :character
##  Median : 0.13805   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 0.08929
##  3rd Qu.: 0.52506
##  Max.   : 1.62990
##  Surgery_type         Class
##  Length:404         Alive:202
##  Class :character   Dead :202
##  Mode  :character
```

```
##
##
##
```

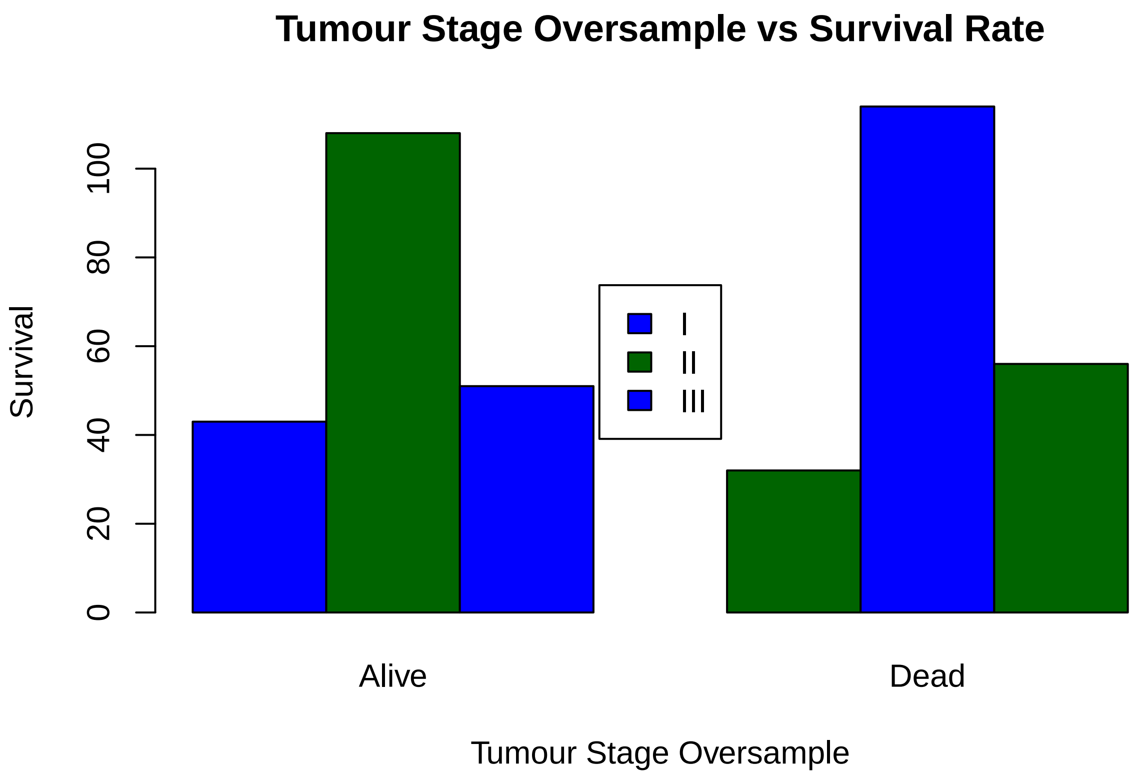
```
#Now our training sample is an equal set of 202 alive patients and 202 dead patients. Let us review the

#TUMOR STAGE vs. SURVIVAL
patient_oversample <- na.omit(trainup)
tumour_survival_oversample = data.frame(patient_oversample$Tumour_Stage,patient_oversample$Class)

tumour_survival_oversample = table(patient_oversample$Tumour_Stage,patient_oversample$Class)

print(tumour_survival_oversample)
```

```
##
##       Alive Dead
##   I      43   32
##   II    108  114
##   III    51   56
```

```
print(chisq.test(tumour_survival_oversample))
```

```
##
##  Pearson's Chi-squared test
##
## data:  tumour_survival_oversample
## X-squared = 2.0091, df = 2, p-value = 0.3662
```

```
barplot(tumour_survival_oversample, beside = TRUE, col = c("blue", "darkgreen"),
        main = "Tumour Stage Oversample vs Survival Rate",
        xlab = "Tumour Stage Oversample", ylab = "Survival")

legend("center", legend = rownames(tumour_survival_oversample), fill = c("blue", "darkgreen"))
```



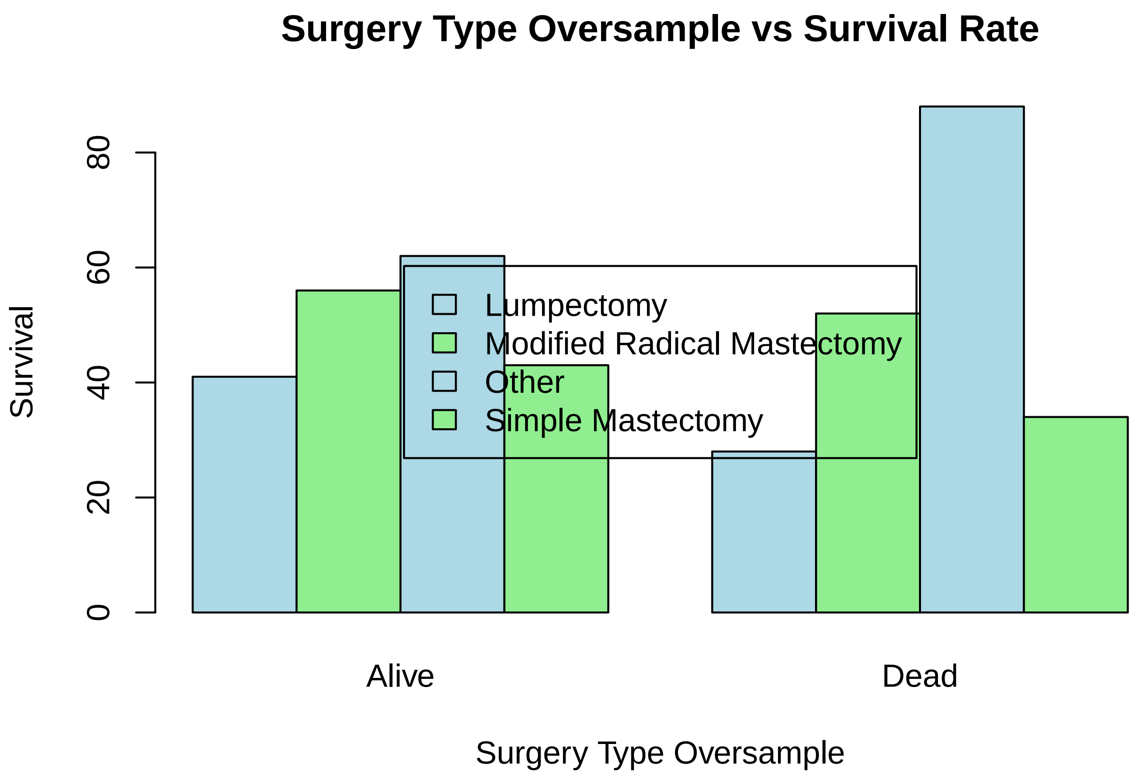
```
#with a p-value of p=0.36 we conclude that tumour stage is independent of survival rate and thus, there

#SURGERY TYPE vs SURVIVAL (chi square)#
cancer_survival_oversample = data.frame(patient_oversample$Surgery_type,patient_oversample$Class)

cancer_survival_oversample = table(patient_oversample$Surgery_type,patient_oversample$Class)

print(cancer_survival_oversample)
```

```
##
##                               Alive Dead
##   Lumpectomy                     41   28
##   Modified Radical Mastectomy    56   52
##   Other                          62   88
##   Simple Mastectomy              43   34
```

```
print(chisq.test(cancer_survival_oversample))
```

```
##
##  Pearson's Chi-squared test
##
## data:  cancer_survival_oversample
## X-squared = 8.156, df = 3, p-value = 0.04289
```

```
barplot(cancer_survival_oversample, beside = TRUE, col = c("lightblue", "lightgreen"),
        main = "Surgery Type Oversample vs Survival Rate",
        xlab = "Surgery Type Oversample", ylab = "Survival")

legend("center", legend = rownames(cancer_survival_oversample), fill = c("lightblue", "lightgreen"))
```



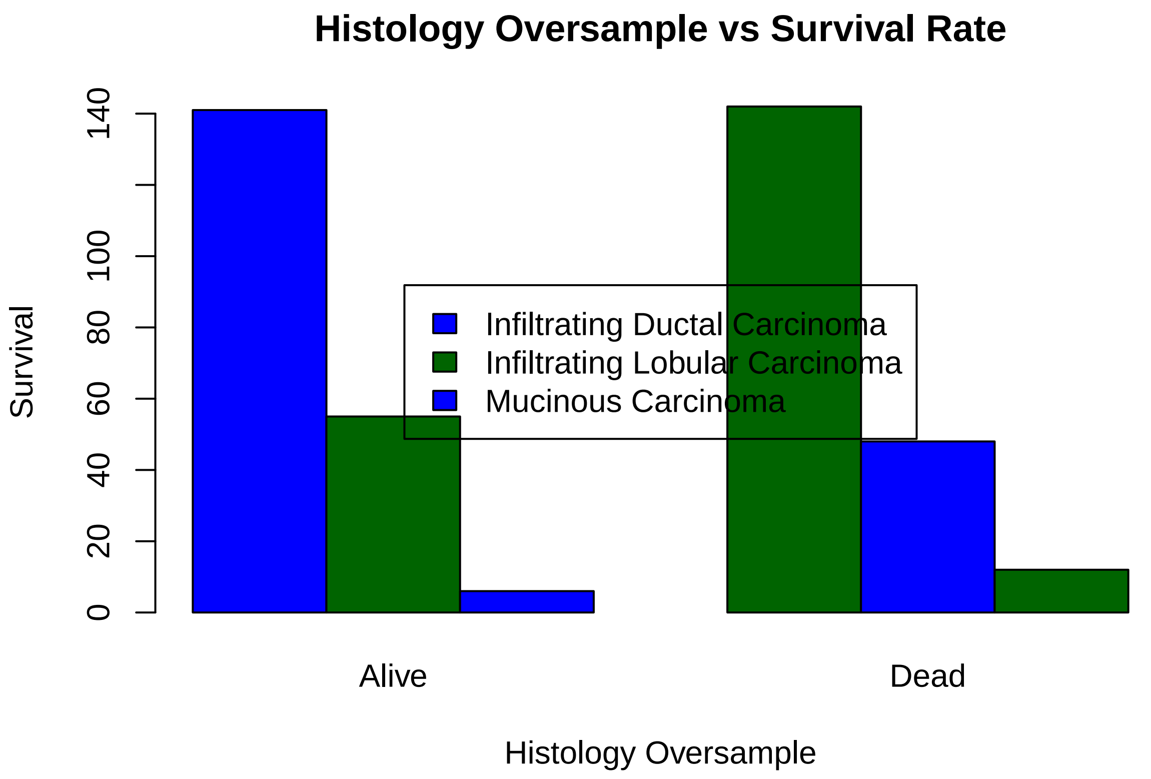
```
    ##With a p-value of .04, we conclude that surgery type has correlation to survival

#HISTOLOGY (chi-square)

histology_survival_oversample = data.frame(patient_oversample$Histology,patient_oversample$Class)

histology_survival_oversample = table(patient_oversample$Histology,patient_oversample$Class)

print(histology_survival_oversample)
```

```
##
##                                   Alive Dead
##   Infiltrating Ductal Carcinoma     141  142
##   Infiltrating Lobular Carcinoma     55   48
##   Mucinous Carcinoma                  6   12
```

```
print(chisq.test(histology_survival_oversample))
```

```
##
##  Pearson's Chi-squared test
##
## data:  histology_survival_oversample
## X-squared = 2.4793, df = 2, p-value = 0.2895
```

```
barplot(histology_survival_oversample, beside = TRUE, col = c("blue", "darkgreen"),
        main = "Histology Oversample vs Survival Rate",
        xlab = "Histology Oversample", ylab = "Survival")

legend("center", legend = rownames(histology_survival_oversample), fill = c("blue", "darkgreen"))
```



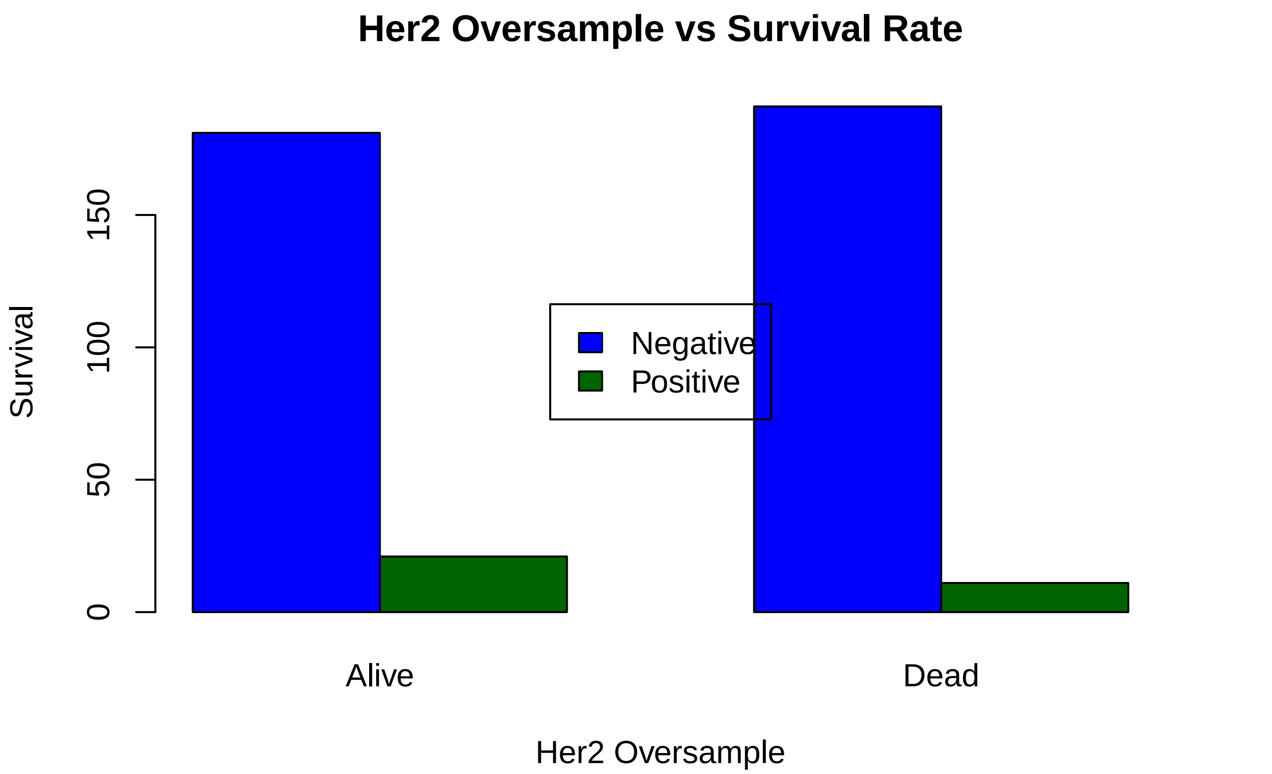
```
#with a p-value of p=0.29 we conclude that tumour stage is independent of survival rate and thus, there


#HER2 (chi-square)
Her2_survival_oversample = data.frame(patient_oversample$HER2.status,patient_oversample$Class)

Her2_survival_oversample = table(patient_oversample$HER2.status,patient_oversample$Class)

print(Her2_survival_oversample)
```

```
##
##              Alive Dead
##   Negative    181  191
##   Positive     21   11
```

```
print(chisq.test(Her2_survival_oversample))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  Her2_survival_oversample
## X-squared = 2.749, df = 1, p-value = 0.09732
```

```
barplot(Her2_survival_oversample, beside = TRUE, col = c("blue", "darkgreen"),
        main = "Her2 Oversample vs Survival Rate",
        xlab = "Her2 Oversample", ylab = "Survival")

legend("center", legend = rownames(Her2_survival_oversample), fill = c("blue", "darkgreen"))
```



```
#with a p-value of p=0.09 we conclude that Her2 is independent of survival rate and thus, there is weak

#Logistic regression

summary(trainup)
```

```
##       Age           Protein1           Protein2          Protein3
##  Min.   :29.00   Min.   :-2.14460   Min.   :-0.9787   Min.   :-1.30710
##  1st Qu.:49.00   1st Qu.:-0.44469   1st Qu.: 0.2875   1st Qu.:-0.49571
##  Median :59.00   Median :-0.04614   Median : 0.9432   Median :-0.17628
##  Mean   :59.37   Mean   :-0.06906   Mean   : 0.9698   Mean   :-0.07038
##  3rd Qu.:68.00   3rd Qu.: 0.32776   3rd Qu.: 1.6806   3rd Qu.: 0.33389
##  Max.   :89.00   Max.   : 1.59360   Max.   : 3.4022   Max.   : 2.19340
##     Protein4        Tumour_Stage        Histology          HER2.status
##  Min.   :-1.76840   Length:404         Length:404         Length:404
##  1st Qu.:-0.26623   Class :character   Class :character   Class :character
##  Median : 0.13805   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 0.08929
##  3rd Qu.: 0.52506
##  Max.   : 1.62990
##  Surgery_type          Class
##  Length:404         Alive:202
##  Class :character   Dead :202
##  Mode  :character
##
##
##
```

```
library(dplyr)
trainup$Tumour_Stage <- recode(trainup$Tumour_Stage, "I" = 1, "II" = 2, "III" = 3)
```

```r
trainup$Histology <- recode(trainup$Histology, "Infiltrating Ductal Carcinoma" = 1, "Mucinous Carcinoma
trainup$HER2.status <- recode(trainup$HER2.status, "Positive" = 1, "Negative" = 0)
trainup$Surgery_type <- recode(trainup$Surgery_type, "Other" = 1, "Lumpectomy" = 2, "Simple Mastectomy"

library(caret)
set.seed(10)
proteinmodelup <- glm(Class~Protein1+Protein2+Protein3+Protein4+Age+Tumour_Stage+Histology+HER2.status,
summary(proteinmodelup)
```

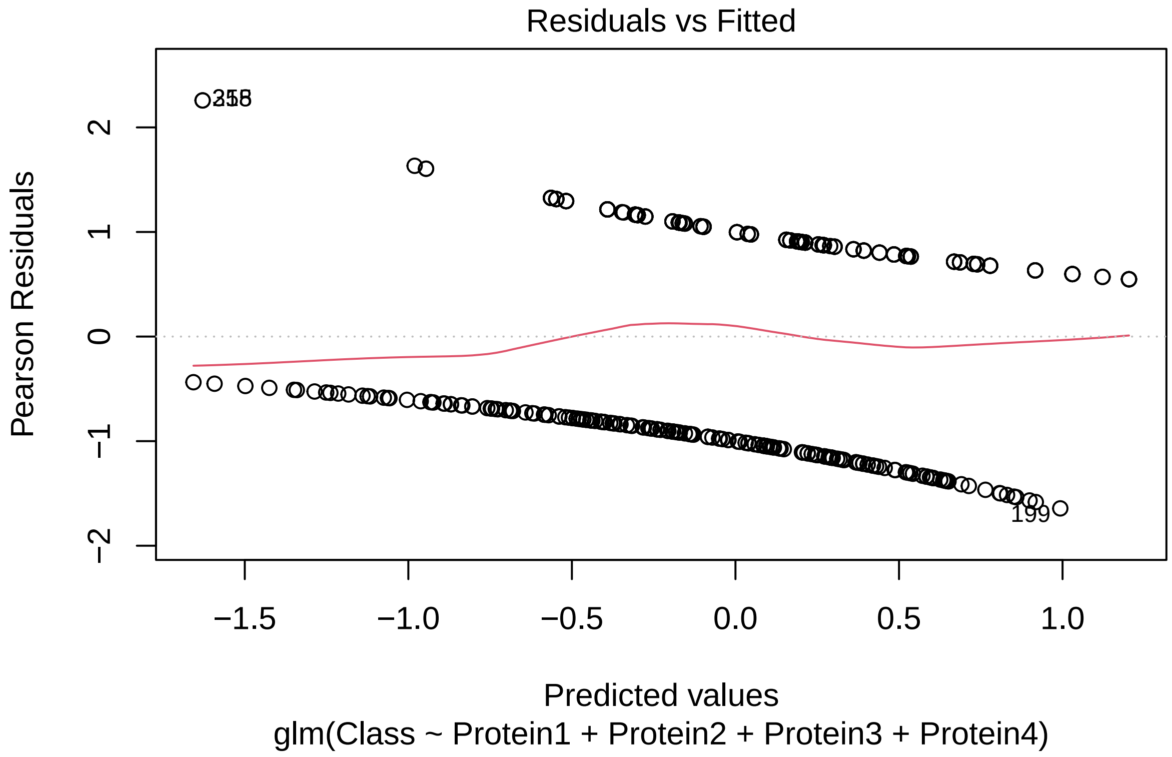
```
##
## Call:
## glm(formula = Class ~ Protein1 + Protein2 + Protein3 + Protein4 +
##     Age + Tumour_Stage + Histology + HER2.status, family = "binomial",
##     data = trainup, weights = Surgery_type)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.372115   0.488461  -2.809 0.004969 **
## Protein1     -0.587397   0.151352  -3.881 0.000104 ***
## Protein2      0.412471   0.089008   4.634 3.59e-06 ***
## Protein3      0.456302   0.133031   3.430 0.000603 ***
## Protein4      0.950313   0.130965   7.256 3.98e-13 ***
## Age           0.004433   0.005638   0.786 0.431721
## Tumour_Stage  0.197668   0.111605   1.771 0.076539 .
## Histology     0.085613   0.080032   1.070 0.284734
## HER2.status  -0.693071   0.272733  -2.541 0.011047 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1316.4  on 403  degrees of freedom
## Residual deviance: 1212.9  on 395  degrees of freedom
## AIC: 1230.9
##
## Number of Fisher Scoring iterations: 4
```

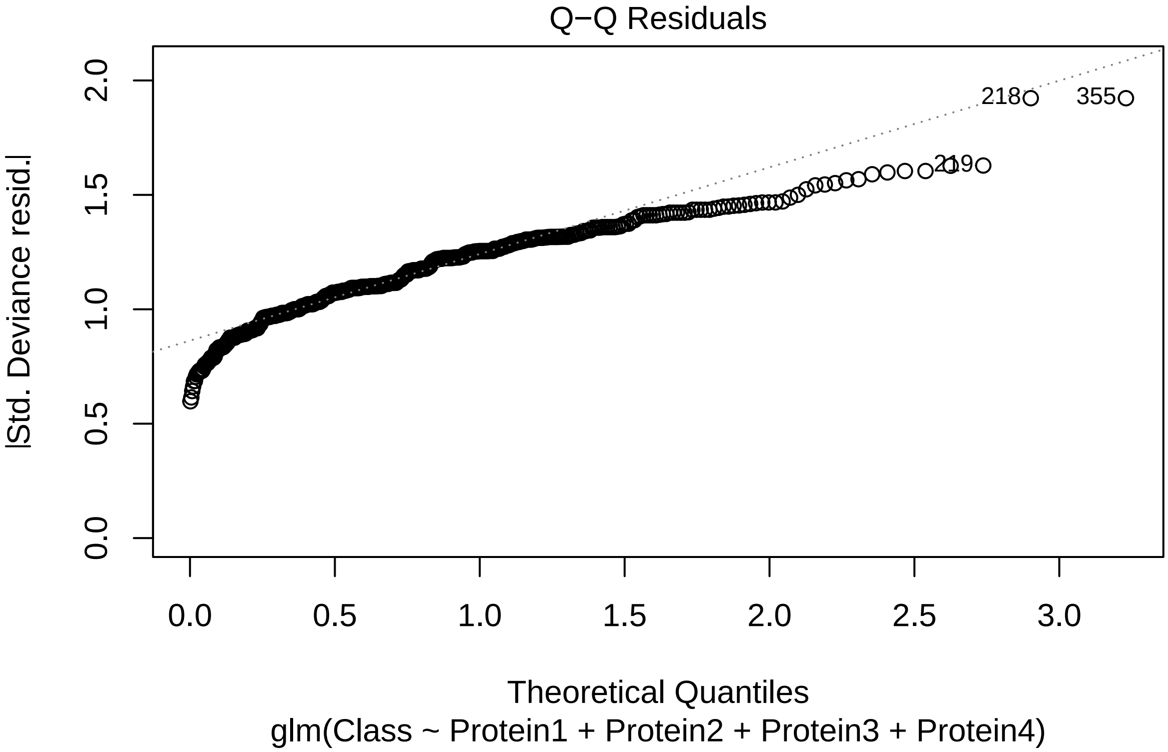
```r
proteinmodelup2 <- glm(Class~Protein1+Protein2+Protein3+Protein4, data=trainup, family = "binomial")
summary(proteinmodelup2)
```

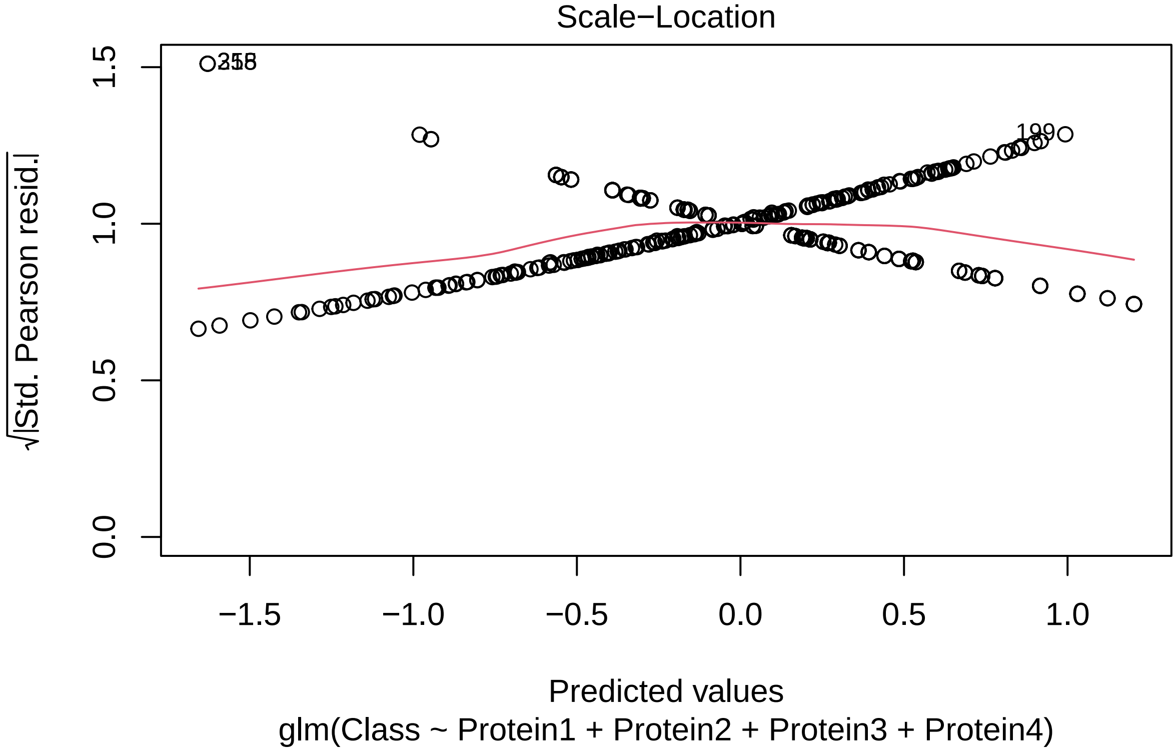
```
##
## Call:
## glm(formula = Class ~ Protein1 + Protein2 + Protein3 + Protein4,
##     family = "binomial", data = trainup)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.4010     0.1645  -2.438   0.0148 *
## Protein1     -0.5420     0.2145  -2.527   0.0115 *
## Protein2      0.3198     0.1291   2.477   0.0133 *
## Protein3      0.3464     0.1949   1.777   0.0755 .
## Protein4      0.8189     0.1964   4.169 3.06e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

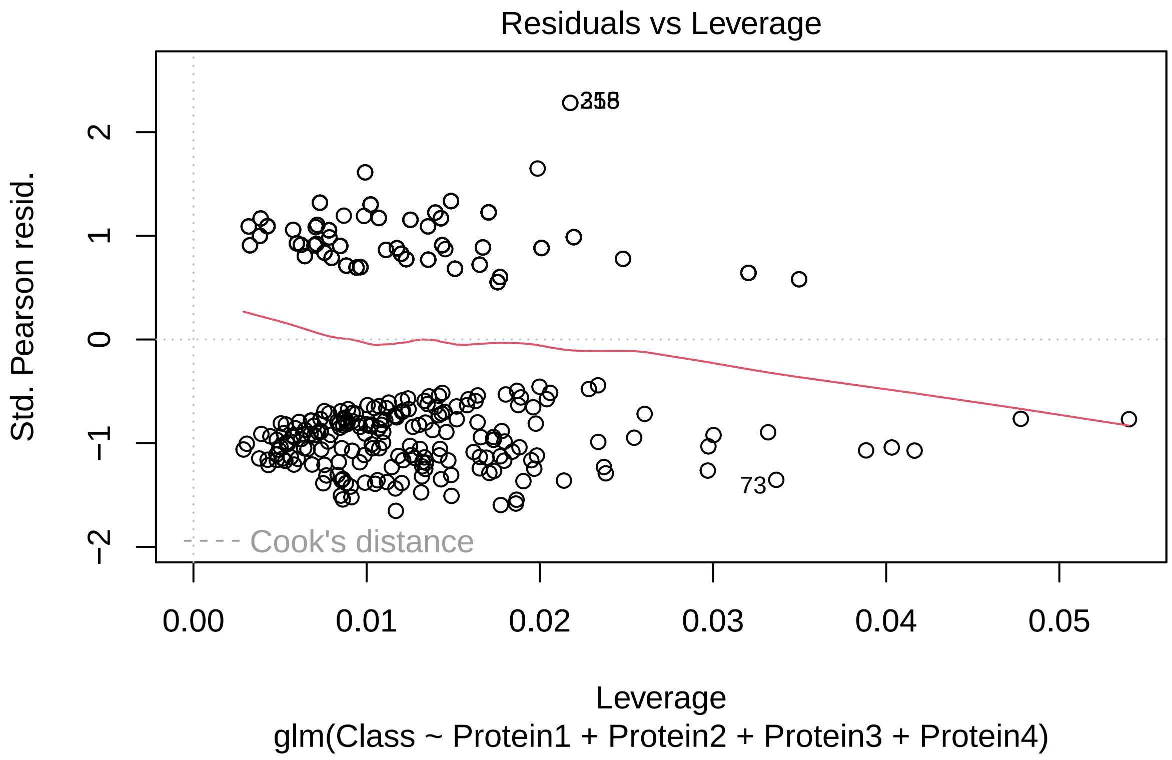
```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 560.06  on 403  degrees of freedom
## Residual deviance: 532.26  on 399  degrees of freedom
## AIC: 542.26
##
## Number of Fisher Scoring iterations: 4
```

```
plot(proteinmodelup2)
```




Q−Q Residuals
|Std. Deviance resid.|
0.0
0.5
1.0
1.5
2.0
218
355
219
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Theoretical Quantiles
glm(Class ~ Protein1 + Protein2 + Protein3 + Protein4)


Scale−Location
√|Std. Pearson resid.|
0.0
0.5
1.0
1.5
218
355
199
−1.5
−1.0
−0.5
0.0
0.5
1.0
Predicted values
glm(Class ~ Protein1 + Protein2 + Protein3 + Protein4)



```
# Convert response variable in test dataset to factor with same levels as training dataset
test$Patient_Status <- factor(test$Patient_Status, levels = levels(trainup$Class))


# Make predictions on the test dataset
pred_prob <- predict(proteinmodelup2, newdata = test, type = "response")

# Convert predicted probabilities to class predictions based on a threshold of 0.5
pred_class <- factor(ifelse(pred_prob > 0.5, "Alive", "Dead"), levels = levels(test$Patient_Status))

# Create confusion matrix
conf_matrix <- confusionMatrix(pred_class, test$Patient_Status)
conf_matrix
```

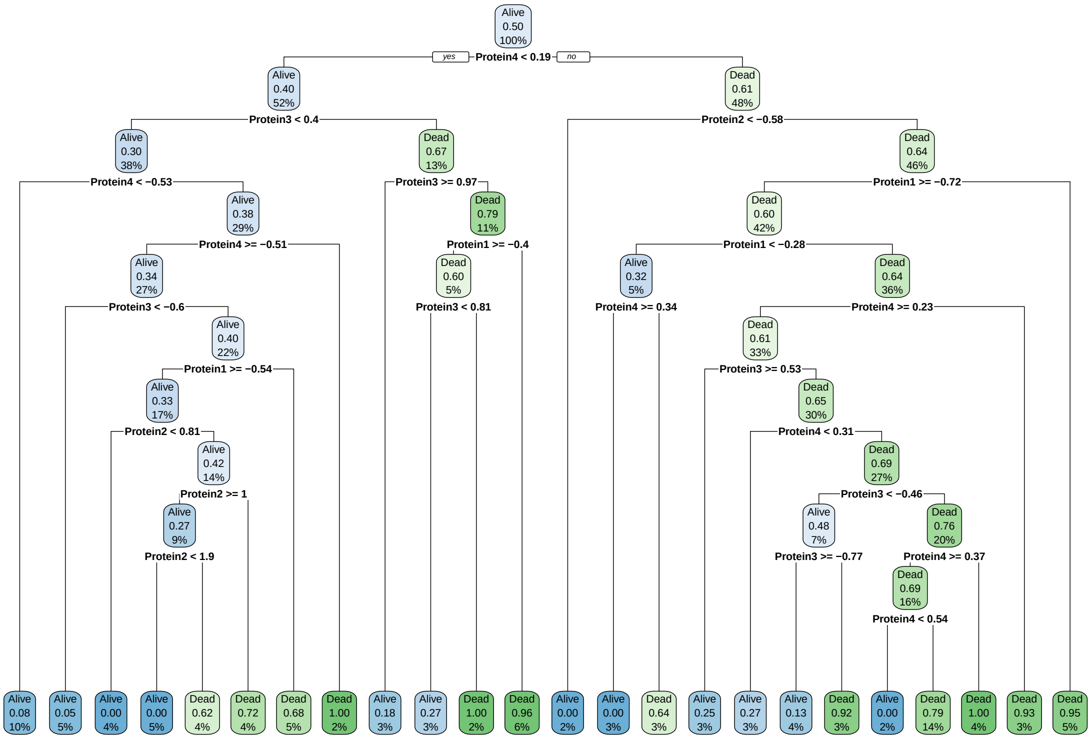
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Alive Dead
##      Alive    24    7
##      Dead     26    6
##
##                Accuracy : 0.4762
##                  95% CI : (0.3488, 0.6059)
##     No Information Rate : 0.7937
##     P-Value [Acc > NIR] : 1.000000
##
##                   Kappa : -0.0379
##
##  Mcnemar's Test P-Value : 0.001728
##
```

```
##             Sensitivity : 0.4800
##             Specificity : 0.4615
##          Pos Pred Value : 0.7742
##          Neg Pred Value : 0.1875
##              Prevalence : 0.7937
##          Detection Rate : 0.3810
##    Detection Prevalence : 0.4921
##       Balanced Accuracy : 0.4708
##
##        'Positive' Class : Alive
##
```

```
#This accuracy is not acceptable. Let's run other predictive models

#DECISION TREE MODEL
library(rpart)
tree_model <- rpart(Class~Protein1+Protein2+Protein3+Protein4, data = trainup)
pred_probtree <- predict(tree_model, newdata = test, type = "class")

confusionMatrix(pred_probtree, test$Patient_Status)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Alive Dead
##      Alive    37    7
##      Dead     13    6
##
##                Accuracy : 0.6825
##                  95% CI : (0.5531, 0.7942)
##     No Information Rate : 0.7937
##     P-Value [Acc > NIR] : 0.9872
##
##                   Kappa : 0.1721
##
##  Mcnemar's Test P-Value : 0.2636
##
##             Sensitivity : 0.7400
##             Specificity : 0.4615
##          Pos Pred Value : 0.8409
##          Neg Pred Value : 0.3158
##              Prevalence : 0.7937
##          Detection Rate : 0.5873
##    Detection Prevalence : 0.6984
##       Balanced Accuracy : 0.6008
##
##        'Positive' Class : Alive
##
```

```
#Here we see the accuracy increase from 47% to 68% using the significant variables in a decision tree#

library(rpart.plot)
rpart.plot(tree_model)
```



```
library(randomForest)
```

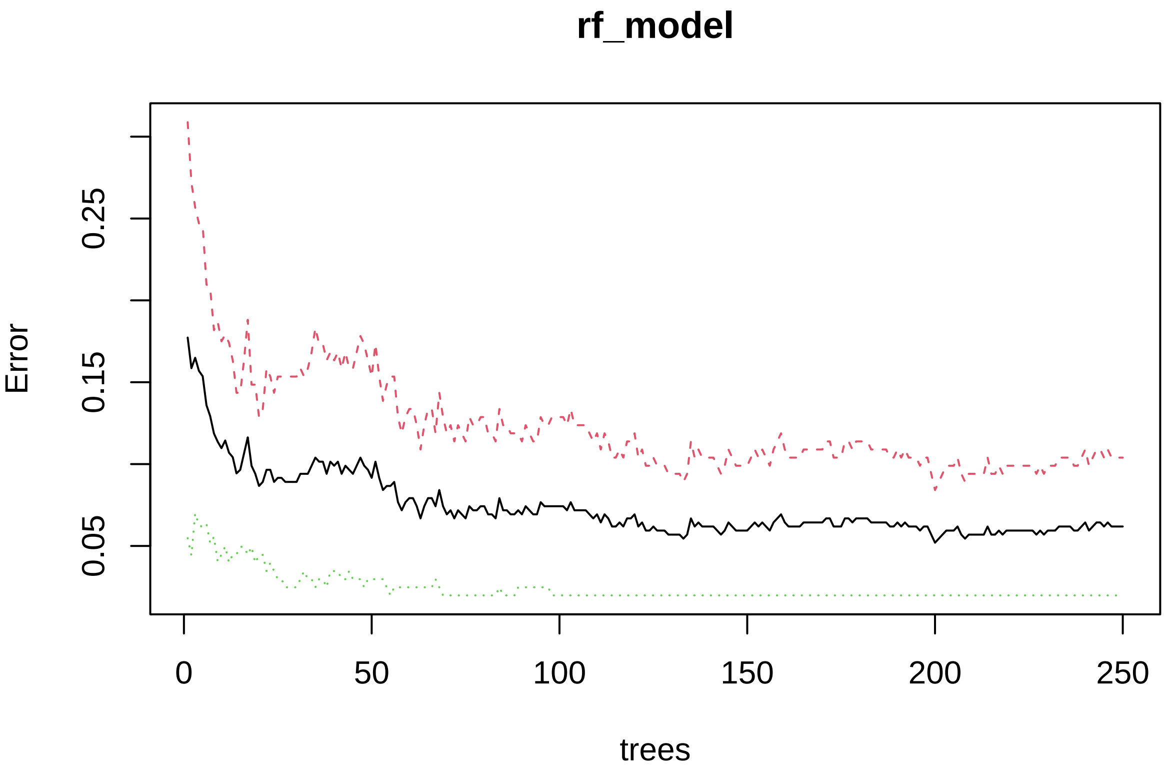
```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
rf_model <- randomForest(Class~Protein1+Protein2+Protein3+Protein4,data = trainup, ntree = 250, mtry = 
predictions <- predict(rf_model, newdata = test)
confusionMatrix(predictions, test$Patient_Status)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Alive Dead
##      Alive    44    9
##      Dead      6    4
##
##                Accuracy : 0.7619
##                  95% CI : (0.6379, 0.8602)
##     No Information Rate : 0.7937
##     P-Value [Acc > NIR] : 0.7854
##
##                   Kappa : 0.2052
##
##  Mcnemar's Test P-Value : 0.6056
##
##             Sensitivity : 0.8800
##             Specificity : 0.3077
##          Pos Pred Value : 0.8302
##          Neg Pred Value : 0.4000
##              Prevalence : 0.7937
##          Detection Rate : 0.6984
##    Detection Prevalence : 0.8413
##       Balanced Accuracy : 0.5938
##
##        'Positive' Class : Alive
##
```

```
plot(rf_model)
```



```
#Overall, the Random Forest has an acceptable accuracy of 76%. The model seems to perform moderately we
```