Ethereum Fraud Detection

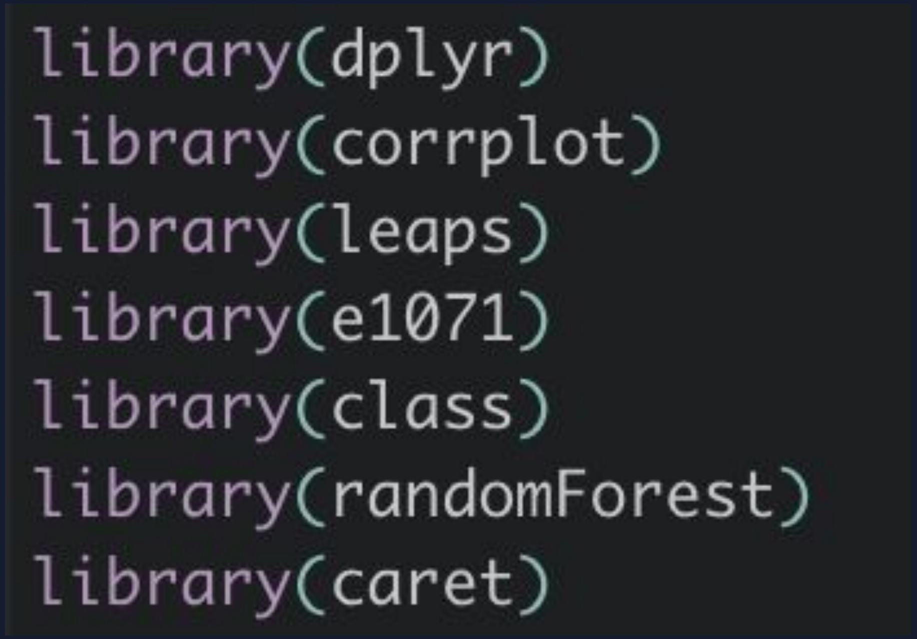
Arjun Chanda| Jai Agrawal | Ambika Mudigonda | Qiuhan Li | Sahithi Muddana

APPENDIX

1. Preprocessing:
   - Libraries
   - Cleaning
   - Balancing Data
2. Models used:
   - Logistic Regression
   - Naive Bayes
   - KNN
   - Decision Tree
   - Random Forest

LIBRARIES

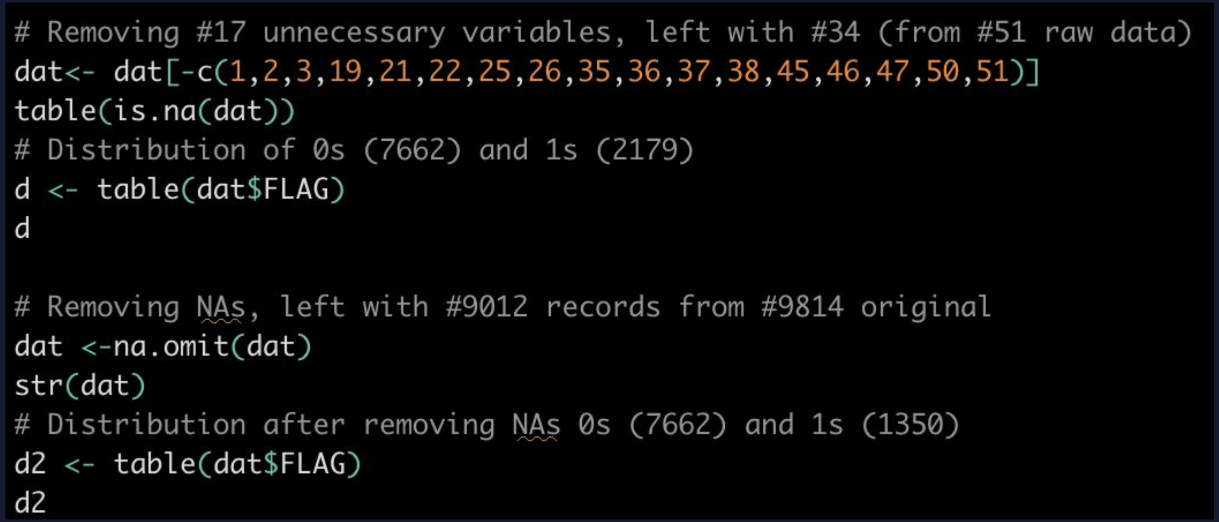
- dplyr
- corrplot
- leaps
- e1071
- class
- randomForest
- caret
- tree



```
library(dplyr)
library(corrplot)
library(leaps)
library(e1071)
library(class)
library(randomForest)
library(caret)
```

CLEANING

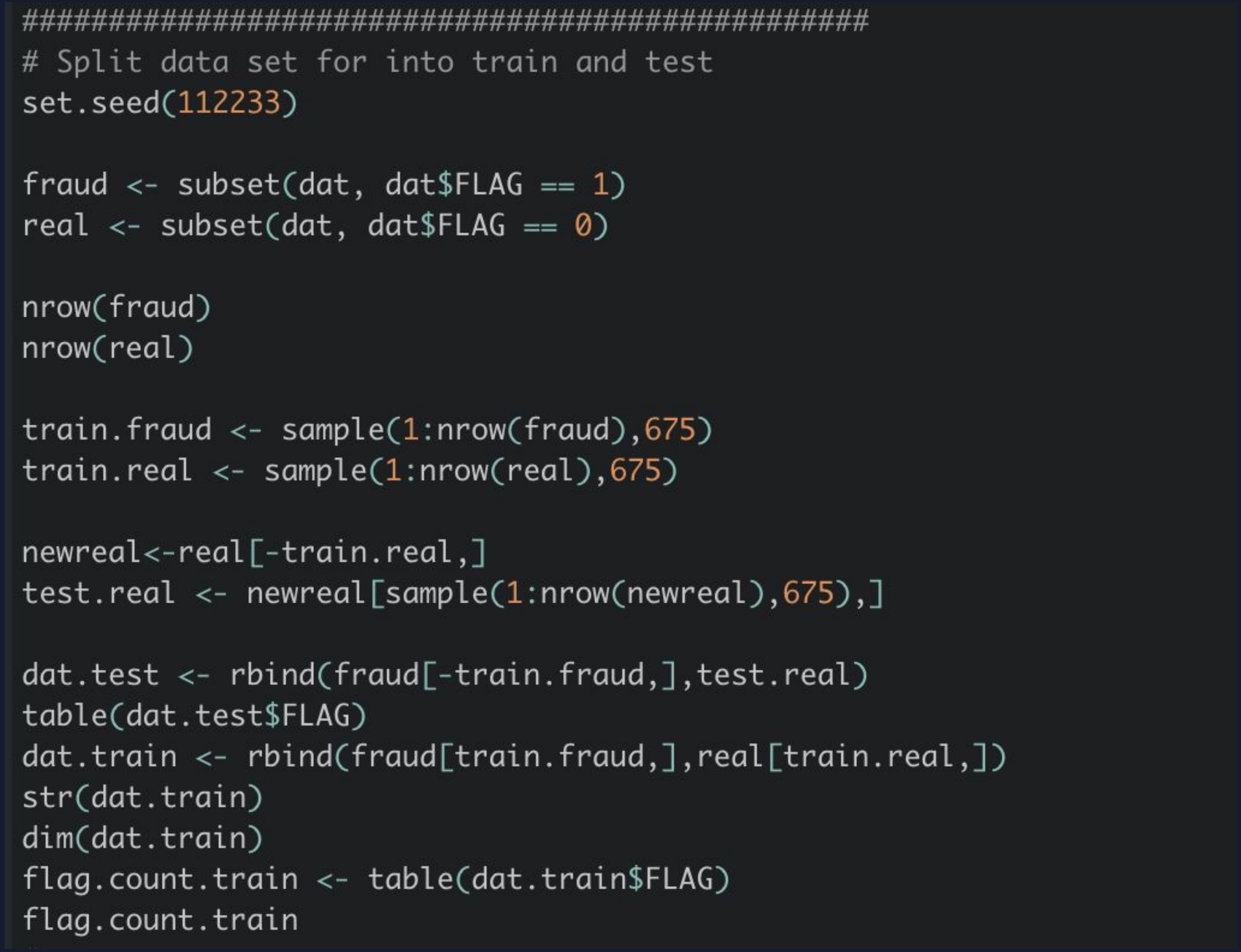
- Removing NA variables
- Removing unnecessary variables (ex. Address)
- Removing highly correlated variables
- Remaining 34 variables 9012 observations



```
# Removing #17 unnecessary variables, left with #34 (from #51 raw data)
dat<- dat[-c(1,2,3,19,21,22,25,26,35,36,37,38,45,46,47,50,51)]
table(is.na(dat))
# Distribution of 0s (7662) and 1s (2179)
d <- table(dat$FLAG)
d

# Removing NAs, left with #9012 records from #9814 original
dat <-na.omit(dat)
str(dat)
# Distribution after removing NAs 0s (7662) and 1s (1350)
d2 <- table(dat$FLAG)
d2
```

BALANCING DATA

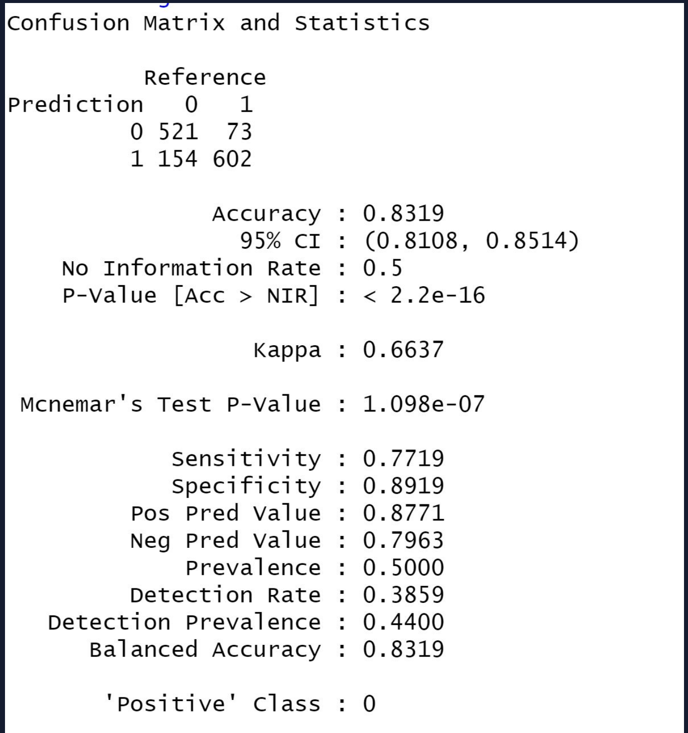
- Since our dataset was imbalance (in favor or real transactions), we made the test train split contain the same amount of real and fraud data so that our models could not get high accuracy just by predicting all outcomes to be real
- Train size: 1350 (equal 0s,1s)
- Test size: 1350 (equal 0s,1s)



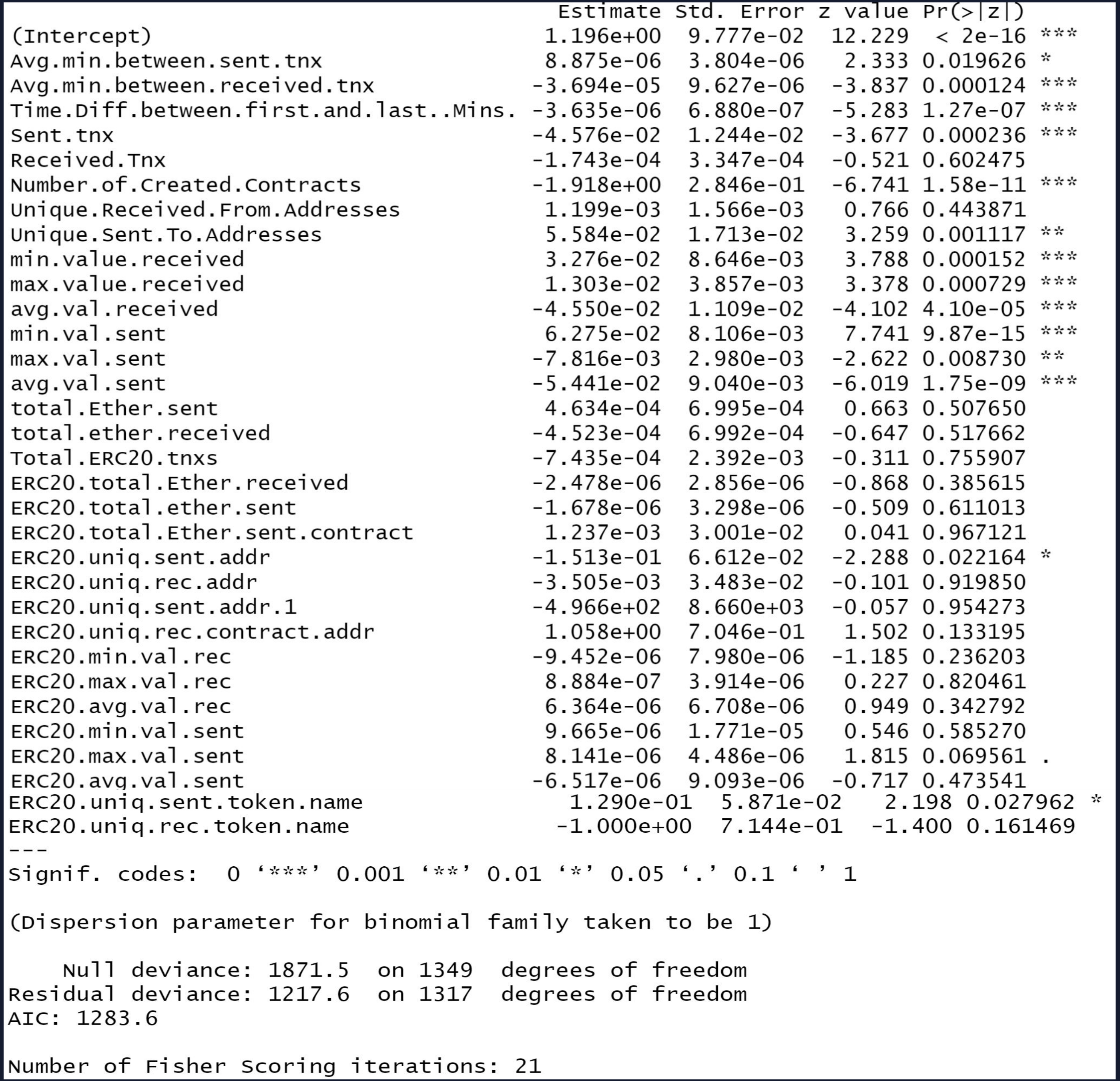
```
#################################################
# Split data set for into train and test
set.seed(112233)

fraud <- subset(dat, dat$FLAG == 1)
real <- subset(dat, dat$FLAG == 0)

nrow(fraud)
nrow(real)

train.fraud <- sample(1:nrow(fraud),675)
train.real <- sample(1:nrow(real),675)

newreal<-real[-train.real,]
test.real <- newreal[sample(1:nrow(newreal),675),]

dat.test <- rbind(fraud[-train.fraud,],test.real)
table(dat.test$FLAG)
dat.train <- rbind(fraud[train.fraud,],real[train.real,])
str(dat.train)
dim(dat.train)
flag.count.train <- table(dat.train$FLAG)
flag.count.train
```

LOGISTIC REGRESSION

- 16 significant variables
- Accuracy = 83.19%
- Precision = 87.71%
- Recall = 77.19%



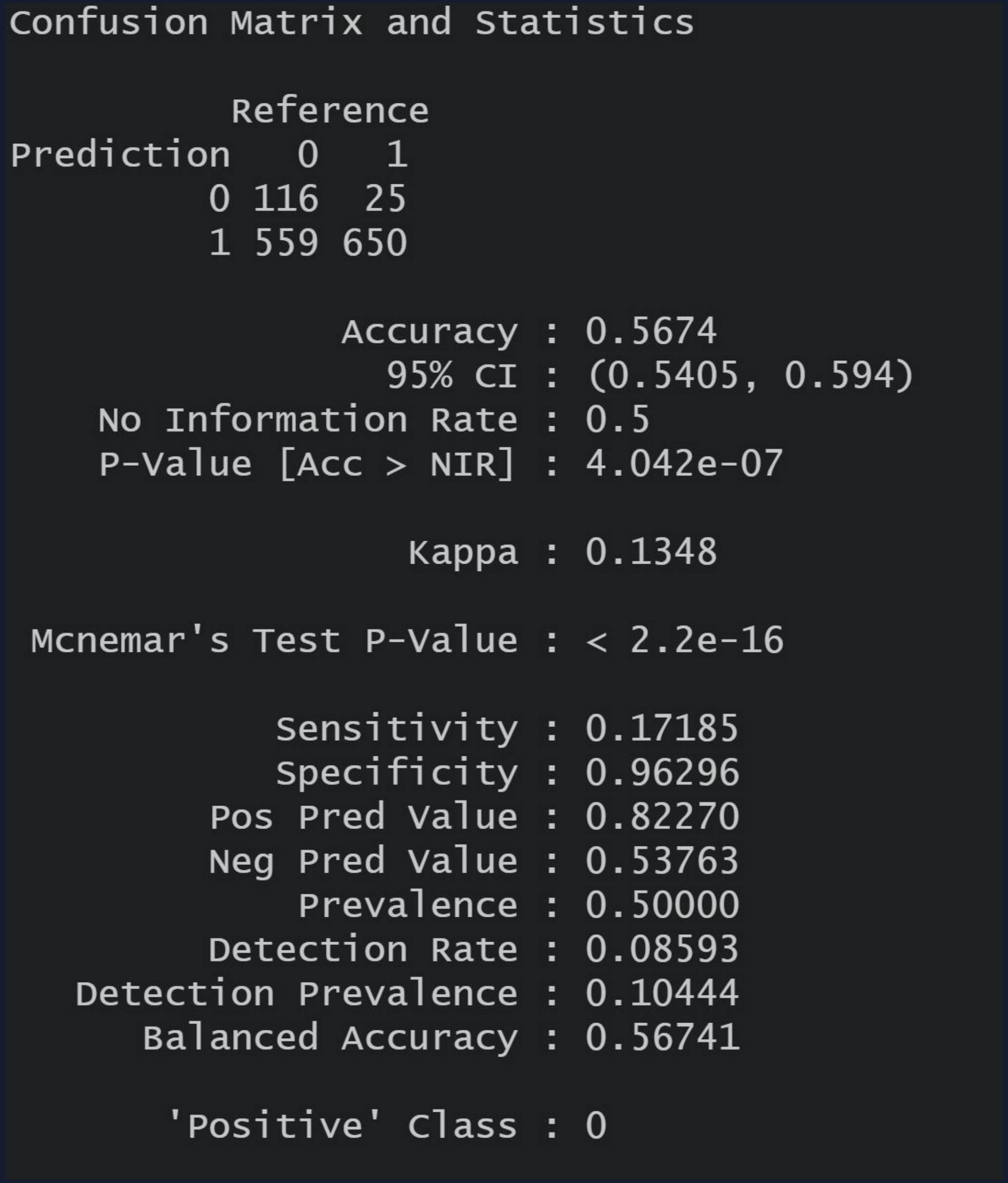
```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 521  73
         1 154 602

               Accuracy : 0.8319
                 95% CI : (0.8108, 0.8514)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6637

 Mcnemar's Test P-Value : 1.098e-07

            Sensitivity : 0.7719
            Specificity : 0.8919
         Pos Pred Value : 0.8771
         Neg Pred Value : 0.7963
             Prevalence : 0.5000
         Detection Rate : 0.3859
   Detection Prevalence : 0.4400
      Balanced Accuracy : 0.8319

       'Positive' Class : 0
```



```
                                          Estimate Std. Error z value Pr(>|z|)    
(Intercept)                              1.196e+00  9.777e-02  12.229  < 2e-16 ***
Avg.min.between.sent.tnx                 8.875e-06  3.804e-06   2.333 0.019626 *  
Avg.min.between.received.tnx            -3.694e-05  9.627e-06  -3.837 0.000124 ***
Time.Diff.between.first.and.last..Mins. -3.635e-06  6.880e-07  -5.283 1.27e-07 ***
Sent.tnx                                -4.576e-02  1.244e-02  -3.677 0.000236 ***
Received.Tnx                            -1.743e-04  3.347e-04  -0.521 0.602475    
Number.of.Created.Contracts             -1.918e+00  2.846e-01  -6.741 1.58e-11 ***
Unique.Received.From.Addresses           1.199e-03  1.566e-03   0.766 0.443871    
Unique.Sent.To.Addresses                 5.584e-02  1.713e-02   3.259 0.001117 ** 
min.value.received                       3.276e-02  8.646e-03   3.788 0.000152 ***
max.value.received                       1.303e-02  3.857e-03   3.378 0.000729 ***
avg.val.received                        -4.550e-02  1.109e-02  -4.102 4.10e-05 ***
min.val.sent                             6.275e-02  8.106e-03   7.741 9.87e-15 ***
max.val.sent                            -7.816e-03  2.980e-03  -2.622 0.008730 ** 
avg.val.sent                            -5.441e-02  9.040e-03  -6.019 1.75e-09 ***
total.Ether.sent                         4.634e-04  6.995e-04   0.663 0.507650    
total.ether.received                    -4.523e-04  6.992e-04  -0.647 0.517662    
Total.ERC20.tnxs                        -7.435e-04  2.392e-03  -0.311 0.755907    
ERC20.total.Ether.received              -2.478e-06  2.856e-06  -0.868 0.385615    
ERC20.total.ether.sent                  -1.678e-06  3.298e-06  -0.509 0.611013    
ERC20.total.Ether.sent.contract          1.237e-03  3.001e-02   0.041 0.967121    
ERC20.uniq.sent.addr                    -1.513e-01  6.612e-02  -2.288 0.022164 *  
ERC20.uniq.rec.addr                     -3.505e-03  3.483e-02  -0.101 0.919850    
ERC20.uniq.sent.addr.1                  -4.966e+02  8.660e+03  -0.057 0.954273    
ERC20.uniq.rec.contract.addr             1.058e+00  7.046e-01   1.502 0.133195    
ERC20.min.val.rec                       -9.452e-06  7.980e-06  -1.185 0.236203    
ERC20.max.val.rec                        8.884e-07  3.914e-06   0.227 0.820461    
ERC20.avg.val.rec                        6.364e-06  6.708e-06   0.949 0.342792    
ERC20.min.val.sent                       9.665e-06  1.771e-05   0.546 0.585270    
ERC20.max.val.sent                       8.141e-06  4.486e-06   1.815 0.069561 .  
ERC20.avg.val.sent                      -6.517e-06  9.093e-06  -0.717 0.473541    
ERC20.uniq.sent.token.name                 1.290e-01  5.871e-02   2.198 0.027962 *
ERC20.uniq.rec.token.name                 -1.000e+00  7.144e-01  -1.400 0.161469
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1871.5  on 1349  degrees of freedom
Residual deviance: 1217.6  on 1317  degrees of freedom
AIC: 1283.6

Number of Fisher Scoring iterations: 21
```

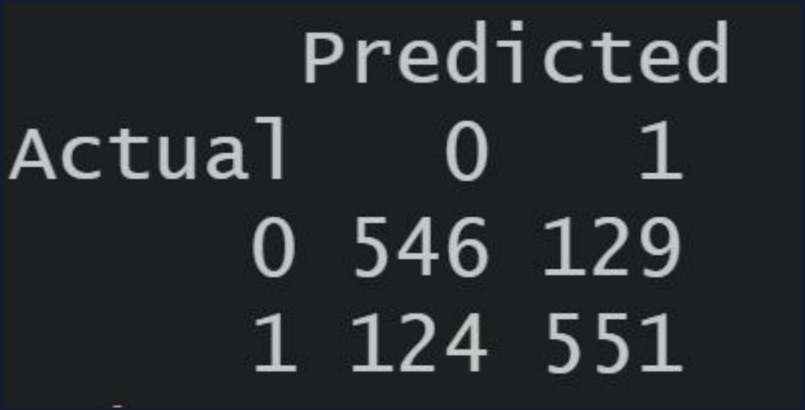
NAIVE BAYES

- Accuracy = 56.74%
- Precision = 82.27%
- Recall = 17.19%
- Naive Bayes performs the worst because it is susceptible to Bayesian poisoning i.e. given the dependent variable the features are not independent.



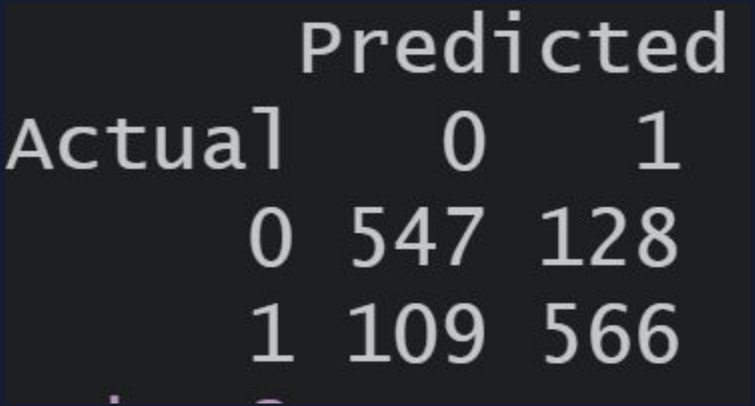
```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 116  25
         1 559 650
                                          
               Accuracy : 0.5674          
                 95% CI : (0.5405, 0.594)
    No Information Rate : 0.5             
    P-Value [Acc > NIR] : 4.042e-07       
                                          
                  Kappa : 0.1348          
                                          
 Mcnemar's Test P-Value : < 2.2e-16       
                                          
            Sensitivity : 0.17185         
            Specificity : 0.96296         
         Pos Pred Value : 0.82270         
         Neg Pred Value : 0.53763         
             Prevalence : 0.50000         
         Detection Rate : 0.08593         
   Detection Prevalence : 0.10444         
      Balanced Accuracy : 0.56741         
                                          
       'Positive' Class : 0               
```

KNN



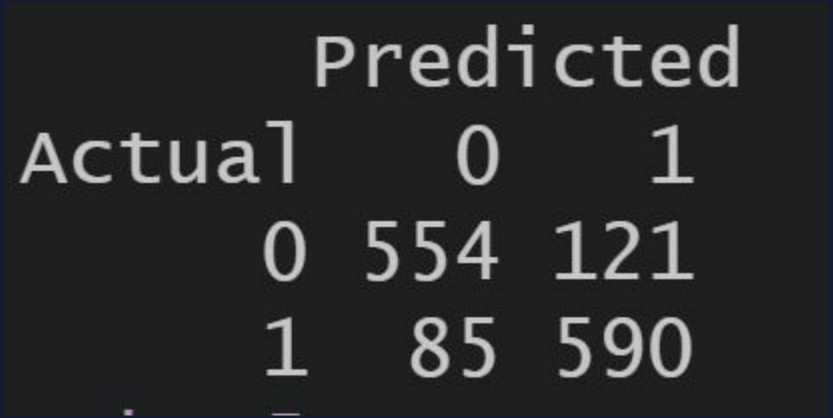
```
      Predicted
Actual   0   1
     0 546 129
     1 124 551
```

K=1; Error = 18.7%



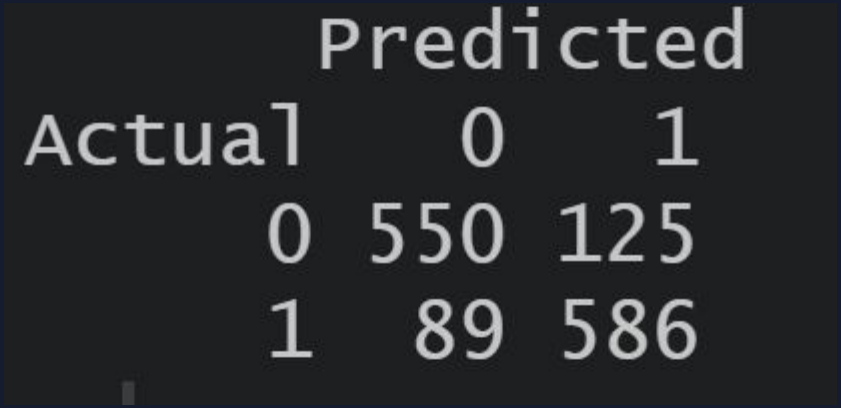
```
      Predicted
Actual   0   1
     0 547 128
     1 109 566
```

K=2; Error = 17.5%



```
      Predicted
Actual   0   1
     0 554 121
     1  85 590
```

K=3; Error = 15.25%

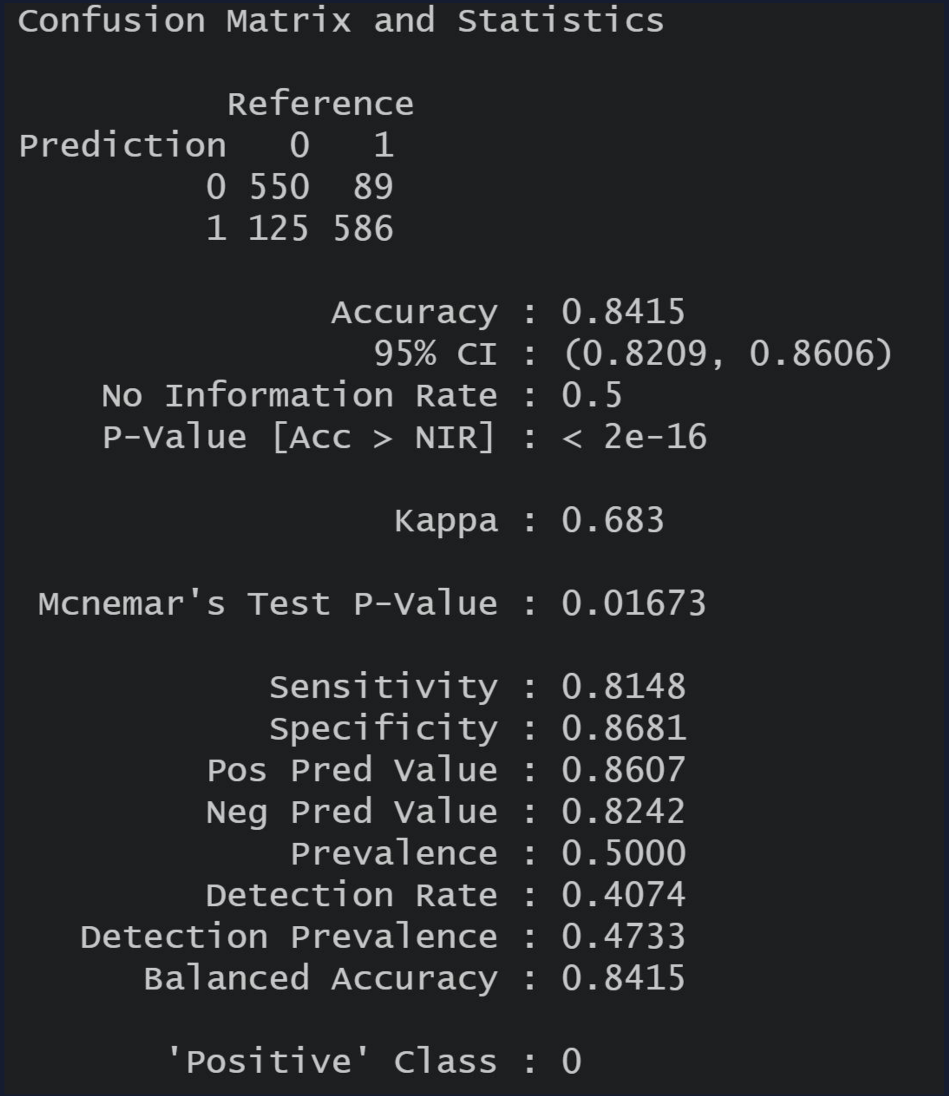


```
      Predicted
Actual   0   1
     0 550 125
     1  89 586
```

K=2; Error = 15.85%

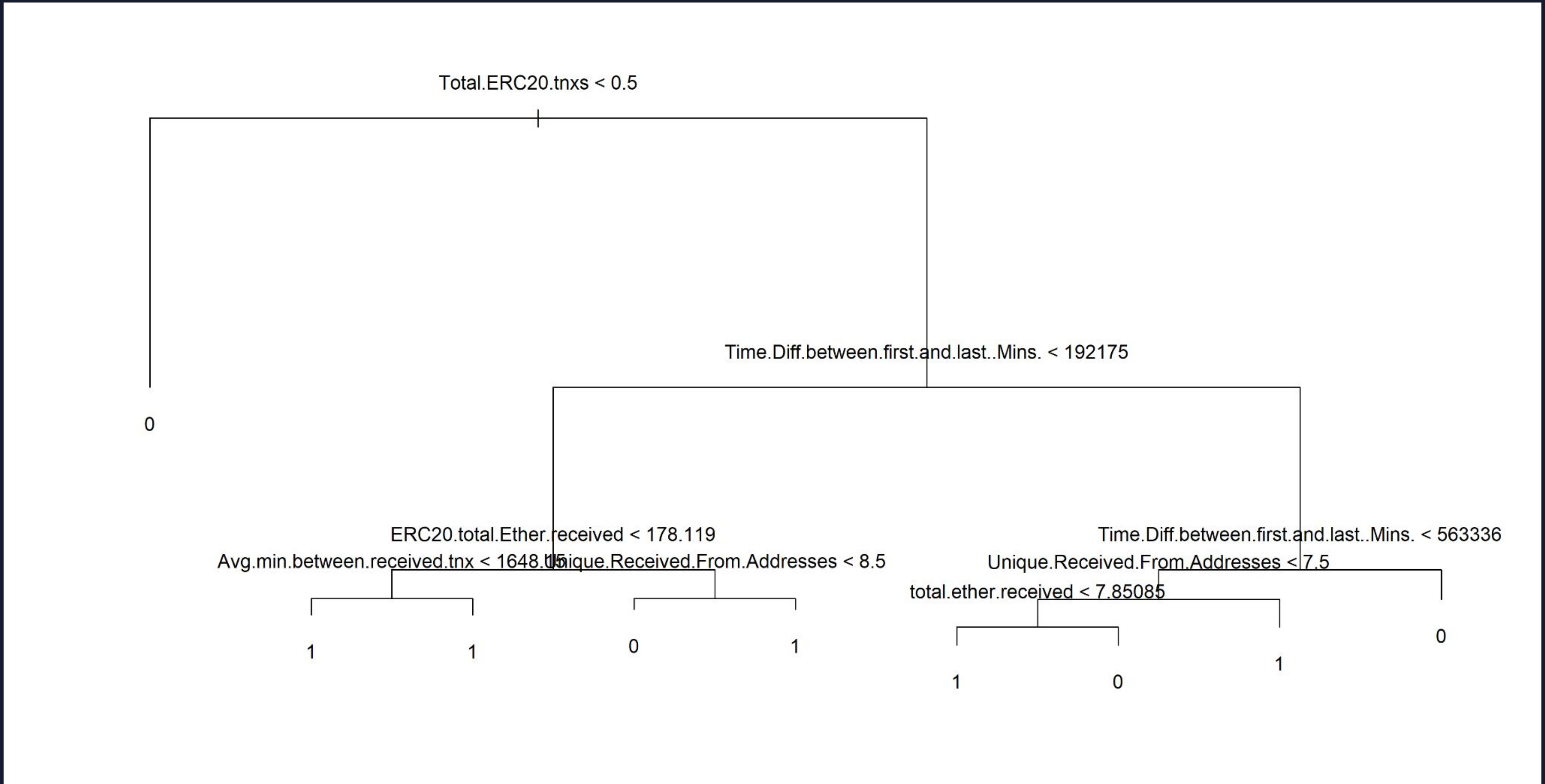
KNN (Contd..)

- K = 5 performed the best
- Accuracy = 84.15%
- Precision = 86.07%
- Recall = 81.48 %



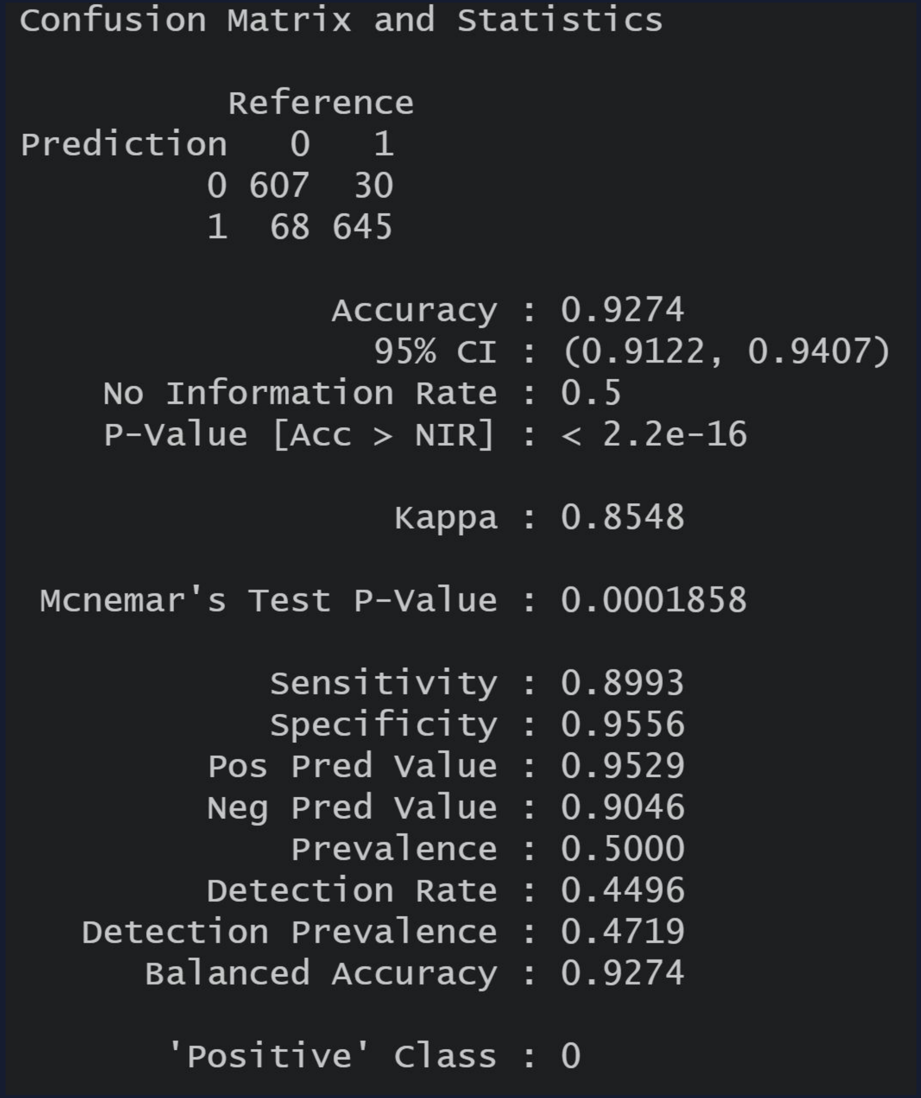
```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 550  89
         1 125 586

               Accuracy : 0.8415
                 95% CI : (0.8209, 0.8606)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2e-16

                  Kappa : 0.683

 Mcnemar's Test P-Value : 0.01673

            Sensitivity : 0.8148
            Specificity : 0.8681
         Pos Pred Value : 0.8607
         Neg Pred Value : 0.8242
             Prevalence : 0.5000
         Detection Rate : 0.4074
   Detection Prevalence : 0.4733
      Balanced Accuracy : 0.8415

       'Positive' Class : 0
```

CLASSIFICATION TREE


Total.ERC20.tnxs < 0.5
0
Time.Diff.between.first.and.last..Mins. < 192175
ERC20.total.Ether.received < 178.119
Avg.min.between.received.tnx < 1648.15
Unique.Received.From.Addresses < 8.5
1
1
0
1
Time.Diff.between.first.and.last..Mins. < 563336
Unique.Received.From.Addresses < 7.5
total.ether.received < 7.85085
1
0
1
0

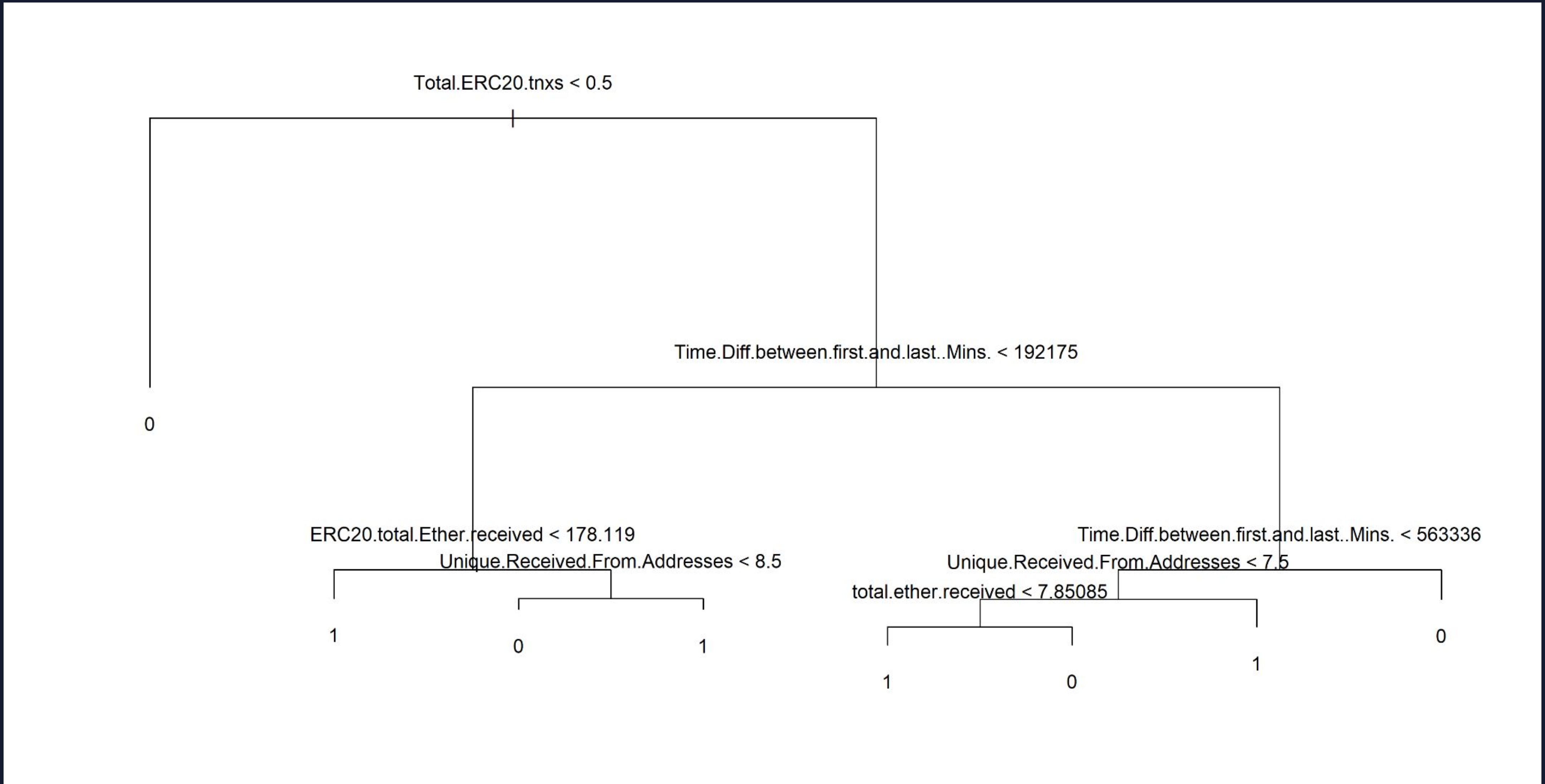
CLASSIFICATION TREE RESULTS

- Accuracy = 92.74%
- Precision = 95.29%
- Recall = 89.93%



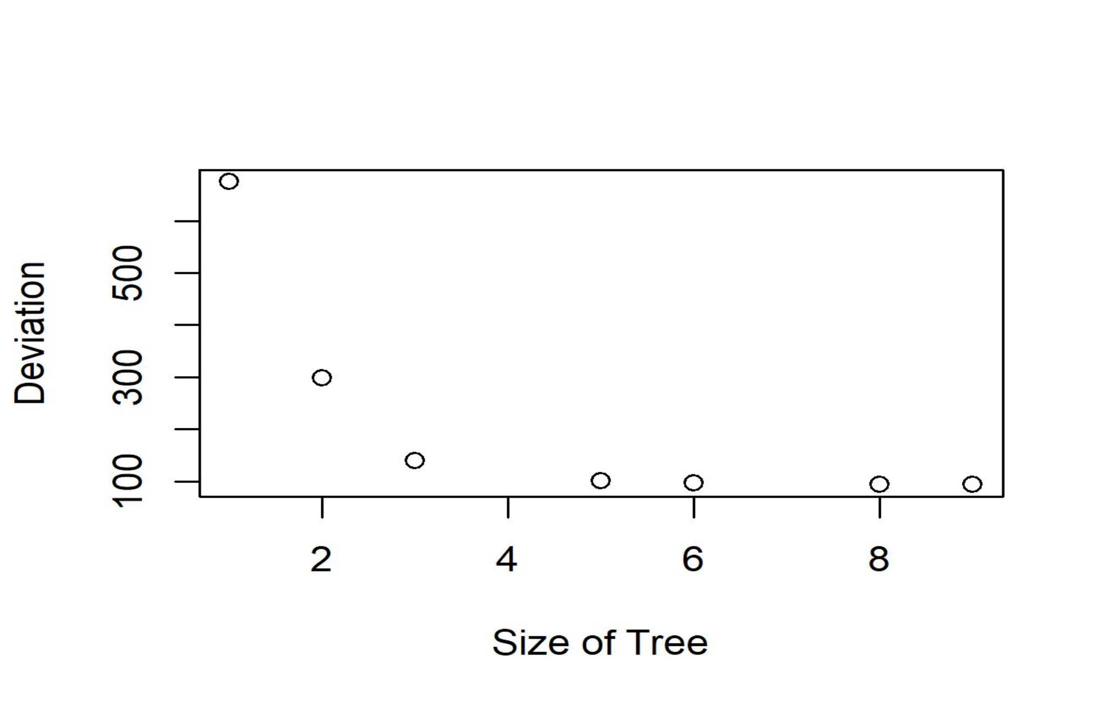
```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 607  30
         1  68 645

               Accuracy : 0.9274
                 95% CI : (0.9122, 0.9407)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8548

 Mcnemar's Test P-Value : 0.0001858

            Sensitivity : 0.8993
            Specificity : 0.9556
         Pos Pred Value : 0.9529
         Neg Pred Value : 0.9046
             Prevalence : 0.5000
         Detection Rate : 0.4496
   Detection Prevalence : 0.4719
      Balanced Accuracy : 0.9274

       'Positive' Class : 0
```

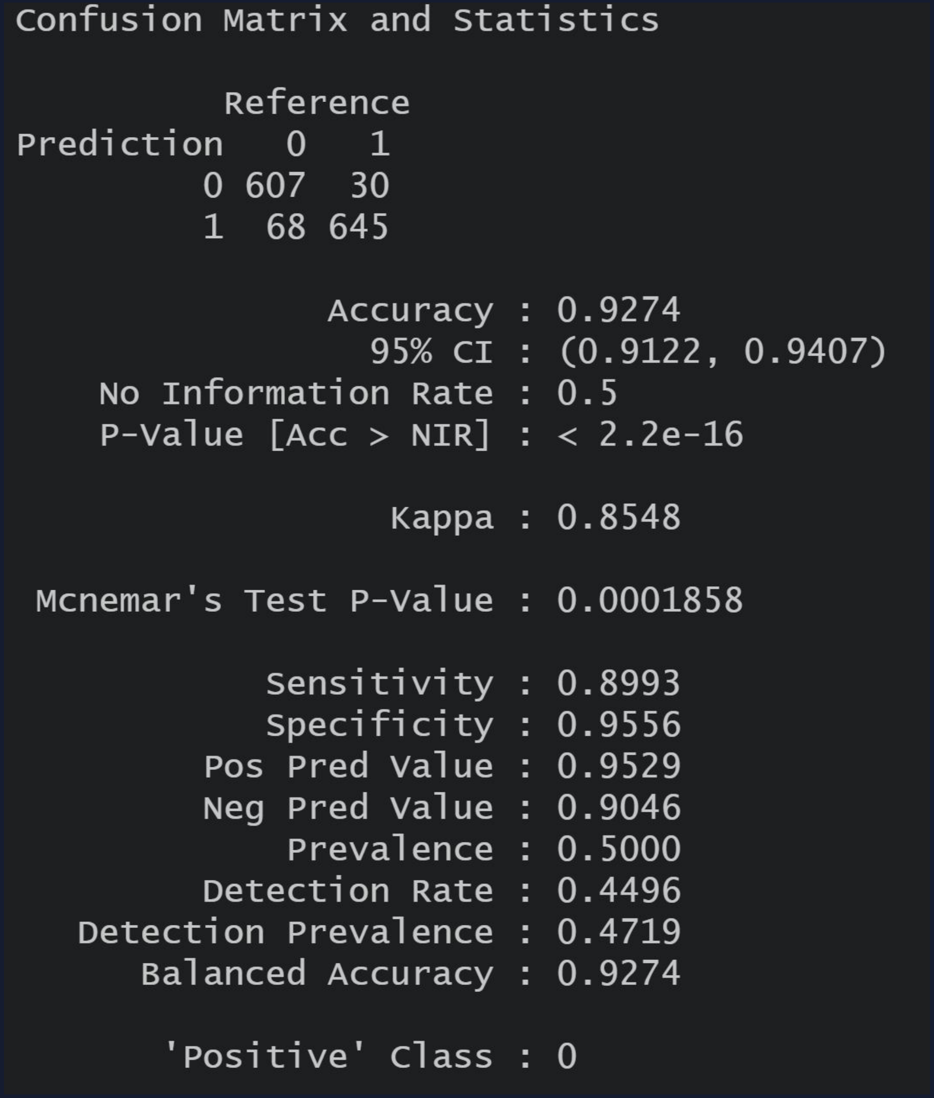
PRUNED CLASSIFICATION TREE


Total.ERC20.tnxs < 0.5
0
Time.Diff.between.first.and.last..Mins. < 192175
ERC20.total.Ether.received < 178.119
Unique.Received.From.Addresses < 8.5
1
0
1
Time.Diff.between.first.and.last..Mins. < 563336
Unique.Received.From.Addresses < 7.5
total.ether.received < 7.85085
1
0
1
0

PRUNE CLASSIFICATION TREE RESULTS

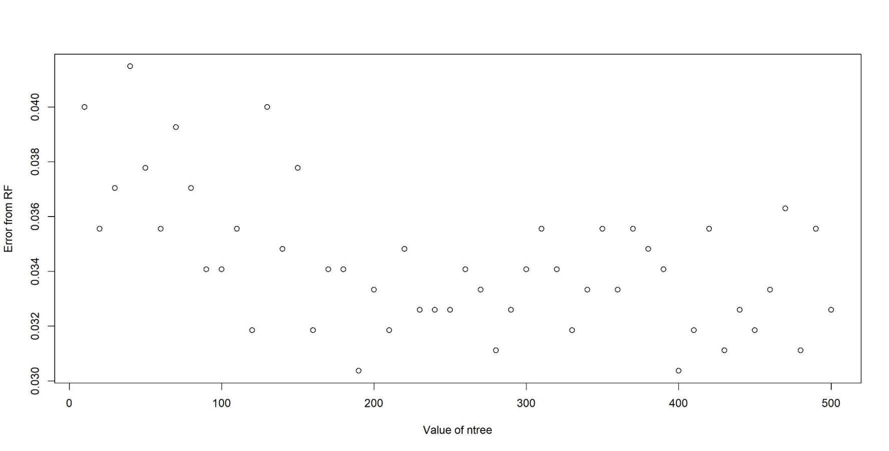
- Pruned the tree from 8 to 7
- Accuracy = 92.74%
- Precision = 95.29%
- Recall = 89.93%

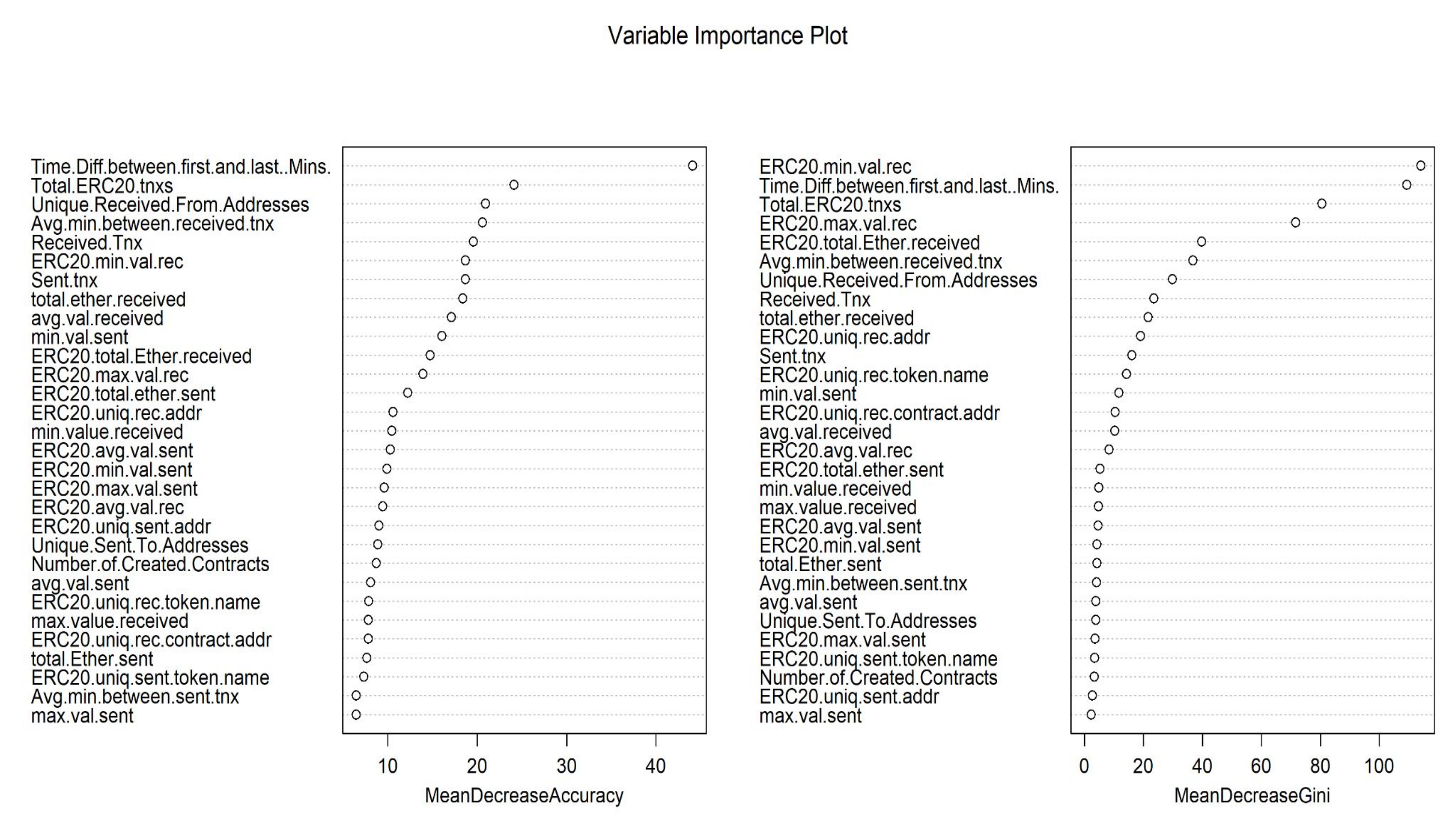




```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 607  30
         1  68 645

               Accuracy : 0.9274
                 95% CI : (0.9122, 0.9407)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8548

 Mcnemar's Test P-Value : 0.0001858

            Sensitivity : 0.8993
            Specificity : 0.9556
         Pos Pred Value : 0.9529
         Neg Pred Value : 0.9046
             Prevalence : 0.5000
         Detection Rate : 0.4496
   Detection Prevalence : 0.4719
      Balanced Accuracy : 0.9274

       'Positive' Class : 0
```

RANDOM FOREST

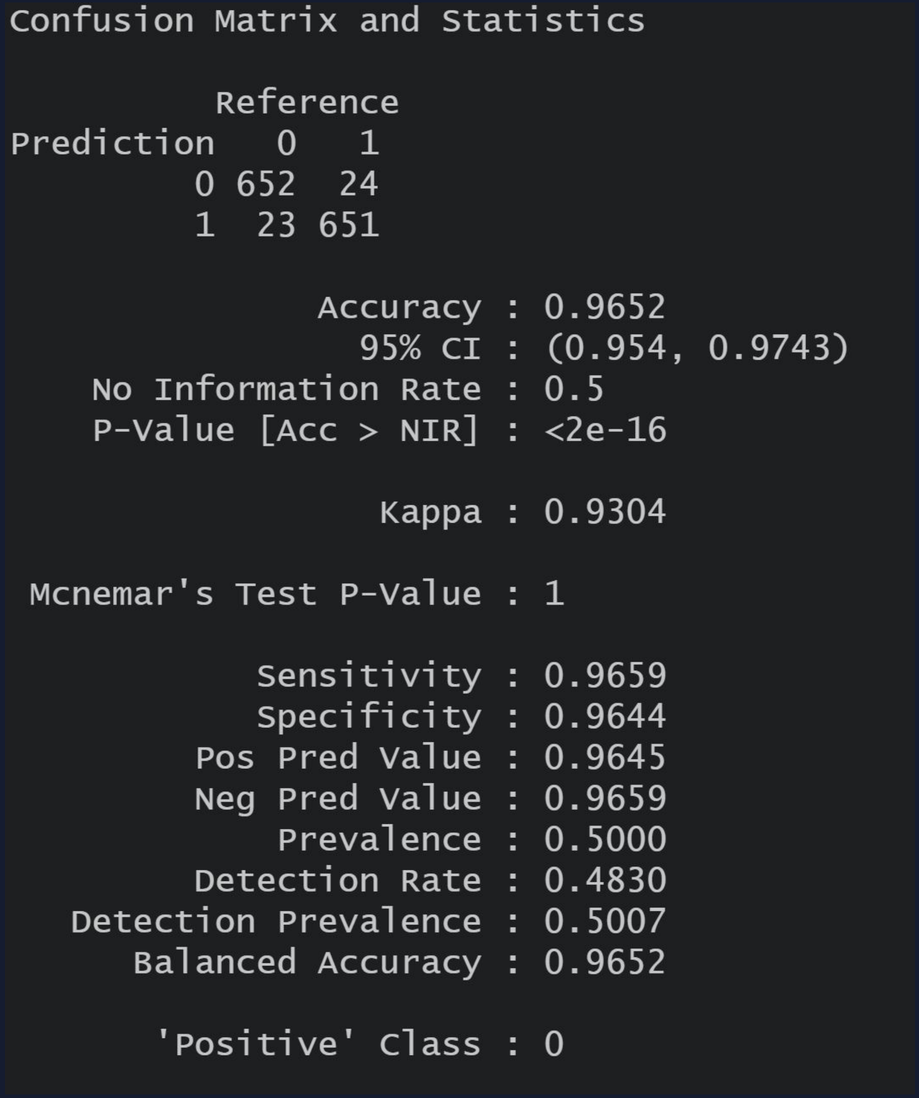




From the above graph, we can see that the optimal value for the hyperparameter ntree is 190 as it has the lowest error.

RANDOM FOREST RESULTS

- Accuracy = 96.52%
- Precision = 96.45%
- Recall = 96.44%



```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 652  24
         1  23 651
                                          
               Accuracy : 0.9652          
                 95% CI : (0.954, 0.9743)
    No Information Rate : 0.5             
    P-Value [Acc > NIR] : <2e-16          
                                          
                  Kappa : 0.9304          
                                          
 Mcnemar's Test P-Value : 1               
                                          
            Sensitivity : 0.9659          
            Specificity : 0.9644          
         Pos Pred Value : 0.9645          
         Neg Pred Value : 0.9659          
             Prevalence : 0.5000          
         Detection Rate : 0.4830          
   Detection Prevalence : 0.5007          
      Balanced Accuracy : 0.9652          
                                          
       'Positive' Class : 0               
```