Credit Card Fraud Detection - XGBOOST *

Javier Ng *EY*

The datasets contains transactions made by credit cards in September 2013 by european cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.

Keywords: xgboost, machine learning, R

Overview

Loading of Libraries

```
setwd("C:/Users/zm679xs/Desktop/R/Essential-R-Codes/EDA Projects/Credit Card Fraud Detection")
library(ggplot2) # Data visualization
library(readr) # CSV file I/O, e.g. the read_csv function
library(caret)
library(DMwR) # smote
library(xgboost)
library(Matrix)
library(reshape) #melt
library(pROC) # AUC

# Input data files are available in the "../input/" directory.
# For example, running this (by clicking run or pressing Shift+Enter) will list the files in the
load("ccdata.RData")
```

Split Data

Split data into train, test and cv using caret.

```
## split into train, test, cv
set.seed(1900)
inTrain <- createDataPartition(y = ccdata$Class, p = .6, list = F) #60% train
train <- ccdata[inTrain,]
testcv <- ccdata[-inTrain,]
inTest <- createDataPartition(y = testcv$Class, p = .5, list = F) #20% test
test <- testcv[inTest,]
cv <- testcv[-inTest,]
train$Class <- as.factor(train$Class)
rm(inTrain, inTest, testcv) #20% cv
```

*PDF is available on EY FDA shared drive. **Current version**: March 11, 2019; **Corresponding author**: javier.ng@sg.ey.com.

Imbalance Dataset

Using SMOTE to make the dataset more balanced.

```
#using SMOTE instead
i <- grep("Class", colnames(train)) # Get index Class column
train_smote <- SMOTE(Class ~ ., as.data.frame(train), perc.over = 20000, perc.under=100)
table(train_smote$Class)
```

```
##
##     0     1
## 60400 60702
```

```
#Identify the Predictors and the dependent variable, aka label.
predictors <- colnames(train_smote[-ncol(train_smote)]) #remove last column
#xgboost works only if the labels are numeric. Hence, convert the labels (Species) to numeric.
label <- as.numeric(train_smote[,ncol(train_smote)])

#Alas, xgboost works only if the numeric labels start from 0. Hence, subtract 1 from the label.
label <- as.numeric(train_smote[,ncol(train_smote)])-1
print(table(label))
```

```
## label
##     0     1
## 60400 60702
```

Setting Parameters for XGBoost

```
# set parameters:
parameters <- list(
  # General Parameters
  booster            = "gbtree",
  silent             = 0,
  # Booster Parameters
  eta                = 0.3,
  gamma              = 0,
  max_depth          = 6,
  min_child_weight   = 1,
  subsample          = 1,
  colsample_bytree   = 1,
  colsample_bylevel  = 1,
  lambda             = 1,
  alpha              = 0,
  # Task Parameters
  objective          = "binary:logistic",
  eval_metric        = "auc",
  seed               = 1900
)
```

Train Model

Train the model with the parameters set above and nrounds = 25 (increasing nrounds does not improve the model anymore). Plots show increasing train and cv AUC in the beginning and stagnating at later rounds as expected.

```
## Training of models
# Original
cv.nround = 200;  # Number of rounds. This can be set to a lower or higher value, if you wish, e
bst.cv <- xgb.cv(
  param=parameters,
  data = as.matrix(train_smote[,predictors]),
  label = label,
  nfold = 3,
  nrounds=cv.nround,
  prediction=T)
```

```
## [1]  train-auc:0.988126+0.000290 test-auc:0.987120+0.000203
## [2]  train-auc:0.994458+0.000626 test-auc:0.993682+0.000973
## [3]  train-auc:0.996438+0.000303 test-auc:0.995692+0.000592
## [4]  train-auc:0.997409+0.000210 test-auc:0.996878+0.000309
## [5]  train-auc:0.998177+0.000100 test-auc:0.997759+0.000171
## [6]  train-auc:0.998640+0.000145 test-auc:0.998255+0.000200
## [7]  train-auc:0.998912+0.000079 test-auc:0.998543+0.000171
## [8]  train-auc:0.999116+0.000082 test-auc:0.998800+0.000144
## [9]  train-auc:0.999324+0.000044 test-auc:0.999043+0.000009
## [10] train-auc:0.999437+0.000048 test-auc:0.999184+0.000020
## [11] train-auc:0.999559+0.000040 test-auc:0.999334+0.000016
## [12] train-auc:0.999609+0.000043 test-auc:0.999400+0.000026
## [13] train-auc:0.999674+0.000033 test-auc:0.999473+0.000032
## [14] train-auc:0.999721+0.000038 test-auc:0.999529+0.000021
## [15] train-auc:0.999774+0.000020 test-auc:0.999595+0.000016
## [16] train-auc:0.999816+0.000006 test-auc:0.999658+0.000022
## [17] train-auc:0.999855+0.000010 test-auc:0.999710+0.000013
## [18] train-auc:0.999868+0.000009 test-auc:0.999729+0.000014
## [19] train-auc:0.999893+0.000004 test-auc:0.999765+0.000009
## [20] train-auc:0.999911+0.000008 test-auc:0.999791+0.000008
## [21] train-auc:0.999932+0.000011 test-auc:0.999824+0.000011
## [22] train-auc:0.999945+0.000008 test-auc:0.999845+0.000012
## [23] train-auc:0.999952+0.000010 test-auc:0.999855+0.000020
## [24] train-auc:0.999959+0.000010 test-auc:0.999868+0.000027
## [25] train-auc:0.999964+0.000008 test-auc:0.999874+0.000027
## [26] train-auc:0.999969+0.000007 test-auc:0.999884+0.000025
## [27] train-auc:0.999976+0.000005 test-auc:0.999892+0.000026
## [28] train-auc:0.999979+0.000005 test-auc:0.999909+0.000024
## [29] train-auc:0.999982+0.000004 test-auc:0.999918+0.000021
## [30] train-auc:0.999983+0.000004 test-auc:0.999922+0.000019
## [31] train-auc:0.999985+0.000004 test-auc:0.999926+0.000021
## [32] train-auc:0.999987+0.000003 test-auc:0.999931+0.000020
## [33] train-auc:0.999988+0.000001 test-auc:0.999940+0.000017
## [34] train-auc:0.999990+0.000002 test-auc:0.999942+0.000018
## [35] train-auc:0.999990+0.000002 test-auc:0.999946+0.000021
## [36] train-auc:0.999991+0.000003 test-auc:0.999949+0.000021
## [37] train-auc:0.999993+0.000002 test-auc:0.999953+0.000022
## [38] train-auc:0.999993+0.000002 test-auc:0.999955+0.000021
## [39] train-auc:0.999994+0.000002 test-auc:0.999959+0.000020
## [40] train-auc:0.999996+0.000002 test-auc:0.999962+0.000019
## [41] train-auc:0.999997+0.000002 test-auc:0.999964+0.000019
## [42] train-auc:0.999997+0.000002 test-auc:0.999966+0.000018
## [43] train-auc:0.999998+0.000002 test-auc:0.999967+0.000017
## [44] train-auc:0.999998+0.000001 test-auc:0.999970+0.000017
## [45] train-auc:0.999998+0.000001 test-auc:0.999971+0.000018
## [46] train-auc:0.999999+0.000001 test-auc:0.999972+0.000017
## [47] train-auc:1.000000+0.000000 test-auc:0.999973+0.000018
## [48] train-auc:1.000000+0.000000 test-auc:0.999974+0.000017
## [49] train-auc:1.000000+0.000000 test-auc:0.999975+0.000016
## [50] train-auc:1.000000+0.000000 test-auc:0.999977+0.000014
## [51] train-auc:1.000000+0.000000 test-auc:0.999978+0.000014
## [52] train-auc:1.000000+0.000000 test-auc:0.999979+0.000013
## [53] train-auc:1.000000+0.000000 test-auc:0.999979+0.000014
## [54] train-auc:1.000000+0.000000 test-auc:0.999979+0.000015
## [55] train-auc:1.000000+0.000000 test-auc:0.999981+0.000013
## [56] train-auc:1.000000+0.000000 test-auc:0.999981+0.000013
## [57] train-auc:1.000000+0.000000 test-auc:0.999981+0.000013
## [58] train-auc:1.000000+0.000000 test-auc:0.999983+0.000011
## [59] train-auc:1.000000+0.000000 test-auc:0.999982+0.000013
## [60] train-auc:1.000000+0.000000 test-auc:0.999983+0.000012
## [61] train-auc:1.000000+0.000000 test-auc:0.999984+0.000011
## [62] train-auc:1.000000+0.000000 test-auc:0.999985+0.000010
## [63] train-auc:1.000000+0.000000 test-auc:0.999985+0.000010
## [64] train-auc:1.000000+0.000000 test-auc:0.999984+0.000011
## [65] train-auc:1.000000+0.000000 test-auc:0.999985+0.000010
## [66] train-auc:1.000000+0.000000 test-auc:0.999985+0.000010
## [67] train-auc:1.000000+0.000000 test-auc:0.999985+0.000011
## [68] train-auc:1.000000+0.000000 test-auc:0.999986+0.000010
## [69] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [70] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [71] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [72] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [73] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [74] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [75] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [76] train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [77] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [78] train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [79] train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [80] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [81] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [82] train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [83] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [84] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [85] train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [86] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [87] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [88] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [89] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [90] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [91] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [92] train-auc:1.000000+0.000000 test-auc:0.999987+0.000009
## [93] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [94] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [95] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [96] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [97] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [98] train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [99] train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [100]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [101]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [102]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [103]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [104]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [105]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [106]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [107]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [108]    train-auc:1.000000+0.000000 test-auc:0.999986+0.000011
## [109]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [110]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [111]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [112]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [113]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [114]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [115]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [116]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [117]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [118]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [119]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [120]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [121]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [122]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [123]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [124]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [125]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [126]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [127]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [128]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [129]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000010
## [130]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [131]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [132]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [133]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [134]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [135]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [136]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [137]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [138]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [139]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [140]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [141]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000009
## [142]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [143]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [144]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [145]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [146]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [147]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [148]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [149]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [150]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [151]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [152]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [153]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [154]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [155]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [156]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [157]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [158]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [159]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [160]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [161]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [162]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [163]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [164]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [165]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [166]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000010
## [167]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [168]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [169]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [170]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [171]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [172]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [173]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [174]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [175]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [176]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [177]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [178]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [179]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [180]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [181]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [182]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [183]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [184]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [185]    train-auc:1.000000+0.000000 test-auc:0.999988+0.000011
## [186]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [187]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [188]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [189]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [190]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [191]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [192]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [193]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [194]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [195]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [196]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [197]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [198]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [199]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
## [200]    train-auc:1.000000+0.000000 test-auc:0.999987+0.000011
```

```r
#Find where the minimum logloss occurred
min.loss.idx = which.max(bst.cv$evaluation_log[, test_auc_mean])
cat("Minimum logloss occurred in round : ", min.loss.idx, "\n")
```

```
## Minimum logloss occurred in round :  139
```

```r
# Minimum logloss
print(bst.cv$evaluation_log[min.loss.idx,])
```

```
##    iter train_auc_mean train_auc_std test_auc_mean test_auc_std
## 1:  139              1             0     0.9999883 9.392662e-06
```

Predict

Predict with dataset and set threshold of 0.5

```r
## Make predictions
bst <- xgboost(
  param=parameters,
  data =as.matrix(train_smote[,predictors]), #training it without the output variable!
  label = label,
  nrounds=min.loss.idx)
```

```
## [1]  train-auc:0.988286
## [2]  train-auc:0.995025
## [3]  train-auc:0.996358
## [4]  train-auc:0.997692
## [5]  train-auc:0.998161
## [6]  train-auc:0.998605
## [7]  train-auc:0.999029
## [8]  train-auc:0.999174
## [9]  train-auc:0.999332
## [10] train-auc:0.999449
## [11] train-auc:0.999527
## [12] train-auc:0.999583
## [13] train-auc:0.999652
## [14] train-auc:0.999680
## [15] train-auc:0.999759
## [16] train-auc:0.999798
## [17] train-auc:0.999830
## [18] train-auc:0.999855
## [19] train-auc:0.999878
## [20] train-auc:0.999900
## [21] train-auc:0.999931
## [22] train-auc:0.999945
## [23] train-auc:0.999957
## [24] train-auc:0.999963
## [25] train-auc:0.999971
## [26] train-auc:0.999975
## [27] train-auc:0.999977
## [28] train-auc:0.999980
## [29] train-auc:0.999982
## [30] train-auc:0.999985
## [31] train-auc:0.999987
## [32] train-auc:0.999989
## [33] train-auc:0.999989
## [34] train-auc:0.999990
## [35] train-auc:0.999991
## [36] train-auc:0.999993
## [37] train-auc:0.999994
## [38] train-auc:0.999995
## [39] train-auc:0.999995
## [40] train-auc:0.999996
## [41] train-auc:0.999996
## [42] train-auc:0.999997
## [43] train-auc:0.999998
## [44] train-auc:0.999998
## [45] train-auc:0.999999
## [46] train-auc:0.999999
## [47] train-auc:0.999999
## [48] train-auc:0.999999
## [49] train-auc:1.000000
## [50] train-auc:1.000000
## [51] train-auc:1.000000
## [52] train-auc:1.000000
## [53] train-auc:1.000000
## [54] train-auc:1.000000
## [55] train-auc:1.000000
## [56] train-auc:1.000000
## [57] train-auc:1.000000
## [58] train-auc:1.000000
## [59] train-auc:1.000000
## [60] train-auc:1.000000
## [61] train-auc:1.000000
## [62] train-auc:1.000000
## [63] train-auc:1.000000
## [64] train-auc:1.000000
## [65] train-auc:1.000000
## [66] train-auc:1.000000
## [67] train-auc:1.000000
## [68] train-auc:1.000000
## [69] train-auc:1.000000
## [70] train-auc:1.000000
## [71] train-auc:1.000000
## [72] train-auc:1.000000
## [73] train-auc:1.000000
## [74] train-auc:1.000000
## [75] train-auc:1.000000
## [76] train-auc:1.000000
## [77] train-auc:1.000000
## [78] train-auc:1.000000
## [79] train-auc:1.000000
## [80] train-auc:1.000000
## [81] train-auc:1.000000
## [82] train-auc:1.000000
## [83] train-auc:1.000000
## [84] train-auc:1.000000
## [85] train-auc:1.000000
## [86] train-auc:1.000000
## [87] train-auc:1.000000
## [88] train-auc:1.000000
## [89] train-auc:1.000000
## [90] train-auc:1.000000
## [91] train-auc:1.000000
## [92] train-auc:1.000000
## [93] train-auc:1.000000
## [94] train-auc:1.000000
## [95] train-auc:1.000000
## [96] train-auc:1.000000
## [97] train-auc:1.000000
## [98] train-auc:1.000000
## [99] train-auc:1.000000
## [100]    train-auc:1.000000
## [101]    train-auc:1.000000
## [102]    train-auc:1.000000
## [103]    train-auc:1.000000
## [104]    train-auc:1.000000
## [105]    train-auc:1.000000
## [106]    train-auc:1.000000
## [107]    train-auc:1.000000
## [108]    train-auc:1.000000
## [109]    train-auc:1.000000
## [110]    train-auc:1.000000
## [111]    train-auc:1.000000
## [112]    train-auc:1.000000
## [113]    train-auc:1.000000
## [114]    train-auc:1.000000
## [115]    train-auc:1.000000
## [116]    train-auc:1.000000
## [117]    train-auc:1.000000
## [118]    train-auc:1.000000
## [119]    train-auc:1.000000
## [120]    train-auc:1.000000
## [121]    train-auc:1.000000
## [122]    train-auc:1.000000
## [123]    train-auc:1.000000
## [124]    train-auc:1.000000
## [125]    train-auc:1.000000
## [126]    train-auc:1.000000
## [127]    train-auc:1.000000
## [128]    train-auc:1.000000
## [129]    train-auc:1.000000
## [130]    train-auc:1.000000
## [131]    train-auc:1.000000
## [132]    train-auc:1.000000
## [133]    train-auc:1.000000
## [134]    train-auc:1.000000
## [135]    train-auc:1.000000
## [136]    train-auc:1.000000
## [137]    train-auc:1.000000
## [138]    train-auc:1.000000
## [139]    train-auc:1.000000
```

```r
# Make prediction on the testing data.
test$prediction <- predict(bst, as.matrix(test[,predictors])) #here, I've removed the output var
```

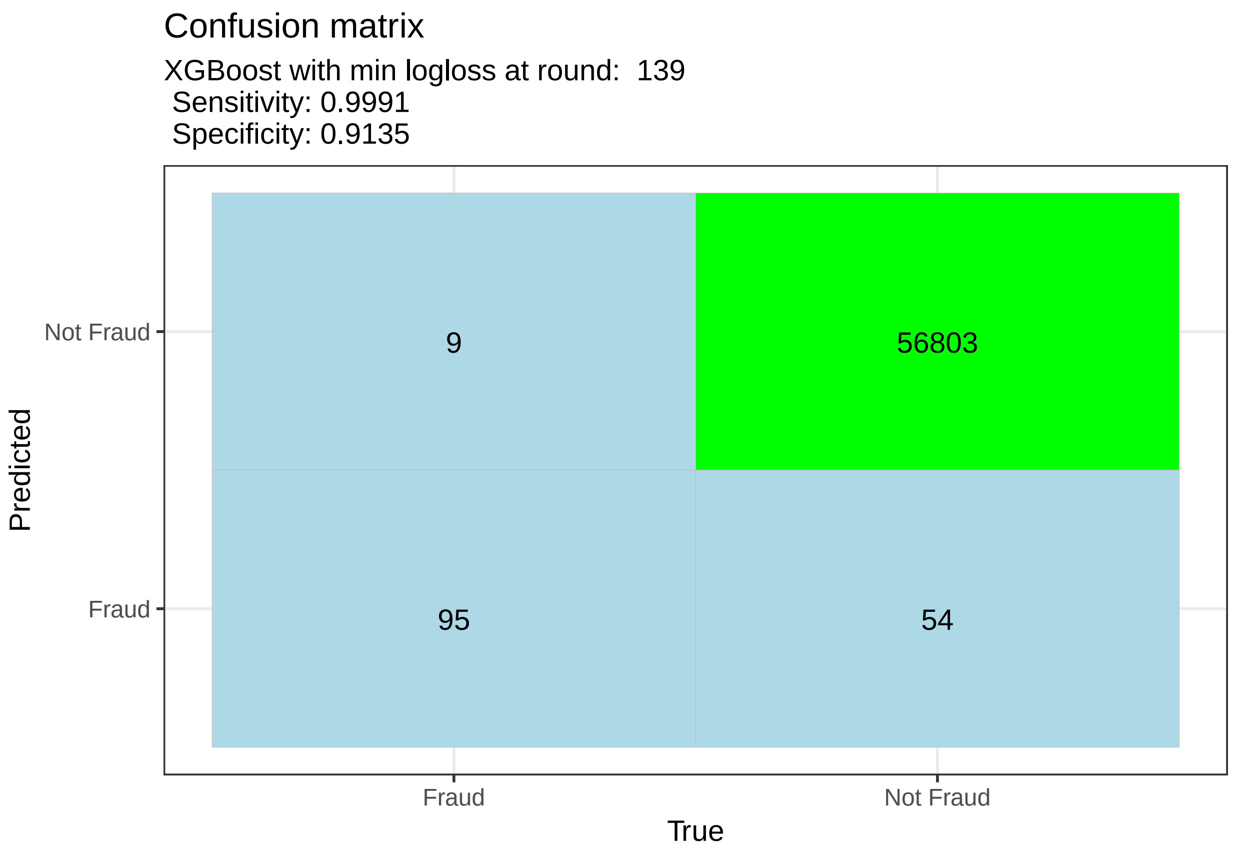
```
test$prediction <- ifelse(test$prediction >= 0.5, 1 , 0)
```

Confusion Matrix

Sensitivity (TPR) = 0.9991 Specificity (TNR) = 0.9135 Accuracy = 0.9989

```
#Compute the accuracy of predictions.
confmatrix_table <- confusionMatrix(as.factor(test$prediction), as.factor(test$Class)) #sensitiv
#accuracy = 0.9989
```

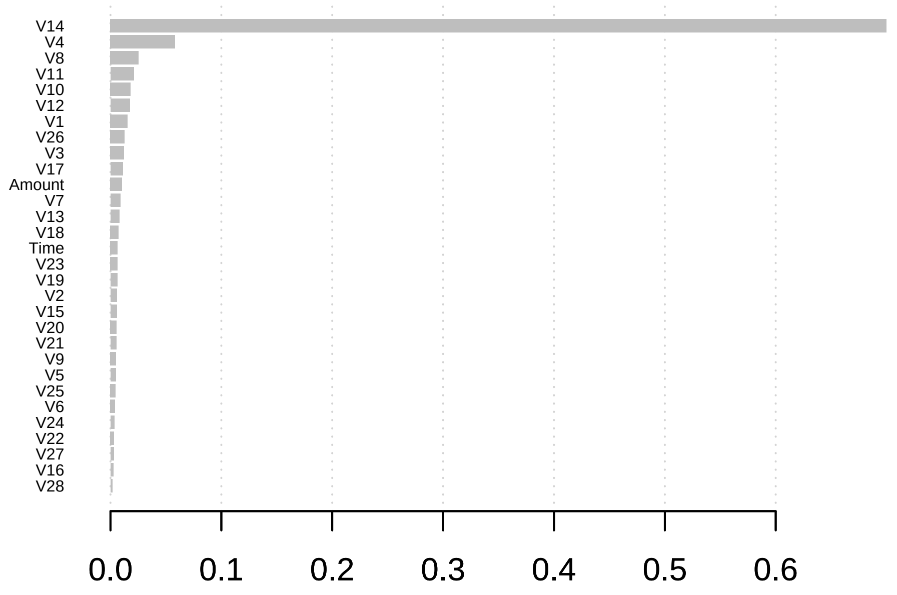
```
plot_confusion_matrix <- function(test_df, sSubtitle) {
  tst <- data.frame(round(test_df$prediction,0), test_df$Class)
  opts <-  c("Predicted", "True")
  names(tst) <- opts
  cf <- plyr::count(tst)
  cf[opts][cf[opts]==0] <- "Not Fraud"
  cf[opts][cf[opts]==1] <- "Fraud"

  ggplot(data =  cf, mapping = aes(x = True, y = Predicted)) +
    labs(title = "Confusion matrix", subtitle = sSubtitle) +
    geom_tile(aes(fill = freq), colour = "grey") +
    geom_text(aes(label = sprintf("%1.0f", freq)), vjust = 1) +
    scale_fill_gradient(low = "lightblue", high = "Green") +
    theme_bw() + theme(legend.position = "none")

}

plot_confusion_matrix(test, paste("XGBoost with", paste("min logloss at round: ", min.loss.idx,
                                  "Sensitivity:", round(confmatrix_table[[4]][1], 4), "\n",
                                  "Specificity:", round(confmatrix_table[[4]][2], 4)))
```



Importance Matrix

Identify features that are most important

```
##plot feature importance
importance_matrix <- xgb.importance(model = bst)
xgb.plot.importance(importance_matrix)
```



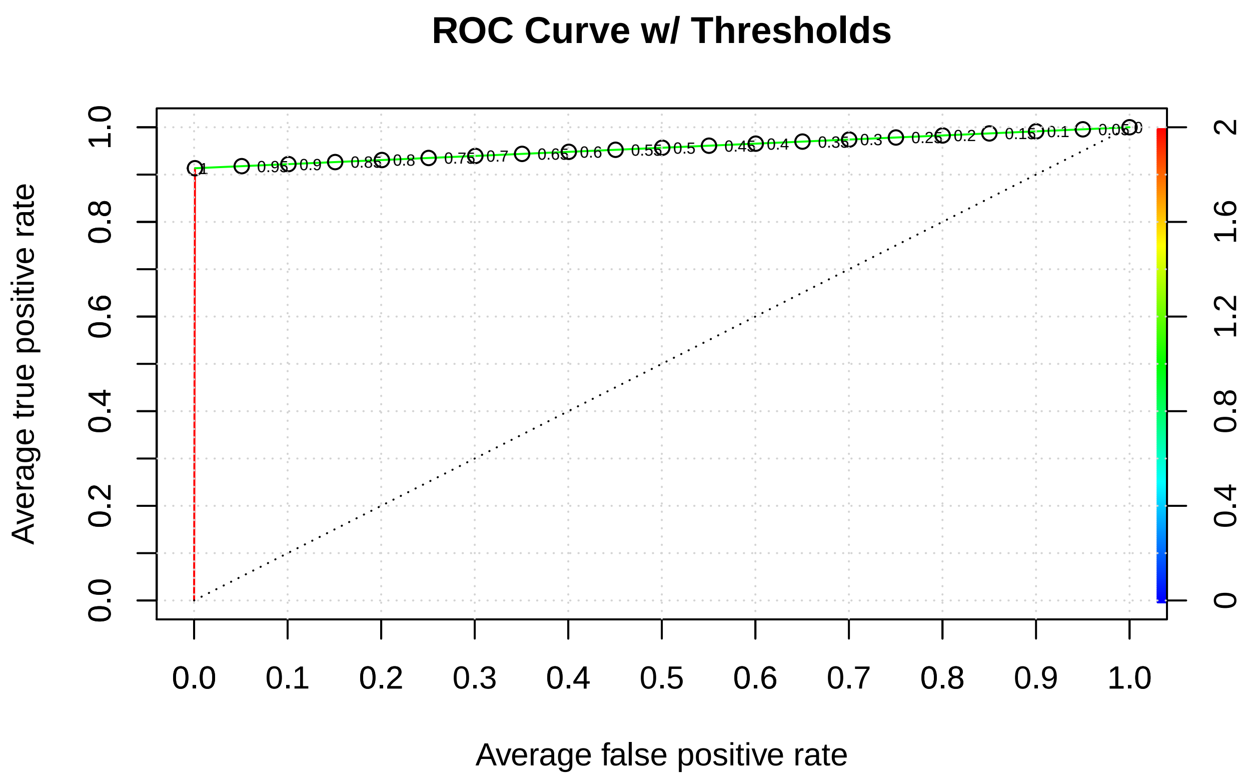
ROC Curve

AUC = 0.956

```
##################################################################
library(ROCR)
```

```
## Loading required package: gplots

##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##     lowess
```

```
# Use ROCR package to plot ROC Curve
xgb.pred <- prediction(test$prediction, test$Class)
xgb.perf <- performance(xgb.pred, "tpr", "fpr")

plot(xgb.perf,
     avg="threshold",
     colorize=TRUE,
     lwd=1,
     main="ROC Curve w/ Thresholds",
     print.cutoffs.at=seq(0, 1, by=0.05),
     text.adj=c(-0.5, 0.5),
     text.cex=0.5)
grid(col="lightgray")
axis(1, at=seq(0, 1, by=0.1))
axis(2, at=seq(0, 1, by=0.1))
abline(v=c(0.1, 0.3, 0.5, 0.7, 0.9), col="lightgray", lty="dotted")
abline(h=c(0.1, 0.3, 0.5, 0.7, 0.9), col="lightgray", lty="dotted")
lines(x=c(0, 1), y=c(0, 1), col="black", lty="dotted")
```



```
auc_ROCR <- performance(xgb.pred, measure = "auc") #gives the AUC rate
auc_ROCR <- auc_ROCR@y.values[[1]]
auc_ROCR
```

```
## [1] 0.9562559
```

```
#AUC = 0.956
```