predict.R

stell

Wed Feb 28 22:18:09 2018

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(data.table)

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

library(rpart)  
  
  
# read all csv files into one data.table  
read.files <- function(){  
 files <- list.files(pattern = '\\.csv')  
 tables <- lapply(files, read.csv, header = TRUE)  
 combined.df <- do.call(rbind , tables)  
 return(data.table(combined.df))  
}  
  
# prepare data table with only DEFICIT, PCT and OUTCOME statistics.  
data <- read.files()  
outcome=data$OUTCOME  
data = data[ ,(grepl('(?:DEFICIT|PCT|OUTCOME).\*(?<!M)$', names(data), perl = T)), with=F]  
data = data[complete.cases(data)]  
data[,DEFICIT\_REB:=NULL]  
  
data$OUTCOME = as.factor(data$OUTCOME)  
# generate data partition for training and testing  
inTrain = createDataPartition(data$OUTCOME, p=0.7, list = FALSE)  
training = data[inTrain,]  
testing = data[-inTrain,]  
  
# train the algoritham on genrated data.  
dtree = train(OUTCOME~., method="rpart", data=training, na.action = na.pass)  
print(dtree$ finalModel)

## n= 6554   
##   
## node), split, n, loss, yval, (yprob)  
## \* denotes terminal node  
##   
## 1) root 6554 3277 0 (0.5000000 0.5000000)   
## 2) DEFICIT\_DREB< 0.4939015 3074 781 0 (0.7459336 0.2540664) \*  
## 3) DEFICIT\_DREB>=0.4939015 3480 984 1 (0.2827586 0.7172414)   
## 6) DEFICIT\_TO>=0.5303309 1701 754 1 (0.4432687 0.5567313)   
## 12) DEFICIT\_DREB< 0.5477821 908 305 0 (0.6640969 0.3359031) \*  
## 13) DEFICIT\_DREB>=0.5477821 793 151 1 (0.1904161 0.8095839) \*  
## 7) DEFICIT\_TO< 0.5303309 1779 230 1 (0.1292861 0.8707139) \*

# predict using decision tree.  
prediction = predict(dtree, newdata = testing)  
print(prediction)

## [1] 0 0 1 1 0 1 0 0 0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1 1  
## [35] 1 1 0 0 0 1 1 0 0 1 0 1 1 0 0 1 0 1 0 0 1 0 0 0 0 0 1 0 0 0 1 1 1 0  
## [69] 1 0 1 0 1 1 1 0 0 1 0 0 0 0 0 0 1 0 1 1 1 0 0 0 1 1 0 0 1 1 0 0 0 0  
## [103] 1 0 1 0 1 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 0 0 1 0 1 1 0 0 1 0  
## [137] 0 1 1 1 0 1 0 0 0 0 0 0 1 0 1 1 0 1 0 1 0 1 0 0 0 0 0 0 0 1 0 0 1 0  
## [171] 1 0 1 0 1 1 0 1 0 0 0 0 1 1 0 1 0 1 0 1 1 1 0 0 0 0 0 0 0 1 1 0 0 1  
## [205] 0 0 0 1 0 0 1 0 0 1 0 1 1 0 1 0 0 0 0 1 1 0 0 0 1 0 0 1 0 0 0 0 1 0  
## [239] 0 1 0 0 1 0 0 1 1 0 0 0 1 0 0 1 1 0 0 1 0 0 0 0 1 0 0 1 1 0 1 0 1 1  
## [273] 0 1 1 0 0 0 1 0 1 0 1 0 0 0 1 0 0 0 1 0 1 0 0 0 1 0 0 1 0 0 0 1 0 0  
## [307] 0 0 0 0 0 0 0 1 1 0 0 1 1 0 1 1 1 1 0 0 0 1 0 1 1 1 1 0 0 0 1 0 0 1  
## [341] 0 0 0 0 0 1 1 1 0 1 1 0 0 1 0 1 1 0 0 0 0 1 1 0 1 0 1 0 0 0 1 0 0 0  
## [375] 0 0 0 1 0 0 0 0 0 0 1 0 1 0 1 1 1 0 0 0 0 1 0 0 0 0 1 0 0 1 1 1 1 0  
## [409] 1 0 1 0 0 1 0 0 1 0 0 1 0 0 0 0 0 1 0 0 0 0 1 0 0 1 1 1 1 0 0 1 1 0  
## [443] 0 0 1 0 0 1 1 1 0 0 0 0 0 1 1 1 0 1 1 0 1 1 0 1 0 0 0 1 1 0 1 1 0 0  
## [477] 0 0 0 1 0 0 0 1 1 0 1 1 0 1 1 1 0 0 0 0 0 1 0 1 1 0 0 0 0 1 0 1 0 1  
## [511] 0 1 0 0 0 0 0 1 0 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1 0 0 0 0 1 0 0 0 1  
## [545] 0 0 0 1 1 0 1 0 0 0 1 0 1 0 1 0 1 0 0 1 0 1 0 0 1 0 0 0 1 1 0 0 0 0  
## [579] 1 1 0 0 1 0 0 0 0 1 1 1 0 1 0 0 0 0 1 0 0 0 0 0 1 0 1 0 0 0 0 1 0 0  
## [613] 1 0 0 1 1 0 1 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0  
## [647] 0 0 0 1 0 0 0 1 0 1 1 1 0 0 1 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0  
## [681] 0 0 0 0 1 0 1 0 1 0 1 0 0 0 0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0 1 1 1 0  
## [715] 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 1 0 1 0 0 0 0 1 1 1 0 0 1 0 0 0 0 1  
## [749] 0 1 0 1 0 0 0 1 0 1 0 0 1 0 1 1 0 1 0 0 0 0 0 0 0 1 0 1 0 1 1 1 1 1  
## [783] 0 1 0 0 0 0 1 0 1 0 1 0 1 0 1 0 0 0 1 1 0 0 1 0 0 1 0 0 0 1 0 0 0 0  
## [817] 1 1 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0 0 1 1 0 0 1 0 0 1 0 0 0 0 0 1 0 0  
## [851] 0 0 0 0 0 1 0 0 1 1 1 0 0 1 0 1 1 0 0 1 0 1 0 1 0 1 0 0 1 0 1 0 0 1  
## [885] 0 0 0 1 1 1 1 0 0 0 1 0 0 1 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 1 0 0  
## [919] 1 0 0 1 0 1 0 0 0 0 0 0 0 1 0 1 1 0 1 1 0 0 0 1 1 0 1 1 0 1 0 0 1 0  
## [953] 1 1 1 0 0 0 0 1 0 1 1 1 0 0 0 1 0 0 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0  
## [987] 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 1 1 0 1 0 1 0 0 1 0  
## [1021] 1 1 1 0 0 0 0 0 0 0 0 1 0 1 0 1 1 1 0 1 0 1 1 0 0 0 1 0 1 1 0 0 1 0  
## [1055] 0 1 1 1 0 0 0 0 1 0 0 1 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 0 0 1 0  
## [1089] 0 0 0 1 1 1 0 1 0 1 1 0 1 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 0 0 0 0 1 0  
## [1123] 0 1 0 0 1 0 1 1 1 0 0 0 0 1 0 1 1 0 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 1  
## [1157] 0 0 0 1 0 0 0 1 1 0 1 0 1 1 1 0 0 0 1 1 0 0 0 1 1 1 0 0 0 0 0 0 1 1  
## [1191] 0 1 0 0 0 0 0 0 0 1 0 0 1 1 0 1 1 0 1 1 1 1 1 0 0 1 0 0 1 0 0 0 1 0  
## [1225] 1 0 1 0 1 1 0 0 0 0 1 0 1 0 0 0 1 0 0 1 1 1 0 1 1 0 0 1 0 1 0 0 1 0  
## [1259] 0 0 1 0 0 1 0 0 1 1 0 0 1 1 0 0 1 0 0 1 0 0 1 0 1 0 0 0 1 0 0 1 1 0  
## [1293] 0 1 1 0 0 0 1 0 1 0 1 0 0 0 1 0 0 0 1 0 1 0 1 1 0 0 0 1 1 0 0 1 0 0  
## [1327] 1 0 0 1 1 0 1 0 0 0 1 1 0 1 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 1 0 1 0 0  
## [1361] 1 0 0 1 0 1 0 1 0 1 0 1 1 0 1 0 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0 1 1 0  
## [1395] 0 0 1 1 1 0 0 0 1 0 1 0 0 0 0 0 0 1 0 1 1 0 1 0 0 1 0 1 1 0 1 0 0 1  
## [1429] 1 1 0 0 0 1 1 0 0 1 1 0 0 1 1 1 0 0 1 0 1 1 0 0 1 0 0 0 1 1 0 0 1 1  
## [1463] 1 0 0 1 0 0 1 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 0 0 1 0 1 0 0 0 0 1  
## [1497] 0 0 0 1 0 0 1 0 0 0 0 1 1 1 0 0 0 0 1 0 0 0 0 0 0 0 1 1 0 0 0 0 1 0  
## [1531] 1 0 0 1 0 0 1 1 0 0 1 1 1 0 1 0 1 0 0 0 0 0 1 1 0 0 1 0 0 1 1 0 0 1  
## [1565] 1 1 1 0 0 1 1 1 1 0 0 0 1 0 0 0 1 0 0 1 1 1 1 0 1 1 0 0 0 0 1 0 1 0  
## [1599] 0 0 0 1 1 1 1 0 1 0 1 1 0 1 0 1 0 0 1 0 1 1 0 1 1 0 0 0 0 0 0 1 0 0  
## [1633] 0 1 0 0 1 1 0 1 1 1 0 1 0 0 1 0 1 0 1 1 0 1 1 0 0 0 0 1 1 0 0 1 0 1  
## [1667] 1 0 1 0 1 0 0 1 1 0 1 0 0 0 1 0 0 1 0 0 0 0 0 1 1 1 1 0 1 0 1 1 0 1  
## [1701] 1 0 0 0 1 1 0 1 1 0 1 0 0 0 0 1 1 0 1 1 0 0 0 1 1 0 1 0 1 1 0 1 1 0  
## [1735] 0 1 0 1 0 1 1 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0 1 0  
## [1769] 0 1 1 0 1 0 1 1 0 0 1 0 0 0 1 1 0 0 1 0 1 0 0 1 1 1 1 0 0 0 0 0 0 0  
## [1803] 0 0 1 1 1 0 0 0 0 1 0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0  
## [1837] 1 0 0 0 1 1 0 0 0 1 0 1 1 0 1 0 1 1 0 0 1 1 0 0 0 0 1 0 0 0 0 1 1 0  
## [1871] 0 0 0 0 1 0 0 1 1 1 0 1 0 1 0 0 0 0 1 0 0 1 0 1 1 0 0 1 0 1 1 0 1 0  
## [1905] 1 0 1 1 0 0 0 1 0 1 0 0 0 0 0 1 1 1 0 0 1 1 0 0 0 0 0 0 1 0 0 1 1 0  
## [1939] 0 0 0 0 1 0 0 1 0 1 0 0 0 1 0 0 1 0 1 0 1 1 0 0 1 1 0 1 1 1 0 1 0 0  
## [1973] 0 0 1 1 0 0 1 0 1 0 1 0 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 1 0 0 1 0  
## [2007] 0 0 0 1 1 0 0 0 0 0 0 1 1 0 1 0 1 0 0 0 0 0 0 0 1 0 1 0 1 1 0 0 0 0  
## [2041] 1 0 1 0 0 0 1 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0 1 0 0 1 1 1 0 1 0 0 0 0  
## [2075] 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 1 0 1 0 0 0 0 0 0 1 0 1 0 1 0 0 0 1  
## [2109] 1 0 0 1 0 0 1 0 0 1 0 1 0 0 0 1 0 1 0 1 0 1 0 0 1 1 0 1 1 0 0 1 0 0  
## [2143] 1 1 1 1 0 0 0 0 1 1 0 0 1 0 0 1 0 1 1 1 1 1 0 0 0 0 0 1 0 1 1 1 0 1  
## [2177] 0 1 0 0 0 1 0 0 0 1 0 0 1 0 1 1 1 0 0 1 1 0 0 0 1 1 0 0 1 0 1 0 1 0  
## [2211] 0 0 1 0 0 1 0 1 0 0 0 1 0 0 0 0 1 0 1 1 0 1 0 0 0 0 0 0 1 0 1 1 0 1  
## [2245] 0 1 0 1 0 1 0 1 0 0 0 0 0 1 1 1 0 0 0 0 1 0 1 0 0 0 0 0 1 0 1 0 0 0  
## [2279] 1 1 0 0 0 0 1 0 1 1 0 1 1 0 0 0 1 1 0 0 0 0 1 0 0 0 1 0 1 0 0 1 0 0  
## [2313] 0 1 1 0 1 1 1 1 1 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 1 0 1 1 0  
## [2347] 0 1 0 1 0 0 1 0 0 0 1 0 0 0 1 1 0 1 0 0 1 0 1 0 1 1 0 0 1 0 1 1 1 0  
## [2381] 0 0 1 1 0 1 0 0 1 0 1 0 0 0 1 1 0 0 0 1 1 0 0 0 0 1 0 0 0 0 1 0 0 0  
## [2415] 1 0 0 1 1 0 1 0 1 0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 0 1 0 0 1 0 0 0 0 0  
## [2449] 1 0 0 0 1 0 0 1 0 1 0 0 0 1 0 0 0 0 1 0 1 0 1 0 0 0 0 1 0 1 0 1 0 0  
## [2483] 0 0 1 1 0 1 1 0 0 0 0 1 0 0 1 0 1 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 1  
## [2517] 1 0 1 1 1 0 0 0 1 0 1 0 0 1 1 1 0 0 1 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0  
## [2551] 0 0 1 1 0 0 1 0 0 1 0 0 0 1 0 0 0 1 1 0 0 0 1 0 1 0 0 0 0 0 0 1 0 1  
## [2585] 1 0 0 1 1 0 0 1 1 0 0 1 0 0 1 0 1 0 0 1 1 0 0 0 1 0 1 1 1 1 1 0 1 0  
## [2619] 0 0 0 0 1 0 0 1 1 1 0 0 1 1 1 0 0 0 1 0 0 0 1 0 1 1 0 1 1 0 0 0 0 1  
## [2653] 1 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 1 1 0 0 0 1 0 1 1 1 0  
## [2687] 0 0 0 1 0 1 1 0 1 0 1 0 1 0 1 0 0 0 0 1 0 1 0 0 1 1 1 1 0 1 1 0 1 0  
## [2721] 0 0 0 0 1 1 0 0 1 0 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 0 1 1  
## [2755] 0 0 1 1 0 0 0 0 1 1 1 0 1 1 0 0 0 1 1 0 0 1 0 0 0 0 1 1 1 1 0 0 1 0  
## [2789] 1 0 0 1 0 1 0 0 1 0 0 1 1 1 0 0 1 0 1 1  
## Levels: 0 1

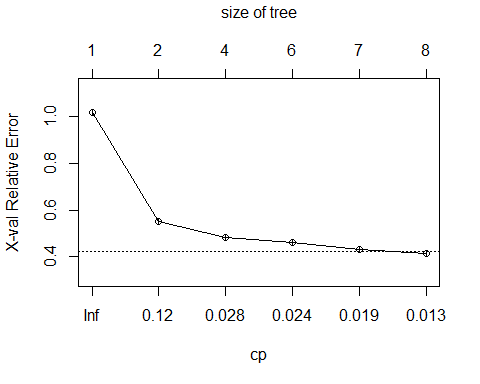
# check using confusion matrix how good is the prediction.  
cm = confusionMatrix(prediction, testing$OUTCOME)  
print(cm)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 1233 501  
## 1 171 903  
##   
## Accuracy : 0.7607   
## 95% CI : (0.7445, 0.7764)  
## No Information Rate : 0.5   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.5214   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.8782   
## Specificity : 0.6432   
## Pos Pred Value : 0.7111   
## Neg Pred Value : 0.8408   
## Prevalence : 0.5000   
## Detection Rate : 0.4391   
## Detection Prevalence : 0.6175   
## Balanced Accuracy : 0.7607   
##   
## 'Positive' Class : 0   
##

# grow a decision tree for visualization.  
fit <- rpart(OUTCOME~., method = "class", data=data)  
  
printcp(fit)

##   
## Classification tree:  
## rpart(formula = OUTCOME ~ ., data = data, method = "class")  
##   
## Variables actually used in tree construction:  
## [1] DEFICIT\_AST DEFICIT\_DREB DEFICIT\_TO   
##   
## Root node error: 4681/9362 = 0.5  
##   
## n= 9362   
##   
## CP nsplit rel error xerror xstd  
## 1 0.458876 0 1.00000 1.01987 0.0103331  
## 2 0.033540 1 0.54112 0.55138 0.0092367  
## 3 0.023820 3 0.47404 0.48280 0.0088455  
## 4 0.023286 5 0.42640 0.45930 0.0086941  
## 5 0.016022 6 0.40312 0.42940 0.0084875  
## 6 0.010000 7 0.38710 0.41508 0.0083827

plotcp(fit)



summary(fit)

## Call:  
## rpart(formula = OUTCOME ~ ., data = data, method = "class")  
## n= 9362   
##   
## CP nsplit rel error xerror xstd  
## 1 0.45887631 0 1.0000000 1.0198675 0.010333084  
## 2 0.03353984 1 0.5411237 0.5513779 0.009236723  
## 3 0.02381970 3 0.4740440 0.4828028 0.008845484  
## 4 0.02328562 5 0.4264046 0.4593036 0.008694090  
## 5 0.01602222 6 0.4031190 0.4293954 0.008487462  
## 6 0.01000000 7 0.3870968 0.4150822 0.008382743  
##   
## Variable importance  
## DEFICIT\_DREB DEFICIT\_AST DEFICIT\_TO FG2\_PCT DEFICIT\_FGA   
## 37 23 16 8 7   
## DEFICIT\_STL DEFICIT\_FG2A DEFICIT\_FTA FG3\_PCT   
## 7 1 1 1   
##   
## Node number 1: 9362 observations, complexity param=0.4588763  
## predicted class=0 expected loss=0.5 P(node) =1  
## class counts: 4681 4681  
## probabilities: 0.500 0.500   
## left son=2 (4408 obs) right son=3 (4954 obs)  
## Primary splits:  
## DEFICIT\_DREB < 0.4939015 to the left, improve=989.0303, (0 missing)  
## DEFICIT\_AST < 0.4961538 to the left, improve=812.2671, (0 missing)  
## FG2\_PCT < 0.4939015 to the left, improve=366.9091, (0 missing)  
## FG3\_PCT < 0.3785 to the left, improve=364.8455, (0 missing)  
## DEFICIT\_BLK < 0.4564394 to the left, improve=185.6757, (0 missing)  
## Surrogate splits:  
## DEFICIT\_TO < 0.4743421 to the left, agree=0.645, adj=0.246, (0 split)  
## FG2\_PCT < 0.4846117 to the left, agree=0.640, adj=0.236, (0 split)  
## DEFICIT\_FGA < 0.5064519 to the right, agree=0.622, adj=0.198, (0 split)  
## DEFICIT\_AST < 0.4836022 to the left, agree=0.621, adj=0.195, (0 split)  
## DEFICIT\_STL < 0.5400641 to the right, agree=0.589, adj=0.128, (0 split)  
##   
## Node number 2: 4408 observations, complexity param=0.0238197  
## predicted class=0 expected loss=0.2563521 P(node) =0.4708396  
## class counts: 3278 1130  
## probabilities: 0.744 0.256   
## left son=4 (2893 obs) right son=5 (1515 obs)  
## Primary splits:  
## DEFICIT\_AST < 0.5038462 to the left, improve=214.59200, (0 missing)  
## DEFICIT\_TO < 0.4696691 to the right, improve=211.37090, (0 missing)  
## DEFICIT\_STL < 0.5797963 to the left, improve=143.32030, (0 missing)  
## DEFICIT\_DREB < 0.457537 to the left, improve= 92.36021, (0 missing)  
## FG3\_PCT < 0.38 to the left, improve= 91.53520, (0 missing)  
## Surrogate splits:  
## DEFICIT\_FTA < 0.310728 to the right, agree=0.667, adj=0.030, (0 split)  
## FG3\_PCT < 0.479 to the left, agree=0.662, adj=0.015, (0 split)  
## FG2\_PCT < 0.5929065 to the left, agree=0.659, adj=0.009, (0 split)  
## DEFICIT\_TO < 0.3066239 to the right, agree=0.659, adj=0.009, (0 split)  
## DEFICIT\_FG3A < 0.6753378 to the left, agree=0.659, adj=0.007, (0 split)  
##   
## Node number 3: 4954 observations, complexity param=0.03353984  
## predicted class=1 expected loss=0.2832055 P(node) =0.5291604  
## class counts: 1403 3551  
## probabilities: 0.283 0.717   
## left son=6 (1772 obs) right son=7 (3182 obs)  
## Primary splits:  
## DEFICIT\_AST < 0.4961538 to the left, improve=275.7802, (0 missing)  
## DEFICIT\_TO < 0.5303309 to the right, improve=244.3928, (0 missing)  
## DEFICIT\_STL < 0.4220648 to the left, improve=162.5502, (0 missing)  
## DEFICIT\_DREB < 0.5419177 to the left, improve=130.4785, (0 missing)  
## FG3\_PCT < 0.3905 to the left, improve=107.4368, (0 missing)  
## Surrogate splits:  
## DEFICIT\_FTA < 0.6879058 to the right, agree=0.655, adj=0.035, (0 split)  
## FG3\_PCT < 0.234 to the left, agree=0.653, adj=0.031, (0 split)  
## FG2\_PCT < 0.4364917 to the left, agree=0.648, adj=0.017, (0 split)  
## DEFICIT\_FG3A < 0.3246622 to the left, agree=0.646, adj=0.010, (0 split)  
## DEFICIT\_TO < 0.6566964 to the right, agree=0.645, adj=0.007, (0 split)  
##   
## Node number 4: 2893 observations  
## predicted class=0 expected loss=0.1434497 P(node) =0.3090152  
## class counts: 2478 415  
## probabilities: 0.857 0.143   
##   
## Node number 5: 1515 observations, complexity param=0.0238197  
## predicted class=0 expected loss=0.4719472 P(node) =0.1618244  
## class counts: 800 715  
## probabilities: 0.528 0.472   
## left son=10 (564 obs) right son=11 (951 obs)  
## Primary splits:  
## DEFICIT\_TO < 0.4696691 to the right, improve=107.86040, (0 missing)  
## DEFICIT\_STL < 0.4749373 to the left, improve= 72.10273, (0 missing)  
## DEFICIT\_FTA < 0.4586749 to the left, improve= 47.81823, (0 missing)  
## DEFICIT\_PF < 0.4470885 to the right, improve= 47.20962, (0 missing)  
## DEFICIT\_DREB < 0.457385 to the left, improve= 38.15017, (0 missing)  
## Surrogate splits:  
## DEFICIT\_STL < 0.49 to the left, agree=0.776, adj=0.397, (0 split)  
## DEFICIT\_FGA < 0.4942195 to the left, agree=0.670, adj=0.113, (0 split)  
## DEFICIT\_FG2A < 0.4693093 to the left, agree=0.661, adj=0.090, (0 split)  
## DEFICIT\_PF < 0.6018868 to the right, agree=0.638, adj=0.027, (0 split)  
## DEFICIT\_DREB < 0.4855042 to the right, agree=0.635, adj=0.020, (0 split)  
##   
## Node number 6: 1772 observations, complexity param=0.03353984  
## predicted class=0 expected loss=0.493228 P(node) =0.1892758  
## class counts: 898 874  
## probabilities: 0.507 0.493   
## left son=12 (1222 obs) right son=13 (550 obs)  
## Primary splits:  
## DEFICIT\_TO < 0.5113695 to the right, improve=116.63410, (0 missing)  
## DEFICIT\_STL < 0.5250627 to the left, improve= 77.35023, (0 missing)  
## DEFICIT\_PF < 0.5222332 to the right, improve= 57.20654, (0 missing)  
## DEFICIT\_FTA < 0.5413251 to the left, improve= 57.08282, (0 missing)  
## DEFICIT\_DREB < 0.542615 to the left, improve= 53.92732, (0 missing)  
## Surrogate splits:  
## DEFICIT\_STL < 0.5227743 to the left, agree=0.792, adj=0.329, (0 split)  
## DEFICIT\_FG2A < 0.5585001 to the left, agree=0.714, adj=0.078, (0 split)  
## DEFICIT\_FGA < 0.5059174 to the left, agree=0.709, adj=0.062, (0 split)  
## FG3\_PCT < 0.683 to the left, agree=0.692, adj=0.007, (0 split)  
## DEFICIT\_FTA < 0.776644 to the left, agree=0.691, adj=0.005, (0 split)  
##   
## Node number 7: 3182 observations  
## predicted class=1 expected loss=0.1587052 P(node) =0.3398846  
## class counts: 505 2677  
## probabilities: 0.159 0.841   
##   
## Node number 10: 564 observations  
## predicted class=0 expected loss=0.2269504 P(node) =0.06024354  
## class counts: 436 128  
## probabilities: 0.773 0.227   
##   
## Node number 11: 951 observations, complexity param=0.01602222  
## predicted class=1 expected loss=0.382755 P(node) =0.1015809  
## class counts: 364 587  
## probabilities: 0.383 0.617   
## left son=22 (371 obs) right son=23 (580 obs)  
## Primary splits:  
## DEFICIT\_DREB < 0.4522179 to the left, improve=57.99040, (0 missing)  
## FG3\_PCT < 0.322 to the left, improve=31.89405, (0 missing)  
## DEFICIT\_PF < 0.4435337 to the right, improve=23.15053, (0 missing)  
## DEFICIT\_TO < 0.4027027 to the right, improve=21.18845, (0 missing)  
## DEFICIT\_FTA < 0.4586749 to the left, improve=20.68016, (0 missing)  
## Surrogate splits:  
## DEFICIT\_TO < 0.3623737 to the left, agree=0.633, adj=0.059, (0 split)  
## DEFICIT\_FGA < 0.5540315 to the right, agree=0.624, adj=0.035, (0 split)  
## FT\_PCT < 0.5095 to the left, agree=0.621, adj=0.030, (0 split)  
## FG2\_PCT < 0.4128709 to the left, agree=0.615, adj=0.013, (0 split)  
## DEFICIT\_FTA < 0.2545696 to the left, agree=0.614, adj=0.011, (0 split)  
##   
## Node number 12: 1222 observations, complexity param=0.02328562  
## predicted class=0 expected loss=0.3715221 P(node) =0.1305277  
## class counts: 768 454  
## probabilities: 0.628 0.372   
## left son=24 (813 obs) right son=25 (409 obs)  
## Primary splits:  
## DEFICIT\_DREB < 0.5477821 to the left, improve=84.22482, (0 missing)  
## DEFICIT\_PF < 0.5533815 to the right, improve=31.72801, (0 missing)  
## FG3\_PCT < 0.408 to the left, improve=25.98388, (0 missing)  
## DEFICIT\_FTA < 0.5413251 to the left, improve=25.69796, (0 missing)  
## DEFICIT\_TO < 0.5972973 to the right, improve=24.78092, (0 missing)  
## Surrogate splits:  
## FG3\_PCT < 0.518 to the left, agree=0.683, adj=0.054, (0 split)  
## DEFICIT\_TO < 0.6376263 to the left, agree=0.674, adj=0.027, (0 split)  
## DEFICIT\_FGA < 0.4306688 to the right, agree=0.673, adj=0.022, (0 split)  
## FG2\_PCT < 0.6044089 to the left, agree=0.669, adj=0.012, (0 split)  
## DEFICIT\_FTA < 0.2481406 to the right, agree=0.667, adj=0.005, (0 split)  
##   
## Node number 13: 550 observations  
## predicted class=1 expected loss=0.2363636 P(node) =0.05874813  
## class counts: 130 420  
## probabilities: 0.236 0.764   
##   
## Node number 22: 371 observations  
## predicted class=0 expected loss=0.3989218 P(node) =0.03962828  
## class counts: 223 148  
## probabilities: 0.601 0.399   
##   
## Node number 23: 580 observations  
## predicted class=1 expected loss=0.2431034 P(node) =0.06195257  
## class counts: 141 439  
## probabilities: 0.243 0.757   
##   
## Node number 24: 813 observations  
## predicted class=0 expected loss=0.2398524 P(node) =0.08684042  
## class counts: 618 195  
## probabilities: 0.760 0.240   
##   
## Node number 25: 409 observations  
## predicted class=1 expected loss=0.3667482 P(node) =0.04368725  
## class counts: 150 259  
## probabilities: 0.367 0.633

plot(fit, uniform = TRUE, main="Classification Tree for nba stats")  
text(fit, use.n = TRUE, all = TRUE, cex=.8)

