library(tidyverse)

## -- Attaching packages ------------------------------------------------ tidyverse 1.2.1 --

## v ggplot2 3.1.0 v purrr 0.3.2   
## v tibble 2.1.1 v dplyr 0.8.0.1  
## v tidyr 0.8.3 v stringr 1.4.0   
## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts --------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

library(rpart)  
library(rattle)

## Rattle: A free graphical interface for data science with R.  
## Version 5.2.0 Copyright (c) 2006-2018 Togaware Pty Ltd.  
## Type 'rattle()' to shake, rattle, and roll your data.

library(RColorBrewer)

Parole=read.csv("parole.csv")  
Parole=Parole%>% mutate(male= as\_factor(as.character(male))) %>%  
 mutate(male= fct\_recode(male,  
 "female"="0",  
 "male"= "1"))  
Parole=Parole%>% mutate(race= as\_factor(as.character(race))) %>%  
 mutate(race= fct\_recode(race,  
 "white"="1",  
 "otherwise"= "2"))  
Parole=Parole%>% mutate(multiple.offenses= as\_factor(as.character(multiple.offenses))) %>%  
 mutate(multiple.offenses= fct\_recode(multiple.offenses,  
 "multiple offenses"="0",  
 "otherwise"= "1"))  
Parole=read.csv("parole.csv")  
Parole=Parole%>% mutate(male= as\_factor(as.character(male))) %>%  
 mutate(male= fct\_recode(male,  
 "female"="0",  
 "male"= "1"))  
Parole=Parole%>% mutate(race= as\_factor(as.character(race))) %>%  
 mutate(race= fct\_recode(race,  
 "white"="1",  
 "otherwise"= "2"))  
Parole=Parole%>% mutate(multiple.offenses= as\_factor(as.character(multiple.offenses))) %>%  
 mutate(multiple.offenses= fct\_recode(multiple.offenses,  
 "multiple offenses"="0",  
 "otherwise"= "1"))  
Parole=Parole%>% mutate(violator= as\_factor(as.character(violator))) %>%  
 mutate(violator= fct\_recode(violator,  
 " completed"="0",  
 "violated"= "1"))  
Parole=Parole%>% mutate(state= as\_factor(as.character(state))) %>%  
 mutate(state= fct\_recode(state,  
 "any other state"="1",  
 "Kentucky"="2",   
 "Louisiana"= "3",  
 "Virginia"= "4"))  
Parole=Parole%>% mutate(crime= as\_factor(as.character(crime))) %>%  
 mutate(crime= fct\_recode(crime,  
 "any other crime"="1",  
 "larceny"="2",   
 "drug related"= "3",  
 "driving related"= "4"))

str(Parole)

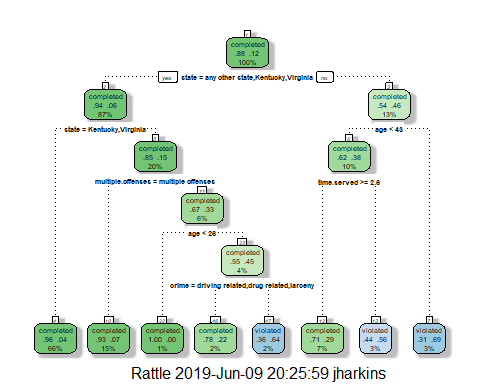
## 'data.frame': 675 obs. of 9 variables:  
## $ male : Factor w/ 2 levels "male","female": 1 2 1 1 1 1 1 2 2 1 ...  
## $ race : Factor w/ 2 levels "white","otherwise": 1 1 2 1 2 2 1 1 1 2 ...  
## $ age : num 33.2 39.7 29.5 22.4 21.6 46.7 31 24.6 32.6 29.1 ...  
## $ state : Factor w/ 4 levels "any other state",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ time.served : num 5.5 5.4 5.6 5.7 5.4 6 6 4.8 4.5 4.7 ...  
## $ max.sentence : int 18 12 12 18 12 18 18 12 13 12 ...  
## $ multiple.offenses: Factor w/ 2 levels "multiple offenses",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ crime : Factor w/ 4 levels "driving related",..: 1 2 2 3 3 1 2 3 2 4 ...  
## $ violator : Factor w/ 2 levels " completed","violated": 1 1 1 1 1 1 1 1 1 1 ...

summary(Parole)

## male race age state   
## male :545 white :389 Min. :18.40 any other state:143   
## female:130 otherwise:286 1st Qu.:25.35 Kentucky :120   
## Median :33.70 Louisiana : 82   
## Mean :34.51 Virginia :330   
## 3rd Qu.:42.55   
## Max. :67.00   
## time.served max.sentence multiple.offenses  
## Min. :0.000 Min. : 1.00 multiple offenses:313   
## 1st Qu.:3.250 1st Qu.:12.00 otherwise :362   
## Median :4.400 Median :12.00   
## Mean :4.198 Mean :13.06   
## 3rd Qu.:5.200 3rd Qu.:15.00   
## Max. :6.000 Max. :18.00   
## crime violator   
## driving related:101 completed:597   
## drug related :153 violated : 78   
## any other crime:315   
## larceny :106   
##   
##

set.seed(12345)  
train.rows = createDataPartition(y = Parole$violator, p=0.7, list = FALSE)  
train2 = Parole[train.rows,]   
test2 = Parole[-train.rows,]

tree1 = rpart(violator ~., train2, method="class")  
fancyRpartPlot(tree1)

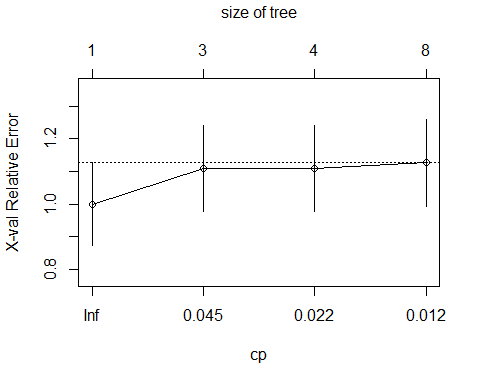


Since the individual was from Louisiana I would answer no to the first question and move to the left. THe next question age < 43 would be a yes answer and I would go to the left for the third question of time served >= to 2.6 years and I would arrive that I would complete parole.

printcp(tree1)

##   
## Classification tree:  
## rpart(formula = violator ~ ., data = train2, method = "class")  
##   
## Variables actually used in tree construction:  
## [1] age crime multiple.offenses state   
## [5] time.served   
##   
## Root node error: 55/473 = 0.11628  
##   
## n= 473   
##   
## CP nsplit rel error xerror xstd  
## 1 0.054545 0 1.00000 1.0000 0.12676  
## 2 0.036364 2 0.89091 1.1091 0.13253  
## 3 0.013636 3 0.85455 1.1091 0.13253  
## 4 0.010000 7 0.80000 1.1273 0.13345

plotcp(tree1)



Task 4 the CP value of .036 should be used.

treepred = predict(tree1, train2, type = "class")  
head(treepred)

## 1 3 4 5 6 7   
## completed completed completed completed completed completed   
## Levels: completed violated

confusionMatrix(treepred,train2$violator)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction completed violated  
## completed 402 28  
## violated 16 27  
##   
## Accuracy : 0.907   
## 95% CI : (0.8771, 0.9316)  
## No Information Rate : 0.8837   
## P-Value [Acc > NIR] : 0.06272   
##   
## Kappa : 0.5   
##   
## Mcnemar's Test P-Value : 0.09725   
##   
## Sensitivity : 0.9617   
## Specificity : 0.4909   
## Pos Pred Value : 0.9349   
## Neg Pred Value : 0.6279   
## Prevalence : 0.8837   
## Detection Rate : 0.8499   
## Detection Prevalence : 0.9091   
## Balanced Accuracy : 0.7263   
##   
## 'Positive' Class : completed   
##

treepred\_test = predict(tree1, newdata=test2, type = "class")  
head(treepred\_test)

## 2 11 13 14 17 18   
## completed violated completed completed completed completed   
## Levels: completed violated

confusionMatrix(treepred\_test,test2$violator)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction completed violated  
## completed 170 19  
## violated 9 4  
##   
## Accuracy : 0.8614   
## 95% CI : (0.8059, 0.9059)  
## No Information Rate : 0.8861   
## P-Value [Acc > NIR] : 0.88631   
##   
## Kappa : 0.1525   
##   
## Mcnemar's Test P-Value : 0.08897   
##   
## Sensitivity : 0.9497   
## Specificity : 0.1739   
## Pos Pred Value : 0.8995   
## Neg Pred Value : 0.3077   
## Prevalence : 0.8861   
## Detection Rate : 0.8416   
## Detection Prevalence : 0.9356   
## Balanced Accuracy : 0.5618   
##   
## 'Positive' Class : completed   
##

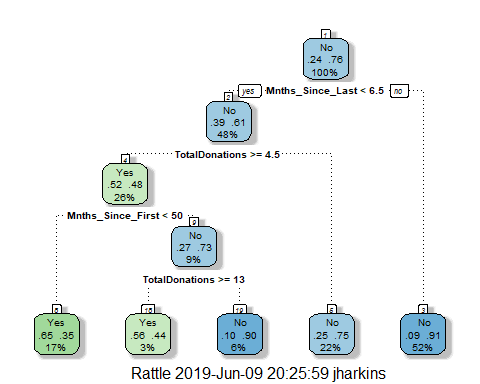
The accuracy of the model is 86 percent on the test model. This is a fairly high number and a good predictor of bhavior.

Blood=read.csv("blood.csv")

Blood=Blood%>% mutate(DonatedMarch= as\_factor(as.character(DonatedMarch))) %>%  
 mutate(DonatedMarch= fct\_recode(DonatedMarch,  
 " No"="0",  
 "Yes"= "1"))

set.seed(1234)  
train.rows = createDataPartition(y = Blood$DonatedMarch, p=0.7, list = FALSE)  
train = Blood[train.rows,]   
test = Blood[-train.rows,]

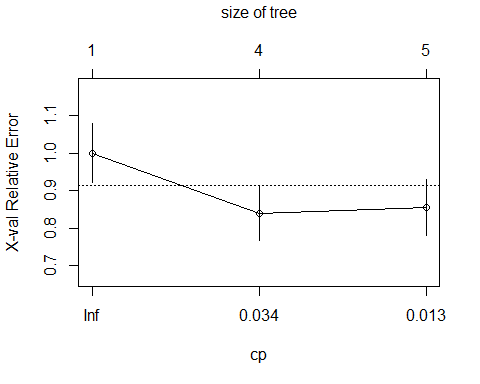
tree3 = rpart(DonatedMarch ~., train, method="class")  
fancyRpartPlot(tree3)



printcp(tree3)

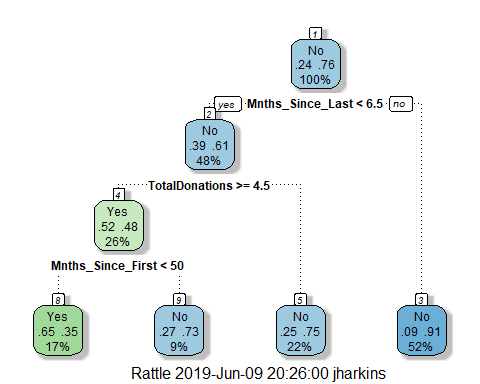
##   
## Classification tree:  
## rpart(formula = DonatedMarch ~ ., data = train, method = "class")  
##   
## Variables actually used in tree construction:  
## [1] Mnths\_Since\_First Mnths\_Since\_Last TotalDonations   
##   
## Root node error: 125/524 = 0.23855  
##   
## n= 524   
##   
## CP nsplit rel error xerror xstd  
## 1 0.072 0 1.000 1.000 0.078049  
## 2 0.016 3 0.784 0.840 0.073304  
## 3 0.010 4 0.768 0.856 0.073822

plotcp(tree3)



The cp is .010 with 4 splits. This model has alot of breaks and inconsistencies. If your total donations are greater than 4.5 and it has been less than 50 months since your first donation you will donate blood. Or if your total donations are greater or equal to 13 you will donate blood.

tree4 = prune(tree3,cp= tree3$cptable[which.min(tree3$cptable[,"xerror"]),"CP"])  
fancyRpartPlot(tree4)



This model is a little clearer than the previous version, if your total donations are greater than 4.5 and it has been less than 50 months since your first donation you will donate.