HW 07

Tuan Bui, SBU ID: 113141951

11/1/2021

library(rattle)

## Loading required package: tibble

## Loading required package: bitops

## Rattle: A free graphical interface for data science with R.  
## Version 5.4.0 Copyright (c) 2006-2020 Togaware Pty Ltd.  
## Type 'rattle()' to shake, rattle, and roll your data.

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✓ ggplot2 3.3.5 ✓ dplyr 1.0.7  
## ✓ tidyr 1.1.3 ✓ stringr 1.4.0  
## ✓ readr 2.0.1 ✓ forcats 0.5.1  
## ✓ purrr 0.3.4

## ── 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)  
  
titanic\_data <- read.csv('~/OneDrive - Stony Brook University/SBU/MAT + AMS/Fall 2021/AMS 380/hw/07/Titanic.csv', header = T)

# Question 01:

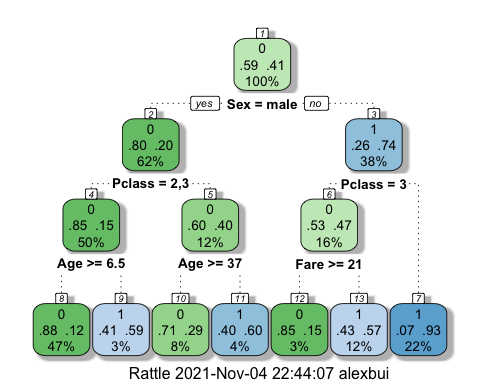
titanic\_data <- subset(titanic\_data, select = -c(Name, Ticket, Cabin))  
  
titanic\_data <- na.omit(titanic\_data)  
str(titanic\_data)

## 'data.frame': 714 obs. of 9 variables:  
## $ PassengerId: int 1 2 3 4 5 7 8 9 10 11 ...  
## $ Survived : int 0 1 1 1 0 0 0 1 1 1 ...  
## $ Pclass : int 3 1 3 1 3 1 3 3 2 3 ...  
## $ Sex : chr "male" "female" "female" "female" ...  
## $ Age : num 22 38 26 35 35 54 2 27 14 4 ...  
## $ SibSp : int 1 1 0 1 0 0 3 0 1 1 ...  
## $ Parch : int 0 0 0 0 0 0 1 2 0 1 ...  
## $ Fare : num 7.25 71.28 7.92 53.1 8.05 ...  
## $ Embarked : chr "S" "C" "S" "S" ...  
## - attr(\*, "na.action")= 'omit' Named int [1:177] 6 18 20 27 29 30 32 33 37 43 ...  
## ..- attr(\*, "names")= chr [1:177] "6" "18" "20" "27" ...

# There are 714 observations left after omitting the missing data  
  
titanic\_data$Survived <- as.factor(titanic\_data$Survived)  
titanic\_data$Pclass <- as.factor(titanic\_data$Pclass)  
  
# Generate training and testing data  
set.seed(123)  
training.samples <- titanic\_data$Survived %>%   
 createDataPartition(p = 0.8, list = FALSE)  
train.data <- titanic\_data[training.samples, ]  
test.data <- titanic\_data[-training.samples, ]

# Question 02:

fit <- rpart(Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked, data = train.data, method="class")  
  
fancyRpartPlot(fit)

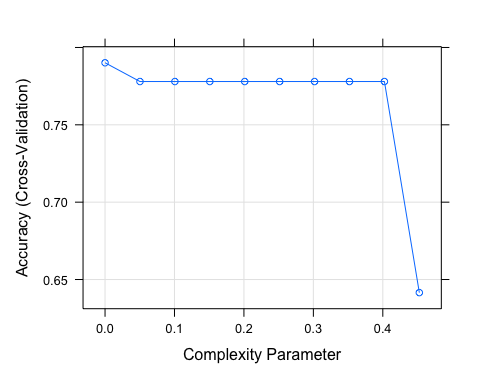


predicted.classes <- fit %>%   
 predict(test.data, type = "class")  
  
predicted.survival <- data.frame(PassengerId = test.data$PassengerId, Survived = predicted.classes)  
predicted.survival

## PassengerId Survived  
## 1 1 0  
## 4 4 1  
## 7 7 0  
## 14 14 0  
## 19 19 1  
## 24 24 1  
## 35 35 1  
## 38 38 0  
## 39 39 1  
## 53 53 1  
## 57 57 1  
## 81 81 0  
## 82 82 0  
## 84 84 1  
## 85 85 1  
## 94 94 0  
## 97 97 0  
## 98 98 1  
## 99 99 1  
## 100 100 0  
## 113 113 0  
## 115 115 1  
## 116 116 0  
## 117 117 0  
## 119 119 1  
## 123 123 0  
## 126 126 0  
## 131 131 0  
## 133 133 1  
## 135 135 0  
## 140 140 1  
## 148 148 0  
## 157 157 1  
## 163 163 0  
## 166 166 0  
## 174 174 0  
## 176 176 0  
## 179 179 0  
## 180 180 0  
## 190 190 0  
## 194 194 1  
## 209 209 1  
## 212 212 1  
## 214 214 0  
## 221 221 0  
## 226 226 0  
## 228 228 0  
## 243 243 0  
## 249 249 0  
## 263 263 0  
## 266 266 0  
## 274 274 0  
## 283 283 0  
## 284 284 0  
## 293 293 0  
## 306 306 1  
## 309 309 0  
## 311 311 1  
## 312 312 1  
## 324 324 1  
## 328 328 1  
## 341 341 1  
## 342 342 1  
## 343 343 0  
## 345 345 0  
## 354 354 0  
## 370 370 1  
## 378 378 1  
## 380 380 0  
## 383 383 0  
## 386 386 0  
## 391 391 1  
## 393 393 0  
## 400 400 1  
## 406 406 0  
## 408 408 1  
## 409 409 0  
## 438 438 1  
## 442 442 0  
## 446 446 1  
## 447 447 1  
## 448 448 1  
## 453 453 1  
## 461 461 0  
## 462 462 0  
## 463 463 0  
## 468 468 0  
## 477 477 0  
## 488 488 0  
## 494 494 0  
## 509 509 0  
## 514 514 1  
## 516 516 0  
## 538 538 1  
## 545 545 0  
## 551 551 1  
## 572 572 1  
## 582 582 1  
## 583 583 0  
## 584 584 1  
## 586 586 1  
## 600 600 0  
## 601 601 1  
## 617 617 0  
## 619 619 1  
## 632 632 0  
## 639 639 0  
## 646 646 0  
## 652 652 1  
## 653 653 0  
## 662 662 0  
## 678 678 1  
## 682 682 1  
## 690 690 1  
## 692 692 1  
## 694 694 0  
## 711 711 1  
## 727 727 1  
## 731 731 1  
## 732 732 0  
## 746 746 0  
## 762 762 0  
## 768 768 1  
## 771 771 0  
## 781 781 1  
## 788 788 0  
## 797 797 1  
## 800 800 0  
## 805 805 0  
## 815 815 0  
## 818 818 0  
## 819 819 0  
## 828 828 1  
## 834 834 0  
## 836 836 1  
## 841 841 0  
## 849 849 0  
## 851 851 1  
## 861 861 0  
## 873 873 1  
## 880 880 1  
## 888 888 1

# Question 03:

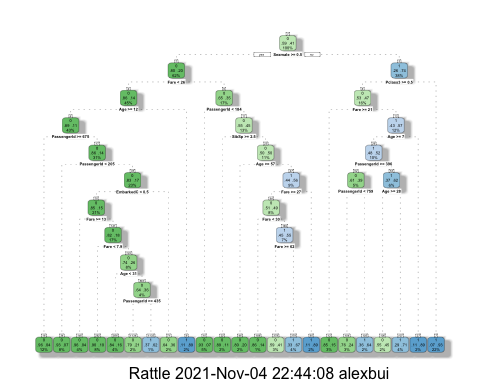
# Fit the model on the training set  
set.seed(123)  
model <- train(  
 Survived ~., data = train.data, method = "rpart",  
 trControl = trainControl("cv", number = 10),  
 tuneLength = 10  
 )  
  
plot(model)



model$bestTune

## cp  
## 1 0

# The best CP value is 0  
  
fancyRpartPlot(model$finalModel)



# Question 04:

predicted.classes2 <- model %>% predict(test.data)  
# Compute model accuracy rate on test data  
  
# confusion matrix  
table(predicted.classes2, test.data$Survived)

##   
## predicted.classes2 0 1  
## 0 71 14  
## 1 13 44

mean(predicted.classes2 == test.data$Survived)

## [1] 0.8098592

# The overall accuracy of the test data is 0.8098592  
  
sum((test.data$Survived == 1)\*(predicted.classes2 == 1))/sum(test.data$Survived ==  
1)

## [1] 0.7586207

# The sensitivity of the test data is 0.7586207  
  
sum((test.data$Survived == 0)\*(predicted.classes2 == 0))/sum(test.data$Survived ==  
0)

## [1] 0.8452381

# The specificity of the test data is 0.8452381