R_Project: Classification - Logistic Regression/ KNN/ Decision Tree

Celio F kelly

2022-06-30

Data info

- Data Name: Airline Passenger Satisfaction.
- Database source: https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction
- Data type: .CSV
- Last Update: 2 years ago.
- Search/ Downloaded Date: 29, June, 2022.
- Rows:103910; Columns:25

Purpose:

Use the data and the algorithms listed above to predict passenger satisfaction based in some variables, as age, class, flight distance, food and drink, and check-in service.

Logistic Regression Algorithm

Steps to get Data into R and necessary libraries.

```
library(readr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.6     v dplyr   1.0.9
## v tibble  3.1.7     v stringr 1.4.0
## v tidyr   1.2.0     v forcats 0.5.1
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

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

```r
library(ggplot2)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(viridis)
```

```
## Loading required package: viridisLite
```

```r
library(viridisLite)
library(class)
library(e1071)
library(caTools)
library(tree)

ps <- read.csv("~/Downloads/UTD/MachineLearn/R-Project/Classification/Airline_Passenger_Satisfaction.csv
```

Steps to Data Cleaning

- 1°: removing unnecessary columns.
- 2°: check for NA's and remove them.
- 3°: converting necessary columns as factor.
- 4°: boxplot to analyze the data.
- 5°: check total of passenger satisfied or neutral/ dissatisfied.

Comments: there are two columns that is useless for this prediction. Luckily there is only one column that contains NA's values, 'Arrival Delay in Minutes' there are 310 NA's. After removing those rows with NA's values still left over 103599 rows that is good enough for this project, so we do not need to do any adjustment on the data to fill up NA's values. Make a graph to analyze how the data are spread, and have an idea what is the passenger satisfaction rate comparing with the predictor 'Age'. We generate two graphs to compare 'satisfied' and 'neutral or dissatisfied', as the graphs shows the mean for 'neutral or dissatisfied' is about 38 years old, and there are way more passenger dissatisfied also did not show any outlier. On the other hand we see 'satisfied' passenger mean that is about 43 years old, there are less client and a few outliers.

```r
# remove unnecessary columns
ps$X <- NULL
ps$id <- NULL

# checking columns with NA's
sapply(ps, function(x) sum(is.na(x)==TRUE))
```

```
##                            Gender                     Customer.Type 
##                                 0                                 0 
##                               Age                    Type.of.Travel 
##                                 0                                 0 
##                             Class                   Flight.Distance 
##                                 0                                 0 
##             Inflight.wifi.service Departure.Arrival.time.convenient 
##                                 0                                 0 
##            Ease.of.Online.booking                     Gate.location 
##                                 0                                 0 
##                    Food.and.drink                   Online.boarding 
##                                 0                                 0 
##                      Seat.comfort            Inflight.entertainment 
##                                 0                                 0 
##                  On.board.service                  Leg.room.service 
##                                 0                                 0 
##                  Baggage.handling                   Checkin.service 
##                                 0                                 0 
##                  Inflight.service                       Cleanliness 
##                                 0                                 0 
##        Departure.Delay.in.Minutes          Arrival.Delay.in.Minutes 
##                                 0                               310 
##                      satisfaction 
##                                 0
```

```r
# remove NA's rows
ps <- ps %>% drop_na()

# converting columns to factor
ps$satisfaction <- factor(ps$satisfaction)
ps$Class <- factor(ps$Class)


# boxplot
qplot(data= ps, x=satisfaction, y=Age, fill=satisfaction, geom='boxplot') +
  geom_boxplot(color="black", outlier.size = 0.5) +
  geom_jitter(shape="+", color='#9d0bf7', size=0.4, alpha=1.4) +
  labs(title = "Passanger Satisfaction vs Age", xlab= "Satisfaction", ylab= "Age")
```

Steps to do Data Exploration

- 1°: some data analysis.

Checking some values from the data using str function to see, min, max, mean, median and also checking variables type. The last line shows the total of 'satisfied' and 'neutral or dissatisfied' that's confirme what we saw about in the graph.

```
head(ps)
```

```
##   Gender     Customer.Type Age  Type.of.Travel    Class Flight.Distance
## 1   Male    Loyal Customer  13 Personal Travel Eco Plus             460
## 2   Male disloyal Customer  25 Business travel Business             235
## 3 Female    Loyal Customer  26 Business travel Business            1142
## 4 Female    Loyal Customer  25 Business travel Business             562
## 5   Male    Loyal Customer  61 Business travel Business             214
## 6 Female    Loyal Customer  26 Personal Travel      Eco            1180
##   Inflight.wifi.service Departure.Arrival.time.convenient
## 1                     3                                 4
## 2                     3                                 2
## 3                     2                                 2
## 4                     2                                 5
## 5                     3                                 3
```

```
## 6                     3                                 4
##   Ease.of.Online.booking Gate.location Food.and.drink Online.boarding
## 1                      3             1              5               3
## 2                      3             3              1               3
## 3                      2             2              5               5
## 4                      5             5              2               2
## 5                      3             3              4               5
## 6                      2             1              1               2
##   Seat.comfort Inflight.entertainment On.board.service Leg.room.service
## 1            5                      5                4                3
## 2            1                      1                1                5
## 3            5                      5                4                3
## 4            2                      2                2                5
## 5            5                      3                3                4
## 6            1                      1                3                4
##   Baggage.handling Checkin.service Inflight.service Cleanliness
## 1                4               4                5           5
## 2                3               1                4           1
## 3                4               4                4           5
## 4                3               1                4           2
## 5                4               3                3           3
## 6                4               4                4           1
##   Departure.Delay.in.Minutes Arrival.Delay.in.Minutes            satisfaction
## 1                         25                       18 neutral or dissatisfied
## 2                          1                        6 neutral or dissatisfied
## 3                          0                        0               satisfied
## 4                         11                        9 neutral or dissatisfied
## 5                          0                        0               satisfied
## 6                          0                        0 neutral or dissatisfied
```

```
summary(ps)
```

```
##     Gender          Customer.Type           Age        Type.of.Travel
##  Length:103599      Length:103599      Min.   : 7.00   Length:103599
##  Class :character   Class :character   1st Qu.:27.00   Class :character
##  Mode  :character   Mode  :character   Median :40.00   Mode  :character
##                                        Mean   :39.38
##                                        3rd Qu.:51.00
##                                        Max.   :85.00
##       Class       Flight.Distance Inflight.wifi.service
##  Business:49536   Min.   :  31    Min.   :0.00
##  Eco     :46595   1st Qu.: 414    1st Qu.:2.00
##  Eco Plus: 7468   Median : 842    Median :3.00
##                   Mean   :1189    Mean   :2.73
##                   3rd Qu.:1742    3rd Qu.:4.00
##                   Max.   :4983    Max.   :5.00
##  Departure.Arrival.time.convenient Ease.of.Online.booking Gate.location
##  Min.   :0.00                      Min.   :0.000          Min.   :0.000
##  1st Qu.:2.00                      1st Qu.:2.000          1st Qu.:2.000
##  Median :3.00                      Median :3.000          Median :3.000
##  Mean   :3.06                      Mean   :2.757          Mean   :2.977
##  3rd Qu.:4.00                      3rd Qu.:4.000          3rd Qu.:4.000
##  Max.   :5.00                      Max.   :5.000          Max.   :5.000
##  Food.and.drink  Online.boarding  Seat.comfort  Inflight.entertainment
```

```
##  Min.   :0.000   Min.   :0.00    Min.   :0.00   Min.   :0.000
##  1st Qu.:2.000   1st Qu.:2.00    1st Qu.:2.00   1st Qu.:2.000
##  Median :3.000   Median :3.00    Median :4.00   Median :4.000
##  Mean   :3.202   Mean   :3.25    Mean   :3.44   Mean   :3.358
##  3rd Qu.:4.000   3rd Qu.:4.00    3rd Qu.:5.00   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.00    Max.   :5.00   Max.   :5.000
##  On.board.service Leg.room.service Baggage.handling Checkin.service
##  Min.   :0.000    Min.   :0.000    Min.   :1.000    Min.   :0.000
##  1st Qu.:2.000    1st Qu.:2.000    1st Qu.:3.000    1st Qu.:3.000
##  Median :4.000    Median :4.000    Median :4.000    Median :3.000
##  Mean   :3.383    Mean   :3.351    Mean   :3.632    Mean   :3.304
##  3rd Qu.:4.000    3rd Qu.:4.000    3rd Qu.:5.000    3rd Qu.:4.000
##  Max.   :5.000    Max.   :5.000    Max.   :5.000    Max.   :5.000
##  Inflight.service  Cleanliness    Departure.Delay.in.Minutes
##  Min.   :0.000    Min.   :0.000   Min.   :   0.00
##  1st Qu.:3.000    1st Qu.:2.000   1st Qu.:   0.00
##  Median :4.000    Median :3.000   Median :   0.00
##  Mean   :3.641    Mean   :3.286   Mean   :  14.75
##  3rd Qu.:5.000    3rd Qu.:4.000   3rd Qu.:  12.00
##  Max.   :5.000    Max.   :5.000   Max.   :1592.00
##  Arrival.Delay.in.Minutes                  satisfaction
##  Min.   :   0.00          neutral or dissatisfied:58700
##  1st Qu.:   0.00          satisfied              :44899
##  Median :   0.00
##  Mean   :  15.18
##  3rd Qu.:  13.00
##  Max.   :1584.00
```

```
str(ps)
```

```
## 'data.frame':    103599 obs. of  23 variables:
##  $ Gender                           : chr  "Male" "Male" "Female" "Female" ...
##  $ Customer.Type                    : chr  "Loyal Customer" "disloyal Customer" "Loyal Customer" "Lo
##  $ Age                              : int  13 25 26 25 61 26 47 52 41 20 ...
##  $ Type.of.Travel                   : chr  "Personal Travel" "Business travel" "Business travel" "Bus
##  $ Class                            : Factor w/ 3 levels "Business","Eco",..: 3 1 1 1 1 2 2 1 1 2 ..
##  $ Flight.Distance                  : int  460 235 1142 562 214 1180 1276 2035 853 1061 ...
##  $ Inflight.wifi.service            : int  3 3 2 2 3 3 2 4 1 3 ...
##  $ Departure.Arrival.time.convenient: int  4 2 2 5 3 4 4 3 2 3 ...
##  $ Ease.of.Online.booking           : int  3 3 2 5 3 2 2 4 2 3 ...
##  $ Gate.location                    : int  1 3 2 5 3 1 3 4 2 4 ...
##  $ Food.and.drink                   : int  5 1 5 2 4 1 2 5 4 2 ...
##  $ Online.boarding                  : int  3 3 5 2 5 2 2 5 3 3 ...
##  $ Seat.comfort                     : int  5 1 5 2 5 1 2 5 3 3 ...
##  $ Inflight.entertainment           : int  5 1 5 2 3 1 2 5 1 2 ...
##  $ On.board.service                 : int  4 1 4 2 3 3 3 5 1 2 ...
##  $ Leg.room.service                 : int  3 5 3 5 4 4 3 5 2 3 ...
##  $ Baggage.handling                 : int  4 3 4 3 4 4 4 5 1 4 ...
##  $ Checkin.service                  : int  4 1 4 1 3 4 3 4 4 4 ...
##  $ Inflight.service                 : int  5 4 4 4 3 4 5 5 1 3 ...
##  $ Cleanliness                      : int  5 1 5 2 3 1 2 4 2 2 ...
##  $ Departure.Delay.in.Minutes       : int  25 1 0 11 0 0 9 4 0 0 ...
##  $ Arrival.Delay.in.Minutes         : int  18 6 0 9 0 0 23 0 0 0 ...
##  $ satisfaction                     : Factor w/ 2 levels "neutral or dissatisfied",..: 1 1 2 1 2 1 1
```

```
table(ps$satisfaction)
```

```
##
## neutral or dissatisfied               satisfied
##                   58700                   44899
```

```
median(ps$Age)
```

```
## [1] 40
```

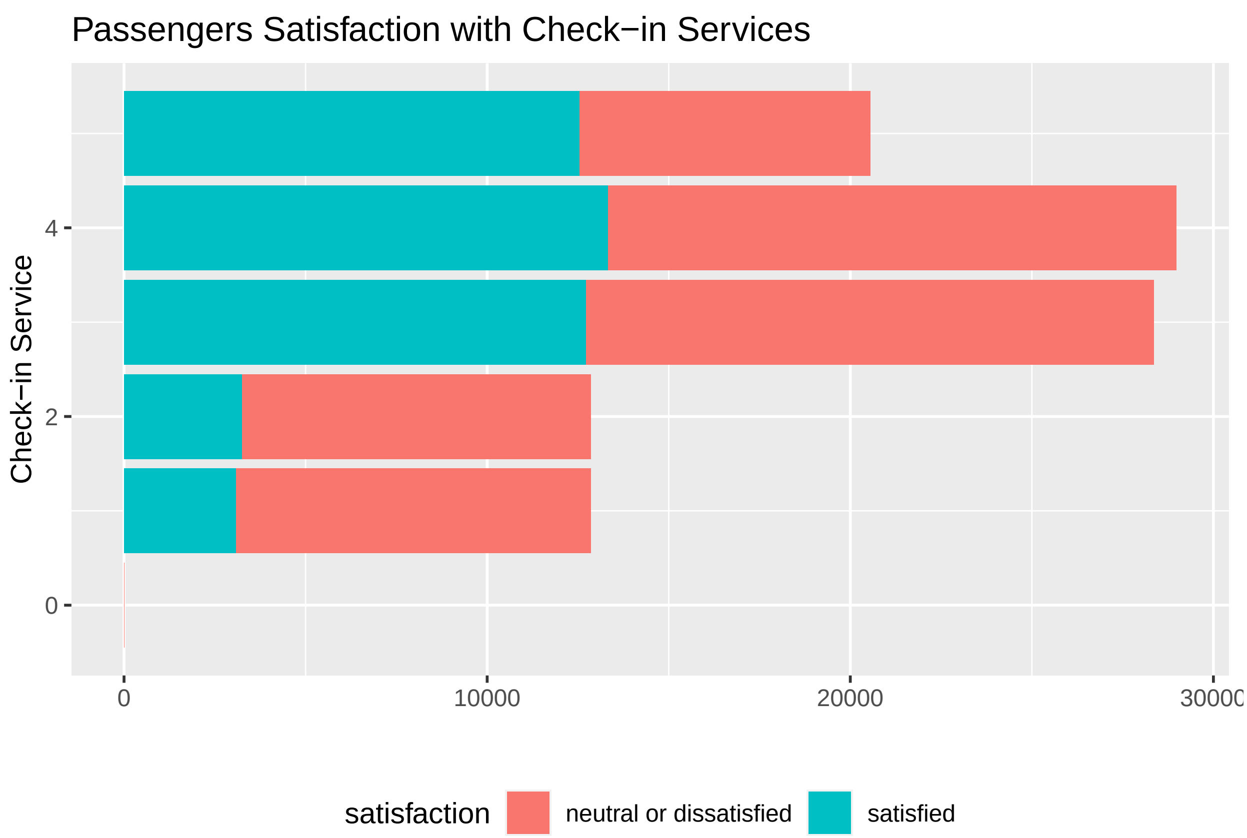
```
mean(ps$Age)
```

```
## [1] 39.37982
```

Steps to Data Exploration (graphs)

- 1s°: graph to satisfied and check-in service.
- 2°: graph analyzing satisfied with class

```
ggplot(ps, aes(y = Checkin.service)) +
  geom_bar(aes(fill = satisfaction), position = position_stack(reverse = FALSE)) +
  theme(legend.position = "bottom") +
  labs(title = "Passengers Satisfaction with Check-in Services", x= "", y= "Check-in Service")
```



Steps for Linear Regression Model

- 1°: dividing the data into 80% train and 20% test.
- 2°: make a logistic regression model with 3 predictors.
- 3°: calculate the probability, prediction and accuracy

Comments: As we can see out of 4 predictors used in this model, only 2 are significantly associated with the target. The coefficient estimated Age has b= 0.0123, which is positive. Meaning that an increase in the age is associated with the probability of the passenger to be satisfied. In the other hand for Class Eco Plus the b= -1.85, meaning a decrease in the probability of the passenger to be satisfied.

The accuracy value is 0.24 not the best result since the good accuracy is equal to 1. At this point I will not assume best or worse algorithm since I will run two more to compare.

```
# divide data into train and test
set.seed(1234)
i <- sample(1:nrow(ps), nrow(ps)*0.8, replace=FALSE)
train <- ps[i,]
test <- ps[-i,]

# make the model
lr_start_time <- Sys.time()
lm1 <- glm(satisfaction ~ Age + Class + Flight.Distance + Food.and.drink + Checkin.service, data= train
lr_end_time <- Sys.time()
summary(lm1)$coef
```

```
##                      Estimate   Std. Error     z value      Pr(>|z|)
## (Intercept)     -2.195448584 4.305467e-02  -50.99211  0.000000e+00
## Age              0.011805559 5.711844e-04   20.66856  6.647197e-95
## ClassEco        -2.048797782 1.947047e-02 -105.22589  0.000000e+00
## ClassEco Plus   -1.697440820 3.418597e-02  -49.65314  0.000000e+00
## Flight.Distance  0.000205056 9.362269e-06   21.90239 2.465180e-106
## Food.and.drink   0.338389494 6.588571e-03   51.36007  0.000000e+00
## Checkin.service  0.325725435 6.973916e-03   46.70624  0.000000e+00
```

```
# calculate probability, prediction and accuracy
probs <- lm1 %>% predict(test, type="response")
pred <- ifelse(probs > 0.5,"neutral or dissatisfied", "satisfied")
acc <- mean(pred == test$satisfaction)

#printing result and time
print(paste("Logistic Regres. - Accuracy: ", acc))
```

```
## [1] "Logistic Regres. - Accuracy:  0.225241312741313"
```

```
print(paste("Logistic Regres. - Time: ", lr_end_time - lr_start_time ))
```

```
## [1] "Logistic Regres. - Time:  0.436469078063965"
```

```
# confuse matrix for logistic regression
table(pred, test$satisfaction)
```

```
##
## pred                         neutral or dissatisfied satisfied
##   neutral or dissatisfied                       2378      6601
##   satisfied                                     9452      2289
```

KNN Algorithm

Steps for KNN

- 1°: divide the data into train and test for KNN classification
- 2°: convert predictors columns on train and test to numeric
- 3°: setting scales for train and test
- 4°: make KNN prediction using k with (3, 15, 26, 34)
- 5°: print results and confuse matrix

Comments: Using different values for K we can see that the accuracy have the same value. Also we have a improvement comparing to the previous algorithm but the final analyses and comparison will be posted at the end after the last technology.

```r
# divide the data into train and test
set.seed(1298)
spt <- sample.split(ps, SplitRatio= 0.7)
ps_train <- subset(ps, spt== "TRUE")
ps_test <- subset(ps, spt== "FALSE")

# convert columns to numeric necessary for KNN classification
ps_train$Age <- as.numeric(ps_train$Age)
ps_train$Class <- as.numeric(ps_train$Class)
ps_train$Checkin.service <- as.numeric(ps_train$Checkin.service)

ps_test$Age <- as.numeric(ps_test$Age)
ps_test$Class <- as.numeric(ps_test$Class)
ps_test$Food.and.drink <- as.numeric(ps_test$Food.and.drink)
ps_test$Flight.Distance <- as.numeric(ps_test$Flight.Distance)
ps_test$Checkin.service <- as.numeric(ps_test$Checkin.service)
str(ps_train)
```

```
## 'data.frame':    72068 obs. of  23 variables:
##  $ Gender                           : chr  "Female" "Female" "Male" "Female" ...
##  $ Customer.Type                    : chr  "Loyal Customer" "Loyal Customer" "Loyal Customer" "Loyal
##  $ Age                              : num  26 25 61 26 52 20 24 53 33 13 ...
##  $ Type.of.Travel                   : chr  "Business travel" "Business travel" "Business travel" "Pe
##  $ Class                            : num  1 1 1 2 1 2 2 2 2 2 ...
##  $ Flight.Distance                  : int  1142 562 214 1180 2035 1061 1182 834 946 486 ...
##  $ Inflight.wifi.service            : int  2 2 3 3 4 3 4 1 4 2 ...
##  $ Departure.Arrival.time.convenient: int  2 5 3 4 3 3 5 4 2 1 ...
##  $ Ease.of.Online.booking           : int  2 5 3 2 4 3 5 4 4 2 ...
##  $ Gate.location                    : int  2 5 3 1 4 4 4 4 3 3 ...
##  $ Food.and.drink                   : int  5 2 4 1 5 2 2 1 4 4 ...
##  $ Online.boarding                  : int  5 2 5 2 5 3 5 1 4 2 ...
##  $ Seat.comfort                     : int  5 2 5 1 5 3 2 1 4 1 ...
```

```
##  $ Inflight.entertainment           : int  5 2 3 1 5 2 2 1 4 4 ...
##  $ On.board.service                 : int  4 2 3 3 5 2 3 1 4 2 ...
##  $ Leg.room.service                 : int  3 5 4 4 5 3 3 1 5 1 ...
##  $ Baggage.handling                 : int  4 3 4 4 5 4 5 3 2 4 ...
##  $ Checkin.service                  : num  4 1 3 4 4 4 3 4 2 1 ...
##  $ Inflight.service                 : int  4 4 3 4 5 3 5 4 2 3 ...
##  $ Cleanliness                      : int  5 2 3 1 4 2 2 1 4 4 ...
##  $ Departure.Delay.in.Minutes       : int  0 11 0 0 4 0 0 28 0 1 ...
##  $ Arrival.Delay.in.Minutes         : int  0 9 0 0 0 0 0 8 0 0 ...
##  $ satisfaction                     : Factor w/ 2 levels "neutral or dissatisfied",..: 2 1 2 1 2 1 1
```

```
# setting the scales
ps_trainSale <- scale(ps_train[,c(3, 5, 6, 11, 18)])
ps_testScale <- scale(ps_test[,c(3, 5, 6, 11, 18)])

# make the knn model for k= 3
knn_start_time <- Sys.time()
kn3_pred <- knn(train= ps_trainSale, test= ps_testScale, cl= ps_train$satisfaction, k= 3)
knn_end_time <- Sys.time()
sp3_error <- mean(kn3_pred != ps_test$satisfaction)

# checking accuracy for k=15
kn15_pred <- knn(train= ps_trainSale, test= ps_testScale, cl= ps_train$satisfaction, k= 15)
sp15_error <- mean(kn15_pred != ps_test$satisfaction)

# checking accuracy for k= 26
kn26_pred <- knn(train= ps_trainSale, test= ps_testScale, cl= ps_train$satisfaction, k= 26)
sp26_error <- mean(kn26_pred != ps_test$satisfaction)

# checking accuracy for k= 34
kn34_pred <- knn(train= ps_trainSale, test= ps_testScale, cl= ps_train$satisfaction, k= 34)
sp34_error <- mean(kn34_pred != ps_test$satisfaction)

print(paste("KNN k= 3 - Accuracy =", 1 - sp3_error))
```

```
## [1] "KNN k= 3 - Accuracy = 0.750277504677936"
```

```
print(paste("KNN k= 13 - Accuracy =", 1 - sp15_error))
```

```
## [1] "KNN k= 13 - Accuracy = 0.79052361168374"
```

```
print(paste("KNN k= 23 - Accuracy =", 1 - sp26_error))
```

```
## [1] "KNN k= 23 - Accuracy = 0.793377945513939"
```

```
print(paste("KNN k= 32 - Accuracy =", 1 - sp34_error))
```

```
## [1] "KNN k= 32 - Accuracy = 0.794995401351051"
```

```
print(paste("KNN avg - Time: ", knn_end_time - knn_start_time ))
```

```
## [1] "KNN avg - Time:  8.61600780487061"
```

```
# confuse matrix for knn
table(ps_test$satisfaction, kn3_pred)
```

```
##                           kn3_pred
##                            neutral or dissatisfied satisfied
##   neutral or dissatisfied                    14361      3582
##   satisfied                                   4292      9296
```

Decision Tree Algorithm

Steps to do Decision Tree

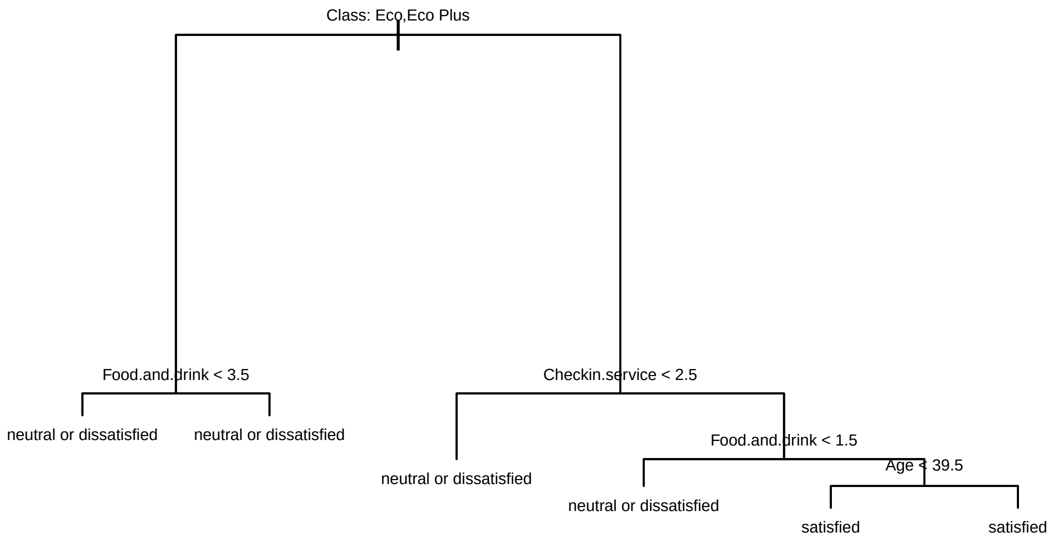
- 1°: make a decision tree model using 5 predictors
- 2°: plot the DT with all predictors based on the target
- 3°: make a prediction and calculate accuracy
- 4°: printing the result and confusing matrix

Comments: The graph of Decision Tree defined that the 'Class' is the best predictor for the target used and Classes (Eco, Eco Plus) have higher probability of satisfied clients, following the tree path, we can see that 'satisfied' and 'neutral or unsatisfied' clients are shown on the tree with each respective probability on top.

```
# making the prediction with the target and predictors
dt_start_time <- Sys.time()
tre <- tree(satisfaction ~ Age + Flight.Distance + Food.and.drink + Class + Checkin.service, data= trai
dt_end_time <- Sys.time()
summary(tre)
```

```
##
## Classification tree:
## tree(formula = satisfaction ~ Age + Flight.Distance + Food.and.drink +
##     Class + Checkin.service, data = train)
## Variables actually used in tree construction:
## [1] "Class"           "Food.and.drink"  "Checkin.service" "Age"
## Number of terminal nodes:  6
## Residual mean deviance:  1.003 = 83150 / 82870
## Misclassification error rate: 0.2183 = 18090 / 82879
```

```
# plotting the prediction
plot(tre)
text(tre, cex= 0.5, pretty= 0)
```




```
# make prediction and find correlation and mse
tre_pred <- predict(tre, newdata =test, type = "class")
tre_acc <- mean(tre_pred == test$satisfaction)

print(paste("Dec. Tree - Accuracy: ", tre_acc))
```

```
## [1] "Dec. Tree - Accuracy:  0.787065637065637"
```

```
print(paste("Dec. Tree - Time: ", dt_end_time - dt_start_time ))
```

```
## [1] "Dec. Tree - Time:  0.15024995803833"
```

```
# confuse matrix for decision tree
table(tre_pred, test$satisfaction)
```

```
##
## tre_pred                  neutral or dissatisfied satisfied
##   neutral or dissatisfied                   10202      2784
##   satisfied                                  1628      6106
```

Final conclusion and analyse.

-Linear Regression:

```
* "Logistic Regres. - Accuracy:  0.225241312741313"
* "Logistic Regres. - Time:  0.267198085784912"
```

-KNN for K=3

```
* "KNN k= 3 - Accuracy = 0.75203780646389"
* "KNN avg - Time:  8.6551718711853"
```

-Scaled KNN for K=13

```
* "KNN k= 13 - Accuracy = 0.786799454470487"
* "KNN avg - Time:  8.6551718711853"
```

-KNN for K=23

```
* "KNN k= 23 - Accuracy = 0.788321862412382"
* "KNN avg - Time:  8.6551718711853"
```

-KNN for K=32

```
* "KNN k= 32 - Accuracy = 0.789431951536681"
* "KNN avg - Time:  8.6551718711853"
```

-Decision Tree

```
* "Dec. Tree - Accuracy:  0.787065637065637"
* "Dec. Tree - Time:  0.242020130157471"
```

Since the best classification algorithm should give an accuracy value equal 1, analyzing the result in this project, we can conclude that the best algorithm in this case taking in consideration result closer to 1 would be KNN where k= 32. Besides Logistic Regression that had the worst accuracy result, KNN and Decision Tree had almost the same results, the large difference that we can take into a count is time, KNN took 8.4 seconds more then Decision Tree. So in conclusion, to decide which technology performed better in this specific case will I would say Decision Tree because it has the lower run time.