Airline Passenger Satisfaction Classification

Süleyman Erim, Giacomo Schiavo, Mattia Varagnolo

30.05.2023

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
# import libraries
library(tidyverse)

data_train = read.csv("train.csv")
data_test = read.csv("test.csv")
```

```
# merge train and test data
data = rbind(data_train, data_test)
attach(data)
```

```
# print dimension of data
print(dim(data))
```

```
## [1] 129880     25
```

```
summary(data)
```

```
##        X                id            Gender          Customer.Type
##  Min.   :     0   Min.   :     1   Length:129880      Length:129880
##  1st Qu.: 16235   1st Qu.: 32471   Class :character   Class :character
##  Median : 38964   Median : 64941   Mode  :character   Mode  :character
##  Mean   : 44159   Mean   : 64941
##  3rd Qu.: 71433   3rd Qu.: 97410
##  Max.   :103903   Max.   :129880
##
##       Age        Type.of.Travel        Class           Flight.Distance
##  Min.   : 7.00   Length:129880      Length:129880      Min.   :  31
##  1st Qu.:27.00   Class :character   Class :character   1st Qu.: 414
##  Median :40.00   Mode  :character   Mode  :character   Median : 844
##  Mean   :39.43                                         Mean   :1190
##  3rd Qu.:51.00                                         3rd Qu.:1744
##  Max.   :85.00                                         Max.   :4983
```

```
##
##  Inflight.wifi.service Departure.Arrival.time.convenient Ease.of.Online.booking
##  Min.   :0.000         Min.   :0.000                     Min.   :0.000
##  1st Qu.:2.000         1st Qu.:2.000                     1st Qu.:2.000
##  Median :3.000         Median :3.000                     Median :3.000
##  Mean   :2.729         Mean   :3.058                     Mean   :2.757
##  3rd Qu.:4.000         3rd Qu.:4.000                     3rd Qu.:4.000
##  Max.   :5.000         Max.   :5.000                     Max.   :5.000
##
##  Gate.location   Food.and.drink  Online.boarding  Seat.comfort
##  Min.   :0.000   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :3.000   Median :3.000   Median :3.000   Median :4.000
##  Mean   :2.977   Mean   :3.205   Mean   :3.253   Mean   :3.441
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:5.000
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000
##
##  Inflight.entertainment On.board.service Leg.room.service Baggage.handling
##  Min.   :0.000          Min.   :0.000    Min.   :0.000    Min.   :1.000
##  1st Qu.:2.000          1st Qu.:2.000    1st Qu.:2.000    1st Qu.:3.000
##  Median :4.000          Median :4.000    Median :4.000    Median :4.000
##  Mean   :3.358          Mean   :3.383    Mean   :3.351    Mean   :3.632
##  3rd Qu.:4.000          3rd Qu.:4.000    3rd Qu.:4.000    3rd Qu.:5.000
##  Max.   :5.000          Max.   :5.000    Max.   :5.000    Max.   :5.000
##
##  Checkin.service Inflight.service  Cleanliness    Departure.Delay.in.Minutes
##  Min.   :0.000   Min.   :0.000    Min.   :0.000   Min.   :   0.00
##  1st Qu.:3.000   1st Qu.:3.000    1st Qu.:2.000   1st Qu.:   0.00
##  Median :3.000   Median :4.000    Median :3.000   Median :   0.00
##  Mean   :3.306   Mean   :3.642    Mean   :3.286   Mean   :  14.71
##  3rd Qu.:4.000   3rd Qu.:5.000    3rd Qu.:4.000   3rd Qu.:  12.00
##  Max.   :5.000   Max.   :5.000    Max.   :5.000   Max.   :1592.00
##
##  Arrival.Delay.in.Minutes satisfaction
##  Min.   :   0.00          Length:129880
##  1st Qu.:   0.00          Class :character
##  Median :   0.00          Mode  :character
##  Mean   :  15.09
##  3rd Qu.:  13.00
##  Max.   :1584.00
##  NA's   :393
```

```
# replace dots with underscores in column names
names(data) = gsub("\\.", "_", names(data))
```

```
# print column names
print(names(data))
```

```
##  [1] "X"                                "id"
##  [3] "Gender"                           "Customer_Type"
##  [5] "Age"                              "Type_of_Travel"
##  [7] "Class"                            "Flight_Distance"
##  [9] "Inflight_wifi_service"            "Departure_Arrival_time_convenient"
```

```
## [11] "Ease_of_Online_booking"            "Gate_location"
## [13] "Food_and_drink"                    "Online_boarding"
## [15] "Seat_comfort"                      "Inflight_entertainment"
## [17] "On_board_service"                  "Leg_room_service"
## [19] "Baggage_handling"                  "Checkin_service"
## [21] "Inflight_service"                  "Cleanliness"
## [23] "Departure_Delay_in_Minutes"        "Arrival_Delay_in_Minutes"
## [25] "satisfaction"
```

```
# drop X and id column
#TODO: explain why
data = data %>% select(-X, -id)
```

```
# insert all categorical variables into a list
categorical_var = c(data$Cleanliness,
  data$Inflight_wifi_service,
  data$Departure_Arrival_time_convenient,
  data$Ease_of_Online_booking,
  data$Gate_location,
  data$Food_and_drink,
  data$Online_boarding,
  data$Seat_comfort,
  data$Inflight_entertainment,
  data$On_board_service,
  data$Leg_room_service,
  data$Baggage_handling,
  data$Checkin_service,
  data$Inflight_service,
  data$Cleanliness
)

# convert categorical variables to factors and then to numeric
categorical_var = as.factor(categorical_var)
categorical_var = as.numeric(categorical_var)

# convert gender to numeric and then to factor
data$Gender = as.numeric(as.factor(data$Gender))

# change type of customer to 0 and disloyal customer to 1
data$Customer_Type = as.numeric(factor(data$Customer_Type, levels = c("Loyal Customer", "disloyal Custo

# change type of tr avel to 0 and personal travel to 1
data$Type_of_Travel = as.numeric(factor(data$Type_of_Travel, levels = c("Personal Travel", "Business tr

# change class Business is 2, Eco Plus is 1 and Eco is 0
data$Class = as.numeric(factor(data$Class, levels = c("Business", "Eco Plus", "Eco"))) - 1

data$satisfaction = as.numeric(factor(data$satisfaction, levels = c("neutral or dissatisfied", "satisfi
```

```
# drop na values in Arrival Delay in Minutes
# TODO: explain why (now it's dropped to simplify the analysis)
data = data %>% drop_na(Arrival_Delay_in_Minutes)
```

```r
# DATA BALANCE: quite balanced
prop.table(table(data$satisfaction))
```

```
##
##         0         1
## 0.5655008 0.4344992
```

```r
# Train-test split
set.seed(123)
train_index = sample(1:nrow(data), 0.8*nrow(data))
# 80% of data is used for training
train = data[train_index,]
# 20% of data is used for testing
test = data[-train_index,]

# print dimension of train and test data
dim(train)
```

```
## [1] 103589     23
```

```r
dim(test)
```

```
## [1] 25898    23
```

```r
# save true values of test satisfaction column
test_true = test$satisfaction
```

```r
# drop satisfaction column from test data
test = test %>% select(-satisfaction)
```

```r
# print proportion of satisfied and dissatisfied customers in train and test data
prop.table(table(train$satisfaction))
```

```
##
##         0         1
## 0.5668845 0.4331155
```

```r
prop.table(table(test_true))
```

```
## test_true
##        0        1
## 0.559966 0.440034
```

DATA ANALYSIS

```r
# save satisfaction column of train data
train_satisfaction = train$satisfaction
```

```
# correlation matrix only for numeric variables
correlation_matrix = cor(train[, sapply(train, is.numeric)])
```

```
# plot correlation matrix
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
corrplot(correlation_matrix, method = "circle")
```