علي جمال برغوث 201910317

هبة وائل عوض 201912014

شذى باسل رداد 201910532

Introduction

- Bank Marketing Data Set, This data is related with direct marketing campaigns (phone calls) of a Portuguese banking institution. The classification goal is to predict if the client will subscribe a term deposit (variable y).

- the tool used is (R).

Data set

- link for the dataset :

https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

- about dataset :
Number of Instances: 45211
Number of Attributes: 16 + output attribute

- Attribute Information:

Input variables:
1 - age (numeric)
2 - job : type of job (categorical: 'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')
3 - marital : marital status (categorical: 'divorced','married','single','unknown'; note: 'divorced' means divorced or widowed)
4 - education (categorical: 'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown')
5 - default: has credit in default? (categorical: 'no','yes','unknown')
6 - housing: has housing loan? (categorical: 'no','yes','unknown')
7 - loan: has personal loan? (categorical: 'no','yes','unknown')
8 - contact: contact communication type (categorical: 'cellular','telephone')
9 - month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')
10 - day_of_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')
11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is

obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.
other attributes:
12 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)
13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)
14 - previous: number of contacts performed before this campaign and for this client (numeric)
15 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')
16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)
17 - cons.price.idx: consumer price index - monthly indicator (numeric)
18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)
19 - euribor3m: euribor 3 month rate - daily indicator (numeric)
20 - nr.employed: number of employees - quarterly indicator (numeric)

Output variable (desired target):
21 - y - has the client subscribed a term deposit? (binary: 'yes','no')

- attached needed library : -
library(dplyr)
library(Hmisc)
library(e1071)

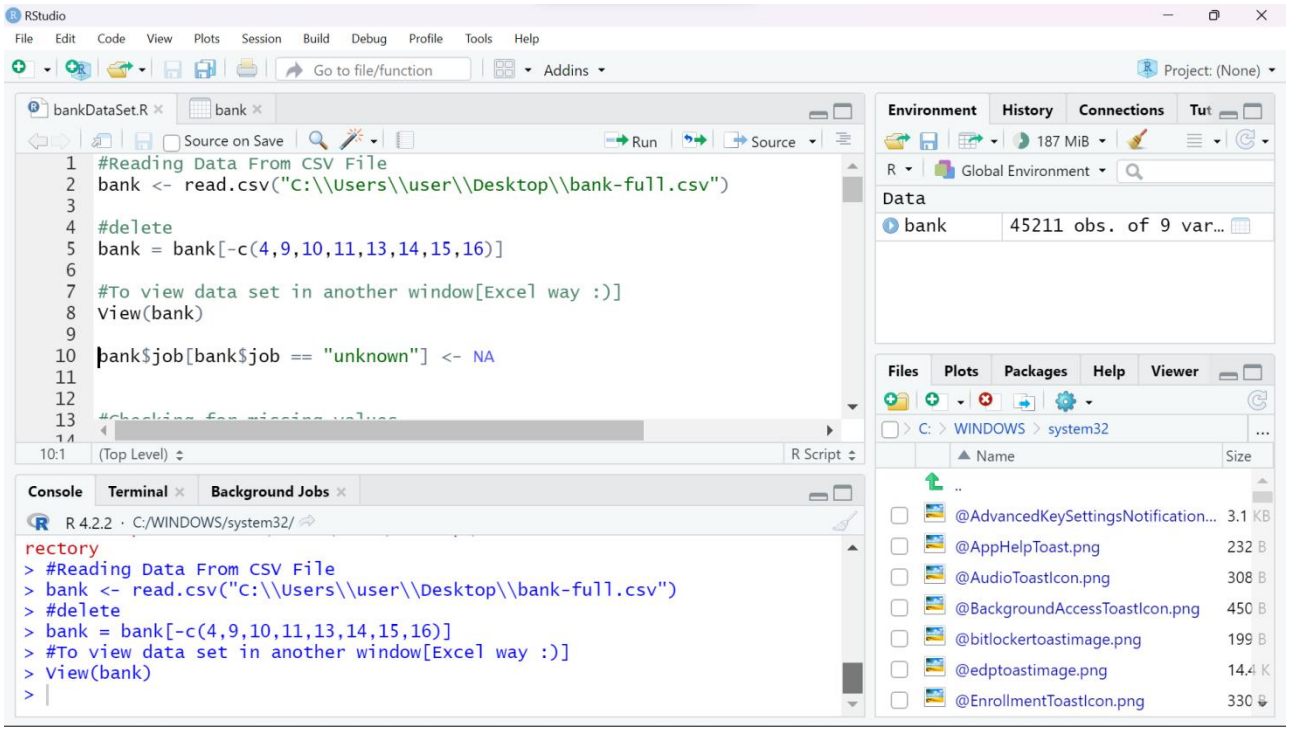
Problem definition

Determine the categories of customers who can deposit in the bank
Data preparation .

cleaning dataset :

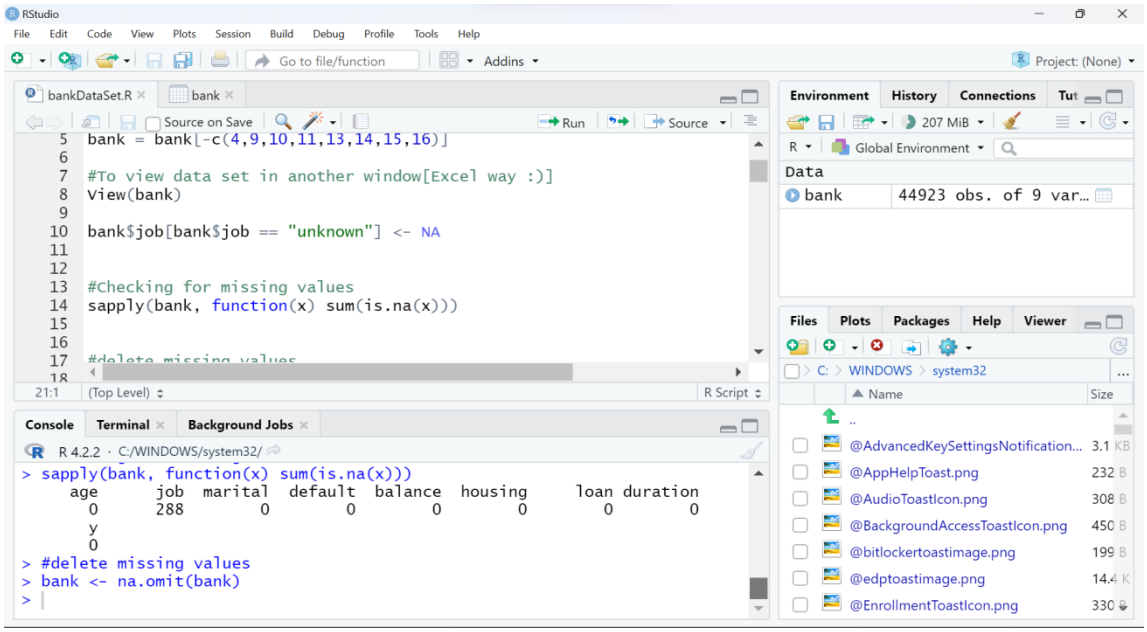
Step 1: Reading dataset from CSV file.
Step 2: Delete this columns (education, day and month), because they are not useful for this analysis.
And delete this columns (pdays, previous, poutcome), because they are duplicated.
And delete contact column because it is not important and contain Missing value.
By using dplyr library

Step 3: view data set in another window[Excel way :)]



Step 4: Convert unknown values to (NA).

Step 5: checking the NA values and delete missing values



<u>Normalization</u>

Step 1: Factoring columns

(<u>default</u> ,<u>housing</u> ,<u>loan</u> ,<u>marital</u> ,<u>job</u>) variable

Step 2: Removing Outliers

(Removing values higher than q3 + 1.5*iqr)

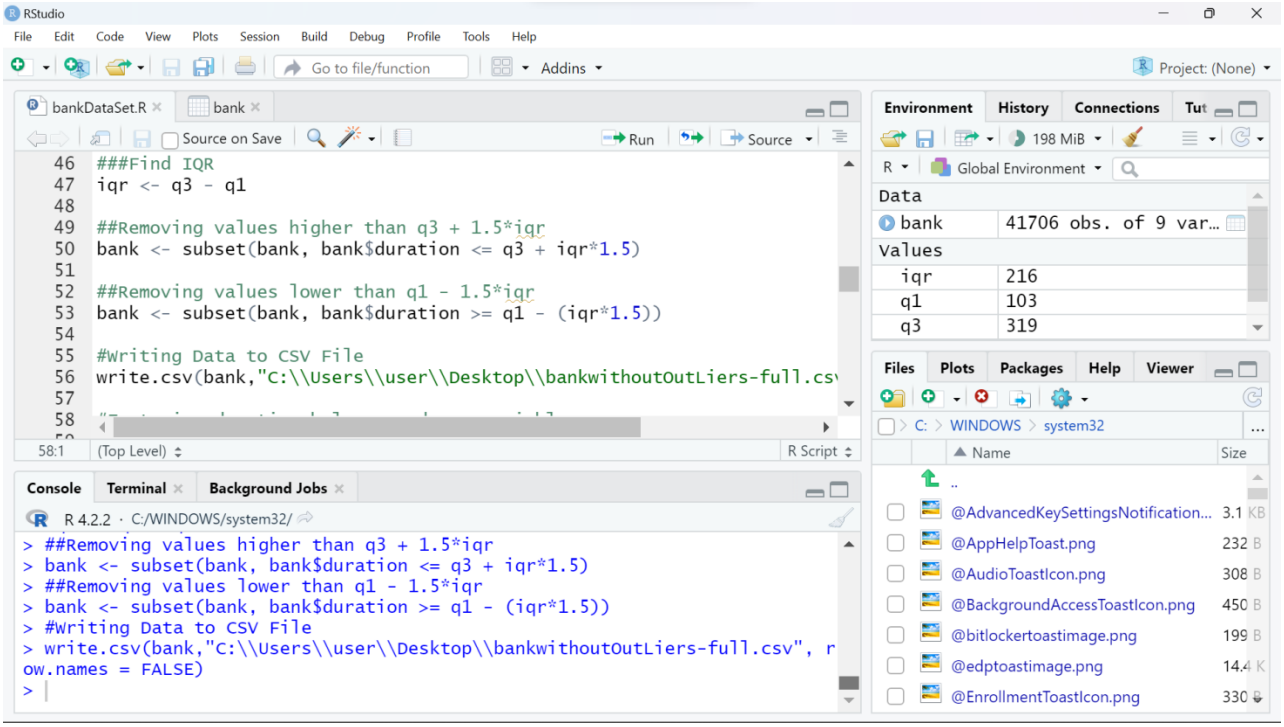
(Removing values lower than q1 - 1.5*iqr)

Tack out outliers in duration column .

```r
#Writing Data to CSV File
write.csv(bank,"C:\\Users\\user\\Desktop\\bank-withoutMissingValues

install.packages("e1071")
library(e1071)

#Factoring default,housing,loan,marital,job variable
bank$default <- as.factor(bank$default)
bank$housing <- as.factor(bank$housing)
bank$loan <- as.factor(bank$loan)
bank$marital <- as.factor(bank$marital)
bank$job <- as.factor(bank$job)

#Removing Outliers
##Calculating IQR
###Summary gives 6 values : min, q1, median, mean, q3, max
summary = summary(bank$duration)
```

```r
#Removing Outliers
##Calculating IQR
###Summary gives 6 values : min, q1, median, mean, q3, max
summary = summary(bank$duration)

####Save q1 and q3 as numeric values
q1 = as.numeric(summary(bank$duration)[2])
q3 = as.numeric(summary(bank$duration)[5])

###Find IQR
iqr <- q3 - q1
```

```
> ###Summary gives 6 values : min, q1, median, mean, q3, max
> summary = summary(bank$duration)
> ####Save q1 and q3 as numeric values
> q1 = as.numeric(summary(bank$duration)[2])
> q3 = as.numeric(summary(bank$duration)[5])
> ###Find IQR
> iqr <- q3 - q1
>
```



Step 3: convert data type of this columns (duration ,balance and age) from numeric to ordinal

.

by dividing the range of the numerical variable into bins and assigning values to each bin.

```r
write.csv(bank,"C:\\Users\\user\\Desktop\\bankwithoutOutLiers-full.cs

#Factoring duration,balance and age variables
#0 - 200 = Short, 200-400 = Normal, <400 = Long
##duration
bank$duration2[bank$duration < 200] <- "Short"
bank$duration2[bank$duration >=200 & bank$duration < 400] <- "Normal"
bank$duration2[bank$duration >= 400] <- "Long"
bank$duration <- bank$duration2
bank$duration <- as.factor(bank$duration)
```

```
> #0 - 200 = Short, 200-400 = Normal, <400 = Long
> ##duration
> bank$duration2[bank$duration < 200] <- "Short"
> bank$duration2[bank$duration >=200 & bank$duration < 400] <- "Normal"
> bank$duration2[bank$duration >= 400] <- "Long"
> bank$duration <- bank$duration2
> bank$duration <- as.factor(bank$duration)
>
```

```r
#>10000 = Low, 10000-60000 = Normal, <60000 = High
##balance
bank$balance2[bank$balance < 10000] <- "Low"
bank$balance2[bank$balance >=10000 & bank$balance < 60000] <- "Normal"
bank$balance2[bank$balance >= 60000] <- "High"
bank$balance <- bank$balance2
bank$balance <- as.factor(bank$balance)

#>25 = Young, 25-60 = Mature, <60 = Old
##age
bank$age2[bank$age < 25] <- "Young"
bank$age2[bank$age >=25 & bank$age < 60] <- "Mature"
```

```
> #>10000 = Low, 10000-60000 = Normal, <60000 = High
> ##balance
> bank$balance2[bank$balance < 10000] <- "Low"
> bank$balance2[bank$balance >=10000 & bank$balance < 60000] <- "Normal"
> bank$balance2[bank$balance >= 60000] <- "High"
> bank$balance <- bank$balance2
> bank$balance <- as.factor(bank$balance)
>
```

```r
##age
bank$age2[bank$age < 25] <- "Young"
bank$age2[bank$age >=25 & bank$age < 60] <- "Mature"
bank$age2[bank$age >= 60] <- "Old"
bank$age <- bank$age2
bank$age <- as.factor(bank$age)


##delete the temporary age2,duratin2,balance2
#delete
bank = bank[-c(10,11,12)]
```

```r
#>25 = Young, 25-60 = Mature, <60 = Old
##age
bank$age2[bank$age < 25] <- "Young"
bank$age2[bank$age >=25 & bank$age < 60] <- "Mature"
bank$age2[bank$age >= 60] <- "Old"
bank$age <- bank$age2
bank$age <- as.factor(bank$age)
```

```
Data
bank        41706 obs. of 12 va...
  $ age       : Factor w/ 3 lev...
  $ job       : Factor w/ 11 le...
  $ marital   : Factor w/ 3 lev...
  $ default   : Factor w/ 2 lev...
  $ balance   : Factor w/ 3 lev...
  $ housing   : Factor w/ 2 lev...
  $ loan      : Factor w/ 2 lev...
  $ duration  : Factor w/ 3 lev...
  $ y         : chr  "no" "no" ...
  $ duration2 : chr  "Normal" "...
  $ balance2  : chr  "Low" "Low...
  $ age2      : chr  "Mature" "...
Values
  jgr         216
```

data mining model

classification:

we used the Naive Bayes classifier, which is quite faster in comparison to other classification algorithms.

The classification goal is to predict if the client will subscribe a term deposit (variable y).

Results:

The accuracy is more than 80%, so we can accept the results.