# Activitat 4

Marc Cervera Rosell

21-06-2024

```
setRepositories(ind=2) # Per descarregar paquets de CRAN
```

# 1 Preprocessament

## 1.1 Variables Income i Year_Birth

```
tryCatch({
  data <- read.csv("marketing.csv", header = TRUE)
  print("Fitxer llegit correctament")
}, error = function(e){
  cat("ERROR en la lectura del fitxer:",conditionMessage(e),"\n")
})
```

```
## [1] "Fitxer llegit correctament"
```

```
columns <- names(data)
type <- sapply(data, class)
for (i in seq_along(columns)) {
  cat("La columna", columns[i], "és de tipus", type[i], "\n")
}
```

```
## La columna ID és de tipus integer
## La columna Year_Birth és de tipus integer
## La columna Education és de tipus character
## La columna Marital_Status és de tipus character
## La columna Income és de tipus integer
## La columna Kidhome és de tipus integer
## La columna Teenhome és de tipus integer
## La columna Dt_Customer és de tipus character
## La columna Recency és de tipus integer
## La columna MntWines és de tipus integer
## La columna MntFruits és de tipus integer
## La columna MntMeatProducts és de tipus integer
## La columna MntFishProducts és de tipus integer
## La columna MntSweetProducts és de tipus integer
## La columna MntGoldProds és de tipus integer
## La columna NumDealsPurchases és de tipus integer
## La columna NumWebPurchases és de tipus integer
## La columna NumCatalogPurchases és de tipus integer
## La columna NumStorePurchases és de tipus integer
## La columna NumWebVisitsMonth és de tipus integer
## La columna AcceptedCmp3 és de tipus integer
## La columna AcceptedCmp4 és de tipus integer
```

```
## La columna AcceptedCmp5 és de tipus integer
## La columna AcceptedCmp1 és de tipus integer
## La columna AcceptedCmp2 és de tipus integer
## La columna Complain és de tipus integer
## La columna Z_CostContact és de tipus integer
## La columna Z_Revenue és de tipus integer
## La columna Response és de tipus integer
```

Després d'observar els tipus de les variables del conjunt de dades, solament es procedirà a fer un canvi de tipus. Aquest canvi es produirà en la variable *Dt_Customer* que passarà de ser de tipus *character* a tipus *date*.

```
data_transformed <- transform(data,
                              Dt_Customer = as.Date(Dt_Customer))
```

```
columns <- names(data_transformed)
type <- sapply(data_transformed, class)
for (i in seq_along(columns)) {
  cat("La columna", columns[i], "és de tipus", type[i], "\n")
}
```

```
## La columna ID és de tipus integer
## La columna Year_Birth és de tipus integer
## La columna Education és de tipus character
## La columna Marital_Status és de tipus character
## La columna Income és de tipus integer
## La columna Kidhome és de tipus integer
## La columna Teenhome és de tipus integer
## La columna Dt_Customer és de tipus Date
## La columna Recency és de tipus integer
## La columna MntWines és de tipus integer
## La columna MntFruits és de tipus integer
## La columna MntMeatProducts és de tipus integer
## La columna MntFishProducts és de tipus integer
## La columna MntSweetProducts és de tipus integer
## La columna MntGoldProds és de tipus integer
## La columna NumDealsPurchases és de tipus integer
## La columna NumWebPurchases és de tipus integer
## La columna NumCatalogPurchases és de tipus integer
## La columna NumStorePurchases és de tipus integer
## La columna NumWebVisitsMonth és de tipus integer
## La columna AcceptedCmp3 és de tipus integer
## La columna AcceptedCmp4 és de tipus integer
## La columna AcceptedCmp5 és de tipus integer
## La columna AcceptedCmp1 és de tipus integer
## La columna AcceptedCmp2 és de tipus integer
## La columna Complain és de tipus integer
## La columna Z_CostContact és de tipus integer
## La columna Z_Revenue és de tipus integer
## La columna Response és de tipus integer
```

S'observa que després de l'aplicació de la funció *transform()* el tipus de la variable *Dt_Customer* queda modificat.

Finalment, cal excloure les variables *Z_CostContact* i *Z_Revenue*, atès que són variables de control i que així s'indica a l'enunciat de l'activitat.

```
columns_to_exclude <- c("Z_CostContact", "Z_Revenue")
data_with_no_control_variables <- data_transformed[, !(names(data_transformed) %in%
                                                         columns_to_exclude)]
print(names(data_with_no_control_variables))
```

```
##  [1] "ID"                 "Year_Birth"          "Education"
##  [4] "Marital_Status"     "Income"              "Kidhome"
##  [7] "Teenhome"           "Dt_Customer"         "Recency"
## [10] "MntWines"           "MntFruits"           "MntMeatProducts"
## [13] "MntFishProducts"    "MntSweetProducts"    "MntGoldProds"
## [16] "NumDealsPurchases"  "NumWebPurchases"     "NumCatalogPurchases"
## [19] "NumStorePurchases"  "NumWebVisitsMonth"   "AcceptedCmp3"
## [22] "AcceptedCmp4"       "AcceptedCmp5"        "AcceptedCmp1"
## [25] "AcceptedCmp2"       "Complain"            "Response"
```

S'observa que en treure per pantalla les columnes del *dataset data_with_no_control_variables* les variables de control indicades anteriorment ja no hi són. Per tant, aquest conjunt final queda completament operatiu per a poder treballar.

## 1.2 Valors absents

Abans de res, encara que l'enunciat ja ho diu, és bona pràctica comprovar si realment hi ha valors absents.

```
any(is.na(data_with_no_control_variables$Income))
```

```
## [1] TRUE
```

```
any(is.na(data_with_no_control_variables$Year_Birth))
```

```
## [1] FALSE
```

```
library(VIM)
```

```
## Loading required package: colorspace
## Loading required package: grid
## VIM is ready to use.
## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues
##
## Attaching package: 'VIM'
## The following object is masked from 'package:datasets':
##
##     sleep
```

Primer cal seleccionar aquelles variables que s'usaran per al càlcul de la distància de Gower. Com s'especifica en l'enunciat seran les variables 10 a 15.

```
variables_gower <- c("MntWines", "MntFruits", "MntMeatProducts", "MntFishProducts",
                     "MntSweetProducts", "MntGoldProds")
```

Com s'indica en les instruccions de l'activitat en desenvolupament, per imputar els valors *NA* de la varible *Income* s'aplica la funció *kNN* de la llibreria *VIM*.

```
income_imputed <- kNN(data_with_no_control_variables, variable = "Income",
                      dist_var = variables_gower, k = 5)
```

Un cop aplicada la funció *kNN* s'observa que la variable *Income* ja no té valors *NA*

```r
any(is.na(income_imputed$Income))
```

```
## [1] FALSE
```

En primer lloc, abans de calcular la mitjana d'edat cal seleccionar les persones que són vídues.

```r
widowed_people <- subset(income_imputed, Marital_Status == "Widow")
```

Un cop seleccionades aquestes persones ja es pot procedir al càlcul de la mitjana d'edat.

```r
mean_widow_age <- mean(widowed_people$Year_Birth)
cat("L'edat mitjana de les persones vidues és:", round(mean_widow_age))
```

```
## L'edat mitjana de les persones vidues és: 1959
```

```r
income_imputed$Year_Birth <- ifelse(is.na(income_imputed$Year_Birth), mean_widow_age,
                                    income_imputed$Year_Birth)
```

```r
any(is.na(income_imputed$Year_Birth))
```

```
## [1] FALSE
```

S'observa que després de l'execució de la cel · la que conté la sentència *ifelse* els valors *NA* de la variable *Year_Birth* ja no són tals.

```r
any(is.na(income_imputed))
```

```
## [1] FALSE
```

Un cop eliminats els valors *NA* de les variables *Income* i *Year_Birth* es pot observar (en la cel · la anterior) el conjunt de dades queda completament lliure de valors *NA*, per tant, cal redenominar aquest conjunt a *markclean*.

```r
markclean <- income_imputed
```

Finalment, es demana una reflexió sobre el nombre de valors *NA* del fitxer, per tant, cal, en primer lloc, comptar aquests valors.

```r
na_per_column <- colSums(is.na(data_with_no_control_variables))
total_na_values <- sum(na_per_column)
cat("El nombre total de valors NA és:", total_na_values)
```

```
## El nombre total de valors NA és: 24
```

Tot i que el nombre de valors *NA* del fitxer no és considerablement elevat, és important abordar aquests valors. El fet de l'existència de valors *NA* pot ser un indicador de problemes en les dades com: problemes en el moment de la recollida de les dades o senzillament que en el moment d'introduir les dades en el fitxer s'ha comès error humà. Com s'acaba d'esmentar, malgrat que el nombre de valors absents no és molt significatiu, és molt important gestionar aquests buits per evitar problemes en les anàlisis que es puguin arribar a fer tant en aquesta activitat com les anàlisis que pugui fer una altra persona completament aliena a la UOC.