

# Tipologia i cicle de dades. Pràctica 2

*Marcos F. Vilaboa & Joaquim Salomon*

*22 de mayo de 2019*

## Índex

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Introducció</b>  | <b>1</b> |
| 1.1      | Competències . . . . .  | 1        |
| 1.2      | Objectius . . . . .   | 2        |
| <b>2</b> | <b>Resolució</b>  | <b>2</b> |
| 2.1      | Descripció del <i>dataset</i> . . . . .                               | 2        |
| 2.1.1    | Càrrega inicial de dades . . . . .                                    | 2        |
| 2.1.2    | Descripció de les variables . . . . .                                 | 3        |
| 2.1.3    | Importància i objectius . . . . .                                     | 3        |
| 2.2      | Pre-processament . . . . .  | 3        |
| 2.2.1    | Integració i selecció de les dades . . . . .                          | 3        |
| 2.2.2    | Neteja de les dades . . . . .   | 4        |
| 2.2.3    | Exportació de les dades preprocessades . . . . .                      | 9        |
| 2.3      | Anàlisi de les dades . . . . .  | 10       |
| 2.3.1    | Selecció dels grups de dades . . . . .                                | 10       |
| 2.3.2    | Comprovació de la normalitat i homogeneïtat de la variància . . . . . | 12       |
| 2.3.3    | Aplicació de proves estadístiques . . . . .                           | 16       |
| 2.4      | Representació dels resultats . . . . .                                | 18       |
| 2.5      | Resolució del problema . . . . .                                      | 18       |

---

## 1 Introducció

En aquesta pràctica s'elabora un cas pràctic orientat a aprendre a identificar les dades rellevants per un projecte analític i usar les eines d'integració, neteja, validació i anàlisi de les mateixes.

### 1.1 Competències

En aquesta pràctica es desenvolupen les següents competències del Màster de Data Science: - Capacitat d'analitzar un problema en el nivell d'abstracció adequat a cada situació i aplicar les habilitats i coneixements adquirits per abordar-lo i resoldre'l. - Capacitat per aplicar les tècniques específiques de tractament de dades (integració, transformació, neteja i validació) per al seu posterior anàlisi.

## 1.2 Objectius

Els objectius concrets d'aquesta pràctica són:

- Aprendre a aplicar els coneixements adquirits i la seva capacitat de resolució de problemes en entorns nous o poc coneguts dintre de contextos més amplis o multidisciplinaris.
- Saber identificar les dades rellevants i els tractaments necessaris (integració, neteja i validació) per dur a terme un projecte analític.
- Aprendre a analitzar les dades adequadament per abordar la informació continguda en les dades.
- Identificar la millor representació dels resultats per tal d'aportar conclusions sobre el problema plantejat en el procés analític.
- Actuar amb els principis ètics i legals relacionats amb la manipulació de dades en funció de l'àmbit d'aplicació.
- Desenvolupar les habilitats d'aprenentatge que els permetin continuar estudiant d'una manera que haurà de ser en gran manera autodirigida o autònoma.
- Desenvolupar la capacitat de cerca, gestió i ús d'informació i recursos en l'àmbit de la ciència de dades.

## 2 Resolució

### 2.1 Descripció del *dataset*

El conjunt de dades utilitzat en el present anàlisi s'ha extret de la web kaggle.com. Concretament s'ha utilitzat el *set* d'entrenament (train.csv) que forma part del total de dades de Titanic: Machine Learning from Disaster (<https://www.kaggle.com/c/titanic/data>).

#### 2.1.1 Càrrega inicial de dades

Per tal de descriure el conjunt, realitzarem una càrrega inicial de les dades amb R:

```
titanic.original <- read.csv("../data/titanic_train.csv", header=TRUE)
str(titanic.original)
```

```
## 'data.frame':    891 obs. of  12 variables:
## $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
## $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
## $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
## $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",...: 109 191 358 277 16 559 520 629 417 58...
## $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
## $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
## $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
## $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
## $ Ticket     : Factor w/ 681 levels "110152","110413",...: 524 597 670 50 473 276 86 396 345 133 ...
## $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
## $ Cabin      : Factor w/ 148 levels "", "A10", "A14",...: 1 83 1 57 1 1 131 1 1 1 ...
## $ Embarked   : Factor w/ 4 levels "", "C", "Q", "S": 4 2 4 4 4 3 4 4 4 2 ...
```

Inicialment, el *dataset* es compon de 12 variables (columnes) amb un total de 891 observacions (registreS).

### 2.1.2 Descripció de les variables

La definició de cada camp és la següent:

- **PassengerId** (*int*): identificador únic del passatger (i de cada registre).
- **Survived** (*int*): si el passatger va sobreviure o no. “0” = No i “1” = Si
- **Pclass** (*int*): classe del bitllet d'embarcament. “1” = primera classe, “2” = segona i “3” = tercera.
- **Name** (*int*): nom del passatger. Inclou el títol com “Mr.”, “Mrs.”, “Dr.”, ...
- **Sex** (*Factor*): gènere del passatger. “female” = dona i “male” = home.
- **Age** (*num*): edat.
- **SibSp** (*Factor*): nombre de germans i cònjuges a bord.
- **Parch** (*int*): nombre de pares i fills a bord.
- **Ticket** (*Factor*): número de tiquet.
- **Fare** (*num*): tarifa del passatger.
- **Cabin** (*Factor*): número de camarot. Consta d'una lletra que significa la coberta i el número de camarot: “A10”, “C85”,...
- **Embarked** (*Factor*): port a on el passatger va embarcar: “C” = Cherbourg, “S” = Southampton i “Q” = Queenstown

### 2.1.3 Importància i objectius

El Titanic es va enfonsar, durant el seu viatge inaugural el 15 d'abril de 1912, xocant amb un iceberg. Van morir 1502 passatgers i tripulants d'un total de 2224.

La raó principal d'aquest número tan important de víctimes de la tragèdia va ser la quantitat escassa de botes salvavides envers el nombre de vides a bord. Es diu que, per preferència, els nens, les dones i la classe alta tenien més possibilitats de sobreviure.

L'objectiu principal d'aquest estudi és el de conèixer si aquesta afirmació és certa. Es pretén doncs, respondre a la pregunta de quin grup de persones va tenir més possibilitats de sobreviure i quin tipus de característiques té.

## 2.2 Pre-processament

### 2.2.1 Integració i selecció de les dades

La integració de les dades consisteix a combinar les dades de diferents fonts de dades. En aquest cas, com que ens basem en un *dataset* concret, no serà necessari integrar més fonts.

En canvi,

```
titanic <- titanic.original[,~which(names(titanic.original) %in% c("Embarked","Ticket","PassengerId"))]
```

## 2.2.2 Neteja de les dades

### 2.2.2.1 Zeros y elements buits

En primer lloc, cal comprovar que les dades no continguin elements buits o zeros. Per a fer-ho, primerament ens fixem en la primera mostra de les dades que s'ha pogut veure unes línies més amunt on es pot veure dades que equivalen a valors buits " " i també valors nul·ls representats com NA. Aleshores, anem a veure quins camps contenen aquestes dades nul·les o buides. Per a veure les dades buides executem la següent funció per veure el nombre d'atributs que contenen algun camp buit.

```
colSums(titanic=="")
```

```
## Survived   Pclass   Name     Sex     Age     SibSp   Parch   Fare
##          0         0         0         0      NA         0         0         0
##   Cabin
##      687
```

I en aquest pas es farà el mateix per a dades nul·les.

```
colSums(is.na(titanic))
```

```
## Survived   Pclass   Name     Sex     Age     SibSp   Parch   Fare
##          0         0         0         0    177         0         0         0
##   Cabin
##          0
```

Així doncs, els atributs *Cabin* i *Age* contenen dades a tractar.

Per a l'atribut *Cabin* veiem que una gran part dels valors de l'atribut són buits o nul·ls, aleshores s'haurà de prescindir d'aquest atribut ja que no pot aportar cap informació rellevant.

Eliminem l'atribut.

```
titanic["Cabin"] <- NULL
```

Per últim, els valors nul·ls de l'atribut *Age* els substituïm per la mitjana dels valors no nul·ls:

```
titanic$Age[is.na(titanic$Age)] <- mean(titanic$Age,na.rm=T)
```

### 2.2.2.2 Valors extrems

A continuació creem una funció per a trobar els valors extrems dins dels atributs numèrics i una altra per esbrinar-los en cas que sigui necessari. En aquest cas concret en tenim quatre: *Age*, *SibSp*, *Parch* i *Fare*

```
seeOutlierValues <- function(dataset,arrayToCheck) {
  mean <- mean(arrayToCheck)
  standardDev <- sd(mean(arrayToCheck))
  min_value <- mean(arrayToCheck)-3*sd(arrayToCheck)
  max_value <- mean(arrayToCheck)+3*sd(arrayToCheck)
  newDatasetWOutliers <- dataset[(arrayToCheck<=min_value | arrayToCheck>=max_value),]
  outliers_count <- nrow(dataset)-nrow(newDatasetWOutliers)
  return (newDatasetWOutliers)
}
```

```

removeOutlierValues <- function(dataset,arrayToCheck) {
  mean <- mean(arrayToCheck)
  standardDev <- sd(mean(arrayToCheck))
  min_value <- mean(arrayToCheck)-3*sd(arrayToCheck)
  max_value <- mean(arrayToCheck)+3*sd(arrayToCheck)
  newDatasetWoOutliers <- dataset[(arrayToCheck>=min_value & arrayToCheck<=max_value),]
  outliers_count <- nrow(dataset)-nrow(newDatasetWoOutliers)
  cat("From", deparse(substitute(arrayToCheck)), outliers_count, "skipped tuples", "\n\n")
  return (newDatasetWoOutliers)
}

```

Primerament, es miren els outliers per a cada variable i després de fer una valoració es decideix borrar-los o mantenir-los.

En el case de *Fare* com es pot veure a continuació els outliers, considerant outliers els valors que estan a més de tres desviacions estàndard de la mitja, no són discordants. Per tant, no es veu la necessitat d'esborrar-los.

```
seeOutlierValues(titanic, titanic$Fare)
```

| ##     | Survived | Pclass   | Name  |
|--------|----------|----------|---|
| ## 28  | 0        | 1        | Fortune, Mr. Charles Alexander                        |
| ## 89  | 1        | 1        | Fortune, Miss. Mabel Helen                            |
| ## 119 | 0        | 1        | Baxter, Mr. Quigg Edmond                              |
| ## 259 | 1        | 1        | Ward, Miss. Anna                                      |
| ## 300 | 1        | 1        | Baxter, Mrs. James (Helene DeLaudeniére Chaput)       |
| ## 312 | 1        | 1        | Ryerson, Miss. Emily Borie                            |
| ## 342 | 1        | 1        | Fortune, Miss. Alice Elizabeth                        |
| ## 378 | 0        | 1        | Widener, Mr. Harry Elkins                             |
| ## 381 | 1        | 1        | Bidois, Miss. Rosalie                                 |
| ## 439 | 0        | 1        | Fortune, Mr. Mark                                     |
| ## 528 | 0        | 1        | Farthing, Mr. John                                    |
| ## 558 | 0        | 1        | Robbins, Mr. Victor                                   |
| ## 680 | 1        | 1        | Cardeza, Mr. Thomas Drake Martinez                    |
| ## 690 | 1        | 1        | Madill, Miss. Georgette Alexandra                     |
| ## 701 | 1        | 1        | Astor, Mrs. John Jacob (Madeleine Talmadge Force)     |
| ## 717 | 1        | 1        | Endres, Miss. Caroline Louise                         |
| ## 731 | 1        | 1        | Allen, Miss. Elisabeth Walton                         |
| ## 738 | 1        | 1        | Lesurer, Mr. Gustave J                                |
| ## 743 | 1        | 1        | Ryerson, Miss. Susan Parker "Suzette"                 |
| ## 780 | 1        | 1        | Robert, Mrs. Edward Scott (Elisabeth Walton McMillan) |
| ##     | Sex      | Age      | SibSp Parch Fare                                      |
| ## 28  | male     | 19.00000 | 3 2 263.0000  |
| ## 89  | female   | 23.00000 | 3 2 263.0000  |
| ## 119 | male     | 24.00000 | 0 1 247.5208  |
| ## 259 | female   | 35.00000 | 0 0 512.3292  |
| ## 300 | female   | 50.00000 | 0 1 247.5208  |
| ## 312 | female   | 18.00000 | 2 2 262.3750  |
| ## 342 | female   | 24.00000 | 3 2 263.0000  |
| ## 378 | male     | 27.00000 | 0 2 211.5000  |
| ## 381 | female   | 42.00000 | 0 0 227.5250  |
| ## 439 | male     | 64.00000 | 1 4 263.0000  |
| ## 528 | male     | 29.69912 | 0 0 221.7792  |
| ## 558 | male     | 29.69912 | 0 0 227.5250  |

```
## 680 male 36.00000 0 1 512.3292
## 690 female 15.00000 0 1 211.3375
## 701 female 18.00000 1 0 227.5250
## 717 female 38.00000 0 0 227.5250
## 731 female 29.00000 0 0 211.3375
## 738 male 35.00000 0 0 512.3292
## 743 female 21.00000 2 2 262.3750
## 780 female 43.00000 0 1 211.3375
```

Per les variables *SibSP* i *Parch* es decideix unir les variables, ja que tots fan referència a família a bord del vaixell. Aleshores, amb la variable conjunta es miren els outliers i es considera que tampoc són discordants ja que les famílies de mida més petita són les que tenen algun component que sobrevisqué.

```
titanic$Family_size <- titanic$SibSp + titanic$Parch
seeOutlierValues(titanic, titanic$Family_size)
```

```
##      Survived Pclass
## 14          0      3
## 26          1      3
## 60          0      3
## 69          1      3
## 72          0      3
## 120         0      3
## 160         0      3
## 181         0      3
## 183         0      3
## 202         0      3
## 234         1      3
## 262         1      3
## 325         0      3
## 387         0      3
## 481         0      3
## 542         0      3
## 543         0      3
## 611         0      3
## 679         0      3
## 684         0      3
## 793         0      3
## 814         0      3
## 847         0      3
## 851         0      3
## 864         0      3
##
##                                     Name      Sex
## 14                                Andersson, Mr. Anders Johan  male
## 26  Asplund, Mrs. Carl Oscar (Selma Augusta Emilia Johansson) female
## 60                                Goodwin, Master. William Frederick  male
## 69                                Andersson, Miss. Erna Alexandra female
## 72                                Goodwin, Miss. Lillian Amy female
## 120                               Andersson, Miss. Ellis Anna Maria female
## 160                               Sage, Master. Thomas Henry  male
## 181                               Sage, Miss. Constance Gladys female
## 183                               Asplund, Master. Clarence Gustaf Hugo  male
## 202                               Sage, Mr. Frederick  male
```

```

## 234          Asplund, Miss. Lillian Gertrud female
## 262          Asplund, Master. Edvin Rojj Felix   male
## 325          Sage, Mr. George John Jr    male
## 387          Goodwin, Master. Sidney Leonard   male
## 481          Goodwin, Master. Harold Victor    male
## 542          Andersson, Miss. Ingeborg Constanzia female
## 543          Andersson, Miss. Sigrid Elisabeth female
## 611 Andersson, Mrs. Anders Johan (Alfrida Konstantia Brogren) female
## 679          Goodwin, Mrs. Frederick (Augusta Tyler) female
## 684          Goodwin, Mr. Charles Edward    male
## 793          Sage, Miss. Stella Anna female
## 814          Andersson, Miss. Ebba Iris Alfrida female
## 847          Sage, Mr. Douglas Bullen    male
## 851          Andersson, Master. Sigvard Harald Elias   male
## 864          Sage, Miss. Dorothy Edith "Dolly" female
##          Age SibSp Parch    Fare Family_size
## 14  39.00000    1     5 31.2750         6
## 26  38.00000    1     5 31.3875         6
## 60  11.00000    5     2 46.9000         7
## 69  17.00000    4     2  7.9250         6
## 72  16.00000    5     2 46.9000         7
## 120  2.00000    4     2 31.2750         6
## 160 29.69912    8     2 69.5500        10
## 181 29.69912    8     2 69.5500        10
## 183  9.00000    4     2 31.3875         6
## 202 29.69912    8     2 69.5500        10
## 234  5.00000    4     2 31.3875         6
## 262  3.00000    4     2 31.3875         6
## 325 29.69912    8     2 69.5500        10
## 387  1.00000    5     2 46.9000         7
## 481  9.00000    5     2 46.9000         7
## 542  9.00000    4     2 31.2750         6
## 543 11.00000    4     2 31.2750         6
## 611 39.00000    1     5 31.2750         6
## 679 43.00000    1     6 46.9000         7
## 684 14.00000    5     2 46.9000         7
## 793 29.69912    8     2 69.5500        10
## 814  6.00000    4     2 31.2750         6
## 847 29.69912    8     2 69.5500        10
## 851  4.00000    4     2 31.2750         6
## 864 29.69912    8     2 69.5500        10

```

Descartem els atributs origen:

```

titanic["SibSp"] <- NULL
titanic["Parch"] <- NULL

```

I per últim, a la variable *Age*, si que es decideixen suprimir els outliers ja que precisament la persona més gran és la que sobreviu i això pot comportar a errors d'anàlisi.

```

seeOutlierValues(titanic, titanic$Age)

```

```

##          Survived Pclass          Name Sex Age    Fare

```

```
## 97      0      1      Goldschmidt, Mr. George B male 71.0 34.6542
## 117     0      3      Connors, Mr. Patrick male 70.5  7.7500
## 494     0      1      Artagaveytia, Mr. Ramon male 71.0 49.5042
## 631     1      1  Barkworth, Mr. Algernon Henry Wilson male 80.0 30.0000
## 673     0      2      Mitchell, Mr. Henry Michael male 70.0 10.5000
## 746     0      1      Crosby, Capt. Edward Gifford male 70.0 71.0000
## 852     0      3      Svensson, Mr. Johan male 74.0  7.7750
##      Family_size
## 97      0
## 117     0
## 494     0
## 631     0
## 673     0
## 746     2
## 852     0
```

```
titanic <- removeOutlierValues(titanic, titanic$Age)
```

```
## From titanic$Age 7 skipped tuples
```

### 2.2.2.3 Transformació de les variables

En aquest cas, l'atribut *Name* pot tenir algun valor, ja que en aquesta s'hi pot trobar el títol de la persona. Així, es decideix extreure aquest títol del nom

```
titanic$Title <- as.factor(gsub('(.*, )|(\\.*)', '', titanic$Name))
```

i conservar només el nou atribut *Title* derivat de *Name*.

```
titanic["Name"] <- NULL #La variable Name ja no té cap valor
```

S'unifiquen valors per reduir la grandària del grup

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##      filter, lag

## The following objects are masked from 'package:base':
##
##      intersect, setdiff, setequal, union
```

```
levels(titanic$Title)
```

```
## [1] "Col"      "Don"      "Dr"       "Jonkheer"
## [5] "Lady"     "Major"    "Master"   "Miss"
## [9] "Mlle"     "Mme"      "Mr"       "Mrs"
## [13] "Ms"       "Rev"      "Sir"      "the Countess"
```



```
titles_lookup <- data.frame(Title = c("Capt", "Col", "Don", "Dr", "Jonkheer", "Major", "Rev", "Sir",
                                     "Mr", "Master",
                                     "Lady", "Mlle", "Mme", "Ms", "the Countess",
                                     "Mrs", "Miss"),
                             New.Title = c(rep("Noble male", 8),
                                             "Mr", "Master",
                                             rep("Noble female", 5),
                                             "Mrs", "Miss"),
                             stringsAsFactors = FALSE)
```

S'inclouen en el dataset

```
titanic <- titanic %>%
  left_join(titles_lookup, by = "Title")
```

```
## Warning: Column `Title` joining factor and character vector, coercing into
## character vector
```

```
titanic <- titanic %>%
  mutate(Title = New.Title) %>%
  select(-New.Title)
```

i es visualitzen possibles errors de sexe en el títol

```
titanic %>%
  filter((Sex == "female" & (Title == "Noble male" | Title == "Mr" | Title == "Master") |
         (Sex == "male" & (Title == "Noble female" | Title == "Mrs" | Title == "Miss"))))
```

```
##   Survived Pclass   Sex Age   Fare Family_size   Title
## 1         1       1 female 49 25.9292          0 Noble male
```

Com es pot veure, ha detectat una dona com a *Noble male* i la corregim:

```
titanic <- titanic %>%
  mutate(Title=replace(Title, (Sex == "female" & (Title == "Noble male")), "Noble female"))
```

Per finalitzar, caldrà transformar les variables categòriques en factors per poder tractar-les més fàcilment en l'anàlisi

```
titanic$Survived <- as.factor(titanic$Survived)
titanic$Pclass <- as.factor(titanic$Pclass)
titanic$Title <- as.factor(titanic$Title)
```

### 2.2.3 Exportació de les dades preprocessades

Un cop transformat el dataset s'exporta en un ".csv"

```
write.csv(titanic, "../data/titanic_train_transformed.csv")
```

## 2.3 Anàlisi de les dades

### 2.3.1 Selecció dels grups de dades

Primer de tot, i com que *Survived* és la variable de classe i, com a tal de tipus factor, però per als següents càlculs la farem servir com a referència numèrica, la passarem a tipus *integer*.

```
#Es resta 1 al convertir-lo a integer ja que els valors passen a ser 1 i 2 al transformar-lo
titanic$Survived <- as.integer(titanic$Survived)-1
```

En aquesta secció es preparen els grups dividint-los segons els valors dels diferents atributs i amb la funció *seeGroupStatics* (creada a continuació) es podrà fer una primer anàlisi.

```
seeGroupStatics <- function(resultArray, categoricalArray){
  aggregate(resultArray, list(categoricalArray), FUN = function(x) c(mean = mean(x), count = length(x))
}
```

Una de les agrupacions és a partir de la variable *Pclass* a on podem categoritzar els passatgers segons si van embarcar amb 1a, 2a o 3a classe.

```
levels(titanic$Pclass)
```

```
## [1] "1" "2" "3"
```

```
seeGroupStatics(titanic$Survived, titanic$Pclass)
```

```
##   Group.1      x.mean    x.count
## 1      1  0.6367925 212.0000000
## 2      2  0.4754098 183.0000000
## 3      3  0.2433538 489.0000000
```

```
t_pclass_1 <- titanic %>% filter(Pclass == "1")
t_pclass_2 <- titanic %>% filter(Pclass == "2")
t_pclass_3 <- titanic %>% filter(Pclass == "3")
```

La següent es *Title*

```
levels(titanic$Title)
```

```
## [1] "Master"      "Miss"        "Mr"          "Mrs"
## [5] "Noble female" "Noble male"
```

```
seeGroupStatics(titanic$Survived, titanic$Title)
```

```
##      Group.1      x.mean    x.count
## 1      Master  0.5750000  40.0000000
## 2      Miss   0.6978022 182.0000000
## 3      Mr     0.1565558 511.0000000
## 4      Mrs    0.7920000 125.0000000
## 5 Noble female 1.0000000   7.0000000
## 6 Noble male  0.2631579  19.0000000
```

```
t_title_Master <- titanic %>% filter(Title == "Master")
t_title_Miss <- titanic %>% filter(Title == "Miss")
t_title_Mr <- titanic %>% filter(Title == "Mr")
t_title_Mrs <- titanic %>% filter(Title == "Mrs")
t_title_Noble_female <- titanic %>% filter(Title == "Noble female")
t_title_Noble_male <- titanic %>% filter(Title == "Noble male")
```

Per *Sex*:

```
levels(titanic$Sex)
```

```
## [1] "female" "male"
```

```
seeGroupStatics(titanic$Survived, titanic$Sex)
```

```
##   Group.1      x.mean    x.count
## 1  female    0.7420382 314.0000000
## 2   male    0.1894737 570.0000000
```

```
t_sex_male <- titanic %>% filter(Sex == "male")
t_sex_female <- titanic %>% filter(Sex == "female")
```

A *Age* els agrupem en les categories *Youth*, *Young Adult*, *Adult* i *Senior*, segons si tenen de 0 a 15 anys, de 16 a 35, de 36 a 50 i de 51 a 70 respectivament.

```
max(titanic$Age)
```

```
## [1] 66
```

```
titanic$AgeCategorical<-cut(titanic$Age, seq(0,70,5))
seeGroupStatics(titanic$Survived, titanic$AgeCategorical)
```

```
##   Group.1      x.mean    x.count
## 1  (0,5]    0.7045455  44.0000000
## 2  (5,10]   0.3500000  20.0000000
## 3 (10,15]   0.5789474  19.0000000
## 4 (15,20]   0.3437500  96.0000000
## 5 (20,25]   0.3442623 122.0000000
## 6 (25,30]   0.3298246 285.0000000
## 7 (30,35]   0.4659091  88.0000000
## 8 (35,40]   0.4179104  67.0000000
## 9 (40,45]   0.3617021  47.0000000
## 10 (45,50]  0.4102564  39.0000000
## 11 (50,55]  0.4166667  24.0000000
## 12 (55,60]  0.3888889  18.0000000
## 13 (60,65]  0.2857143  14.0000000
## 14 (65,70]  0.0000000   1.0000000
```

```
titanic$AgeCategorical <- cut(titanic$Age, breaks=c(0, 15, 35, 50, 70), labels=c("Youth", "Young Adult",
seeGroupStatics(titanic$Survived, titanic$AgeCategorical)
```

```
##      Group.1      x.mean      x.count
## 1      Youth    0.5903614  83.0000000
## 2 Young Adult    0.3553299 591.0000000
## 3      Adult    0.3986928 153.0000000
## 4      Senior    0.3684211  57.0000000
```

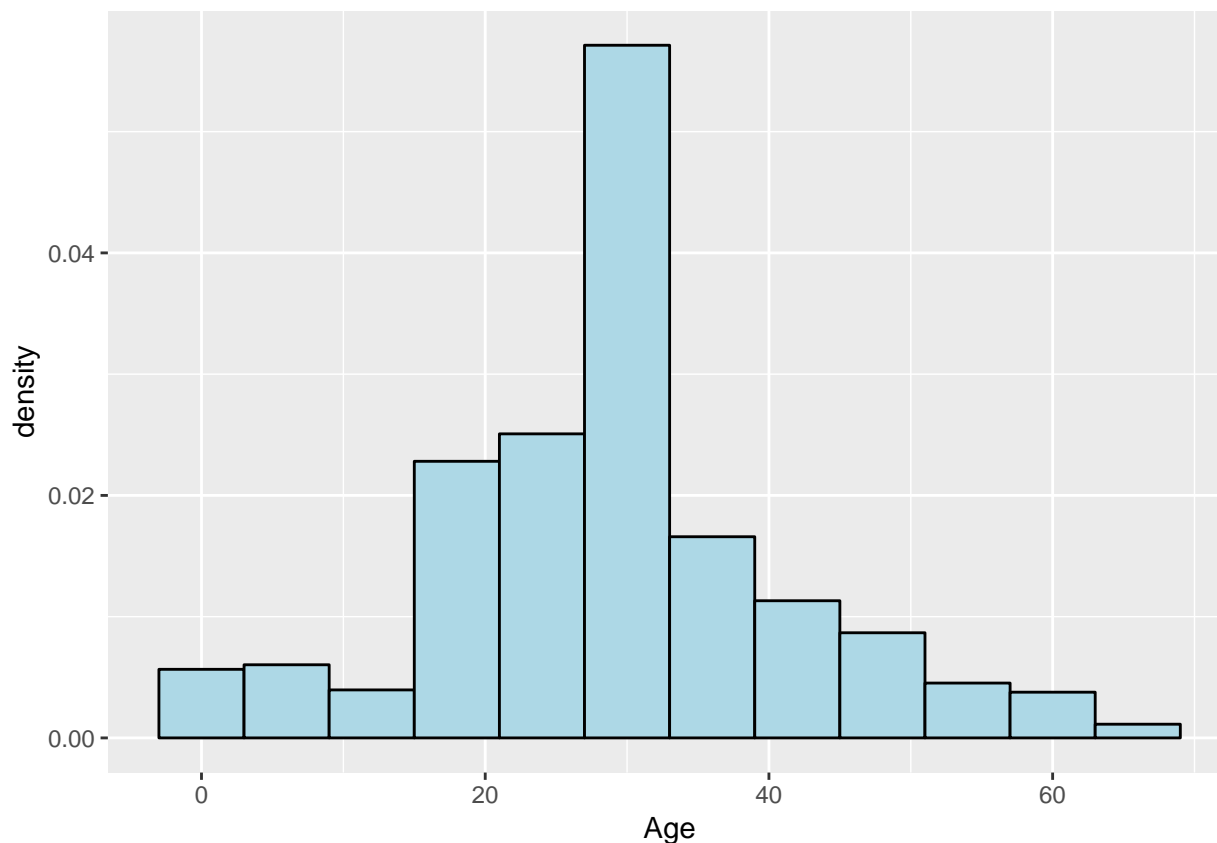
```
t_age_youth <- titanic %>% filter(AgeCategorical == "Youth")
t_age_youngAdult <- titanic %>% filter(AgeCategorical == "Young Adult")
t_age_adult <- titanic %>% filter(AgeCategorical == "Adult")
t_age_senior <- titanic %>% filter(AgeCategorical == "Senior")
```

## 2.3.2 Comprovació de la normalitat i homogeneïtat de la variància

### 2.3.2.1 Normalitat

Comprovar si les dades segueixen una distribució normal es pot realitzar de diverses maneres. Es pot comprovar gràficament si segueix una corba en forma de campana. Es a dir la probabilitat d'obtenir una observació serà més alta al centre de la corba mentres que disminueix a mesura que ens allunyem del mig. Farem servir la llibreria *ggplot2* per visualitzar l'histograma de les variables numèriques. Per exemple, per al cas de la variable *Age*:

```
library(ggplot2)
### See normality of 'Age' by plot
ggplot(titanic, aes(x=Age)) +
  geom_histogram(aes(y=..density..), binwidth = 6, colour="black", fill="lightblue")
```



A simple vista, sembla seguir una distribució normal.

Una altra manera de descriure la seva normalitat és pels paràmetres mitjana i desviació estàndard. Molts algorismes de prova de la normalitat supòsen que les dades s'adapten a la distribució de probabilitat normal mitjançant la desviació estàndard i la mitjana. Aquestes proves, denominades paramètriques, les podem trobar als test *Kolmogorov-Smirnov* o *Shapiro-Wilk*. Assumeixen la hipòtesi nul·la de que les dades es distribueixen normalment. Si el *p-valor* és més baix al nivell de significància (assumirem  $\alpha = 0.05$ ) es rebutja la hipòtesi nul·la i s'assumeix que la població no segueix una distribució normal.

En el cas de *Age*:

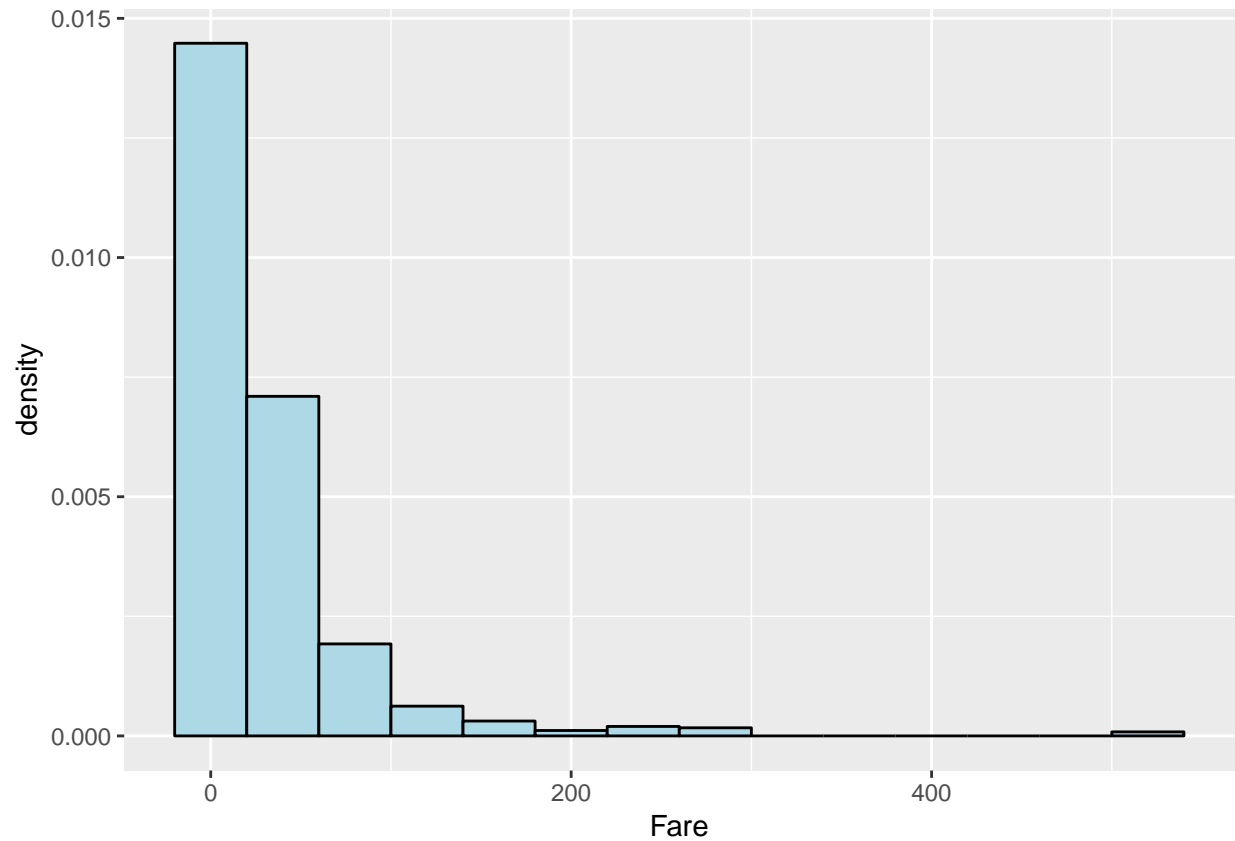
```
shapiro.test(titanic$Age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic$Age
## W = 0.96369, p-value = 5.094e-14
```

Resulta en distribució no normal. El *p-valor* és 5.094e-14, molt inferior al nivell de significància.

Provarem amb la variable *Fare*. Primer gràficament:

```
ggplot(titanic, aes(x=Fare)) +
  geom_histogram(aes(y=..density..), binwidth = 40, colour="black", fill="lightblue")
```



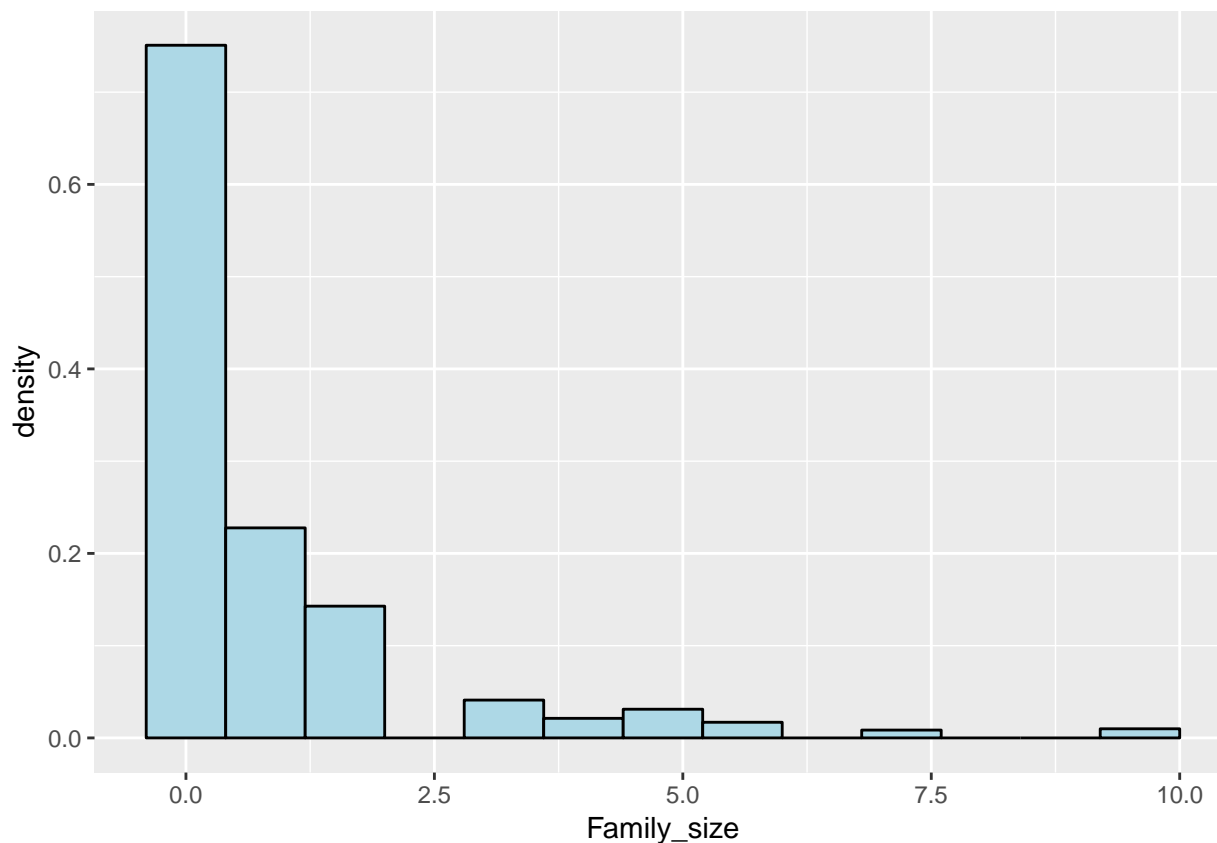
Es tracta d'una distribució no normal de cua a la dreta. Ho comprovem amb *Shapiro-Wilk*:

```
shapiro.test(titanic$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic$Fare
## W = 0.52118, p-value < 2.2e-16
```

Per últim, revisarem l'atribut *Family Size*:

```
ggplot(titanic, aes(x=Family_size)) +
  geom_histogram(aes(y=..density..), binwidth = 0.8, colour="black", fill="lightblue")
```



Tornem a tenir una distribució no normal de cua a la dreta. Revisem-ho amb el test:

```
shapiro.test(titanic$Family_size)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic$Family_size
## W = 0.61634, p-value < 2.2e-16
```

### 2.3.2.2 Homogeneïtat

Bàsicament, la homogeneïtat es tracta de la igualtat de variàncies entre els grups a comparar. Els algorismes poden ser el test de *Levene* si les dades segueixen una distribució normal o bé el test no paramètric de *Fligner-Killeen* en cas de no normalitat en la mostra. Anàlogament al cas dels tests de normalitat, s'assumeix la hipòtesi nul·la amb un nivell de significància  $\alpha = 0.05$  i, si el *p-value* es superior a aquesta, indicarà que les variàncies entre els grups són iguals i, per tant, homògenes.

Degut als resultats de les proves de normalitat, s'utilitzarà el test *Fligner-Killeen*.

Per *Age*:

```
library(car)
```

```
## Warning: package 'car' was built under R version 3.5.3

## Loading required package: carData
```

```
## Warning: package 'carData' was built under R version 3.5.2
```

```
##  
## Attaching package: 'car'
```

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

```
fligner.test(as.integer(Survived) ~ Age, data = titanic)
```

```
##  
## Fligner-Killeen test of homogeneity of variances  
##  
## data: as.integer(Survived) by Age  
## Fligner-Killeen:med chi-squared = 71.281, df = 83, p-value = 0.817
```

Es comprova que el *p-valor* és superior al nivell de significància i, per tant, acceptem la hipòtesi nul·la significant que ambdues mostres són homogènies.

Per a *Fare*:

```
fligner.test(as.integer(Survived) ~ Fare, data = titanic)
```

```
##  
## Fligner-Killeen test of homogeneity of variances  
##  
## data: as.integer(Survived) by Fare  
## Fligner-Killeen:med chi-squared = 255.32, df = 246, p-value =  
## 0.3282
```

També són homogènies.

Per últim, *Family-size*:

```
fligner.test(as.integer(Survived) ~ Family_size, data = titanic)
```

```
##  
## Fligner-Killeen test of homogeneity of variances  
##  
## data: as.integer(Survived) by Family_size  
## Fligner-Killeen:med chi-squared = 26.365, df = 8, p-value =  
## 0.0009094
```

El *p-valor* 0.0009094, menor al nivell de significància, rebutja la hipòtesi nul·la indicant variacions estadísticament diferents (heterogeneïtat).

### 2.3.3 Aplicació de proves estadístiques

**TO-DO:** En funció de les dades i de l'objectiu de l'estudi, aplicar proves de contrast d'hipòtesis, correlacions, regressions, etc. Aplicar almenys tres mètodes d'anàlisi diferents.



### 2.3.3.1 Correlacions

Primerament es volen trobar les variables més rellevants i per a fer-ho es compararan totes les variables del nostre dataset amb *Survived* que és la variable objectiu. Per a fer-ho, s'utilitzarà el Pearson's Chi-squared del qual agafant el valor de *p.value* podrem saber si les variables tenen una correlació si és menor de 0.05. En aquest cas ens interessen les variables més correlacionades amb *Survived* ja que seran les que ens podran explicar millor si aquell individu va o no sobreviure.

```
for (i in colnames(titanic)[2:8]){
  pvalue = chisq.test(titanic[i], titanic$Survived)$p.value
  cat("Pvalue for", i, "vs Survived =", pvalue)
  cat("\n")
}

## Pvalue for Pclass vs Survived = 2.092646e-23
## Pvalue for Sex vs Survived = 3.517646e-58

## Warning in chisq.test(titanic[i], titanic$Survived): Chi-squared
## approximation may be incorrect

## Pvalue for Age vs Survived = 0.03288862

## Warning in chisq.test(titanic[i], titanic$Survived): Chi-squared
## approximation may be incorrect

## Pvalue for Fare vs Survived = 9.710544e-12

## Warning in chisq.test(titanic[i], titanic$Survived): Chi-squared
## approximation may be incorrect

## Pvalue for Family_size vs Survived = 3.703918e-14

## Warning in chisq.test(titanic[i], titanic$Survived): Chi-squared
## approximation may be incorrect

## Pvalue for Title vs Survived = 2.444076e-61
## Pvalue for AgeCategorical vs Survived = 0.0006571085
```

On es pot veure que les variables més correlacionades són *Sex*, *PClass*, i *Title*.

I ara és interessant veure la correlació entre aquestes tres variables

```
ps = chisq.test(titanic$Pclass, titanic$Sex)$p.value
pt = chisq.test(titanic$Pclass, titanic$Title)$p.value

## Warning in chisq.test(titanic$Pclass, titanic$Title): Chi-squared
## approximation may be incorrect

st = chisq.test(titanic$Sex, titanic$Title)$p.value

## Warning in chisq.test(titanic$Sex, titanic$Title): Chi-squared
## approximation may be incorrect
```

```
cormatrix = matrix(c(1, ps, pt,
                     ps, 1, st,
                     pt, st, 1),
                   3, 3, byrow = TRUE)

row.names(cormatrix) = colnames(cormatrix) = c("Pclass", "Sex", "Title")
cormatrix
```

```
##           Pclass           Sex           Title
## Pclass 1.000000e+00 1.260108e-04 6.989522e-13
## Sex    1.260108e-04 1.000000e+00 7.723412e-189
## Title  6.989522e-13 7.723412e-189 1.000000e+00
```

Com era d'esperar *Title* està forçament correlacionada amb *Sex* i *Pclass* ja que el títol és molt semblant a la classe i s'ha extès amb el sexe. Així, les variables més valuoses seran *Sex* i *Pclass*.

### 2.3.3.2 Regressió

Donat que en la secció anterior hem trobat les variables que poden ser més valuoses es decideix fer una regressió combinant aquestes dues variables. Cal a dir, que després de provar més d'una combinació, aquesta és la més lògica amb un valor de  $r^2$  més elevat.

```
sex_Pclass_lm <- lm(Survived~Sex*Pclass, data=titanic)
cat("El valor de r2 de la regressió és:", summary(sex_Pclass_lm)$r.squared)
```

```
## El valor de r2 de la regressió és: 0.3940623
```

```
cat("\n")
```

```
coef(sex_Pclass_lm)
```

```
##      (Intercept)      Sexmale      Pclass2      Pclass3
##      0.96808511    -0.59520375    -0.04703247    -0.46808511
## Sexmale:Pclass2 Sexmale:Pclass3
##      -0.16697038      0.23143563
```

On es pot veure que el valor de  $r^2$  no és gaire alt, però és suficient per poder extreure algunes conclusions.

Els homes tenen bastantes menys probabilitats de viure que les dones i malgrat que la classe 2 és la que té menys probabilitat de sobreviure.

En la classe 3 també es difícil sobreviure, tot i que pels homes no hi ha tanta diferència de trobar-se en la classe 2 o 3 com en les dones.

## 2.4 Representació dels resultats

TO-DO: Taules i gràfiques

## 2.5 Resolució del problema

TO-DO: A partir dels resultats obtinguts, quines són les conclusions? Els resultats permeten respondre al problema?