Pràctica 2: Neteja i anàlisi de les dades

Jorgina Arrés Cardona, desembre 2020

Enllaç a GitHub: https://github.com/jorginaarres1/TitanicPractica2

Continguts

---

1. Descripció del dataset.

Per dur a terme aquesta pràctica utilitzarem el dataset de Kaggle que podem trobar en aquesta url:

https://www.kaggle.com/c/titanic/overview

En aquest repte es demana crear un model predictiu que respongui a la pregunta següent:

"Quin tipus de gent va tenir mes probabilitats de sobreviure?"

Tal com es descriu en aquesta competició, tot i que la majoria de passatgers del Titanic no van sobreviure, n'hi havia que tenien més i menys possibilitats de supervivència, i volem saber quines són les característiques d'aquests passatgers per tenir més oportunitat.

Per respondre aquesta pregunta, començarem per descriure les dades de les que disposem.

Disposem de tres datasets, el primer, gender_submission.csv, està compost per 2 característiques (columnes) que representen 1309 passatgers (files o registres). A continuació podem veure una part del dataset:

```
dsGenderSubmission <- read.csv("./titanic/gender_submission.csv",  sep =
',' , header=TRUE)
head(dsGenderSubmission)
```

```
##   PassengerId Survived
## 1         892        0
## 2         893        1
## 3         894        0
## 4         895        0
## 5         896        1
## 6         897        0
```

Els camps que observem són: - PassengerId: identificador numèric del passatger - Survived: el passatger ha sobreviscut (1), o no ha sobreviscut(0)

Els següents datasets dels que disposem son test.csv i train.csv. El primer disposa de 11 característiques i el segon de 12.El dataset de test té 1309 registres i el de train té 891 registres. A continuació podem veure una part del dataset test.csv:

```
dsTest <- read.csv("./titanic/test.csv",  sep = ',' , header=TRUE)
head(dsTest)
```

```
##   PassengerId Pclass                                          Name
Sex  Age
## 1         892      3                              Kelly, Mr. James
male 34.5
## 2         893      3              Wilkes, Mrs. James (Ellen Needs)
female 47.0
## 3         894      2                     Myles, Mr. Thomas Francis
male 62.0
## 4         895      3                              Wirz, Mr. Albert
male 27.0
## 5         896      3 Hirvonen, Mrs. Alexander (Helga E Lindqvist)
female 22.0
```

```
## 6         897      3                    Svensson, Mr. Johan Cervin
male 14.0
##   SibSp Parch  Ticket    Fare Cabin Embarked
## 1     0     0  330911  7.8292              Q
## 2     1     0  363272  7.0000              S
## 3     0     0  240276  9.6875              Q
## 4     0     0  315154  8.6625              S
## 5     1     1 3101298 12.2875              S
## 6     0     0    7538  9.2250              S
```

Els camps que observem al dataset de Test són:

- PassengerId: identificador numèric del passatger
- Pclass: classe del bitllet comprat, 1ra classe(1), 2na(2) o 3ra(3)
- Name: Cognom i nom del passatger
- Sex: Sexe, home o dona (male o female)
- Age: Edat
- SibSp: Nombre de germans / cònjuges a bord del Titanic
- Parch: Nombre de pares / fills a bord del Titanic
- Ticket: Identificador del bitllet
- Fare: Tarifa, preu del bitllet
- Cabin: Identificador de cabina
- Embarked: Port on han embarcat (C = Cherbourg, Q = Queenstown, S = Southampton)

I una part del dataset train.csv:

```
dsTrain <- read.csv("./titanic/train.csv",  sep = ',' , header=TRUE)
head(dsTrain)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                  Name    Sex Age SibSp
Parch
## 1                             Braund, Mr. Owen Harris   male  22     1
0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
0
## 3                              Heikkinen, Miss. Laina female  26     0
0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1
0
## 5                            Allen, Mr. William Henry   male  35     0
0
```

```
## 6                                    Moran, Mr. James   male  NA     0
0
##             Ticket    Fare Cabin Embarked
## 1        A/5 21171  7.2500              S
## 2         PC 17599 71.2833   C85        C
## 3 STON/O2. 3101282  7.9250              S
## 4           113803 53.1000  C123        S
## 5           373450  8.0500              S
## 6           330877  8.4583              Q
```

Els camps que treobem en el dataset test son els mateixos que en test, afegint el camp:
- Survived: el passatger ha sobreviscut (1), o no ha sobreviscut(0)

---

2. Integració i selecció de les dades d'interès a analitzar.

A partir d'aquestes dades, podem plantejar-nos diferents preguntes, hi ha mes possibilitats de sobreviure depenent de si el bitllet es d'una classe o una altra? o si ets home o dona o més gran o més petit de x edat? Depèn del port on hàgis embarcat?

Primer identifiquem les dades que NO ens interessen, ja que considerem que no es tracta d'informació important per a resoldre el nostre problema:

- (Cabin) el numero de cabina, que per general és una dada buida, la podem obviar, no l'estudiarem donat que no ens donarà resultats que considerem importants ni acotats.
- (Ticket) L'identificador del bitllet tampoc ens interessa, ja que estudiarem el nostre model depenent de l'edat, sexe, port d'embarcament i en quina classe viatjava el passatger.
- (Name) El nom del passatger tampoc és una dada rellevant pel model.

SibSp, Parch i Fare de moment els mantindrem, perquè podrien o no ser importants, ho decidirem més endavant.

Les dades que inicialment ens interessaran més són: Pclass, Sex, Age i Embarked.

---

3. Neteja de les dades.

Després de seleccionar les dades del dataset que ens potencialment ens poden interessar, mirarem com tractem els elements buits i els elements extrems en els següents apartats:

3.1 Gestió de zeros o elements buits.

Tot i que moltes vegades s'utilitza el 0 per indicar la manca de dades (centinella), en els datasets de train i de test trobem tots els elements que no sabem de la forma: NA.

```
sapply(dsTrain, function(x) sum(is.na(x)))
```

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

```
sapply(dsTest, function(x) sum(is.na(x)))
```

```
## PassengerId      Pclass        Name         Sex         Age
SibSp
##           0           0           0           0          86
0
##       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           1           0           0
```

Tant test com train tenen elements buits en la variable Age. I per tant haurem de decidir com tractem aquests elements, podriem eliminar aquestes entrades però també perdríem dades que ens poden ser de valor. En conclusió, utilitzarem kNN-imputació que ens ajudarà a posar valors a aquests registres buits basats en k veins més propers.

Si ens fixem en detall, en el dataset de test, Fare també te un element buit, el tractarem de la mateixa manera que Age a continuació:

```
# Imputació de valors amb kNN() del paquet VIM
#install.packages('VIM')
suppressWarnings(suppressMessages(library(VIM)))
dsTrain$Age <- kNN(dsTrain)$Age

dsTest$Age <- kNN(dsTest)$Age
dsTest$Fare <- kNN(dsTest)$Fare
```

I tornem a fer la comprovació inicial:

```
sapply(dsTrain, function(x) sum(is.na(x)))
```

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

```
Embarked
##           0           0           0           0           0
0

sapply(dsTest, function(x) sum(is.na(x)))

## PassengerId      Pclass        Name         Sex         Age
SibSp
##           0           0           0           0           0
0
##       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           0           0
```

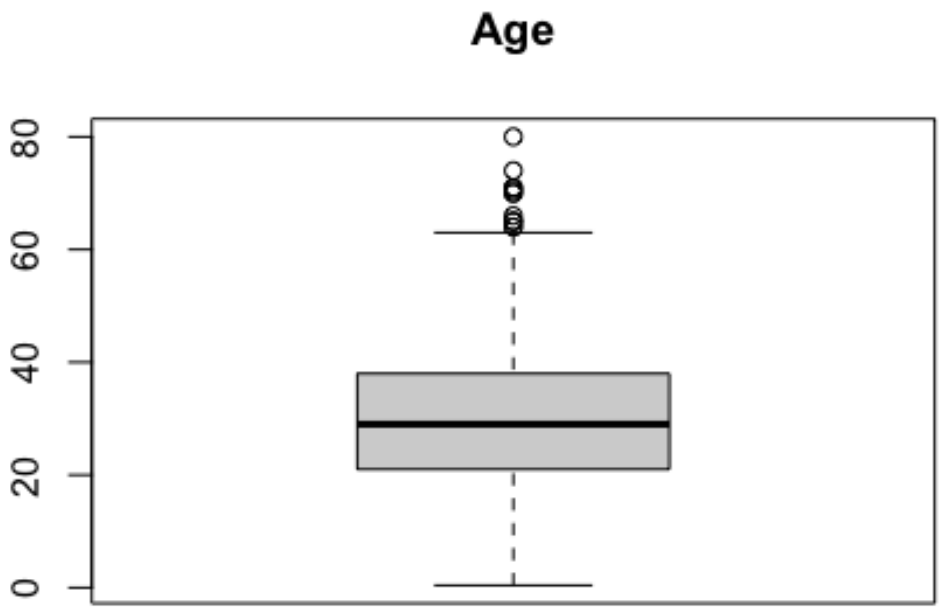
Veiem que ja no tenim elements buits.

3.2 Identificació i tractament de valors extrems.

De les variables numèriques que no són identificadors podem obtenir els valors extrems de diferents maneres, la primera seria a partir del seu boxplot i la segona partir d'aquesta funció boxplots.stats() de R. Primer mirarem les variables pel dataset dsTrain i després pel dsTest:

dsTrain

```
boxplot(dsTrain$Age, main = 'Age')
```



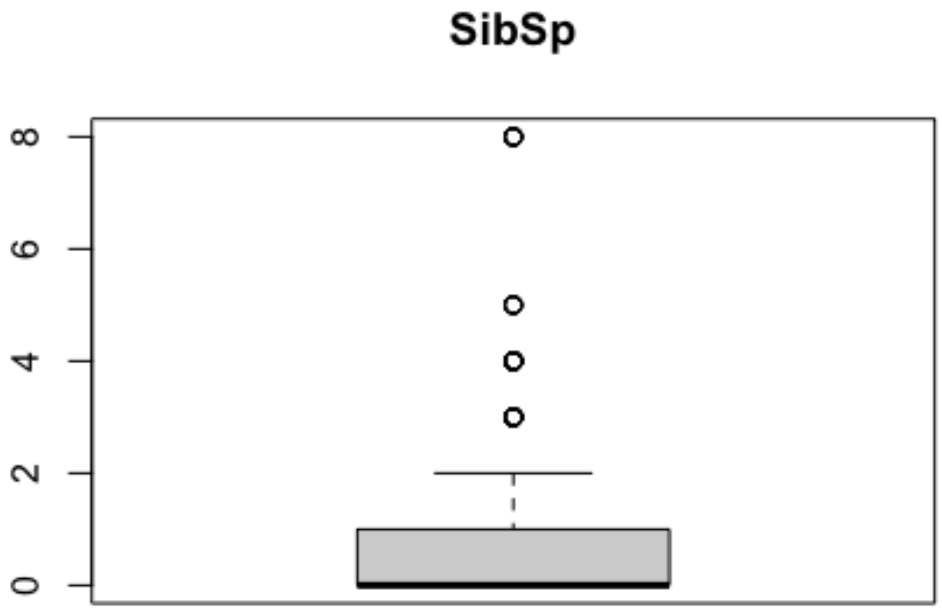
Per la variable age veiem que detectem elements fora dels bigotis, per tant trobem valors extrems que hem de tractar:

```
boxplot.stats(dsTrain$Age)$out
```

```
##  [1] 66.0 65.0 71.0 70.5 65.0 64.0 65.0 71.0 64.0 80.0 70.0 70.0 74.0
```

En aquest cas de la variable Age detectem que la majoria dels passatgers ronden entre els 20 i 40 anys i que en hi ha pocs entre 60 i 80, per això ens surten aquests valors extrems, però com ens interessa estudiar-los el tractament que en farem serà deixar-los igual que estan.

```
boxplot(dsTrain$SibSp, main = 'SibSp')
```




```
boxplot.stats(dsTrain$SibSp)$out
```

```
##  [1] 3 4 3 3 4 5 3 4 5 3 3 4 8 4 4 3 8 4 8 3 4 4 4 4 8 3 3 5 3 5 3 4 4
3 3 5 4 3
## [39] 4 8 4 3 4 8 4 8
```

Podem veure que el valor més extrem es troba al 8, aquest nombre si que el tractarem, posant-lo primer com a NA i seguidament amb el valor de la mitjana dels nombres de SibSp.

```
dsTrain$SibSp[ dsTrain$SibSp>7 ] <- NA
boxplot.stats(dsTrain$SibSp)$out
```

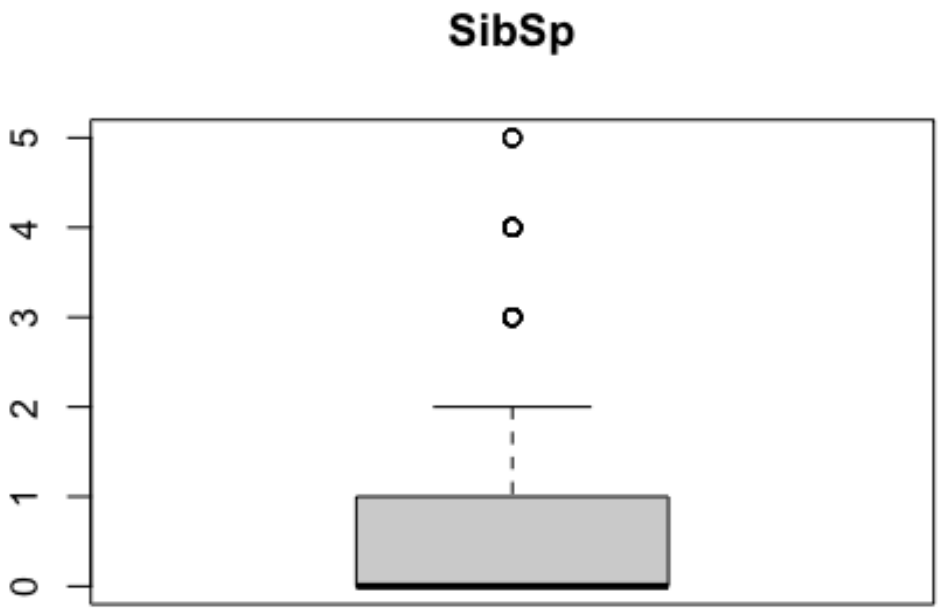
```
##  [1] 3 4 3 3 4 5 3 4 5 3 3 4 4 4 3 4 3 4 4 4 4 3 3 5 3 5 3 4 4 3 3 5 4
3 4 4 3 4
## [39] 4
```

```
idx <- which(is.na(dsTrain$SibSp))
length(idx) #nombre de valors perduts
```

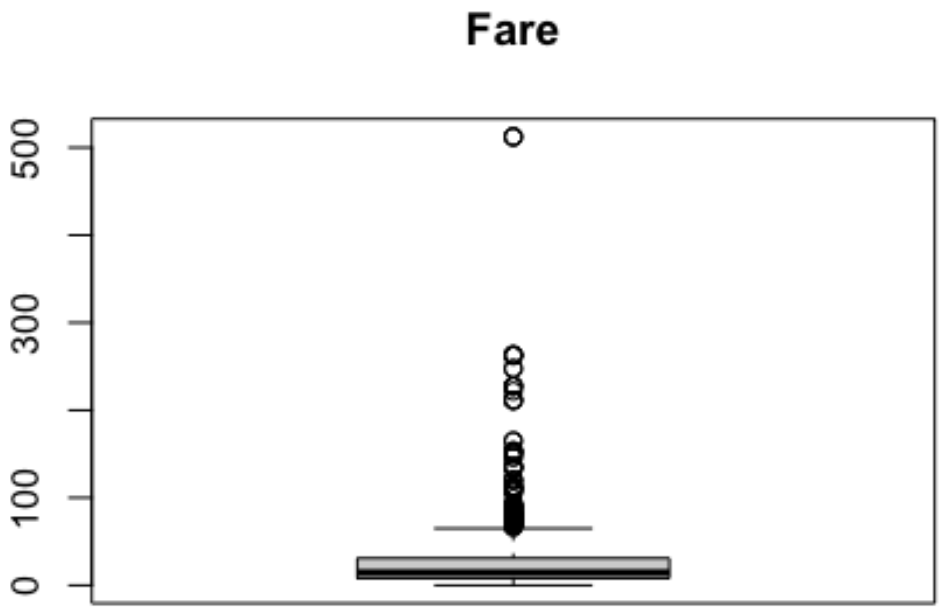
```
## [1] 7
```

```
for (i in 1:length(idx)){
index <- idx[i]
SibSp <- dsTrain[index,]$SibSp
```

```
dsTrain[index,]$SibSp <- median( dsTrain$SibSp, na.rm=TRUE ) #imputació
}
dsTrain$SibSp[idx] #mostrem resultat

## [1] 0 0 0 0 0 0 0

boxplot(dsTrain$SibSp, main = 'SibSp')
```

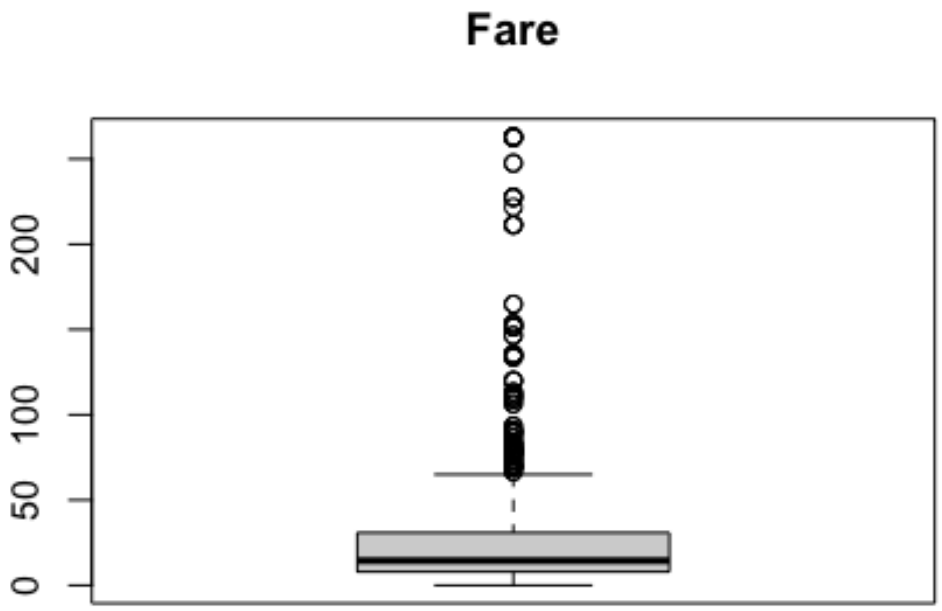



```
boxplot(dsTrain$Fare, main = 'Fare')
```



Veiem un element especialment extrem, per sobre del valor 400 en la variable fare, aquest el tractarem a continuació:

```
dsTrain$Fare[ dsTrain$Fare>400 ] <- NA
boxplot(dsTrain$Fare, main = 'Fare')
```



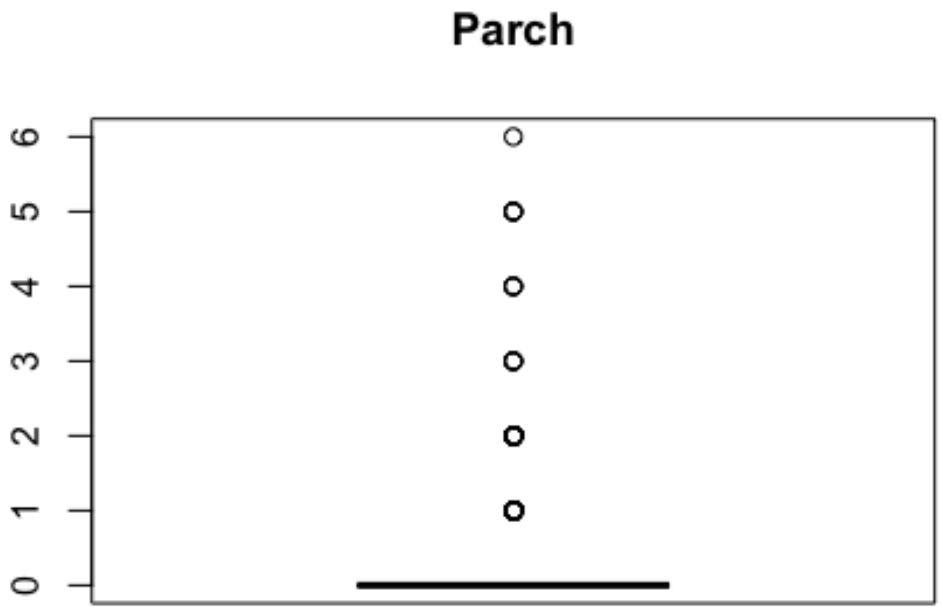

```
idx <- which(is.na(dsTrain$Fare))
length(idx) #nombre de valors perduts
```

```
## [1] 3
```

```
for (i in 1:length(idx)){
index <- idx[i]
Fare <- dsTrain[index,]$Fare
dsTrain[index,]$Fare <- median( dsTrain$Fare, na.rm=TRUE ) #imputació
}
dsTrain$Fare[idx] #mostrem resultat
```

```
## [1] 14.4542 14.4542 14.4542
```

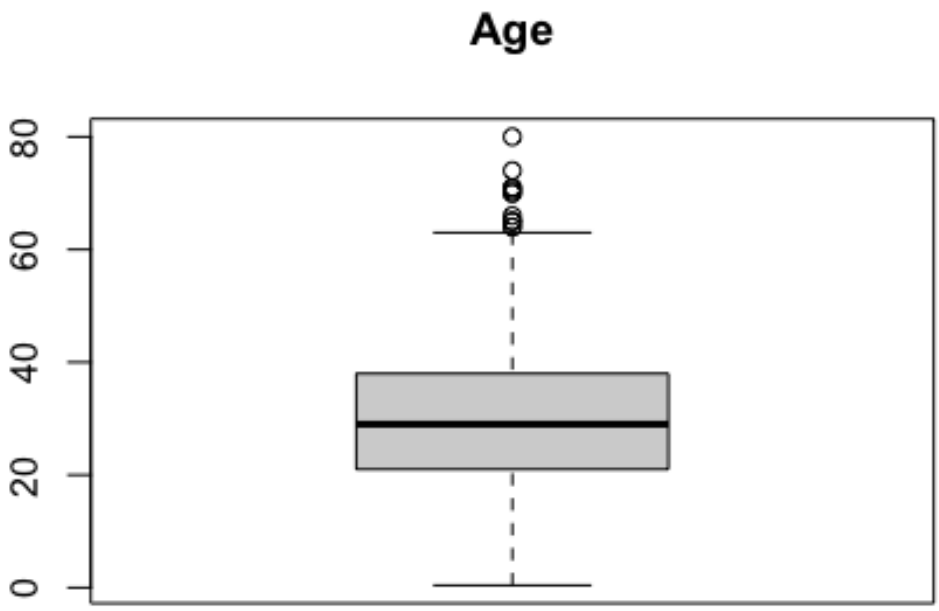
```
boxplot(dsTrain$Parch, main = 'Parch')
```



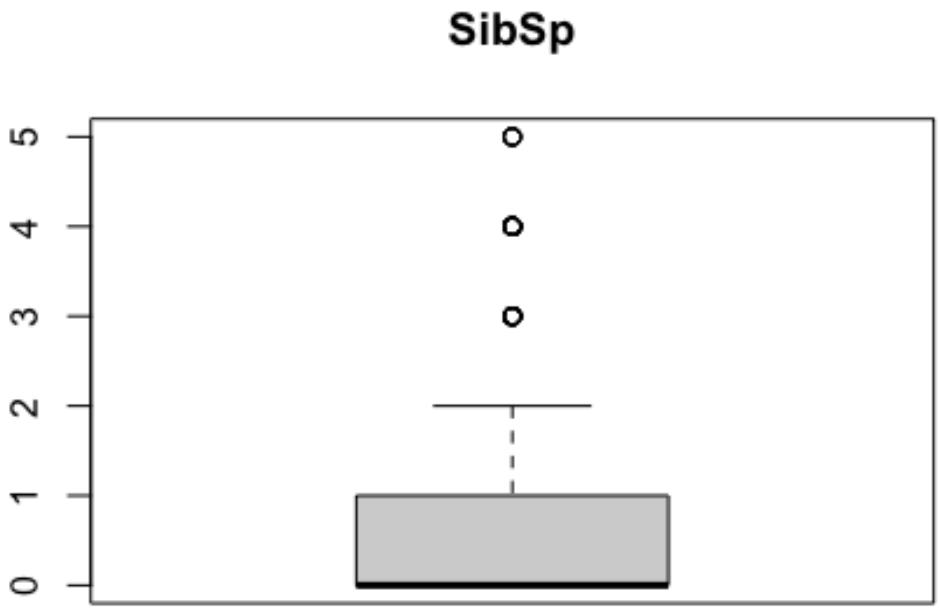
Les variables que hem analitzat presenten totes valors extrems, o que no son els més habituals, tant Age, SibSp, Fare com Parch presenten aquests valors, el noste tractament serà deixar les variables tal com estàn perquè ens interessen de cara al model tenirlos, no volem estudiar només els passatgers de 20-40 anys sinó tots els possibles, el mateix per les altres variables tot i que es donin pocs casos. Si que hem anat tractant els casos més extrems, però no tots, com hem anat explicant. D'aquesta manera tenim una mostra més gran però els casos molt raros no els tenim en compte.

dsTest

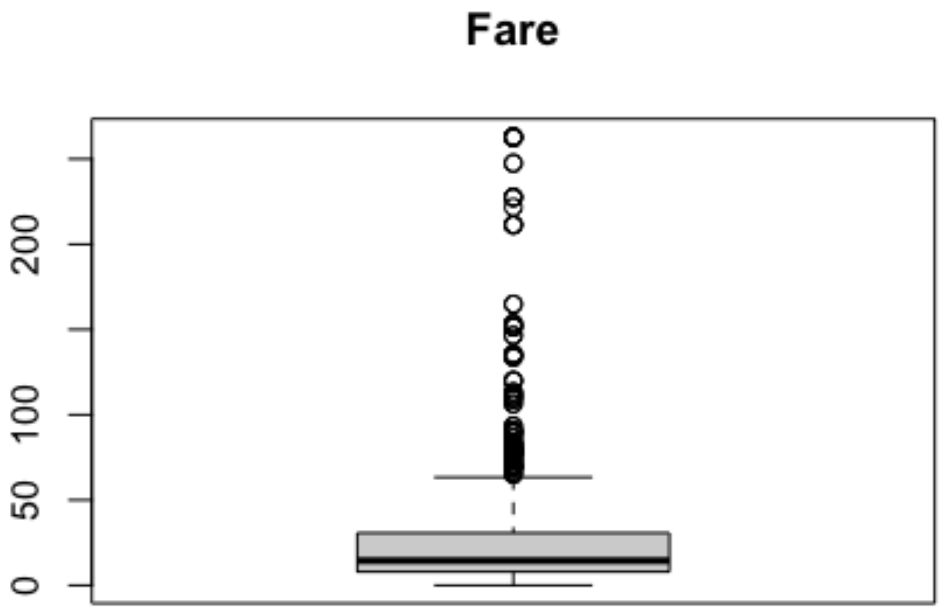
```
boxplot(dsTrain$Age, main = 'Age')
```



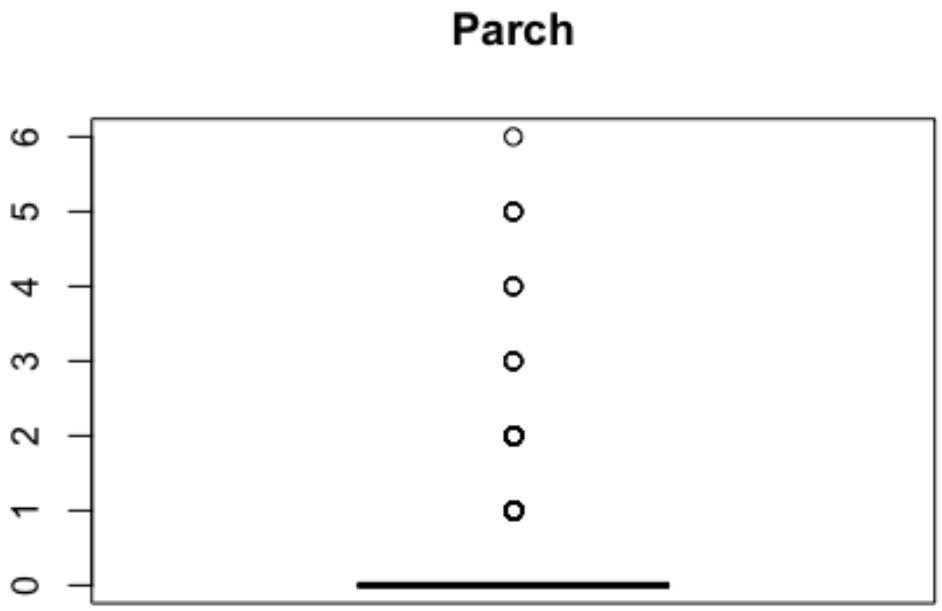

```r
boxplot(dsTrain$SibSp, main = 'SibSp')
```




```r
boxplot(dsTrain$Fare, main = 'Fare')
```




```r
boxplot(dsTrain$Parch, main = 'Parch')
```




```
boxplot.stats(dsTrain$Age)$out
```

```
##  [1] 66.0 65.0 71.0 70.5 65.0 64.0 65.0 71.0 64.0 80.0 70.0 70.0 74.0
```

```
boxplot.stats(dsTrain$SibSp)$out
```

```
##  [1] 3 4 3 3 4 5 3 4 5 3 3 4 4 4 3 4 3 4 4 4 4 3 3 5 3 5 3 4 4 3 3 5 4
3 4 4 3 4
## [39] 4
```

```
boxplot.stats(dsTrain$Fare)$out
```

```
##   [1]  71.2833 263.0000 146.5208  82.1708  76.7292  80.0000  83.4750
73.5000
##   [9] 263.0000  77.2875 247.5208  73.5000  77.2875  79.2000  66.6000
69.5500
##  [17]  69.5500 146.5208  69.5500 113.2750  76.2917  90.0000  83.4750
90.0000
##  [25]  79.2000  86.5000  79.6500 153.4625 135.6333  77.9583  78.8500
91.0792
##  [33] 151.5500 247.5208 151.5500 110.8833 108.9000  83.1583 262.3750
164.8667
##  [41] 134.5000  69.5500 135.6333 153.4625 133.6500  66.6000 134.5000
263.0000
```

```
##  [49]  75.2500  69.3000 135.6333  82.1708 211.5000 227.5250  73.5000
120.0000
##  [57] 113.2750  90.0000 120.0000 263.0000  81.8583  89.1042  91.0792
90.0000
##  [65]  78.2667 151.5500  86.5000 108.9000  93.5000 221.7792 106.4250
71.0000
##  [73] 106.4250 110.8833 227.5250  79.6500 110.8833  79.6500  79.2000
78.2667
##  [81] 153.4625  65.0000  77.9583  69.3000  76.7292  73.5000 113.2750
133.6500
##  [89]  73.5000  76.7292 211.3375 110.8833 227.5250 151.5500 227.5250
211.3375
##  [97]  78.8500 262.3750  71.0000  65.0000  86.5000 120.0000  77.9583
211.3375
## [105]  79.2000  69.5500 120.0000  93.5000  80.0000  83.1583  69.5500
89.1042
## [113] 164.8667  69.5500  83.1583
```

```
boxplot.stats(dsTrain$Parch)$out
```

```
##   [1] 1 2 1 5 1 1 5 2 2 1 1 2 2 2 1 2 2 2 3 2 2 1 1 1 1 2 1 1 2 2 1 2
2 2 1 2 1
##  [38] 1 2 1 4 1 1 1 1 2 2 1 2 1 1 1 2 1 1 2 2 2 1 1 2 2 1 2 1 1 1 1 1
1 1 2 1 2
##  [75] 2 1 1 2 1 1 2 1 1 1 1 2 1 1 1 4 1 1 2 2 2 2 2 1 1 1 2 2 1 1 2 2
3 4 1 2 1
## [112] 1 2 1 2 1 2 1 1 2 2 1 1 1 1 2 2 2 2 2 2 1 1 2 1 4 1 1 2 1 2 1 1
2 5 2 1 1
## [149] 1 2 1 5 2 1 1 1 2 1 6 1 2 1 2 1 1 1 1 1 1 1 3 2 1 1 1 1 2 1 2 3
1 2 1 2 2
## [186] 1 1 2 1 2 1 2 1 1 1 2 1 1 2 1 2 1 1 1 1 3 2 1 1 1 1 5 2
```

Passa exactament el mateix que en el dataset Train , i ho tractarem de la mateixa manera, però no hi ha valors extrems que destaquin sobre la resta, així que aquest dataset el tractarem deixant-lo així.

---

4. Anàlisi de les dades.

4.1 Selecció dels grups de dades.

A continuació, seleccionem els grups de dades que ens poden resultar interessants per l'estudi. Més endavant elegirem quins utilitzem i quins no.

```
# Agrupació per tipus de tarifa, Pclass 1ra 2a 3ra
dsTrain.Pclass1 <- dsTrain[dsTrain$Pclass == 1,]
dsTrain.Pclass2 <- dsTrain[dsTrain$Pclass == 2,]
```

```r
dsTrain.Pclass3 <- dsTrain[dsTrain$Pclass == 3,]

# Agrupació por sexe
dsTrain.female <- dsTrain[dsTrain$Sex == "female",]
dsTrain.male <- dsTrain[dsTrain$Sex == "male",]

# Agrupació per port d'embarcament
dsTrain.Cherbourg <- dsTrain[dsTrain$Embarked == "C",]
dsTrain.Queenstown <- dsTrain[dsTrain$Embarked == "Q",]
dsTrain.Southampton <- dsTrain[dsTrain$Embarked == "S",]

# Agrupació per edats
dsTrain.adult <- dsTrain[dsTrain$Age > 30 ,]
dsTrain.young <- dsTrain[dsTrain$Age <= 30,]
```

4.2 Comprovació de la normalitat i homogeneïtat de la variància.

Per comprobar que els valors de les nostres variables quantitatives provenen d'una població amb distribució normal, s'utilitza la proba d'Anderson-Darling.

```r
library(nortest)
alpha = 0.05
col.names = colnames(dsTrain)

for (i in 1:ncol(dsTrain)) {
  if (i == 1) cat("Variables que no seguieixen distribució normal:\n")
  if (is.integer(dsTrain[,i]) | is.numeric(dsTrain[,i])) {
    p_val = ad.test(dsTrain[,i])$p.value
    if (p_val < alpha) {
      cat(col.names[i])
      cat("\n")
    }
  }
}
```

```
## Variables que no seguieixen distribució normal:
## PassengerId
## Survived
## Pclass
## Age
## SibSp
## Parch
## Fare
```

Si s'obté un p-valor superior al nivell de signidicança prefixat = 0.05, llavors es considera que la variable segueix una distribució normal. Totes les variables mencionades a la resposta no segueixen aquesta distribució normal.

A continuació, volem estudiar la homogeneitat de variàncies mitjançant un test Fligner-Killeen. Estudiarem el test pel sexe. La hipòtesis és que les dues variàncies son iguals,

```
fligner.test(Survived ~ Sex, data = dsTrain)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Survived by Sex
## Fligner-Killeen:med chi-squared = 5.7729, df = 1, p-value = 0.01627
```

Donat que obtenum un p-valor inferior a 0.05 rebutgem la hipòtesis que les variances de les dos mostres son homogènies.

4.3 Aplicació de proves estadístiques.

4.3.1 Quines variables quantitatives influeixen més en la supervivència?

Farem un anàlisis de correlació de diferents variables per descobrir quines d'elles influencia sobre la supervivència dels passatgers. Com les nostres dades no segueixen una distribució normal, farem servir el coeficient de correlació de Spearman.

```
corr_matrix <- matrix(nc = 2, nr = 0)
colnames(corr_matrix) <- c("estimate", "p-value")
# Calcul del coeficient de correlació per cada variable quantitativa
# respecto al campo Survived
for (i in 1:(ncol(dsTrain))) {
  if ((is.integer(dsTrain[,i]) | is.numeric(dsTrain[,i]) )& !i==2) {
    spearman_test = cor.test(dsTrain[,i],
    dsTrain[,2],
    method = "spearman", exact=FALSE)
    corr_coef = spearman_test$estimate
    p_val = spearman_test$p.value
    # Add row to matrix
    pair = matrix(ncol = 2, nrow = 1)
    pair[1][1] = corr_coef
    pair[2][1] = p_val
    corr_matrix <- rbind(corr_matrix, pair)
    rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(dsTrain)[i]
  }
}

print(corr_matrix)
```

```
##                 estimate      p-value
## PassengerId -0.005006661 8.813658e-01
## Pclass      -0.339667937 1.687608e-25
## Age         -0.098905533 3.122665e-03
## SibSp        0.107973503 1.247092e-03
```

```
## Parch        0.138265633 3.453591e-05
## Fare         0.317893819 2.264158e-22
```

La interpretació del coeficient de Spearman concorda en valors pròxims a 1, que indiquen una correlació forta i positiva(en el nostre cas la correlació positiva més forta és amb Fare). Valors pròxims a -1 indiquen una correlació forta i negativa(en el nostre cas, passa amb Pclass). Valors propers a 0 indiquen que no hi ha correlació lineal, però podria existir algun altre tipus de correlació.

En resum, quant més Fare, preu, més supervivència. La qual cosa quadra ja que més preu és classe 1 i menys és classe 3. I hem vist que com més petit és el numero de Pclass, més supervivència també. I el Pclass més petit és primera classe, que es la tarifa més cara.

4.3.2 La supervivència és més probable si el passatger és dona?

La segona proba és un contrast d'hipòtesi sobre dos mostres, per determinar si la supervivència de les dones és superior a la dels homes Utilitzarem dues mostres, la supervivència de les dones i la supervivència dels homes

Per a la realització del test necessitem dades amb distribució normal. En aquest cas n>30 i per tant el contrast d'hipòtesis es pot fer:

```
dsTrain.female.survived <-
dsTrain[dsTrain$Sex == "female",]$Survived
dsTrain.male.survived <-
dsTrain[dsTrain$Sex == "male",]$Survived
```

Contrast d'hipòtesis de dos mostres sobre la diferència:

H0 : μ1-μ2 = 0 H1 : μ1-μ2 < 0

μ1 és la mitjana de la població de la que se s'extreu la primera mostra i μ2 és la mitjana de la població de la que s'extreu la segona. Alfa = 0, 05.

```
t.test(dsTrain.female.survived, dsTrain.male.survived,
alternative = "less")

##
##  Welch Two Sample t-test
##
## data:  dsTrain.female.survived and dsTrain.male.survived
## t = 18.672, df = 584.43, p-value = 1
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf 0.6019342
## sample estimates:
## mean of x mean of y
## 0.7420382 0.1889081
```

p-value és més gran que alfa, no rebutjem la hipòtesi nul·la, i per tant concloem que la supervivència és major si el passatger és dona que si és home.

4.3.3 Model de regressió lineal

```
modelo<-lm(formula = Survived ~ Sex + Age +  Pclass + Fare + Embarked +
Parch+ SibSp , data = dsTrain )
summary(modelo)
```

```
##
## Call:
## lm(formula = Survived ~ Sex + Age + Pclass + Fare + Embarked +
##     Parch + SibSp, data = dsTrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.0715 -0.2160 -0.0716  0.2245  1.0339
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.6174597  0.2772368   5.834 7.59e-09 ***
## Sexmale     -0.4881547  0.0283588 -17.214  < 2e-16 ***
## Age         -0.0075376  0.0010556  -7.140 1.95e-12 ***
## Pclass      -0.2126153  0.0215257  -9.877  < 2e-16 ***
## Fare        -0.0003495  0.0004212  -0.830  0.40679
## EmbarkedC   -0.1116371  0.2687021  -0.415  0.67790
## EmbarkedQ   -0.1464752  0.2717319  -0.539  0.58999
## EmbarkedS   -0.1833816  0.2682173  -0.684  0.49434
## Parch       -0.0195059  0.0180044  -1.083  0.27893
## SibSp       -0.0429354  0.0161924  -2.652  0.00816 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3759 on 881 degrees of freedom
## Multiple R-squared:  0.4091, Adjusted R-squared:  0.4031
## F-statistic: 67.78 on 9 and 881 DF,  p-value: < 2.2e-16
```

El coeficient R^2 ajustat és de 0,4031, no és molt bo però anirem a comparar els resultats amb el dataset de test i el que prediu el nostre model:

```
pred<-predict(modelo,dsTest, type= "response")
pred[pred<0.5]<- 0
pred[pred>=0.5]<-1
solution <- data.frame(real = dsGenderSubmission$Survived,
                       predicted = pred
                       )
colnames(solution)<- c("Real","Predicció" )
head(solution)
```

```
##   Real Predicció
## 1    0         0
```

```
## 2    1         0
## 3    0         0
## 4    0         0
## 5    1         1
## 6    0         0
```

Anem a veure la taxa d'errror de la nostra predicció respecte al nostre model:

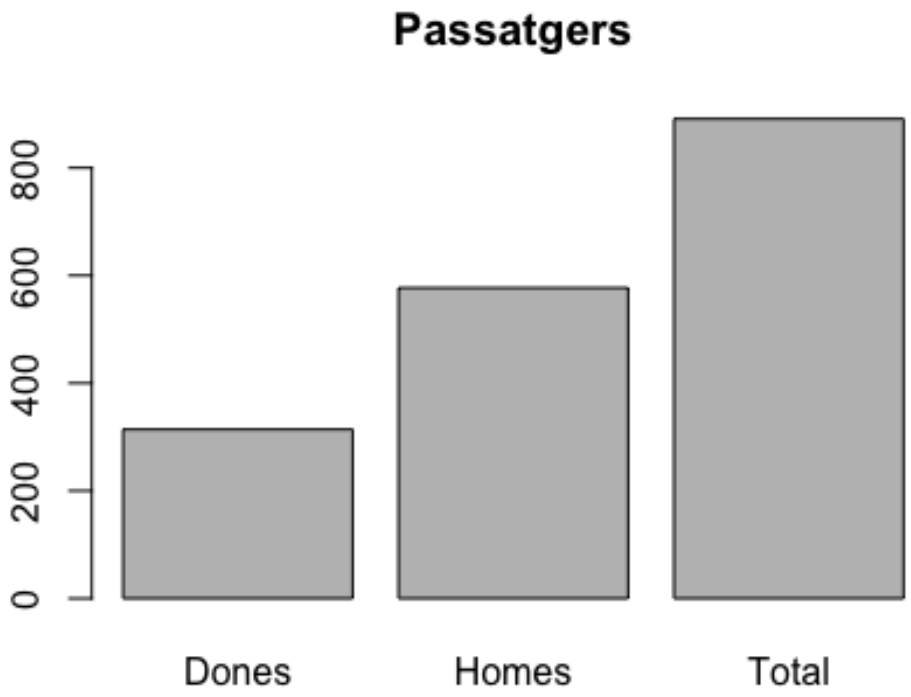
```
#Nombre d'entrades totals de la predicció
nrow(solution)

## [1] 418

# Nombre d'entrades en que la predicció s'ajusta a la solució real
nrow(solution[solution$Real== solution$Predicció,])

## [1] 398

# Percentatge d'error
encert <- nrow(solution[solution$Real==
solution$Predicció,])/nrow(solution) *100
encert

## [1] 95.21531
```

Tot i que el nostre model no semblava l'òptim, en treiem aproximadament un 95% d'encert en les nostres dades i per tant el donarem per vàlid.

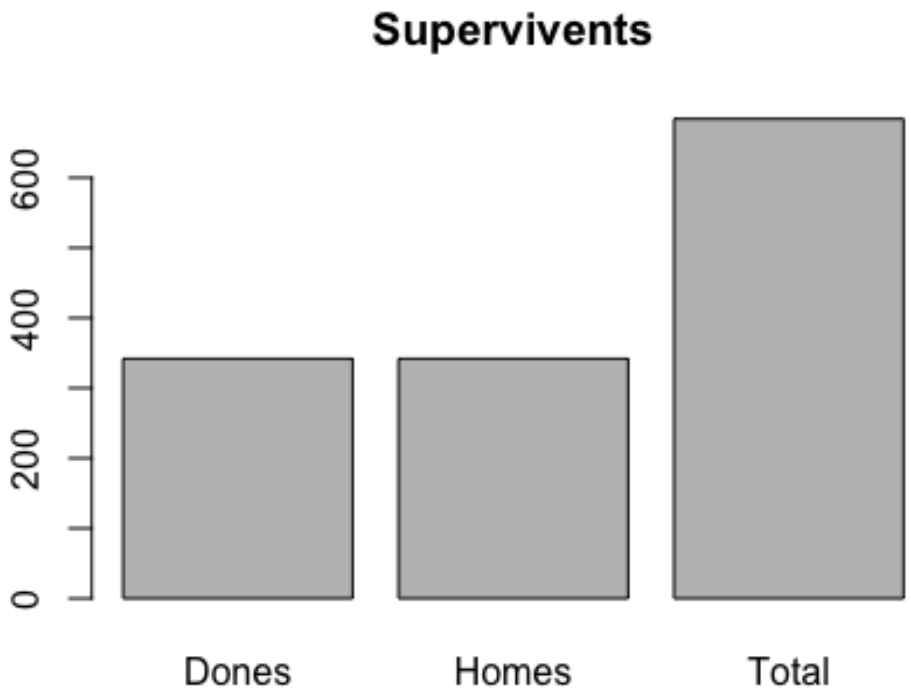
---

5. Representació dels resultats.

```
passatgers<-dsTrain
passatgersFemale <- passatgers[passatgers$Sex =="female",]
passatgersMale <- passatgers[passatgers$Sex =="male",]
counts<- c(nrow(passatgersFemale), nrow(passatgersMale),
nrow(passatgers))
barplot(counts, names = c("Dones","Homes", "Total"), main = "Passatgers"
)
```



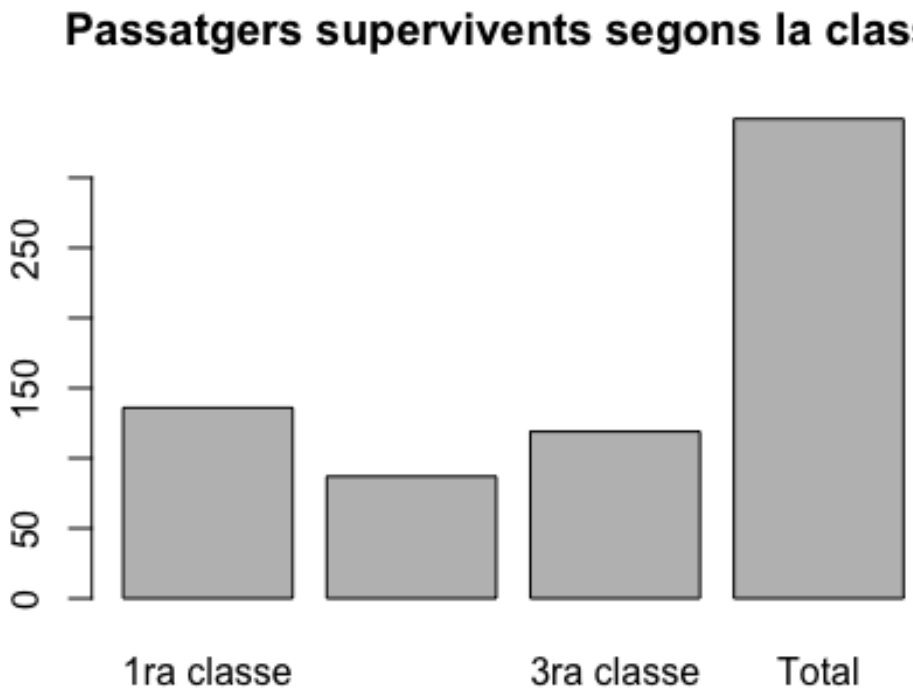

```
superviventsFemale <- passatgersFemale[passatgers$Survived ==1,]
superviventsMale <- passatgersMale[passatgers$Survived ==1,]
counts<- c(nrow(superviventsFemale), nrow(superviventsMale),
nrow(superviventsFemale)+nrow(superviventsMale))
barplot(counts, names = c("Dones","Homes", "Total"), main =
"Supervivents" )
```



En aquestes dues gràfiques podem veure que tot i que el nombre de passsatgers homes fos superior al nombre de pasatgers dones, hi ha gairebé els mateixos supervivents dones que homes.

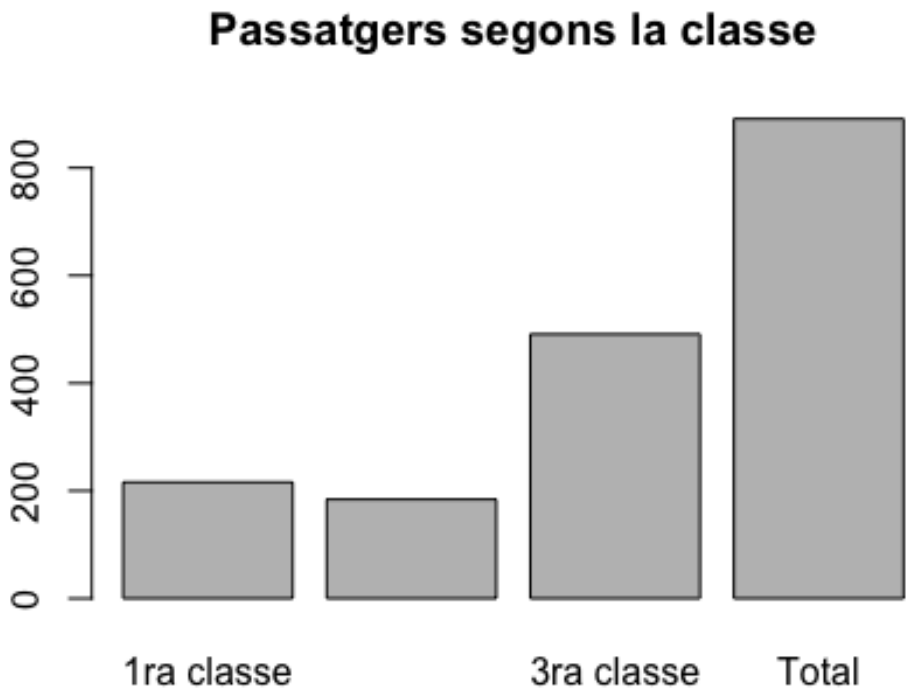
A continuació veurem el mateix per la classe en la que viatjaven els passatgers.

```
supervivents <- passatgers[passatgers$Survived ==1,]
supervivents1class <- supervivents[supervivents$Pclass == 1,]
supervivents2class <- supervivents[supervivents$Pclass == 2,]
supervivents3class <- supervivents[supervivents$Pclass == 3,]
counts<- c(nrow(supervivents1class), nrow(supervivents2class),
nrow(supervivents3class), nrow(supervivents))
barplot(counts, names = c("1ra classe","2na classe", "3ra classe",
"Total"), main = "Passatgers supervivents segons la classe" )
```



Anem a comparar la gràfica amb el nombre de passatgers totals supervivents i no per classe:

```
passatgers1class <- passatgers[passatgers$Pclass == 1,]
passatgers2class <- passatgers[passatgers$Pclass == 2,]
passatgers3class <- passatgers[passatgers$Pclass == 3,]
counts<- c(nrow(passatgers1class), nrow(passatgers2class),
nrow(passatgers3class), nrow(passatgers))
barplot(counts, names = c("1ra classe","2na classe", "3ra classe",
"Total"), main = "Passatgers segons la classe" )
```



Podem veure que els passatgers que viatjaven en 3ra són molts més que els que viatjaven en 1ra i en canvi, han sobreviscut més passatgers que viatjaven en 1ra.

---

6. Resolució del problema.

Les conclusions que hem anat extraient a mesura que hem desenvolupat la pràctica són:

- Les dones han tingut més capacitat de supervivència que els homes
- Els passatgers que viatjaven en primera classe i per tant que van pagar més pels bitllets, van tenir major taxa de supervivència.

El problema que ens havíem plantejat al principi de la pràctica era: *"Quin tipus de gent va tenir mes probabilitats de sobreviure?"* i per tant hem pogut respondre a aquesta pregunta amb ajuda de les proves estadístiques que hem implementat.

---

7. Codi.

Codi en R.

El codi en R està inclòs en aquest mateix fitxer amb extensió rmd. També es pot descarregar en aquest repositori de Github.

Dades finals analitzades.

Les dades amb les que hem treballat(dades finals analitzades) o de sortida s'exporten de la següent manera:

```
write.csv(dsTrain, file ="./titanic/test_clean_out.csv")
write.csv(dsTest, file ="./titanic/train_clean_out.csv")
write.csv(solution, file
="./titanic/solution_prediction_vs_real_out.csv")
```

Tenim les dades de test i train amb les que hem treballat i hem generat també un csv amb la comparació de les dades que prediu el nostre model i les que tenim etiquetades en gender_submission.csv per test.csv.

---

8. Contribucions al treball

Els recursos utilitzats per a l'execució de la pràctica són els proposats com a Material de la pràctica, tenint en consideració els exemples proposats.

Aquest treball ha estat realitzat de forma individual per Jorgina Arrés Cardona.

Investigació prèvia : J.A.C, Redacció de les Respostes: J.A.C, Desenvolupament del codi: J.A.C.