Pràctica 2: Tipologia i cicle de vida de les dades

Victor Garcia Domingo i Ivan Jalencas Lobera

4 de enero, 2021

Contents

1. Descripció del dataset. Per què és important i quina pregunta/problema pretén respondre?

```
df_train <- read.csv("dataset/train.csv")
head(df_train)
```

```
##   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
## 1                             Braund, Mr. Owen Harris   male  22     1
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
## 3                              Heikkinen, Miss. Laina female  26     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1
## 5                            Allen, Mr. William Henry   male  35     0
## 6                                    Moran, Mr. James   male  NA     0
##   Parch           Ticket    Fare Cabin Embarked
## 1     0        A/5 21171  7.2500              S
## 2     0         PC 17599 71.2833   C85        C
```

```
## 3     0 STON/O2. 3101282  7.9250             S
## 4     0           113803 53.1000  C123       S
## 5     0           373450  8.0500             S
## 6     0           330877  8.4583             Q
```

```
str(df_train)
```

```
## '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 16 559 520 629 417 581
##  $ 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 ...
```

```
sapply(df_train, class)
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##   "integer"   "integer"   "integer"    "factor"    "factor"   "numeric"
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##   "integer"   "integer"    "factor"   "numeric"    "factor"    "factor"
```

```
df_test <- read.csv("dataset/test.csv")
head(df_test)
```

```
##   PassengerId Pclass                                          Name    Sex
## 1         892      3                              Kelly, Mr. James   male
## 2         893      3              Wilkes, Mrs. James (Ellen Needs) female
## 3         894      2                     Myles, Mr. Thomas Francis   male
## 4         895      3                               Wirz, Mr. Albert   male
## 5         896      3 Hirvonen, Mrs. Alexander (Helga E Lindqvist) female
## 6         897      3                     Svensson, Mr. Johan Cervin   male
##    Age SibSp Parch  Ticket    Fare Cabin Embarked
## 1 34.5     0     0  330911  7.8292              Q
## 2 47.0     1     0  363272  7.0000              S
## 3 62.0     0     0  240276  9.6875              Q
## 4 27.0     0     0  315154  8.6625              S
## 5 22.0     1     1 3101298 12.2875              S
## 6 14.0     0     0    7538  9.2250              S
```

```
str(df_test)
```

```
## 'data.frame':    418 obs. of  11 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
##  $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
##  $ Name       : Factor w/ 418 levels "Abbott, Master. Eugene Joseph",..: 210 409 273 414 182 370 85 5
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 2 2 1 2 1 2 1 2 ...
##  $ Age        : num  34.5 47 62 27 22 14 30 26 18 21 ...
##  $ SibSp      : int  0 1 0 0 1 0 0 1 0 2 ...
##  $ Parch      : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ Ticket     : Factor w/ 363 levels "110469","110489",..: 153 222 74 148 139 262 159 85 101 270 ...
```

```
##  $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
##  $ Cabin      : Factor w/ 77 levels "","A11","A18",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 2 3 2 3 3 3 2 3 1 3 ...
```

```
sapply(df_test, class)
```

```
## PassengerId      Pclass        Name         Sex         Age       SibSp
##   "integer"   "integer"    "factor"    "factor"   "numeric"   "integer"
##       Parch      Ticket        Fare       Cabin    Embarked
##   "integer"    "factor"   "numeric"    "factor"    "factor"
```

El dataset és un clàssic que descriu els passatgers del Titanic. Està dividit en dos conjunt de dades amb 12 variables. El conjunt d'entrenament es composa de 891 observacions i el de test de 418, tot i que aquest li falta la variable Survived. Les variables són les següents:

- **PassengerId**: *identificador de passatger*
- **Survived:** *si ha sobreviscut (1) o no (0)*
- **Pclass:** *classe a la que viatjava (1, 2 o 3)*
- **Name:** *nom*
- **Sex:** *sexe (female o male)*
- **Age:** *edat*
- **SibSp:** *nombre de germans*
- **Parch:** *nombre de pares o fills a bord*
- **Ticket:** *número de tiquet*
- **Fare:** *tarifa*
- **Cabin:** *cabina*
- **Embarked:** *port d'embarcament (C = Cherbourg; Q = Queenstown; S = Southampton)*

Aquest dataset és important perquè permet descobrir si els passatgers del titanic van sobreviure i quines variables hi estan relacionades. Al ser un dataset tant conegut també serveix com a benchmark a l'hora de provar nous models de predicció.

La pregunta que ens plantegem és, hi ha factors que tenen més incidència a l'hora d'explicar la supervivència? Ens podem fer preguntes més concretes com si van sobreviure més els viatges de primera classe sobrevisqués respecte als de segona o tercera classe. Viatjar sol garantia més probabilitats de supervivència que anant en familia o potser va ser a l'inversa?

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

En aquest apartat, carregarem i seleccionarem les dades que ens seran útils per a posteriors anàlisis. Treballarem principalment amb el grup de dades d'entrenament, però per algun apartat treballarem amb el conjunt de dades sencer, per lo que les haurem d'integrar en un mateix dataframe.

Seleccionem les que ens interessen. Per a fer l'anàlisi posterior, no ens interessa l'identificador de passatger, el nom, el número de tiquet ni la cabina. Tampoc la tarifa (Fare), ja que creiem que la variable *Pclass* ja dóna informació relativa al poder adquisitiu del passatger. Seleccionarem la resta de variables i posarem la variable *Survived* al final, ja que serà la dependent per a fer prediccions. També convertirem a factors les variables *Sex*, *Embarked*, *Pclass* i *Survived*.

```
if(!require(dplyr)){
  install.packages("dplyr")
  library(dplyr)
}
```

```
## Loading required package: 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
```

```
attach(df_train)

df_sel_train <- df_train %>% select(Pclass, Sex, Age, SibSp, Parch, Embarked, Survived)

df_sel_train$isTrain <- 1   # Aquesta variable ens servirà per distingir del conjunt d'entrenament i el

df_sel_train$hasFamily <- ifelse(df_sel_train$Parch + df_sel_train$SibSp > 0, "Family", "No family") # A

df_sel_train$Sex <- factor(df_sel_train$Sex)
df_sel_train$Embarked <- factor(df_sel_train$Embarked)
df_sel_train$Pclass <- factor(df_sel_train$Pclass)
df_sel_train$Survived <- factor(df_sel_train$Survived)
df_sel_train$hasFamily <- factor(df_sel_train$hasFamily)

head(df_sel_train)
```

```
##   Pclass    Sex Age SibSp Parch Embarked Survived isTrain hasFamily
## 1      3   male  22     1     0        S        0       1    Family
## 2      1 female  38     1     0        C        1       1    Family
## 3      3 female  26     0     0        S        1       1 No family
## 4      1 female  35     1     0        S        1       1    Family
## 5      3   male  35     0     0        S        0       1 No family
## 6      3   male  NA     0     0        Q        0       1 No family
```

```
attach(df_test)
```

```
## The following objects are masked from df_train:
##
##     Age, Cabin, Embarked, Fare, Name, Parch, PassengerId, Pclass,
##     Sex, SibSp, Ticket
```

```
df_sel_test <- df_test %>% select(Pclass, Sex, Age, SibSp, Parch, Embarked)

attach(df_sel_test)
```

```
## The following objects are masked from df_test:
##
##     Age, Embarked, Parch, Pclass, Sex, SibSp
```

```
## The following objects are masked from df_train:
##
##     Age, Embarked, Parch, Pclass, Sex, SibSp
```

```
df_sel_test$hasFamily <- ifelse(df_sel_test$Parch + df_sel_test$SibSp > 0, "Family", "No family")

df_sel_test$Sex <- factor(df_sel_test$Sex)
df_sel_test$Embarked <- factor(df_sel_test$Embarked)
df_sel_test$Pclass <- factor(df_sel_test$Pclass)
df_sel_test$isTrain <- 1   # Aquesta variable ens servirà per distingir del conjunt d'entrenament i el
df_sel_test$hasFamily <- factor(df_sel_test$hasFamily)
```

```
df_sel_test$Survived <- NA
df_sel_test$isTrain <- 0

head(df_sel_test)
```

```
##   Pclass    Sex  Age SibSp Parch Embarked hasFamily isTrain Survived
## 1      3   male 34.5     0     0        Q No family       0       NA
## 2      3 female 47.0     1     0        S    Family       0       NA
## 3      2   male 62.0     0     0        Q No family       0       NA
## 4      3   male 27.0     0     0        S No family       0       NA
## 5      3 female 22.0     1     1        S    Family       0       NA
## 6      3   male 14.0     0     0        S No family       0       NA
```

```
df_sel_complete <- rbind(df_sel_train, df_sel_test)
df_sel_complete$isTrain <- factor(df_sel_complete$isTrain)
levels(df_sel_complete$Survived)[levels(df_sel_complete$Survived) == 0] <- "Didn't survive"
levels(df_sel_complete$Survived)[levels(df_sel_complete$Survived) == 1] <- "Survived"

levels(df_sel_complete$isTrain)[levels(df_sel_complete$isTrain) == 0] <- "Test"
levels(df_sel_complete$isTrain)[levels(df_sel_complete$isTrain) == 1] <- "Train"

str(df_sel_complete)
```

```
## 'data.frame':    1309 obs. of  9 variables:
##  $ Pclass   : Factor w/ 3 levels "1","2","3": 3 1 3 1 3 3 1 3 3 2 ...
##  $ 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 ...
##  $ Embarked : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
##  $ Survived : Factor w/ 2 levels "Didn't survive",..: 1 2 2 2 1 1 1 1 2 2 ...
##  $ isTrain  : Factor w/ 2 levels "Test","Train": 2 2 2 2 2 2 2 2 2 2 ...
##  $ hasFamily: Factor w/ 2 levels "Family","No family": 1 1 2 1 2 2 2 1 1 1 ...
```

3. Neteja de les dades

3.1. Les dades contenen zeros o elements buits? Com gestionaries aquests casos?

A continuació, es descriuen els estadístics de les dades i els valors nuls. Podem observar que les variables *Age* i *Survived* tenen valors nuls (NA) i la variables *Embarked*, valors buits.

```
summary(df_sel_complete)
```

```
##  Pclass      Sex           Age            SibSp           Parch
##  1:323   female:466   Min.   : 0.17   Min.   :0.0000   Min.   :0.000
##  2:277   male  :843   1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.000
##  3:709                Median :28.00   Median :0.0000   Median :0.000
##                       Mean   :29.88   Mean   :0.4989   Mean   :0.385
##                       3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.000
##                       Max.   :80.00   Max.   :8.0000   Max.   :9.000
##                       NA's   :263
##  Embarked           Survived    isTrain        hasFamily
##   :  2    Didn't survive:549   Test :418   Family   :519
```

```
##  C:270    Survived      :342   Train:891   No family:790
##  Q:123    NA's          :418
##  S:914
##
##
##
```

Per a la variable *Age*, s'ha decidit imputar el valor amb k-means, ja que generalment imputar els valors dels veïns més propers és bastant eficaç. Per a la variable *Embarked*, amb el valor més comú, que és 'S'. Deixarem els valosr nuls de la variable *Survived* com estan ja que aquests s'haurien d'omplir mitjançant un model de predicció, que no abordarem en aquesta pràctica.

- Imputació de la variable *Age*:

```
if(!require(VIM)){
  install.packages("VIM")
  library(VIM)
}
```

```
## Loading required package: VIM
```

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

```
## Loading required package: colorspace
```

```
## Loading required package: grid
```

```
## Loading required package: data.table
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## VIM is ready to use.
##  Since version 4.0.0 the GUI is in its own package VIMGUI.
##
##           Please use the package to use the new (and old) GUI.
```

```
## Suggestions and bug-reports can be submitted at: https://github.com/alexkowa/VIM/issues
```

```
##
## Attaching package: 'VIM'
```

```
## The following object is masked from 'package:datasets':
##
##     sleep
```

```
sum(is.na(df_sel_complete$Age))
```

```
## [1] 263
```

```
# imputation through 3 nearest neighbours
df_sel_complete <- kNN(df_sel_complete, variable = c("Age"), k = 3)
df_sel_complete <- df_sel_complete %>% select(Pclass, Sex, Age, SibSp, Parch, Embarked, isTrain, hasFami

sum(is.na(df_sel_complete$Age))
```

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

Podem comprovar que dels 263 valors nuls inicials, ja no hi ha cap.

- Imputació de la variable *Embarked*:

```
df_sel_complete[which(df_sel_complete$Embarked == ""),]$Embarked <- "S"

summary(df_sel_complete$Embarked)
```

```
##       C   Q   S
##   0 270 123 916
```

```
df_sel_complete$Embarked <- factor(df_sel_complete$Embarked)
```

Comprovem també que la variable *Embarked* ja no té valors buits

```
df_sel_train <- subset(df_sel_complete, isTrain == "Train")

str(df_sel_train)
```

```
## 'data.frame':    891 obs. of  9 variables:
##  $ Pclass   : Factor w/ 3 levels "1","2","3": 3 1 3 1 3 3 1 3 3 2 ...
##  $ Sex      : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age      : num  22 38 26 35 35 65 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 ...
##  $ Embarked : Factor w/ 3 levels "C","Q","S": 3 1 3 3 3 2 3 3 3 1 ...
##  $ isTrain  : Factor w/ 2 levels "Test","Train": 2 2 2 2 2 2 2 2 2 2 ...
##  $ hasFamily: Factor w/ 2 levels "Family","No family": 1 1 2 1 2 2 2 1 1 1 ...
##  $ Survived : Factor w/ 2 levels "Didn't survive",..: 1 2 2 2 1 1 1 1 2 2 ...
```

3.2. Identificació i tractament de valors extrems

Primer de tot, descriurem els valors de les dades i analitzarem els valors extrems.

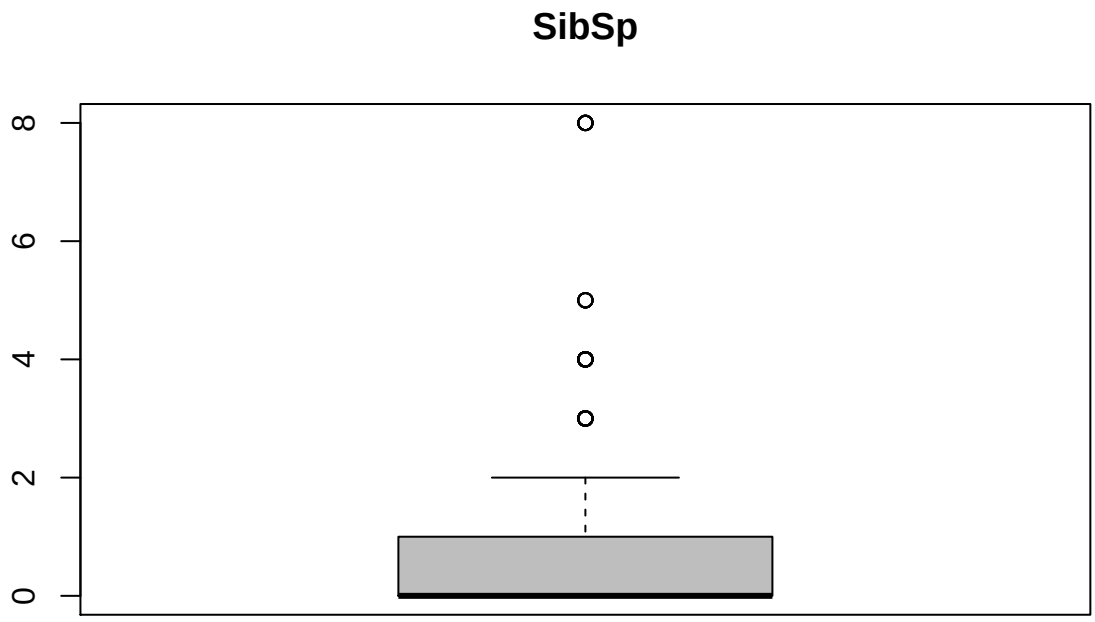
```
summary(df_sel_train)
```

```
##  Pclass      Sex           Age            SibSp           Parch
##  1:216   female:314   Min.   : 0.42   Min.   :0.000   Min.   :0.0000
##  2:184   male  :577   1st Qu.:21.00   1st Qu.:0.000   1st Qu.:0.0000
##  3:491                Median :28.00   Median :0.000   Median :0.0000
##                       Mean   :29.64   Mean   :0.523   Mean   :0.3816
##                       3rd Qu.:38.00   3rd Qu.:1.000   3rd Qu.:0.0000
##                       Max.   :80.00   Max.   :8.000   Max.   :6.0000
##  Embarked  isTrain        hasFamily            Survived
##  C:168    Test :  0   Family   :354   Didn't survive:549
##  Q: 77    Train:891   No family:537   Survived      :342
##  S:646
##
##
##
```

Podem veure que la variable SibSp té com a mitjana 0,523 i com a valor màxim 8. La variable Parch, té com a mitjana el valor 0,3816 i com a màxim, el valor 8. També observem com el valor màxim de la variable Age és el doble del valor del tercer quartil. Aquestes tres variables són candidates a tenir valors extrems. Analitzem els diagrames de caixes.

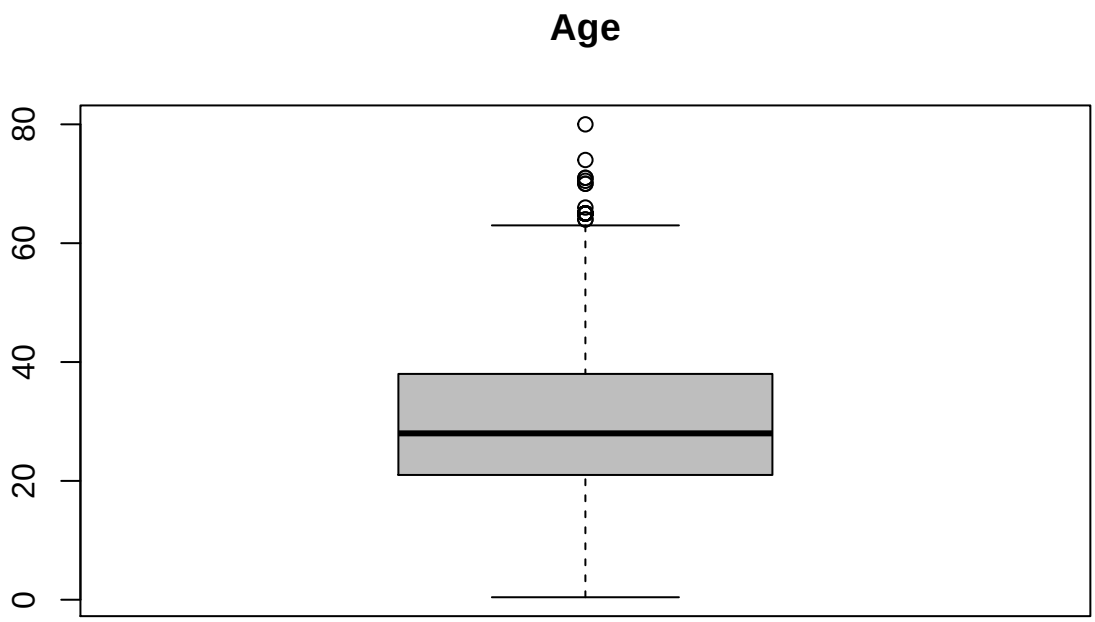
```
quantitativeVars <- c("SibSp", "Parch", "Age")

for (i in 1:length(quantitativeVars)){
  boxplot(df_sel_train[quantitativeVars[i]], main=quantitativeVars[i], col = "gray")
```

```
  print(quantitativeVars[i])
  print(boxplot.stats(df_sel_train[[quantitativeVars[i]]])$out)
}
```




```
## [1] "SibSp"
##  [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
## [36] 5 4 3 4 8 4 3 4 8 4 8
```


```
## [1] "Parch"
##   [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
##  [36] 2 1 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
##  [71] 1 2 1 2 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
## [106] 2 3 4 1 2 1 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
## [141] 2 1 1 2 5 2 1 1 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
## [176] 1 2 1 2 3 1 2 1 2 2 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
## [211] 1 5 2
```




```
## [1] "Age"
##  [1] 65.0 66.0 65.0 71.0 70.5 65.0 65.0 65.0 65.0 65.0 65.0 65.0 64.0 65.0
## [15] 65.0 65.0 71.0 65.0 64.0 65.0 65.0 65.0 65.0 80.0 70.0 65.0 70.0 65.0
## [29] 65.0 65.0 65.0 74.0
```

S'observa que, si compten els valors de les variables *SibSp* i *Parch*, ens trobem amb que la primera té set valors igual a 8 i cinc igual a 5, i la segona té un valor igual a 6. En el cas de la variable *Age* hi ha 8 casos amb 66 anys o més. Els eliminarem,

```
df_sel_train %>% count(SibSp, sort = TRUE)
```

```
## # A tibble: 7 x 2
##   SibSp     n
##   <int> <int>
## 1     0   608
## 2     1   209
## 3     2    28
## 4     4    18
## 5     3    16
## 6     8     7
## 7     5     5
```

```
df_sel_train %>% count(Parch, sort = TRUE)
```

```
## # A tibble: 7 x 2
##   Parch     n
##   <int> <int>
## 1     0   678
```

```
## 2     1   118
## 3     2    80
## 4     3     5
## 5     5     5
## 6     4     4
## 7     6     1
```

```
df_sel_train %>% count(Age, sort = FALSE)
```

```
## # A tibble: 88 x 2
##      Age     n
##    <dbl> <int>
##  1  0.42     1
##  2  0.67     1
##  3  0.75     2
##  4  0.83     2
##  5  0.92     1
##  6  1        7
##  7  2       10
##  8  3       10
##  9  4       11
## 10  5        4
## # ... with 78 more rows
```

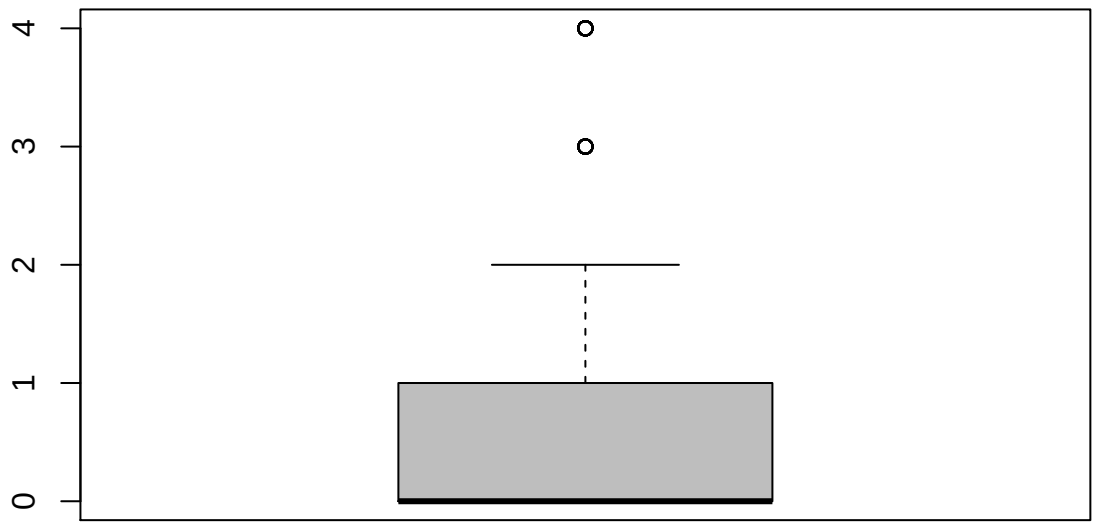
Eliminem les observacions indicades. Ens quedem 870 observacions.

```
df_sel_train<-df_sel_train[!(df_sel_train$SibSp==5 | df_sel_train$SibSp==8 | df_sel_train$Parch==6| df_s

str(df_sel_train)
```

```
## 'data.frame':    870 obs. of  9 variables:
##  $ Pclass   : Factor w/ 3 levels "1","2","3": 3 1 3 1 3 3 1 3 3 2 ...
##  $ Sex      : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age      : num  22 38 26 35 35 65 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 ...
##  $ Embarked : Factor w/ 3 levels "C","Q","S": 3 1 3 3 3 2 3 3 3 1 ...
##  $ isTrain  : Factor w/ 2 levels "Test","Train": 2 2 2 2 2 2 2 2 2 2 ...
##  $ hasFamily: Factor w/ 2 levels "Family","No family": 1 1 2 1 2 2 2 1 1 1 ...
##  $ Survived : Factor w/ 2 levels "Didn't survive",..: 1 2 2 2 1 1 1 1 2 2 ...
```

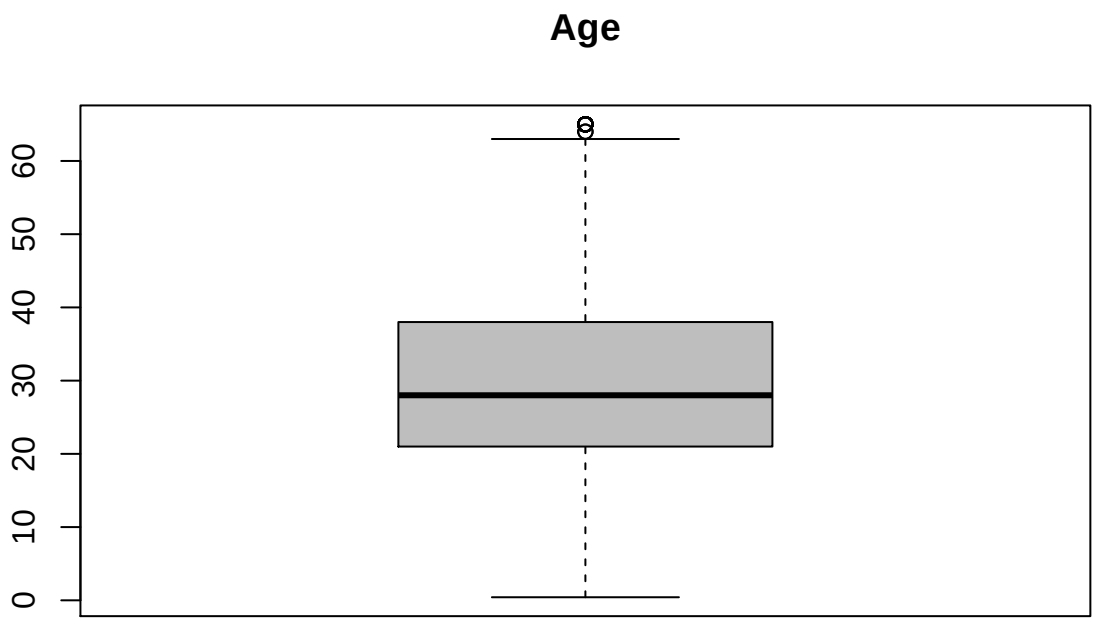
Observem la nova distribució.

```
quantitativeVars <- c("SibSp", "Parch", "Age")

for (i in 1:length(quantitativeVars)){
  boxplot(df_sel_train[quantitativeVars[i]], main=quantitativeVars[i], col = "gray")
  print(quantitativeVars[i])
  print(boxplot.stats(df_sel_train[[quantitativeVars[i]]])$out)
}
```




```
## [1] "SibSp"
##  [1] 3 4 3 3 4 3 4 3 3 4 4 4 3 4 3 4 4 4 4 3 3 3 3 4 4 3 3 4 3 4 4 3 4 4
```


```
## [1] "Parch"
##   [1] 1 2 1 5 1 1 5 2 2 1 1 2 2 1 2 2 3 2 2 1 1 1 1 2 1 1 2 2 1 2 2 2 1 1 1
##  [36] 2 1 4 1 1 1 1 2 1 2 1 1 1 1 1 2 2 2 1 1 2 2 1 2 1 1 1 1 1 1 1 2 1 2 2
##  [71] 1 1 2 1 1 1 1 1 1 2 1 1 1 4 1 1 2 2 2 2 1 1 1 2 2 1 1 2 2 3 4 1 2 1 1
## [106] 2 1 2 1 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
## [141] 1 2 1 5 2 1 1 1 2 1 1 1 2 1 1 1 1 1 1 1 3 2 1 1 1 2 1 2 3 1 2 1 2 1 1
## [176] 2 1 2 1 2 1 1 1 2 1 1 1 2 1 1 1 1 3 1 1 1 1 5 2
```




```
## [1] "Age"
##  [1] 65 65 65 65 65 65 65 65 65 64 65 65 65 65 64 65 65 65 65 65 65 65 65
## [24] 65
```

```
write.csv(df_sel_complete,"titanic.csv", row.names = FALSE)
```

4. Anàlisis de les dades

4.1. Selecció dels grups de dades.

Les variables disponibles són Pclass, Sex, Age, SibSp, Parch, Embarked i Survived. Podem agrupar les dades de la següent manera:

- Per sexe:

```
titanic.female <- df_sel_train[df_sel_train$Sex == "female",]
titanic.male <- df_sel_train[df_sel_train$Sex == "male",]
```

- Per Edat:

```
titanic.zerototwelve <- df_sel_train[df_sel_train$Age < 13,]
titanic.thirteentoeighteen <- df_sel_train[df_sel_train$Age > 12 & df_sel_train$Age < 19,]
titanic.nineteento59 <- df_sel_train[df_sel_train$Age > 18 & df_sel_train$Age < 60,]
titanic.sixtyormore <- df_sel_train[df_sel_train$Age > 59,]
```

- Per si viatgen anb la familia o no:

```
titanic.familyyes <- df_sel_train[df_sel_train$SibSp > 0 | df_sel_train$Parch > 0,]
titanic.familyno <- df_sel_train[df_sel_train$SibSp == 0 & df_sel_train$Parch == 0,]
```

- Per port d'embarcament:

```
titanic.portS <- df_sel_train[df_sel_train$Embarked == "S",]
titanic.portC <- df_sel_train[df_sel_train$Embarked == "C",]
titanic.portQ <- df_sel_train[df_sel_train$Embarked == "Q",]
```

- Per si han sobreviscut o no:

```
titanic.survivedyes <- df_sel_train[df_sel_train$Survived == "Survived",]
titanic.survivedno <- df_sel_train[df_sel_train$Survived == "Didn't survive",]
```

4.2. Comprovació de la normalitat i homogeneïtat de la variància.

4.2.1. Comprovació de la normalitat:

```
if(!require(nortest)){
  install.packages("nortest")
  library(nortest)
}
```

```
## Loading required package: nortest
```

```
alpha = 0.05
col.names = colnames(df_sel_complete)

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

      if (i < ncol(df_sel_complete) - 1) cat(", ")
      if (i %% 3 == 0) cat("\n")
    }
  }
}
```

```
## Variables que no siguen una distribución normal:
## Age,
## SibSp, Parch,
```

4.2.2. Homogeneïtat de la variància:

```
fligner.test(Age ~ Survived, data = df_sel_complete)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Age by Survived
## Fligner-Killeen:med chi-squared = 0.01647, df = 1, p-value =
## 0.8979
```

La majòria de les variables són categòriques. Per això, fem l'anàlisi de l'homogeneitat de la variància de la variable *Age* pel cas dels que han sobresviscut i els que no. Com el p-value és superior a 0.05 no podem descartar la hipòtesis nul · La de que les variàncies són homogènees.

4.3. Proves estadístiques.

4.3.1 Correlacions

Per a fer el test de correlacions, com que la major part de les dades són categòriques, obtindrem una matriu de V de Cramer.

```
if(!require(vcd)){
  install.packages("vcd")
  library(vcd)
}
```

```
## Loading required package: vcd
```

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

```
ccatcorrm <- function(vars, dat)
  sapply(vars, function(y)
  sapply(vars, function(x)
  assocstats(table(dat[,x], dat[,y]))$cramer))

ccatcorrm(c('Pclass', 'Sex', 'Age', 'SibSp', 'Parch', 'Embarked', 'Survived'), df_sel_train)
```

```
##              Pclass       Sex       Age     SibSp      Parch  Embarked
## Pclass   1.00000000 0.1461773 0.4845108 0.1542006 0.09113043 0.2606711
## Sex      0.14617726 1.0000000 0.4045389 0.2182026 0.26780076 0.1247533
## Age      0.48451082 0.4045389 1.0000000 0.4582249 0.43290374 0.5648919
## SibSp    0.15420056 0.2182026 0.4582249 1.0000000 0.25120435 0.1136442
## Parch    0.09113043 0.2678008 0.4329037 0.2512043 1.00000000 0.0918220
## Embarked 0.26067112 0.1247533 0.5648919 0.1136442 0.09182200 1.0000000
## Survived 0.33772021 0.5504655 0.4469344 0.1858145 0.19885197 0.1695898
##           Survived
## Pclass   0.3377202
## Sex      0.5504655
## Age      0.4469344
## SibSp    0.1858145
## Parch    0.1988520
## Embarked 0.1695898
## Survived 1.0000000
```

Com es pot observar, no hi ha correlacions molt intenses, sent les màximes entre *Sex* i *Survived* i entre *Embarked* i *Age*.

4.3.2 Contrast d'hipòtesis

Volem saber si l'edat dels que van sobreviure és més gran que la dels que no. Per a això, realitzem un contrast d'hipòtesis en el que la hipòtesi nul · la serà que són iguals,i l'alternativa, que l'edat és més alta per als que van sobreviure.

Podriem fer servir el t-student, ja que les mostres són més grans de 30 i això permet suposar normalitat, però aplicarem el test no paramètric de Mann-Whitney perquè hem vist que les variables Pclass, Age, SibSp, Parch, Survived no segueixen una distribució normal.

```r
wilcox.test(titanic.survivedyes$Age, titanic.survivedno$Age, alternative = "greater")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  titanic.survivedyes$Age and titanic.survivedno$Age
## W = 80872, p-value = 0.995
## alternative hypothesis: true location shift is greater than 0
```

El p-value és de 0.9951 per lo que no podem rebutjar la hipòtesi nul · la de que l'edat dels que han sobreviscut és distribueix igual a la dels que no han sobreviscut.

En qualsevol cas, si apliquem el t student, el p-value és 0,9996, no molt allunyat de Mann-Whitney.

```r
t.test(titanic.survivedyes$Age, titanic.survivedno$Age, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  titanic.survivedyes$Age and titanic.survivedno$Age
## t = -3.3736, df = 729.29, p-value = 0.9996
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  -4.900969       Inf
## sample estimates:
## mean of x mean of y
##  27.50636  30.79962
```

Per altra banda, volem comparar si la proporció de supervivents de primera clase és significativament superior a la dels dos altres grups.

```r
cross <- table(df_sel_train$Survived, df_sel_train$Pclass)
addmargins(cross)
```

```
##
##                    1   2   3 Sum
##   Didn't survive  77  95 357 529
##   Survived       135  87 119 341
##   Sum            212 182 476 870
```

```r
prop.test(c(135, 87+119), c(212, 182 + 476), alternative = "greater")
```

```
##
##  2-sample test for equality of proportions with continuity
##  correction
##
## data:  c(135, 87 + 119) out of c(212, 182 + 476)
## X-squared = 69.153, df = 1, p-value < 2.2e-16
## alternative hypothesis: greater
## 95 percent confidence interval:
##  0.258669 1.000000
## sample estimates:
##    prop 1    prop 2
## 0.6367925 0.3130699
```

El p-value és molt inferior a 0.05, per lo que podem descartar la hipòtesis nul · la de que les proporcions són iguals i acceptar la hipòtesis alternativa de que la supervivència a la primera classe era superior a la de resta de classes.

```r
cross <- table(df_sel_train$Survived, df_sel_train$hasFamily)
addmargins(cross)
```

```
##
##                  Family No family Sum
##   Didn't survive    161       368 529
##   Survived          179       162 341
##   Sum               340       530 870
```

```r
prop.test(c(179, 162), c(340, 530), alternative = "greater")
```

```
##
##  2-sample test for equality of proportions with continuity
##  correction
##
## data:  c(179, 162) out of c(340, 530)
## X-squared = 41.453, df = 1, p-value = 6.037e-11
## alternative hypothesis: greater
## 95 percent confidence interval:
##  0.1630139 1.0000000
## sample estimates:
##    prop 1    prop 2
## 0.5264706 0.3056604
```

A l'igual que abans, el p-value és molt més petit que 0.05, per lo que rebutgem la hipòtesi nul · la de igualtat a les proporcions i acceptem la hipòtesi alternativa que afirma que la tasa de supervivència era superior en el cas de viatjar amb família.

4.3.3 Regressió logística

Com que la variable a predir és categòrica, necessitem fer servir un model de regressió logística. Les variables més relacionades són SibSp, Parch i Sex. Provarem amb diferents combinacions de les tres.

```r
# Variables independents
sex <- df_sel_train$Sex
parentschildren <- df_sel_train$Parch
siblings <- df_sel_train$SibSp
pclass <- df_sel_train$Pclass
age <- df_sel_train$Age

# Variable dependent
survived <- df_sel_train$Survived

model1 <- glm(survived ~ sex + parentschildren + siblings, data = df_sel_train, family = 'binomial')
model2 <- glm(survived ~ sex + parentschildren, data = df_sel_train, family = 'binomial')
model3 <- glm(survived ~ sex + siblings, data = df_sel_train, family = 'binomial')
model4 <- glm(survived ~ sex, data = df_sel_train, family = 'binomial')
model5 <- glm(survived ~ sex + parentschildren + siblings + pclass + age, data = df_sel_train, family =
```

És interessant parar atenció a l'Akaike Information Criterion (AIC). Aquest és un test equivalent al R2, però aplicat a la regressió logística. Permet comparar la bondat d'ajust entre diferents models, veure com de rellevants són els regressors i evitar que es produeixi overfitting. Contra més baix sigui l'índex, millor un model respecte a un altre.A partir de la taula inferior, podem veure que el model 5 és el millor, ja que té un AIC inferior a la resta. Per tant, les tres variables que més correlacionades estan amb la supervivència són les que expliquen millor el model.

```
tabla.coeficientes <- matrix(c(1, AIC(model1),
                               2, AIC(model2),
                               3, AIC(model3),
                               4, AIC(model4),
                               5, AIC(model5)
                               ),
ncol = 2, byrow = TRUE)
colnames(tabla.coeficientes) <- c("Modelo", "AIC")
tabla.coeficientes
```

```
##      Modelo      AIC
## [1,]      1 899.1069
## [2,]      2 899.7433
## [3,]      3 897.1456
## [4,]      4 898.3578
## [5,]      5 781.5252
```

```
summary(model5)
```

```
##
## Call:
## glm(formula = survived ~ sex + parentschildren + siblings + pclass +
##     age, family = "binomial", data = df_sel_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7900  -0.6068  -0.3911   0.6331   2.4957
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      4.405891   0.429044  10.269  < 2e-16 ***
## sexmale         -2.681492   0.199441 -13.445  < 2e-16 ***
## parentschildren -0.046111   0.120395  -0.383   0.7017
## siblings        -0.344758   0.123927  -2.782   0.0054 **
## pclass2         -1.319804   0.268049  -4.924 8.49e-07 ***
## pclass3         -2.598090   0.263991  -9.842  < 2e-16 ***
## age             -0.048779   0.008165  -5.974 2.31e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1165.13  on 869  degrees of freedom
## Residual deviance:  767.53  on 863  degrees of freedom
## AIC: 781.53
##
## Number of Fisher Scoring iterations: 5
```

Fent un resum del model 5, podem afirmar que el cas en que el viatger era una dona, que viatjava en primera classe i no tenia familila era el més favorable a l'hora de sobreviure. Tots els actors factors eren negatius, sent home i viatjant en tercera classe la pitjor de les combinacions. Veiem alguns exemples.

```
newdata <- data.frame(
  sex = "female",
  parentschildren = 3,
```

```
  siblings = 4,
  pclass = "1",
  age = 15
)

print(paste0('Dona amb 3 pares i/o fills, 4 germans, de primera classe i de 15 anys: ', predict(model5,
```

```
## [1] "Dona amb 3 pares i/o fills, 4 germans, de primera classe i de 15 anys: 2.15684602210339"
```

```
newdata <- data.frame(
  sex = "female",
  parentschildren = 3,
  siblings = 4,
  pclass = "1",
  age = 30
)

print(paste0('Dona amb 3 pares i/o fills, 4 germans, de primera classe i de 30 anys: ', predict(model5,
```

```
## [1] "Dona amb 3 pares i/o fills, 4 germans, de primera classe i de 30 anys: 1.42516835680465"
```

```
newdata <- data.frame(
  sex = "female",
  parentschildren = 3,
  siblings = 4,
  pclass = "2",
  age = 30
)

print(paste0('Dona amb 3 pares i/o fills, 4 germans, de segona classe i de 30 anys: ', predict(model5, n
```

```
## [1] "Dona amb 3 pares i/o fills, 4 germans, de segona classe i de 30 anys: 0.10536454212537"
```

```
newdata <- data.frame(
  sex = "male",
  parentschildren = 3,
  siblings = 4,
  pclass = "1",
  age = 15
)
# Preder el precio
print(paste0('Home amb 3 pares i/o fills, 4 germans, de primera classe i de 15 anys: ', predict(model5,
```

```
## [1] "Home amb 3 pares i/o fills, 4 germans, de primera classe i de 15 anys: -0.524646102660461"
```

Els resultats permeten observar com el sexe influeix molt en la supervivència, més que les altres variables.

5 Representació dels resultats a partir de taules i gràfiques.

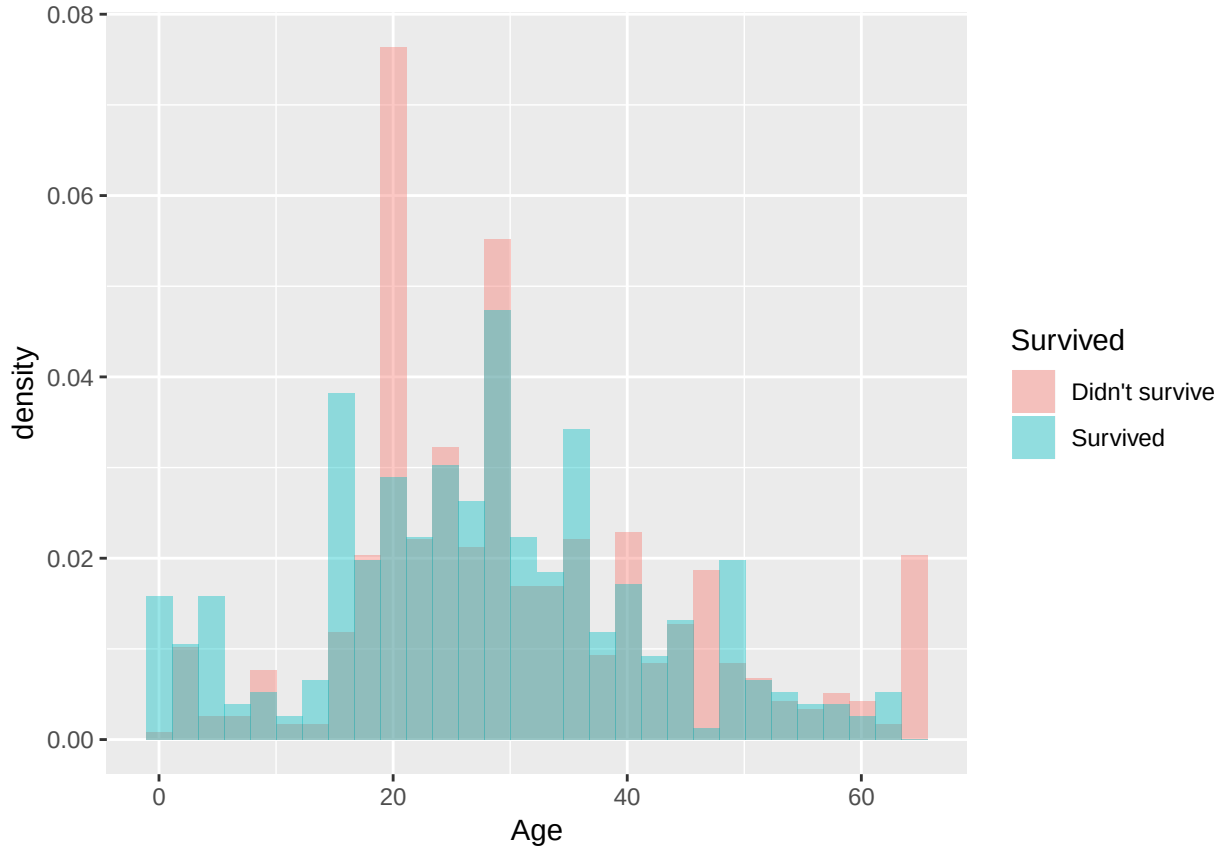
Començarem representant les distribucions per edat dels que van sobreviure i els que no, vinculat al constrast d'hipòtesis.

```
if(!require(ggpubr)){
install.packages("ggpubr")
library(ggpubr)
}
```

```
## Loading required package: ggpubr
## Loading required package: ggplot2
## Loading required package: magrittr
```

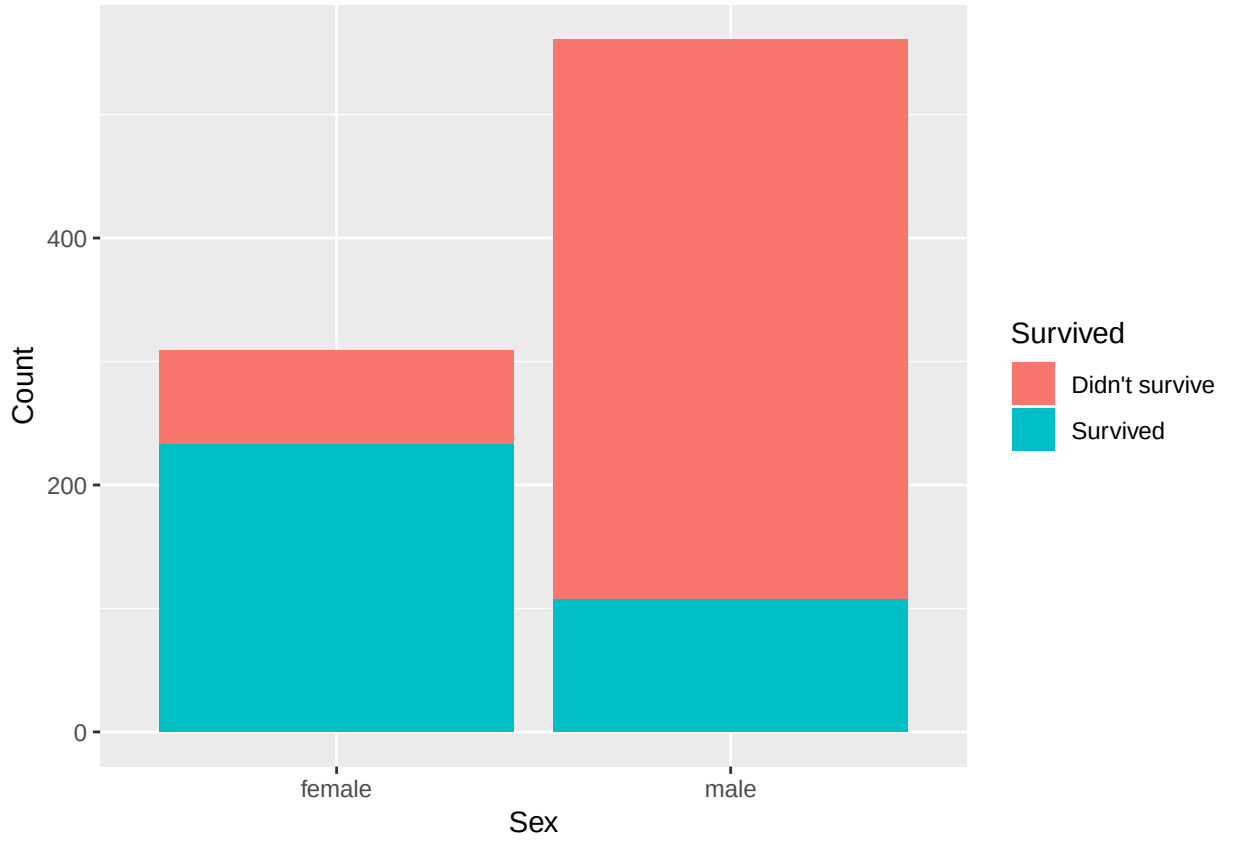
```
ggplot(df_sel_train, aes(Age, fill = Survived)) +
   geom_histogram(alpha = 0.4, aes(y = ..density..), position = 'identity')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Aprofitem per fer un repàs a les tasas de supervivència en funció de les diferents variables, amb gràfic i taula.

```
ggplot(df_sel_train, aes(Sex, fill=Survived)) + geom_bar(position="stack")+ylab("Count")
```



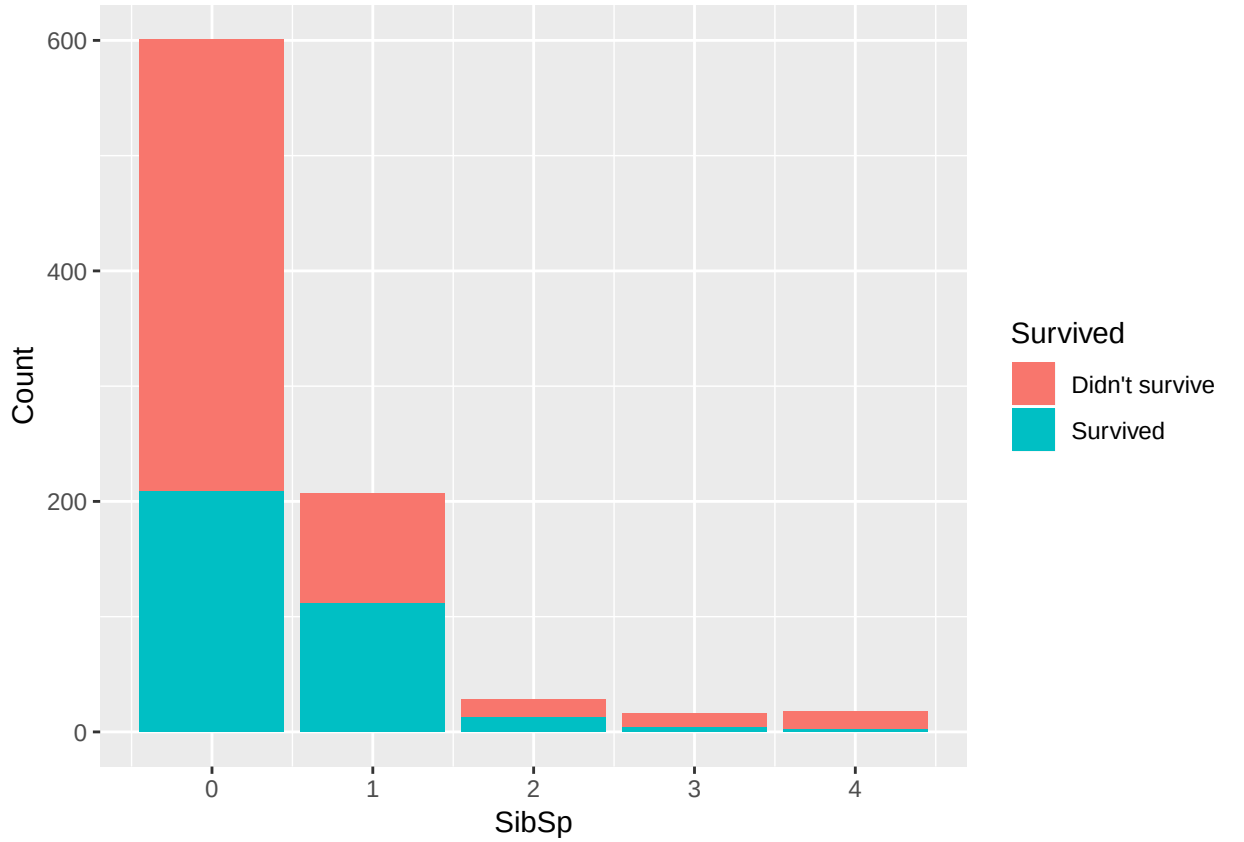

```
taula_SST <- table(df_sel_train$Sex, df_sel_train$Survived)
taula_SST
```

```
##
##          Didn't survive Survived
##   female             76      233
##   male              453      108
```

```
prop.table(taula_SST, margin = 1)
```

```
##
##          Didn't survive  Survived
##   female      0.2459547 0.7540453
##   male        0.8074866 0.1925134
```

```
ggplot(df_sel_train, aes(SibSp, fill=Survived)) + geom_bar(position="stack")+ylab("Count")
```



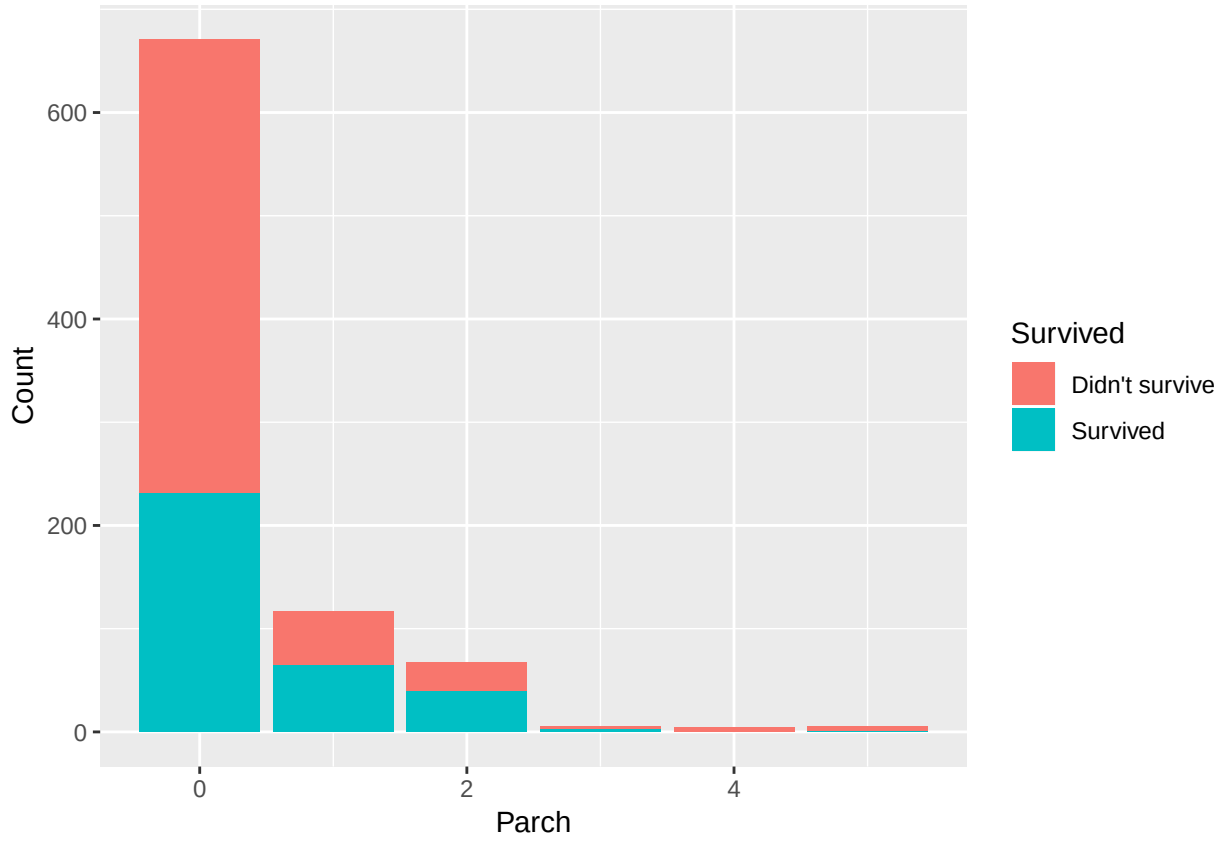

```
taula_SibST <- table(df_sel_train$SibSp, df_sel_train$Survived)
taula_SibST
```

```
##
##     Didn't survive Survived
##   0            392      209
##   1             95      112
##   2             15       13
##   3             12        4
##   4             15        3
```

```
prop.table(taula_SibST, margin = 1)
```

```
##
##     Didn't survive  Survived
##   0      0.6522463 0.3477537
##   1      0.4589372 0.5410628
##   2      0.5357143 0.4642857
##   3      0.7500000 0.2500000
##   4      0.8333333 0.1666667
```

```
ggplot(df_sel_train, aes(Parch, fill=Survived)) + geom_bar(position="stack")+ylab("Count")
```



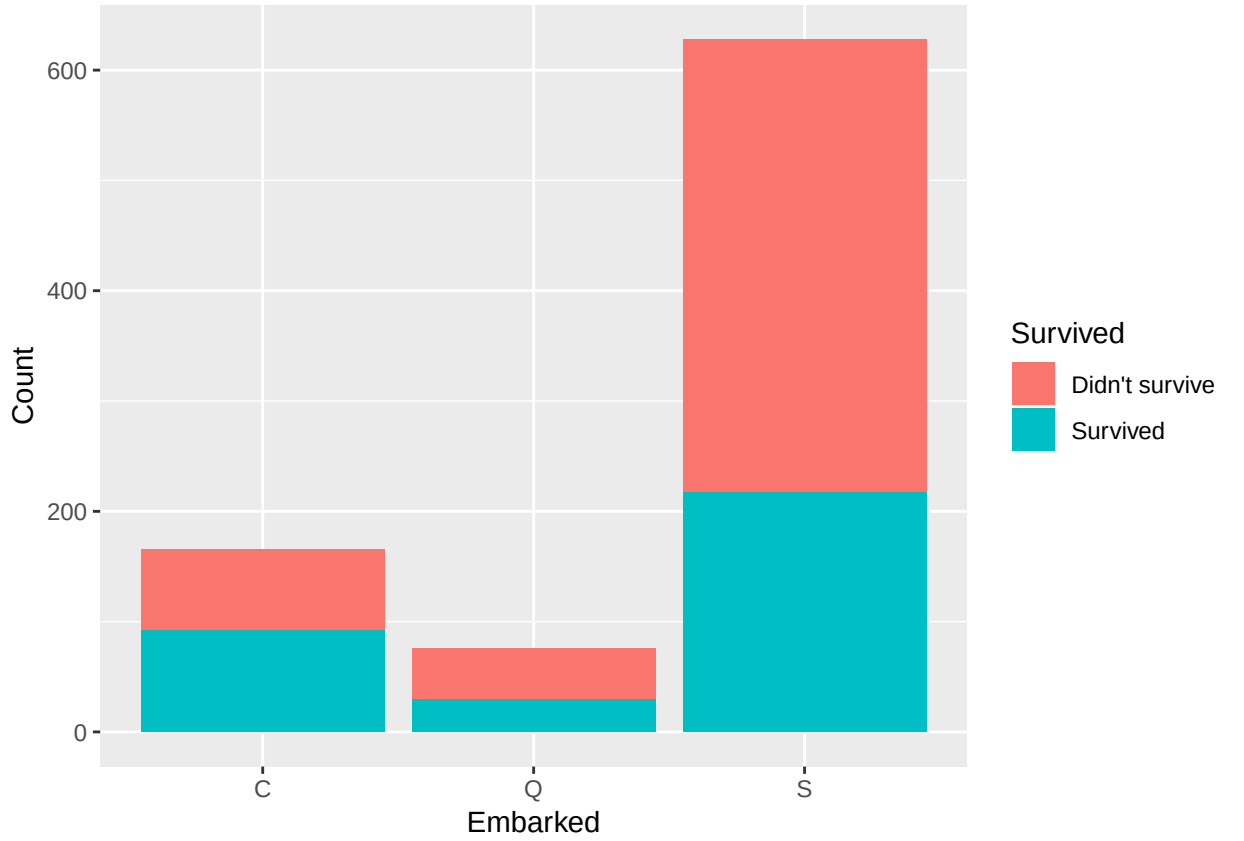

```
taula_PST <- table(df_sel_train$Parch, df_sel_train$Survived)
taula_PST
```

```
##
##     Didn't survive Survived
##   0            439      232
##   1             52       65
##   2             28       40
##   3              2        3
##   4              4        0
##   5              4        1
```

```
prop.table(taula_PST, margin = 1)
```

```
##
##     Didn't survive  Survived
##   0      0.6542474 0.3457526
##   1      0.4444444 0.5555556
##   2      0.4117647 0.5882353
##   3      0.4000000 0.6000000
##   4      1.0000000 0.0000000
##   5      0.8000000 0.2000000
```

```
ggplot(df_sel_train, aes(Embarked, fill=Survived)) + geom_bar(position="stack")+ylab("Count")
```



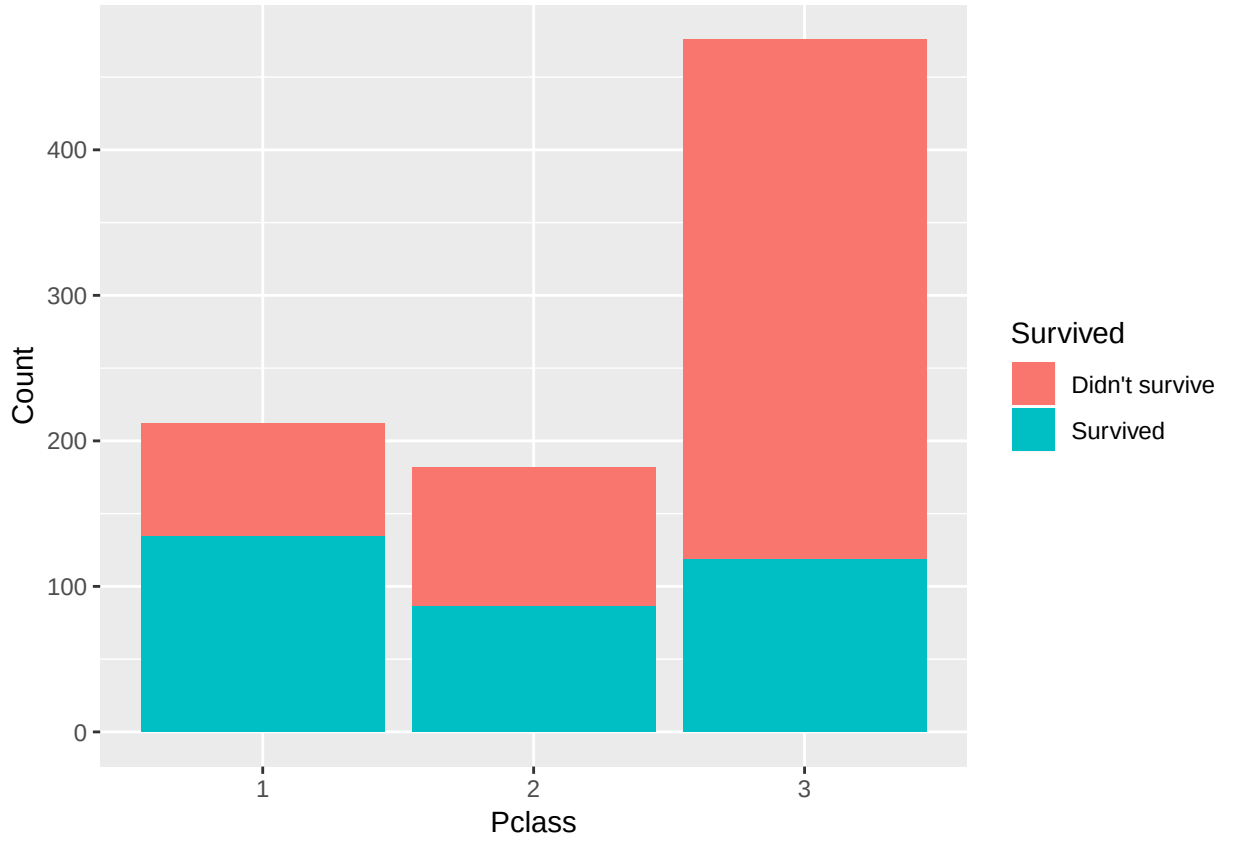

```
taula_EST <- table(df_sel_train$Embarked, df_sel_train$Survived)
taula_EST
```

```
##
##     Didn't survive Survived
##   C             73       93
##   Q             46       30
##   S            410      218
```

```
prop.table(taula_EST, margin = 1)
```

```
##
##     Didn't survive  Survived
##   C      0.4397590 0.5602410
##   Q      0.6052632 0.3947368
##   S      0.6528662 0.3471338
```

```
ggplot(df_sel_train, aes(Pclass, fill=Survived)) + geom_bar(position="stack")+ylab("Count")
```




```
taula_PcST <- table(df_sel_train$Pclass, df_sel_train$Survived)
taula_PcST
```

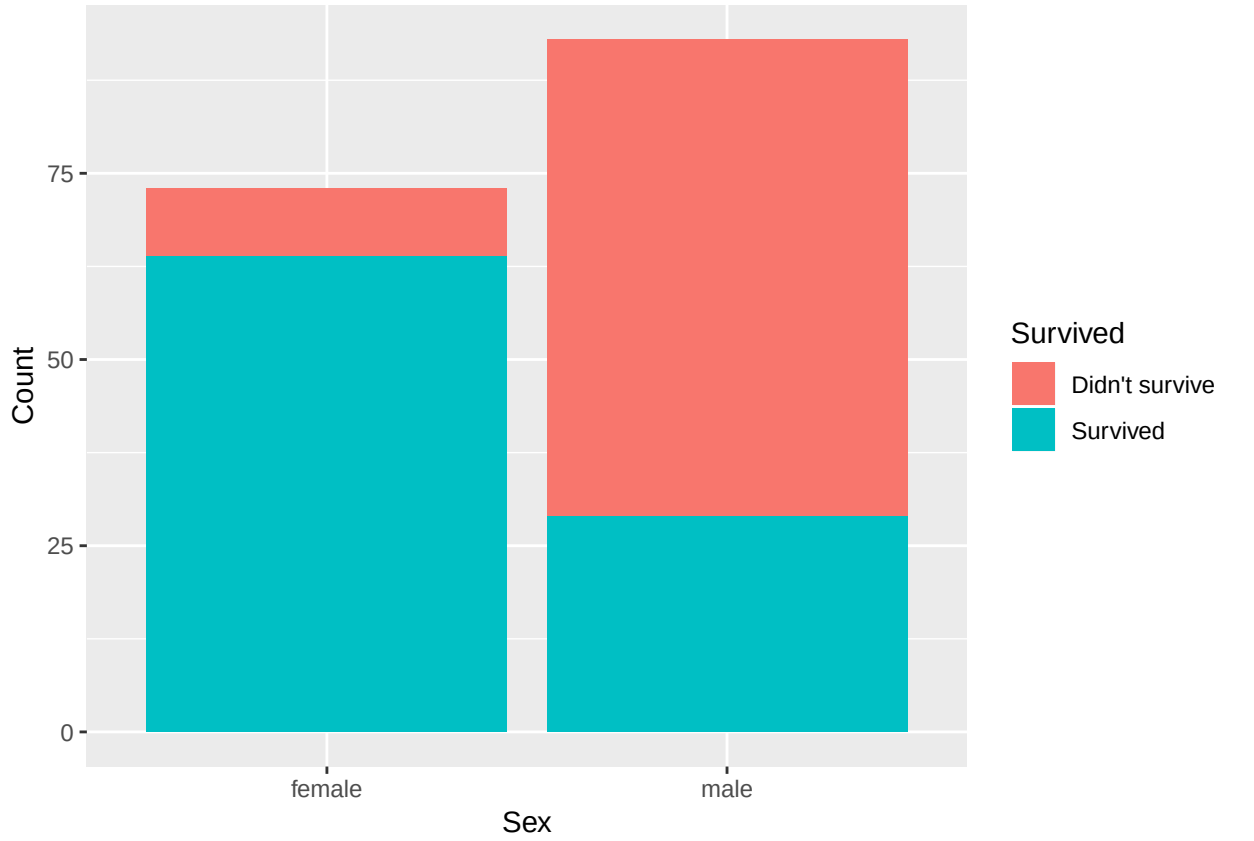
```
##
##     Didn't survive Survived
##   1             77      135
##   2             95       87
##   3            357      119
```

```
prop.table(taula_PcST, margin = 1)
```

```
##
##     Didn't survive  Survived
##   1      0.3632075 0.6367925
##   2      0.5219780 0.4780220
##   3      0.7500000 0.2500000
```

Sembla curiós el cas dels embarcats al port de Cherbourg que en principi tenien millor tasa de supervivència. Una possible explicació pot ser que en aquest port embarquessin significativament més persones de primera classe que a la resta, sent moltes d'elles dones.

```
ggplot(titanic.portC, aes(Sex, fill=Survived)) + geom_bar(position="stack")+ylab("Count")
```



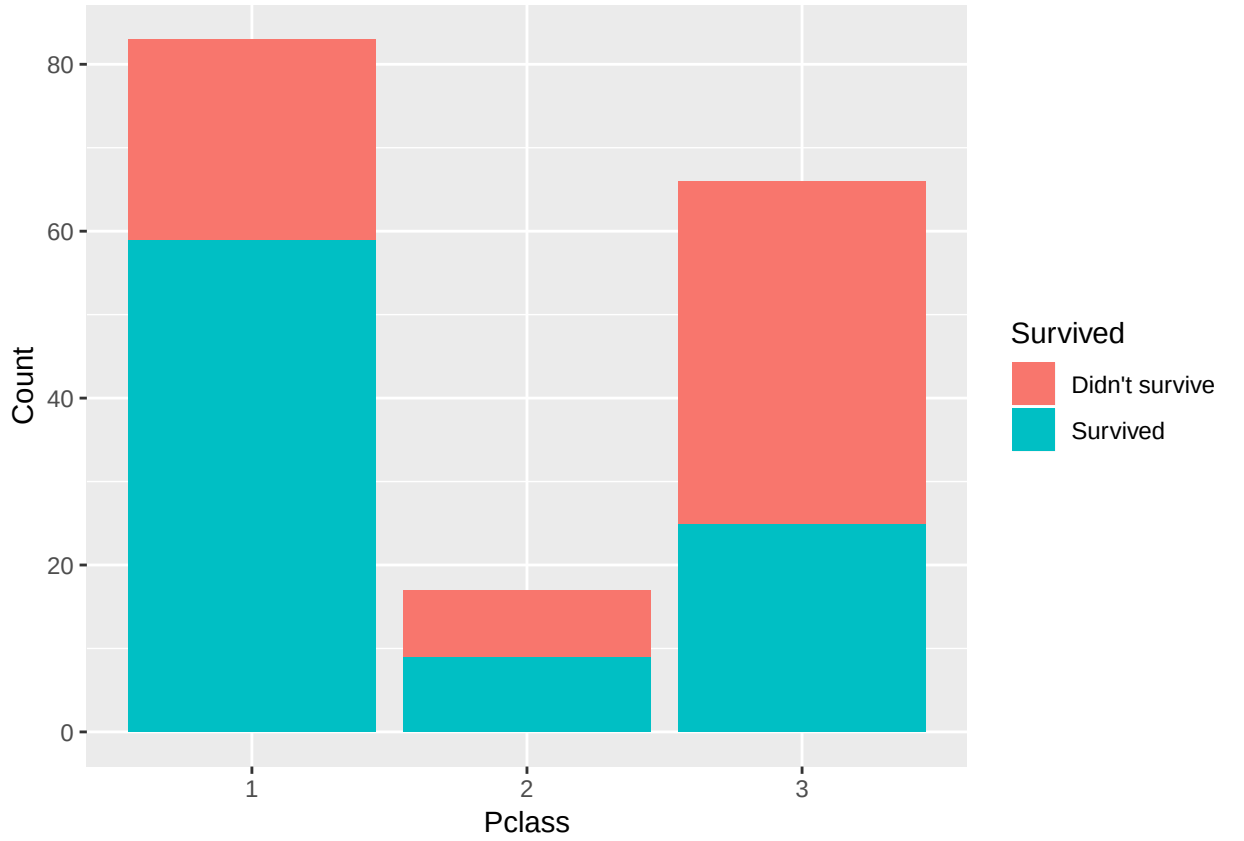

```
taula_portCSST <- table(titanic.portC$Sex, titanic.portC$Survived)
taula_portCSST
```

```
##
##          Didn't survive Survived
##   female              9       64
##   male               64       29
```

```
prop.table(taula_portCSST, margin = 1)
```

```
##
##          Didn't survive  Survived
##   female      0.1232877 0.8767123
##   male        0.6881720 0.3118280
```

```
ggplot(titanic.portC, aes(Pclass, fill=Survived)) + geom_bar(position="stack")+ylab("Count")
```




```
taula_portCPcST <- table(titanic.portC$Pclass, titanic.portC$Survived)
taula_portCPcST
```

```
##
##     Didn't survive Survived
##   1             24       59
##   2              8        9
##   3             41       25
```

```
prop.table(taula_portCPcST, margin = 1)
```

```
##
##     Didn't survive  Survived
##   1      0.2891566 0.7108434
##   2      0.4705882 0.5294118
##   3      0.6212121 0.3787879
```

6 Resolució del problema. A partir dels resultats obtinguts, quines són les conclusions? Els resultats permeten respondre al problema?

A partir dels resultats obtinguts, podem dir que un anàlisi detallat del conjunt de dades pot donar resposta a les preguntes plantejades, i a altres que no hem fet. Sabem que els factors amb més incidència per explicar la supervivència són el fet de ser dona i viatjar en primera classe. També hem vist que els que viatjaven sol van sobreviure menys que els que viatjaven en familia.

Contribucions	Signa
Recerca prèvia	VGD, IJL
Redacció de les respostes	VGD, IJL
Desenvolupament codi	VGD, IJL

Figure 1: Taula de contribucions

7 Taula de contribucions