Tipología y ciclo de vida de los datos

PRACTICA 2

Autor: Stela Diez y Silvia Ruiz

Mayo 2020

Contents

Descripción del dataset

Para la práctica hemos escogido el dataset con información sobre los pasajeros del Titanic, transatlántico hundido en abril del 1912 durante su viaje inaugural desde Southhampton a New York.

Nuestro fichero de datos (train.csv) contiene 891 registros/observaciones y 12 variables o atributos.

La información de los atributos es la siguiente:

- PassengerId: identificador de cada pasajero.
- Survived: si el pasajero sobrevivió al naufragio (0 = muere, 1 = sobrevive).
- Pclass: clase a la que pertenecía el pasajero (1 = 1st, 2 = 2nd o 3 = 3rd).
- Name: nombre del pasajero.
- Sex: sexo del pasajero.
- Age: edad del pasajero.
- SibSp: número de hermanos, hermanas, hermanastros, hermanastras, esposo o esposa, que se encuentran a bordo.
- Parch: número de padres e hijos que se encuentran a bordo.
- Ticket: identificador del billete.
- Fare: precio pagado por el billete.
- Cabin: identificador del camarote asignado al pasajero.
- Embarked: puerto en el que embarcó el pasajero (C= Cherbourg, Q = Queenstown, S = Southampton).

Como ya sabemos, una de las razones por las que hubo tantas muertes fue por la falta de botes salvavidas. Esto provocó la muerte de un gran número de pasajeros. Con este estudio, deseamos comprender el impacto que pudieron tener factores como la edad, el sexo o la clase (entre otros) en la supervivencia de los pasajeros.

Intentaremos predecir qué factores fueron los determinantes para la supervivencia y lo comprobaremos intentando predecir qué pasajeros sobrevivieron y cuáles no. Para ello, analizaremos cada variable de interés en relación con la variable Survived.

De este modo intentaremos identificar aquellas variables que están más relacionadas con la supervivencia e intentaremos responder a las siguientes preguntas:

(a) Tal y como siempre se ha contado, ¿los pasajeros de primera clase tuvieron más probabilidades de sobrevivir?.

Estudiaremos la relación entre las variables Survived vs Pclass.

(b) Fue realmente cierta la frase ¿mujeres y niños primero? ¿Los ancianos podrían haber tenido dificultades para alcanzar los botes?. Estudiaremos las variables Survived vs Age y Survived vs Sex.

En el caso de que esta afirmación fuese cierta, ¿todos los niños y todas las mujeres tuvieron las mismas oportunidades de sobrevivir o la clase en la que estaban embarcados influyó en su supervivencia?

Estudiaremos también las variables Survived vs Age vs Pclass y Survived vs Sex vs Pclass.

(c) ¿Haber pagado un billete caro te aseguraba un bote salvavidas?

Estudiaremos las variables Fare vs Survived y veremos también la relación de estas variables con la clase, Fare vs Survived vs Pclass.

(d) ¿El puerto de embarque pudo tener relación con la supervivencia? ¿Podría haber alguna relación entre el puerto de embarque y un mayor poder adquisitivo (relacionado por tanto con la clase)?

Estudiaremos la relación entre las variables Embarked vs Survived.

(e) ¿Tuvo más probabilidad de sobrevivir la gente que viajaba en familia que los que iban solos? ¿Existe alguna relación entre el número de familiares a bordo del buque y las probabilidades de sobrevivir?

Estudiaremos la relación entre las variables FamilySize (obtenida a partir de SibSp y Parch) vs Survived. También estudiaremos la relación entre estas variables y la clase (Survived vs FamilySize vs Pclass) y entre estas variables y el sexo (Survived vs FamilySize vs Sex).

NOTA: Aunque contamos también con los datos de test.csv, como en ese fichero no contamos con los valores objetivos del conjunto, no lo utilizaremos, ya que no nos permitiría evaluar la eficiencia.

Integración y selección de los datos de interés a analizar

Partimos de dos dataset (train y test) que hemos descargado de *https://www.kaggle.com/c/titanic/data*.

Lo primero que haremos será cargar las librerías que vamos a usar:

```
# Cargamos las librerías que vamos a usar
if(!require(dplyr)){
    install.packages('dplyr')
    library(dplyr)
}


if(!require(ggplot2)){
    install.packages('ggplot2', repos='http://cran.us.r-project.org')
    library(ggplot2)
}


if(!require(grid)){
    install.packages('grid', repos='http://cran.us.r-project.org')
    library(grid)
}


if(!require(gridExtra)){
    install.packages('gridExtra', repos='http://cran.us.r-project.org')
    library(gridExtra)
}


if(!require(C50)){
    install.packages('C50')
    library(C50)
}


if(!require(gmodels)){
```

```
    install.packages('gmodels', repos='http://cran.us.r-project.org')
    library(gmodels)
}


if(!require(NbClust)){
    install.packages('NbClust')
    library(NbClust)
}


if(!require(modeest)){
    install.packages('modeest')
    library(modeest)
}


if(!require(cluster)){
    install.packages('cluster')
    library(cluster)
}


if(!require(factoextra)){
    install.packages('factoextra')
    library(factoextra)
}


if(!require(corrplot)){
    install.packages('corrplot')
    library(corrplot)
}


if(!require(gplots)){
    install.packages('gplots')
    library(gplots)
}


if (!require(kableExtra)){
    install.packages("kableExtra", dependences=TRUE)
    library(kableExtra)
}
```

Después, cargaremos los datos que vamos a utilizar y comprobaremos la estructura de los mismos:

```
# Cargamos los datos descargados
test <- read.csv('test.csv',stringsAsFactors = FALSE)
train <- read.csv('train.csv', stringsAsFactors = FALSE)

# Vemos su estructura
```

```
str(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       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
##  $ Sex        : chr  "male" "female" "female" "female" ...
##  $ 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     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  "" "C85" "" "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
```

Como esperábamos tenemos 891 registros en 12 variables.

Eliminaremos las siguientes variables de nuestro estudio:

- Name, ya que siendo el nombre de los pasajeros, no nos aporta información relevante para nuestro estudio.
- Cabin, como en el caso anterior, el nombre del camarote no aporta información relevante para nuestro estudio.
- Ticket, del mismo modo, el número de ticket no aporta información relevante para nuestro estudio.

Crearemos la siguiente variable calculada:

- FamilySize (tamaño de familia), a partir de las vaiables SibSp y Parch, calcularemos cuantos miembros de una misma familia viajan juntos.

```
# Creamos el dataset que vamos a estudiar sin las variables que no nos interesan
data <- select(train, -Name, - Cabin, -Ticket)

# Creamos la variable FamilySize
data$FamilySize <- data$SibSp + data$Parch +1;

# Mostramos información de los datos finales
str(data)
```

```
## 'data.frame':    891 obs. of  10 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 ...
##  $ Sex        : chr  "male" "female" "female" "female" ...
##  $ 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 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
##  $ FamilySize : num  2 2 1 2 1 1 1 5 3 2 ...
```

Limpieza de los datos

Elementos vacios

Buscamos si hay elementos vacios en nuestros datos.

```
# Estadísticas de valores vacíos
colSums(is.na(data))
```

```
## PassengerId    Survived      Pclass         Sex         Age       SibSp
##           0           0           0           0         177           0
##       Parch        Fare    Embarked  FamilySize
##           0           0           0           0
```

```
colSums(data=="")
```

```
## PassengerId    Survived      Pclass         Sex         Age       SibSp
##           0           0           0           0          NA           0
##       Parch        Fare    Embarked  FamilySize
##           0           0           2           0
```

Como vemos las variables Embarked y Age tienen valores nulos. Tomaremos estrategias distintas:

Embarked

Imputaremos el valor de la moda para los datos desconocidos.

```
# Calculamos la moda para imputar ese valor a los datos desconocidos
mlv(data$Embarked, method = "mfv")[1]
```

```
## [1] "S"
```

Como podemos ver, al calcular la moda hemos obtenido el valor “S”. Por tanto, imputaremos el valor “S” en los datos desconocidos.

```
# Imputamos el valor "S" en los valores vacíos
data$Embarked[data$Embarked==""]="S"
```

Age

Como la variable Age es numerica, podemos imputar la media de la variable, para poder trabajar con esos datos.

```
# Imputamos el valor de la media de la variable "Age" en los valores vacios
data$Age[is.na(data$Age)] <- mean(data$Age,na.rm=T)
```

Comprobamos que hemos subsanado el problema:

```
# Estadísticas de valores vacíos
colSums(is.na(data))
```

```
## PassengerId    Survived      Pclass         Sex         Age       SibSp
##           0           0           0           0           0           0
##       Parch        Fare    Embarked  FamilySize
##           0           0           0           0
```

```
colSums(data=="")
```

```
## PassengerId    Survived      Pclass         Sex         Age       SibSp
##           0           0           0           0           0           0
##       Parch        Fare    Embarked  FamilySize
##           0           0           0           0
```

Vemos que, efectivamente, ya no aparece ningún elemento vacío.

Discretización

Buscamos las variables para las que tendría sentido discretizar.

```
# Buscamos los diferentes valores para las variables
apply(data,2, function(x) length(unique(x)))
```

```
## PassengerId    Survived      Pclass         Sex         Age       SibSp
##         891           2           3           2          89           7
##       Parch        Fare    Embarked  FamilySize
##           7         248           3           9
```

Como vemos las variables Survived, Pclass, Sex y Embarked tienen pocos valores diferentes, por lo que son muy buenas candidatas para la discretización.

```
# Discretizamos las variables mencionadas
cols<-c("Survived","Pclass","Sex","Embarked")
for (i in cols){
  data[,i] <- as.factor(data[,i])
}

# Después de los cambios, vemos de nuevo estructura del dataset
str(data)
```

```
## 'data.frame':    891 obs. of  10 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 1 2 2 ...
##  $ 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 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 3 1 3 3 3 2 3 3 3 1 ...
##  $ FamilySize : num  2 2 1 2 1 1 1 5 3 2 ...
```

Valores extremos

Se puede considerar un outlier, según el test de Tukey, a los valores que se encuentran a más de 1,5 veces el valor del rango intercuartílico.

En nuestro caso, las variables discretizadas, no son candidatas a tener valores extremos. Tampoco PassengerId, puesto que es una variable secuencial.

Por tanto, estudiaremos los valores extremos en: Age, SibSp, Parch, Fare y FamilySize

Age

```
# Visualizamos el boxplot
boxplot(data$Age, col = "orange", border = "brown", horizontal = TRUE, main = "Ouliers de Age")
```


```
# Mostramos los valores atípicos en una lista
cat("\nLista de valores atipicos:\n")
```

```
##
## Lista de valores atipicos:
```

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

```
##  [1]  2.00 58.00 55.00  2.00 66.00 65.00  0.83 59.00 71.00 70.50  2.00 55.50
## [13]  1.00 61.00  1.00 56.00  1.00 58.00  2.00 59.00 62.00 58.00 63.00 65.00
## [25]  2.00  0.92 61.00  2.00 60.00  1.00  1.00 64.00 65.00 56.00  0.75  2.00
```

```
## [37] 63.00 58.00 55.00 71.00  2.00 64.00 62.00 62.00 60.00 61.00 57.00 80.00
## [49]  2.00  0.75 56.00 58.00 70.00 60.00 60.00 70.00  0.67 57.00  1.00  0.42
## [61]  2.00  1.00 62.00  0.83 74.00 56.00
```

```
# Mostramos el valor máximo y mínimo
cat("\nValor máximo: ")
```

```
##
## Valor máximo:
```

```
max(data$Age)
```

```
## [1] 80
```

```
cat("\nValor mínimo: ")
```

```
##
## Valor mínimo:
```

```
min(data$Age)
```

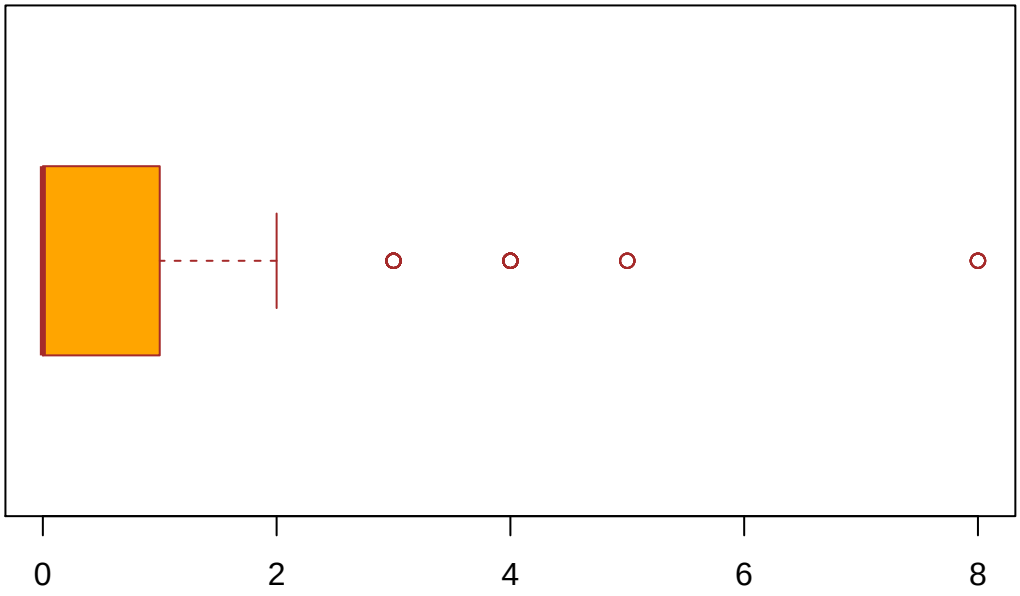
```
## [1] 0.42
```

Como vemos, hay muchos candidatos a ser valores atípicos, pero es perfectamente posible que una persona tenga una edad comprendida entre 0.42 y 80 años, por lo que realmente no son outliers y debemos mantener esos registros.

SibSp

```
#Visualizamos el boxplot
boxplot(data$SibSp, col = "orange", border = "brown", horizontal = TRUE, main = "Ouliers de SubSp")
```

Ouliers de SubSp




```
# Mostramos los valores atipicos en una lista
cat("\nLista de valores atipicos:\n")
```

```
##
## Lista de valores atipicos:
```

```
boxplot.stats(data$SibSp)$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
```

```
#Mostramos el valor máximo y mínimo
cat("\nValor máximo: ")
```

```
##
## Valor máximo:
```

```
max(data$SibSp)
```

```
## [1] 8
```

```
cat("\nValor mínimo: ")
```

```
##
## Valor mínimo:
```

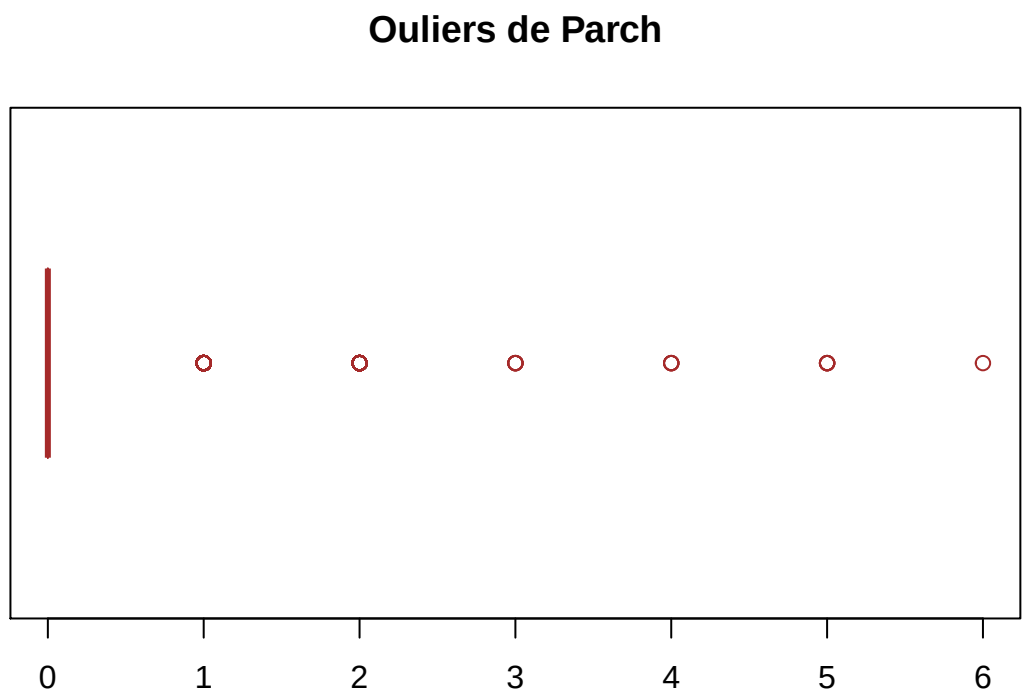
```
min(data$SibSp)
```

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

La variable SibSp mide el número de esposas o hermanos que hay embarcados, por lo que de nuevo, aunque tenemos muchos candidatos para ser valores atípicos, en 1910 no era descabellado que alguien tuviera 8 hermanos, por lo que no podemos descartar esos datos.

Parch

```
# Visualizamos el boxplot
boxplot(data$Parch, col = "orange", border = "brown", horizontal = TRUE, main = "Ouliers de Parch")
```




```
# Mostramos los valores atípicos en una lista
cat("\nLista de valores atipicos:\n")
```

```
##
## Lista de valores atipicos:
```

```
boxplot.stats(data$Parch)$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
```

```
# Mostramos el valor máximo y mínimo
cat("\nValor máximo: ")
```

```
##
## Valor máximo:
```

```
max(data$Parch)
```

```
## [1] 6
```

```
cat("\nValor mínimo: ")
```

```
##
## Valor mínimo:
```

```
min(data$Parch)
```

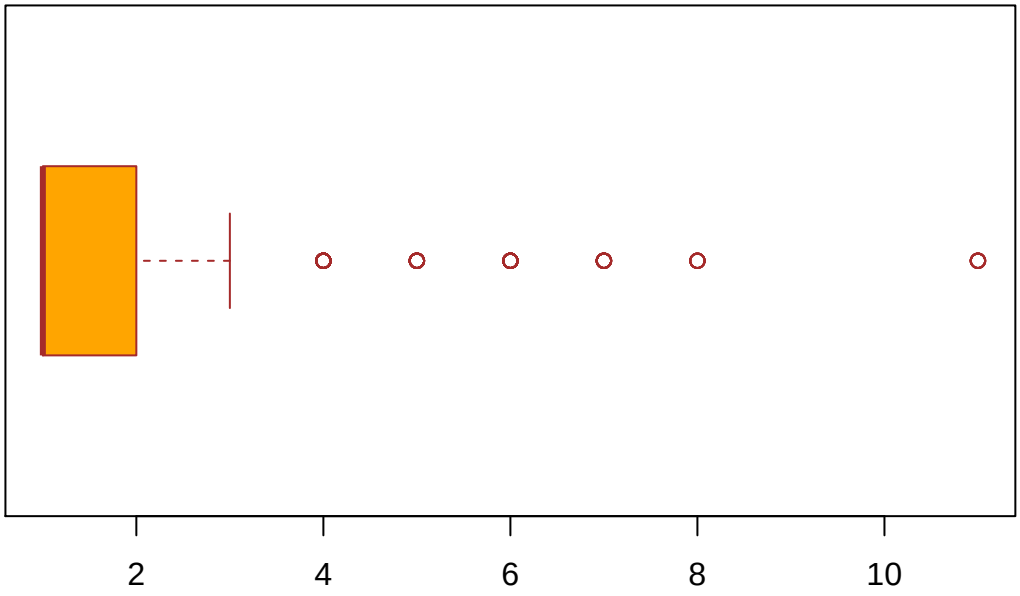
```
## [1] 0
```

En este caso, aunque claramente la gran mayoría viajaban sin hijos, de nuevo no podemos descartar que una familia viajara con 6 hijos.

FamilySize

```
# Visualizamos el boxplot
boxplot(data$FamilySize, col = "orange", border = "brown", horizontal = TRUE, main = "Outliers de Family
```

Ouliers de FamilySize




```
# Mostramos los valores atípicos en una lista
cat("\nLista de valores atipicos:\n")
```

```
##
## Lista de valores atipicos:
```

```
boxplot.stats(data$FamilySize)$out
```

```
##  [1]  5  7  6  5  7  6  4  6  4  8  6  7  8  4  5  6  4  7  5 11  6  6  6  5 11
## [26]  7  4 11  5  7  7  6  6  4  4  5 11  6  6  5  8  4  5  4  5  6  6  4  4  4
## [51]  4  8  5  4  4  7  7  5  4  4  7  4  4  6  6  6  4  8  8  4  6  4  5  5  4
## [76]  4  5  4  6  4 11  4  7  6  6 11  7  4 11  6  4
```

```
# Mostramos el valor máximo y mínimo
cat("\nValor máximo: ")
```

```
##
## Valor máximo:
```

```
max(data$FamilySize)
```

```
## [1] 11
```

```
cat("\nValor mínimo: ")
```

```
##
## Valor mínimo:
```

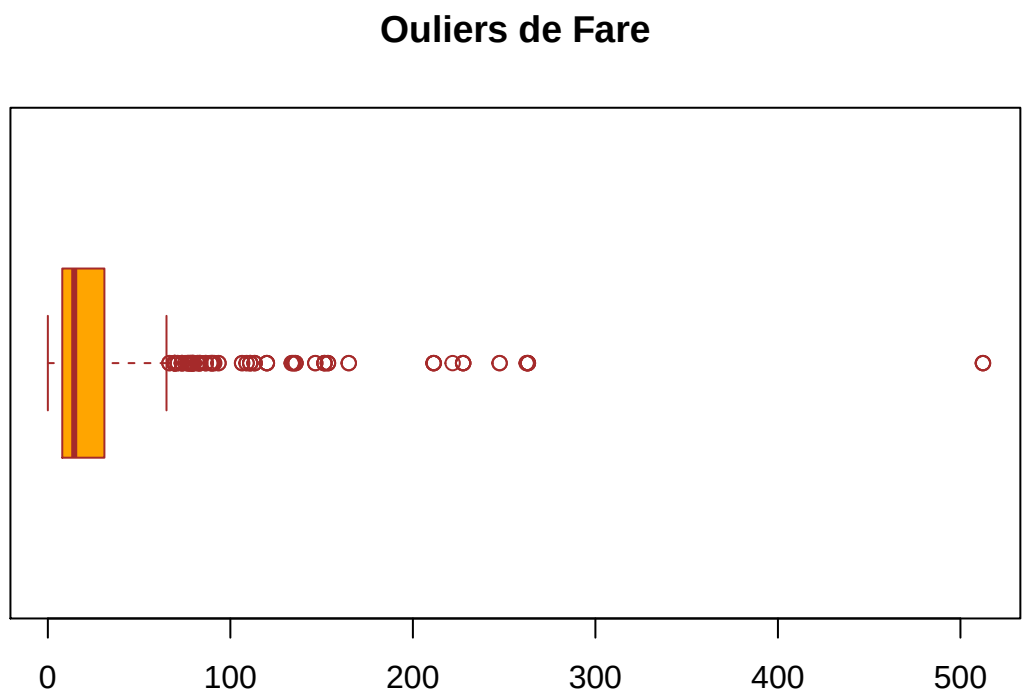
```
min(data$FamilySize)
```

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

De nuevo FamilySize no presenta valores que no puedan corresponderse con la realidad. Lo cual, teniendo en cuenta que es una variable calculada a partir de otras dos que no presentaban valores atípicos reales, era esperable.

Fare

```
# Visualizamos el boxplot
boxplot(data$Fare, col = "orange", border = "brown", horizontal = TRUE, main = "Ouliers de Fare")
```




```
# Mostramos los valores atípicos en una lista
cat("\nLista de valores atipicos:\n")
```

```
##
## Lista de valores atipicos:
```

```
boxplot.stats(data$Fare)$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 512.3292  79.6500 153.4625 135.6333  77.9583  78.8500
##  [33]  91.0792 151.5500 247.5208 151.5500 110.8833 108.9000  83.1583 262.3750
##  [41] 164.8667 134.5000  69.5500 135.6333 153.4625 133.6500  66.6000 134.5000
##  [49] 263.0000  75.2500  69.3000 135.6333  82.1708 211.5000 227.5250  73.5000
##  [57] 120.0000 113.2750  90.0000 120.0000 263.0000  81.8583  89.1042  91.0792
##  [65]  90.0000  78.2667 151.5500  86.5000 108.9000  93.5000 221.7792 106.4250
##  [73]  71.0000 106.4250 110.8833 227.5250  79.6500 110.8833  79.6500  79.2000
##  [81]  78.2667 153.4625  77.9583  69.3000  76.7292  73.5000 113.2750 133.6500
##  [89]  73.5000 512.3292  76.7292 211.3375 110.8833 227.5250 151.5500 227.5250
##  [97] 211.3375 512.3292  78.8500 262.3750  71.0000  86.5000 120.0000  77.9583
## [105] 211.3375  79.2000  69.5500 120.0000  93.5000  80.0000  83.1583  69.5500
## [113]  89.1042 164.8667  69.5500  83.1583
```

```
# Mostramos el valor máximo y mínimo
cat("\nValor máximo: ")
```

```
##
## Valor máximo:
```

```
max(data$Fare)
```

```
## [1] 512.3292
```

```
cat("\nValor mínimo: ")
```

```
##
## Valor mínimo:
```

```
min(data$Fare)
```

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

Vemos que tenemos muchos candidatos a valores extremos. Comprobamos también que hay muchos valores de Fare que están entre 100 y 200 y aunque sean candidatos a outliers, parecen cuantías razonables para los pasajeros de primera clase, por lo que vamos a anilizar los datos a partir del coste de 200.

```
# Filtramos los datos y los ordenamos por Fare
aux <- filter(data, Fare>=200)
aux[order(aux$Fare), ]
```

```
##    PassengerId Survived Pclass    Sex      Age SibSp Parch     Fare Embarked
## 14         690        1      1 female 15.00000     0     1 211.3375        S
## 17         731        1      1 female 29.00000     0     0 211.3375        S
## 20         780        1      1 female 43.00000     0     1 211.3375        S
## 8          378        0      1   male 27.00000     0     2 211.5000        C
```

```
## 11         528        0      1   male 29.69912     0     0 221.7792        S
## 9          381        1      1 female 42.00000     0     0 227.5250        C
## 12         558        0      1   male 29.69912     0     0 227.5250        C
## 15         701        1      1 female 18.00000     1     0 227.5250        C
## 16         717        1      1 female 38.00000     0     0 227.5250        C
## 3          119        0      1   male 24.00000     0     1 247.5208        C
## 5          300        1      1 female 50.00000     0     1 247.5208        C
## 6          312        1      1 female 18.00000     2     2 262.3750        C
## 19         743        1      1 female 21.00000     2     2 262.3750        C
## 1           28        0      1   male 19.00000     3     2 263.0000        S
## 2           89        1      1 female 23.00000     3     2 263.0000        S
## 7          342        1      1 female 24.00000     3     2 263.0000        S
## 10         439        0      1   male 64.00000     1     4 263.0000        S
## 4          259        1      1 female 35.00000     0     0 512.3292        C
## 13         680        1      1   male 36.00000     0     1 512.3292        C
## 18         738        1      1   male 35.00000     0     0 512.3292        C
##    FamilySize
## 14          2
## 17          1
## 20          2
## 8           3
## 11          1
## 9           1
## 12          1
## 15          2
## 16          1
## 3           2
## 5           2
## 6           5
## 19          5
## 1           6
## 2           6
## 7           6
## 10          6
## 4           1
## 13          2
## 18          1
```

Como esperabamos todos corresponden a primera clase. Si investigamos un poco (https://money.com/titanic-most-expensive-ticket/) para ver el precio del billete podemos ver que 512, no es uno de los más caros. Por lo que tampoco podemos eliminar estos registros por valores atípicos.

Exportación de datos limpios

Una vez tenemos los datos limpios procedemos a guardarlos en un fichero csv.

```
# Exportación de los datos limpios
write.csv(data, "Titanic_train_clean.csv")
```

Analisis de los datos

Planificación

En el primer apartado se han mencionado todas las variables que se pretende estudiar con el fin de responder a una serie de preguntas, para ello realizaremos distintos tests y enfrentaremos diferentes variables:

En primer lugar, estudiaremos la normalidad y la homogeineidad de la varianza para nuestras variables cuantitativas.

En segúndo lugar, aplicaremos diferentes métodos de análisis:

- Supervisados: un arbol de decisión y una regresión logística
- No supervisados: clustering
- Correlación

Finalmente, representaremos enfrentadas las variables en tablas y gráficas:

- Survived vs Pclass
- Survived vs Age
- Survived vs Sex
- Survived vs Fare
- Survived vs Embarked
- Survived vs FamilySize
- Survived vs Age vs Pclass
- Survived vs Age vs Sex
- Survived vs Sex vs Class
- Survived vs Fare vs Class
- Survived vs FamilySize vs Class
- Survived vs FamilySize vs Sex

Normalidad y homogeneidad de la varianza

Normalidad

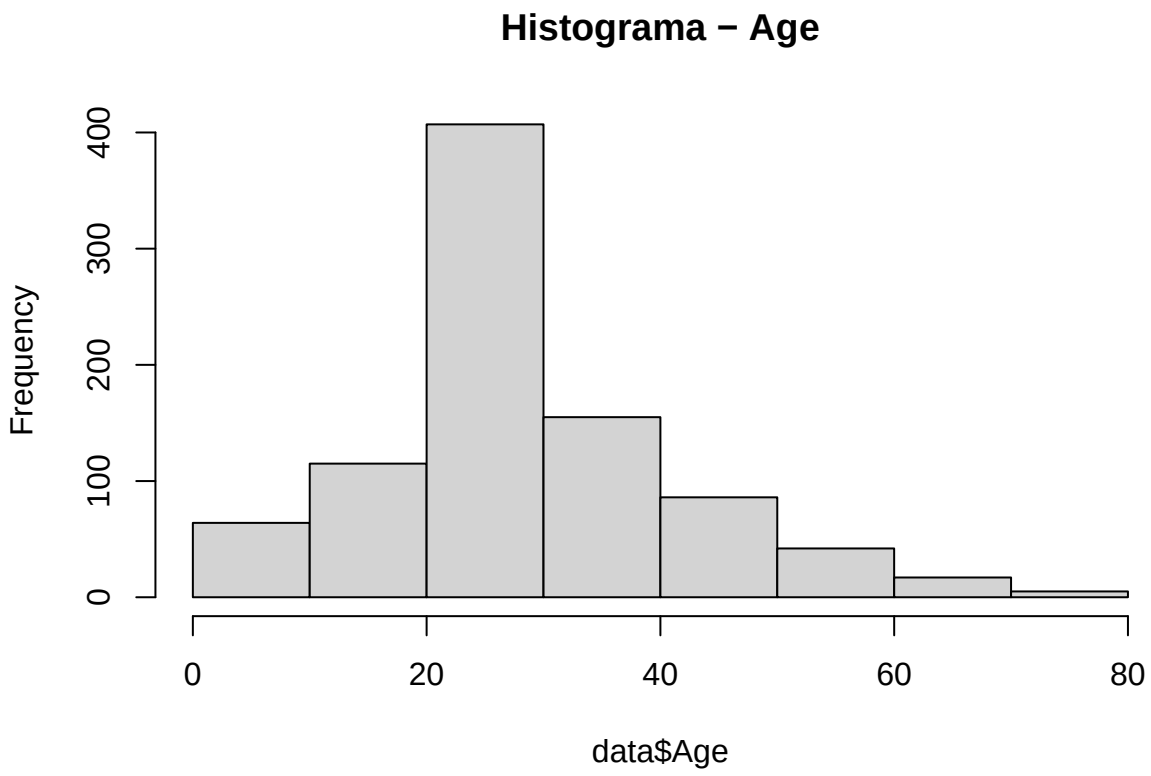
En primer lugar vamos a llevar a cabo un análisis de normalidad de las variables numéricas (Age, Fare y FamilySize).

Para hacernos una idea de si nuestras variables siguen una distribución normal haremos uso del histograma y el gráfico Q-Q (gráfico de cuantiles teóricos). A continuación, con el objetivo de verificar si existe o no normalidad en los datos aplicaremos el test de Shapiro-Wilk.

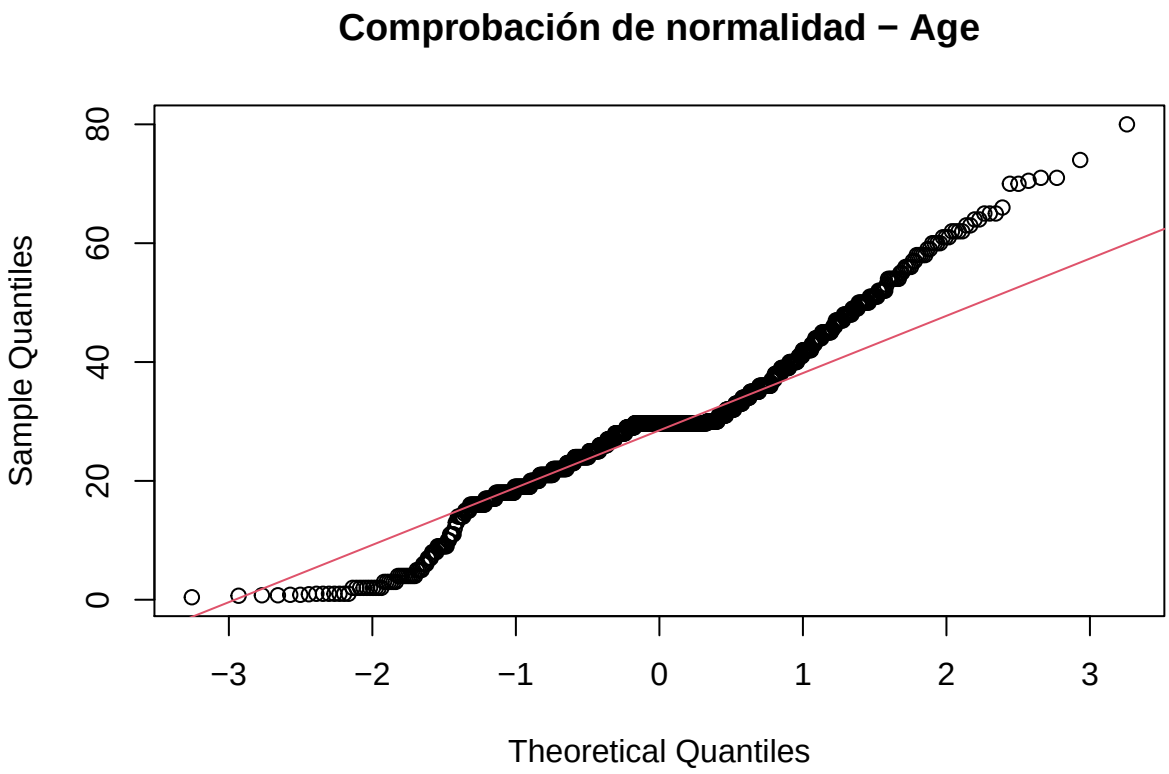
Para el test Shapiro-Wilk la hipótesis nula es que los datos se distribuyen normalmente. Por lo tanto si podemos rechazar la hipotesis nula, sabremos sin lugar a dudas que los datos no siguen una normal. Si por el contrario, no podemos rechazarla, no podremos asegurar que los datos sigan esta distribución.

Age

```
# Histograma
hist(data$Age, main="Histograma - Age")
```




```
# Gráfico Q-Q
qqnorm(data$Age, main="Comprobación de normalidad - Age")
qqline(data$Age, col=2)
```



Aunque la normalidad deberá comprobarse mediante el test estadístico, en el histograma la variable Age si parece seguir una distribución normal. Sin embargo, en el gráfico Q-Q vemos una desviación bastante pronunciada, respecto a la normal (recta roja), especialmente en los extremos.

Aplicamos el test Shapiro-Wilk para ver si realmente los datos siguen una distribución normal.

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

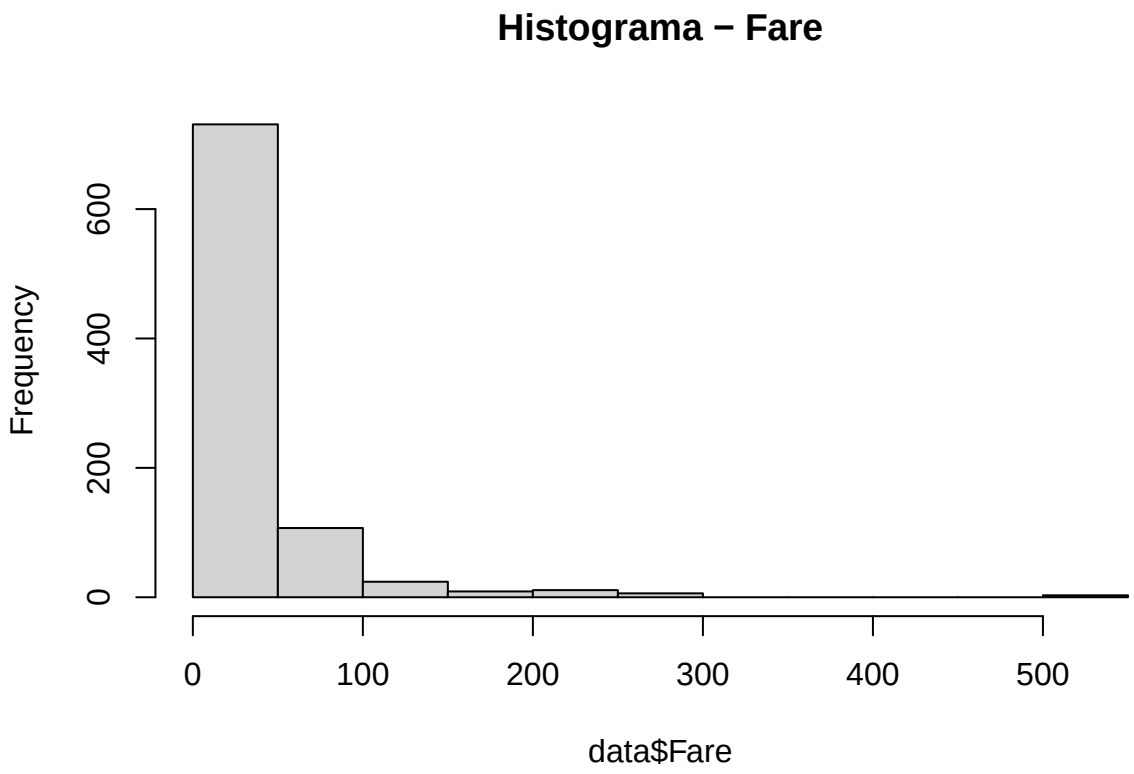
```
##
##  Shapiro-Wilk normality test
##
## data:  data$Age
## W = 0.95882, p-value = 3.969e-15
```

Como podemos observar al aplicar el test, el valor de $p < 0.05$. Como el p-valor es menor al nivel de significancia, podemos rechazar la hipótesis nula y por lo tanto, estamos seguros de que los datos no siguen una distribución normal.

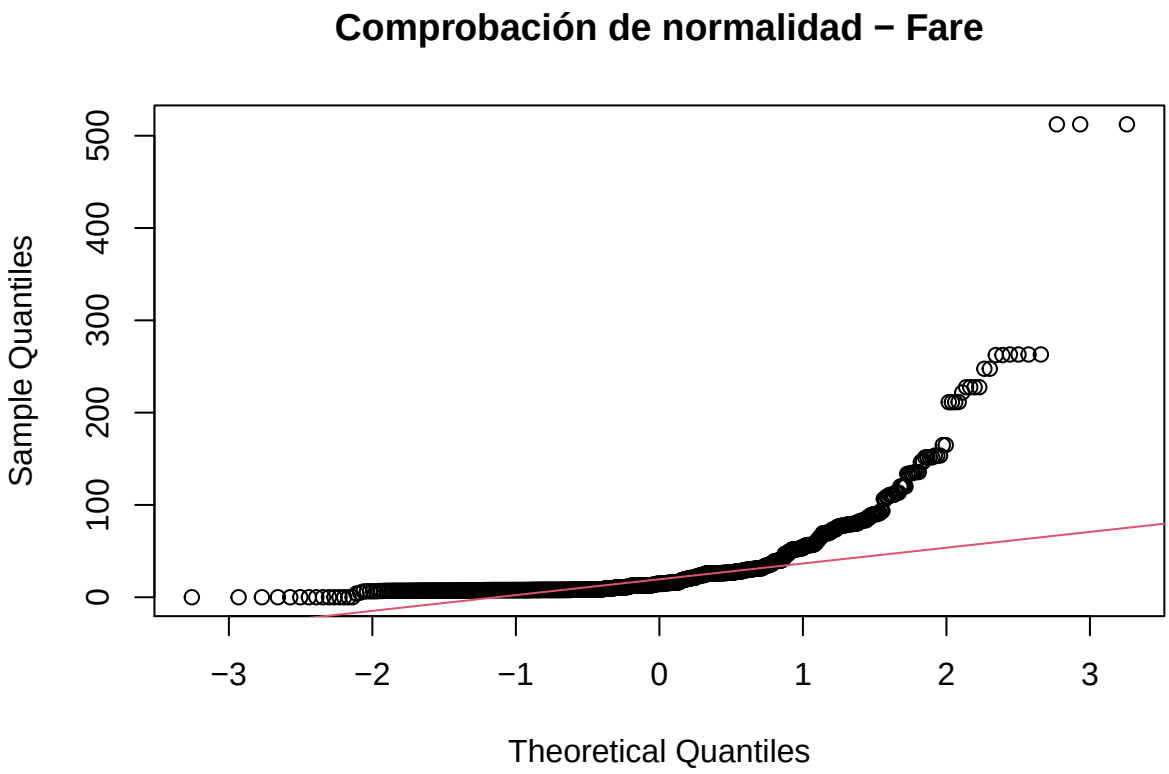
Fare

Obtenemos el histograma y el gráfico Q-Q.

```
# Histograma
hist(data$Fare, main="Histograma - Fare")
```




```
# Gráfico Q-Q
qqnorm(data$Fare, main="Comprobación de normalidad - Fare")
qqline(data$Fare, col=2)
```



En este caso, la variable Fare no parece presentar una distribución normal en ninguna de las gráficas. En el gráfico Q-Q al igual que ocurría en el caso anterior vemos como los datos se desvían mucho de la normal en los valores extremos.

Lo verificamos con el test estadístico.

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

```
##
##  Shapiro-Wilk normality test
##
## data:  data$Fare
## W = 0.52189, p-value < 2.2e-16
```

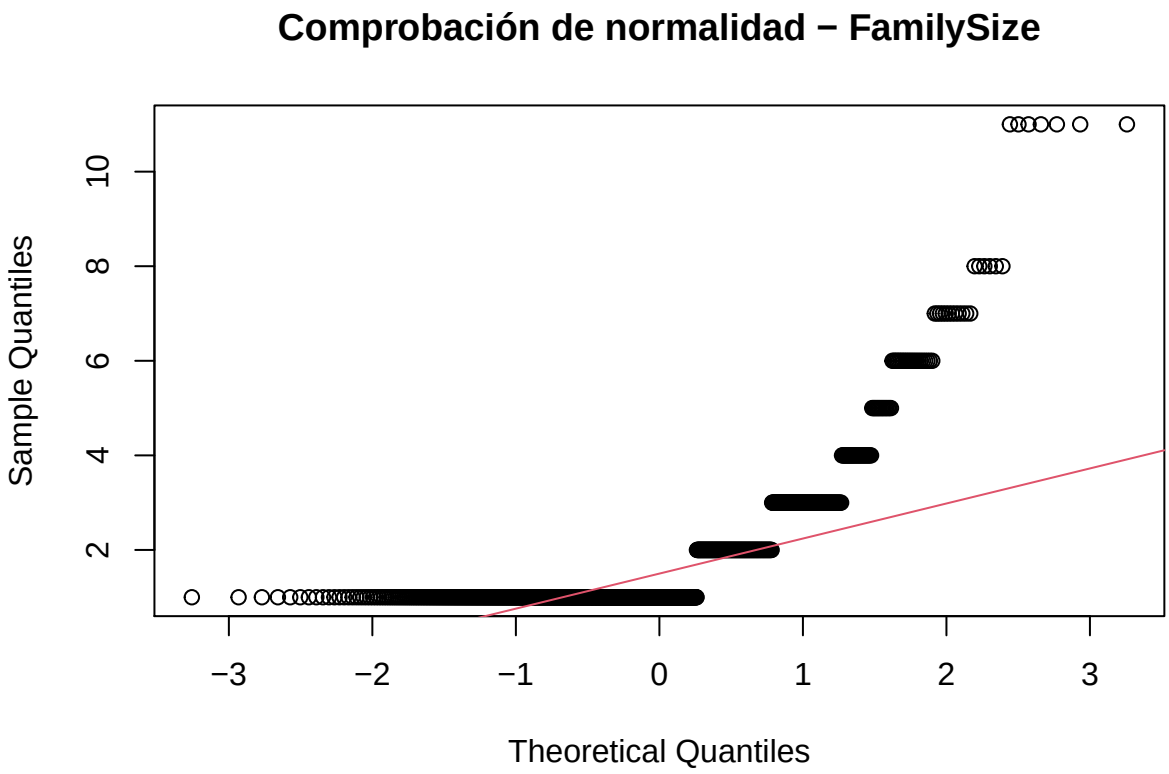
Tal y como se esperaba el valor de $p < 0.05$. Como el p-valor es menor al nivel de significancia se concluye que los datos no cuentan con una distribución normal.

FamilySize

Obtenemos el histograma y el gráfico Q-Q.

```
# Histograma
hist(data$FamilySize, main="Histograma - FamilySize")
```


```
# Gráfico Q-Q
qqnorm(data$FamilySize, main="Comprobación de normalidad - FamilySize")
qqline(data$FamilySize, col=2)
```



En este caso, la variable FamilySize tampoco parece presentar una distribución normal en ninguna de las gráficas. Lo verificamos con el test estadístico.

```
shapiro.test(data$FamilySize)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$FamilySize
## W = 0.61508, p-value < 2.2e-16
```

Tal y como se esperaba el valor de $p < 0.05$. Por tanto, los datos no cuentan con una distribución normal.

Homogenidad de la varianza

Una vez estudiada la normalidad de las variables numéricas, vamos a analizar si hay homogeneidad de la varianza.

Este estudio también lo realizaremos sobre las variables numéricas: Age, Fare y FamilySize.

Ya que nuestros datos no siguen una ditribución normal, para comprobar la homogeneidad de la varianza usaremos el test de Fligner-Killeen que se trata de una alternativa no paramétrica del test de Levene (que se usa cuando la distribución de datos es normal).

En el test de Fligner-Killeen la hipótesis nula asume igualdad de varianzas en los diferentes grupos de datos, por lo que p-valores inferiores al nivel de significancia indicarán heterocedasticidad, heterogeneidad

en las varianzas. Por lo que si rechazamos la homogeneidad en las varianzas, estaremos seguros de que el efecto de heterocedasticidad se produce; mientras que si no podemos rechazarlas no podremos asegurar la homogeneidad.

Age vs Survived

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

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Age by Survived
## Fligner-Killeen:med chi-squared = 5.4227, df = 1, p-value = 0.01988
```

Vemos que del test se obtiene un p-valor inferior al nivel de significancia ($< 0{,}05$). Por tanto, se rechaza la hipótesis nula de homocedasticidad y se concluye que la variable Age presenta varianzas estadísticamente diferentes para los diferentes grupos de Survived.

Fare

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

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Fare by Survived
## Fligner-Killeen:med chi-squared = 96.253, df = 1, p-value < 2.2e-16
```

Al igual que en el caso anterior se obtiene un $p < 0.05$. Por tanto, se rechaza la hipótesis nula y se concluye que la variable Fare presenta varianzas estadísticamente diferentes para los diferentes grupos de Survided.

FamilySize

```
fligner.test(FamilySize ~ Survived, data = data)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  FamilySize by Survived
## Fligner-Killeen:med chi-squared = 19.647, df = 1, p-value = 9.317e-06
```

Al igual que en los casos anteriores se obtiene un $p < 0.05$. Por tanto, se rechaza la hipótesis nula y se concluye que la variable FamilySize también presenta varianzas estadísticamente diferentes.

Analisis: Método supervisado: Arbol de decisión

Crearemos un arbol de decisión para evaluar si los pasajeros del Titanic sobreviven o no a la tragedia.

Para ello, tenemos que separar los datos de entrenamiento en 4 conjuntos:

- trainX, contendrá todas las variables menos la variable objetivo: Survived; y contendrá 2/3 partes de las observaciones.

- trainy, contendrá la variable objetivo: Survived; y contendrá 2/3 partes de las observaciones.
- testX, contendrá todas las variables menos la variable objetivo: Survived; y contendrá 1/3 partes de las observaciones.
- testy, contendrá la variable objetivo: Survived; y contendrá 1/3 partes de las observaciones.

NOTA: Aunque tenemos por separado los datos de test.csv, como no tenemos los valores objetivos del conjunto de test.csv, no podríamos evaluar la eficiencia de nuestro árbol de decisión; por lo tanto, tenemos que dividir el conjunto de los datos train.csv en dos grupos, uno para entrenar el árbol y otro para testearlo.

```
# Separamos los datos de entrenamiento en dos conjuntos
set.seed(666)
y <- select(data, Survived)
X <- select(data, -Survived, -PassengerId, -SibSp, -Parch)

# Para simplificar el arbol de decisión discretizaremos las variables numéricas:
# Variable Age
X$Age_cat <- X$Age
levels(X$Age_cat) <- c(levels(X$Age_cat), "Kid", "teenager", "adult", "old person")
X[X$Age < 13, "Age_cat"] <- "kid"
X[X$Age >= 13 & X$Age < 20, "Age_cat"] <- "teenager"
X[X$Age >= 20 & X$Age < 65, "Age_cat"] <- "adult"
X[X$Age >= 65, "Age_cat"] <- "old person"

# Variable Fare
X$Fare_cat <- X$Fare
levels(X$Fare_cat) <- c(levels(X$Fare_cat), "Cheapest", "Medium", "Expensive")
X[X$Fare < 50, "Fare_cat"] <- "Cheapest"
X[X$Fare >= 50 & X$Fare < 100, "Fare_cat"] <- "Medium"
X[X$Fare >= 100, "Fare_cat"] <- "Expensive"

# Variable FamilySize
X$FamilySize_cat <- X$Fare
levels(X$FamilySize_cat) <- c(levels(X$FamilySize_cat), "Alone", "Pair", "Team")
X[X$FamilySize == 1, "FamilySize_cat"] <- "Alone"
X[X$FamilySize == 2, "FamilySize_cat"] <- "Pair"
X[X$FamilySize > 2, "FamilySize_cat"] <- "Team"

# Discretizamos las variables
cols<-c("Age_cat","Fare_cat","FamilySize_cat")
for (i in cols){
  X[,i] <- as.factor(X[,i])
}

# Eliminamos las variables originales
X <- select(X, -Age, -Fare, -FamilySize)

# Separamos en los 4 conjuntos objetivo
indexes = sample(1:nrow(data), size=floor(2/3*nrow(data)))

trainX <- X[indexes,]
trainy <- y[indexes,"Survived"]
testX <- X[-indexes,]
testy <- y[-indexes,"Survived"]
```

Ahora crearemos un modelo a partir de los datos de entrenamiento:

```
# Creamos y mostramos las caracteristicas del árbol
tree_mod <- C50::C5.0(trainX, trainy, rules=TRUE )
summary(tree_mod)
```

```
##
## Call:
## C5.0.default(x = trainX, y = trainy, rules = TRUE)
##
##
## C5.0 [Release 2.07 GPL Edition]      Thu May 21 21:52:06 2020
## -------------------------------
##
## Class specified by attribute 'outcome'
##
## Read 594 cases (7 attributes) from undefined.data
##
## Rules:
##
## Rule 1: (374/59, lift 1.3)
##  Sex = male
##  Age_cat in {adult, old person, teenager}
##  ->  class 0  [0.840]
##
## Rule 2: (323/73, lift 1.2)
##  Pclass = 3
##  ->  class 0  [0.772]
##
## Rule 3: (106/5, lift 2.6)
##  Pclass in {1, 2}
##  Sex = female
##  ->  class 1  [0.944]
##
## Rule 4: (10, lift 2.5)
##  Pclass in {1, 2}
##  Sex = male
##  Age_cat = kid
##  ->  class 1  [0.917]
##
## Default class: 0
##
##
## Evaluation on training data (594 cases):
##
##          Rules
##    ----------------
##      No      Errors
##
##       4  112(18.9%)   <<
##
##
##     (a)   (b)    <-classified as
##    ----  ----
```

```
##     371     5    (a): class 0
##     107   111    (b): class 1
##
##
##  Attribute usage:
##
##   82.49% Sex
##   73.91% Pclass
##   64.65% Age_cat
##
##
## Time: 0.0 secs
```

Errors muestra el número y porcentaje de casos mal clasificados en el conjunto de entrenamiento. El árbol obtenido clasifica erróneamente 108 de los 594 casos, una tasa de error del 18.2%.

Podemos observar las reglas de clasificación que hemos extraido:

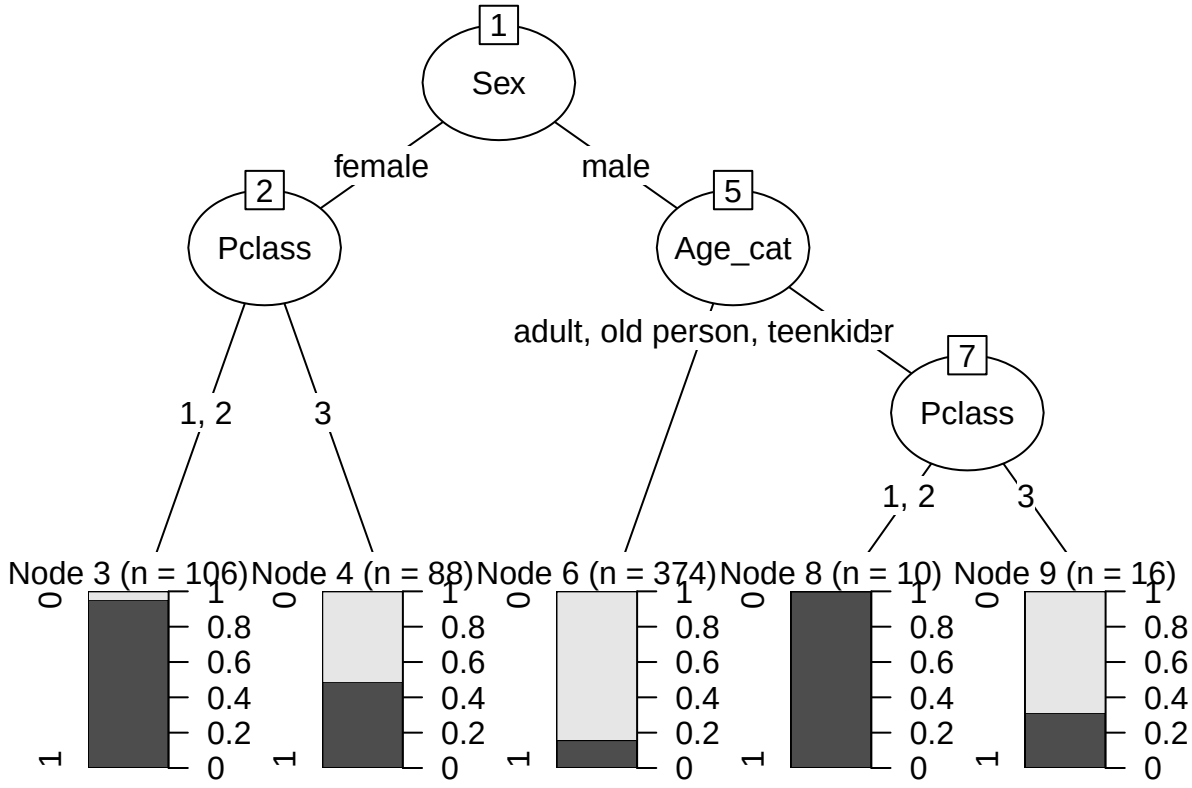
- Regla 1: los hombres, que no son niños, mueren con una validez del 82,4%.
- Regla 2: Las personas que iban en 3ª clase en familia mueren con una validez del 75,6%.
- Regla 3: Las mujeres de 1ª y 2ª clase sobrevivieron con una validez del 93,4%
- Regla 4: Las mujeres que iban solas o con otra persona, sobrevivieron con una validez del 80,4%
- Regla 5: los niños varones, sobrevivieron con una validez del 58,3%
- La clasificación por defecto es: muere.

Por lo que podemos concluir que las mujeres y los niños sí tuvieron prioridad a la hora de salvarse.

Es muy interesante también, fijarse en la importancia de las variables en nuestro árbol de decisión: la variable más importante es el sexo, después la edad, luego la cantidad de personas con las que se viajaba y por último la clase.

Mostramos gráficamente nuestro arbol de decisión.

```
tree_mod_2 <- C50::C5.0(trainX, trainy)
plot(tree_mod_2)
```



Procedemos ahora a la validación de nuestro árbol de decisión. Para ello utilizaremos los conjuntos de test que hemos creado a partir de los datos train.csv.

```
# Realizamos una predicción de los datos
predicted_model <- predict( tree_mod_2, testX, type="class" )

# Comparamos con los datos reales, para poder calcular la precisión
print(sprintf("La precisión del árbol es: %.4f %%",100*sum(predicted_model == testy) / length(predicted
```

```
## [1] "La precisión del árbol es: 77.7778 %"
```

Para evaluar mejor los errores de nuestro arbol de decisión utilizamos la función crossTable:

```
CrossTable(testy, predicted_model,prop.chisq  = FALSE, prop.c = FALSE, prop.r =FALSE,dnn = c('Reality',
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  297
```

```
##
##
##               | Prediction
##      Reality |         0 |         1 | Row Total |
## -------------|-----------|-----------|-----------|
##            0 |       169 |         4 |       173 |
##              |     0.569 |     0.013 |           |
## -------------|-----------|-----------|-----------|
##            1 |        62 |        62 |       124 |
##              |     0.209 |     0.209 |           |
## -------------|-----------|-----------|-----------|
## Column Total |       231 |        66 |       297 |
## -------------|-----------|-----------|-----------|
##
##
```

Como podemos ver:

- El porcentaje de predicciones correctas es del 80,5%.
- El porcentaje de errores tipo I (falso positivo) es del 12,1%.
- El porcentaje de errores tipo II (falso negativo) es del 7,4%.

Análisis: Método supervisado: Regresión logística

La regresión logística se utiliza para problemas de clasificación, como el que tenemos aquí. Por lo tanto, aplicaremos la regresión logística para saber si las personas del conjunto test.csv sobrevivieron o no a la catástofre.

```
# Seleccionamos los datos con los que vamos a trabajar
datos_reg <- select(data, Survived, Pclass, Sex, Age)
summary(datos_reg)
```

```
##  Survived Pclass      Sex           Age
##  0:549    1:216   female:314   Min.   : 0.42
##  1:342    2:184   male  :577   1st Qu.:22.00
##           3:491                Median :29.70
##                                Mean   :29.70
##                                3rd Qu.:35.00
##                                Max.   :80.00
```

Regresión logística de una variable:

Comenzaremos realizando la regresión logistica con una sola variable: Sex.

```
# Ajuste de un modelo logístico
regLog_1 <- glm(Survived ~ Sex, data = datos_reg, family = "binomial")

# Mostramos los resultados
summary(regLog_1)
```

```
##
## Call:
## glm(formula = Survived ~ Sex, family = "binomial", data = datos_reg)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6462  -0.6471  -0.6471   0.7725   1.8256
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.0566     0.1290   8.191 2.58e-16 ***
## Sexmale      -2.5137     0.1672 -15.036  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.7  on 890  degrees of freedom
## Residual deviance:  917.8  on 889  degrees of freedom
## AIC: 921.8
##
## Number of Fisher Scoring iterations: 4
```

En la columna "Estimate" podemos encontrar los valores de los coeficientes beta para nuestra regresión.

La columna "standarderror" representa la exactitud de los coeficientes. A mayor error, peor será la estimación. La columna "z value" nos da el valor del coeficiente del estimador dividido por el error estandar.

La columna "Pr(>|z|)", nos muestra el p-valor correspondiente con el estadistico z. Cuanto menor es el p-valor, mayor es la importancia del estimador.

Por lo que como ambos coeficientes beta tienen un p-valor mucho menor que 0.001, tendremos una confianza de más del 99.9% en nuestro estimador.

Como el coeficiente beta para la variable Sexo(hombre) es negativo, significa que si alguien es hombre tiene menos prosibilidades de sobrevivir. Esto concuerda perfectamente con lo visto anteriormente.

Además, ser hombre disminuirá las probabilidades de supervivencia en exp(-2.51) = 0.081 veces.

Por lo que podemos concluir que ser hombre en el Titanic fue un factor de riesgo.

Según estos resultados podemos expresar la probabilidad de supervivencia (si solo tenemos en cuenta el sexo del pasajero) como: P(Supervivencia) = exp(1.0566 - 2.5137 sexo_masculino)/ [1 + exp(1.0566 - 2.5137 sexo_masculino)].

Regresión logística de varias variables

Realizaremos ahora la regresión logística con las variables clásicas: Sexo, edad y clase.

```
# Ajuste de un modelo logístico.
regLog_2 <- glm(Survived ~ Pclass + Sex + Age, data = datos_reg, family = "binomial")

# Mostramos los resultados
summary(regLog_2)
```

```
##
```

```
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age, family = "binomial",
##     data = datos_reg)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6490  -0.6636  -0.4198   0.6328   2.4283
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.54474    0.36537   9.702  < 2e-16 ***
## Pclass2     -1.12216    0.25773  -4.354 1.34e-05 ***
## Pclass3     -2.32917    0.24089  -9.669  < 2e-16 ***
## Sexmale     -2.61131    0.18671 -13.986  < 2e-16 ***
## Age         -0.03330    0.00737  -4.519 6.21e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.66  on 890  degrees of freedom
## Residual deviance:  805.29  on 886  degrees of freedom
## AIC: 815.29
##
## Number of Fisher Scoring iterations: 5
```

En la columna "Estimate" podemos encontrar los valores de los coeficientes beta para nuestra regresión.

La columna "standarderror" representa la exactitud de los coeficientes. A mayor error, peor será la estimación. La columna "z value" nos da el valor del coeficiente del estimador dividido por el error estandar.

La columna "Pr(>|z|)", nos muestra el p-valor correspondiente con el estadistico z. Cuanto menor es el p-valor, mayor es la importancia del estimador.

Por lo que como todos los coeficientes beta tienen un p-valor mucho menor que 0.001, tendremos una confianza de más del 99.9% en nuestro estimador.

Como el coeficiente beta para la variable Sexo(hombre), Pclass(2), Pclass(3) y Age son negativos, significa que si alguien cumple alguna de esas condiciones tiene menos posibilidades de sobrevivir. Además:

- ser hombre disminuirá las probabilidades de supervivencia en exp(-2.61) = 0.0735 veces.
- Pertenecer a 2ª clase disminuirá las probabilidades de supervivencia en exp(-1.12) = 0.326 veces.
- Pertenecer a 3ª clase disminuirá las probabilidades de supervivencia en exp(-2.33) = 0.097veces.
- La edad disminuirá las probabilidades de supervicencia en exp(-0.033) = 0.97 veces.

Con esta regresión podemos precedir la supervivencia de los datos extraidos del conjunto test.csv. Asignaremos que alguien sobrevive si tiene un porcentaje de supervivencia de más del 50%.

Cargamos los datos y los preparamos:

```
# Seleccionamos los datos que vamos a utilizar del conjunto test
datos_test_reg <- select(test, Pclass, Sex, Age)

# Marcamos que son datos discretos
```

```r
datos_test_reg$Pclass <- as.factor(datos_test_reg$Pclass)
datos_test_reg$Sex <- as.factor(datos_test_reg$Sex)

# Comprobamos que no hay datos vacios
colSums(is.na(datos_test_reg))
```

```
## Pclass    Sex    Age
##      0      0     86
```

```r
# Eliminamos los registros con datos vacios, ya que no vamos a poder realizar la categorización correcta
datos_test_reg <- datos_test_reg[!is.na(datos_test_reg$Age),]

# Realizamos la predicción
prob = predict(regLog_2, datos_test_reg, interval="prediction", type="response", level = 0.95)

# Almacenamos el porcentaje de supervivencia
predicted_class_cat <- round(prob*100,2)

# Almancenamos la predicción.
predicted_class <- ifelse(prob> 0.5, 1, 0)
predicted_class <- as.factor(predicted_class)
summary(predicted_class)
```

```
##   0   1
## 202 130
```

Como vemos predecimos que 202 personas morirán y 130 se salvarán.

Si tuviésemos los datos objetivo podríamos ver que precisión real tendría nuestra regresión.

Análisis: Método no supervisado: Clustering

Uno de los métodos de clustering por excelencia es el k-means; pero para aplicarlo tendremos que transformar nuestros datos en numéricos. No es lo ideal, ya que no existe una distancia lógica entre los datos categoricos tipo hombre/mujer, pero nos permitirá aplicar el método.

```r
# Unimos los dos juetos de datos en uno solo
totalData <- bind_rows(train,test)
filas=dim(train)[1]

# Eliminamos la variable objetivo Survived, las variables que hemos eliminado para otros estudios y tamb
totalData <- select(totalData, -Survived, -PassengerId, -Name, - Cabin, -Ticket)

# Creamos la variable FamilySize
totalData$FamilySize <- totalData$SibSp + totalData$Parch +1;

# Eliminamos SibSp y Parch
totalData <- select(totalData, -SibSp, -Parch)

# Marcamos las variables como discretas
cols<-c("Sex","Embarked")
for (i in cols){
```

```
  totalData[,i] <- as.factor(totalData[,i])
}
for (i in cols){
  totalData[,i] <- as.integer(totalData[,i])
}

colSums(is.na(totalData))
```

```
##     Pclass        Sex        Age       Fare   Embarked FamilySize
##          0          0        263          1          0          0
```

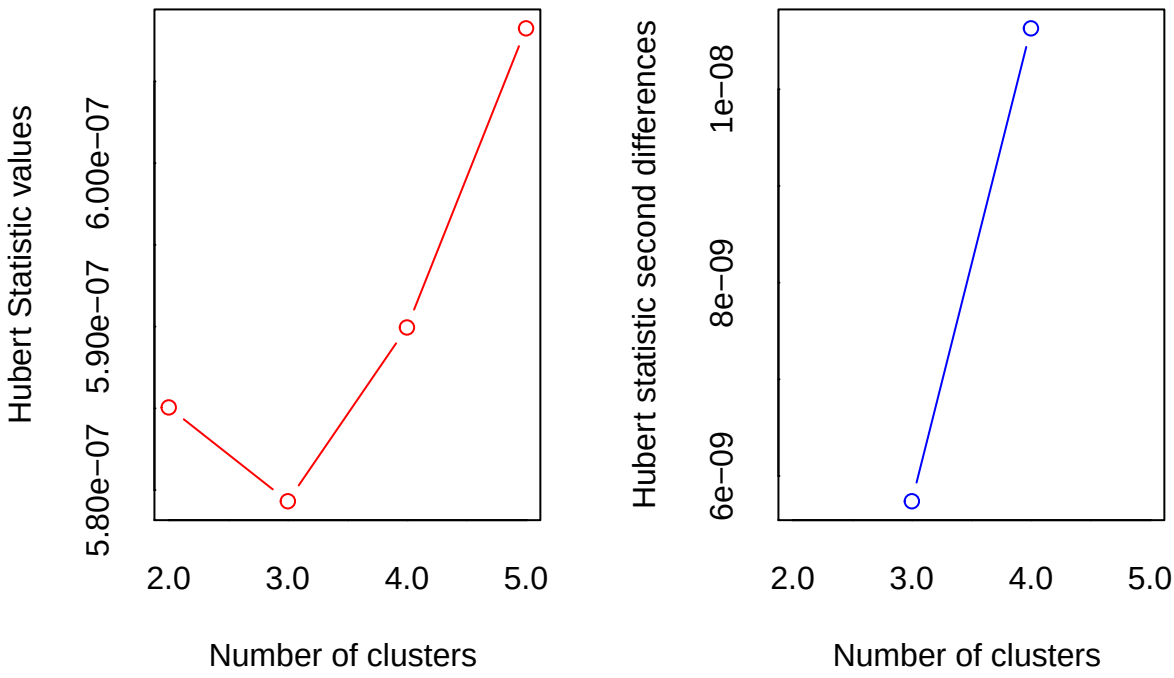
```
# Eliminamos los registros con datos vacios
totalData <- totalData[!is.na(totalData$Age),]
totalData <- totalData[!is.na(totalData$Fare),]

# Verificamos la estructura del juego de datos
str(totalData)
```

```
## 'data.frame':    1045 obs. of  6 variables:
##  $ Pclass    : int  3 1 3 1 3 1 3 3 2 3 ...
##  $ Sex       : int  2 1 1 1 2 2 2 1 1 1 ...
##  $ Age       : num  22 38 26 35 35 54 2 27 14 4 ...
##  $ Fare      : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Embarked  : int  4 2 4 4 4 4 4 4 2 4 ...
##  $ FamilySize: num  2 2 1 2 1 1 5 3 2 3 ...
```

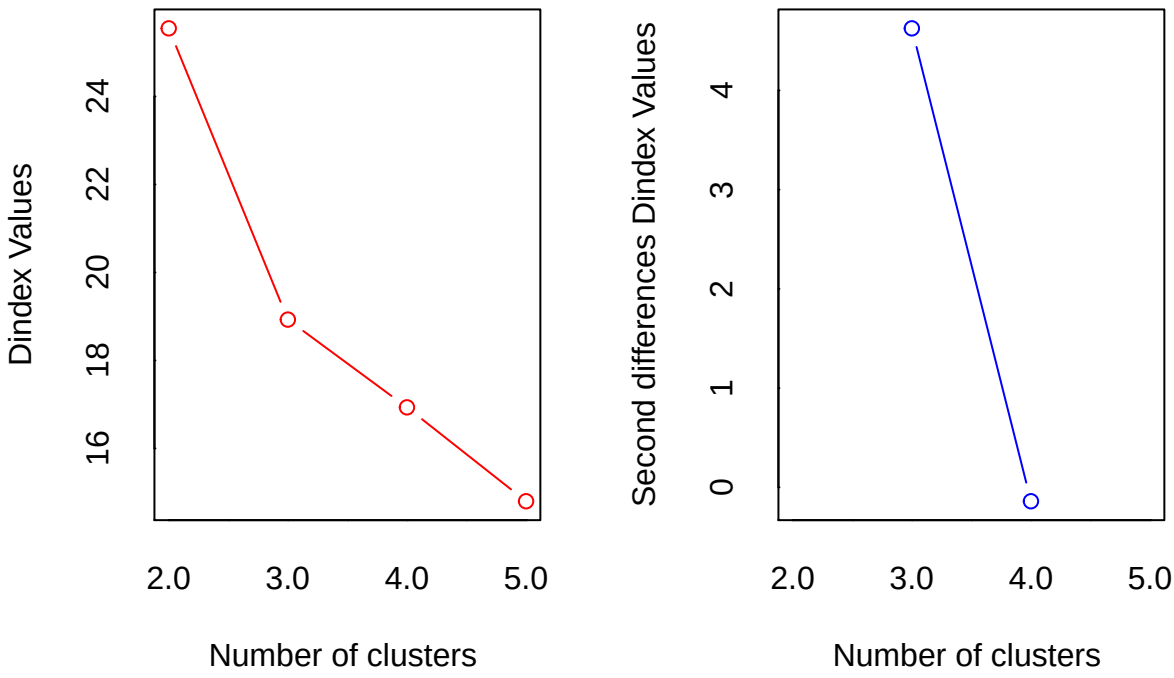
Aunque sabemos que deberíamos agrupar los datos en dos clusters (supervivientes: si o no), realizaremos un estudio para ver cuál es el número óptimo de clusters en los que deberíamos agruparlos.

```
numero_clusters <- NbClust(data = totalData, distance = "euclidean", min.nc = 2, max.nc = 5, method = "
```

```
## [1] "Frey index : No clustering structure in this data set"
```




```
## *** : The Hubert index is a graphical method of determining the number of clusters.
##                 In the plot of Hubert index, we seek a significant knee that corresponds to a
##                 significant increase of the value of the measure i.e the significant peak in Hubert
##                 index second differences plot.
##
```

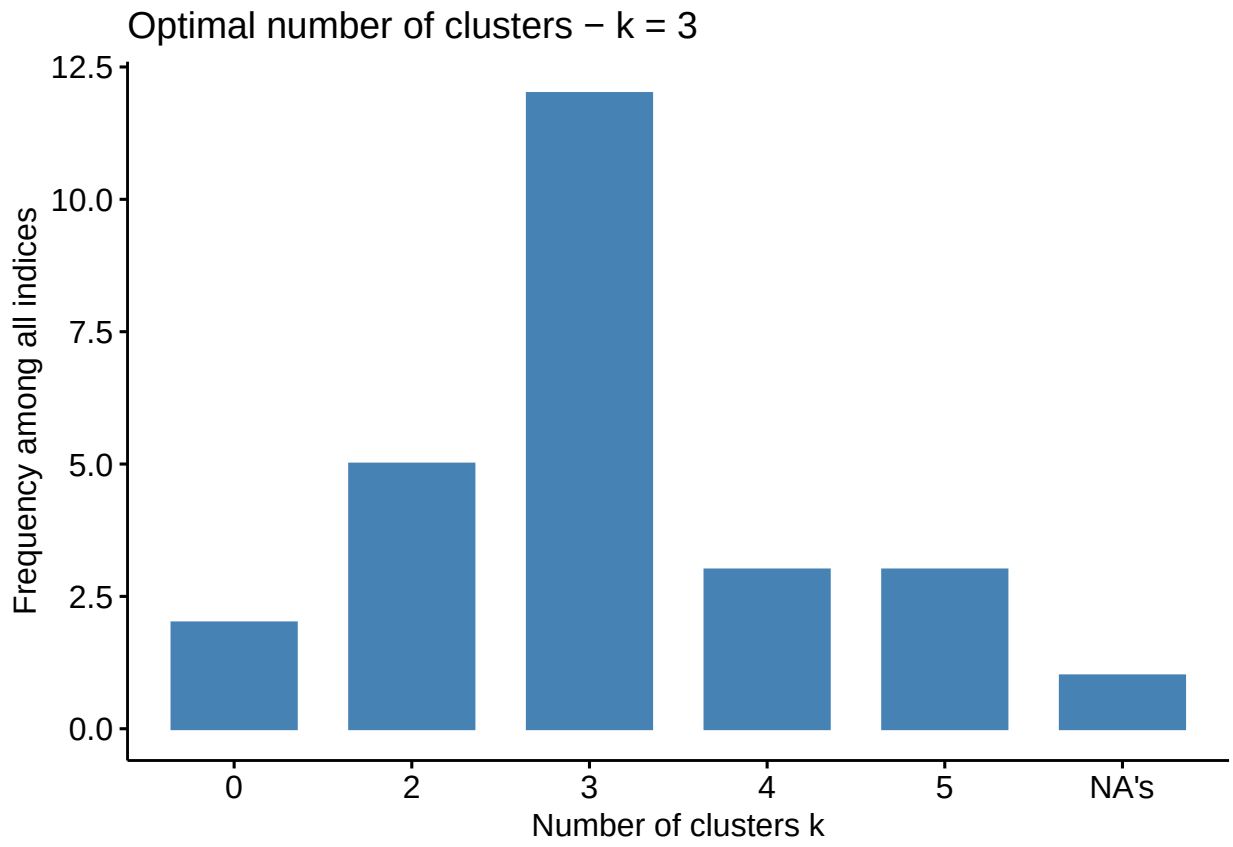



```
## *** : The D index is a graphical method of determining the number of clusters.
##                 In the plot of D index, we seek a significant knee (the significant peak in Dindex
##                 second differences plot) that corresponds to a significant increase of the value of
##                 the measure.
##
## *******************************************************************
## * Among all indices:
## * 5 proposed 2 as the best number of clusters
## * 12 proposed 3 as the best number of clusters
## * 3 proposed 4 as the best number of clusters
## * 3 proposed 5 as the best number of clusters
##
##                     ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  3
##
##
## *******************************************************************
```

```
fviz_nbclust(numero_clusters)
```

```
## Among all indices:
## ==================
## * 2 proposed  0 as the best number of clusters
## * 5 proposed  2 as the best number of clusters
```

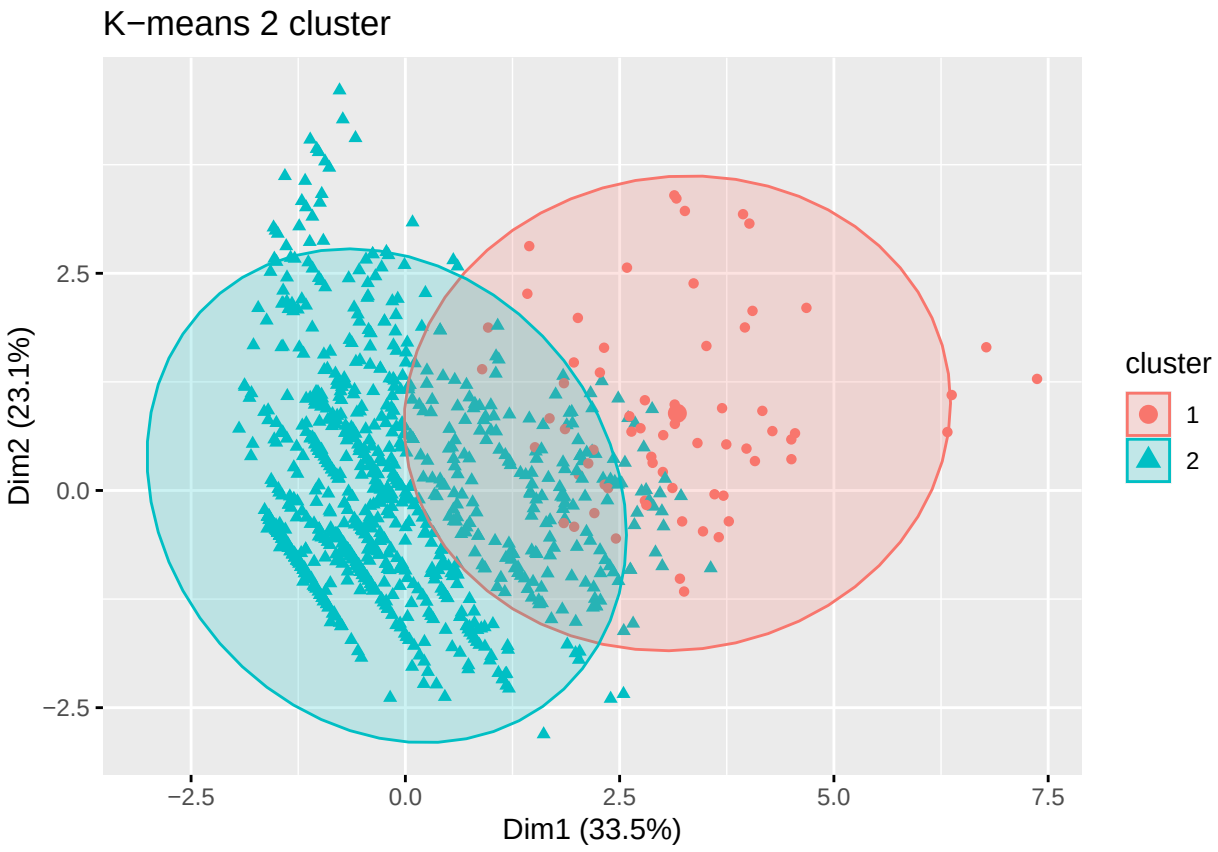
```
## * 12 proposed  3 as the best number of clusters
## * 3 proposed  4 as the best number of clusters
## * 3 proposed  5 as the best number of clusters
## * 1 proposed  NA's as the best number of clusters
##
## Conclusion
## =========================
## * According to the majority rule, the best number of clusters is  3 .
```



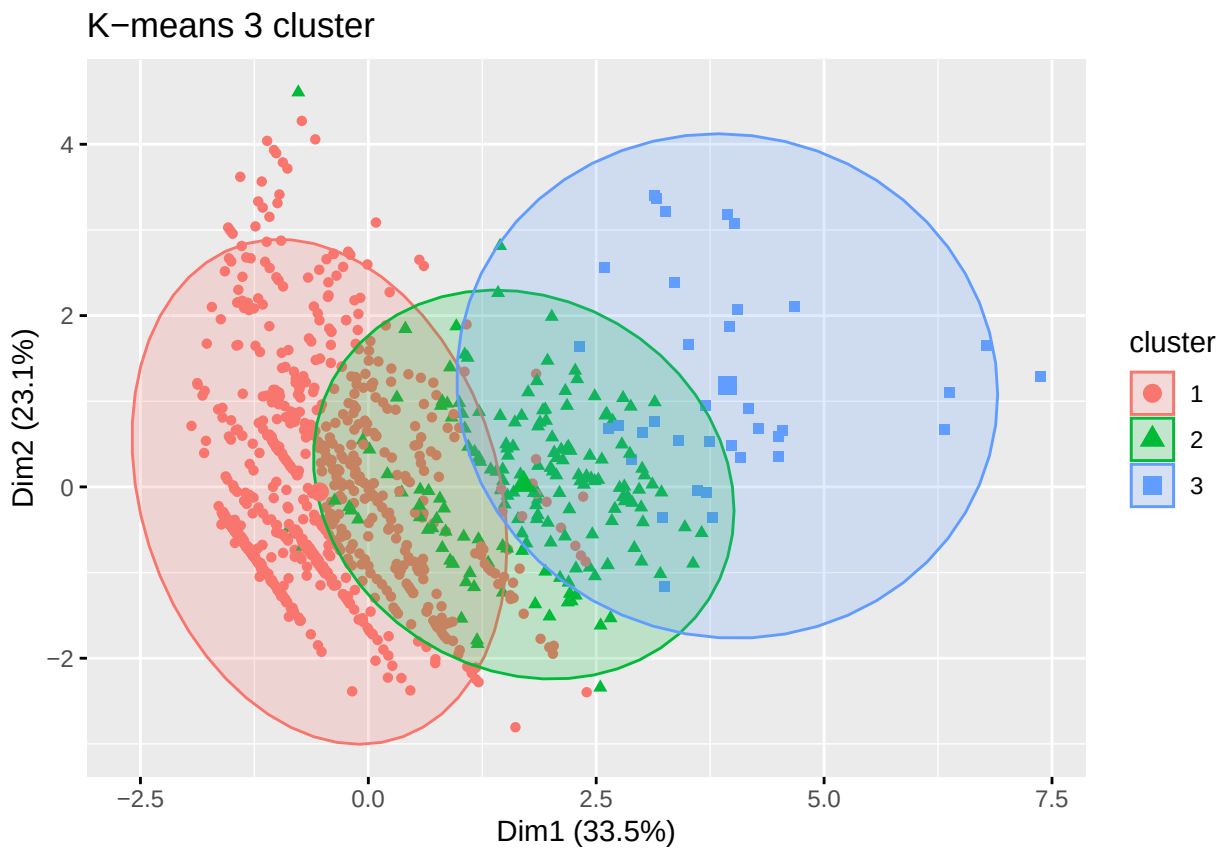
Como podemos comprobar el número óptimo de clusters según 3 de los métodos es k = 3, pero para 5 de los métodos es 2.

Procederemos a aplicar el k-means a ambos.

```
data2cluster <- kmeans(totalData,2)
data3cluster <- kmeans(totalData,3)

fviz_cluster(data2cluster, data = totalData, show.clust.cent=TRUE, geom="point", main="K-means 2 cluste
```




```
fviz_cluster(data3cluster, data = totalData, show.clust.cent=TRUE, geom="point", main="K-means 3 cluste
```



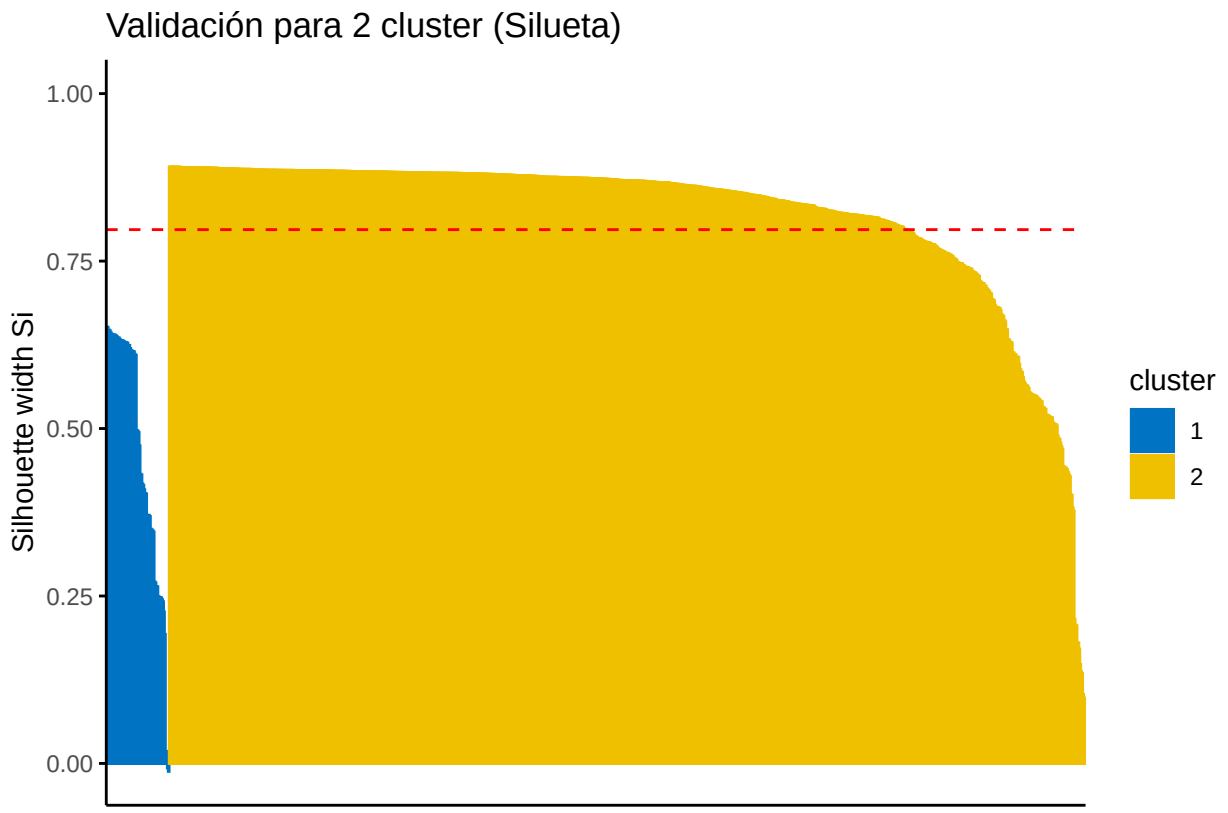
Utilizaremos el coeficiente de la silueta para validar nuestra agrupación.

El coeficiente de silueta es un metodo interno de validación que da una medida de la similitud de nuestro agrupamiento (los valores cercanos a 1 son los deseables).

Para k = 2

```
val_cluster2 <- eclust(x = totalData, FUNcluster = "kmeans", k = 2, seed = 123,
                       hc_metric = "euclidean", nstart = 50, graph = FALSE)
fviz_silhouette(sil.obj = val_cluster2, print.summary = TRUE, main="Validación para 2 cluster (Silueta)"
                ggtheme = theme_classic())
```

```
##   cluster size ave.sil.width
## 1       1   67          0.46
## 2       2  978          0.82
```




```
# Media del coeficiente de la silueta por cluster
val_cluster2$silinfo$clus.avg.widths
```

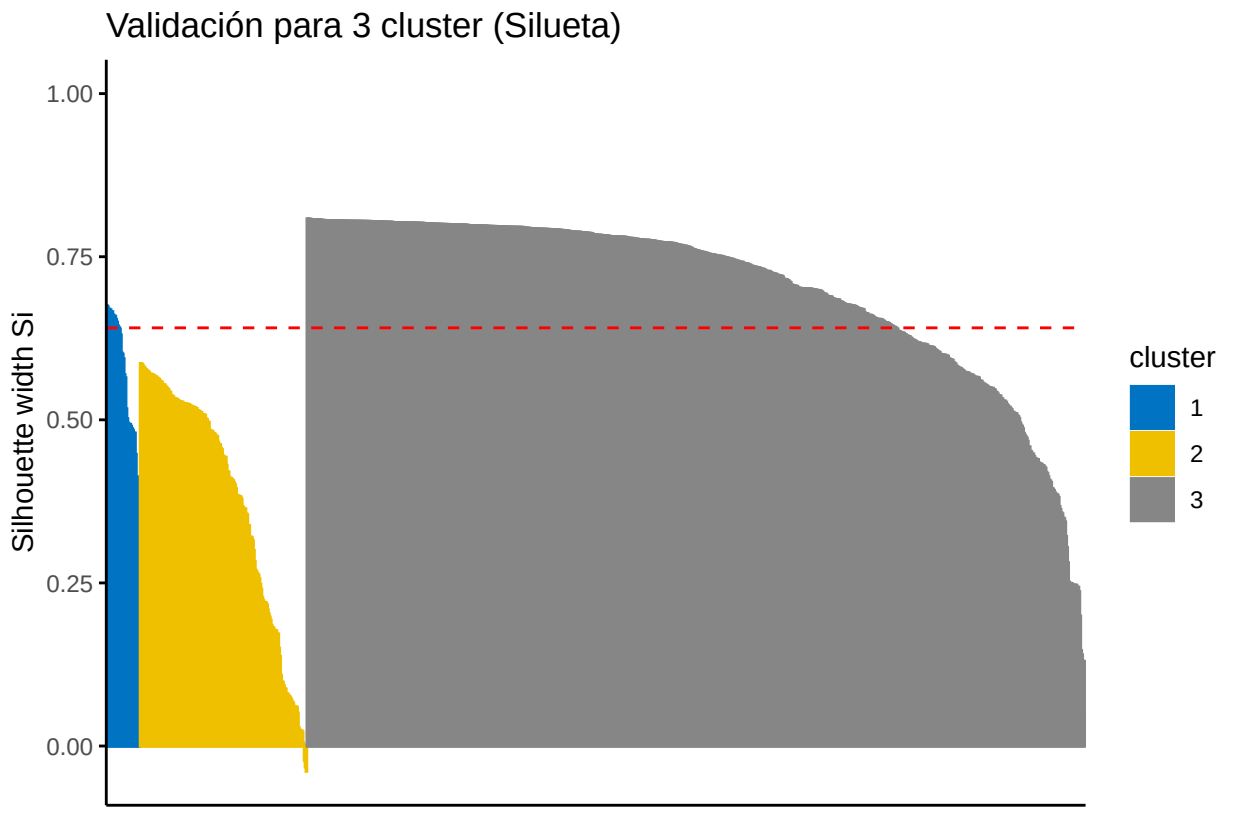
```
## [1] 0.4617272 0.8199298
```

Observamos coeficientes de 0.82 y 0.46, el primero muestra una buena similitud dentro del grupo, pero el segundo, indica que podría haber datos mal agrupados.

Para k = 3

```
# Repetiremos el estudio de la viabilidad para los valores de K=3
val_cluster3 <- eclust(x = totalData, FUNcluster = "kmeans", k = 3, seed = 123,
                       hc_metric = "euclidean", nstart = 50, graph = FALSE)
fviz_silhouette(sil.obj = val_cluster3, print.summary = TRUE, main="Validación para 3 cluster (Silueta)"
                ggtheme = theme_classic())
```

```
##   cluster size ave.sil.width
## 1       1   36          0.57
## 2       2  178          0.38
## 3       3  831          0.70
```




```
# Media del coeficiente de la silueta por cluster
val_cluster3$silinfo$clus.avg.widths
```

```
## [1] 0.5686590 0.3765787 0.7007432
```

Los coeficientes son: 0.70, 0.38 y 0.57.

La similitud del primer cluster es buena, pero la de los otros dos no; por lo tanto podría haber errores de agrupación.

Análisis: Correlación

Para estudiar la correlación entre las variables habrá que tener en cuenta que todas las variables de interés las vamos a analizar en relación con la variable Survived, que es una variable objetivo.

Por tanto, estudiaremos la distribución mediante las tablas de frecuencia y mediremos la correlación con el test chi-cuadrado.

Para cada par de variables:

- Survived vs Pclass
- Survived vs Age
- Survived vs Sex
- Survived vs Fare

- Survived vs Embarked
- Survived vs FamilySize

Seguiremos los siguientes pasos:

- Crearemos y visualizaremos la tabla de frecuencias.
- Pasaremos el test chi-cuadrado. Donde obtendremos los grados de libertad, el valor de chi-cuadrado y el p-value.
- Finalmente acudiremos a la tabla de chi-cuadrado para comprobar si nuestras variables están relacionadas o no

NOTA: Utilizaremos la tabla http://www.mat.uda.cl/hsalinas/cursos/2010/eyp2/Tabla%20Chi-Cuadrado.pdf

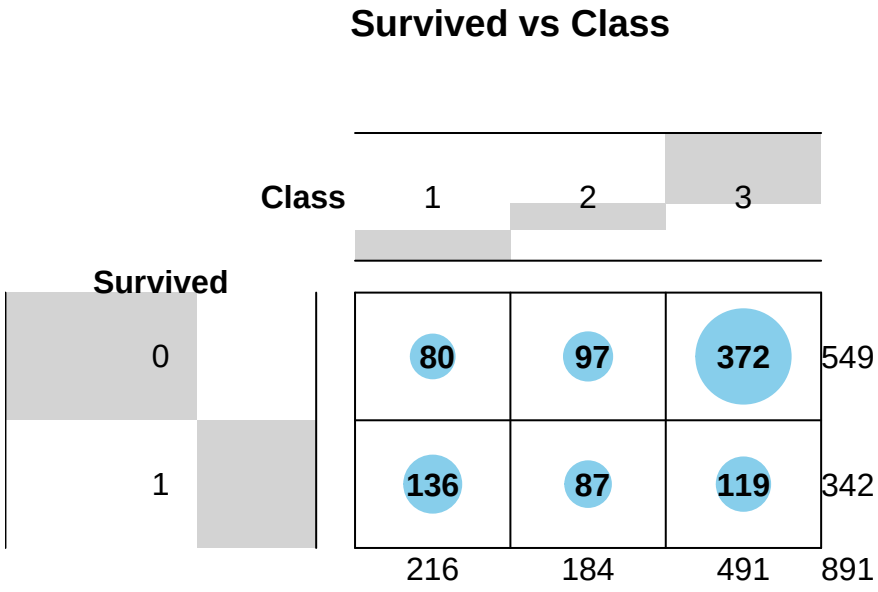
Utilizaremos los datos categoricos para poder visualizar mejor la información:

```
# Cargamos los datos que vamos a usar en la correlación
data_corr <- select (data, -SibSp,-Parch, -PassengerId)

# Categorizamos la variable Age
data_corr$Age_cat <- data_corr$Age
levels(data_corr$Age_cat) <- c(levels(data_corr$Age_cat), "Kid", "teenager", "adult", "old person")
data_corr[data_corr$Age < 13, "Age_cat"] <- "kid"
data_corr[data_corr$Age >= 13 & data_corr$Age < 20, "Age_cat"] <- "teenager"
data_corr[data_corr$Age >= 20 & data_corr$Age < 65, "Age_cat"] <- "adult"
data_corr[data_corr$Age >= 65, "Age_cat"] <- "old person"

# Categorizamos la variable Fare
data_corr$Fare_cat <- data_corr$Fare
levels(data_corr$Fare_cat) <- c(levels(data_corr$Fare_cat), "Cheapest", "Medium", "Expensive")
data_corr[data_corr$Fare < 50, "Fare_cat"] <- "Cheapest"
data_corr[data_corr$Fare >= 50 & data_corr$Fare < 100, "Fare_cat"] <- "Medium"
data_corr[data_corr$Fare >= 100, "Fare_cat"] <- "Expensive"

# Categorizamos la variable FamilySize
data_corr$FamilySize_cat <- data_corr$Fare
levels(data_corr$FamilySize_cat) <- c(levels(data_corr$FamilySize_cat), "Alone", "Pair", "Team")
data_corr[data_corr$FamilySize == 1, "FamilySize_cat"] <- "Alone"
data_corr[data_corr$FamilySize == 2, "FamilySize_cat"] <- "Pair"
data_corr[data_corr$FamilySize > 2, "FamilySize_cat"] <- "Team"

# Discretizamos las variables
cols<-c("Age_cat","Fare_cat","FamilySize_cat")
for (i in cols){
  data_corr[,i] <- as.factor(data_corr[,i])
}

# Eliminamos las variables originales
data_corr <- select(data_corr, -Age, -Fare, -FamilySize)
```

Survived vs Pclass

Creamos y representamos la tabla de frecuencias y pasamos el test de chi-cuadrado.

```
SurvivedvsPclass<- table(data_corr$Survived, data_corr$Pclass, dnn = c("Survived", "Class"))
balloonplot(t(SurvivedvsPclass), main ="Survived vs Class", xlab ="Class", ylab="Survived", label = T, s
```




```
xsq <- chisq.test(SurvivedvsPclass)
xsq
```

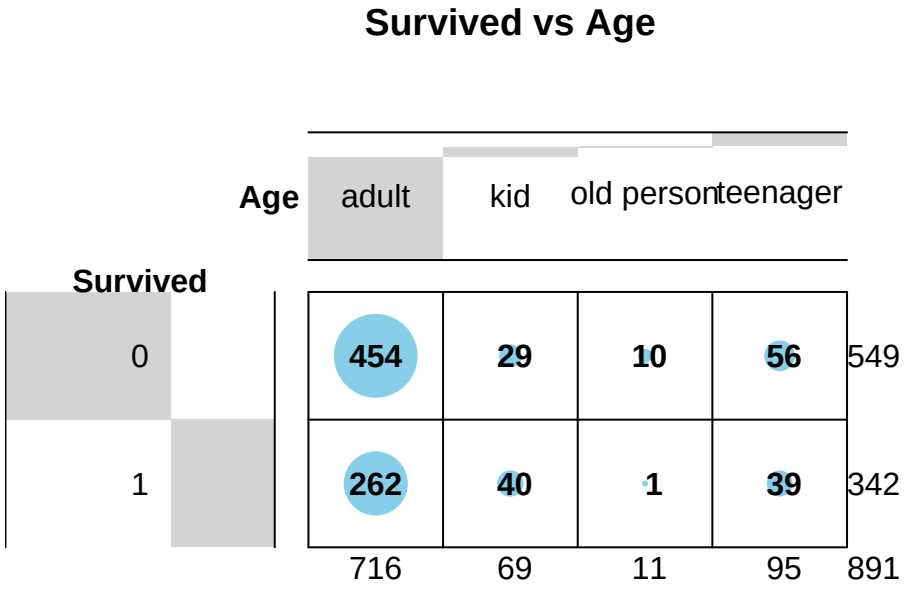
```
##
##  Pearson's Chi-squared test
##
## data:  SurvivedvsPclass
## X-squared = 102.89, df = 2, p-value < 2.2e-16
```

Acudimos a la tabla y comprobamos que para 2 grados de libertad y una confianza de 0.001 el valor que aparece es 13,816. En nuestro caso es 102,89, por lo que si existe correlación entre estas dos variables.

Survived vs Age

Creamos y representamos la tabla de frecuencias y pasamos el test de chi-cuadrado.

```
SurvivedvsAge<- table(data_corr$Survived, data_corr$Age_cat, dnn = c("Survived", "Age"))
balloonplot(t(SurvivedvsAge), main ="Survived vs Age", xlab ="Age", ylab="Survived", label = T, show.ma
```




```
xsq <- chisq.test(SurvivedvsAge)
xsq
```

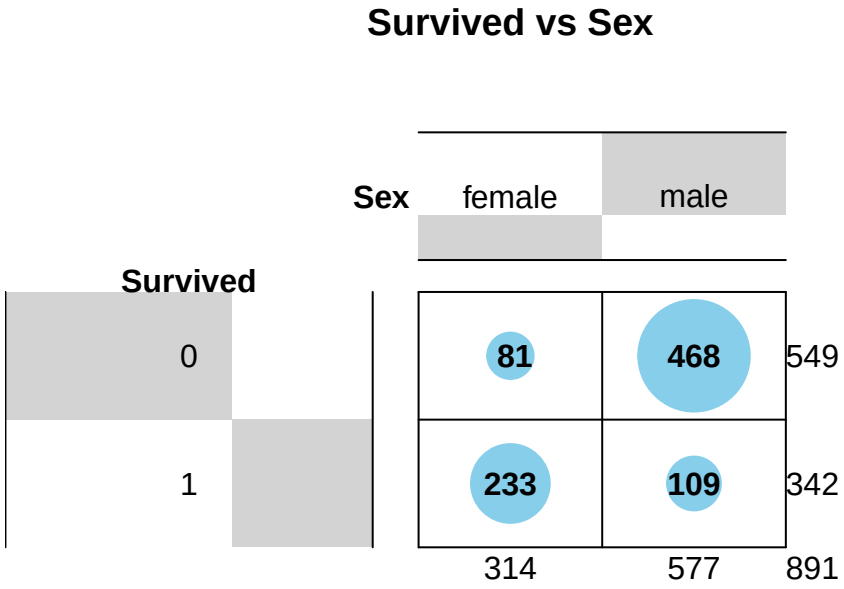
```
##
##  Pearson's Chi-squared test
##
## data:  SurvivedvsAge
## X-squared = 16.442, df = 3, p-value = 0.0009203
```

Acudimos a la tabla y comprobamos que para 3 grados de libertad y una confianza de 0.001 el valor que aparece es 16,266. En nuestro caso es 16.442, por lo que podemos concluir que sí existe correlación entre estas dos variables.

Survived vs Sex

Creamos y representamos la tabla de frecuencias y pasamos el test de chi-cuadrado.

```
SurvivedvsSex<- table(data_corr$Survived, data_corr$Sex, dnn = c("Survived", "Sex"))
balloonplot(t(SurvivedvsSex), main ="Survived vs Sex", xlab ="Sex", ylab="Survived", label = T, show.mar
```




```
xsq <- chisq.test(SurvivedvsSex)
xsq
```

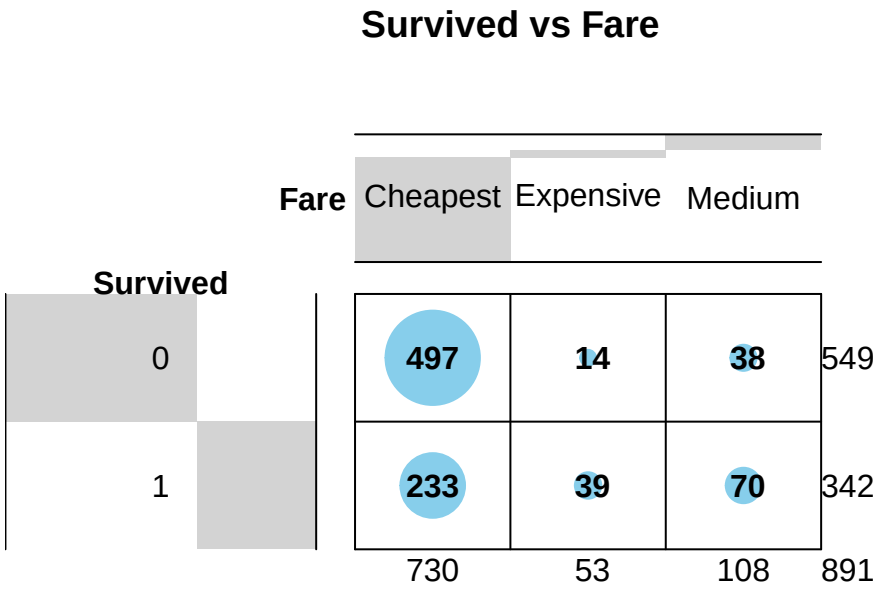
```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  SurvivedvsSex
## X-squared = 260.72, df = 1, p-value < 2.2e-16
```

Acudimos a la tabla y comprobamos que para 1 grados de libertad y una confianza de 0.001 el valor que aparece es 10,828. En nuestro caso es 260,72, por lo que si existe correlación entre estas dos variables.

Survived vs Fare

Creamos y representamos la tabla de frecuencias y pasamos el test de chi-cuadrado.

```
SurvivedvsFare<- table(data_corr$Survived, data_corr$Fare, dnn = c("Survived", "Fare"))
balloonplot(t(SurvivedvsFare), main ="Survived vs Fare", xlab ="Fare", ylab="Survived", label = T, show
```




```
xsq <- chisq.test(SurvivedvsFare)
xsq
```

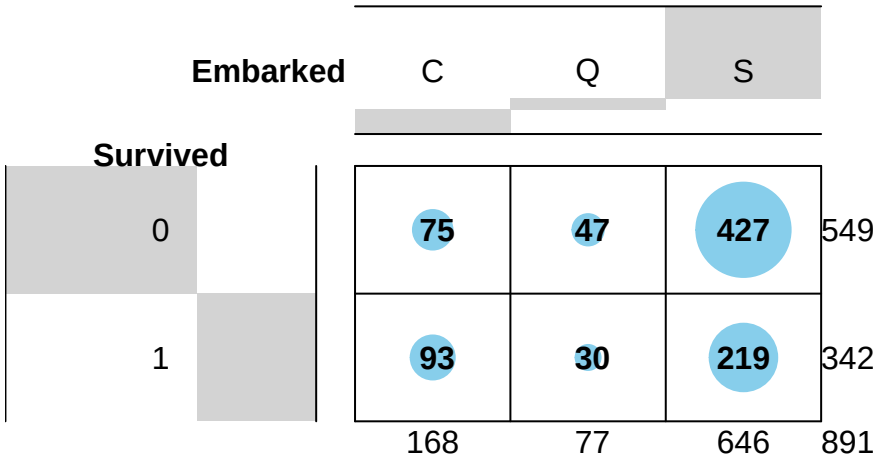
```
##
##  Pearson's Chi-squared test
##
## data:  SurvivedvsFare
## X-squared = 72.574, df = 2, p-value < 2.2e-16
```

Acudimos a la tabla y comprobamos que para 2 grados de libertad y una confianza de 0.001 el valor que aparece es 13,81. En nuestro caso es 72.574, por lo que si existe correlación entre estas dos variables.

Survived vs Embarked

Creamos y representamos la tabla de frecuencias y pasamos el test de chi-cuadrado.

```
SurvivedvsEmbarked<- table(data_corr$Survived, data_corr$Embarked, dnn = c("Survived", "Embarked"))
balloonplot(t(SurvivedvsEmbarked), main ="Survived vs Embarked", xlab ="Embarked", ylab="Survived", lab
```




```
xsq <- chisq.test(SurvivedvsEmbarked)
xsq
```

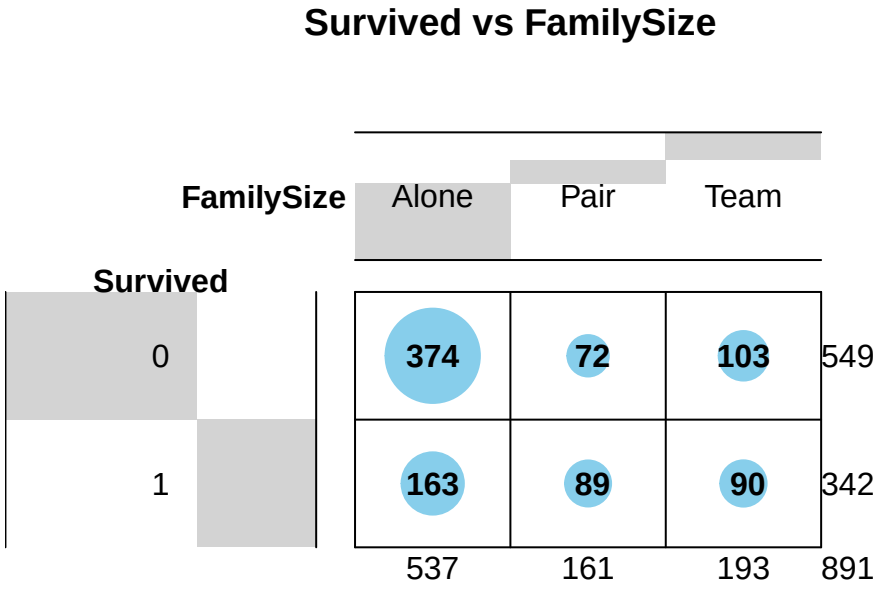
```
##
##  Pearson's Chi-squared test
##
## data:  SurvivedvsEmbarked
## X-squared = 25.964, df = 2, p-value = 2.301e-06
```

Acudimos a la tabla y comprobamos que para 2 grados de libertad y una confianza de 0.001 el valor que aparece es 13,81. En nuestro caso es 25.964, por lo que si existe correlación entre estas dos variables.

Survived vs FamilySize

Creamos y representamos la tabla de frecuencias y pasamos el test de chi-cuadrado.

```
SurvivedvsFamilySize<- table(data_corr$Survived, data_corr$FamilySize, dnn = c("Survived", "FamilySize"))
balloonplot(t(SurvivedvsFamilySize), main ="Survived vs FamilySize", xlab ="FamilySize", ylab="Survived")
```




```
xsq <- chisq.test(SurvivedvsFamilySize)
xsq
```

```
##
##  Pearson's Chi-squared test
##
## data:  SurvivedvsFamilySize
## X-squared = 39.625, df = 2, p-value = 2.486e-09
```

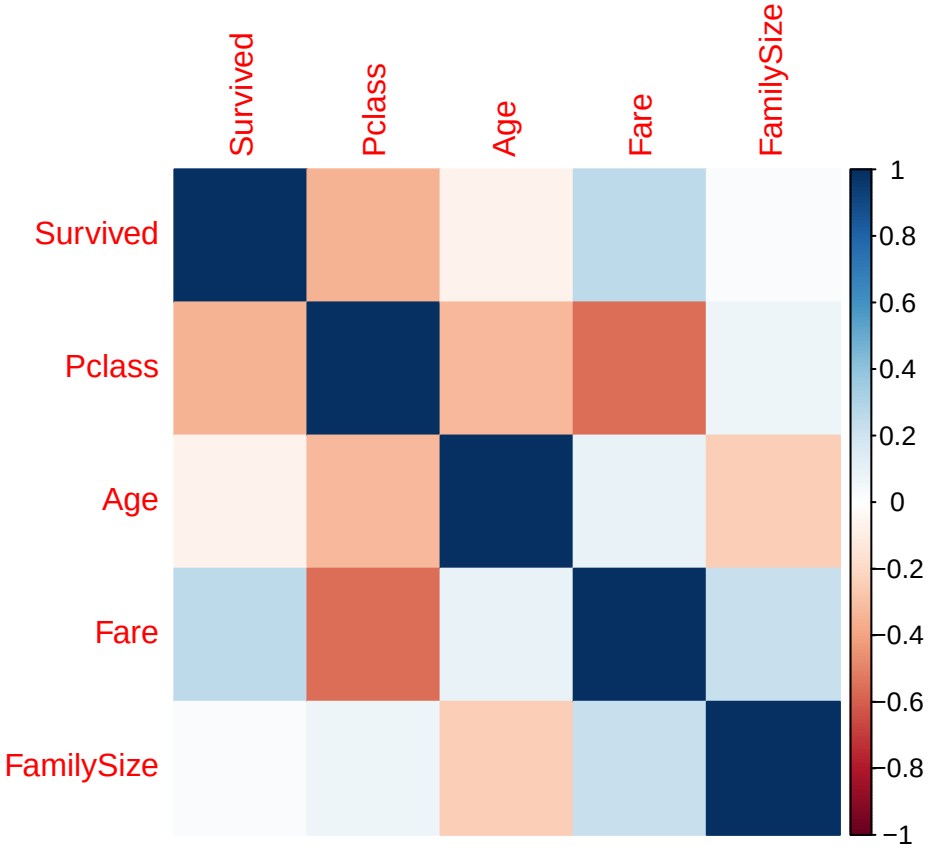
Acudimos a la tabla y comprobamos que para 2 grados de libertad y una confianza de 0.001 el valor que aparece es 13,81. En nuestro caso es 39.625, por lo que si existe correlación entre estas dos variables.

Matriz de correlaciones

Podemos obtener la matriz de correlaciones de las variables numéricas y de las variables categoricas que tengan un sentido de distancia lógica entre ellas (como es el caso de Pclass). En este análisis no podemos incluir el sexo del pasajero, puesto que no puede establecerse una distancia lógica entre sus componentes.

```
# Preparamos los datos para crear la matriz de correlación
data_matrix_corr <- select(data, Survived, Pclass, Age, Fare, FamilySize)

cols<-c("Survived", "Pclass")
for (i in cols){
  data_matrix_corr[,i] <- as.integer(data_matrix_corr[,i])
}
```

```
# Creamos y dibujamos la matriz de correlación
matrix_corr <- round(cor(data_matrix_corr),2)
corrplot(matrix_corr, method = "color", addtextlabel = TRUE)
```



En nuestro caso, lo que más nos interesa comprobar es la relación del atributo Survived con el resto. Como preveíamos, la clase es la variable más importante y relacionada con la supervivencia, aunque de manera inversa (los de 1ª clase se salvaron mucho más que los de 3ª).

También comprobamos que a mayor precio del billete, más facil era la supervivencia. Cosa lógica y completamente relacionado con la clase. Como podemos comprobar las variables PClass y Fare están de nuevo inversamente correlacionadas, además de una manera fuerte (a mayor precio de billete, menor número de clase).

La siguiente variable en importancia es la edad y por último el tamaño de la familia.

Tablas y gráficas

En primer lugar vamos a copiar nuestros datos en una nueva variable y a categorizar los datos numéricos, ya que eso facilitará la representación de los datos.

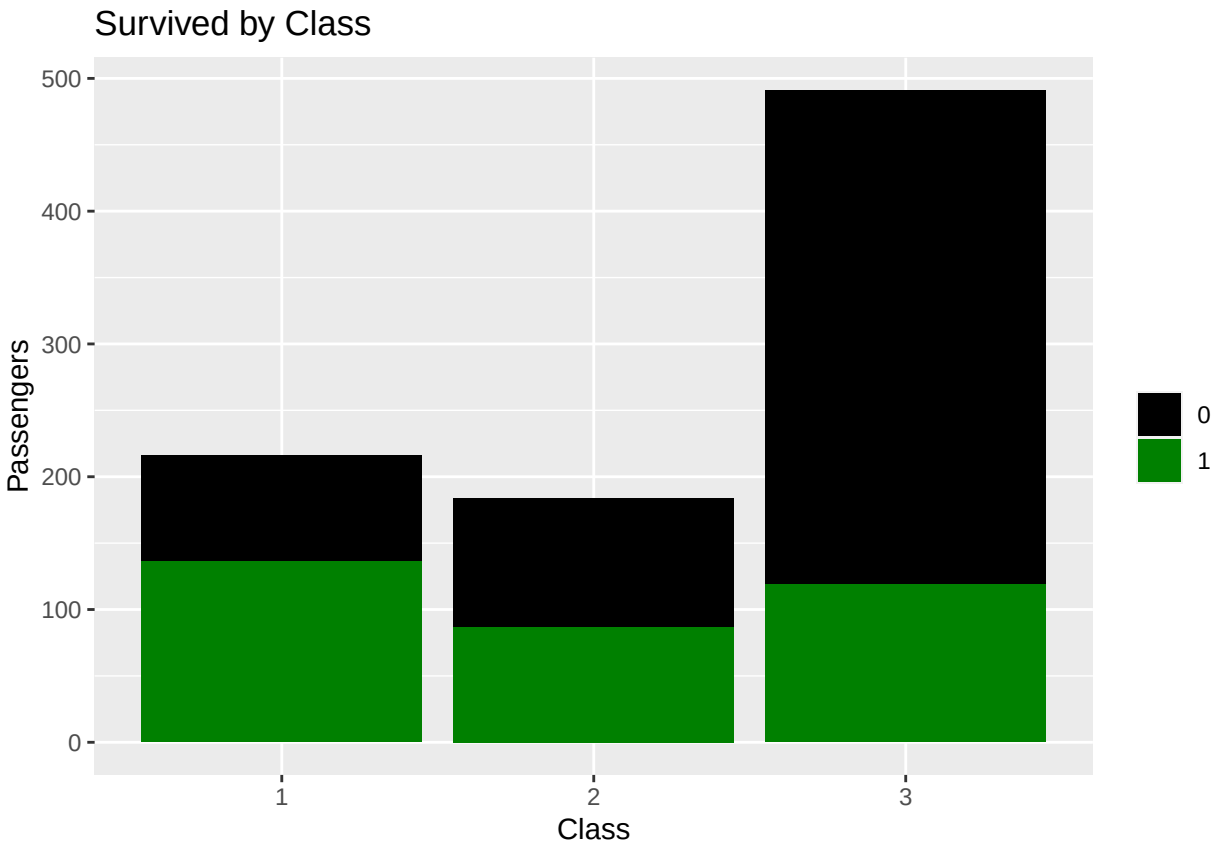
```
# Copiamos nuestros datos en una nueva variable
data_rep <- data

# Para simplificar la representación discretizaremos las variables numéricas:
# Variable Age
```

```
data_rep$Age_cat <- data_rep$Age
levels(data_rep$Age_cat) <- c(levels(data_rep$Age_cat), "Kid", "teenager", "adult", "old person")
data_rep[data_rep$Age < 13, "Age_cat"] <- "kid"
data_rep[data_rep$Age >= 13 & data_rep$Age < 20, "Age_cat"] <- "teenager"
data_rep[data_rep$Age >= 20 & data_rep$Age < 65, "Age_cat"] <- "adult"
data_rep[data_rep$Age >= 65, "Age_cat"] <- "old person"

# Variable Fare
data_rep$Fare_cat <- data_rep$Fare
levels(data_rep$Fare_cat) <- c(levels(data_rep$Fare_cat), "Cheapest", "Medium", "Expensive")
data_rep[data_rep$Fare < 50, "Fare_cat"] <- "Cheapest"
data_rep[data_rep$Fare >= 50 & data_rep$Fare < 100, "Fare_cat"] <- "Medium"
data_rep[data_rep$Fare >= 100, "Fare_cat"] <- "Expensive"

# Variable FamilySize
data_rep$FamilySize_cat <- data_rep$Fare
levels(data_rep$FamilySize_cat) <- c(levels(data_rep$FamilySize_cat), "Alone", "Pair", "Team")
data_rep[data_rep$FamilySize == 1, "FamilySize_cat"] <- "Alone"
data_rep[data_rep$FamilySize == 2, "FamilySize_cat"] <- "Pair"
data_rep[data_rep$FamilySize > 2, "FamilySize_cat"] <- "Team"

# Discretizamos las variables
cols<-c("Age_cat","Fare_cat","FamilySize_cat")
for (i in cols){
  data_rep[,i] <- as.factor(data_rep[,i])
}
```

Nos interesa describir la relación entre la supervivencia y cada una de las variables mencionadas anteriormente. Para ello, utilizaremos diagramas de barras.

Survived vs Pclass

```
ggplot(data=data_rep,aes(x=Pclass,fill=Survived))+geom_bar()+labs(x="Class", y="Passengers")+ guides(fi
```



Para obtener los datos que se muestran en la gráfica calcularemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t1<-table(data_rep$Survived, data_rep$Pclass)
t1 %>% kable() %>% kable_styling()
```

	1	2	3
0	80	97	372
1	136	87	119

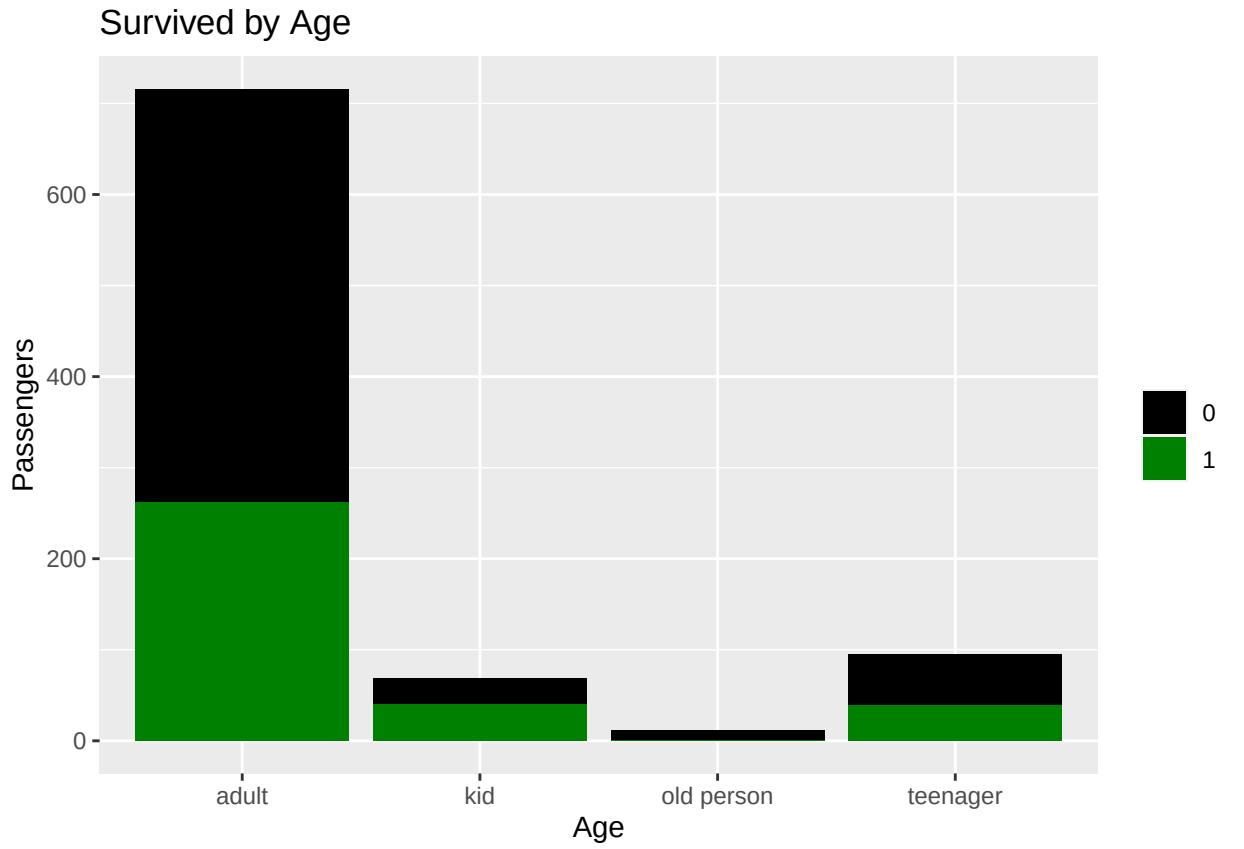
```
# Tabla de frecuencias relativas
t1_2 <- round(prop.table(x=t1)*100,2)
t1_2 %>% kable() %>% kable_styling()
```

	1	2	3
0	8.98	10.89	41.75
1	15.26	9.76	13.36

Como podemos observar, tanto en la gráfica como en las tablas, se confirma que el mayor porcentaje de personas fallecidas pertenecían a la tercera clase. Para los pasajeros de segunda clase, el porcentaje de fallecidos fue ligeramente superior al de supervivientes y los pasajeros de primera clase son la única clase en la que hubo más supervivientes que fallecidos. Se observa de forma muy clara que los pasajeros de tercera clase fueron los más afectados.

Survived vs Age

```
ggplot(data=data_rep,aes(x=Age_cat,fill=Survived))+geom_bar()+labs(x="Age", y="Passengers")+ guides(fil
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t2<-table(data_rep$Survived, data_rep$Age_cat)
t2 %>% kable() %>% kable_styling()
```

	adult	kid	old person	teenager
0	454	29	10	56
1	262	40	1	39

```
# Tabla de frecuencias relativas
t2_2 <- round(prop.table(x=t2)*100,2)
t2_2 %>% kable() %>% kable_styling()
```

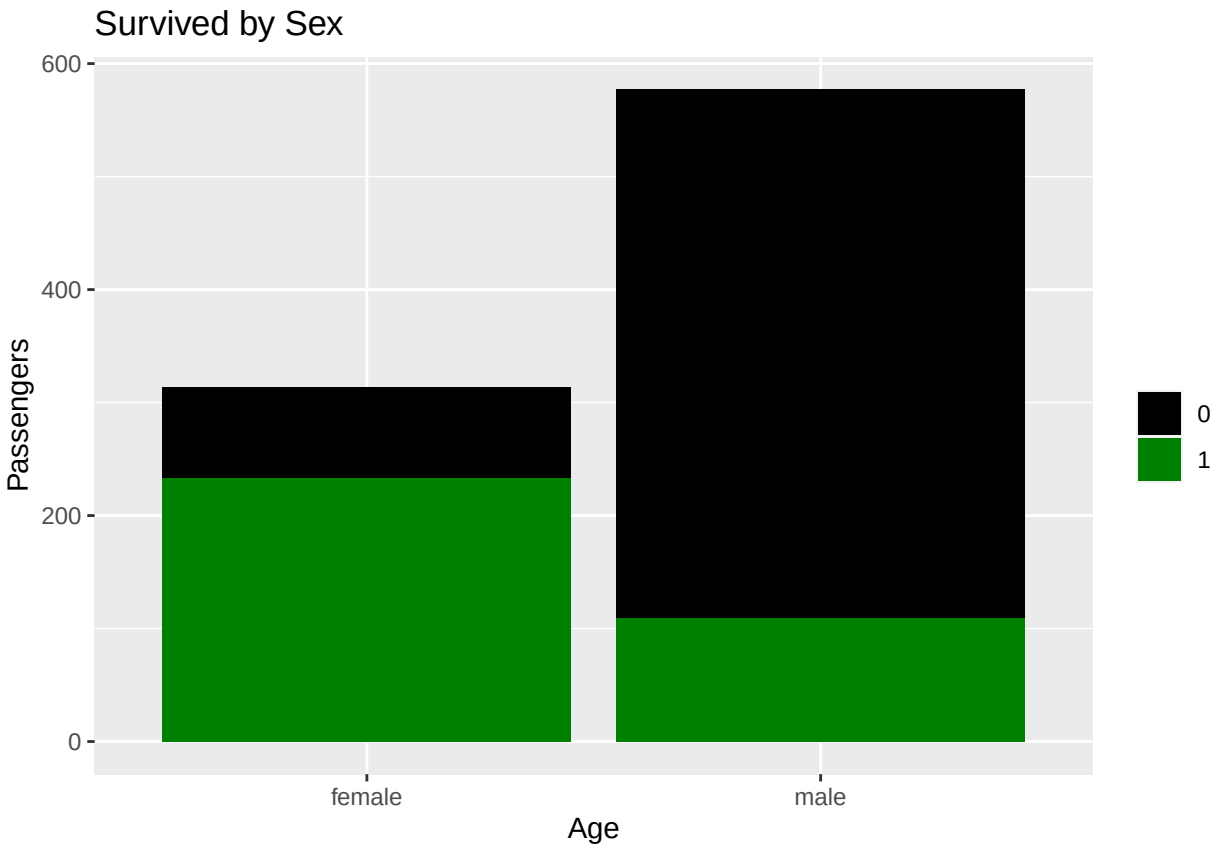
	adult	kid	old person	teenager
0	50.95	3.25	1.12	6.29
1	29.41	4.49	0.11	4.38

Como podemos observar sí parece ser cierto que los niños tuvieron preferencia a la hora de subirse a los botes salvavidas. Los niños son el único rango de edad en el que hubo más supervivientes que fallecidos. El rango de edad que, con mucha diferencia, se vió más afectado fueron los adultos, con la mitad de los

fallecimientos. Los ancianos presentan una tasa de supervivencia muy baja, pero cabe mencionar que la proporción de pasajeros pertenecientes a este rango de edad es muy pequeña.

- Survived vs Sex

```
ggplot(data=data_rep,aes(x=Sex,fill=Survived))+geom_bar()+labs(x="Age", y="Passengers")+ guides(fill=gu
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t3<-table(data_rep$Survived, data_rep$Sex)
t3 %>% kable() %>% kable_styling()
```

	female	male
0	81	468
1	233	109

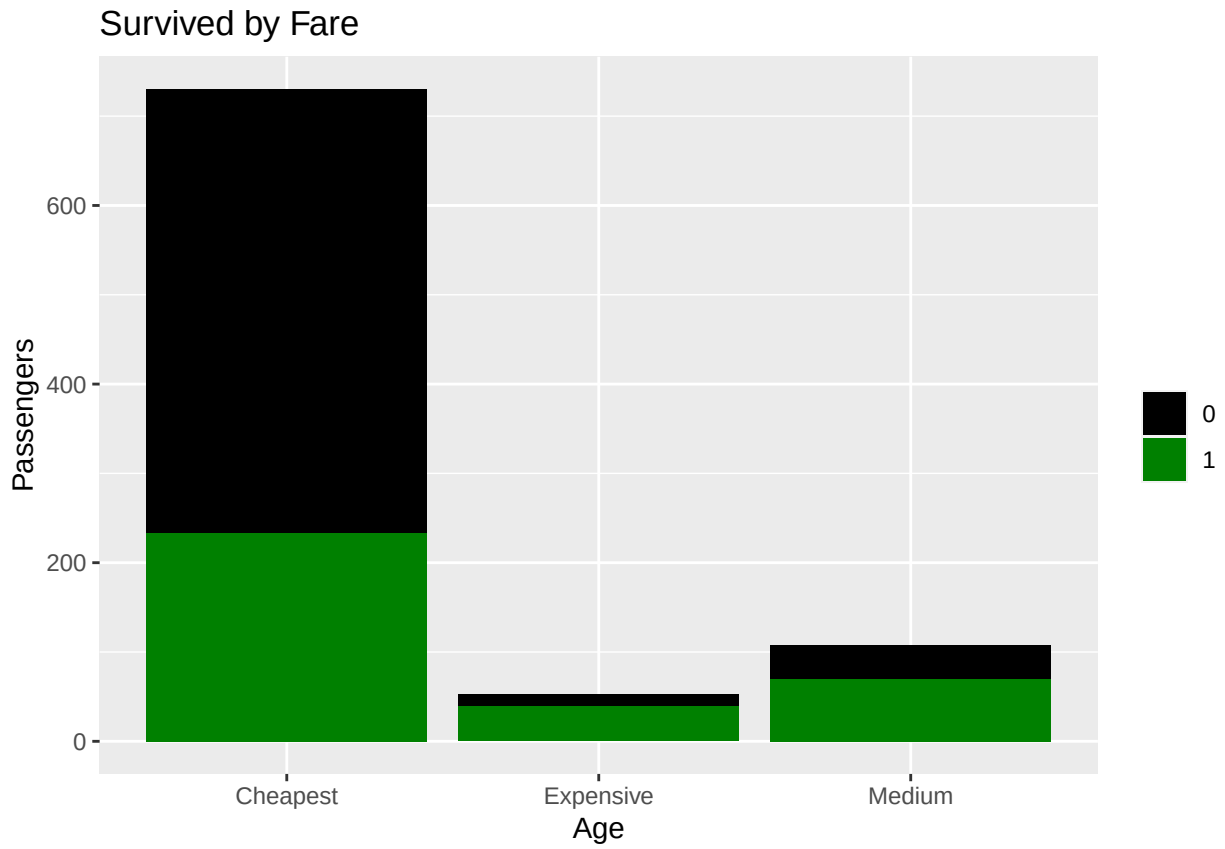
```
# Tabla de frecuencias relativas
t3_2 <- round(prop.table(x=t3)*100,2)
t3_2 %>% kable() %>% kable_styling()
```

	female	male
0	9.09	52.53
1	26.15	12.23

También se verifica que las mujeres tuvieron preferencia, junto con los niños, a la hora de optar a los botes salvavidas. Se observa una clara diferencia entre los hombres y las mujeres. De hecho, el porcentaje de hombres fallecidos es casi 6 veces superior al de mujeres fallecidas. Además, cabe mencionar que el porcentaje de mujeres que sobrevivieron es casi tres veces superior al de mujeres que fallecieron.

Survived vs Fare

```
ggplot(data=data_rep,aes(x=Fare_cat,fill=Survived))+geom_bar()+labs(x="Age", y="Passengers")+ guides(fi
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t4<-table(data_rep$Survived, data_rep$Fare_cat)
t4 %>% kable() %>% kable_styling()
```

	Cheapest	Expensive	Medium
0	497	14	38
1	233	39	70

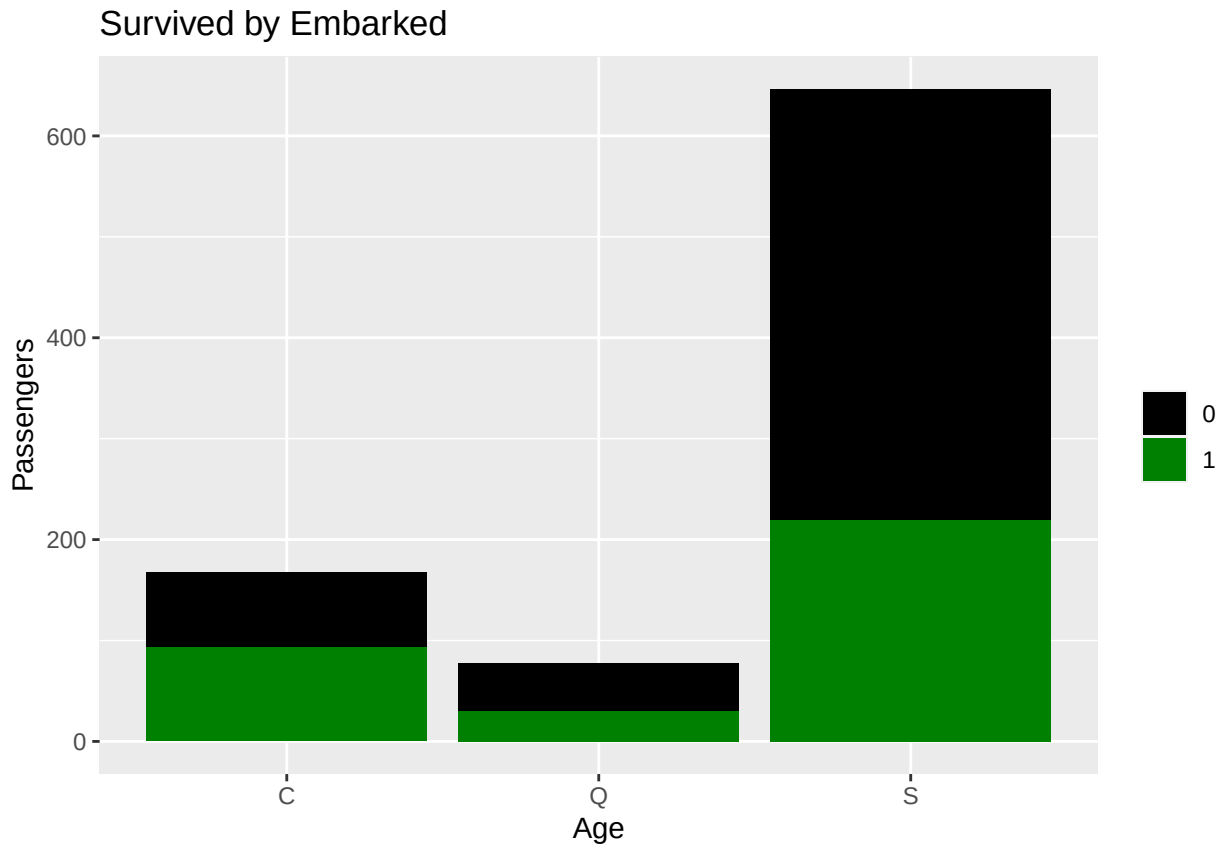
```
# Tabla de frecuencias relativas
t4_2 <- round(prop.table(x=t4)*100,2)
t4_2 %>% kable() %>% kable_styling()
```

	Cheapest	Expensive	Medium
0	55.78	1.57	4.26
1	26.15	4.38	7.86

Aqui podemos observar como la mayoría de los fallecidos había pagado un precio barato por su billete al Titanic. Esto era de esperar, ya que, como hemos observado previamente, el precio del billete está muy correlacionado con la clase.

Survived vs Embarked

```
ggplot(data=data_rep,aes(x=Embarked,fill=Survived))+geom_bar()+labs(x="Age", y="Passengers")+ guides(fi
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t5<-table(data_rep$Survived, data_rep$Embarked)
t5%>% kable() %>% kable_styling()
```

	C	Q	S
0	75	47	427
1	93	30	219

```
# Tabla de frecuencias relativas
t5_2 <- round(prop.table(x=t5)*100,2)
t5_2 %>% kable() %>% kable_styling()
```

	C	Q	S
0	8.42	5.27	47.92
1	10.44	3.37	24.58

Aquí obtenemos un resultado que también es muy interesante. Podemos observar que la mayoría de pasajeros que fallecieron embarcaron en S (Southampton), aunque hay que tener en cuenta que éste fue el puerto desde donde partió el Titanic y, como consecuencia, el porcentaje de pasajeros que embarcaron aquí era bastante superior al de los otros puertos. Sin embargo, consideramos interesante el hecho de que únicamente para aquellos que embarcaron en el puerto C (Cherbourg) encontramos un porcentaje de supervivientes superior al de fallecidos. Este resultado nos hace preguntarnos si este hecho podría tener alguna relación con el poder adquisitivo de las diferentes zonas y, por lo tanto, estar relacionado con la clase. Con el fin de saber si podría haber alguna relación, vamos a analizar cuántos pasajeros de cada clase embarcaron en los diferentes puertos:

```
C <- data$Pclass[data$Embarked=="C"]
S <- data$Pclass[data$Embarked=="S"]
Q <- data$Pclass[data$Embarked=="Q"]

table(C) %>% kable() %>% kable_styling()
```

C	Freq
1	85
2	17
3	66

```
table(S) %>% kable() %>% kable_styling()
```

S	Freq
1	129
2	164
3	353

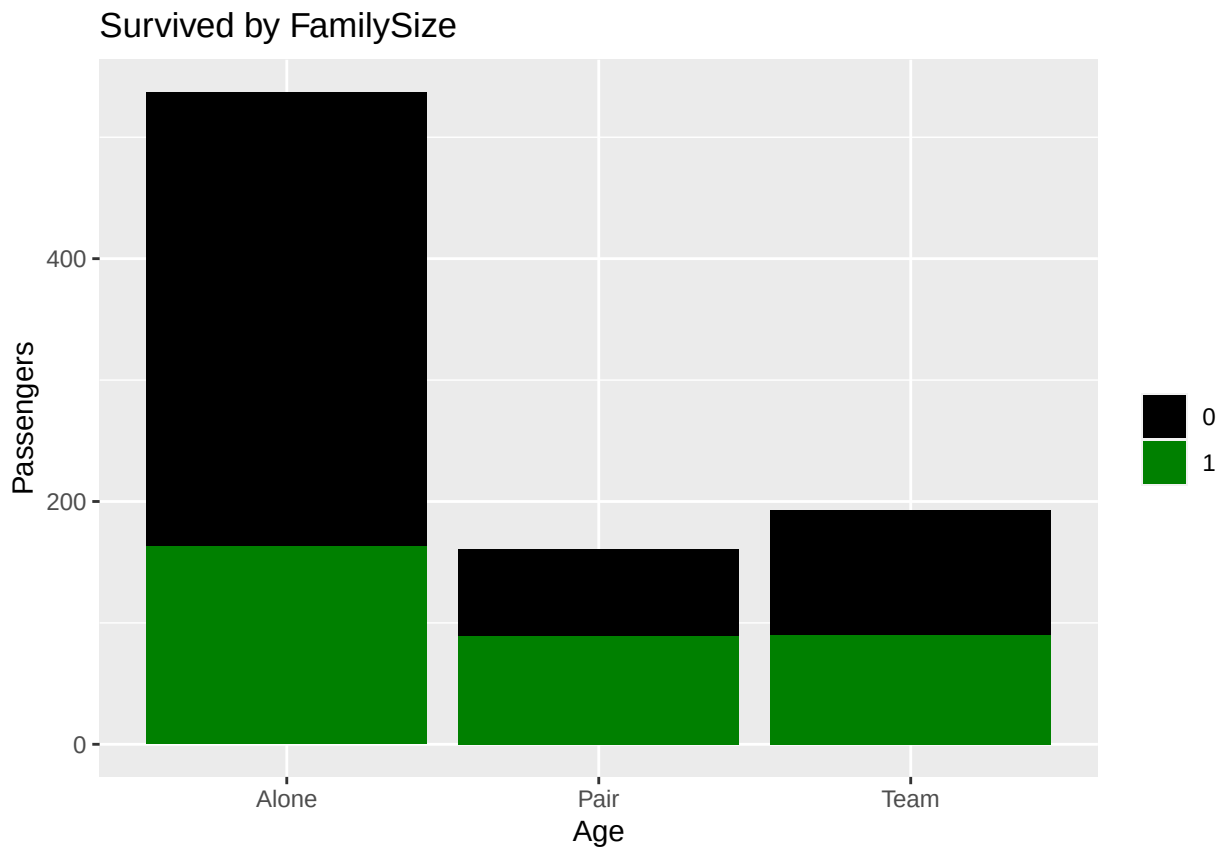
```
table(Q) %>% kable() %>% kable_styling()
```

Q	Freq
1	2
2	3
3	72

Como podemos ver en las tablas, curiosamente en el puerto C (Cherbourg) embarcaron más pasajeros de primera clase que de segunda y tercera juntos. Mientras que en el resto de puertos el número de pasajeros predominante correspondía a la tercera clase. No podemos saber con exactitud si nuestras suposiciones son ciertas pero los resultados sí nos hacen pensar que podría haber alguna relación entre el puerto de embarque y el poder adquisitivo de las diferentes zonas.

Survived vs FamilySize

```
ggplot(data=data_rep,aes(x=FamilySize_cat,fill=Survived))+geom_bar()+labs(x="Age", y="Passengers")+ gui
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t6<-table(data_rep$Survived, data_rep$FamilySize_cat)
t6 %>% kable() %>% kable_styling()
```

	Alone	Pair	Team
0	374	72	103
1	163	89	90

```
# Tabla de frecuencias relativas
t6_2 <- round(prop.table(x=t6)*100,2)
t6_2%>% kable() %>% kable_styling()
```

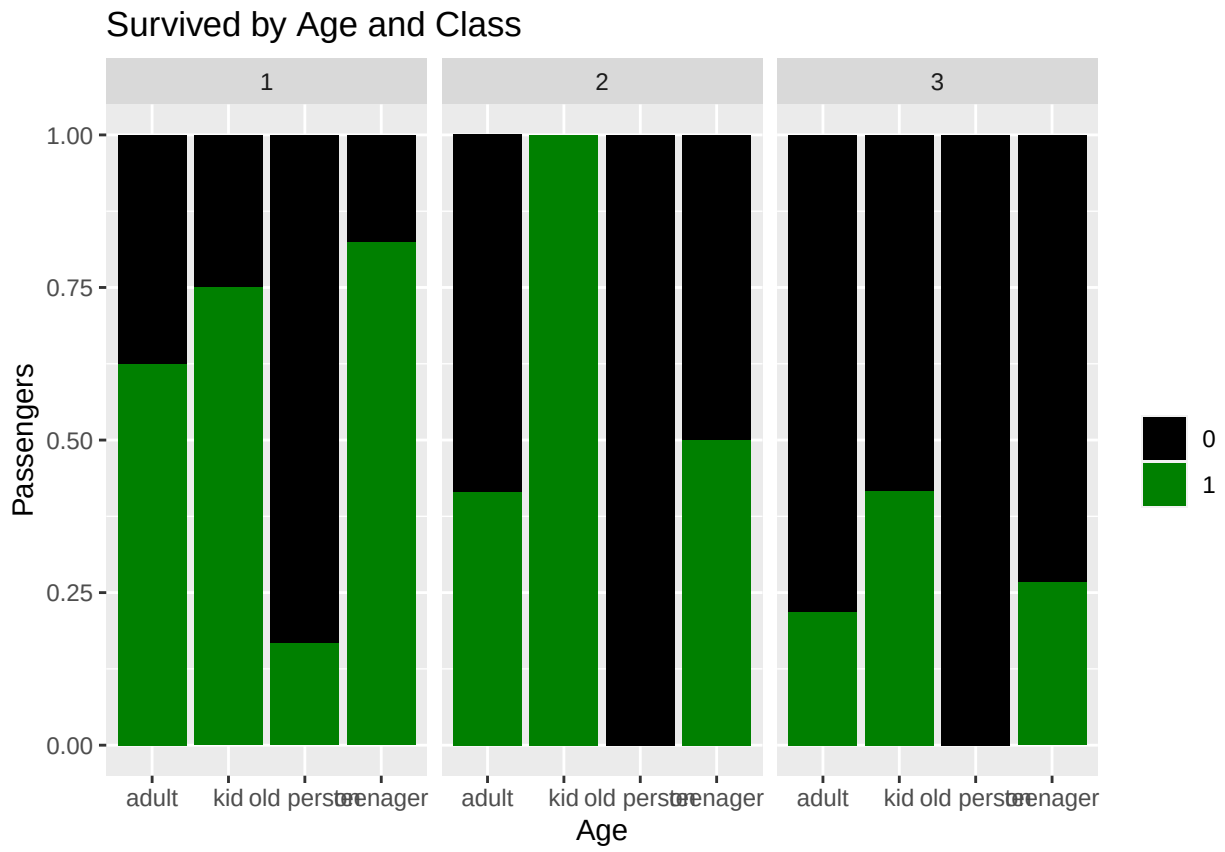
	Alone	Pair	Team
0	41.98	8.08	11.56
1	18.29	9.99	10.10

En los resultados obtenidos vemos que la mayoría de pasajeros que fallecieron iban solos. Podemos comprobar que para los que íban solos el porcentaje de fallecidos fue bastante superior al de supervivientes, mientras que para los que iban en familia los porcentajes de supervivencia y fallecimiento fueron bastante similares.

Survived vs Age vs Pclass

Hemos visto que los niños tuvieron más probabilidades de sobrevivir, vamos a estudiar si esas probabilidades disminuyen según la clase.

```
ggplot(data = data_rep,aes(x=Age_cat,fill=Survived))+geom_bar(position="fill")+facet_wrap(~Pclass)+labs
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t7<-ftable(xtabs(~data_rep$Survived+data_rep$Pclass+data_rep$Age_cat),data_rep)
t7
```

```
##                                 data_rep$Age_cat adult kid old person teenager
## data_rep$Survived data_rep$Pclass
## 0                 1                                71   1          5         3
##                   2                                86   0          2         9
##                   3                               297  28          3        44
## 1                 1                               118   3          1        14
##                   2                                61  17          0         9
##                   3                                83  20          0        16
```

```
# t7 %>% kable() %>% kable_styling()

# Tabla de frecuencias relativas
t7_2 <- round(prop.table(x=t7)*100,2)
t7_2
```

```
##                                    data_rep$Age_cat adult   kid old person teenager
## data_rep$Survived data_rep$Pclass
## 0                 1                           7.97  0.11        0.56     0.34
##                   2                           9.65  0.00        0.22     1.01
##                   3                          33.33  3.14        0.34     4.94
## 1                 1                          13.24  0.34        0.11     1.57
##                   2                           6.85  1.91        0.00     1.01
##                   3                           9.32  2.24        0.00     1.80
```

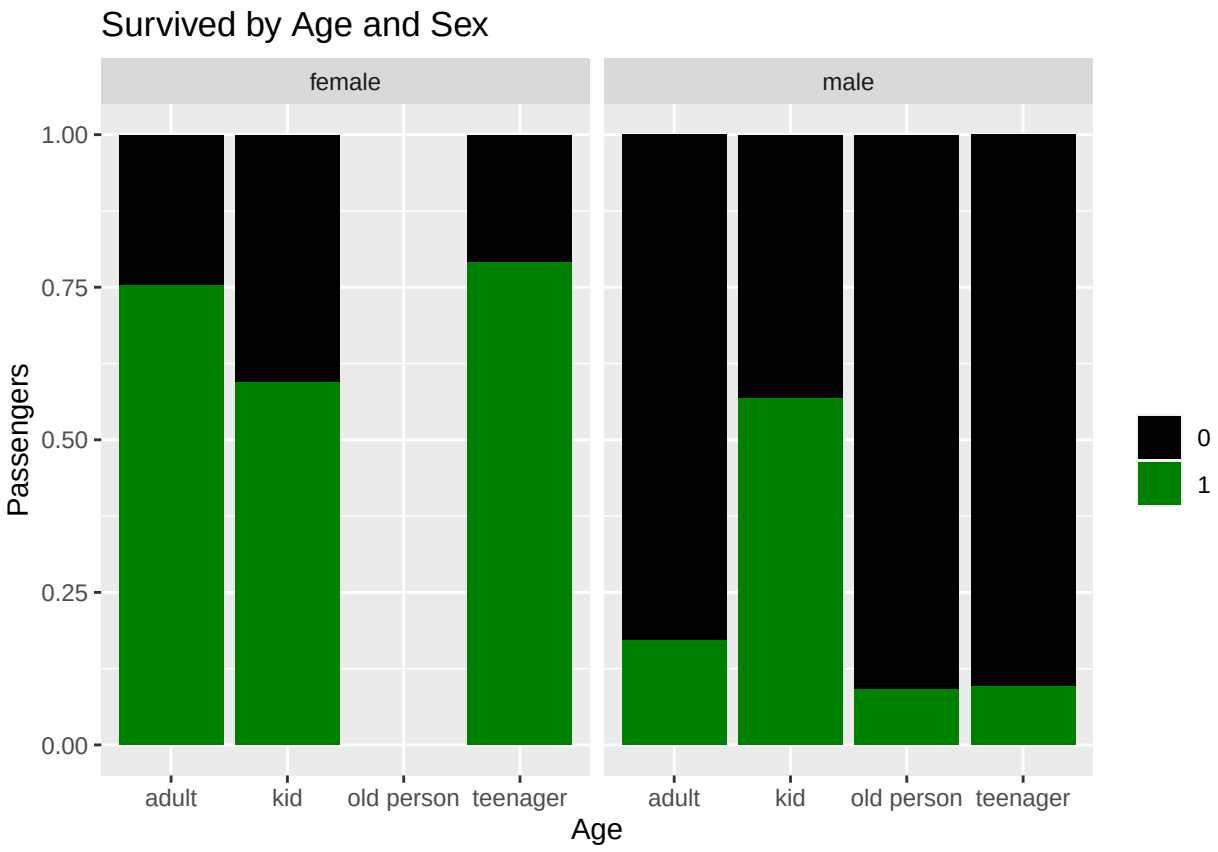
```
# t7_2 %>% kable() %>% kable_styling()
```

Como podemos observar, los niños tuvieron preferencia para subir a los botes salvavidas pero las probabilidades de subir eran menores si pertenecían a la tercera clase. Vemos que prácticamente todos los niños fallecidos pertenecían a la tercera clase. De hecho, los niños pertenecientes a segunda se salvaron todos.

Survived vs Age vs Sex

Otra variable que hemos visto que tuvo mucha importancia es el Sexo, salvándose muchas más mujeres que hombres. Vamos a estudiar también si entre los niños que se salvaron había diferencias en cuanto al sexo.

```
ggplot(data = data_rep,aes(x=Age_cat,fill=Survived))+geom_bar(position="fill")+facet_wrap(~Sex)+labs(x="
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t8<-ftable(xtabs(~data_rep$Survived+data_rep$Sex+data_rep$Age_cat),data_rep)
t8
```

```
##                                 data_rep$Age_cat adult kid old person teenager
## data_rep$Survived data_rep$Sex
## 0                 female                          59  13          0        9
##                   male                           395  16         10       47
## 1                 female                         180  19          0       34
##                   male                            82  21          1        5
```

```
# t8 %>% kable() %>% kable_styling()

# Tabla de frecuencias relativas
t8_2 <- round(prop.table(x=t8)*100,2)
t8_2
```

```
##                                 data_rep$Age_cat adult   kid old person teenager
## data_rep$Survived data_rep$Sex
## 0                 female                         6.62  1.46       0.00     1.01
##                   male                          44.33  1.80       1.12     5.27
## 1                 female                        20.20  2.13       0.00     3.82
##                   male                           9.20  2.36       0.11     0.56
```

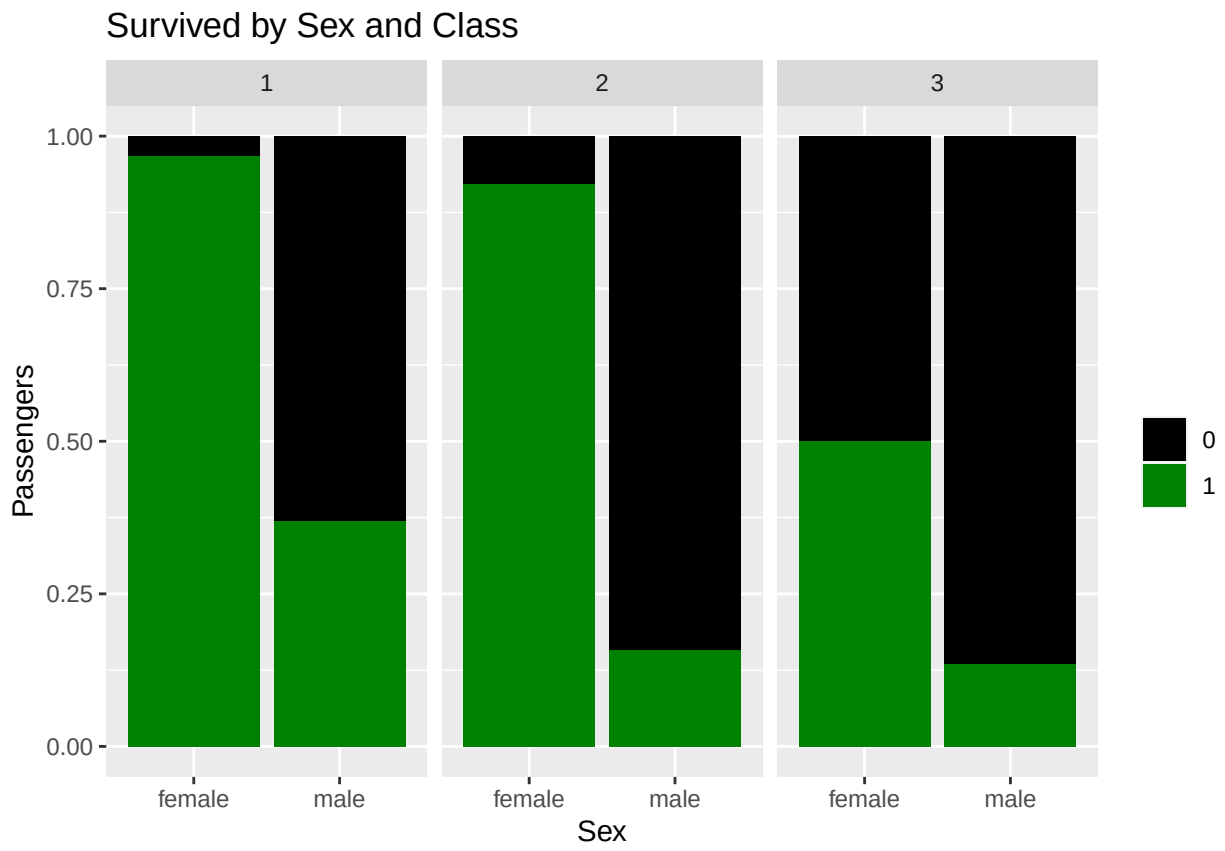
```
# t8_2 %>% kable() %>% kable_styling()
```

Sí hemos visto que la clase influyó mucho en los niños que fallecieron pero vemos que no ocurre los mismo con la variable sexo. El porcentaje de niños fallecidos es muy similar para ambos sexos, lo mismo con el porcentaje de supervivientes.

Survived vs Sex vs Pclass

Como hemos visto previamente, las mujeres tuvieron muchas más probabilidades de sobrevivir que los hombres. Pero, al igual que hemos hecho con los niños, vamos a ver si estas probabilidades se reducían con la clase.

```
ggplot(data = data_rep,aes(x=Sex,fill=Survived))+geom_bar(position="fill")+facet_wrap(~Pclass)+labs(x="S
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t9<-ftable(xtabs(~data_rep$Survived+data_rep$Sex+data_rep$Pclass),data_rep)
t9
```

```
##                                 data_rep$Pclass   1   2   3
## data_rep$Survived data_rep$Sex
## 0                 female                         3   6  72
##                   male                          77  91 300
## 1                 female                        91  70  72
##                   male                          45  17  47
```

```
# t9 %>% kable() %>% kable_styling()

# Tabla de frecuencias relativas
t9_2 <- round(prop.table(x=t9)*100,2)
t9_2
```

```
##                                 data_rep$Pclass     1     2     3
## data_rep$Survived data_rep$Sex
## 0                 female                         0.34  0.67  8.08
##                   male                           8.64 10.21 33.67
## 1                 female                        10.21  7.86  8.08
##                   male                           5.05  1.91  5.27
```

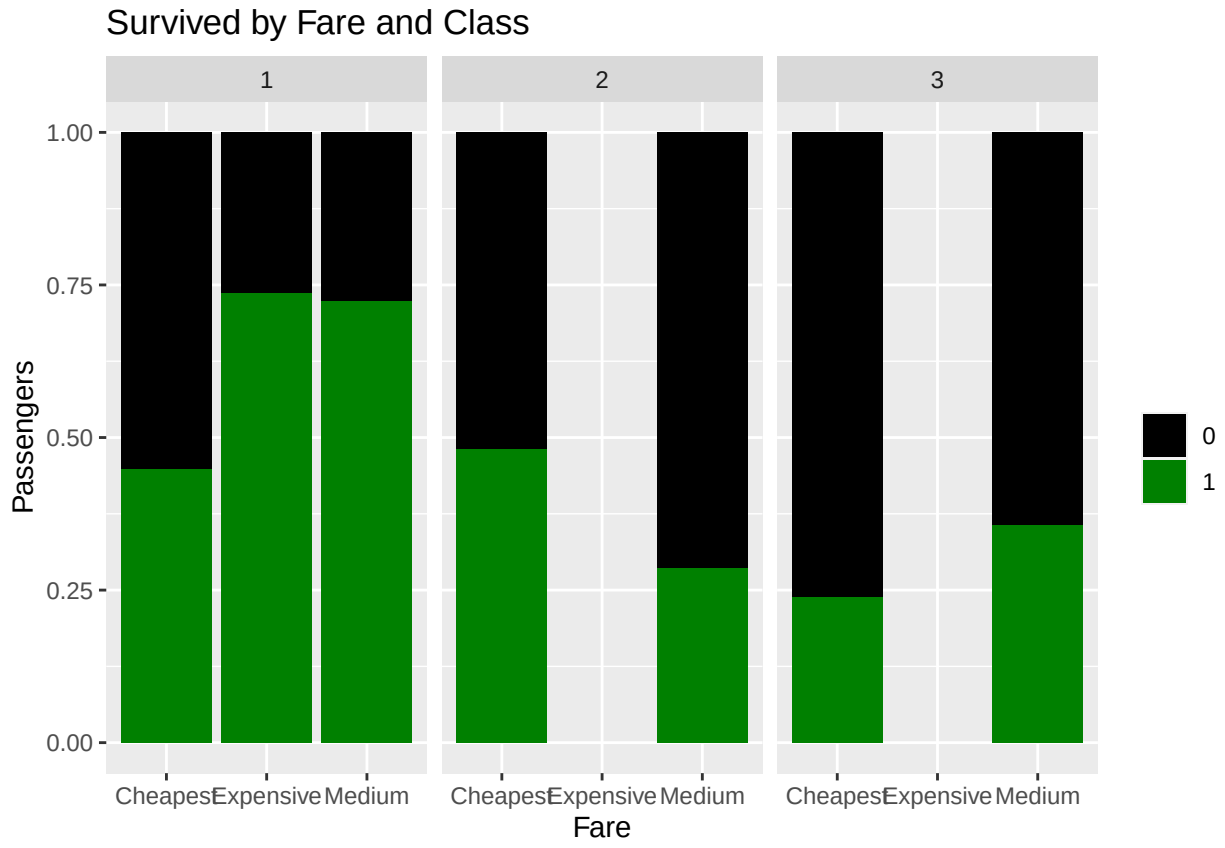
```
# t9_2 %>% kable() %>% kable_styling()
```

Podemos verificar que así fue, aunque el porcentaje de supervivientes no difiere mucho entre las mujeres de las distintas clases, sí podemos observar que en el caso de los fallecimientos es bastante superior. Es decir, ser mujer te facilitaba el acceso a los botes salvavidas pero las probabilidades de fallecer si eras una mujer de tercera clase eran bastante mayores que si eras de segunda o primera clase.

Fare vs Survived vs Pclass

Estudiamos la relación entre la supervivencia, el precio del billete y la clase.

```
ggplot(data = data_rep,aes(x=Fare_cat,fill=Survived))+geom_bar(position="fill")+facet_wrap(~Pclass)+labs
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t10<-ftable(xtabs(~data_rep$Survived+data_rep$Fare_cat+data_rep$Pclass),data_rep)
t10
```

```
##                                   data_rep$Pclass   1   2   3
## data_rep$Survived data_rep$Fare_cat
## 0                 Cheapest                       42  92 363
##                   Expensive                      14   0   0
##                   Medium                         24   5   9
## 1                 Cheapest                       34  85 114
##                   Expensive                      39   0   0
##                   Medium                         63   2   5
```

```
# t10 %>% kable() %>% kable_styling()

# Tabla de frecuencias relativas
t10_2 <- round(prop.table(x=t10)*100,2)
t10_2
```

```
##                                      data_rep$Pclass     1     2     3
## data_rep$Survived data_rep$Fare_cat
## 0                 Cheapest                          4.71 10.33 40.74
##                   Expensive                         1.57  0.00  0.00
##                   Medium                            2.69  0.56  1.01
## 1                 Cheapest                          3.82  9.54 12.79
##                   Expensive                         4.38  0.00  0.00
##                   Medium                            7.07  0.22  0.56
```

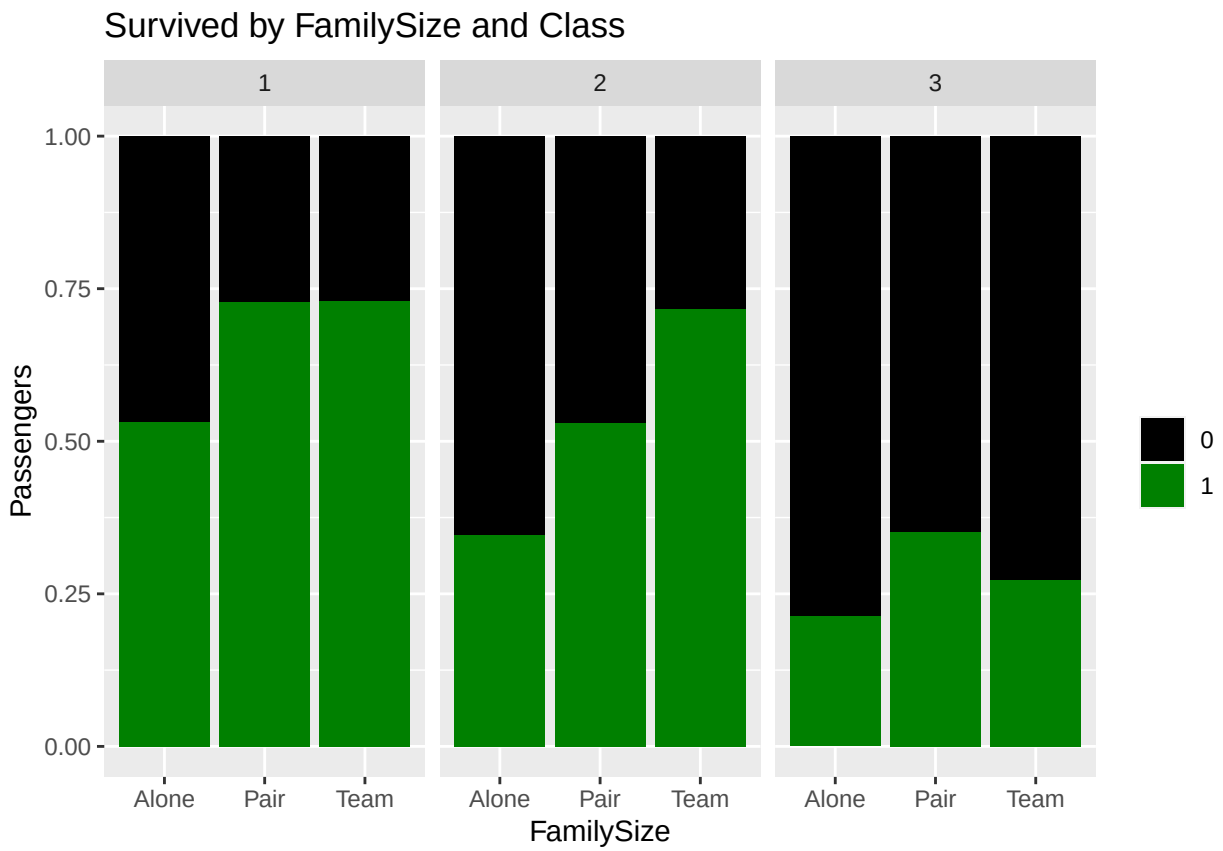
```
# t10_2 %>% kable() %>% kable_styling()
```

En este caso el resultado era de esperar ya que el precio del billete y la clase son dos variables que hemos visto que presentan una alta correlación. Por tanto, podemos ver que la mayoría de fallecidos pertenecían a tercera clase y pagaron un billete barato.

FamilySize vs Survived vs Pclass

Anteriormente hemos visto que aquellos que viajaban solos tenían menos probabilidad de sobrevivir que los que iban en familia. Vamos a ver como influyó el hecho de pertenecer a una clase o a otra.

```
ggplot(data = data_rep,aes(x=FamilySize_cat,fill=Survived))+geom_bar(position="fill")+facet_wrap(~Pclass
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t11<-ftable(xtabs(~data_rep$Survived+data_rep$FamilySize_cat+data_rep$Pclass),data_rep)
t11
```

```
##                                          data_rep$Pclass   1   2   3
## data_rep$Survived data_rep$FamilySize_cat
## 0                 Alone                                  51  68 255
##                   Pair                                   19  16  37
##                   Team                                   10  13  80
## 1                 Alone                                  58  36  69
##                   Pair                                   51  18  20
##                   Team                                   27  33  30
```

```
# t11 %>% kable() %>% kable_styling()

# Tabla de frecuencias relativas
t11_2 <- round(prop.table(x=t11)*100,2)
t11_2
```

```
##                                          data_rep$Pclass     1     2     3
## data_rep$Survived data_rep$FamilySize_cat
## 0                 Alone                                  5.72  7.63 28.62
##                   Pair                                   2.13  1.80  4.15
##                   Team                                   1.12  1.46  8.98
## 1                 Alone                                  6.51  4.04  7.74
##                   Pair                                   5.72  2.02  2.24
##                   Team                                   3.03  3.70  3.37
```

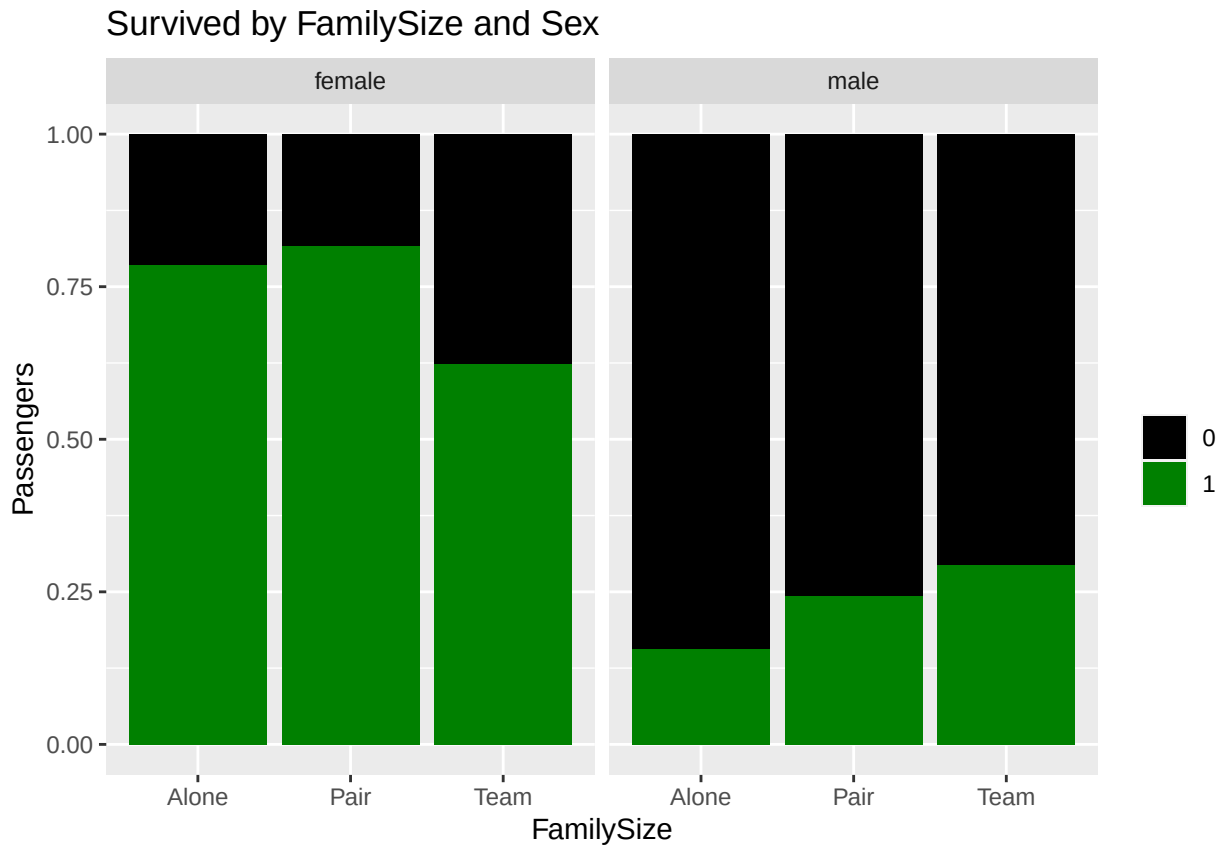
```
# t11_2 %>% kable() %>% kable_styling()
```

Como podemos ver en los resultados, el porcentaje de fallecimientos fue superior para aquellos que viajaban solos pero, además, este porcentaje se incrementa considerablemente para aquellos que iban solos y llevaban un billete de tercera clase. De hecho, el porcentaje de fallecimientos de los que iban solos y pertenecían a tercera clase fue el el doble que los porcentajes de los que iban solos y en primera o segunda clase juntos. Esto no ocurre con el porcentaje de supervivientes de los que iban solos, en este caso no se observa una diferencia considerable entre las diferentes clases.

FamilySize vs Survived vs Sex

Finalmente, vamos a estudiar también la relación entre la supervivencia, el tamaño de la familia y el sexo.

```
ggplot(data = data_rep,aes(x=FamilySize_cat,fill=Survived))+geom_bar(position="fill")+facet_wrap(~Sex)+
```



Obtenemos las tablas de contingencia.

```
# Tabla de frecuencias absolutas
t12<-ftable(xtabs(~data_rep$Survived+data_rep$FamilySize_cat+data_rep$Sex),data_rep)
t12
```

```
##                                        data_rep$Sex female male
## data_rep$Survived data_rep$FamilySize_cat
## 0                 Alone                                  27  347
##                   Pair                                   16   56
##                   Team                                   38   65
## 1                 Alone                                  99   64
##                   Pair                                   71   18
##                   Team                                   63   27
```

```
# t12 %>% kable() %>% kable_styling()

# Tabla de frecuencias relativas
t12_2 <- round(prop.table(x=t12)*100,2)
t12_2
```

```
##                                        data_rep$Sex female  male
## data_rep$Survived data_rep$FamilySize_cat
## 0                 Alone                             3.03 38.95
##                   Pair                              1.80  6.29
```

```
##                  Team                                4.26  7.30
## 1                Alone                              11.11  7.18
##                  Pair                                7.97  2.02
##                  Team                                7.07  3.03
```

```
# t12_2 %>% kable() %>% kable_styling()
```

Hemos visto que aquellos que iban solos tenían más probabilidades de fallecer. Sin embargo, si tenemos en cuenta el sexo vemos que ésto es cierto para el caso de los hombres. El porcentaje de hombres que viajaban solos y fallecieron es el más elevado. Sin embargo, para el caso de las mujeres no existen grandes diferencias en el porcentaje de fallecimientos, siendo éste similar para aquellas mujeres que iban solas o en familia.

Conclusiones

A lo largo de esta práctica se han llevado a cabo una serie de análisis que nos han permitido responder a las preguntas planteadas inicialmente.

Mujeres y niños primero

La gran afirmación sobre el Titanic ha resultado ser cierta. Las mujeres y niños sí tuvieron prioridad a la hora de salvarse. De hecho, el Sexo y después la Edad son dos de las variables más importantes, como hemos podido comprobar también en el árbol de decisión.

El porcentaje de mujeres que sobrevivieron dobla al de hombres. Hemos podido comprobar que más de la mitad de los hombres que iban a bordo fallecieron.

Los niños también fueron el único grupo de edad en el que hubo más supervivientes que fallecidos, algo que no ocurrió en el resto de grupos de edad, siendo el grupo de los adultos el que se vio más afectado. No podemos saber si los ancianos se encontraron con más dificultades, debido a su avanzada edad, a la hora de acceder a los botes salvavidas, pero sí podemos ver que la tasa de supervivencia es muy reducida.

También podemos comprobar que los niños se salvaron en el mismo rango de probabilidades indepdendientemente de si eran niños o niñas, mientras que las mujeres (adolescentes o adultas) tuvieron muchas más opciones que sus homologos masculinos.

Hemos verificado que las mujeres y niños tuvieron más probabilidades de salvarse, pero ¿cómo se relaciona la clase con estas variables? Podemos observar que de los niños que murieron la gran mayoría pertenecían a tercera clase y lo mismo ocurre con las mujeres. Aun así, hemos podido comprobar que la cantidad de mujeres que fallece perteneciente a tercera clase es inferior a la cantidad de hombres que sobrevive de primera clase.

Pero la clase importa

Tal y como siempre se ha contado la clase fue una variable importante en cuanto a las probabilidades de supervivencia. Los análisis han demostrado que hubo más pasajeros de 1ª clase que se salvaron que pasajeros de 2ª y 3ª clase. De hecho, el porcentaje de fallecidos para los pasajeros de segunda y tercera clase es superior al porcentaje de supervivientes, mientras que para los de primera clase fue al contrario.

Hemos podido comprobar que si pagabas un billete caro las probabilidades de sobrevivir se incrementaban, ya que la mayoría de fallecidos habían embarcado con el billete más barato. Esto era de esperar ya que el precio del billete está fuertemente correlacionado con la clase.

Puerto de embarque

Hemos comprobado que el puerto de embarque tenía menos relevancia. Aunque existen diferencias entre ellos, una gran parte del pasaje embarcó en Southampton y entre ellos el porcentaje de fallecidos fue muy alto. Mientras que aquellos pasajeros que embarcaron en Cherbourg fueron los únicos cuya tasa de supervivencia

fue superior a la de los fallecidos. Esto hecho parece estar correlacionado con la clase, ya que es el único puerto en el que embarcaron más pasajeros de primera que del resto de clases.

En los otros dos puertos embarcaron más pasajeros de tercera clase que de primera y segunda.

Viajar solo o en familia

Finalmente, hemos visto que hubo muchos más fallecidos viajando solos que en familia, pero la importancia de esta variable es mucho menor que el sexo o la clase.

Tabla de integrantes

```
contribuciones <- c("Investigación previa", "Redacción de las respuestas","Desarrollo código")
firma <- c("S.D.A., S.R.E.","S.D.A., S.R.E.", "S.D.A., S.R.E.")

info <- data.frame( contribuciones, firma )
info %>% kable() %>% kable_styling()
```

contribuciones	firma
Investigación previa	S.D.A., S.R.E.
Redacción de las respuestas	S.D.A., S.R.E.
Desarrollo código	S.D.A., S.R.E.