PRA2 - Limpieza y análisis de datos

sguillen1 – Sandra Guillén Resina
cperezceb – Carlos Pérez Cebrián

—

- Descripción del dataset
- Integración y selección de los datos de interés a analizar.
- Lectura del dataset
- Limpieza de los datos.
    - Varible *Name*
    - Variable *SibSp* y *Parch*
    - Variable *Survived*
    - Variable *PClass*
    - Variable *Sex*
    - Variable *Embarked*
    - Variable *PrecioTicket*
- Selección de datos
- Valores vacios y outliers
    - Varible *Fare*
    - variable *Embarked*
    - Variable *Age*
- Valores extremos
    - Variable *Age*
    - Variable *Fare*
    - Variable *Famsize*
- Exportación de los datos
- Análisis de los datos
- Selección de los datos a comparar
- Relaciones de las variables independientes respecto la variable *Survived*
    - Comparativa entre las variables Sex y Survived
    - Comparativa entre las variables Pclass y Survived
    - Comparativa entre las variables Embarked y Survived
    - Comparativa entre las variables Fsize y Survived
    - Comparativa entre las variables Age y Survived
    - Comparativa entre las variables Fare y Survived
- Relaciones de dos variables independientes con la variable dependiente
    - Comparativa de las variables Sex y Pclass con la variable Survived
    - Comparativa de las variables Sex y Fsize con la variable Survived
    - Comparativa de las variables Sex y Embarked con la variable Survived
    - Comparativa de las variables Sex y Age con la variable Survived
    - Comparativa de las variables Sex y Fare con la variable Survived
- Comprobación de la normalidad y homogeneidad de la varianza
    - Normalidad
    - Homegeneidad de la varianza
- Apliación de pruebas estadísticas para comparar los grupos de datos.
    - Contraste de hipótesis
    - Corelación
    - Regresión Logística
    - Random forest
- Representacó de los resultados a partir de tablas y gráficas
- Resolución del problema
- Recursos

Requeriremos de los paquetes ggplot2, gridExtra y grid de R.

Descripción del dataset

Para el desarrollo de esta práctica se ha optado por la elección del dataset: “Titanic: Machine Learning from Disaster (https://www.kaggle.com/c/titanic (https://www.kaggle.com/c/titanic))”. Este dataset contiene información de uno de los naufragios más conocido de la historia, donde se tienen datos relativos a sus pasajeros como edad, sexo, clase en que viajaban, ... Este dataset es muy utilizado para el entrenamiento de algoritmos, y actualmente forma parte de una competición de Kaggle.

A partir del análisis de este dataset, se pretende averiguar si existe alguna variable que influya en la supervivencia o no.Así como conocer qué tipo de personas en bases a sus características físicas y las de su viaje tenían más probabilidades se sobrevivir.

La actividad se centra en el tratamiento de un dataset. En primer lugar, tendremos una fase de estudio previo para conocer cómo son nuestros datos, su estructura y su comportamiento. Una segunda fase de preparación, dónde aplicaremos la técnicas necesarias para la adecuación de los datos para la siguiente fase de análisis y, finalmente, y extraeremos conclusiones.

Disponemos de tres conjuntos de datos que se describen a continuación:

- train.csv: Es el que debe usarse para construir los modelos de aprendizare automático. Se porporciona el resultado de la variable *Survived* que incida la superivivencia de cada pasajero que es sobre la que se va a llevar a cabo la práctica. Contiene un total de 891 observaciones.
- test.csv: Se compone de un total de 418 observaciones y es el que debe usarse para ver la precisión del modelo generado con el dataset anterior. No se proporciona la variable *Survived*.
- gender_submission.csv: Es el que contiene los valores de la variable *Survived* para cada pasajero.

Disponemos de un total de 1309 muestras y 12 variables distintas; 5 de tipo carácter, 5 de tipo integer y 2 numéricas. A continuación, se va a llevar a cabo la descripción de los atributos que forman parte del dataset:

- **PassengerId**: Identificador del pasajero o tripulante.
- **Survived**: Indica si el pasajero o tripulante ha sobrevivido. El valor *0* indica que no ha sobrevivido y el *1* que sí. Es la variable que se trata de predecir en el conjunto de test.
- **Pclass**: Es la clase en la que ha navegado el pasajero. Existen los siguientes valores: *1* es primera clase, *2* es segunda clase y *3* es tercera clase.
- **Name**: Nombre completo del pasajero.
- **Sex**: Género del pasajerondividuo. Existen los siguientes Valores: *male* para hombre y *female* para mujer.
- **Age**: Edad en años del pasajero.
- **SibSp**: Número de hermanos o cónyuges a bordo del barco.
- **Parch**: Número de padres o hijos a bordo del barco.
- **Ticket**: El indentificador del ticket. Todos los miembros de una familia tendrán el mismo identificador de ticket, un único billete.
- **Fare**: Tarifa del billete.
- **Cabin**: Número de cabina.
- **Embarked**: Puerto en el que ha embarcado el pasajero. Exiten los siguientes valores: *C* corresponde a Cherbourg, *Q* corresponde a Queenstown y *S* corresponde a Southampton.

Integración y selección de los datos de interés a analizar.

Lectura del dataset

En este aparatado se va a llevar acabo la integración y la selección de los datos que van a ser de interés para llevar a cabo el análisis.

Como se ha comentado en el apartado anterior, hay tres conjunto de datos. Cada conjunto tiene un identificador único que es la variable *PassengerId* que permite relacionar

```
# Carga de los diferentes datasets

dataTrain<-read.csv("./data/train.csv",header=T,sep=",")
dataTest <-read.csv("./data/test.csv",header=T,sep=",")
dataReferencias <-read.csv("./data/gender_submission.csv",header=T,sep=",")


# Unimos los tres dataset en uno

dataTest <- merge(dataTest, dataReferencias, by="PassengerId")

data <- rbind(dataTrain, dataTest)
```

Una vez se ha unificado los datos, se inspeccionará el conjuntos de datos:

```
# Observamos la estructura de los datos

str(data)
```

```
## 'data.frame':    1309 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)" "Heikkinen,
Miss. Laina" "Futrelle, Mrs. Jacques Heath (Lily May Peel)" ...
##  $ 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" ...
```

```
summary(data)
```

```
##   PassengerId      Survived          Pclass          Name
##  Min.   :   1   Min.   :0.0000   Min.   :1.000   Length:1309
##  1st Qu.: 328   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median : 655   Median :0.0000   Median :3.000   Mode  :character
##  Mean   : 655   Mean   :0.3774   Mean   :2.295
##  3rd Qu.: 982   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :1309   Max.   :1.0000   Max.   :3.000
##
##      Sex                 Age            SibSp            Parch
##  Length:1309        Min.   : 0.17   Min.   :0.0000   Min.   :0.000
##  Class :character   1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.000
##  Mode  :character   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
##     Ticket              Fare            Cabin              Embarked
##  Length:1309        Min.   :  0.000   Length:1309        Length:1309
##  Class :character   1st Qu.:  7.896   Class :character   Class :character
##  Mode  :character   Median : 14.454   Mode  :character   Mode  :character
##                     Mean   : 33.295
##                     3rd Qu.: 31.275
##                     Max.   :512.329
##                     NA's   :1
```

Limpieza de los datos.

Varible *Name*

En primer lugar analizaremos la variables *Name*:

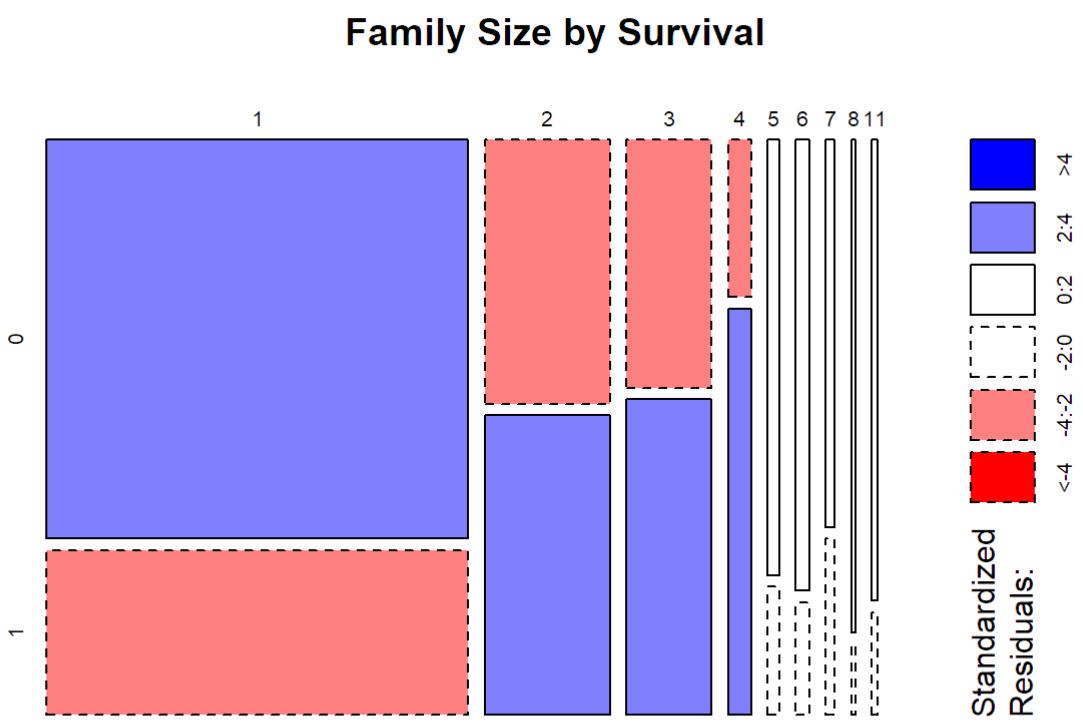
```
data$Title <- gsub('(.*, )|(\\..*)', '', data$Name)
rare_title <- c('Dona', 'Lady', 'the Countess','Capt', 'Col', 'Don',
'Dr', 'Major', 'Rev', 'Sir', 'Jonkheer')
data$Title[data$Title == 'Mlle'] <- 'Miss'
data$Title[data$Title == 'Ms'] <- 'Miss'
data$Title[data$Title == 'Mme'] <- 'Mrs'
data$Title[data$Title %in% rare_title] <- 'Rare Title'
data$Title <- as.factor(data$Title)
data$Surname <- sapply(data$Name,
function(x) strsplit(x, split = '[,.]')[[1]][1])

data$Surname <- as.factor(data$Surname)
```

Variable *SibSp* y *Parch*

En este apartado se combinarán la variables *SibSp* y *Parch* para crear una nueva variables llamada *Famsize*.

```
data$Famsize <- data$SibSp + data$Parch + 1
```

```
data$Fsize[data$Famsize == 1] <- 'singleton'
data$Fsize[data$Famsize < 5 & data$Famsize > 1] <- 'small'
data$Fsize[data$Famsize > 4] <- 'large'

data$Fsize <- as.factor(data$Fsize)


mosaicplot(table(data$Famsize, data$Survived), main='Family Size by Survival', shade=TRUE)
```



Variable *Survived*

En primer lugar se convertirán a factores las variables *Survived*, *PClass*, *Sex* y *Embarked*.

```
data$Survived <- as.factor(data$Survived)
```

Variable *PClass*

```
data$Pclass <- as.factor(data$Pclass)
```

Variable *Sex*

```
data$Sex <- as.factor(data$Sex)
```

Variable *Embarked*

```
data$Embarked <- as.factor(data$Embarked)
```

Variable *PrecioTicket*

Creamos una nueva variable con el precio medio pagado por cada pasajero. Se calcula en función del identificador del ticket y el número de personas asociadas a dicho ticket

```
data <- data %>%
          group_by(Ticket) %>%
          mutate(PrecioTicket = Fare/n())
```

Selección de datos

En el siguiente paso se eliminaran aquellas columnas que no contengan información útil para el desarrollo de esta práctica.

```
# Eliminamos las columnas

data <- subset(data, select = -c(PassengerId, Ticket, Cabin))
```

Valores vacios y outliers

En este apartado se analizará si las variables contienen valores nulos o incompletos.

```
col_mis <- colSums(is.na(data) | data=="")
print(col_mis)
```

```
##     Survived       Pclass         Name          Sex          Age        SibSp
##            0            0            0            0          263            0
##        Parch         Fare     Embarked        Title      Surname      Famsize
##            0            1            2            0            0            0
##        Fsize PrecioTicket
##            0            1
```

A continuación se trataran las variables con valores nulos o incompletos de las diferetnes variables.

Varible *Fare*

Antes de imputar los valores perdidos de la varible *Fare* se comprobará las características de estos valores.

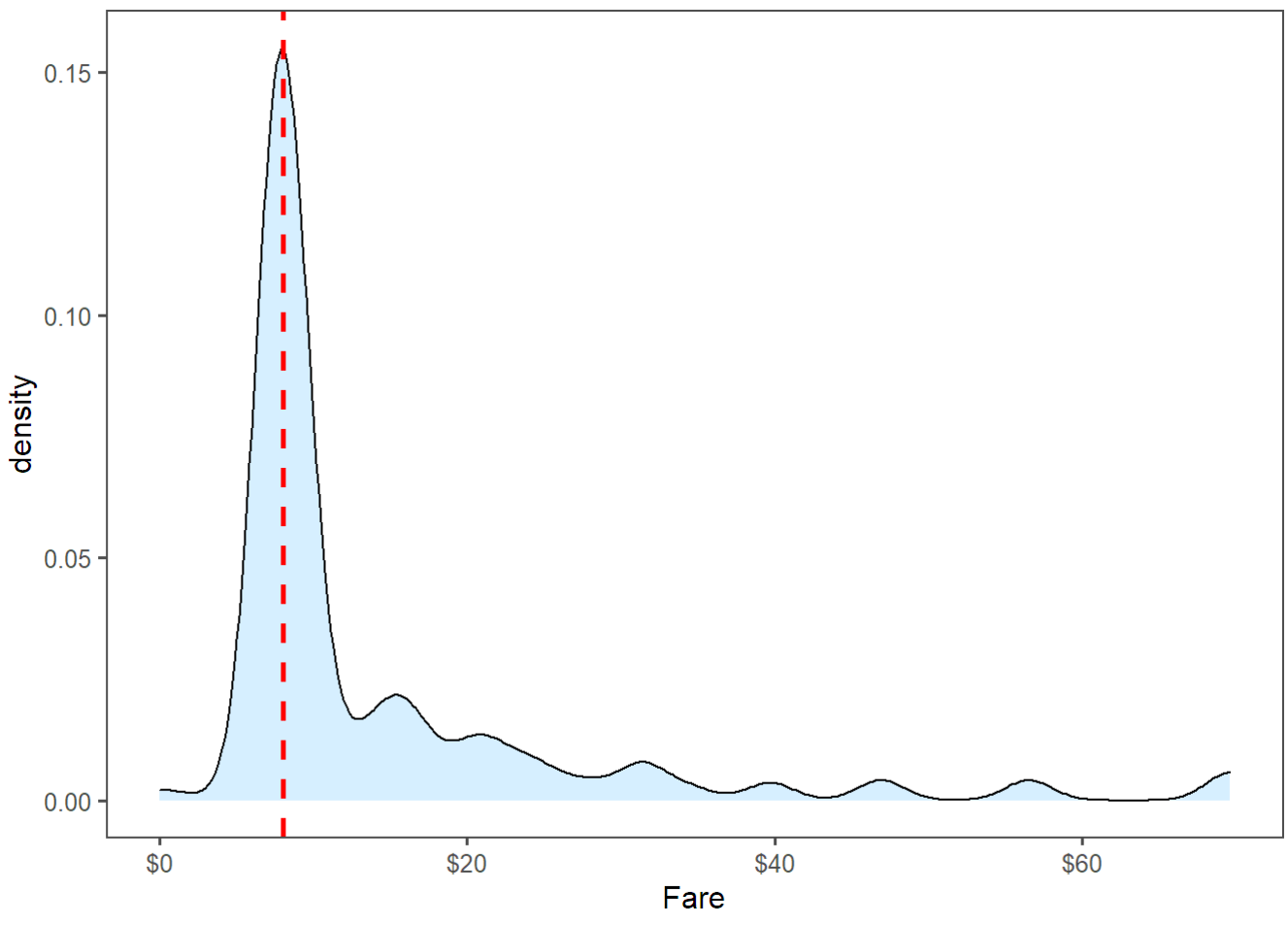
```
miss_fare_index <- which(is.na(data$Fare))
miss_fare <- data[miss_fare_index,]
miss_fare
```

```
## # A tibble: 1 x 14
##   Survived Pclass Name      Sex     Age SibSp Parch  Fare Embarked Title Surname
##   <fct>    <fct>  <chr>     <fct> <dbl> <int> <int> <dbl> <fct>    <fct> <fct>
## 1 0        3      Storey, ~ male   60.5     0     0    NA S        Mr    Storey
## # ... with 3 more variables: Famsize <dbl>, Fsize <fct>, PrecioTicket <dbl>
```

Se visualiza los valores de la variable *Fare* para los otros pasajeros que comparten los mismo valores para las variables *PClass* y *Embarked*.

```
ggplot(data[data$Pclass == '3' & data$Embarked == 'S',],
       aes(x = Fare)) +
  geom_density(fill = '#99d6ff', alpha=0.4) +
  geom_vline(aes(xintercept=median(Fare, na.rm=T)),
             colour='red', linetype='dashed', lwd=1) +
  scale_x_continuous(labels=dollar_format()) +
  theme_few()
```

```
## Warning: Removed 1 rows containing non-finite values (stat_density).
```



Después de visualizar las gráfica anterior se puede dar por válido el reemplazar el valor perdido de la variable *Fare* por la mediana de su clase y embarque.

```
data$Fare[1044] <- median(data[data$Pclass == '3' & data$Embarked == 'S', ]$Fare, na.rm = TRUE)
```

variable *Embarked*

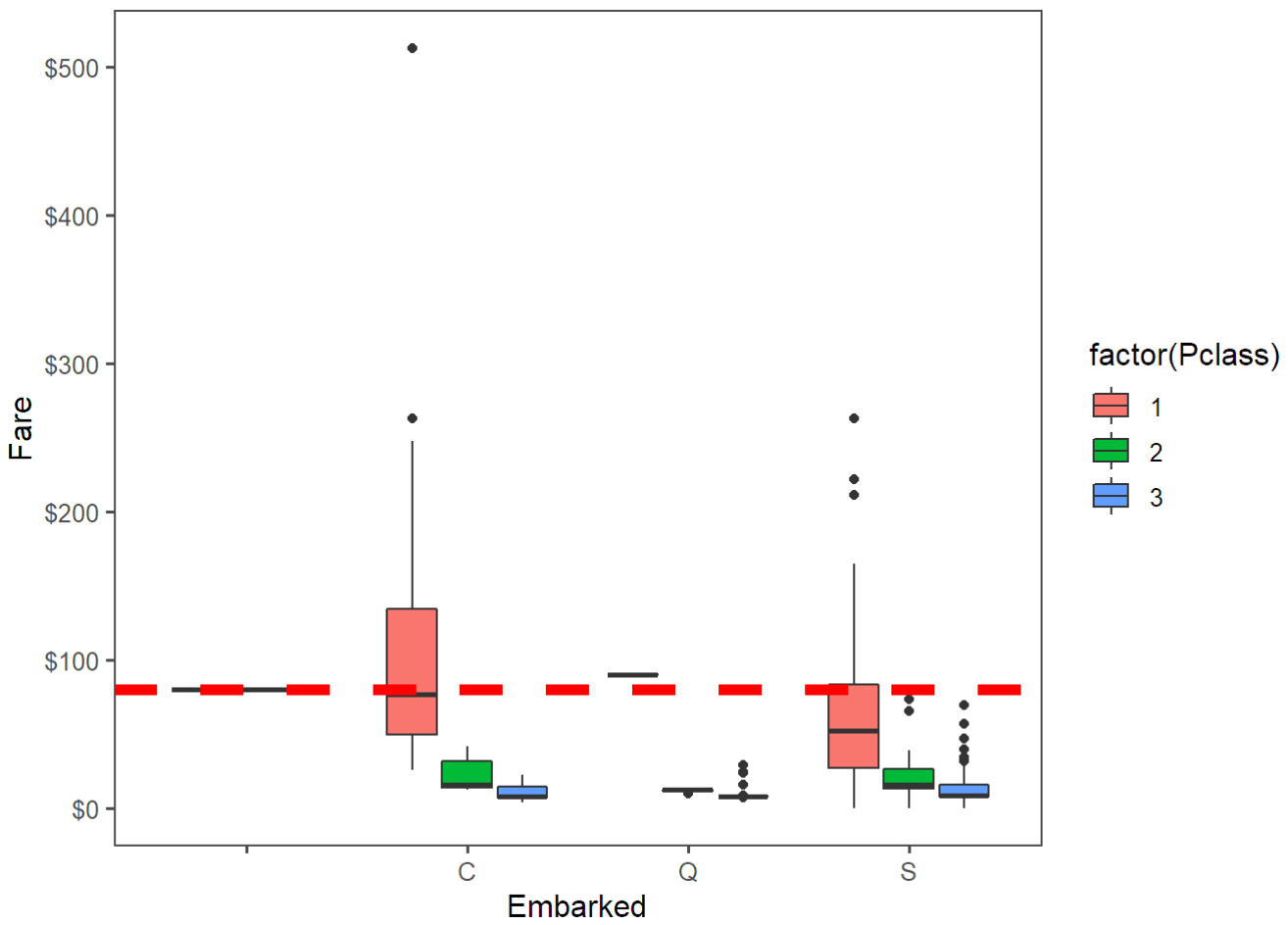
Antes de imputar los valores perdidos de la variable *Embarked* se comprobará las características de estos valores.

```
miss_embark_index <- which(is.na(data$Embarked))
miss_embark <- data[miss_embark_index,]
miss_embark
```

```
## # A tibble: 0 x 14
## # ... with 14 variables: Survived <fct>, Pclass <fct>, Name <chr>, Sex <fct>,
## #   Age <dbl>, SibSp <int>, Parch <int>, Fare <dbl>, Embarked <fct>,
## #   Title <fct>, Surname <fct>, Famsize <dbl>, Fsize <fct>, PrecioTicket <dbl>
```

Se visualiza los valores de la variable *Fare* para los otros pasajeros que comparten los mismo valores para las variables *PClass* y *Fare*.

```
ggplot(data[!is.na(data$Embarked),], aes(x = Embarked, y = Fare, fill = factor(Pclass))) +
  geom_boxplot() +
  geom_hline(aes(yintercept=80),
             colour='red', linetype='dashed', lwd=2) +
  scale_y_continuous(labels=dollar_format()) +
  theme_few()
```



Después de visualizar las gráfica anterior se puede dar por válido el reemplazar el valor perdido de la variable *Embarked* por el valor *C*.

```
data$Embarked[miss_embark_index] <- 'C'
```

Variable *Age*

La variable *Age* contiene un porcentaje mayor de valores nulos. En este caso, vamos a aplicar la imputación por vecinos más cercanos, usando la distancia Gower, considerando en el cómputo de los vecinos más cercanos el resto de variables cuantitativas. Utilizaremos la función *K-Nearest Neighbour Imputation* de la librería *VIM* con un número de vecinos igual a 11.

```
 table(is.na(data$Age))
```

```
##
## FALSE  TRUE
##  1046   263
```

```
if (!require('VIM')){
  install.packages('VIM')
  library(VIM, warn.conflicts = FALSE)
}
```

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

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

```
## VIM is ready to use.
```

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

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

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

```
data<-kNN(
  data,
  variable = "Age",
  k = 11,
  dist_var = c("Pclass","Embarked","Fare"),
  catFun = group_by(Sex),
  imp_var = FALSE
)

#Verificamos que ya no existan valores nulos
 table(is.na(data$Age))
```

```
##
## FALSE
##  1309
```

Aprovechamos para crear una variable que identifique a los pasajeros por rango de edad

```
 data["RangoEdad"] <- cut(data$Age, breaks = c(0,20,40,60,80,100), labels = c("0-19", "20-39","40-59","60-79",">79"
))
str(data)
```

```
## 'data.frame':    1309 obs. of  15 variables:
##  $ 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 ...
##  $ Name        : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)" "Heikkine
n, Miss. Laina" "Futrelle, Mrs. Jacques Heath (Lily May Peel)" ...
##  $ Sex         : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age         : num  22 38 26 35 35 22 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    : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
##  $ Title       : Factor w/ 5 levels "Master","Miss",..: 3 4 2 4 3 3 3 1 4 4 ...
##  $ Surname     : Factor w/ 875 levels "Abbing","Abbott",..: 101 183 335 273 16 544 506 614 388 565 ...
##  $ Famsize     : num  2 2 1 2 1 1 1 5 3 2 ...
##  $ Fsize       : Factor w/ 3 levels "large","singleton",..: 3 3 2 3 2 2 2 1 3 3 ...
##  $ PrecioTicket: num  7.25 35.64 7.92 26.55 8.05 ...
##  $ RangoEdad   : Factor w/ 5 levels "0-19","20-39",..: 2 2 2 2 2 2 3 1 2 1 ...
```

```
table(data$RangoEdad)
```

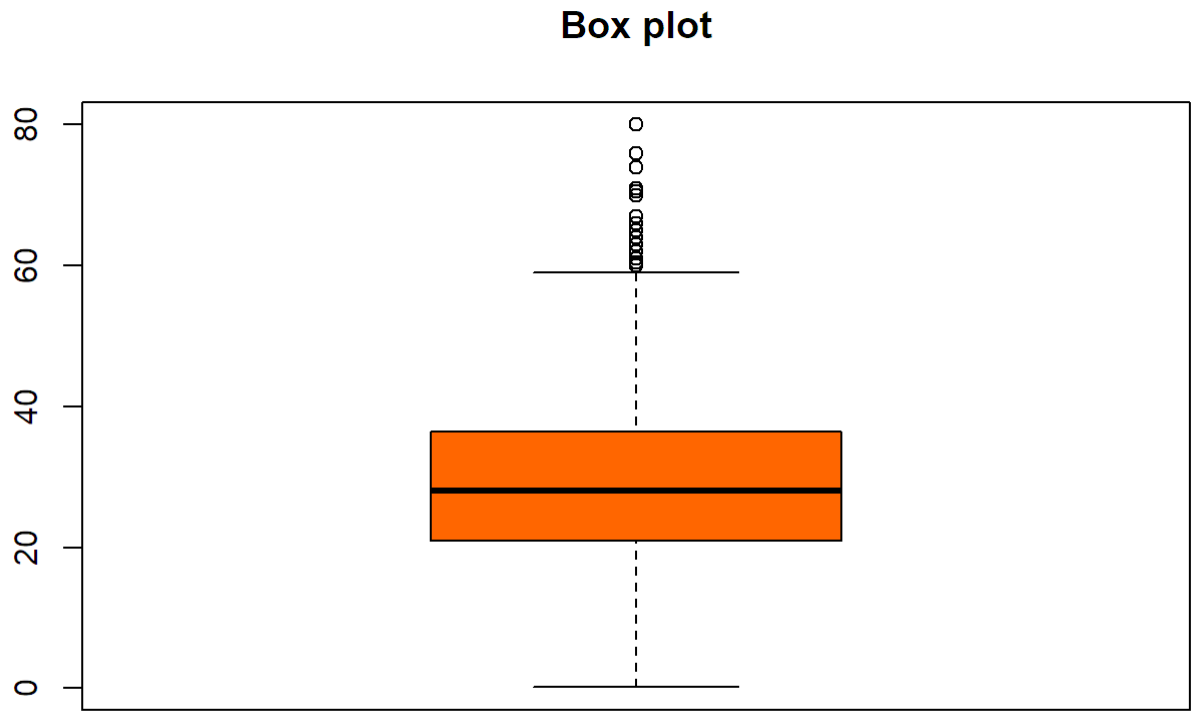
```
##
##  0-19 20-39 40-59 60-79   >79
##   293   758   225    33     0
```

Valores extremos

En este apartado se analizarán los valores extremos de las variables *Age*, *Fare*, *SibSp*, *Parch* y *Fsize*.

Variable *Age*

```
boxplot(data$Age, main="Box plot", col="#FF6600")
```



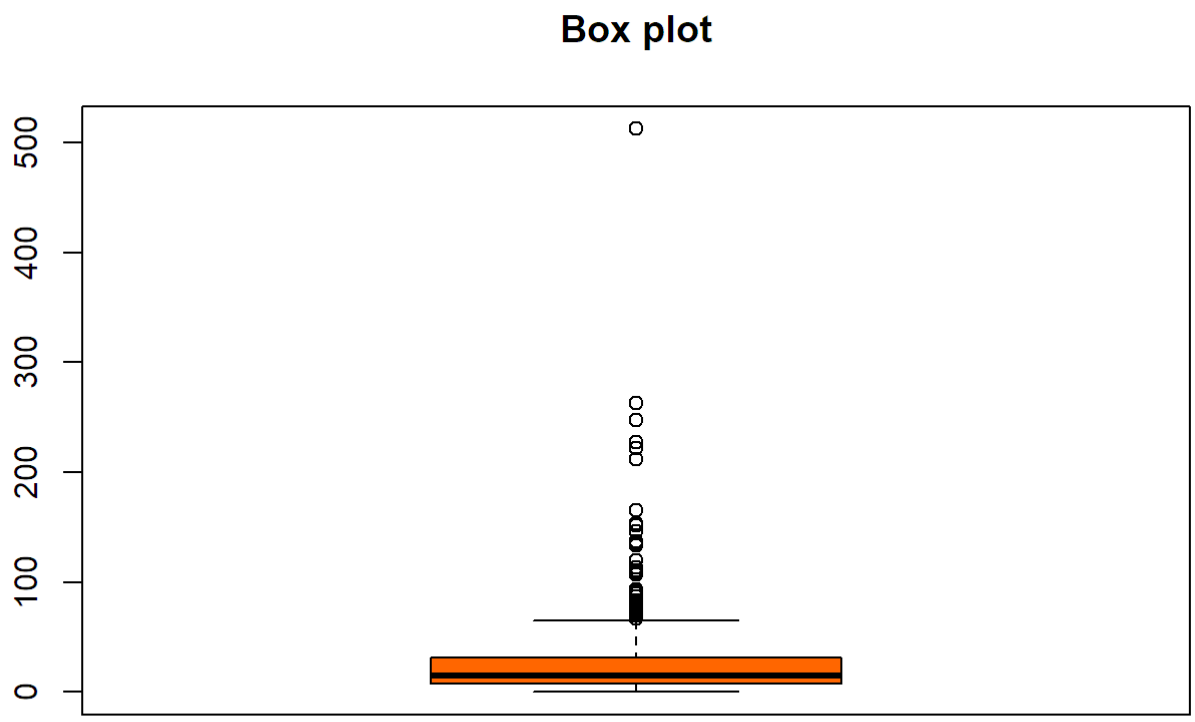

```
boxplot.stats(data$Age)$out
```

```
##  [1] 66.0 65.0 71.0 70.5 61.0 62.0 63.0 65.0 61.0 60.0 64.0 65.0 63.0 71.0 64.0
## [16] 62.0 62.0 60.0 61.0 80.0 70.0 60.0 60.0 70.0 62.0 74.0 62.0 63.0 60.0 60.0
## [31] 67.0 76.0 63.0 61.0 60.5 64.0 61.0 60.0 64.0 64.0
```

Se puede observar que los valores extremos están en un rango normal, ningún pasajero es menor que 0 o mayor que 100.

Variable *Fare*

```
boxplot(data$Fare, main="Box plot", col="#FF6600")
```



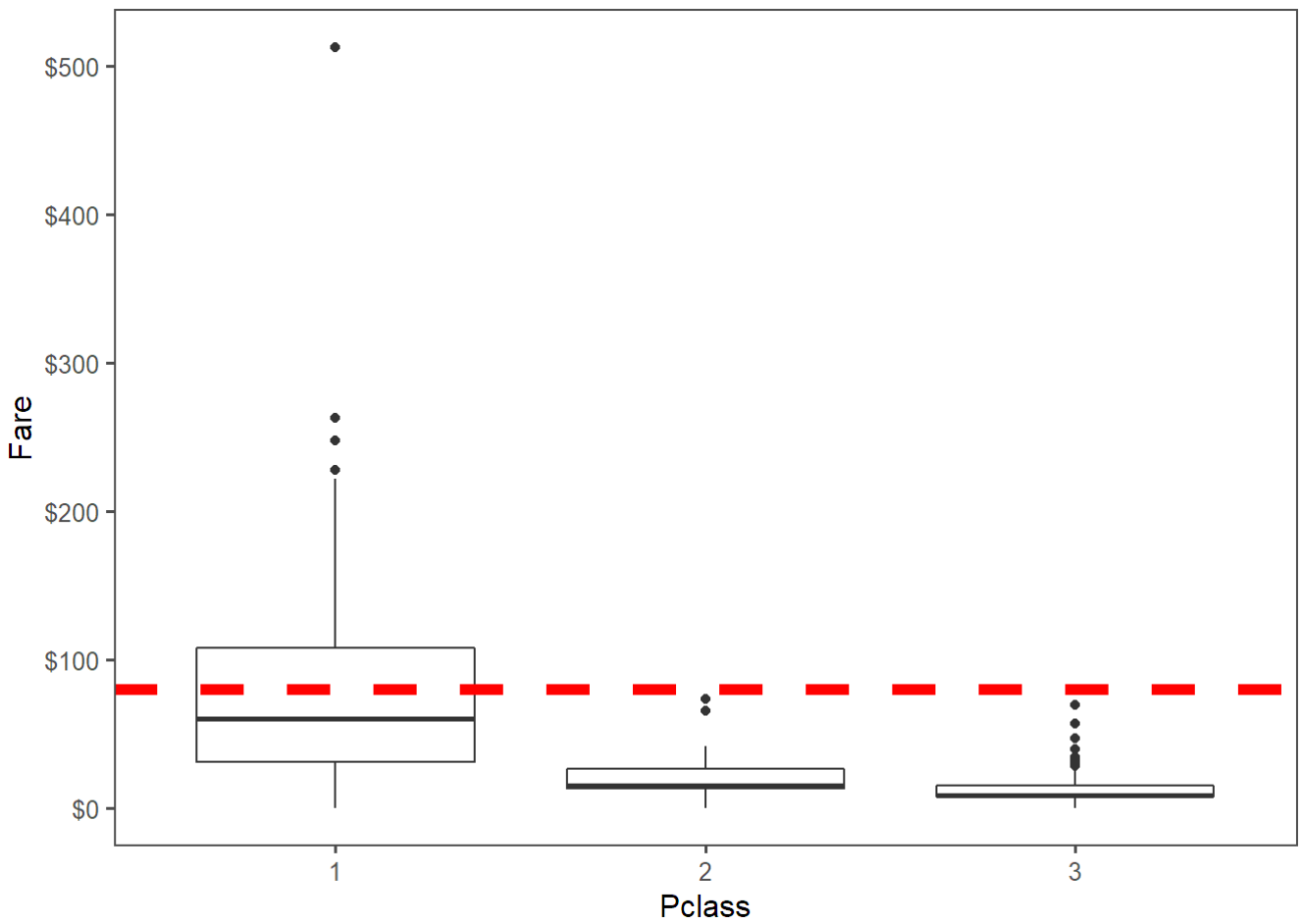

```
boxplot.stats(data$Fare)$out
```

```
##   [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  82.2667 262.3750  76.2917 263.0000
## [121] 262.3750 262.3750 263.0000 211.5000 211.5000 221.7792  78.8500 221.7792
## [129]  75.2417 151.5500 262.3750  83.1583 221.7792  83.1583  83.1583 247.5208
## [137]  69.5500 134.5000 227.5250  73.5000 164.8667 211.5000  71.2833  75.2500
## [145] 106.4250 134.5000 136.7792  75.2417 136.7792  82.2667  81.8583 151.5500
## [153]  93.5000 135.6333 146.5208 211.3375  79.2000  69.5500 512.3292  73.5000
## [161]  69.5500  69.5500 134.5000  81.8583 262.3750  93.5000  79.2000 164.8667
## [169] 211.5000  90.0000 108.9000
```

Evidentemente, esta gráfica no es representativa cuando el establecimiento de la tarifa depende de muchas otras variables como puede ser la clase o el puerto de origen. Además, el precio del ticket la tarifa expuesta engloba incluye al total de

Por lo tanto, repetimos la representación en función de la clase.

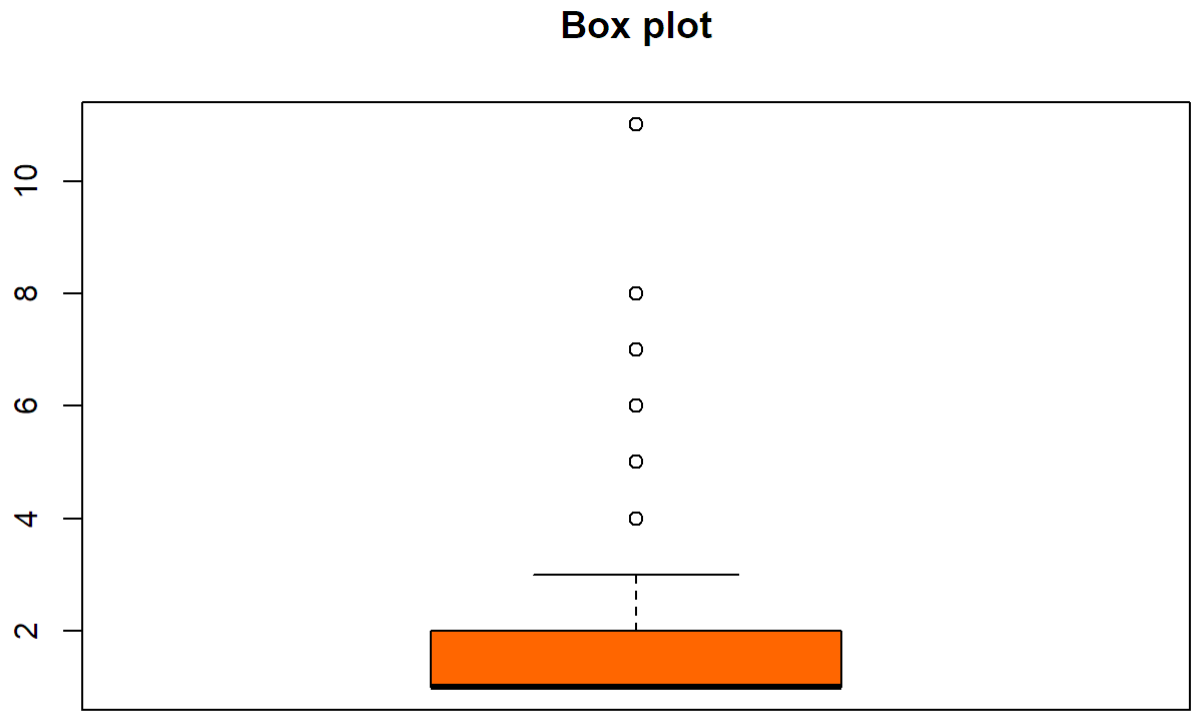
```
ggplot(data, aes(x = Pclass, y = Fare)) +
  geom_boxplot() +
  geom_hline(aes(yintercept=80),
             colour='red', linetype='dashed', lwd=2) +
  scale_y_continuous(labels=dollar_format()) +
  theme_few()
```



Se puede observar que los valores extremos pertenecen a una clase específica, mientras mayor es la clase y mayor es el número de pasajeros, más alta es la tarifa.

Variable *Famsize*

```
boxplot(data$Famsize, main="Box plot", col="#FF6600")
```




```
boxplot.stats(data$Famsize)$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  5  4  4  4  6  6  5  6  4
## [101]  5  8  8  5  4  7  5  7  4 11  7  4  4  4  4  4 11  4  4 11 11  7  4  5  5
```

Se observar que los valores extremos están un rango normal. Por ejemplo, ninguno es menor que cero o mayor que 12. Por tanto, son valores que perfectamente pueden darse.

Exportación de los datos

Se vuelven a revisar los datos para comprobar que no contienen valores nulos y/o vacios.

```
summary(data)
```

```
##  Survived Pclass      Name              Sex          Age       
##  0:815    1:323   Length:1309        female:466   Min.   : 0.17  
##  1:494    2:277   Class :character   male  :843   1st Qu.:21.00  
##           3:709   Mode  :character                Median :28.00  
##                                                   Mean   :29.38  
##                                                   3rd Qu.:36.50  
##                                                   Max.   :80.00  
##                                                                  
##      SibSp           Parch             Fare         Embarked        Title      
##  Min.   :0.0000   Min.   :0.000   Min.   :  0.000    :  2    Master    : 61  
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:  7.896   C:270    Miss      :264  
##  Median :0.0000   Median :0.000   Median : 14.454   Q:123    Mr        :757  
##  Mean   :0.4989   Mean   :0.385   Mean   : 33.276   S:914    Mrs       :198  
##  3rd Qu.:1.0000   3rd Qu.:0.000   3rd Qu.: 31.275            Rare Title: 29  
##  Max.   :8.0000   Max.   :9.000   Max.   :512.329                            
##                                                                              
##       Surname        Famsize             Fsize      PrecioTicket    RangoEdad  
##  Andersson:  11   Min.   : 1.000   large    : 82   Min.   :  0.00   0-19 :293  
##  Sage     :  11   1st Qu.: 1.000   singleton:790   1st Qu.:  7.55   20-39:758  
##  Asplund  :   8   Median : 1.000   small    :437   Median :  8.05   40-59:225  
##  Goodwin  :   8   Mean   : 1.884                   Mean   : 14.76   60-79: 33  
##  Davies   :   7   3rd Qu.: 2.000                   3rd Qu.: 15.01   >79  :  0  
##  Brown    :   6   Max.   :11.000                   Max.   :128.08              
##  (Other)  :1258                                    NA's   :1
```

Una vez revisado nuestro conjunto de datos hacemos una selección de las variables que queremos analizar con mayor profundidad.

El estado actual del dataset es:

```
## 'data.frame':    1309 obs. of  15 variables:
##  $ 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 ...
##  $ Name        : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)" "Heikkine
n, Miss. Laina" "Futrelle, Mrs. Jacques Heath (Lily May Peel)" ...
##  $ Sex         : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age         : num  22 38 26 35 35 22 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    : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
##  $ Title       : Factor w/ 5 levels "Master","Miss",..: 3 4 2 4 3 3 3 1 4 4 ...
##  $ Surname     : Factor w/ 875 levels "Abbing","Abbott",..: 101 183 335 273 16 544 506 614 388 565 ...
##  $ Famsize     : num  2 2 1 2 1 1 1 5 3 2 ...
##  $ Fsize       : Factor w/ 3 levels "large","singleton",..: 3 3 2 3 2 2 2 1 3 3 ...
##  $ PrecioTicket: num  7.25 35.64 7.92 26.55 8.05 ...
##  $ RangoEdad   : Factor w/ 5 levels "0-19","20-39",..: 2 2 2 2 2 2 3 1 2 1 ...
```

```
##   Survived Pclass                                                Name    Sex
## 1        0      3                             Braund, Mr. Owen Harris   male
## 2        1      1 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female
## 3        1      3                              Heikkinen, Miss. Laina female
## 4        1      1       Futrelle, Mrs. Jacques Heath (Lily May Peel) female
## 5        0      3                            Allen, Mr. William Henry   male
## 6        0      3                                    Moran, Mr. James   male
##   Age SibSp Parch    Fare Embarked Title   Surname Famsize     Fsize
## 1  22     1     0  7.2500        S    Mr    Braund       2     small
## 2  38     1     0 71.2833        C   Mrs   Cumings       2     small
## 3  26     0     0  7.9250        S  Miss Heikkinen       1 singleton
## 4  35     1     0 53.1000        S   Mrs  Futrelle       2     small
## 5  35     0     0  8.0500        S    Mr     Allen       1 singleton
## 6  22     0     0  8.4583        Q    Mr     Moran       1 singleton
##   PrecioTicket RangoEdad
## 1      7.25000     20-39
## 2     35.64165     20-39
## 3      7.92500     20-39
## 4     26.55000     20-39
## 5      8.05000     20-39
## 6      8.45830     20-39
```

De todo el conjunto de datos vamos a seleccionar las variables:

- Survived
- Pclass
- Sex
- Age
- Fare
- Embarked
- Fsize

```
# Seleccion de caracteristicas de interes

#clean_data <- select(data, -Name, -Famsize)
clean_data <- data %>%
                    select(c('Survived','Pclass', 'Sex', 'Age', 'Embarked', 'Famsize', 'Fsize', 'Fare') )
# Visualizamos los datos limpios:
summary(clean_data)
```

```
##  Survived Pclass      Sex           Age        Embarked    Famsize
##  0:815    1:323   female:466   Min.   : 0.17    :  2    Min.   : 1.000
##  1:494    2:277   male  :843   1st Qu.:21.00   C:270    1st Qu.: 1.000
##           3:709                Median :28.00   Q:123    Median : 1.000
##                                Mean   :29.38   S:914    Mean   : 1.884
##                                3rd Qu.:36.50            3rd Qu.: 2.000
##                                Max.   :80.00            Max.   :11.000
##        Fsize          Fare
##  large    : 82   Min.   :  0.000
##  singleton:790   1st Qu.:  7.896
##  small    :437   Median : 14.454
##                  Mean   : 33.276
##                  3rd Qu.: 31.275
##                  Max.   :512.329
```

Se divide el dataset en dos conjuntos: train y test.

```
#Dividimos el conjunto de datos en datos de entrenamiento y datos prueba.
clean_train_data <- clean_data[1:nrow(dataTrain),]
clean_test_data <- clean_data[(nrow(dataTrain) + 1):nrow(data),]


# Exportación de los datos limpios en .csv

write.csv(clean_train_data, 'clean_train.csv')
write.csv(clean_test_data, 'clean_test.csv')

write.csv(clean_data, 'clean_full.csv')
```

Análisis de los datos

Selección de los datos a comparar

De las características del conjunto de entrenamiento nos interesa analizar las variables cuantitativas Age y Fare;y las variables cuantitativas Sex, Pclass y FSize.

Para analizar estas variables emplearemos diagramas de histogramas para las variables cuantitativas y diagramas de barras para las variables cualitativas en función de la supervivencia.

Relaciones de las variables independientes respecto la variable *Survived*

Comparativa entre las variables Sex y Survived

```
ggplot(clean_train_data,aes(Sex,fill=Survived))+geom_bar() +labs(x="Género", y="Pasajeros")+ guides(fill=guide_lege
nd(title=""))+ scale_fill_manual(values=c("#003333","#009999"))+ggtitle("Superviviente por género")
```

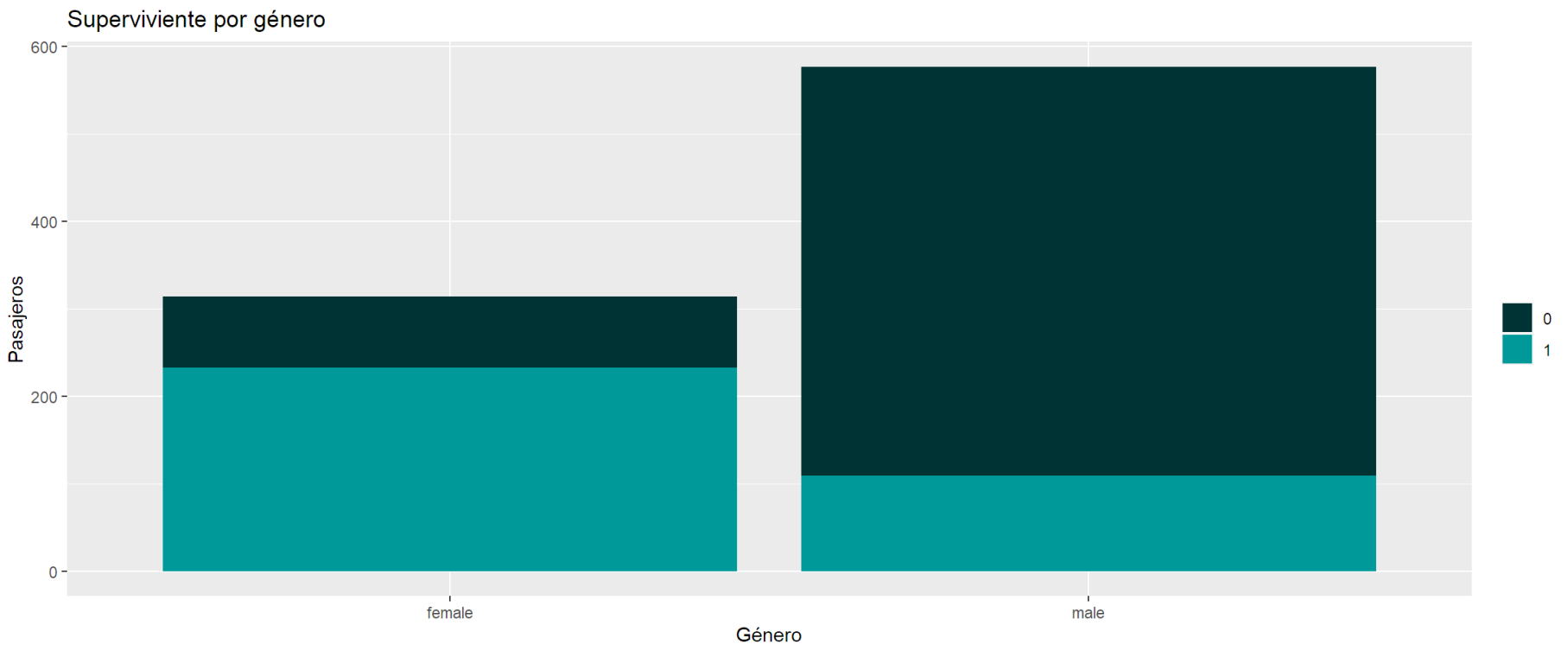


Tabla de contigencia

```
tablaSex <- table(clean_train_data$Sex, clean_train_data$Survived)
tablaSex
```

```
##
##             0   1
##   female   81 233
##   male    468 109
```

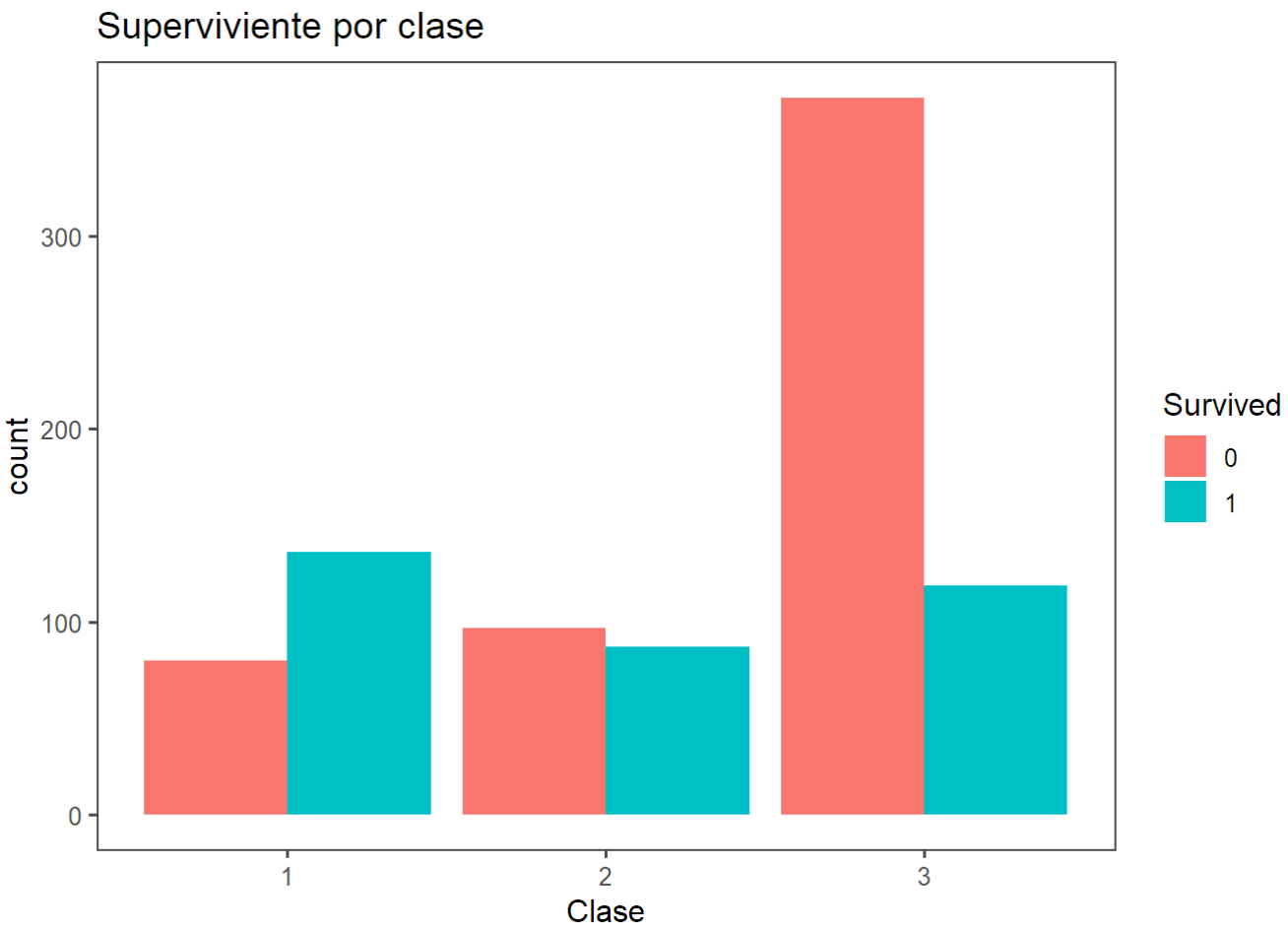
```
prop.table(tablaSex, margin = 1)
```

```
##
##                   0         1
##   female 0.2579618 0.7420382
##   male   0.8110919 0.1889081
```

El diagrama de barras anterior muestra la distribución de supervivencia de mujeres y hombres. Como se intuía está característica parece influir en la supervivencia. El gráfico de barras muestra que un 74% de los pasajeros mujeres sobrevivieron, mientras que solo un 19% de los pasajeros varones sobrevivieron. De tal forma que aquellos pasajeros con sexo femenino tuvieron una tasa de supervivencia más alta que los varones.

Comparativa entre las variables Pclass y Survived

```
ggplot(clean_train_data, aes(x = Pclass, fill = Survived)) +
  geom_bar(stat='count', position='dodge') +
  labs(x = 'Clase') +
  ggtitle("Superviviente por clase") +
  theme_few()
```




```
tablaClass <- table(clean_train_data$Pclass, clean_train_data$Survived)
tablaClass
```

```
##
##       0   1
##   1  80 136
##   2  97  87
##   3 372 119
```

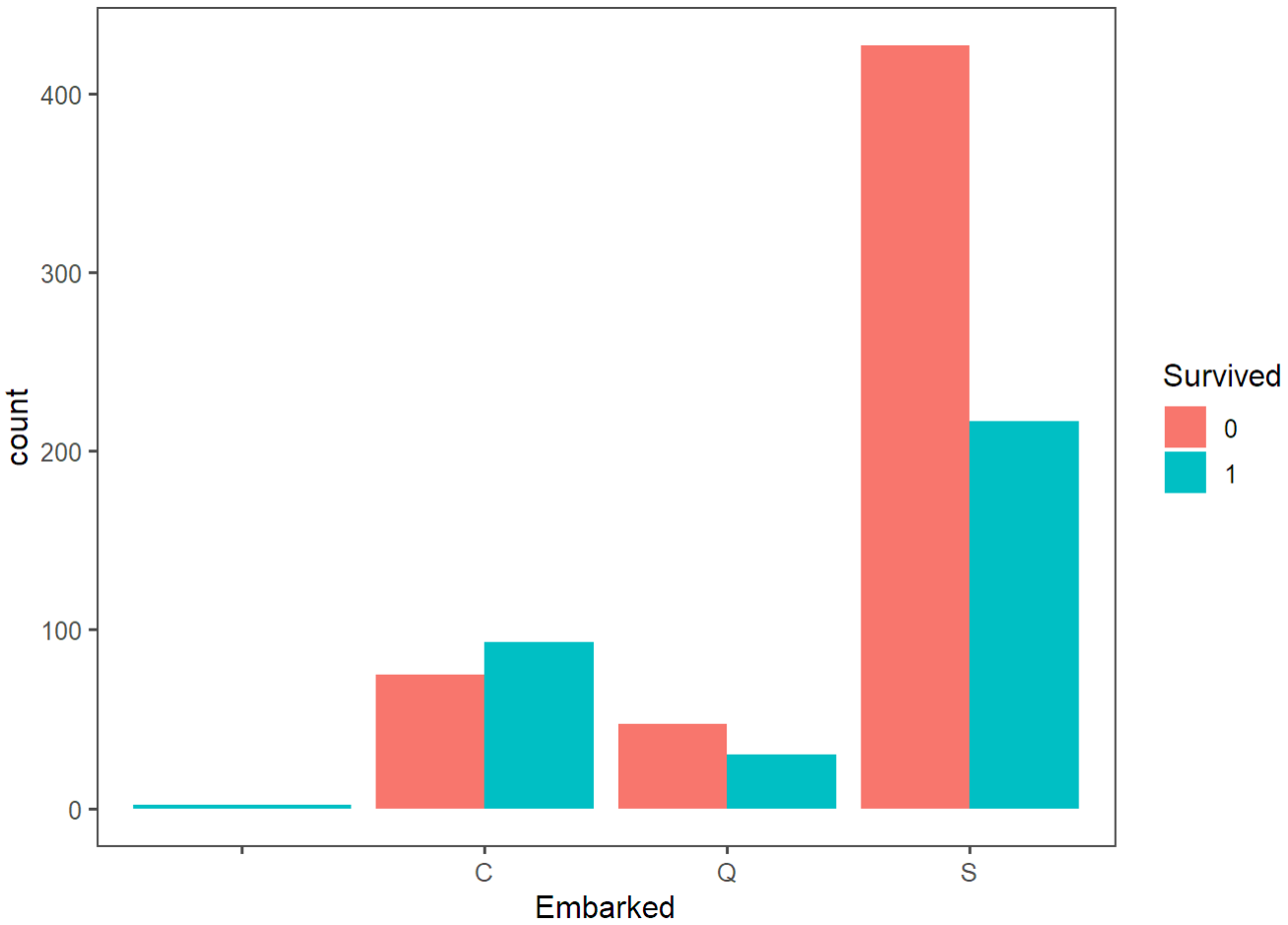
```
prop.table(tablaClass, margin = 1)
```

```
##
##             0         1
##   1 0.3703704 0.6296296
##   2 0.5271739 0.4728261
##   3 0.7576375 0.2423625
```

Las gráficas anteriores muestran la distribución de la supervivencia en función de la clase del pasajero. En el gráfico se observa que esta característica parece influir en la supervivencia. El gráfico de barras muestra que sobre el 63 % de los pasajeros de primera clase sobrevivieron, mientras que sobre el 48 % de los pasajeros de segunda clase sobrevivieron, y solo el 24 % de los pasajeros de tercera clase sobrevivieron. De tal forma que aquellos pasajeros en las clases más altas tienen una tasa de supervivencia más alta que aquellos pasajeros en las clases más bajas.

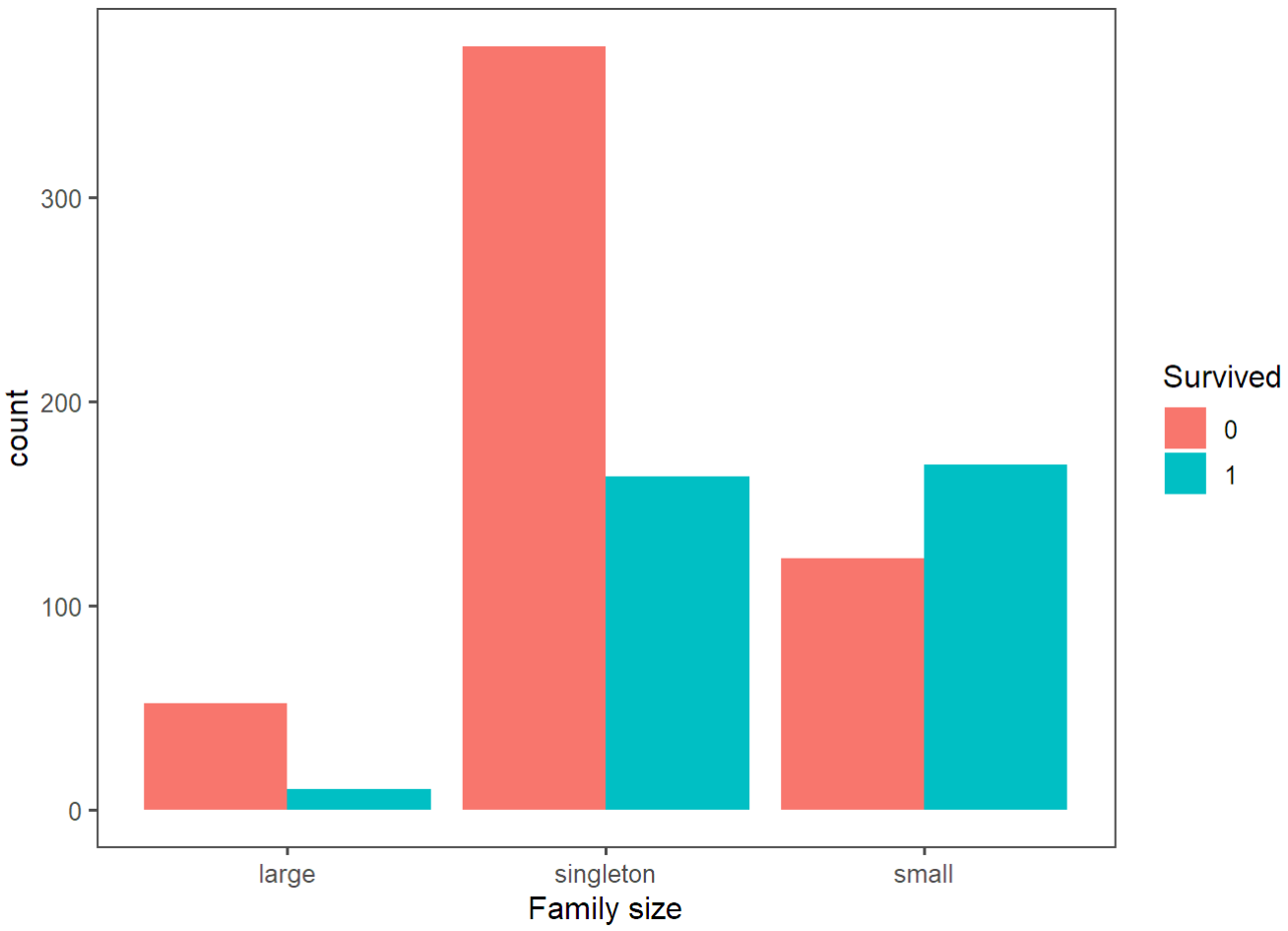
Comparativa entre las variables Embarked y Survived

```
ggplot(clean_train_data, aes(x = Embarked, fill = Survived)) +
  geom_bar(stat='count', position='dodge') +
  labs(x = 'Embarked') +
  theme_few()
```



Comparativa entre las variables Fsize y Survived

```
ggplot(clean_train_data, aes(x = Fsize, fill = Survived)) +
  geom_bar(stat='count', position='dodge') +
  labs(x = 'Family size') +
  theme_few()
```



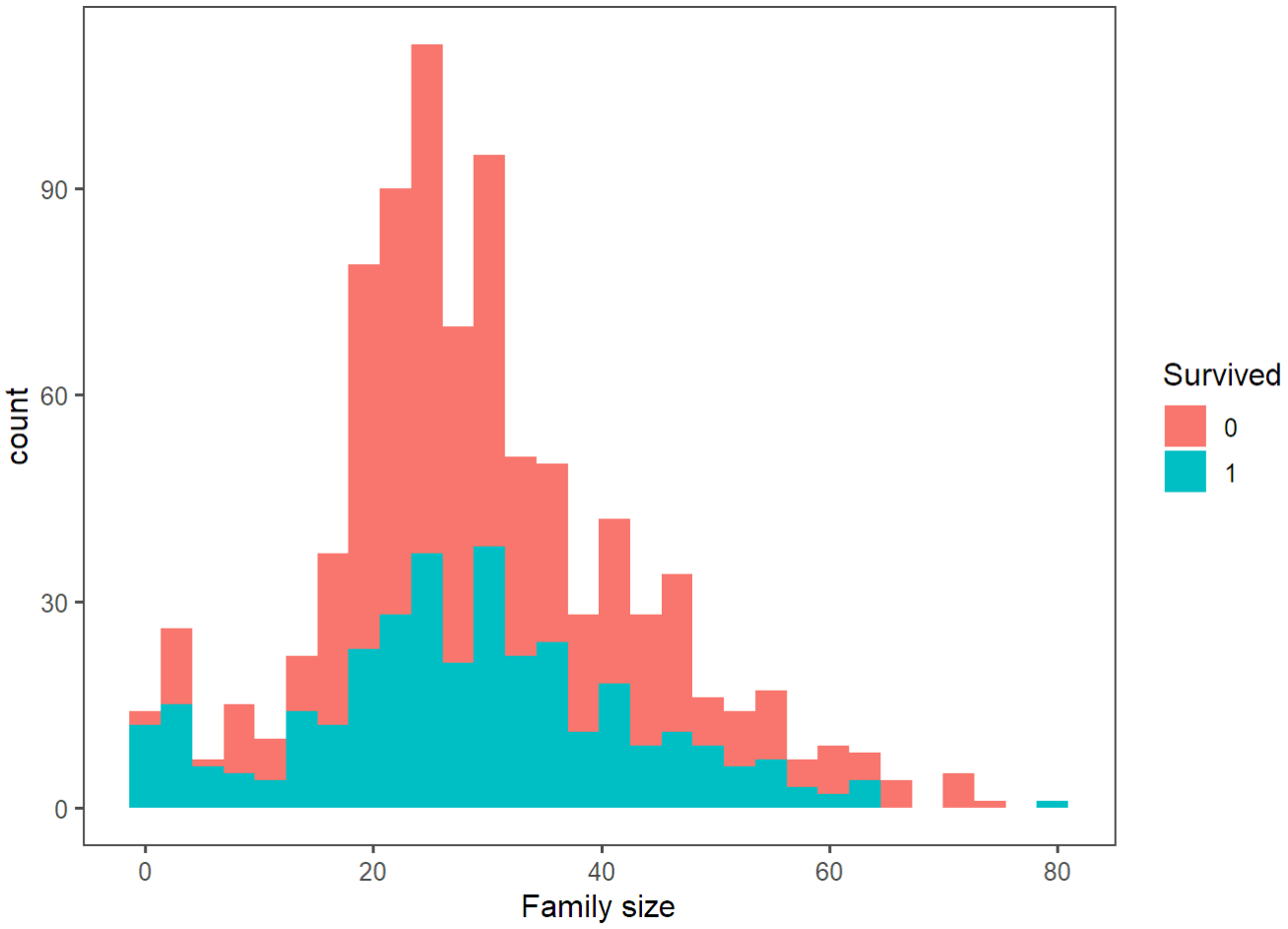
La gráfica anterior muestra que sobre el 70% los pasajeros solteros y sobre el 82% de las familias grandes no sobrevivieron. Respecto al conjunto de solteros suponemos que la mayoría deberían ser varones dado que en la época del accidente sería más habitual que estos viajen solos. Además, suponemos que las familias grandes no cabrían todos en un bote de salvavidas y esto podría influir en su supervivencia.

Más adelante analizaremos esta característica en función del sexo ya que ser soltero y varón debería ser un rasgo que influya en la supervivencia.

Comparativa entre las variables Age y Survived

```
ggplot(clean_train_data, aes(x = Age, fill = Survived)) +
  geom_histogram() +
  labs(x = 'Family size') +
  theme_few()
```

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

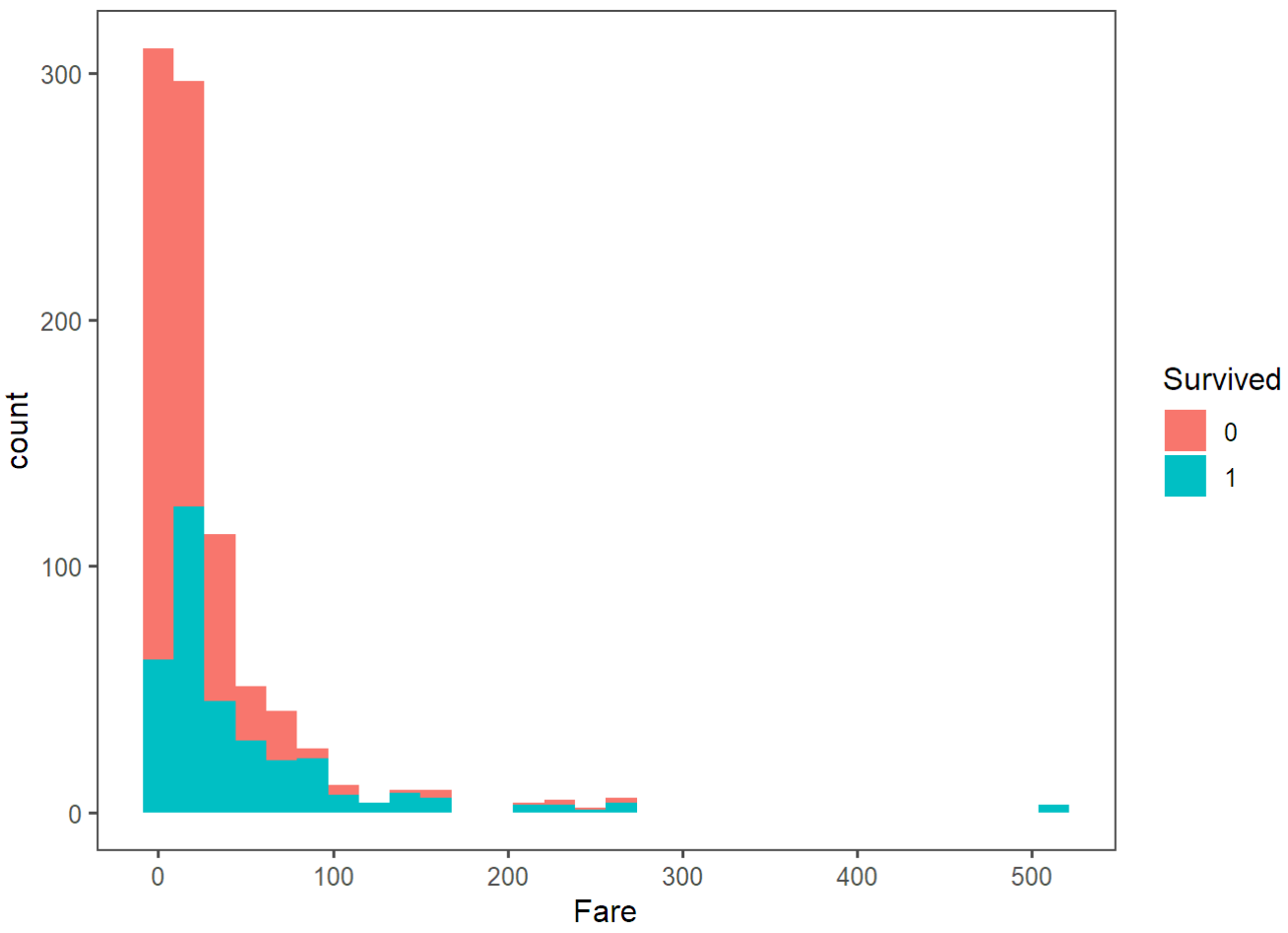


No hay nada fuera de lo común en esta trama, excepto la parte izquierda de la distribución. Demuestra que los niños y los bebés eran la prioridad, por lo tanto, se salvó una buena parte de los niños.

Comparativa entre las variables Fare y Survived

```
ggplot(clean_train_data, aes(x = Fare, fill = Survived)) +
  geom_histogram() +
  labs(x = 'Fare') +
  theme_few()
```

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

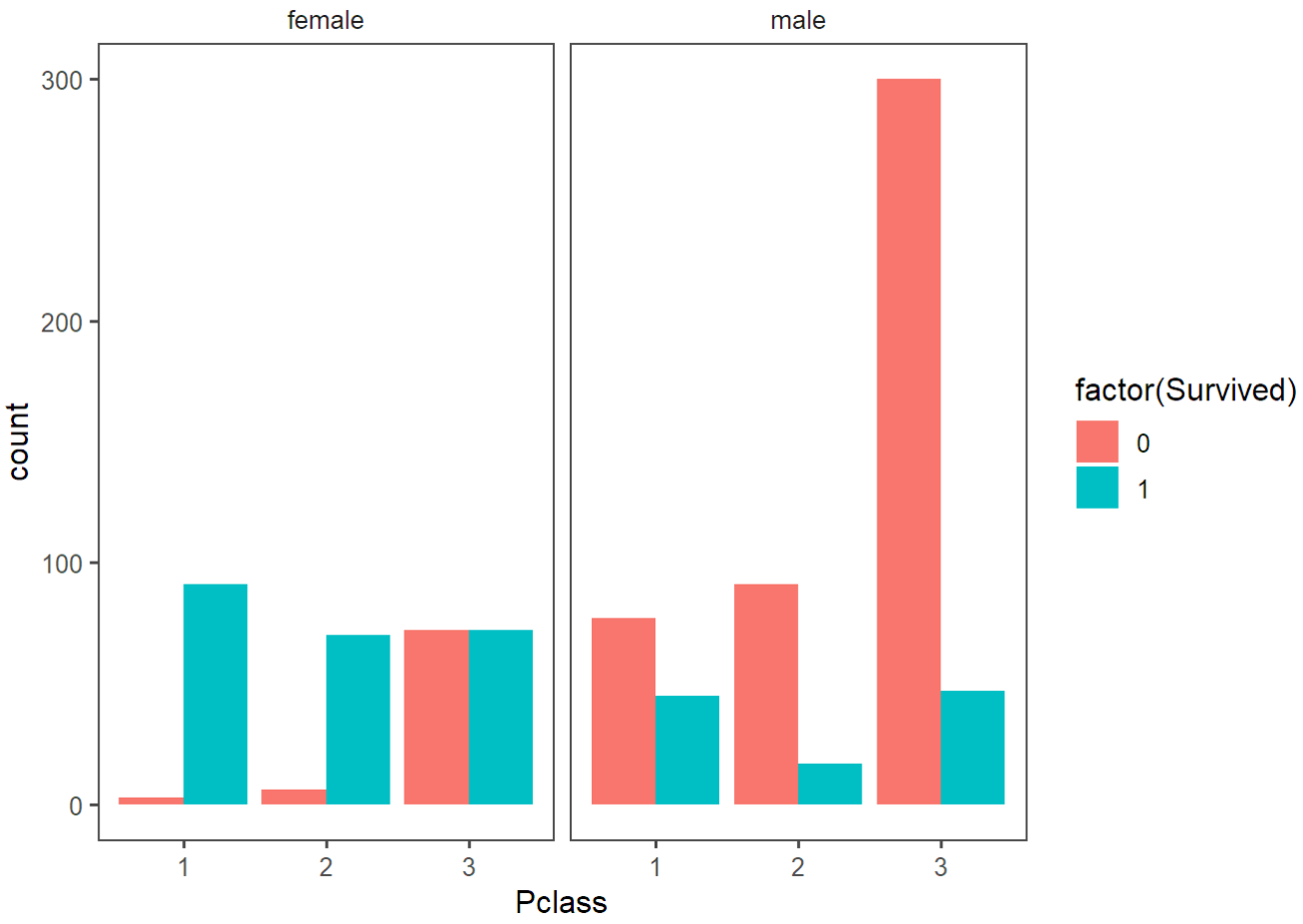


La gráfica anterior muestra algo interesante, existe un pico en los valores de menos de 100 dólares que representa que muchos de los pasajeros que compraron un ticket dentro de ese rango no sobrevivieron. Cuando la tarifa es aproximadamente más de 280 dólares, la tasa de mortalidad es baja, lo que significa que todos los que pasaron de esa la tarifa sobrevivieron.

Relaciones de dos variables independientes con la variable dependiente

Comparativa de las variables Sex y Pclass con la variable Survived

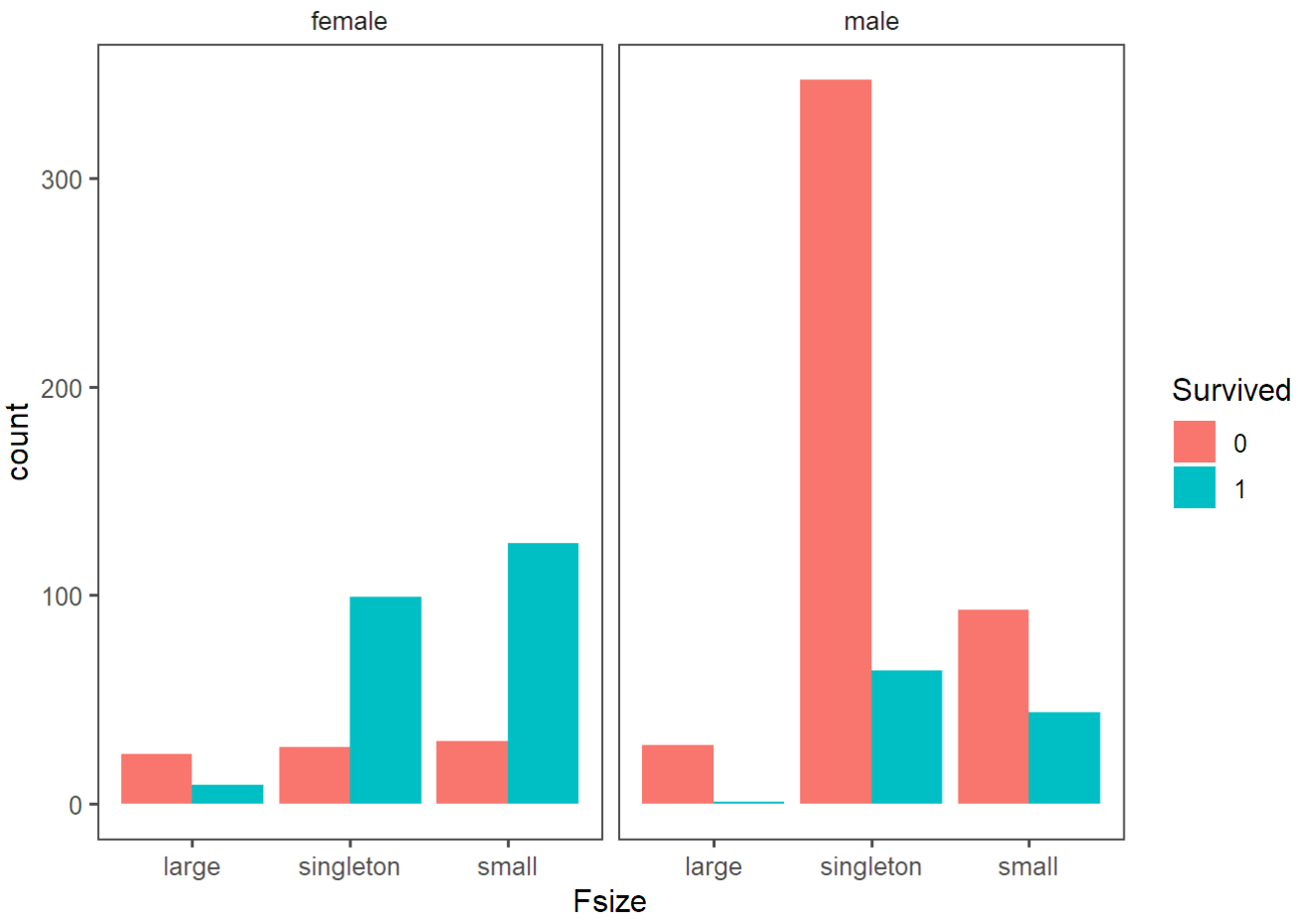
```
ggplot(clean_train_data, aes(Pclass, fill = factor(Survived))) +
geom_bar(position=position_dodge()) +
facet_grid(.~Sex) +
theme_few()
```



La gráfica anterior muestra que los pasajeros mujeres y de una clase alta sobrevivieron en su mayoría. También se observa que los pasajeros varones tuvieron una tasa de supervivencia mucho más baja que las mujeres. Esta tasa de supervivencia va empeorando a medida que la clase del pasajero baje. Se puede concluir que ser de sexo y la clase pueden influir en la supervivencia del pasajero.

Comparativa de las variables Sex y Fsize con la variable Survived

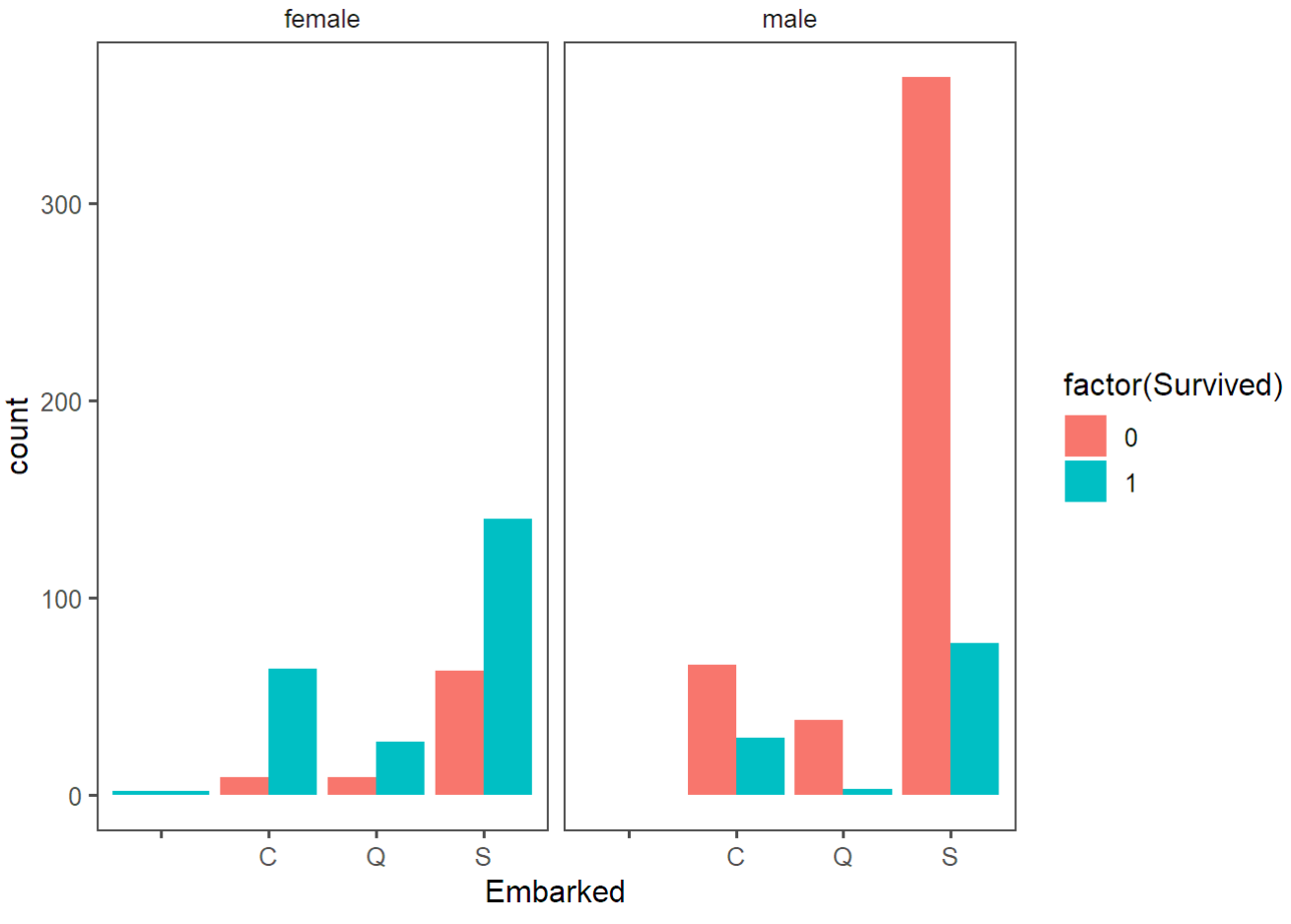
```
ggplot(clean_train_data, aes(Fsize, fill = factor(Survived))) +
geom_bar(position=position_dodge()) +
labs(fill = "Survived") +
facet_grid(.~Sex) +
theme_few()
```



La gráfica anterior muestra que los pasajeros solteros varones tuvieron una tasa de mortalidad más alta. Esta valor es lógico debido que en los botes salvavidas tendrían una preferencia menor a las mujeres y niños.

Comparativa de las variables Sex y Embarked con la variable Survived

```
ggplot(clean_train_data, aes(Embarked, fill = factor(Survived))) +
geom_bar(position=position_dodge()) +
facet_grid(.~Sex) +
theme_few()
```

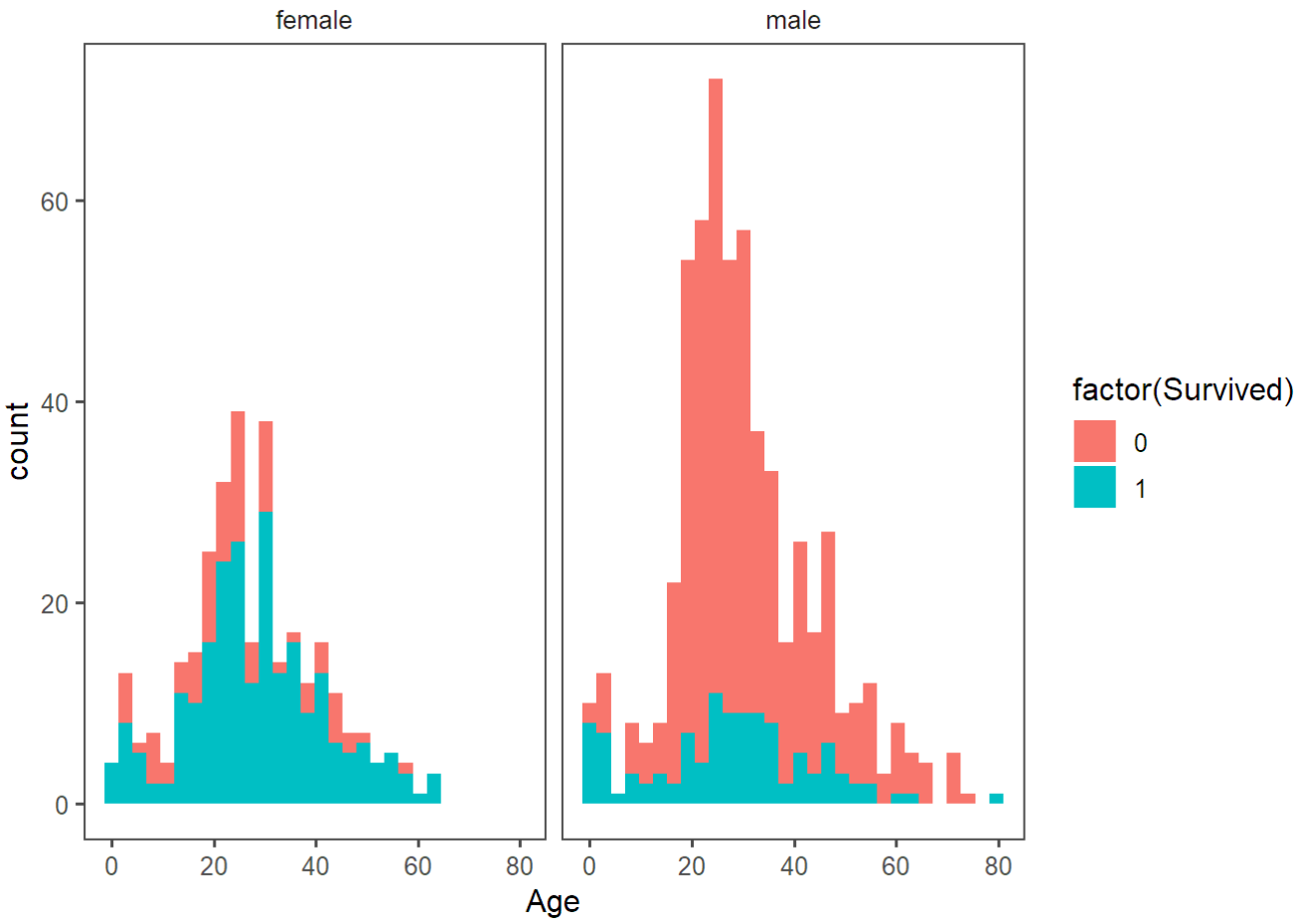


La gráfica anterior muestra que la mayoría de los pasajeros parece ser que embarcaron en Southampton (S). Aunque la mayoría de los pasajeros embarco en Southampton (S) a priori no debería ser relevante para la supervivencia, a menos que tenga alguna relación con la localización del camarote o la clase del pasajero.

Comparativa de las variables Sex y Age con la variable Survived

```
ggplot(clean_train_data, aes(x = Age, fill = factor(Survived))) +
geom_histogram() +
facet_grid(.~Sex) +
theme_few()
```

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

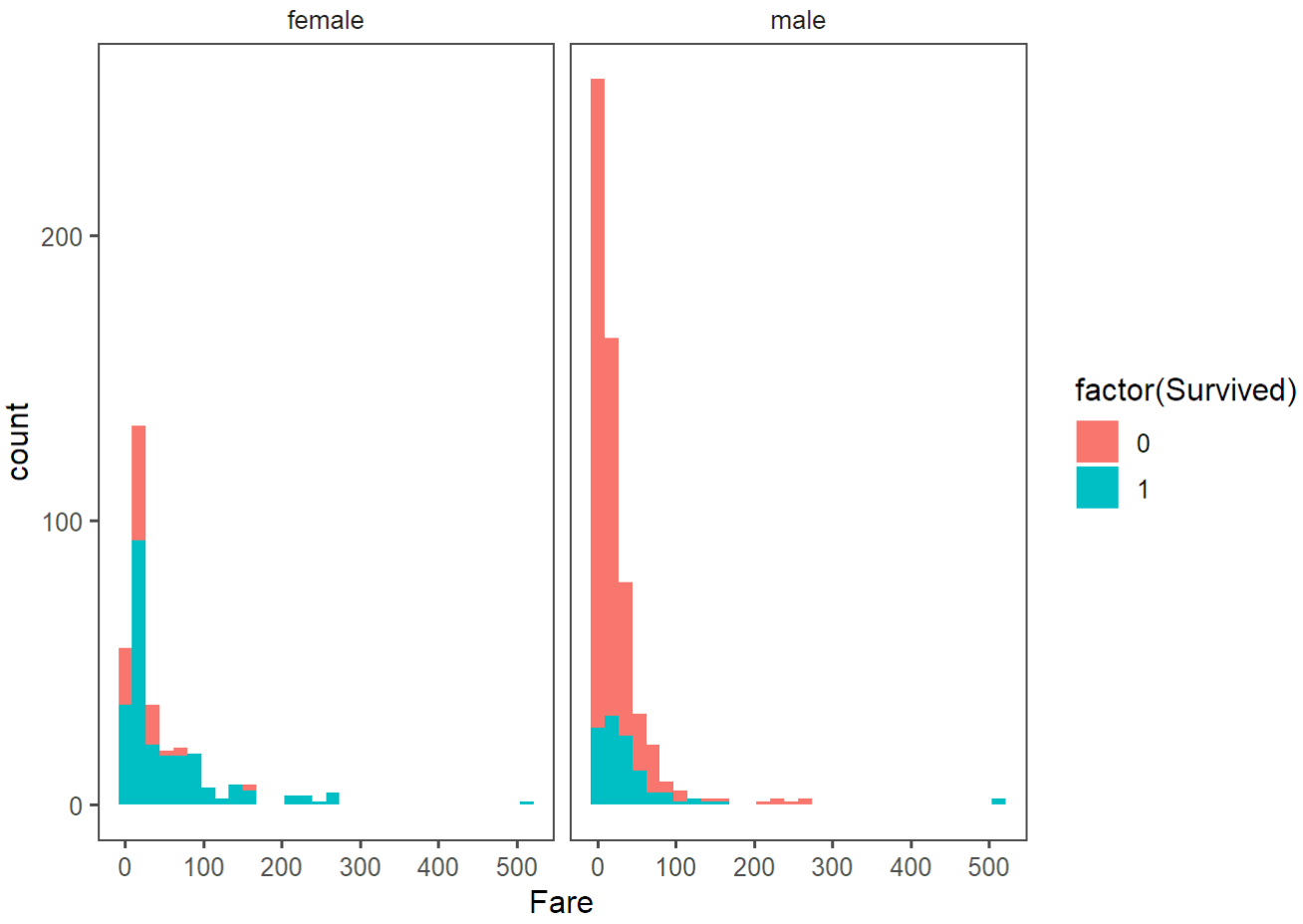


La gráfica anterior muestra que la supervivencia de los varones es baja para los adultos. Los niños varones tienen una tasa de supervivencia alta, esto es lógico debido a la preferencia que tuvieron estos en los botes. Por tanto, podemos concluir que el sexo y la edad de los pasajeros son características que influyen en la supervivencia.

Comparativa de las variables Sex y Fare con la variable Survived

```
ggplot(clean_train_data, aes(x = Fare, fill = factor(Survived))) +
geom_histogram() +
facet_grid(.~Sex) +
theme_few()
```

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



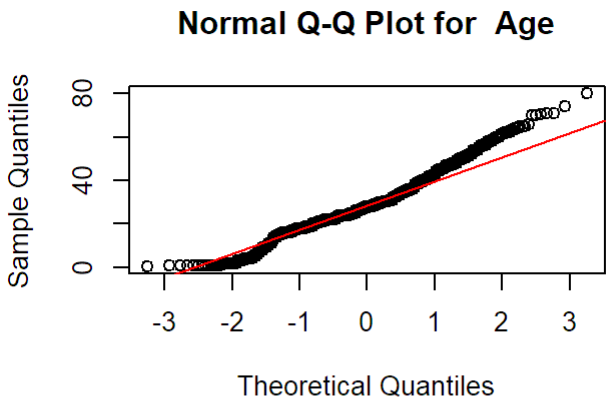
La gráfica anterior no se observa algo nuevo. Solamente que la condición socioeconómica parece un factor que puede influir en la supervivencia.
De las gráficas anteriores se concluye que las características Age,Sex, Fare y Pclass parecen tener influyen en la supervivencia.

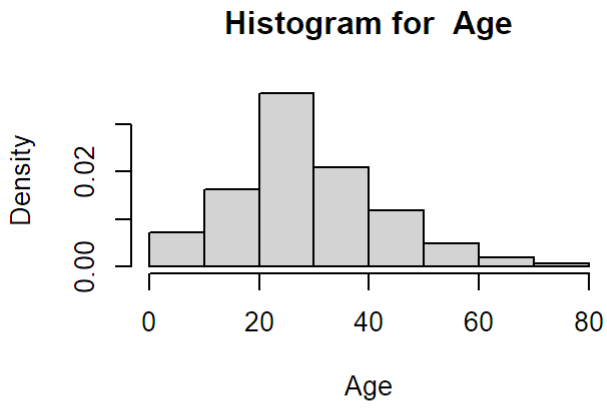
Comprobación de la normalidad y homogeneidad de la varianza

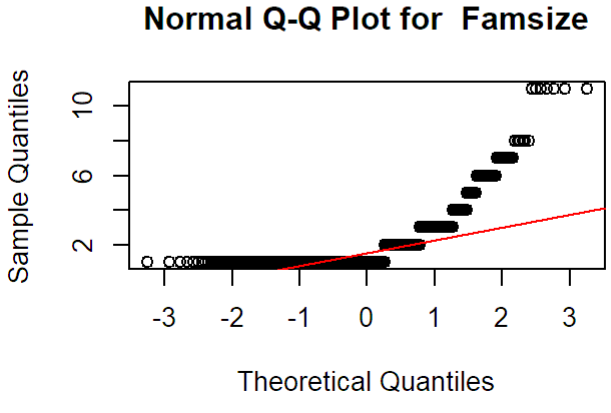
Normalidad

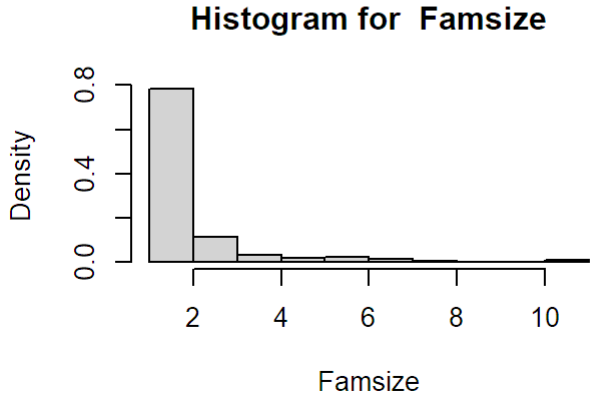
En este apartado se revisará si las variables siguen una distribución normal.

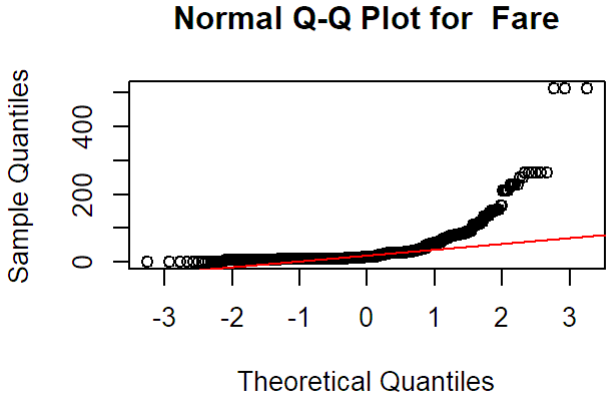
```
alpha = 0.05
drawQQPlotAndtHist <- function(dataset) {
  par(mfrow=c(2,2))
  for(i in 1:ncol(dataset)) {
    if (is.numeric(dataset[,i])){
      qqnorm(dataset[,i],main = paste("Normal Q-Q Plot for ",colnames(dataset)[i]))
      qqline(dataset[,i],col="red")
      hist(dataset[,i],
           main=paste("Histogram for ", colnames(dataset)[i]),
          xlab=colnames(dataset)[i], freq = FALSE)
    }
  }
}
#Mostramos las gráficas.

drawQQPlotAndtHist(clean_train_data)
```

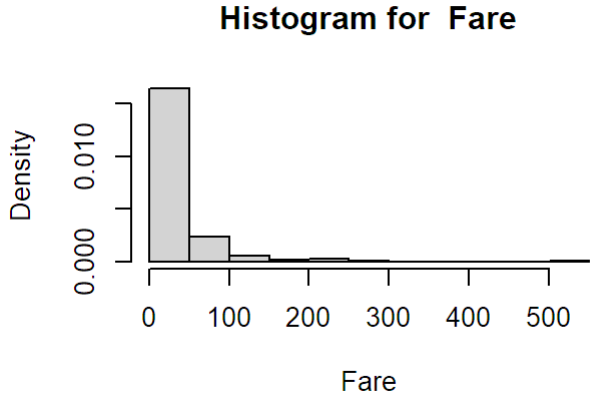












De las gráficas anteriores, se observa que la característica Age pueden ser candidata a la normalización. No obstante, se aplicará el test de Shapiro-Wilk para contrastar esta asunción.

Test Shapiro-Wilk

El test de Shapiro-Wilk se usa para contrastar si un conjunto de datos siguen una distribución normal o no. En nuestro caso se aplicará este test cada una las variables cuantitativas consideradas. De tal forma que la hipótesis nula (H0) y la alternativa (H1) se pueden escribir de la siguiente forma:

- Hipótesis nula (Ho): Los datos de la muestra no son significativamente diferentes de una población normal.
- Hipótesis alternativa (H1): Los datos de la muestra son significativamente diferentes de una población normal.
- Zona de rechazo. Para todo valor de probabilidad mayor que un nivel de significación p = 0.05, se acepta Ho y se rechaza H1.

Para comprobar la asunción de normalidad aplicamos el test Shapiro-Wilk, para ello utilizamos la función **shapiro.test**. A continuación, se muestra la aplicación del test Shapiro-Wilk para las variables cuantitativas consideradas:

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

```
##
##  Shapiro-Wilk normality test
##
## data:  clean_train_data$Age
## W = 0.97494, p-value = 2.993e-11
```

```
shapiro.test(clean_train_data$Famsize)
```

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

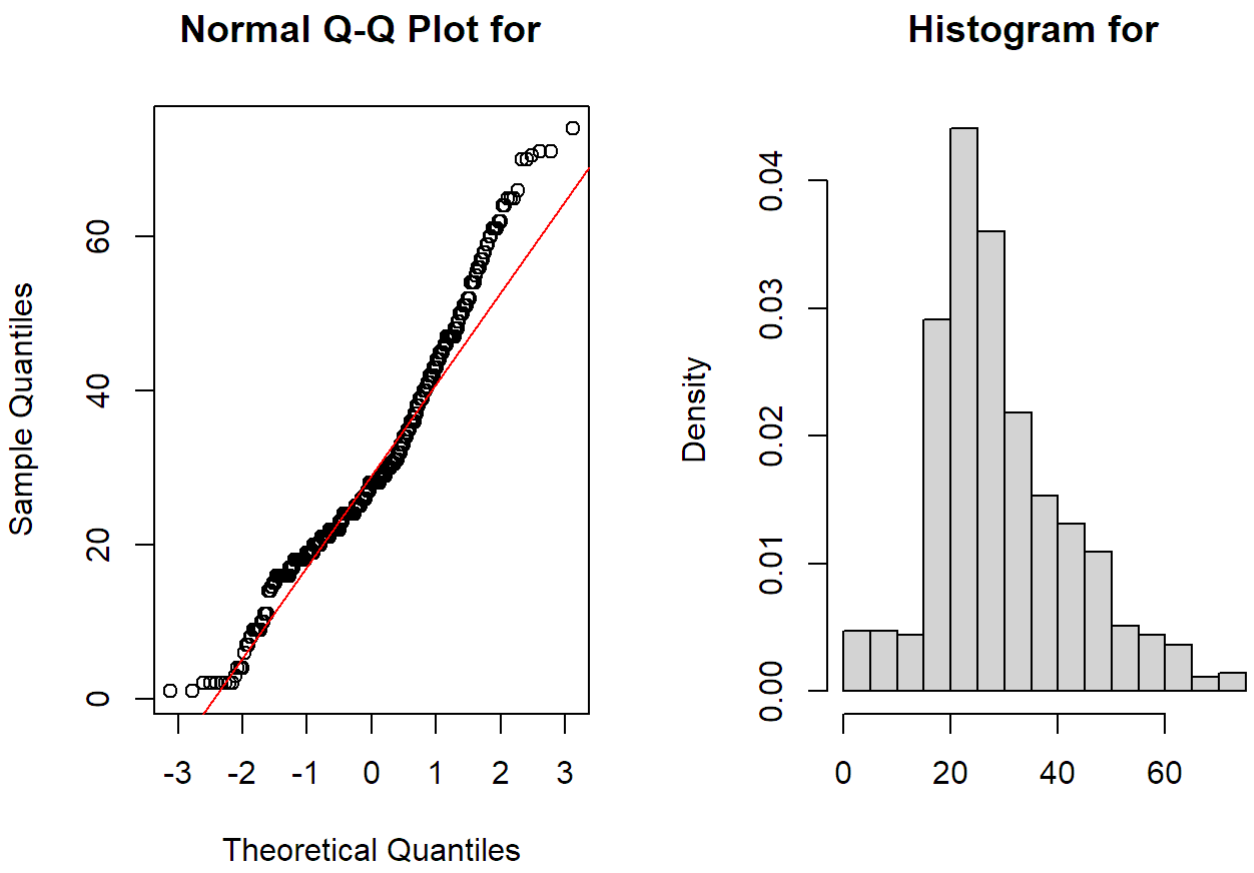
```
shapiro.test(clean_train_data$Fare)
```

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

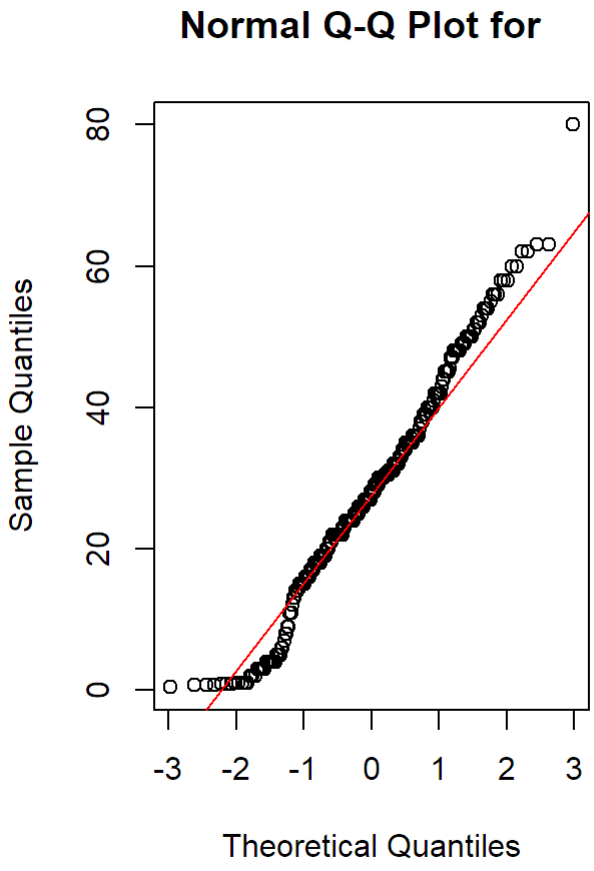
Dado los resultados anteriores, se observa que para las cuatro características consideradas sus correspondientes p-valores son inferiores al nivel de significación (p = 0.05). Por tanto, rechazamos la H0 y concluimos con un 95% de confianza que los datos no se distribuyen normalmente.

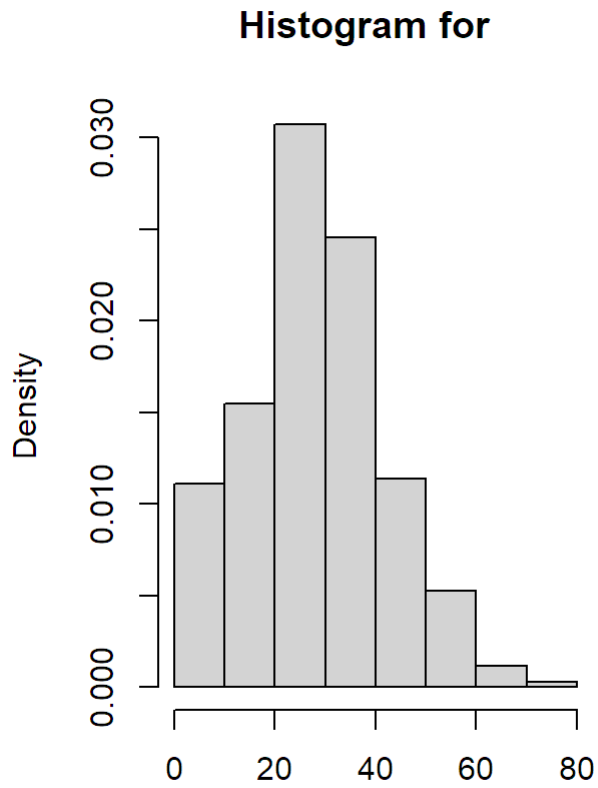
A continuación se aplicará este test para realizar el contraste de si existen diferencias en la edad (Age) en función de la supervivencia (Survived).

```
age_sur_0 <- clean_train_data$Age[clean_train_data$Survived==0]
age_sur_1 <- clean_train_data$Age[clean_train_data$Survived==1]

par(mfrow=c(1,2))
qqnorm(age_sur_0, main = paste("Normal Q-Q Plot for ", colnames(age_sur_0)[1]))
qqline(age_sur_0, col="red")
hist(age_sur_0,
main=paste("Histogram for ", colnames(age_sur_0)[1]),
xlab=colnames(age_sur_0)[1], freq = FALSE)
```




```
par(mfrow=c(1,2))
qqnorm(age_sur_1, main = paste("Normal Q-Q Plot for ", colnames(age_sur_1)[1]))
qqline(age_sur_1, col="red")
hist(age_sur_1,
     main=paste("Histogram for ", colnames(age_sur_1)[1]),
xlab=colnames(age_sur_1)[1], freq = FALSE)
```






```
shapiro.test(age_sur_0)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  age_sur_0
## W = 0.9553, p-value = 7.585e-12
```

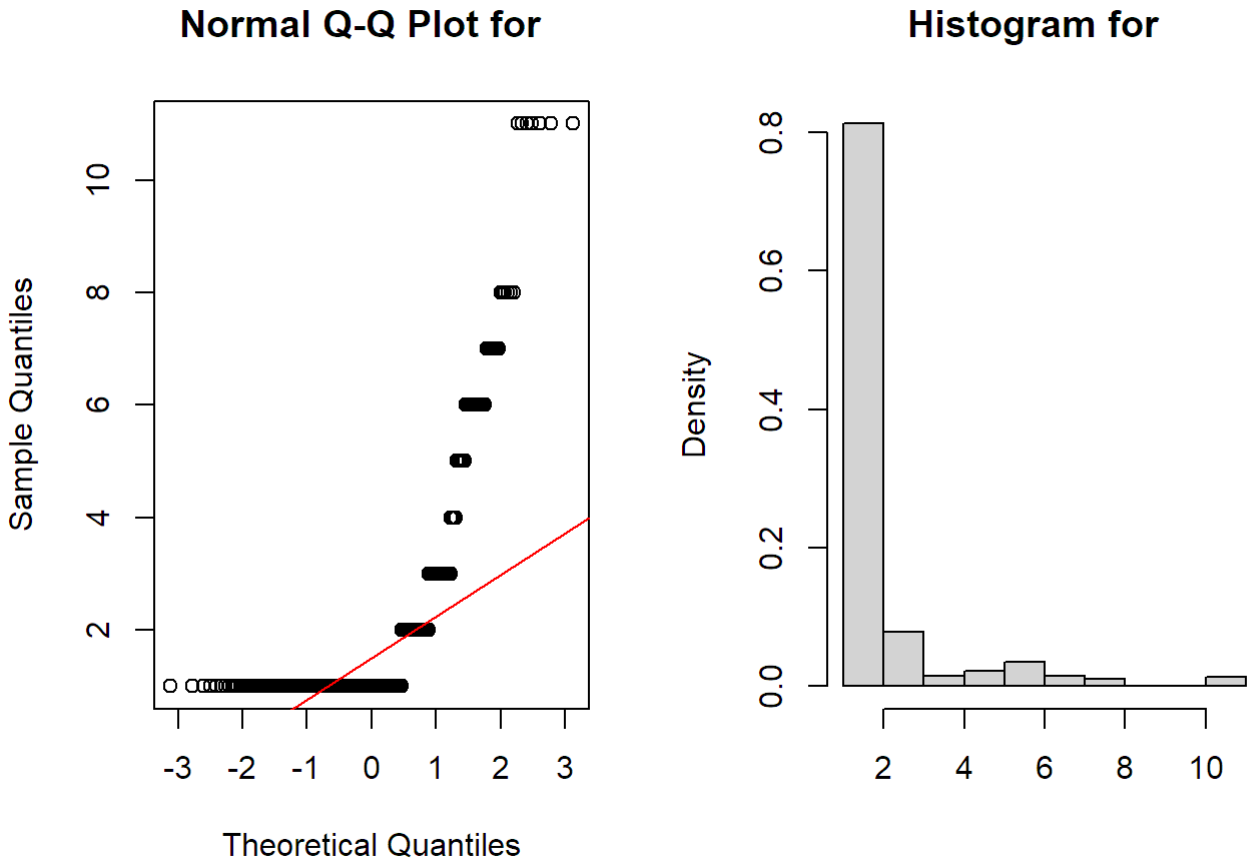
```
shapiro.test(age_sur_1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  age_sur_1
## W = 0.98404, p-value = 0.0007741
```

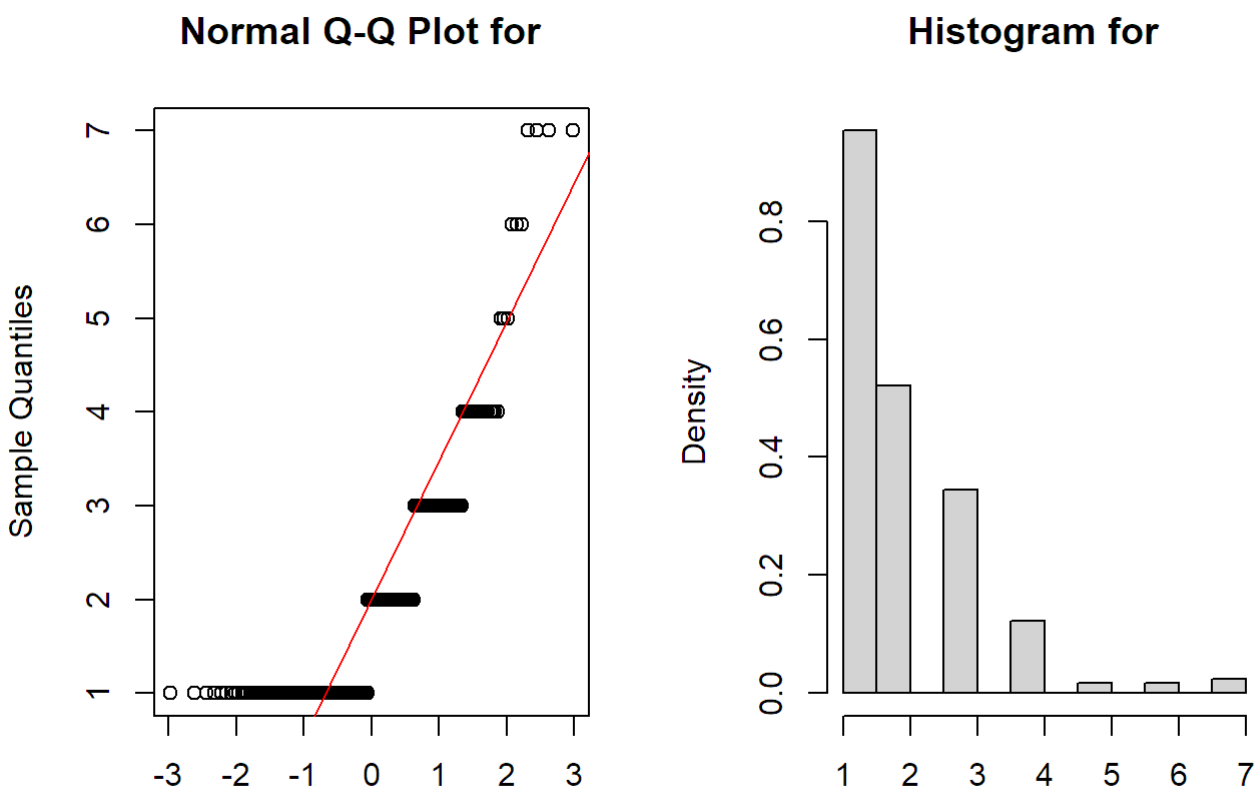
Dado los resultados anteriores, se observa que para las cuatro características consideradas sus correspondientes p-valores son inferiores al nivel de significación (p = 0.05). Por tanto, rechazamos la H0 y concluimos con un 95% de confianza que los datos no se distribuyen normalmente.

A continuación se aplicará este test para realizar el contraste de si existen diferencias en la característica familiares a bordo (SibSp) en función de la supervivencia (Survived).

```
Famsize_sur_0 <- clean_train_data$Famsize[clean_train_data$Survived==0]
Famsize_sur_1 <- clean_train_data$Famsize[clean_train_data$Survived==1]

par(mfrow=c(1,2))
qqnorm(Famsize_sur_0, main = paste("Normal Q-Q Plot for ", colnames(Famsize_sur_0)[1]))
qqline(Famsize_sur_0, col="red")
hist(Famsize_sur_0,
     main=paste("Histogram for ", colnames(Famsize_sur_0)[1]),
xlab=colnames(Famsize_sur_0)[1], freq = FALSE)
```




```
par(mfrow=c(1,2))
qqnorm(Famsize_sur_1, main = paste("Normal Q-Q Plot for ", colnames(Famsize_sur_1)[1]))
qqline(Famsize_sur_1, col="red")
hist(Famsize_sur_1,
main=paste("Histogram for ", colnames(Famsize_sur_1)[1]),
xlab=colnames(Famsize_sur_1)[1], freq = FALSE)
```




```
shapiro.test(Famsize_sur_0)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Famsize_sur_0
## W = 0.55147, p-value < 2.2e-16
```

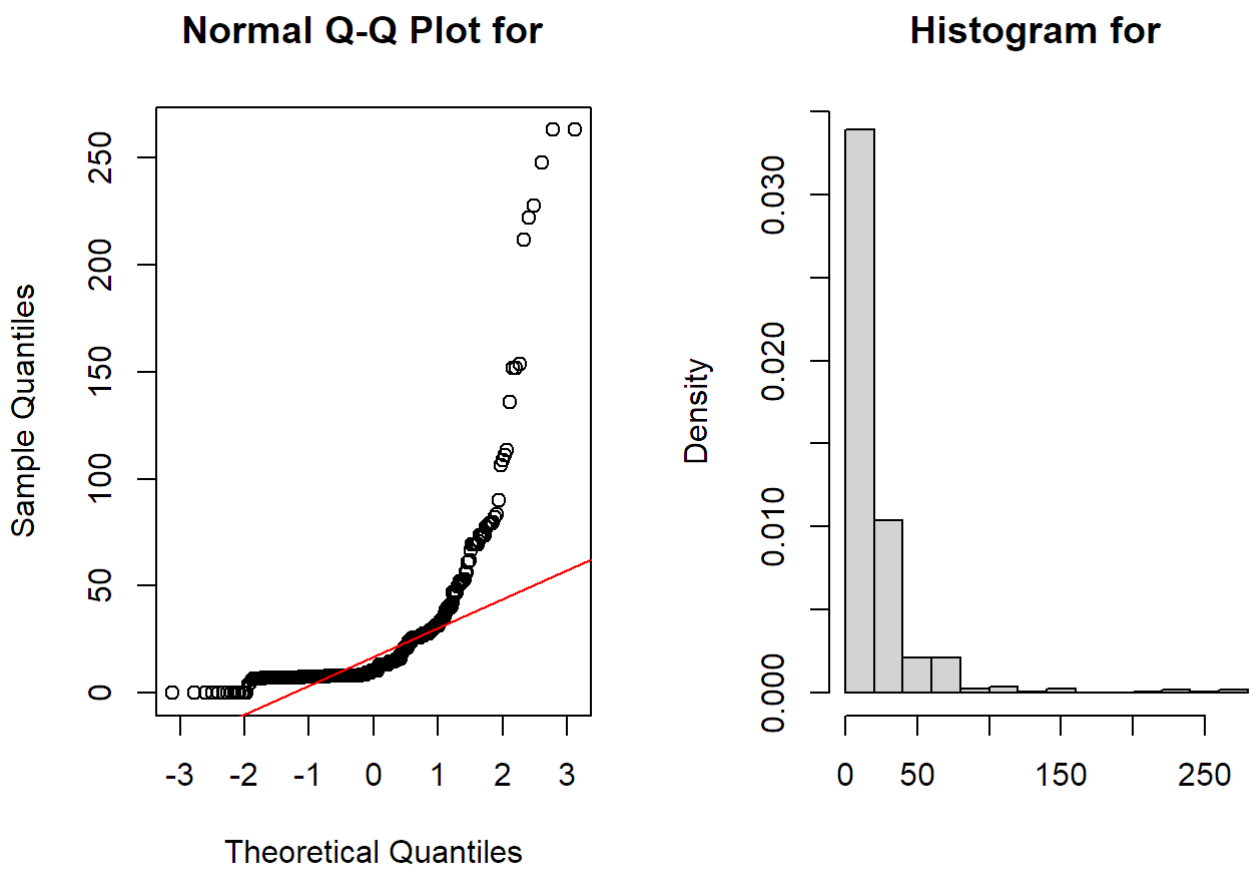
```
shapiro.test(Famsize_sur_1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Famsize_sur_1
## W = 0.76303, p-value < 2.2e-16
```

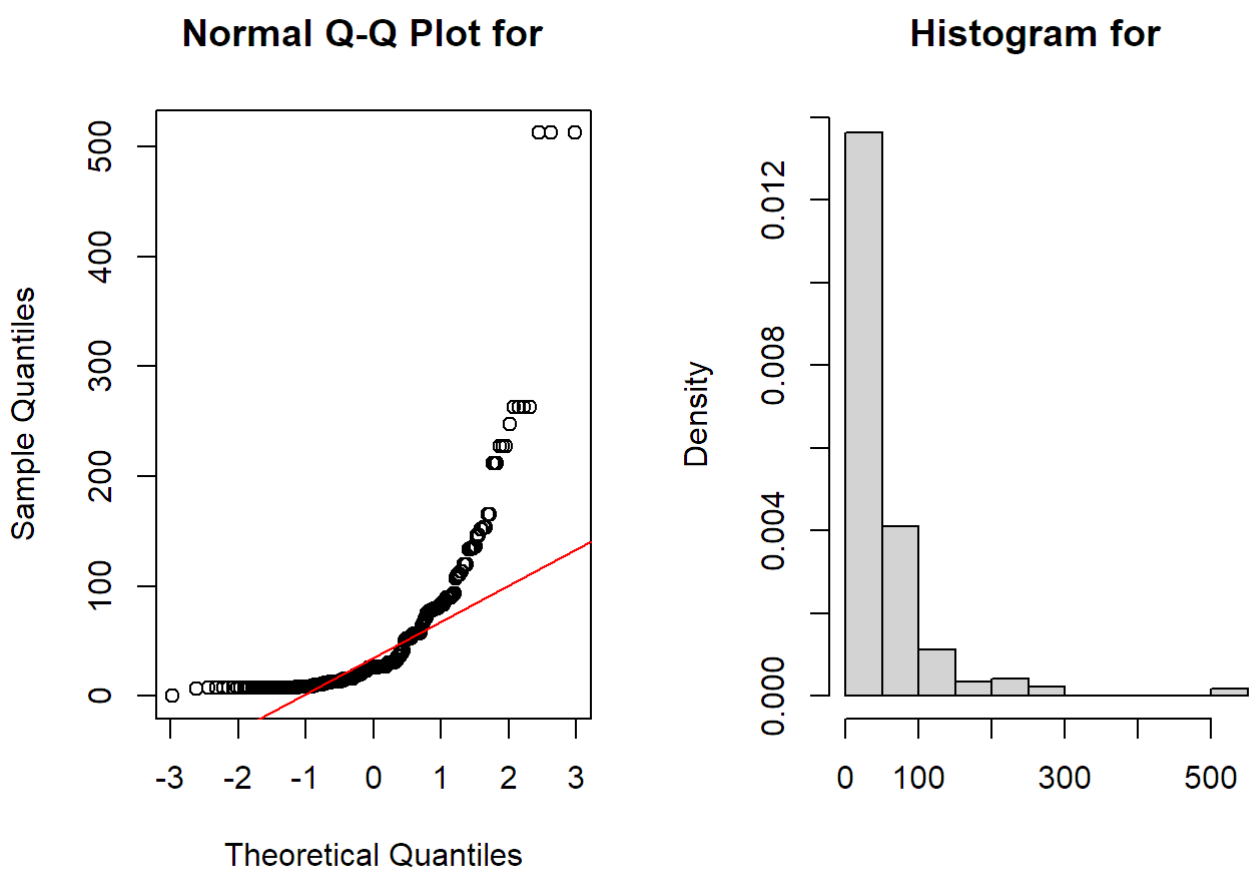
Dado los resultados anteriores, se observa que para las cuatro características consideradas sus correspondientes valores son inferiores al nivel de significación (p = 0.05). Por tanto, rechazamos la H0 y concluimos con un 95% de confianza que los datos no se distribuyen normalmente.

Ahora se aplicará este test para realizar el contraste de si existen diferencias en la característica Tarifa (Fare) en función de la supervivencia (Survived).

```
Fare_sur_0 <- clean_train_data$Fare[clean_train_data$Survived==0]
Fare_sur_1 <- clean_train_data$Fare[clean_train_data$Survived==1]

par(mfrow=c(1,2))
qqnorm(Fare_sur_0, main = paste("Normal Q-Q Plot for ", colnames(Fare_sur_0)[1]))
qqline(Fare_sur_0, col="red")
hist(Fare_sur_0,
main=paste("Histogram for ", colnames(Fare_sur_0)[1]),
xlab=colnames(Fare_sur_0)[1], freq = FALSE)
```




```
par(mfrow=c(1,2))
qqnorm(Fare_sur_1, main = paste("Normal Q-Q Plot for ", colnames(Fare_sur_1)[1]))
qqline(Fare_sur_1, col="red")
hist(Fare_sur_1,
main=paste("Histogram for ", colnames(Fare_sur_1)[1]),
xlab=colnames(Fare_sur_1)[1], freq = FALSE)
```




```
shapiro.test(Fare_sur_0)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Fare_sur_0
## W = 0.51304, p-value < 2.2e-16
```

```
shapiro.test(Fare_sur_1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Fare_sur_1
## W = 0.59673, p-value < 2.2e-16
```

Dado los resultados anteriores, se observa que para las cuatro características consideradas sus correspondientes p-valores son inferiores al nivel de significación (p = 0.05). Por tanto, rechazamos la H0 y concluimos con un 95% de confianza que los datos no se distribuyen normalmente.

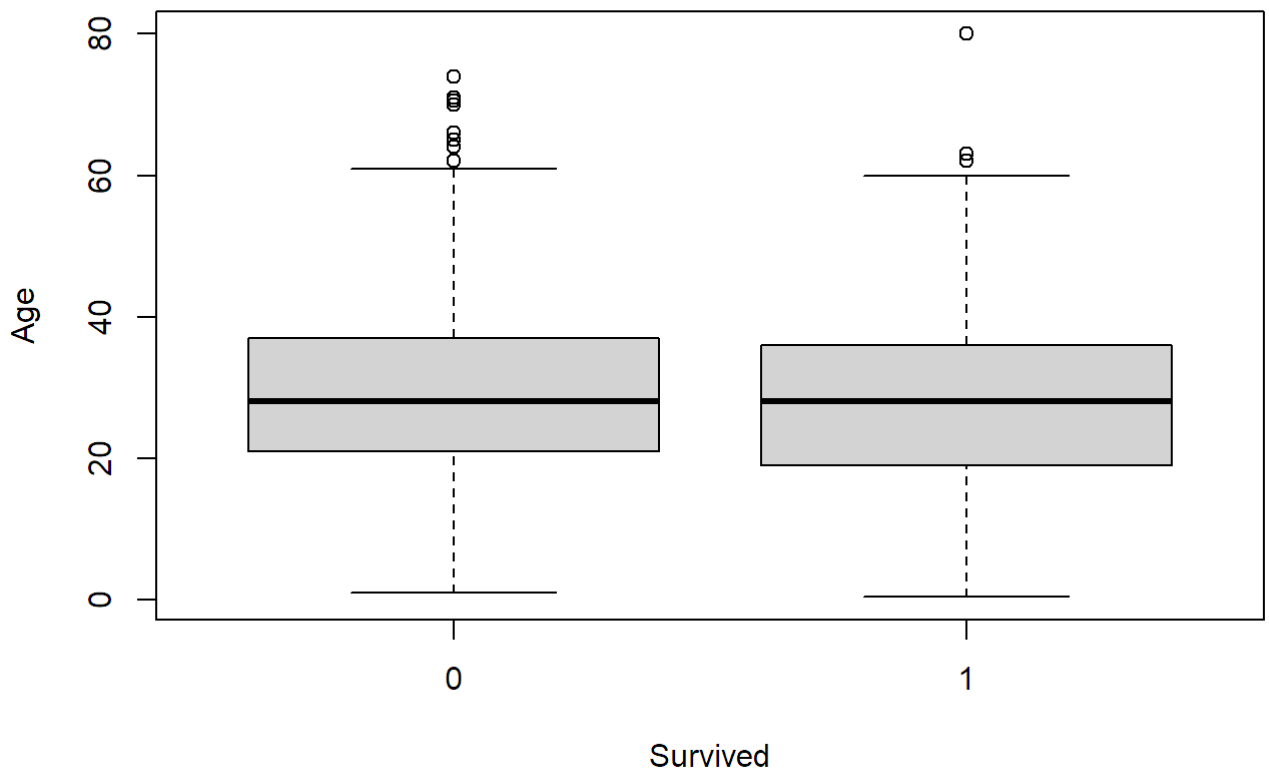
Homegeneidad de la varianza

En este apartado se estudiará la homegeneidad de la varianza utilizando el test de Filgener-killeen. Se trata de un test no paramétrico que compara las varianzas basándose en la mediana. Es una alternativa cuando no se cumple la condición de normalidad en las muestras. De tal forma que la hipótesis nula (H0) y la alternativa (H1) se pueden escribir de la siguiente forma:

- Hipótesis nula (Ho): Todas las varianzas de las poblaciones son iguales.
- Hipótesis alternativa (H1): Al menos dos de ellos difieren.
- Zona de rechazo. Para todo valor de probabilidad mayor que un nivel de significación = 0.05, se acepta Ho y se rechaza H1.

Para realizar el test Fligner-Killeen se utiliza la función fligner.test(). A continuación, se muestra la aplicación del test Fligner-Killeen para la característica cuantitativas Edad (Age) en función de la Supervivencia (Survived):

```
boxplot(Age ~ Survived, data = clean_train_data)
```




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

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Age by Survived
## Fligner-Killeen:med chi-squared = 3.9552, df = 1, p-value = 0.04673
```

Puesto que obtenemos un p-valor superior al nivel de significación (p = 0.05), aceptamos la hipótesis nula (H0), es decir, de que las varianzas de ambas muestras son homogéneas.

A continuación, se muestra la aplicación del test Fligner-Killeen para característica familiares a bordo (Famsize) en función de la Supervivencia (Survived):

```
fligner.test(Famsize ~ Survived, data = clean_train_data)
```

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

Puesto que obtenemos un p-valor superior al nivel de significación (p = 0.05), aceptamos la hipótesis nula (H0), es decir, de que las varianzas de ambas muestras son homogéneas.

A continuación, se muestra la aplicación del test Fligner-Killeen para la característica Tarifa (Fare) en función de la Supervivencia (Survived):

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

Puesto que obtenemos un p-valor inferior al nivel de significación (p = 0.05), rechazamos la hipótesis nula (H0), y podemos concluir que las varianzas son significativamente diferentes.

Apliación de pruebas estadísticas para comparar los grupos de datos.

Contraste de hipótesis

En este apartado se aplicará un contraste de hipótesis sobre dos muestras para determinar si la supervivencia dependiendo de otra variable categórica. Para comparar la dependencia entre dos variables categóricas se utilizará la prueba de chi-cuadrado. El contraste de hipótesis a realizar se expresa así:

- Hipótesis nula (Ho). Los dos factores son independientes.
- Hipótesis alternativa (H1): Los dos factores son dependentes.
- Zona de rechazo. Para todo valor de probabilidad mayor que un nivel de significación p = 0.05, se acepta Ho y se rechaza H1.

Una vez establecido las hipótesis para cada conjunto de variables categóricas consideradas se construirá su correspondiente tabla de contingencia y se aplicará el test chi-cuadrado, para ello se empleará la función chisq.test().

A continuación, se calculan la prueba chi-cuadrado para varios pares de variables categóricas.

Variables Survived-Sex

```
tbl = table(clean_train_data$Survived, clean_train_data$Sex)
tbl
```

```
##
##     female male
##   0     81  468
##   1    233  109
```

Aplicamos la función chisq.test a la tabla de contingencia tbl:

```
chisq.test(tbl)
```

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

Como el valor de p-valor es menor que el nivel de significancia de 0.05, por tanto rechazamos la hipótesis nula (H0) y aceptamos la hipótesis alternativa. Por tanto, concluimos que la supervivenvia depende del sexo del pasajero (Sex)

Variables Survived-Pclass

```
tbl = table(clean_train_data$Survived, clean_train_data$Pclass)
tbl
```

```
##
##       1   2   3
##   0  80  97 372
##   1 136  87 119
```

Aplicamos la función chisq.test a la tabla de contingencia tbl:

```
chisq.test(tbl)
```

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

Como el valor de p-valor es menor que el nivel de significancia de 0.05, por tanto rechazamos la hipótesis nula (H0) y aceptamos la hipótesis alternativa. Por tanto, concluimos que la supervivenvia depende la clase del pasajero (Pclass).

Variables Survived-Fsize

```
tbl = table(clean_train_data$Survived, clean_train_data$Fsize)
tbl
```

```
##
##     large singleton small
##   0    52       374   123
##   1    10       163   169
```

Aplicamos la función chisq.test a la tabla de contingencia tbl:

```
chisq.test(tbl)
```

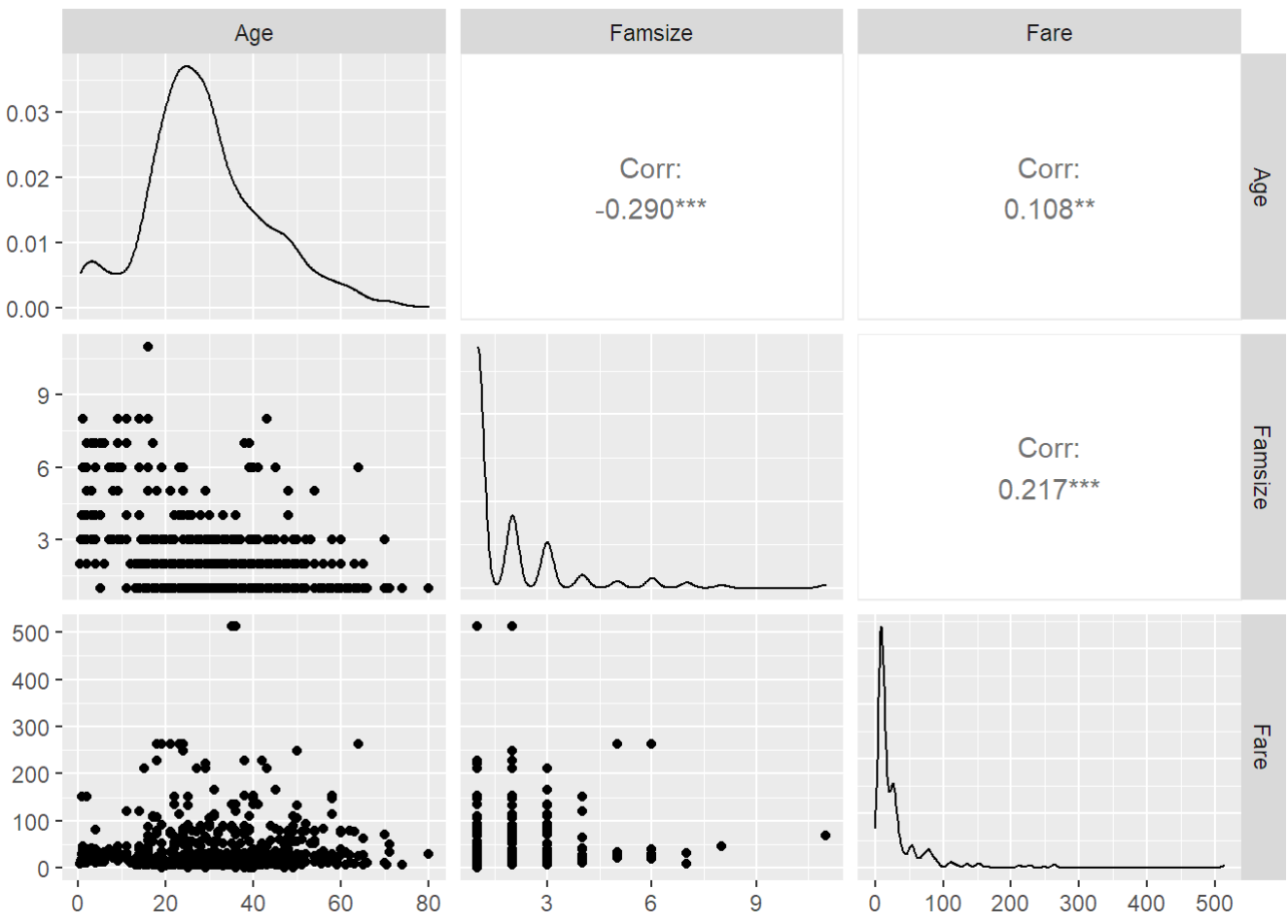
```
##
##  Pearson's Chi-squared test
##
## data:  tbl
## X-squared = 74.537, df = 2, p-value < 2.2e-16
```

Como el valor de p-valor es menor que el nivel de significancia de 0.05, por tanto rechazamos la hipótesis nula (H0) y aceptamos la hipótesis alternativa. Por tanto, concluimos que la supervivenvia depende del tamaño de la familia (Fsize)

Corelación

En este apartado procedemos a realizar un análisis de correlación entre las distintas variables numéricas del conjunto de datos. Cuando dos características o más tienen correlación, eso significa que se están explicando unas a otras al tiempo con lo que proporcionan solo poca o ninguna información nueva.

```
# Calculamos las correlaciones.
corr_data <- select_if(clean_train_data, is.numeric)

corr.res <- cor(corr_data)
# Mostramos las gráficas
ggpairs(corr_data)
```



La gráfica anterior muestra que existe una correlación positiva entre las variables Parch y SibSp. Esto tiene sentido debido a que ambas variables hacen referencia al tamaño de la familia que va a bordo.

Regresión Logística

La estrategia por seguir será partir de un modelo donde la supervivencia dependa de la Edad (Age), la tarifa (Fare), el tamaño de la familia a bordo (Fsize), la embarcación (Embarked), el sexo (Sex) y la clase (Pclass). Partiendo de esta modelo ser irá añadiendo y quitando variables con el propósito de mejorar el modelo. En primer lugar, establecemos categorías de referencia para las variables cualitativas: “F” para la variable Sex, “S” para la variable Embarked, “1” para la variable Pclass, “small” para la variable Fsize; para ello utilizamos la función relevel().

```
# Nivel de significancia

sig_level = 0.05

# Establecemos categoria de referencia conjunto de datos.

clean_train_data$SexR <- relevel(clean_train_data$Sex, ref="female")
clean_train_data$EmbarkedR <- relevel(clean_train_data$Embarked, ref="S")
clean_train_data$PclassR <- relevel(clean_train_data$Pclass, ref="1")
clean_train_data$FsizeD <- relevel(clean_train_data$Fsize, ref="small")

# Establecemos categoria de referencia conjunto de pruebas

clean_test_data$SexR <- relevel(clean_test_data$Sex, ref="female")
clean_test_data$EmbarkedR <- relevel(clean_test_data$Embarked, ref="S")
clean_test_data$PclassR <- relevel(clean_test_data$Pclass, ref="1")
clean_test_data$Fsize <- relevel(clean_test_data$Fsize, ref="small")
```

Calculamos la supervivencia en función de las carácteristicas:

• Modelo 1. Survived = Age + SibSp + Parch + Fare + EmbarkedR + SexR + PclassR.

• Modelo 2. Survived = Age + SibSp + Parch + Fare + EmbarkedR + SexR + PclassR + Fsize.

• Modelo 3. Survived = Age + Fare + SexR + PclassR + Fsize.

• Modelo 4. Survived = Age + SexR + PclassR + Fsize.

```
# Calculamos modelo 1
glm1.fit <- glm(factor(Survived) ~ Age + Fare +
                  EmbarkedR + SexR + PclassR,
                data = clean_train_data,
                family = "binomial")

glm1.summary <- summary(glm1.fit)



# Calculamos modelo 2
glm2.fit <- glm(factor(Survived) ~ Age + Fare +
                  EmbarkedR + SexR + PclassR +
                  Fsize,
                data = clean_train_data,
                family = "binomial")

glm2.summary <- summary(glm2.fit)


# Calculamos modelo 3

glm3.fit <- glm(factor(Survived) ~ Age + Fare +
                  SexR + PclassR +
                  Fsize,
                data = clean_train_data,
                family = "binomial")

glm3.summary <- summary(glm3.fit)


# Calculamos modelo 4

glm4.fit <- glm(factor(Survived) ~ Age +
                  SexR + PclassR +
                  Fsize,
                data = clean_train_data,
                family = "binomial")

glm4.summary <- summary(glm4.fit)
```

Para los anteriores modelos de regresión logística obtenidos, la bondad del modelo se evaluará mediante la medida AIC. Dado que esta medida tiene en cuenta tanto la bondad del ajuste como la complejidad del modelo, cuando se comparen varios modelos candidatos, se seleccionará aquel que resulte en el menor AIC. Para obtener los AIC’s de los modelos se utiliza la función AIC().

```
aui_data <- AIC(glm1.fit, glm2.fit, glm3.fit, glm4.fit)
kable(aui_data) %>%
kable_styling(bootstrap_options = "striped", full_width = F)
```

	df	AIC
glm1.fit	9	813.4902
glm2.fit	11	785.5319
glm3.fit	8	781.6580
glm4.fit	7	780.7780

Dado los resultados anteriores se llega a la conclusión que se obtiene el mejor resultado con el modelo regresor 1 con un valor de 813.49.

```
glm1.summary
```

```
##
## Call:
## glm(formula = factor(Survived) ~ Age + Fare + EmbarkedR + SexR +
##     PclassR, family = "binomial", data = clean_train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6051  -0.6648  -0.3995   0.6324   2.5192
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   3.494561   0.452484   7.723 1.14e-14 ***
## Age          -0.035299   0.007638  -4.622 3.81e-06 ***
## Fare         -0.000428   0.002137  -0.200 0.841248
## EmbarkedR    12.922239 613.182693   0.021 0.983187
## EmbarkedRC    0.488683   0.236405   2.067 0.038721 *
## EmbarkedRQ    0.510629   0.317444   1.609 0.107712
## SexRmale     -2.556413   0.189549 -13.487  < 2e-16 ***
## PclassR2     -1.092117   0.301032  -3.628 0.000286 ***
## PclassR3     -2.476621   0.309205  -8.010 1.15e-15 ***
## ---
## 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:  795.49  on 882  degrees of freedom
## AIC: 813.49
##
## Number of Fisher Scoring iterations: 13
```

```
glm1.coef <- coef(glm1.fit)
glm1.coef_exp <- exp(coef(glm1.fit))
data <- data.frame(Coeficiente = glm1.coef, Exp = glm1.coef_exp)
kable(data) %>%
kable_styling(bootstrap_options = "striped", full_width = F)
```

	Coeficiente	Exp
(Intercept)	3.4945613	3.293584e+01
Age	-0.0352988	9.653170e-01
Fare	-0.0004280	9.995721e-01
EmbarkedR	12.9222391	4.093145e+05
EmbarkedRC	0.4886832	1.630168e+00
EmbarkedRQ	0.5106293	1.666340e+00
SexRmale	-2.5564128	7.758250e-02
PclassR2	-1.0921174	3.355054e-01
PclassR3	-2.4766209	8.402670e-02

Del modelo anterior podemos concluir que:

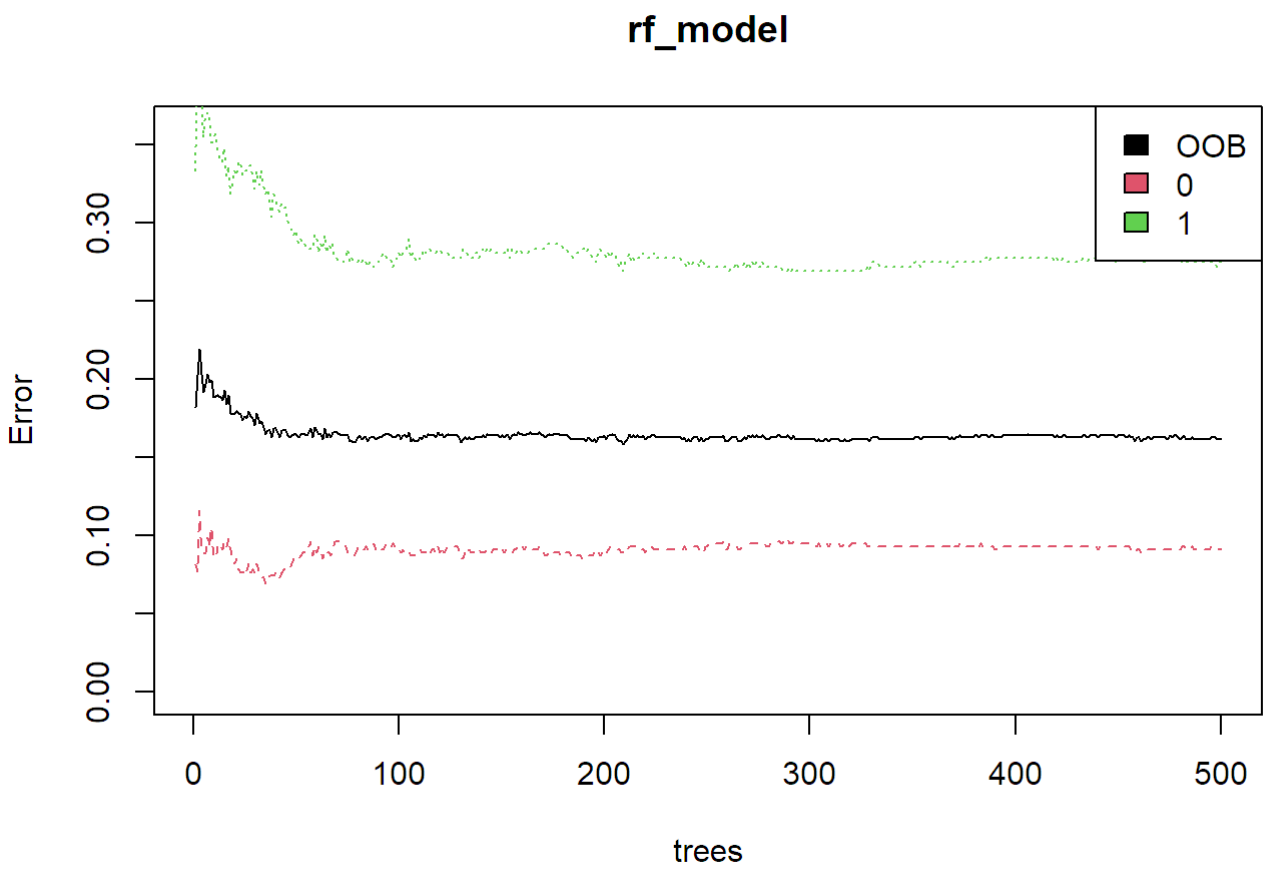
• Para las variables Age, SexRMale, PclassR2 y PclassR3, sus correspondientes p-valores son menores que 0.05, es decir, son significativas para el modelo.

• El resto de variables tienen p-valores son mayores que 0.05, no son significativas y se pueden eliminar del modelo.

Random forest

El RF es un método de clasificación basado en la realización de múltiples árboles de decisión sobre muestras de un conjunto de datos Además, Random Forest permite obtener medidas acerca de la importancia que los diferentes predictores han tenido en el modelo, lo que permite en parte interpretar este. La importancia de los predictores se evalúa como el número de veces que han sido utilizados por los diversos árboles y su capacidad para reducir el índice de Gini en ellos.

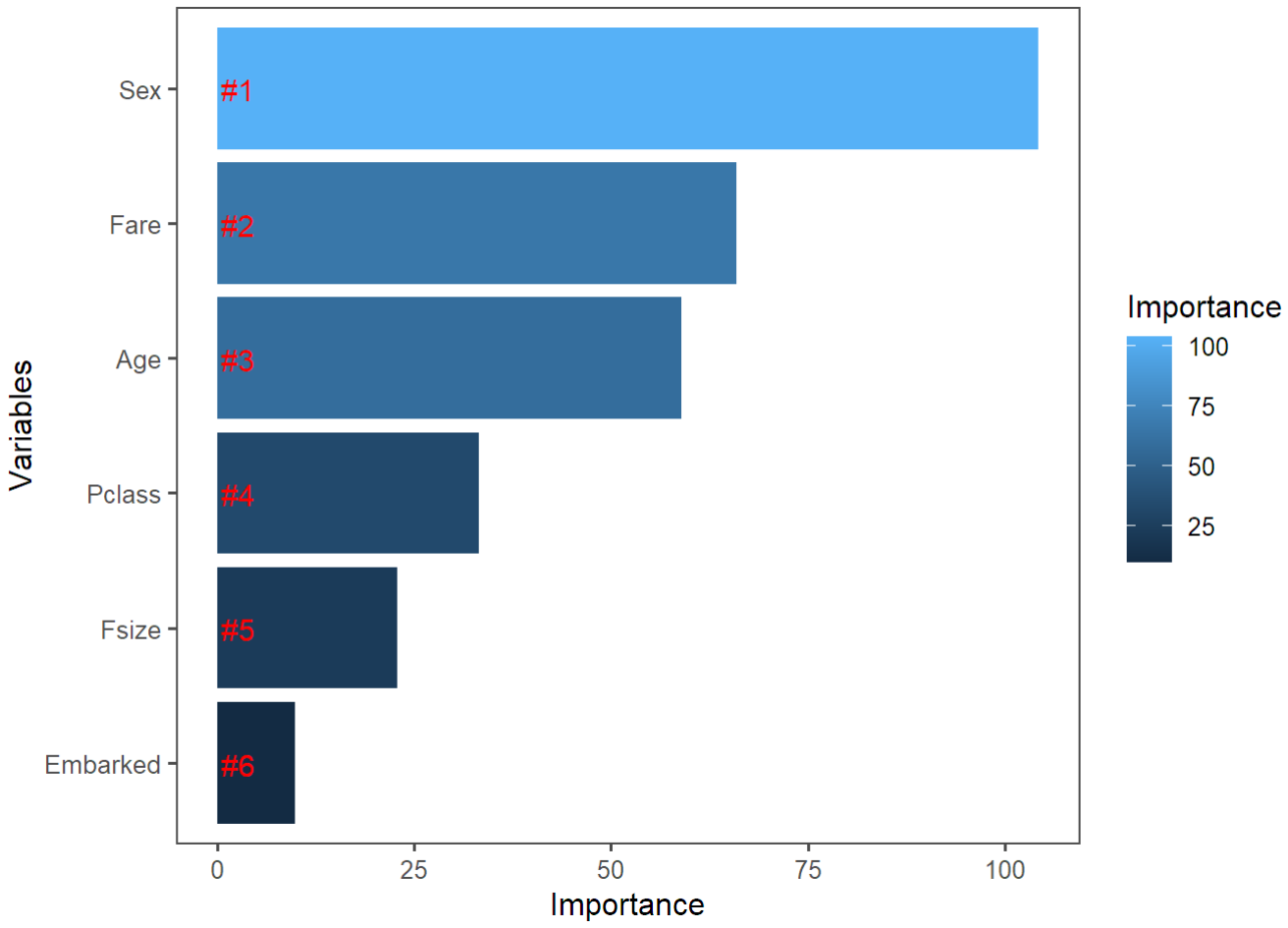
```
#Set a random seed
set.seed(754)

# Build the model
rf_model <- randomForest(factor(Survived) ~ Pclass + Sex + Age +
                           Fare + Embarked + Fsize,
                         data = clean_train_data)

# Show model error
plot(rf_model, ylim=c(0,0.36))
legend('topright', colnames(rf_model$err.rate), col=1:3, fill=1:3)
```



La línea negra muestra la tasa de error general que cae por debajo del 20%. Las líneas rojas y verdes muestran la tasa de error de “muerto” y “sobrevivió” respectivamente. Con alrededor del 10%, nuestro modelo parece ser bueno para predecir mejor la muerte que la supervivencia.

A continuación, se comprobara la importancia de la variable relativa al explorar la disminución media en Gini calculada en todos los árboles.

```
#Get importance
importance <- importance(rf_model)
varImportance <- data.frame(Variables = row.names(importance),
                            Importance = round(importance[ ,'MeanDecreaseGini'],2))

# Create a rank variable based on importance
rankImportance <- varImportance %>%
  mutate(Rank = paste0('#',dense_rank(desc(Importance))))

# Use ggplot2 to visualize the relative importance of variables
ggplot(rankImportance, aes(x = reorder(Variables, Importance),
                           y = Importance, fill = Importance)) +
  geom_bar(stat='identity') +
  geom_text(aes(x = Variables, y = 0.5, label = Rank),
            hjust=0, vjust=0.55, size = 4, colour = 'red') +
  labs(x = 'Variables') +
  coord_flip() +
  theme_few()
```



En la gráfica anterior se observa que la variables Fare se considera la más importantes. Esto contradice al método de regresión logística que las consideraba no significativas. Por otra parte, la variable SibSp está clasificada en séptimo lugar; mientras en la regresión logística era estadísticamente significativa. Sin embargo, la variable Fsize clasifica mejor que las variables SibSp y Parch. Esto tiene sentido ya que Fsize es la discretización de la combinación de estas dos variables.

Representacó de los resultados a partir de tablas y gráficas

Este apartado se ha ido desarrollando a lo largo de esta práctica en apartados anteriores, mediante diagramas de barras, boxplots, tablas, ...

Resolución del problema

En este trabajo se trató de la problemática de determinar qué variables influyeron más sobre la supervivencia de los pasajeros a bordo del Titanic. Para llevar a cabo esta tarea se realizó se utilizó el conjunto de datos de entrenamiento y conjunto de datos de prueba.

Sobre este conjunto de datos se realizó una fase de preprocesamiento que incluye varias tareas de limpieza de datos, (tales como, conversiones, eliminación los valores perdidos o nulos), discretización de valores numéricos, etc. En la imputación de valores perdidos se pueden destacar el trabajo realizado en la variable Edad (Age). Para la imputación de los valores perdidos de la característica edad (Age) se empleó el algoritmo KNN (K-Nearest Neighbour Imputation) de R. Se realizaron pruebas estadísticas para comprobar las dependencias entre Supervivencia y otras variables categóricas del conjunto de datos. Se identifico que existen evidencias estadísticas de dependencias entre la Supervivencia y las variables categóricas: Sexo (Sex), Clase (Pclass) y Tamaño de la familia (Fsize).

Recursos

El desarrollo de la práctica se fundamenta con el material didáctico visto durante el curso:

- Calvo M., Subirats L., Pérez D. (2019). Introducción a la limpieza y análisis de los datos. Editorial UOC.
- Megan Squire (2015). Clean Data. Packt Publishing Ltd
- Ramos Lorenzo, C. (2019). RPubs - Logistic Regressión. https://rpubs.com/MrCristianrl/500969 (https://rpubs.com/MrCristianrl/500969)

El conjunto de datos utilizado está publicado en el repositorio Kaggle y accesible a través de la siguiente url:

- Titanic: Machine Learning from Disaster. (https://www.kaggle.com/c/titanic)