Practica 2

María Pérez Ameneiro

5/1/2021

0 Carga del archivo

Se procede a la carga del archivo **Titanic.csv** añadiendo que el separador es ; en vez de , para su posterior tratamiento. Se guarda en el dataframe **base** y se procede a enseñar por pantalla 5 filas para comprobar su correcta carga.

```
base <- read.csv("Titanic.csv", sep=",",dec=".")
head(base)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                 Name    Sex Age SibSp Parch
## 1                             Braund, Mr. Owen Harris   male  22     1     0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
## 3                              Heikkinen, Miss. Laina female  26     0     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
## 5                            Allen, Mr. William Henry   male  35     0     0
## 6                                    Moran, Mr. James   male  NA     0     0
##             Ticket    Fare Cabin Embarked
## 1        A/5 21171  7.2500              S
## 2         PC 17599 71.2833   C85        C
## 3 STON/O2. 3101282  7.9250              S
## 4           113803 53.1000  C123        S
## 5           373450  8.0500              S
## 6           330877  8.4583              Q
```

Cargamos las librerías necesarias

```
library(Rcmdr)
```

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

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

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

```
## Warning: package 'car' was built under R version 4.0.3
```

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

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

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

```
## Registered S3 methods overwritten by 'lme4':
##   method                          from
##   cooks.distance.influence.merMod car
##   influence.merMod                car
##   dfbeta.influence.merMod         car
##   dfbetas.influence.merMod        car
```

```
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```
## La interfaz R-Commander sólo funciona en sesiones interactivas
```

```
##
## Attaching package: 'Rcmdr'
```

```
## The following object is masked from 'package:base':
##
##     errorCondition
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.3
```

```
library(ISLR)
```

```
## Warning: package 'ISLR' was built under R version 4.0.3
```

```
library(MASS)
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.0.3
```

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

```
## The following object is masked from 'package:MASS':
##
##     select
```

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

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

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

```
library(nortest)
```

```
## Warning: package 'nortest' was built under R version 4.0.3
```

```
library(tables )
```

```
## Warning: package 'tables' was built under R version 4.0.3
```

1 Descripción del dataset

El dataset fue obtenido a partir del enlace de Kaggle y esta constituido por

• PassengerId. Identificador del pasajero registrado en la base de datos • Survived. Indica si sobrevivió, siendo 1 positivo y 0 negativo • Pclass. Clase del tique. Tomando valores de 1, 2 y 3 (1ª,2ª y 3ª) • Name. Nombre del pasajero • Sex. Sexo del pasajero. • Age. Edad en años • Sib. Nº de hermanos o cónyuges embarcados • Sp Parch. Nº de hijos o padres embarcados • Ticket. Nº de tique • Fare. Precio del tique • Cabin. Tipo de camarote • Embarked. Puerto embarcado.

###¿Por qué es importante y qué pregunta/problema pretende responder?

Para esta práctica se ha utilizado el dataset Titanic: Machine Learning from Disater debido a que me parece interesante al estar relacionado con mis estudios anteriores y por su influencia en el mundo marítimo. Debido a él se reúne la OMI (Organización Mundial Internacional) y crea el primer Convenio Internacional para la Seguridad de la Vida Humana en el Mar (SOLAS), el cual fue revisado en varias ocasiones y es de aplicación actualmente. El dataset es importante debido porque nos va a proporcionar información acerca del precio del billete según la edad, sexo, la clase del billete de la camarote, familiares a bordo y puerto de embarque. Nos va a dar información como nos influyen unos factores mas que otros o no influyen en el precio. Esto nos puede servir para como ejercicio para ejercer una campaña de publicidad desde la competencia al conocer los precios de los billetes. También se va a hacer una regresión lineal para ver qué factores influyeron en la supervivencia del pasaje del buque, lo cual nos puede proporcionar si es más seguro viajar en clases altas (camarotes encima de la flotación) o la edad influye (más joven mayor capacidad de resistencia) o familiares a bordo (se intenta salvar a los familiares).

2. Integración y selección de los datos de interés a analizar.

Los datos que no vamos a usar de la base de datos son los siguientes: - Idpassanger. No es un dato relevante porque es una correlación de números. - Name. Los nombres no son interesantes porque los datos para uso de evaluación de datos deben estar anonizados para no vulnerar la legislación -Ticktet. El identificador del tique no proporciona información relevante en nuestro estudio. - Cabin. El numero de camarote no

proporciona información que nos influya en el estudio. Los restantes datos se van a evaluar de cómo influye en el precio del billete los parámetros siguientes: • Pclass. • Sex. • Age. • SibSp. • Parch • Embarked. Puerto embarcado.

Se procede a eliminar las variables no necesarias y guardarlas en un nueva base de datos base2.

```
#Se crea la nueva base de datos
#Eliminan columnas de no necesarias
borrar <- c("PassengerId","Name","Ticket","Cabin")
base2 <- base[ , !(names(base) %in% borrar)]
head(base2, n=9)
```

```
##   Survived Pclass    Sex Age SibSp Parch    Fare Embarked
## 1        0      3   male  22     1     0  7.2500        S
## 2        1      1 female  38     1     0 71.2833        C
## 3        1      3 female  26     0     0  7.9250        S
## 4        1      1 female  35     1     0 53.1000        S
## 5        0      3   male  35     0     0  8.0500        S
## 6        0      3   male  NA     0     0  8.4583        Q
## 7        0      1   male  54     0     0 51.8625        S
## 8        0      3   male   2     3     1 21.0750        S
## 9        1      3 female  27     0     2 11.1333        S
```

```
#base2<-data.frame(base$Survived,base$Pclass,base$Sex,base$Age,base$SibSp,base$Parch,base$Far
e,base$Embarked)
#Se ven los tipos de variables de la base de datos
str(base2)
```

```
## 'data.frame':    891 obs. of  8 variables:
##  $ 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" ...
```

3 Limpieza de datos

3.1 ¿Los datos contienen ceros o elementos vacíos?

Se va a evaluar cada variable para comprobar si hay elementos vacíos o ceros. En cada caso se procederá a cambiar los elementos vacios por un valor y los valores cero se comprobará que son correctos o no. En caso negativo, se procederá a cambiar el valor por el mismo procedimiento que el caso de elemento vacío.

Survived

```
#Se comprueban si hay valores vacíos
which(is.na(base2$Survived))
```

```
## integer(0)
```

```
#En este caso no hay valores nulos
table(base2$Survived)
```

```
##
##   0   1
## 549 342
```

```
#Se comprueba que ha valores cero debido a que corresponde a la gente que no sobrevive
```

PClass

```
#Se comprueban si hay valores vacíos
which(is.na(base2$Pclass))
```

```
## integer(0)
```

```
#En este caso no hay valores nulos
#Se comprueba que no hay valores cero.
table(base2$Pclass)
```

```
##
##   1   2   3
## 216 184 491
```

```
#Todos los datos se agrupan en los valores establecidos 1,2 y 3.
```

Sex

```
#Se comprueban si hay valores vacíos
which(is.na(base2$Sex))
```

```
## integer(0)
```

```
#En este caso no hay valores nulos
#Se comprueba que no hay valores cero.
table(base2$Sex)
```

```
##
## female   male
##    314    577
```

```
#Todos los datos se agrupan en los valores establecidos hombre y mujer
```

Age

```
#Se comprueban si hay valores vacíos
sel <- which(is.na(base2$Age))
sel
```

```
##   [1]   6  18  20  27  29  30  32  33  37  43  46  47  48  49  56  65  66  77
##  [19]  78  83  88  96 102 108 110 122 127 129 141 155 159 160 167 169 177 181
##  [37] 182 186 187 197 199 202 215 224 230 236 241 242 251 257 261 265 271 275
##  [55] 278 285 296 299 301 302 304 305 307 325 331 335 336 348 352 355 359 360
##  [73] 365 368 369 376 385 389 410 411 412 414 416 421 426 429 432 445 452 455
##  [91] 458 460 465 467 469 471 476 482 486 491 496 498 503 508 512 518 523 525
## [109] 528 532 534 539 548 553 558 561 564 565 569 574 579 585 590 594 597 599
## [127] 602 603 612 613 614 630 634 640 644 649 651 654 657 668 670 675 681 693
## [145] 698 710 712 719 728 733 739 740 741 761 767 769 774 777 779 784 791 793
## [163] 794 816 826 827 829 833 838 840 847 850 860 864 869 879 889
```

```
#En este caso hay valores vacíos y se procede a cambiar por la mediana de los datos
base2[sel,"Age"]<-median(base2$Age,na.rm = T)
#Se vuelve a comprobar si hay valores vacíos
which(is.na(base2$Age))
```

```
## integer(0)
```

```
#No hay valores vacíos
#Se comprueba que no hay valores cero.
which(base2$Age==0)
```

```
## integer(0)
```

```
table(base2$Age)
```

```
##
## 0.42 0.67 0.75 0.83 0.92    1    2    3    4    5    6    7    8    9   10   11
##    1    1    2    2    1    7   10    6   10    4    3    3    4    8    2    4
##   12   13   14 14.5   15   16   17   18   19   20 20.5   21   22   23 23.5   24
##    1    2    6    1    5   17   13   26   25   15    1   24   27   15    1   30
## 24.5   25   26   27   28 28.5   29   30 30.5   31   32 32.5   33   34 34.5   35
##    1   23   18   18  202    2   20   25    2   17   18    2   15   15    1   18
##   36 36.5   37   38   39   40 40.5   41   42   43   44   45 45.5   46   47   48
##   22    1    6   11   14   13    2    6   13    5    9   12    2    3    9    9
##   49   50   51   52   53   54   55 55.5   56   57   58   59   60   61   62   63
##    6   10    7    6    1    8    2    1    4    2    5    2    4    3    4    2
##   64   65   66   70 70.5   71   74   80
##    2    3    1    2    1    2    1    1
```

```
#Se comprueba que no hay valores igual a cero y los que hay cercanos a cero son correctos por que en el buque hubo bebes de meses embarcados.
```

SibSp

```
#Se comprueban si hay valores vacíos
which(is.na(base2$SibSp))
```

```
## integer(0)
```

```
#En este caso no hay valores vacíos
#Se comprueba que si hay valores cero.
table(base2$SibSp)
```

```
##
##   0   1   2   3   4   5   8
## 608 209  28  16  18   5   7
```

```
#Se comprueba que hay valores igual a cero pero es razonable porque puede haber pasajeros que no tengan hermanos o cónyuges embarcados.
```

Parch

```
#Se comprueban si hay valores vacíos
which(is.na(base2$Parch))
```

```
## integer(0)
```

```
#En este caso no hay valores vacíos
#Se comprueba que si hay valores cero.
table(base2$Parch)
```

```
##
##   0   1   2   3   4   5   6
## 678 118  80   5   4   5   1
```

```
#Se comprueba que hay valores igual a cero pero es razonable porque puede haber pasajeros que no tengan hijos o padres embarcados.
```

Fare

```
#Se comprueban si hay valores vacíos
which(is.na(base2$Fare))
```

```
## integer(0)
```

```
#En este caso no hay valores vacíos
#Se comprueba que si hay valores cero.
sel1 <- which(base2$Fare==0)
sel1
```

```
##  [1] 180 264 272 278 303 414 467 482 598 634 675 733 807 816 823
```

```
#Se comprueba que hay valores igual a cero pero no es razonable siendo substituidos por la mediana
base2[sel1,"Fare"]<-median(base2$Fare,na.rm = T)
#Se comprueba que se hizo correctamente el cambio
which(base2$Fare==0)
```

```
## integer(0)
```

Embarked

```
#Se comprueban si hay valores vacíos
sel2 <- which(base2$Embarked!="C" & base2$Embarked!="S" & base2$Embarked!="Q")
sel2
```

```
## [1]  62 830
```

```
#Se cambian estos valores por el primer puerto S = Southampton
base2$Embarked[sel2]<-"S"
base2[sel2,]
```

```
##     Survived Pclass    Sex Age SibSp Parch Fare Embarked
## 62         1      1 female  38     0     0   80        S
## 830        1      1 female  62     0     0   80        S
```

```
#Se comprueba que se hizo correctamente el cambio
which(base2$Embarked!="C" & base2$Embarked!="S" & base2$Embarked!="Q")
```

```
## integer(0)
```

```
table(base2$Embarked)
```

```
##
##   C   Q   S
## 168  77 646
```

Se convierten las variables survived, pclass, sex y embarked.

```
#Se convierten en factores las variables survived, pclass, sex y embarked.
base2$Survived <- as.factor(base2$Survived)
base2$Pclass <- as.factor(base2$Pclass)
base2$Sex <- as.factor(base2$Sex)
base2$Embarked <- as.factor(base2$Embarked)
#Se vuelve a comprobar que los tipos de variables son acordes
str(base2)
```

```
## 'data.frame':    891 obs. of  8 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 ...
##  $ Sex     : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age     : num  22 38 26 35 35 28 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/ 3 levels "C","Q","S": 3 1 3 3 3 2 3 3 3 1 ...
```

###3.2.Identificación y tratamiento de valores extremos.

Se va a comprobar los valores extremos para las variables age y fare debido a que las demás variables son factores o se comprobó que no hay valores extraños en el anterior apartado.

Age

```
#Se va a comprobar los valores estadísticos
summary(base2$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.42   22.00   28.00   29.36   35.00   80.00
```

```
print("Varianza")
```

```
## [1] "Varianza"
```

```
var(base2$Age)
```

```
## [1] 169.5125
```

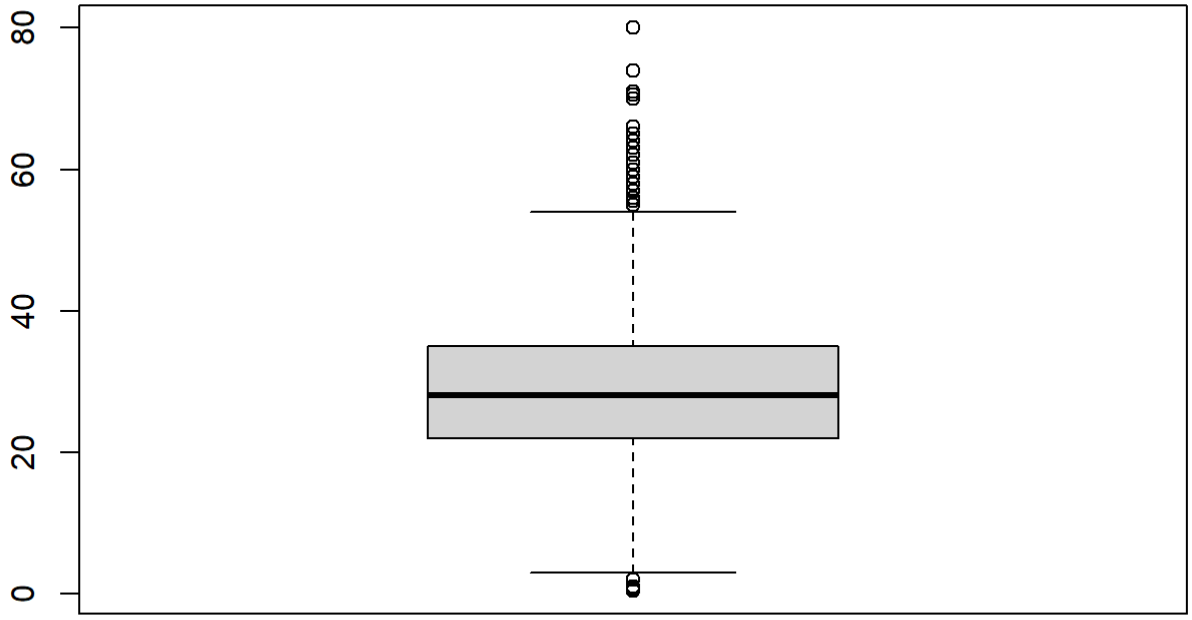
```
print("Desviación")
```

```
## [1] "Desviación"
```

```
sd(base2$Age)
```

```
## [1] 13.0197
```

```
#Se va a hacer un gráfico de cajas para ver los valores
boxplot(base2$Age)
```




```
#Se comprueban que hay valores extremos, se procede a ver estos valores
boxplot.stats(base2$Age)$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
```

```
#Estos valores son válidos porque pudo haber pasajeros con esta edad.
```

Fare

```
#Se va a comprobar los valores estadísticos
summary(base2$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.013   7.925  14.454  32.447  31.000 512.329
```

```
print("Varianza")
```

```
## [1] "Varianza"
```

```
var(base2$Fare)
```

```
## [1] 2457.208
```

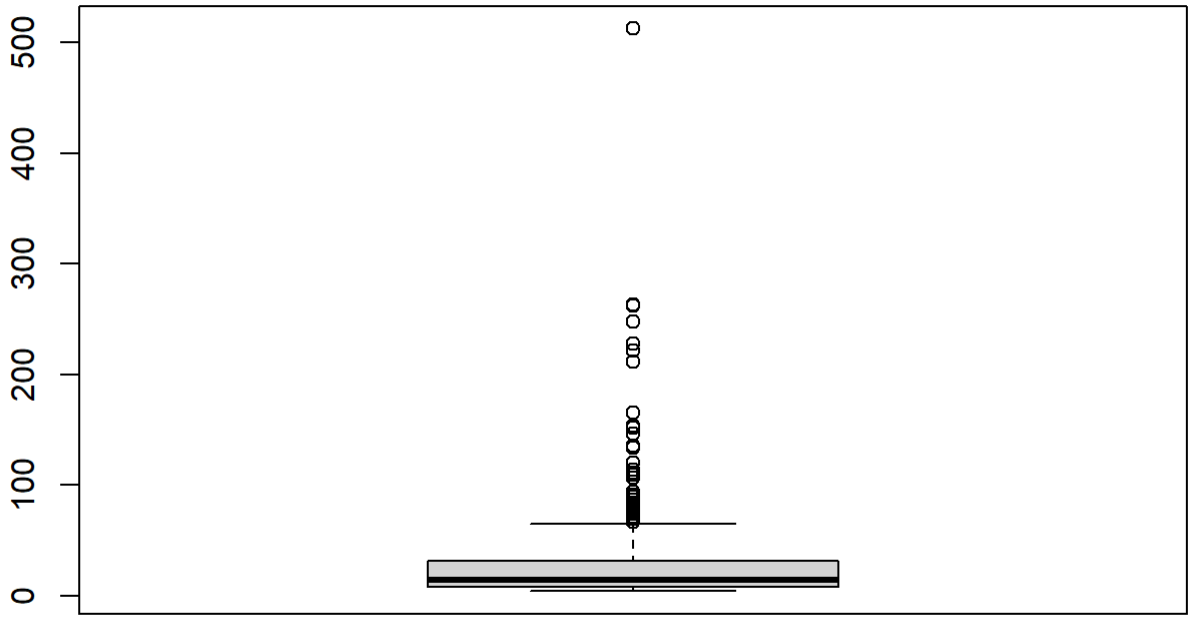
```
print("Desviación")
```

```
## [1] "Desviación"
```

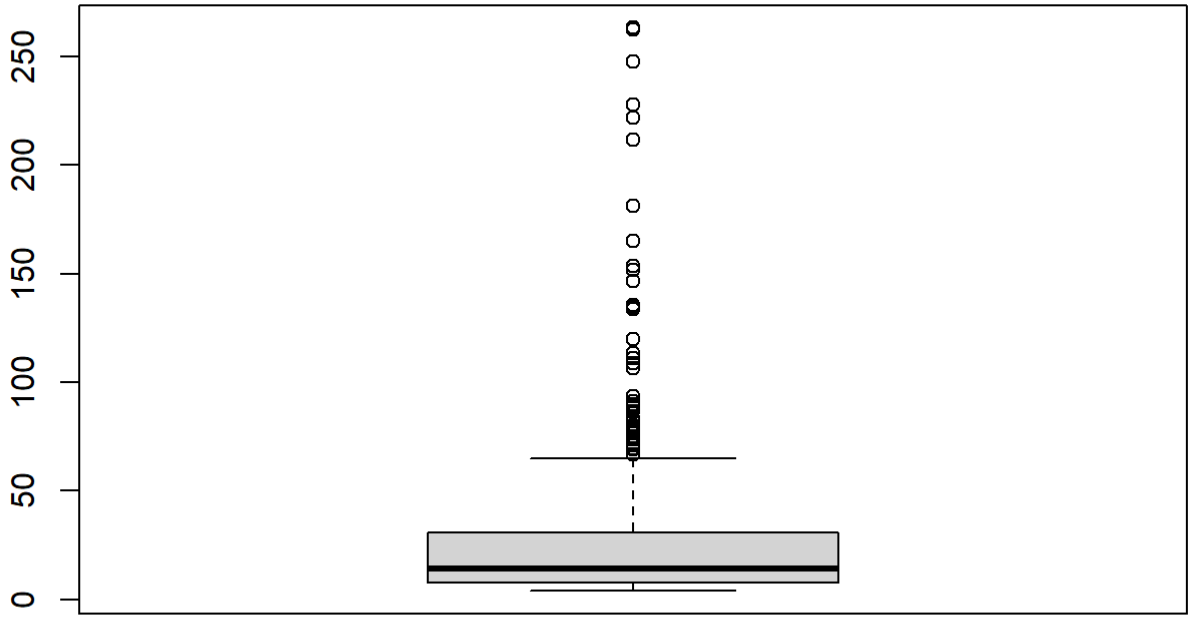
```
sd(base2$Fare)
```

```
## [1] 49.57024
```

```
#Se va a hacer un gráfico de cajas para ver los valores
boxplot(base2$Fare)
```




```
#Se comprueban que hay valores extremos siendo uno muy llamativo (cercano a 500)
#Se procede a cambiar este valor por la media mas 3 veces la desviación
maxi <- max(base2$Fare)
base2$Fare[base2$Fare==max(base2$Fare)] <- mean(base2$Fare)+3*sd(base2$Fare)
#Se procede a comprobar de nuevo el gráfico
boxplot(base2$Fare)
```




```
boxplot.stats(base2$Fare)$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 181.1583  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 181.1583  76.7292 211.3375 110.8833 227.5250 151.5500 227.5250
##  [97] 211.3375 181.1583  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
```

```
#Estos valores son válidos porque se ven precios elevados pero una cierta continuidad sin pun
tos singulares como anteriormente.
#Se vuelven a comprobar los valores estadísticos
summary(base2$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.013   7.925  14.454  31.332  31.000 263.000
```

```
print("Varianza")
```

```
## [1] "Varianza"
```

```
var(base2$Fare)
```

```
## [1] 1754.262
```

```
print("Desviación")
```

```
## [1] "Desviación"
```

```
sd(base2$Fare)
```

```
## [1] 41.88391
```

Se guarda la base de datos final

```
write.csv(base2, file = "Titanic_clean.csv")
```

4 Análisis de los datos

4.1 Selección de los grupos de datos que se quieren analizar.

Se va a analizar el precio del billete según sea mujer o hombre, también por la clase del billete o por el puerto de salida para ello vamos a agrupar los datos:

```
#Agrupamos el precio por sexo
precio.mujer <- base2[base2$Sex != "male",]
precio.hombre <- base2[base2$Sex == "male",]
#Agrupamos el precio por clase
precio.primera <- base2[base2$Pclass ==1,]
precio.segunda <- base2[base2$Pclass ==2,]
precio.tercera <- base2[base2$Pclass ==3,]
#Agrupamos por puerto de salida
precio.cherbourg <- base2[base2$Embarked == "C",]
precio.queenstrown <- base2[base2$Embarked == "Q",]
precio.southampton <- base2[base2$Embarked == "S",]
```

4.2 Comprobación de la normalidad y homogeneidad de la varianza

Para la comprobación de la normalidad de los datos se va a utilizar el test de Shapiro-Wilk con un valor de confianza del 95%.

```
alpha = 0.05
col.names = colnames(base2)

for (i in 1:ncol(base2)){
  if (i==1) cat ("Variables que no siguen una distribución normal: \n")
  if (is.integer(base2[,i])|is.numeric(base2[,i])){
    p_val = shapiro.test(base2[,i])$p.value
    if (p_val < alpha){
      cat(col.names[i])
      #Format output
      if (i < ncol(base2)-1) cat (", ")
        if (i %%3 == 0) cat ("\n")
    }
  }
}
```

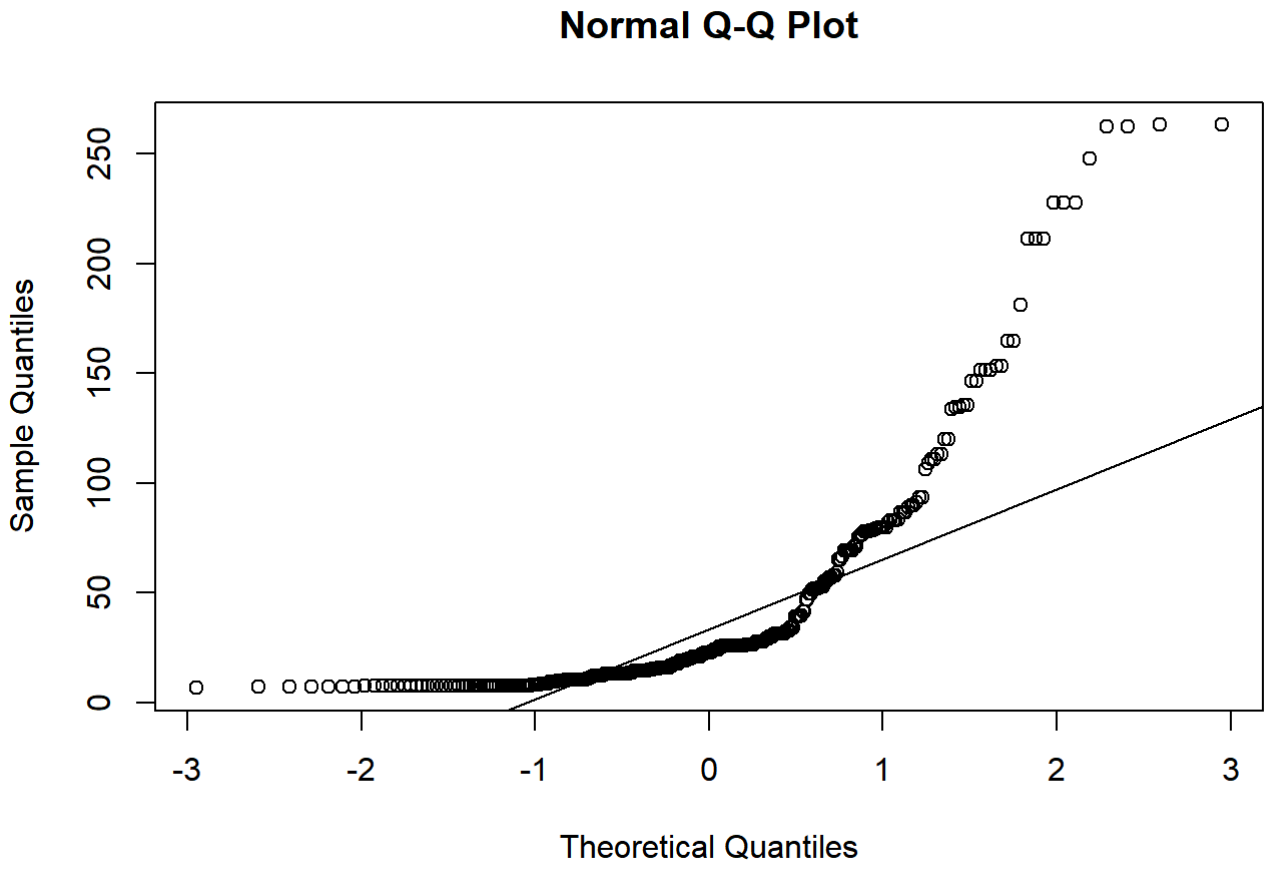
```
## Variables que no siguen una distribución normal:
## Age, SibSp, Parch,
## Fare
```

Se va a comprobar que los datos de las variables anteriores siguen una distribución normal:

```
#Se calcula el Test de Shapiro-Wilk para la muestra de mujer
shapiro.test(precio.mujer$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  precio.mujer$Fare
## W = 0.68775, p-value < 2.2e-16
```

```
#Se calcula el gráfico QQ para la muestra de mujer
qqnorm(precio.mujer$Fare)
qqline(precio.mujer$Fare)
```

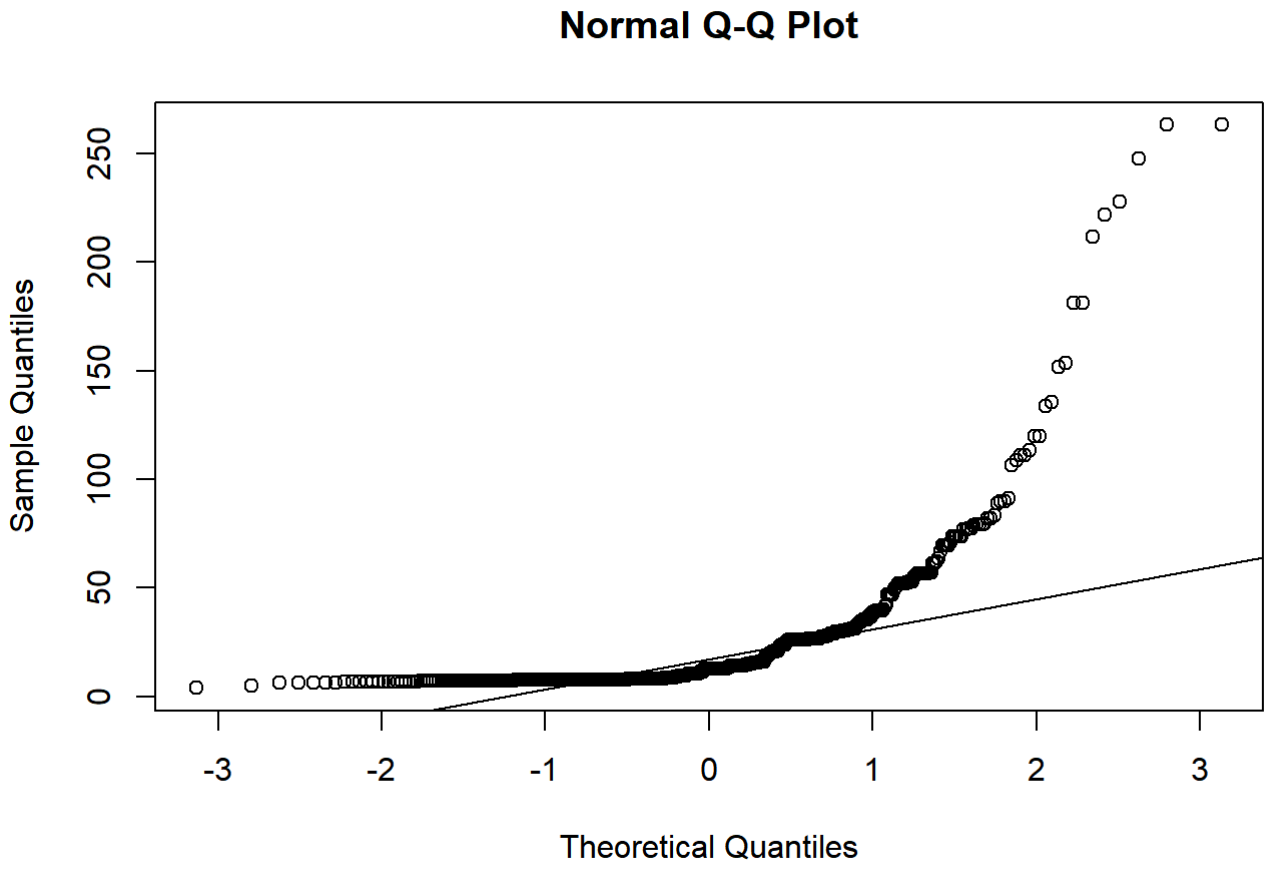



```
#Se comprueba que no sigue una distribución normal como se comprobó anteriormente

#Se calcula el Test de Shapiro-Wilk para la muestra de hombre
shapiro.test(precio.hombre$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  precio.hombre$Fare
## W = 0.54863, p-value < 2.2e-16
```

```
#Se calcula el gráfico QQ para la muestra de hombre
qqnorm(precio.hombre$Fare)
qqline(precio.hombre$Fare)
```

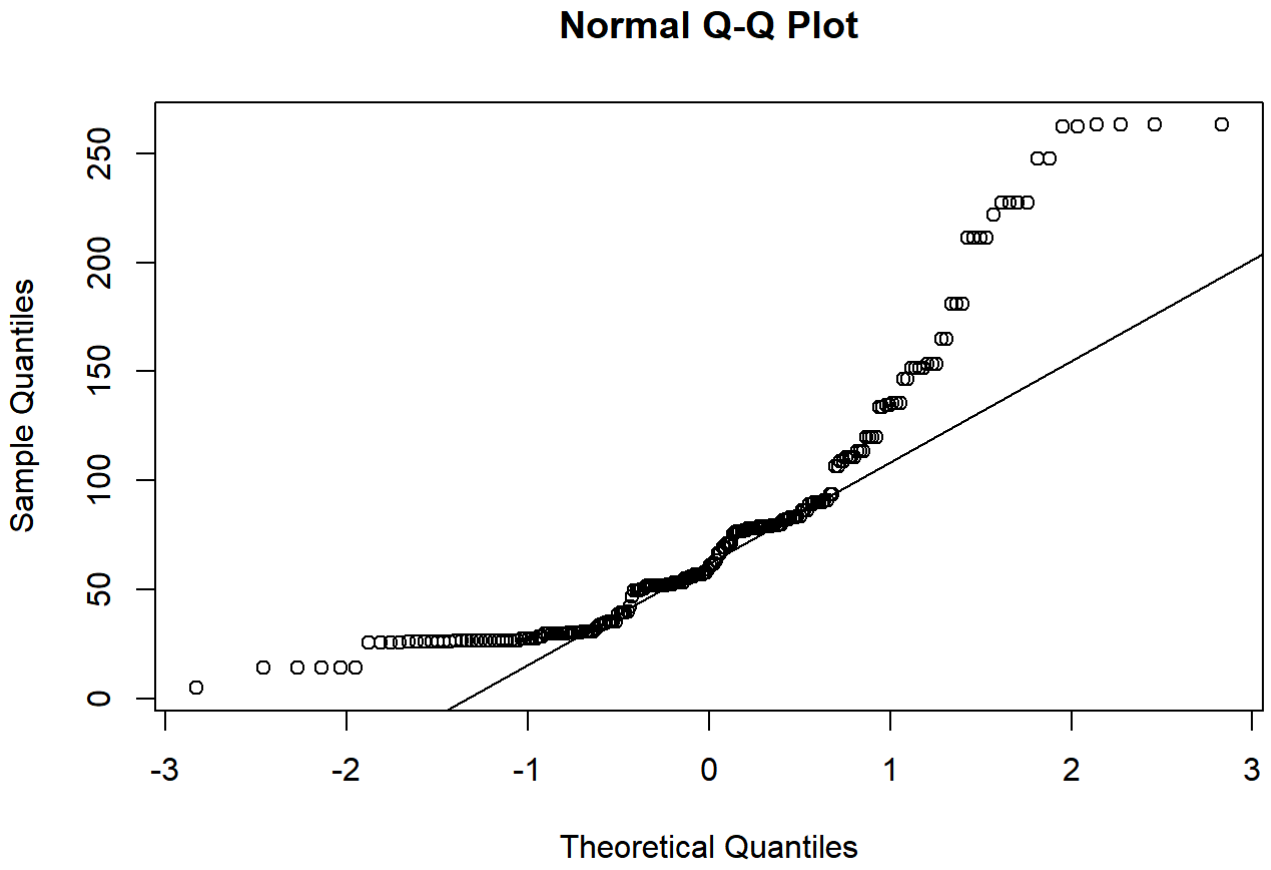



```
#Se comprueba que no sigue una distribución normal como se comprobó anteriormente

#Se calcula el Test de Shapiro-Wilk para la muestra de primera clase
shapiro.test(precio.primera$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  precio.primera$Fare
## W = 0.82901, p-value = 1.116e-14
```

```
#Se calcula el gráfico QQ para la muestra de primera clase
qqnorm(precio.primera$Fare)
qqline(precio.primera$Fare)
```

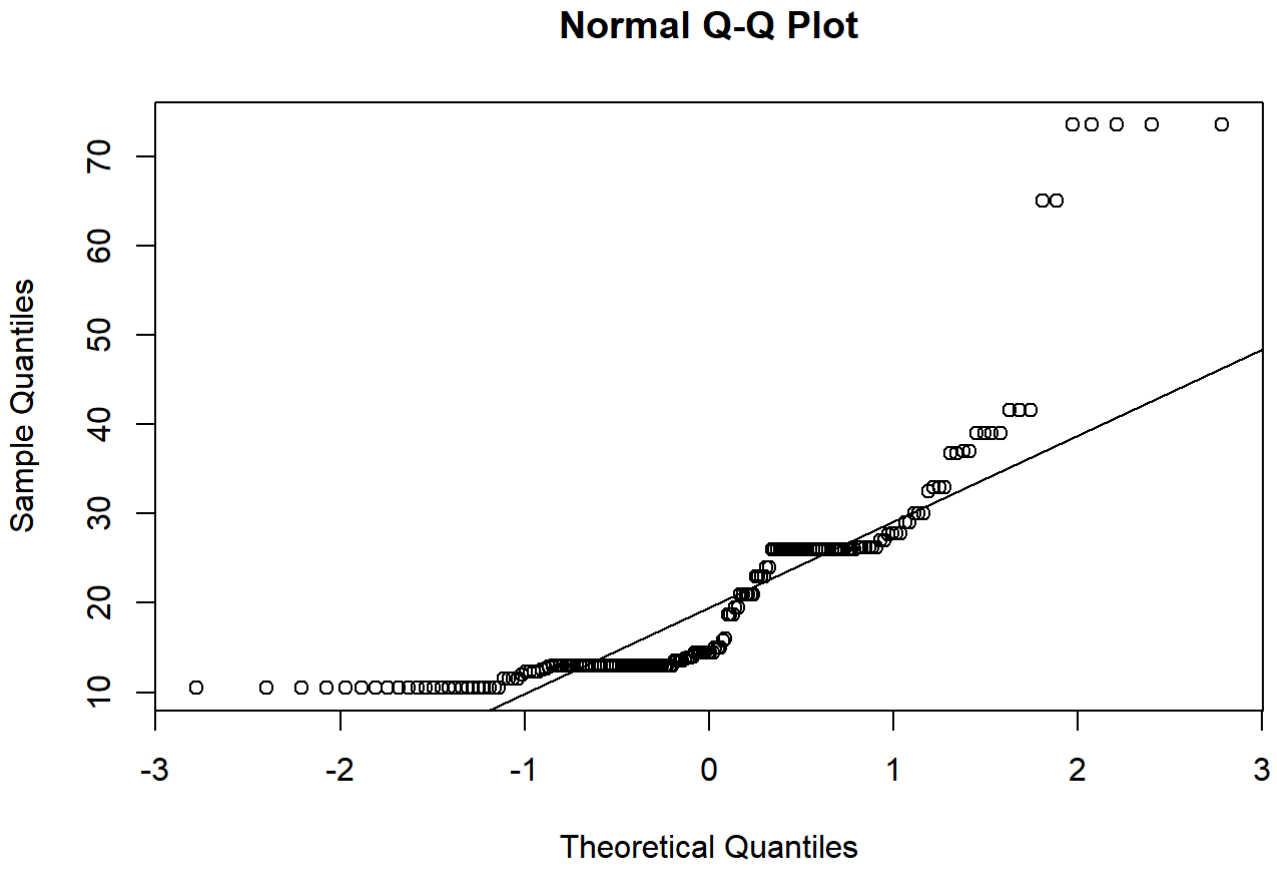



```
#Se comprueba que no sigue una distribución normal como se comprobó anteriormente

#Se calcula el Test de Shapiro-Wilk para la muestra de segunda clase
shapiro.test(precio.segunda$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  precio.segunda$Fare
## W = 0.72251, p-value < 2.2e-16
```

```
#Se calcula el gráfico QQ para la muestra de segunda clase
qqnorm(precio.segunda$Fare)
qqline(precio.segunda$Fare)
```

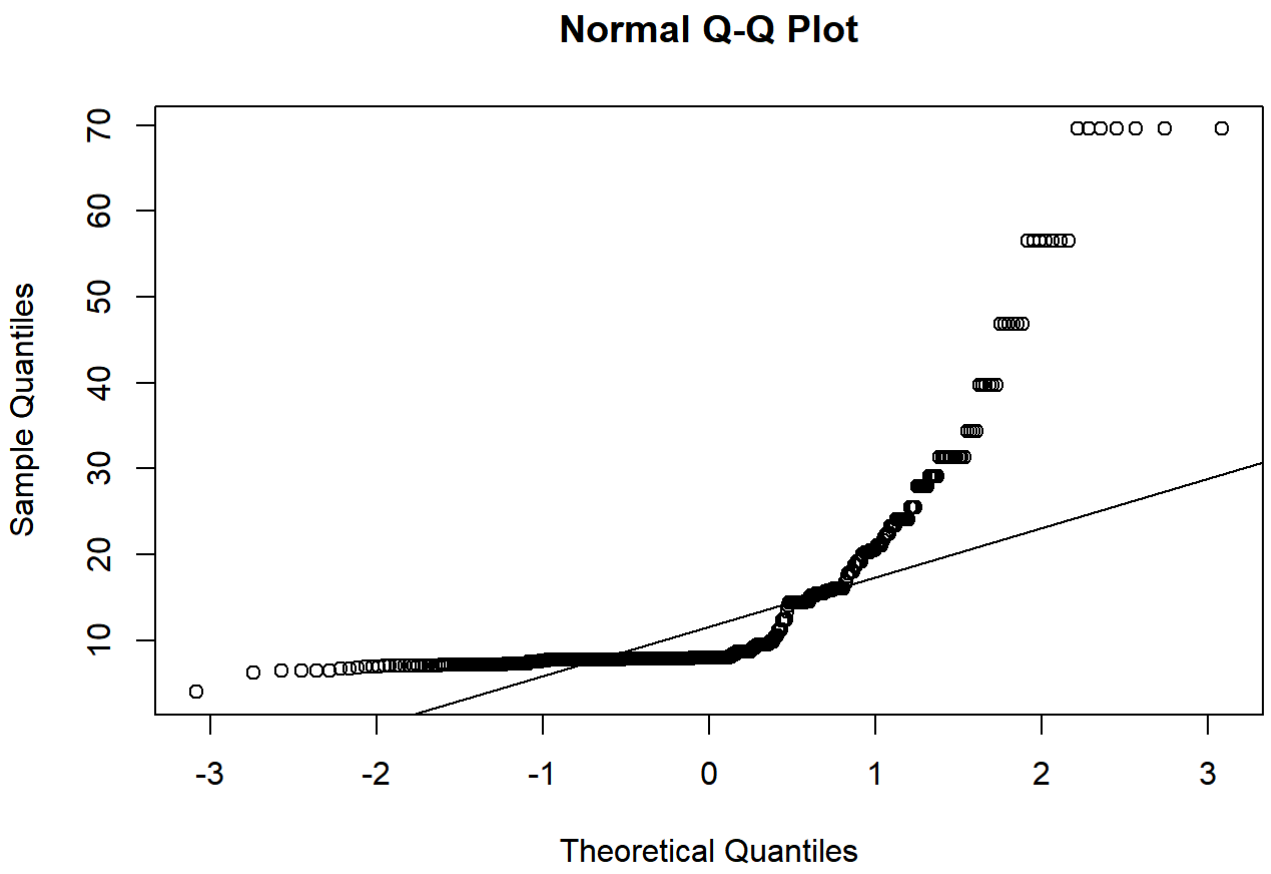



```
#Se comprueba que no sigue una distribución normal como se comprobó anteriormente

#Se calcula el Test de Shapiro-Wilk para la muestra de tercera clase
shapiro.test(precio.tercera$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  precio.tercera$Fare
## W = 0.60699, p-value < 2.2e-16
```

```
#Se calcula el gráfico QQ para la muestra de tercera clase
qqnorm(precio.tercera$Fare)
qqline(precio.tercera$Fare)
```

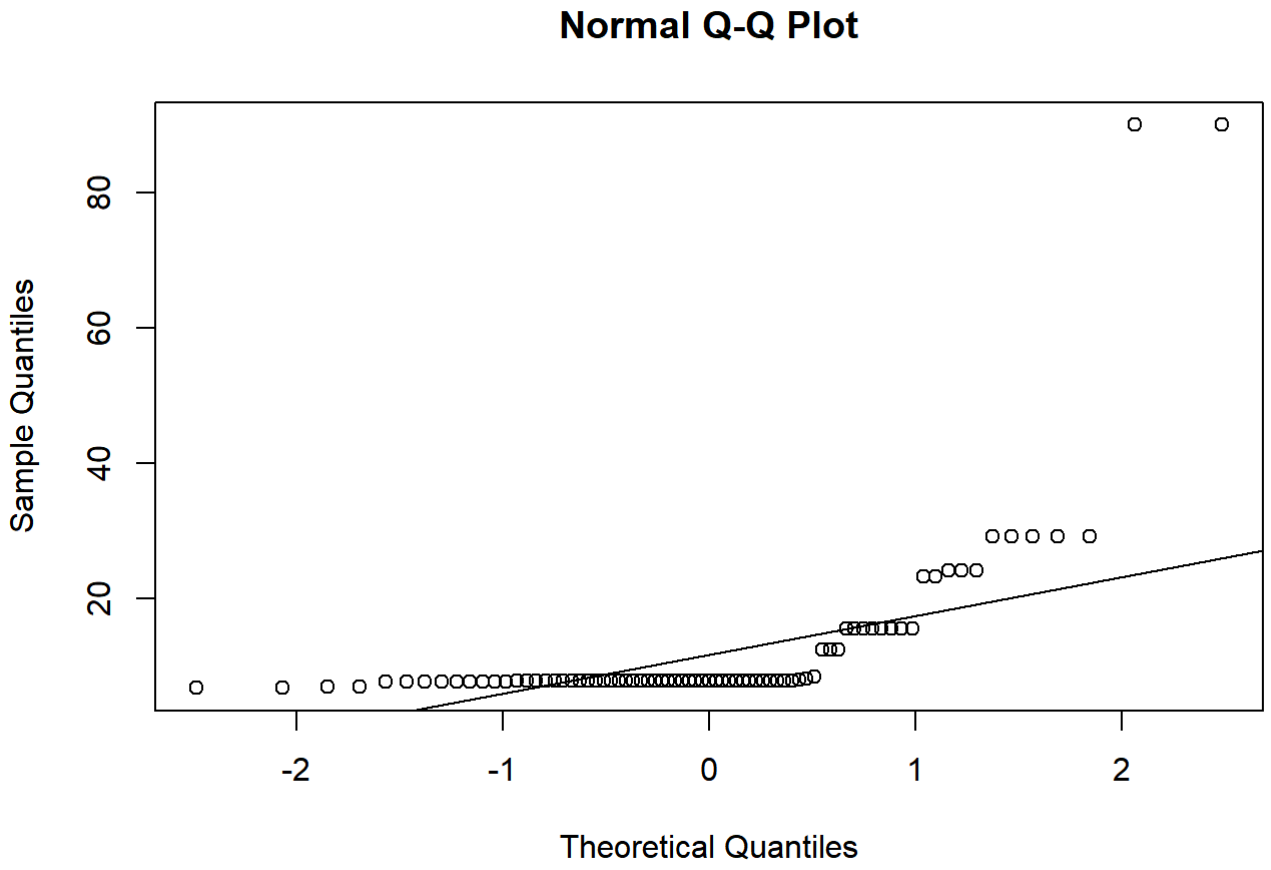



```
#Se comprueba que no sigue una distribución normal como se comprobó anteriormente

#Se calcula el Test de Shapiro-Wilk para la muestra del puerto Queenstrown
shapiro.test(precio.queenstrown$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  precio.queenstrown$Fare
## W = 0.43638, p-value = 1.151e-15
```

```
#Se calcula el gráfico QQ para la muestra del puerto Queenstrown
qqnorm(precio.queenstrown$Fare)
qqline(precio.queenstrown$Fare)
```

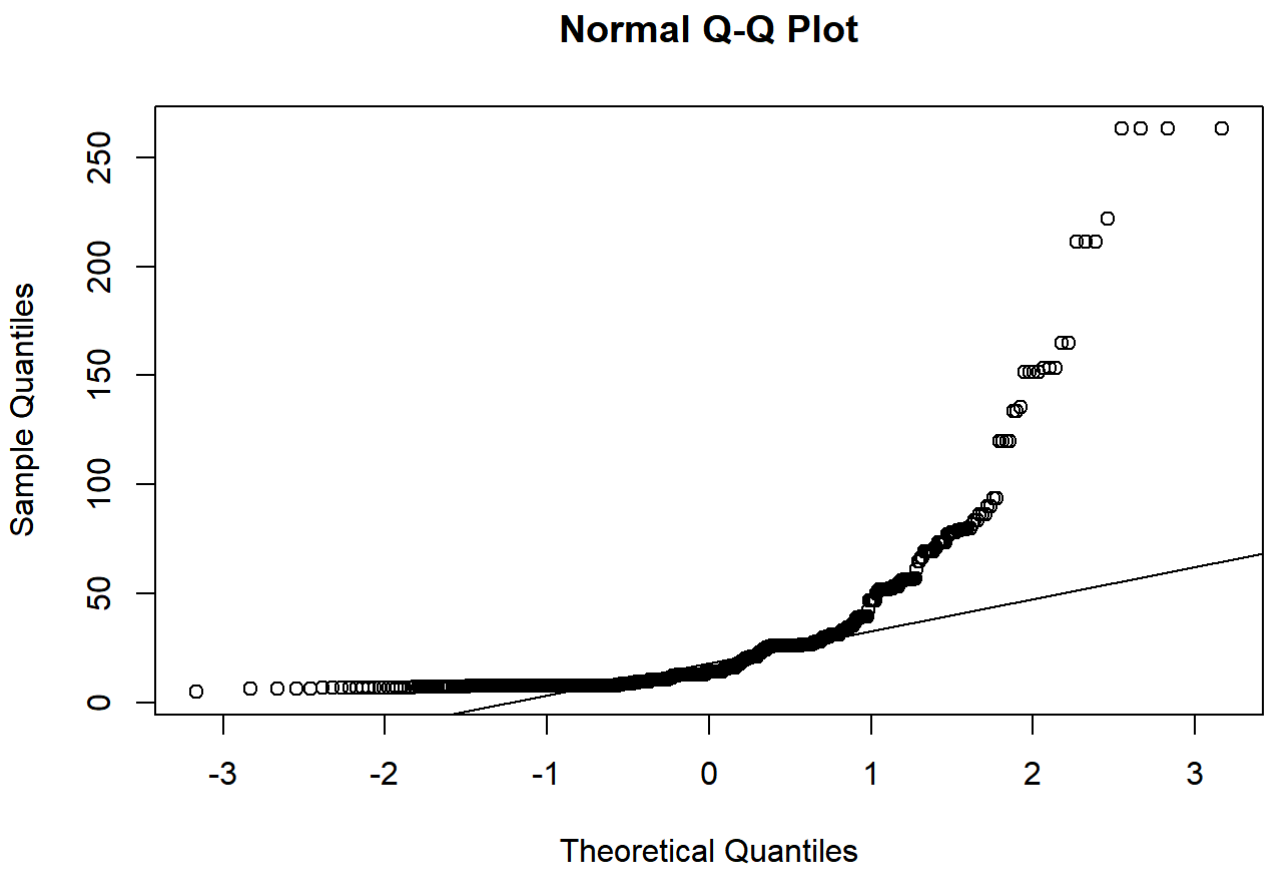



```
#Se comprueba que no sigue una distribución normal como se comprobó anteriormente

#Se calcula el Test de Shapiro-Wilk para la muestra del puerto Southampton
shapiro.test(precio.southampton$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  precio.southampton$Fare
## W = 0.57375, p-value < 2.2e-16
```

```
#Se calcula el gráfico QQ para la muestra del puerto Southampton
qqnorm(precio.southampton$Fare)
qqline(precio.southampton$Fare)
```

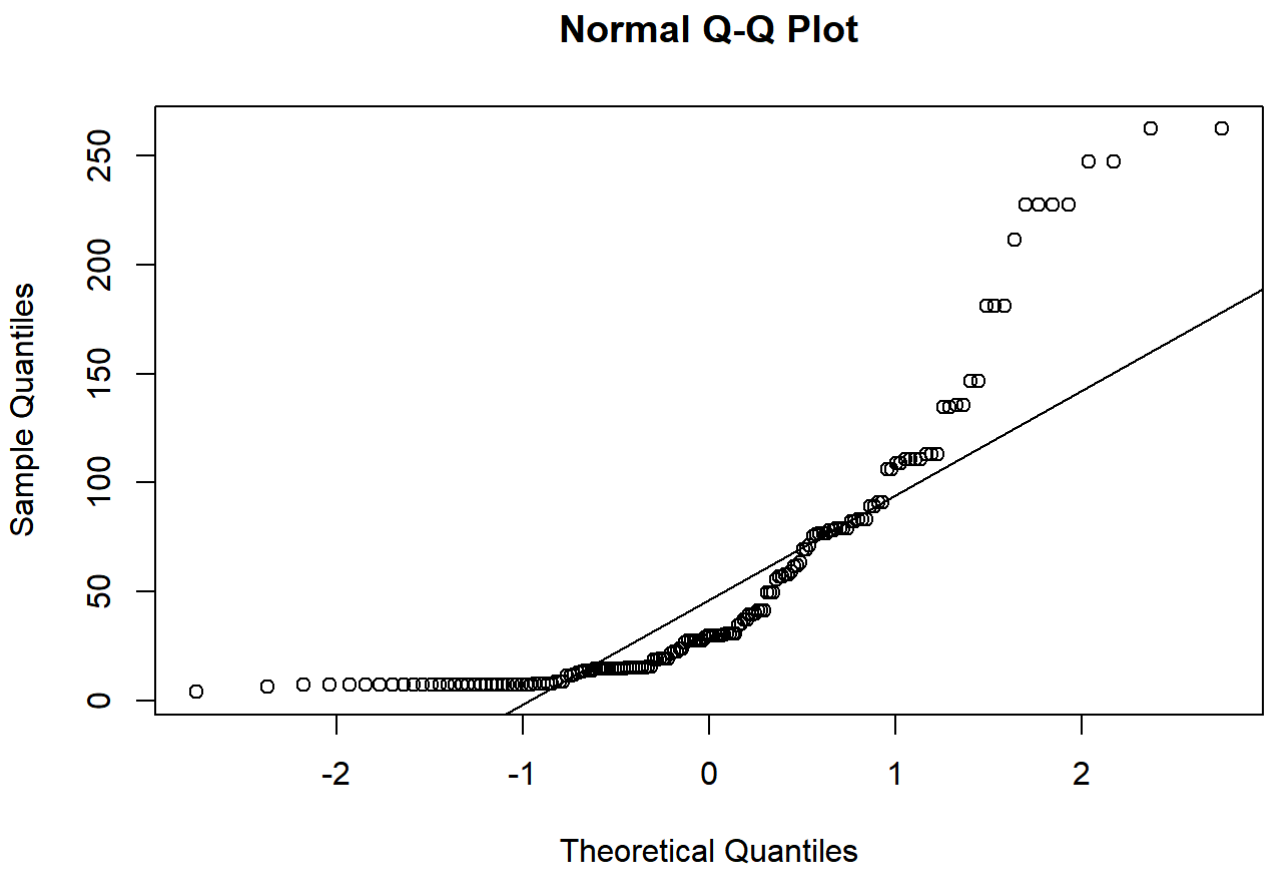



```
#Se comprueba que no sigue una distribución normal como se comprobó anteriormente

#Se calcula el Test de Shapiro-Wilk para la muestra del puerto Cherbourg
shapiro.test(precio.cherbourg$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  precio.cherbourg$Fare
## W = 0.76286, p-value = 3.413e-15
```

```
#Se calcula el gráfico QQ para la muestra del puerto Cherbourg
qqnorm(precio.cherbourg$Fare)
qqline(precio.cherbourg$Fare)
```




```
#Se comprueba que no sigue una distribución normal como se comprobó anteriormente
```

Se va a comprobar la homogeneidad de las varianzas con el Test de Fligner-Killen.

```
#Se comprueba las varianzas para el precio y la edad
fligner.test(Fare~Age, data=base2)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Fare by Age
## Fligner-Killeen:med chi-squared = 147.43, df = 87, p-value = 5.634e-05
```

```
#No se acepta la hipótesis nula (igualdad de varianzas) porque el p-value es menor que alpha
 (1-0.95)
```

4.3 Aplicación de pruebas estadísticas

Variables que influyen en el precio y su proporción

Para ello se va a realizar un análisis de correlación entre las distintas variables para determinar las más influyentes. Se procede a calcuar el factor de correlación con el Test de Spearman y su p-valor para comprobar que cumple el nivel de confianza.

```
corr_matrix <- matrix(nc = 2, nr = 0)
colnames(corr_matrix) <- c("estimado", "p-valor")
#Se van a seleccionar las variables con valores numéricos
borrar2 <- c("Survived","Pclass","Sex","Embarked")
base3 <- base2[ , !(names(base2) %in% borrar2)]
# Calcular el coeficiente de correlación para cada variable cuantitativa con respecto al camp
o "precio"
for (i in 1:(ncol(base3) - 1)) {
if (is.integer(base3[,i]) | is.numeric(base3[,i])) {
spearman_test = cor.test(base3[,i],base3[,length(base3)], method = "spearman")
corr_coef = spearman_test$estimate
p_val = spearman_test$p.value
# Añadimos las filas a la matriz
pair = matrix(ncol = 2, nrow = 1)
pair[1][1] = corr_coef
pair[2][1] = p_val
corr_matrix <- rbind(corr_matrix, pair)
rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(base3)[i]
corr_matrix
}
}
```

```
## Warning in cor.test.default(base3[, i], base3[, length(base3)], method =
## "spearman"): Cannot compute exact p-value with ties

## Warning in cor.test.default(base3[, i], base3[, length(base3)], method =
## "spearman"): Cannot compute exact p-value with ties

## Warning in cor.test.default(base3[, i], base3[, length(base3)], method =
## "spearman"): Cannot compute exact p-value with ties
```

```
#Se procede a imprimir los resultados
print(corr_matrix)
```

```
##        estimado      p-valor
## Age   0.1353788 5.034672e-05
## SibSp 0.4427160 4.636445e-44
## Parch 0.4071481 6.711774e-37
```

Se comprueba que la variable que más influye en el precio es Parch frente a Age y SibSp con p-valor menos que alfa (1-0.95) estando dentro del nivel de confianza establecido.

¿El precio del billete fue superior en caso de ser mujer o hombre?

Se va a realizar un contraste de hipótesis sobre dos muestras para determinar si el precio del billete depende si es mujer o hombre. Se establecen dos variables con los precios si es mujer o hombre. Para poder realizar un análisis de hipótesis se supone que las muestras siguen una distribución normal (n>30 en ambas variables):

```
precio.mujer.precio <- precio.mujer$Fare
precio.hombre.precio <- precio.hombre$Fare
```

Así, se plantea el siguiente contraste de hipótesis de dos muestras sobre la diferencia de medias, el cual es unilateral atendiendo a la formulación de la hipótesis alternativa: o H0 : μ1 − μ2 = 0 o H1 : μ1 − μ2 < 0 donde μ1 es la media de la población de la que se extrae la primera muestra y μ2 es la media de la población de la que extrae la segunda. Así, tomaremos α = 0, 05.

```
t.test(precio.mujer.precio,precio.hombre.precio, alternative = "less")
```

```
##
##  Welch Two Sample t-test
##
## data:  precio.mujer.precio and precio.hombre.precio
## t = 5.7377, df = 454.82, p-value = 1
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##      -Inf 24.03752
## sample estimates:
## mean of x mean of y
##  43.42513  24.75175
```

Como el valor de p-valor (1) es mayor que el valor de significación fijado (0.05), se da por válida la hipótesis nula. Lo que nos indica que el valor del precio del billete no influye si es mujer o hombre.

####Modelo de regresión lineal para el precio

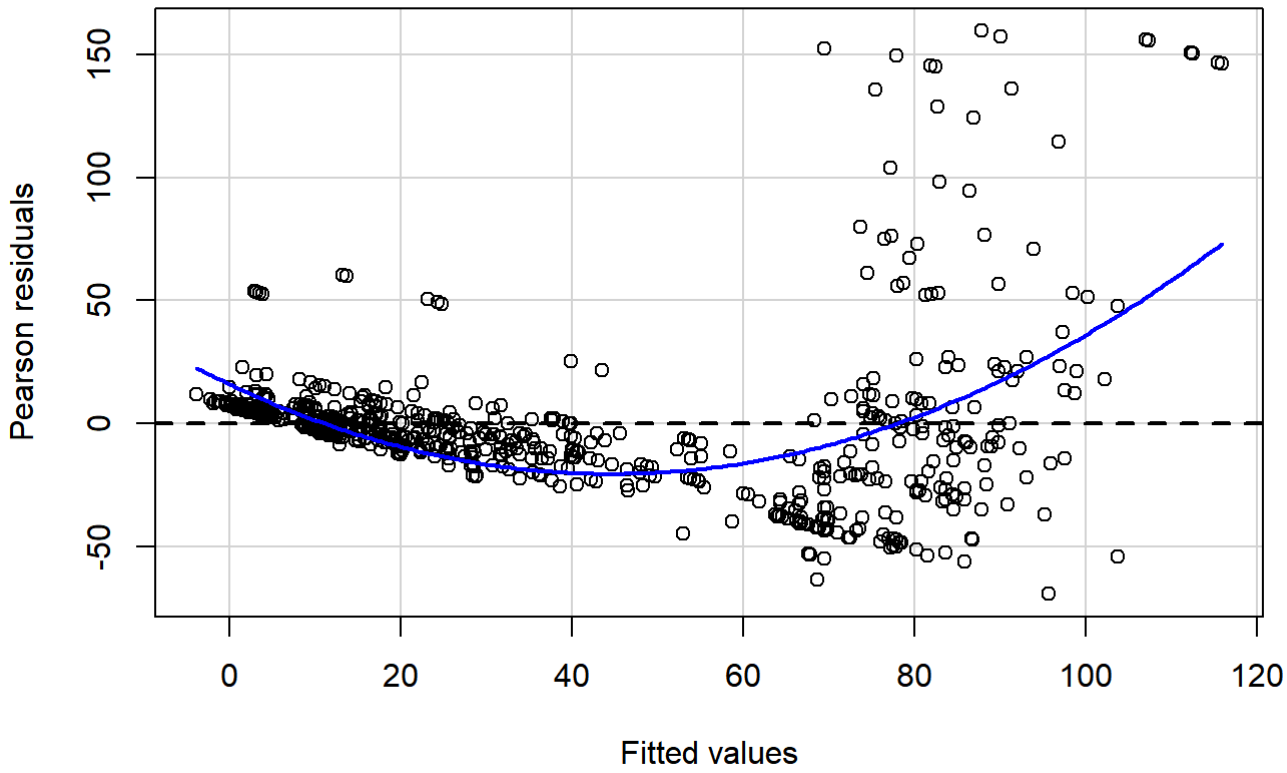
Se calcula la variable fare (precio) como explicada y las variables Survived, Pclass, Sex, Age, SibSp, Parch y Embarked como explicativas. Obteniendo:

```
logit_model_1 <- lm(formula=Fare~factor(Survived)+factor(Pclass)+factor(Sex)+Age+SibSp+Parch+
factor(Embarked), data=base2)
summary(logit_model_1)
```

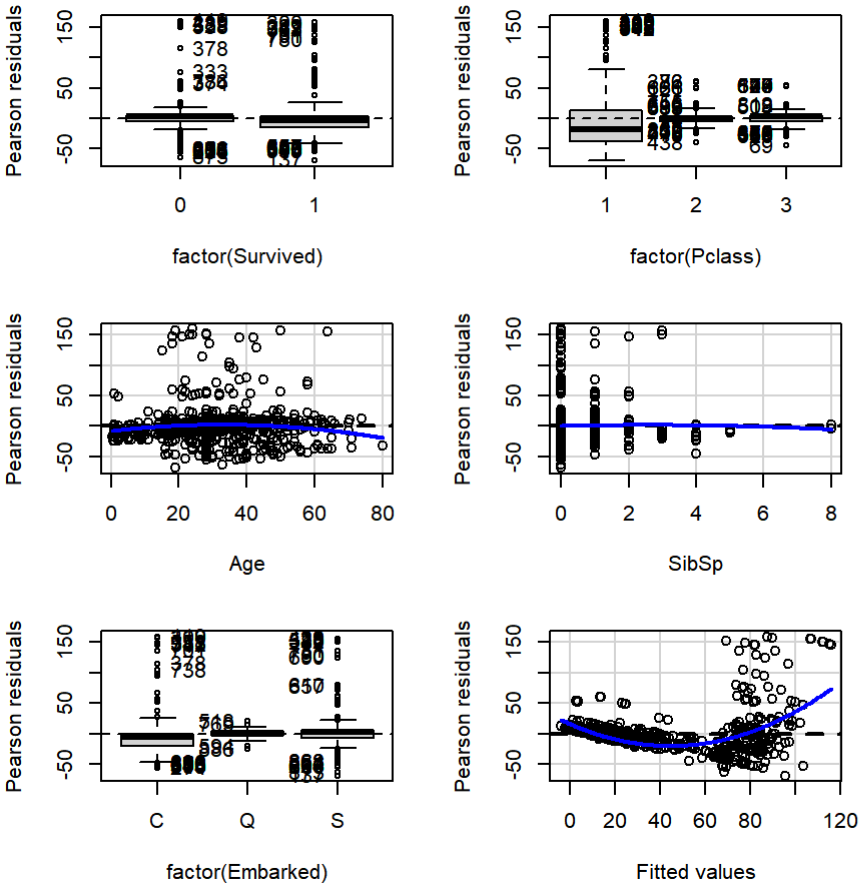
```
##
## Call:
## lm(formula = Fare ~ factor(Survived) + factor(Pclass) + factor(Sex) +
##     Age + SibSp + Parch + factor(Embarked), data = base2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -69.423  -9.232   0.015   4.883 159.665
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         87.89225    5.01541  17.524  < 2e-16 ***
## factor(Survived)1    0.35172    2.53138   0.139  0.88953
## factor(Pclass)2    -57.25792    3.06090 -18.706  < 2e-16 ***
## factor(Pclass)3    -66.27064    2.80628 -23.615  < 2e-16 ***
## factor(Sex)male     -5.77190    2.47966  -2.328  0.02015 *
## Age                 -0.15275    0.08247  -1.852  0.06433 .
## SibSp                5.79461    0.97809   5.924 4.49e-09 ***
## Parch                9.40180    1.34746   6.977 5.91e-12 ***
## factor(Embarked)Q   -7.27062    4.14613  -1.754  0.07985 .
## factor(Embarked)S   -8.43866    2.60489  -3.240  0.00124 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 28.53 on 881 degrees of freedom
## Multiple R-squared:  0.5408, Adjusted R-squared:  0.5361
## F-statistic: 115.3 on 9 and 881 DF,  p-value: < 2.2e-16
```

Se comprueba que las variables Survived, Pclass, SibSp y Parch tienen un nivel de significación cercano al cero y con un nivel de significación del 1% la variable Embarked. El coeficiente de determinación ajustado de 0,5361 de este modelo por lo cual se puede decir que no se ajusta a la perfección a una recta. El p-value del modelo es significativo (2.2e-16) por lo que se puede aceptar que el modelo no es por azar, al menos uno de los coeficientes parciales de regresión es distinto de 0.

```
#Se calculan los errores del modelo
residuosmult.cual <- residuals(logit_model_1)
#Se calculan los valores ajustados al modelo
valores.ajustados.mult.cual <-fitted(logit_model_1)
#Se representan los valores estimados frente a los errores
residualPlot(logit_model_1)
```




```
#Se representan los valores de cada variable frente a los errores
residualPlots(logit_model_1)
```



```
##                  Test stat Pr(>|Test stat|)
## factor(Survived)
## factor(Pclass)
## factor(Sex)
## Age                -2.9642         0.003117 **
## SibSp              -0.7993         0.424310
## Parch              -1.8089         0.070806 .
## factor(Embarked)
## Tukey test         17.9338        < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Se comprueba que los errores están cercanos al eje de abscisas de los valores estimados (Cercanos a cero) y no sigue una distribución conocida. También se comprueba que los valores de los errores frente a las diferentes variables no siguen un modelo muy definido como se puede comprobar en los diferentes gráficos.

Se procede a predecir el valor de precio con los siguientes valores: o Age = 55 o Survived = 1 o Pclass = 3 o Sex = “male” o SibSp = 0 o Parch = 0 o Embarked = “C”

```
#Se va a predecir el precio por unos valores determinados
nuevos.valores <- data.frame(Age = 55, Survived = 1, Pclass = 3, Sex = "male", SibSp = 0, Par
ch = 0, Embarked = "C")
#Calculamos el valor de precio
precioest <-  predict(logit_model_1, nuevos.valores)
precioest
```

```
##        1
## 7.800433
```

Modelo de regresión logística

Se calcula una regresión lineal de la variable Survived como explicada y las variables Fare, Pclass, Sex, Age, SibSp, Parch y Embarked como explicativas. Obteniendo:

```
logit_model_2 <- glm(formula=Survived~Fare+factor(Pclass)+factor(Sex)+Age+SibSp+Parch+factor
(Embarked), data=base2, family = binomial(link = 'logit'))
summary(logit_model_2)
```

```
##
## Call:
## glm(formula = Survived ~ Fare + factor(Pclass) + factor(Sex) +
##     Age + SibSp + Parch + factor(Embarked), family = binomial(link = "logit"),
##     data = base2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5852  -0.6070  -0.4145   0.6212   2.4594
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        4.2202774  0.4969709   8.492  < 2e-16 ***
## Fare               0.0005802  0.0031858   0.182  0.85549
## factor(Pclass)2   -1.0117026  0.3116950  -3.246  0.00117 **
## factor(Pclass)3   -2.2638986  0.3209783  -7.053 1.75e-12 ***
## factor(Sex)male   -2.7230968  0.2009880 -13.549  < 2e-16 ***
## Age               -0.0388469  0.0078615  -4.941 7.76e-07 ***
## SibSp             -0.3118112  0.1099087  -2.837  0.00455 **
## Parch             -0.0743435  0.1201445  -0.619  0.53606
## factor(Embarked)Q -0.0762307  0.3811902  -0.200  0.84150
## factor(Embarked)S -0.4644293  0.2383889  -1.948  0.05139 .
## ---
## 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:  785.98  on 881  degrees of freedom
## AIC: 805.98
##
## Number of Fisher Scoring iterations: 5
```

Se procede a predecir la posibilidad de sobrevivir con los siguientes valores: o Age = 55 o Fare = 7 o Pclass = 3 o Sex = “male” o SibSp = 0 o Parch = 0 o Embarked = “C”

```
#Se va a predecir el precio por unos valores determinados
nuevos.valores2 <- data.frame(Age = 55, Fare = 7, Pclass = 3, Sex = "male", SibSp = 0, Parch
 = 0, Embarked = "C")
#Calculamos la probabilidad de sobrevivir
superv <-  round(predict(logit_model_2, nuevos.valores2, type = "response"))
superv
```

```
## 1
## 0
```

Se comprueba que no se va a sobrevivir.

5.Representación de los resultados a partir de tablas y gráficas.

Se va a proceder a tabular los valores siguientes: • precio.mujer • precio.hombre • precio.primera • precio.segunda • precio.tercera • precio.cherbourg • precio.queenstrown • precio.southampton

Tabla de valores Precio y Edad por Sexo.

```
ttDescri <- tabular( Sex ~ (Fare + Age) *( mean + sd + min + max ) + ( n = 1 ), data = base2
 )
ttDescri
```

	Fare				Age				
Sex	**mean**	**sd**	**min**	**max**	**mean**	**sd**	**min**	**max**	**n**
female	43.43	52.18	6.750	263	27.93	12.86	0.7563		314
male	24.75	33.28	4.013	263	30.14	13.05	0.4280		577

Tabla de valores Precio y Edad por Clase.

```
ttDescri2 <- tabular( Pclass ~ (Fare + Age) *( mean + sd + min + max ) + ( n = 1 ), data = ba
se2 )
ttDescri2
```

	Fare				Age				
Pclass	**mean**	**sd**	**min**	**max**	**mean**	**sd**	**min**	**max**	**n**
1	79.89	60.39	5.000	263.00	36.81	14.18	0.9280		216
2	21.13	12.93	10.500	73.50	29.77	13.58	0.6770		184
3	13.79	11.71	4.013	69.55	25.93	10.70	0.4274		491

Tabla de valores Precio y Edad por Puerto

```
ttDescri3 <- tabular( Embarked ~ (Fare + Age) *( mean + sd + min + max ) + ( n = 1 ), data =
 base2 )
ttDescri3
```

	Fare				Age				
Embarked	**mean**	**sd**	**min**	**max**	**mean**	**sd**	**min**	**max**	**n**
C	54.04	59.95	4.013	262.4	30.18	13.62	0.4271.0		168
Q	13.28	14.19	6.750	90.0	28.03	10.08	2.0070.5		77
S	27.58	35.76	5.000	263.0	29.31	13.17	0.6780.0		646

Procedemos a hacer los gráficos:

Gráfico de Edad frente Precio

```
ggplot(data = base2, aes( x = Age, y = Fare )) + geom_point( aes( colour = Sex ), size = 2) +
labs( title = "Precio según Edad", x = "Edad", y = "Precio")
```

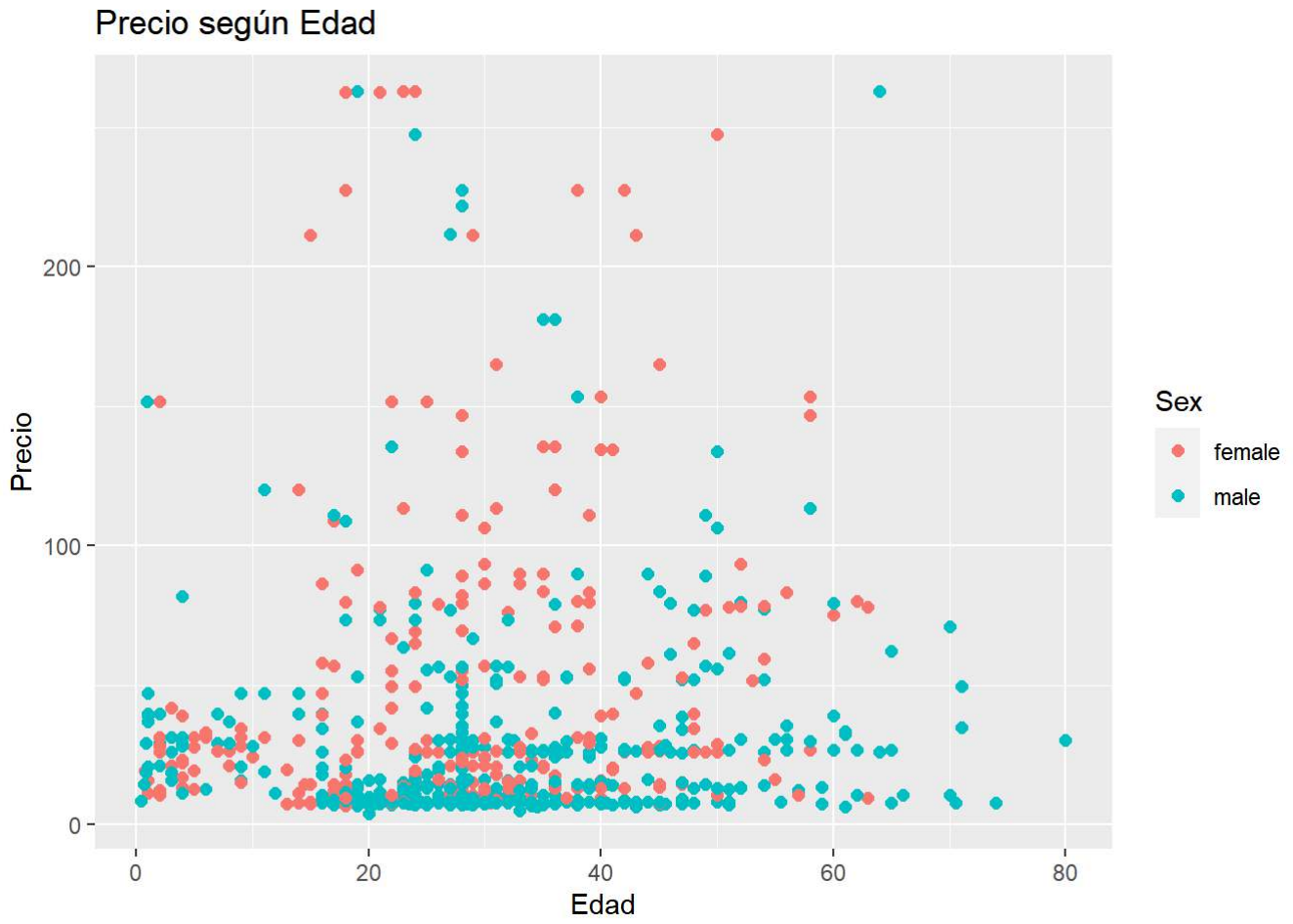


Gráfico de pasajeros por clase.

```
tablashelve <- table(base2$Pclass )
tablashelve <- prop.table(tablashelve)
barplot(tablashelve, main="Gráfico de barras de los pasajeros por clases ", ylab="Frecuencia"
, col="blue")
```

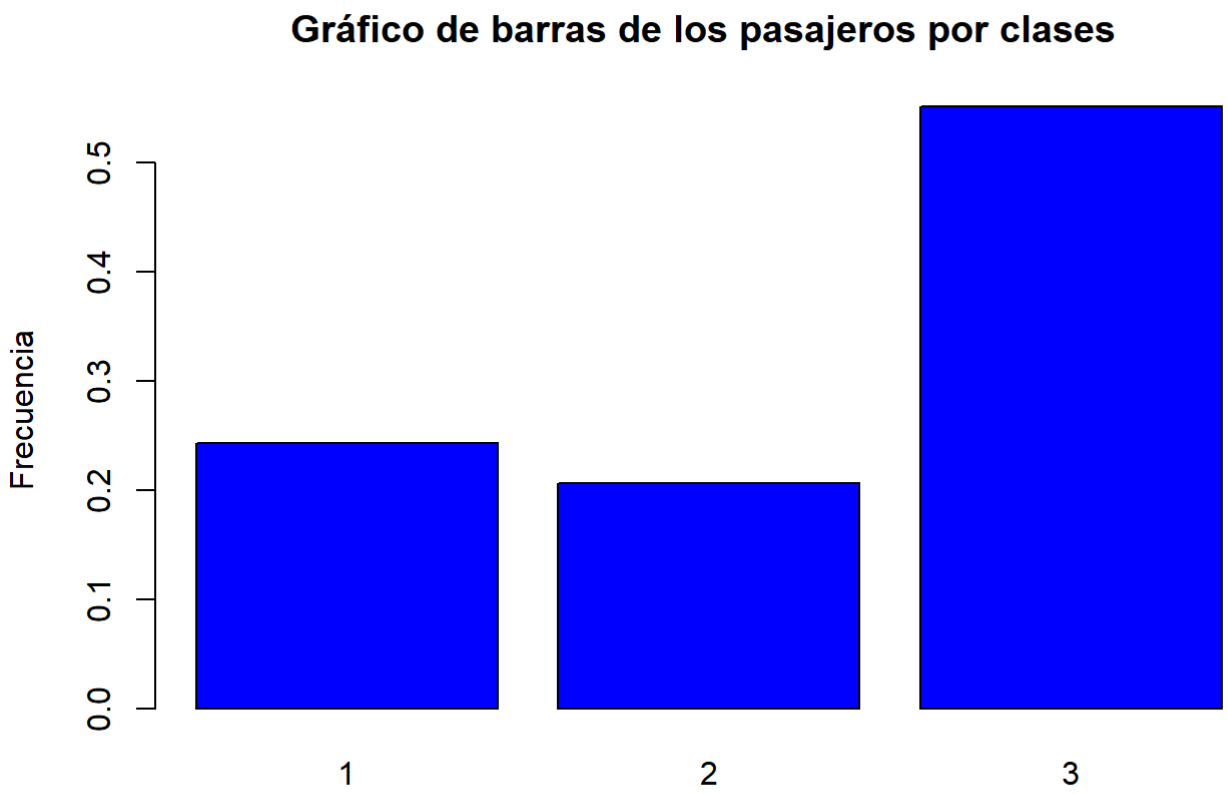


Gráfico de pasajeros que murieron (0) o sobrevivieron (1)

```
tablashelve2 <- table(base2$Survived)
tablashelve2
```

```
##
##   0   1
## 549 342
```

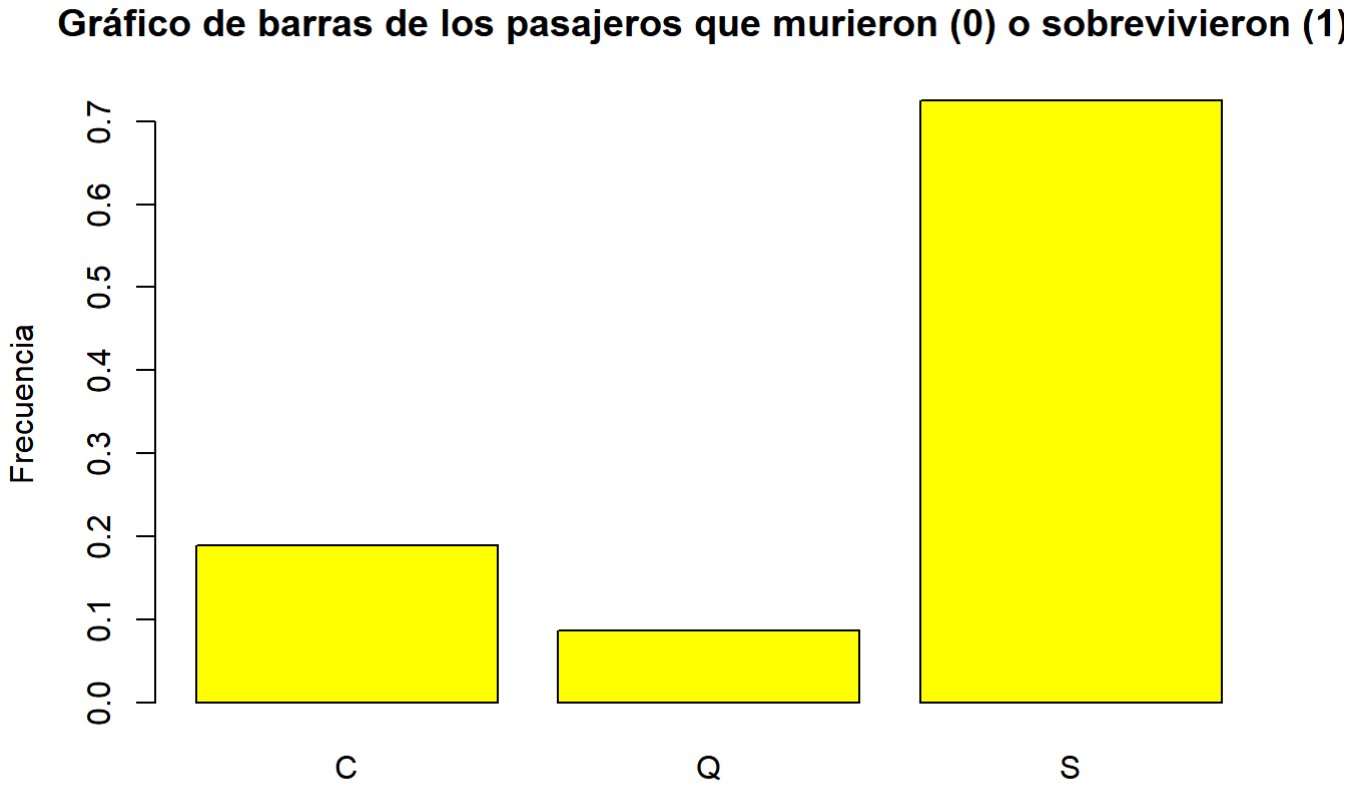
```
tablashelve2<- prop.table(tablashelve2)
barplot(tablashelve2, main="Gráfico de barras de los pasajeros que murieron (0) o sobrevivier
on (1) ", ylab="Frecuencia", col="red")
```

Gráfico de pasajeros por puerto de embarque

```
tablashelve3 <- table(base2$Embarked)
tablashelve3
```

```
##
##   C   Q   S
## 168  77 646
```

```
tablashelve3<- prop.table(tablashelve3)
barplot(tablashelve3, main="Gráfico de barras de los pasajeros que murieron (0) o sobrevivier
on (1) ", ylab="Frecuencia", col="yellow")
```



6 Resolución del problema.

A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

Se puede estimar el precio no depende del sexo (según el estudio de contraste de hipótesis realizados) y que las varianzas de edad y precio no son iguales por el otro estudio realizado. Se llega a obtener una regresión lineal con un coeficiente de fiabilidad del 95% y con un ajuste cercano al 0,5; pudiéndose considerar para estimar valores de precio. Las variables que influyen en el precio son: • Survived. No tengo explicación • Pclass. La clase influye en el precio del billete • SibSp y Parch. Se hacen descuentos cuando se compran billetes en conjunto por lo cual si se tienen familiares embarcados se supone que se han comprado en conjunto los tiques. • Embarked. El puerto de embarque influye en el precio porque cuanto más trayecto aumenta el precio.

También se consigue estimar si un pasajero iba a sobrevivir por las variables evaluadas, siendo las de mayor importancia: • Pclass. Las clases mejores tienen camarotes y zonas comunes sobre la floración siendo más fácil poder escapar del buque. Además, en aquella época tenían derecho a los botes de rescate solo la primera clase. • Age. Las personas jóvenes tienen más resistencia la frio como al ejercicio para mantenerse a flote. • SibSp. Tener familiares a bordo influye porque se intenta buscar y salvar a los familiares desviando esfuerzos para ello.

Los resultados permiten resolver el problema planteado.