Práctica 2: Tipología y ciclo de vida de los datos

Joshelyn Intriago - David Morocho

5/11/2021

Contents

1 Descripción del dataset.

El dataset se ha obtenido de kaggle y describe el estado de supervivencia de pasajeros individuales a bordo del Titanic, el trasatlántico de pasajeros más grande construido que chocó con un iceberg en su viaje inaugural. CUando se hundió mató 1502 personas de 2224 pasajeros y tripulación.

El dataset está dividido en 2 partes, un dataset para de entrenamiento con 11 variables y 891 registros, y un dataset de test o evaluación con 11 variables y 418 registros. Las variables del dataset son:

- **Pclass** Clase del pasajero (1 = primera; 2 = segunda; 3 = tercera)
- **Survival** Indica si sobrevivió (0 = No; 1 = Si)
- **Name** Nombre
- **Sex** Sexo
- **Age** Edad
- **sibsp** Numero de Hermanos/Esposas o Esposos a bordo
- **Parch** Numero de Padres/Hijos a bordo
- **Ticket** número de ticket
- **Fare** costo del ticket abordo en libras británicas
- **Cabin** Camarote del pasajeroo
- **Embarked** Puerto de embarcación (C = Cherbourg; Q = Queenstown; S = Southampton)

1.1 Importancia y objetivos del análisis

El dataset del Titanic cuenta con 11 variables que expresan el contexto social y económico de los pasajeros a bordo del Titanic además de si sobrevivieron al incidente. Estas características del dataset sumado a que representan uno de los eventos históricos más relevantes del siglo XX, propician se continúe estudiando y analizando el incidente con el afán de generar modelos de predicción de la supervivencia de los pasajeros.

El objetivo del análisis del dataset en este trabajo es el de determinar que grupos de personas son más probables de sobrevivir tomando en cuenta datos cómo la edad, el género, la clase socioeconómica, el número de familiares en a bordo, entre otras variables presentes.

2 Integración y selección de los datos de interés a analizar

El dataset está dividido en un conjunto de entrenamiento (*train.csv*), un conjunto de validación (*test.csv*) y un archivo (*gender_submission.csv*) complemento del conjunto de validación que contiene la variable a predecir, survived. Estos conjuntos de datos se juntan para realizar las tareas de limpieza y selección de variables de interés a analizar.

Leemos cada uno de los archivos y en la variable nueva **tipo** se indica a que conjunto de datos pertenece cada registro para luego del proceso de limpieza se pueda dividir el dataset en los conjuntos de entrenamiento y validación iniciales.

Cargar librerías:

```
if (!require("ggplot2")) install.packages("ggplot2")
library("ggplot2")
if (!require("formatR")) install.packages("formatR")
library("formatR")
if (!require("arules")) install.packages("arules")
library(arules)
library(dplyr)
library(tidyr)
```

```
library(corrplot)
library(magrittr)
library(caret)
library(tibble)
library(pROC)
library(rpart)
library(party)
library(randomForest)
library(e1071)
library(gbm)
library(ROCR)
library(C50)
```

Cargar datos:

```
# Se lee el archivo de entrenamiento
train <- read.csv("fuentes/train.csv.", header = T, sep = ",")
train$tipo <- "entrenamiento"

# Se lee el archivo de validación y su complemento, luego se realizar
# un merge de los dos
test <- read.csv("fuentes/test.csv", header = T, sep = ",")
test_gender <- read.csv("fuentes/gender_submission.csv", header = T, sep = ",")
test_merge = merge(test, test_gender, by = "PassengerId")
test_merge$tipo <- "test"

# Juntamos los dos conjuntos, entrenamiento y validación
data <- rbind(train, test_merge)
```

Para verificar los variables de interés a analizar, se transforman las variables categóricas en factores con la objetivo de realizar una exploración de los datos. A continuación se muestra un resumen de los datos.

```
# Definimos las columnas categóricas a transformar en factor
cols <- c("Survived", "Pclass", "Sex", "Ticket", "Cabin", "Embarked", "Name",
    "tipo")
data[cols] <- lapply(data[cols], factor)
summary(data)
```

```
##   PassengerId   Survived Pclass                                    Name
##  Min.   :   1   0:815    1:323   Connolly, Miss. Kate                 :   2
##  1st Qu.: 328   1:494    2:277   Kelly, Mr. James                     :   2
##  Median : 655            3:709   Abbing, Mr. Anthony                  :   1
##  Mean   : 655                    Abbott, Mr. Rossmore Edward          :   1
##  3rd Qu.: 982                    Abbott, Mrs. Stanton (Rosa Hunt)     :   1
##  Max.   :1309                    Abelson, Mr. Samuel                  :   1
##                                  (Other)                              :1301
##      Sex           Age            SibSp           Parch           Ticket
##  female:466   Min.   : 0.17   Min.   :0.0000   Min.   :0.000   CA. 2343:  11
##  male  :843   1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.000   1601    :   8
##               Median :28.00   Median :0.0000   Median :0.000   CA 2144 :   8
##               Mean   :29.88   Mean   :0.4989   Mean   :0.385   3101295 :   7
##               3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.000   347077  :   7
```

```
##                Max.   :80.00   Max.   :8.0000   Max.   :9.000   347082  :   7
##                NA's   :263                                       (Other) :1261
##       Fare                     Cabin      Embarked            tipo
##  Min.   :  0.000                 :1014    :  2    entrenamiento:891
##  1st Qu.:  7.896   C23 C25 C27   :   6   C:270    test         :418
##  Median : 14.454   B57 B59 B63 B66:   5   Q:123
##  Mean   : 33.295   G6            :   5   S:914
##  3rd Qu.: 31.275   B96 B98       :   4
##  Max.   :512.329   C22 C26       :   4
##  NA's   :1         (Other)       : 271
```

Se observa que las variables de interés son: Survived, Pclass, Sex, Age, SibSp, Parch, Fare y Embarked. Survived es la variable que se desea predecir e indica si el pasajero sobrevivió al incidente. Pclass representan la clase del pasajero y está relacionada con costó del pasaje a bordo y el camarote asignado. Las variables sex y age indican el sexo y edad del pasajero respectivamente. SibSp indica el número de hermanos, esposas y esposos a bordo por lo que pueden influenciar en si sobrevivió el pasajero, al igual que Parch, que indica el número de padres, madres e hijos a bordo. Fare y Embarked indican el costo del pasaje a bordo y desde dónde abordó el pasajero.

En cambio las variables que se considera no tienen interés son: PassengerId, Name, Ticket y Cabin. La variable PassengerId corresponde a una secuencia de números que inicia en 1 e identifica a cada uno de los pasajeros pero no aporta mayor información. La variable Name corresponde al nombre y título del pasajero, en el estado actual no tiene tanta relevancia pero se limpiará la variable para conservar solo el título, con lo que se puede asociar pasajeros. Por otro lado la variable Ticket en su mayoría está vacía con 1261 valores faltantes de 1309 por lo que se descartará la variable. Finalmente la variable Cabin indica el camarote que utilizó el pasajero y por consiguiente la ubicación relativa dentro del Titanic, se observa que existen 1285 valores faltantes por lo que se podría eliminar la variable, pero debido a su relevancia se conservará.

En conclusión, las variables que no se tomaran en cuenta para el análisis son: PassengerId y Ticket.

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

3 Limpieza de los datos.

Para verificar la cantidad de valores vacíos o nulos se utiliza el siguiente procedimiento que nos indica que en la variable Age faltan 263 valores, en la variable Fer 1 valor, en la variable Cabin 1014 y en la variable Embarked 2. Para tratar estas variables no se van a eliminar los valores faltantes ya que se pierde información, en cambio se utilizarán diferente métodos para completar la información faltante.

```
colSums(is.na(data) | data == "")
```

```
## Survived   Pclass     Name      Sex      Age    SibSp    Parch     Fare
##        0        0        0        0      263        0        0        1
##    Cabin Embarked     tipo
##     1014        2        0
```

3.1 Tratamiento de la variable Age

Se inicia tratando los 263 valores faltantes de la variable Age, se utilizará la media de cada sexo para completar las edades faltantes de los pasajeros de cada sexo.

```r
summary(is.na(data$Age))
```

```
##    Mode   FALSE    TRUE
## logical    1046     263
```

```r
# Se obtiene el conjunto de edades de las mujeres y se completa los
# valores faltantes
data_subset_female = data[data$Sex == "female", ]
data[which(is.na(data$Age) & data$Sex == "female"), "Age"] <- round(median(data_subset_female$Age,
    na.rm = T), 0)

# Se obtiene el conjunto de edades de las hombres y se completa los
# valores faltantes
data_subset_male = data[data$Sex == "male", ]
data[is.na(data$Age) & data$Sex == "male", "Age"] <- round(median(data_subset_male$Age,
    na.rm = T), 0)

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

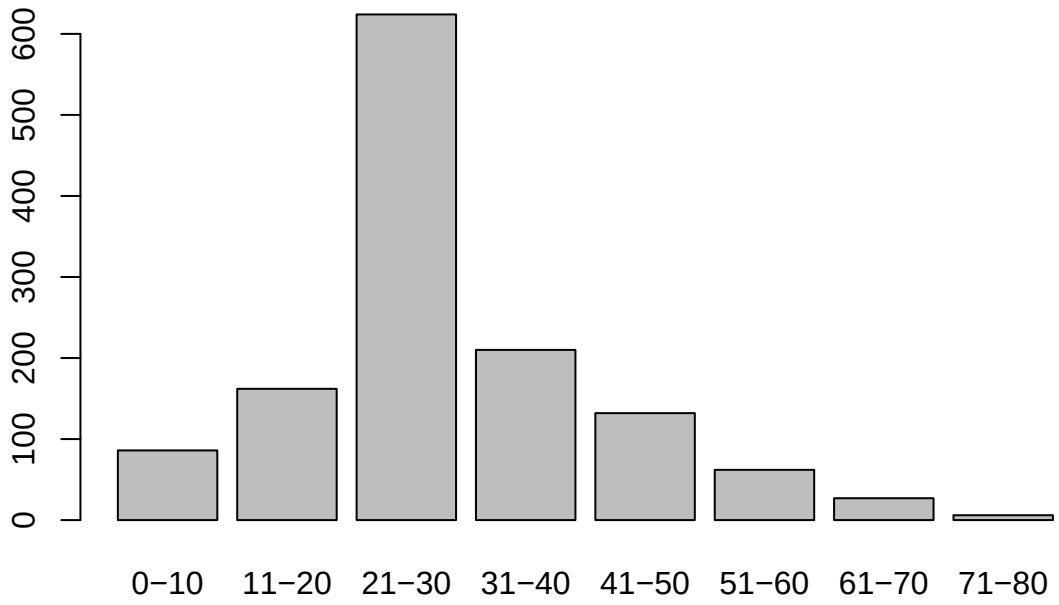
```
##    Mode   FALSE
## logical    1309
```

A continuación se categoriza la edad de los pasageros en rangos de 10 años, en los resultados se observa que la mayor cantidad de personas están en el rango 21 a 30 años mientras que la la minoría está en el rango de 71 a 80 años.

```r
summary(data$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.17   22.00   28.00   29.44   35.00   80.00
```

```r
data$Age_Segmented <- cut(data$Age, breaks = c(0, 10, 20, 30, 40, 50, 60,
    70, 80), labels = c("0-10", "11-20", "21-30", "31-40", "41-50", "51-60",
    "61-70", "71-80"))
plot(data$Age_Segmented)
```



3.2 Tratamiento de la variable Fare

La siguiente variable a tratar es Fare que tiene un valor faltante y 16 con valor cero, para tratar los valores vacíos se utilizará la clase del pasajero y dónde embarcó. Las clases de pasajero con datos faltantes son 1, 2 y 3 y todos embarcaron desde Southampton. A continuación extraemos un subconjunto de datos de las personas que cumplan esas condiciones para obtener el valor promedio del costo de sus pasajes o tarifas.

```
# Revisamos el contexto de persona
data[is.na(data$Fare) | data$Fare == 0, ]
```

```
##      Survived Pclass                                Name  Sex  Age SibSp
## 180         0      3                 Leonard, Mr. Lionel male 36.0     0
## 264         0      1               Harrison, Mr. William male 40.0     0
## 272         1      3         Tornquist, Mr. William Henry male 25.0     0
## 278         0      2          Parkes, Mr. Francis "Frank" male 28.0     0
## 303         0      3       Johnson, Mr. William Cahoone Jr male 19.0     0
## 414         0      2        Cunningham, Mr. Alfred Fleming male 28.0     0
## 467         0      2                 Campbell, Mr. William male 28.0     0
## 482         0      2      Frost, Mr. Anthony Wood "Archie" male 28.0     0
## 598         0      3                  Johnson, Mr. Alfred male 49.0     0
## 634         0      1         Parr, Mr. William Henry Marsh male 28.0     0
## 675         0      2            Watson, Mr. Ennis Hastings male 28.0     0
## 733         0      2                  Knight, Mr. Robert J male 28.0     0
## 807         0      1               Andrews, Mr. Thomas Jr male 39.0     0
## 816         0      1                     Fry, Mr. Richard male 28.0     0
```

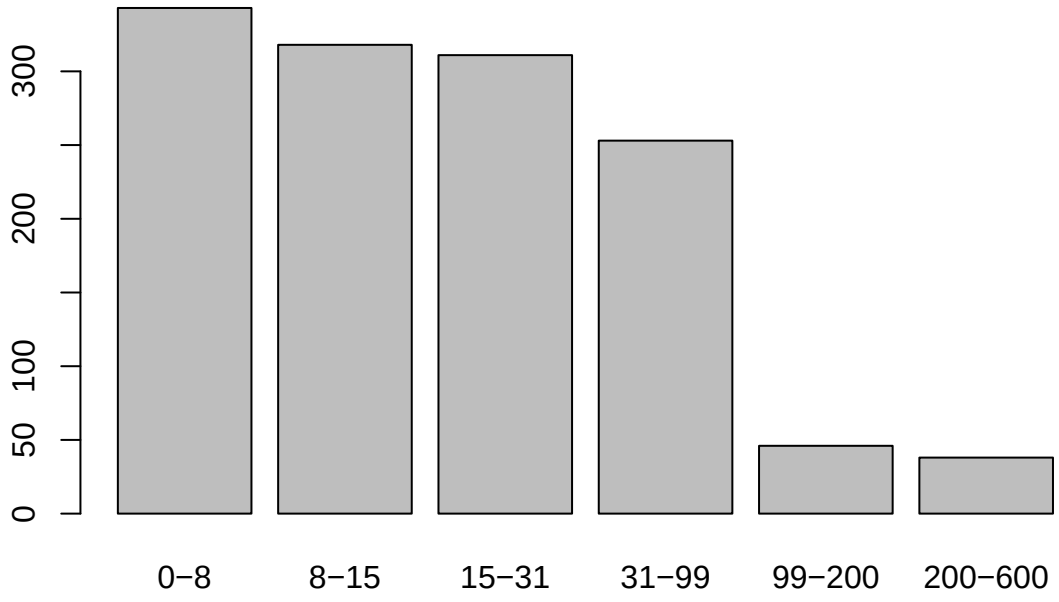
```
## 823         0      1       Reuchlin, Jonkheer. John George male 38.0     0
## 1044        0      3                   Storey, Mr. Thomas male 60.5     0
## 1158        0      1 Chisholm, Mr. Roderick Robert Crispin male 28.0     0
## 1264        0      1               Ismay, Mr. Joseph Bruce male 49.0     0
##      Parch Fare       Cabin Embarked          tipo Age_Segmented
## 180      0    0                   S entrenamiento         31-40
## 264      0    0         B94       S entrenamiento         31-40
## 272      0    0                   S entrenamiento         21-30
## 278      0    0                   S entrenamiento         21-30
## 303      0    0                   S entrenamiento         11-20
## 414      0    0                   S entrenamiento         21-30
## 467      0    0                   S entrenamiento         21-30
## 482      0    0                   S entrenamiento         21-30
## 598      0    0                   S entrenamiento         41-50
## 634      0    0                   S entrenamiento         21-30
## 675      0    0                   S entrenamiento         21-30
## 733      0    0                   S entrenamiento         21-30
## 807      0    0         A36       S entrenamiento         31-40
## 816      0    0        B102       S entrenamiento         21-30
## 823      0    0                   S entrenamiento         31-40
## 1044     0   NA                   S          test         61-70
## 1158     0    0                   S          test         21-30
## 1264     0    0 B52 B54 B56       S          test         41-50
```

```r
# Extraemos el conjunto de datos del que se sacará la media para la
# clase 1 y que embarcaron en Southampton.
data_filteredFare = data[data$Pclass == 1 & data$Embarked == "S", ]
data[data$Fare == 0 & data$Pclass == 1, "Fare"] <- round(mean(data_filteredFare$Fare,
    na.rm = T), 0)

# Extraemos el conjunto de datos del que se sacará la media para la
# clase 2 y que embarcaron en Southampton.
data_filteredFare = data[data$Pclass == 2 & data$Embarked == "S", ]
data[data$Fare == 0 & data$Pclass == 2, "Fare"] <- round(mean(data_filteredFare$Fare,
    na.rm = T), 0)

# Extraemos el conjunto de datos del que se sacará la media para la
# clase 3 y que embarcaron en Southampton.
data_filteredFare = data[data$Pclass == 3 & data$Embarked == "S", ]
data[is.na(data$Fare) & data$Pclass == 3, "Fare"] <- round(mean(data_filteredFare$Fare,
    na.rm = T), 0)
data[data$Fare == 0 & data$Pclass == 3, "Fare"] <- round(mean(data_filteredFare$Fare,
    na.rm = T), 0)

summary(is.na(data$Fare) | data$Fare == 0)
```

```
##    Mode   FALSE
## logical    1309
```

Ahora para discretizar la variable Fare creamos 5 grupos de registros similares, ya que los valores son decimales el valor inicial no incluye pero el final si, solo el entero. Por ejemplo en el rango (0-8], incluye valores desde cero hasta 8.0, los valores mayores a 8.0 corresponden al rango siguiente (8-15].

```
data$Fare_Segmented <- cut(data$Fare, breaks = c(0, 8, 15, 31, 99, 200,
    600), labels = c("0-8", "8-15", "15-31", "31-99", "99-200", "200-600"))
plot(data$Fare_Segmented)
```

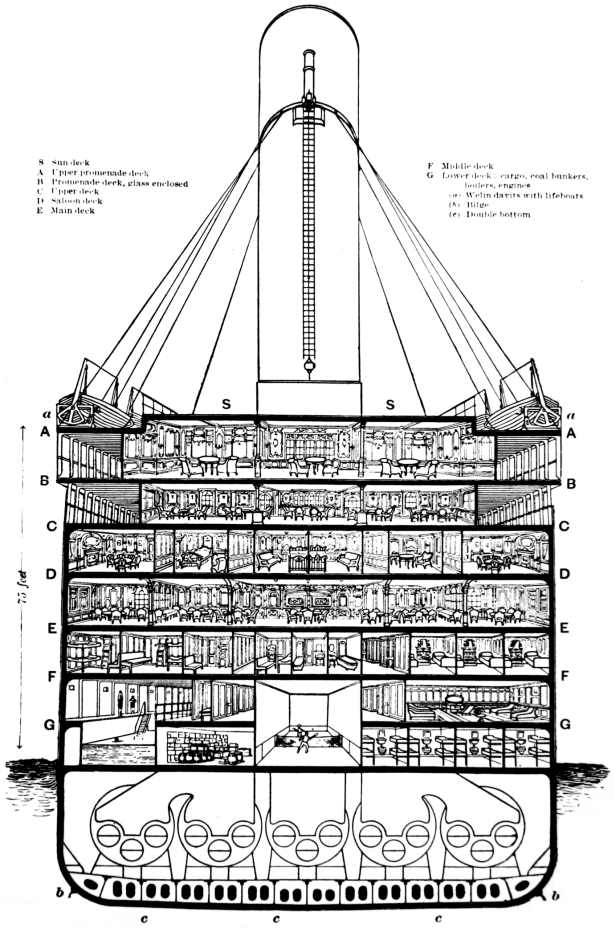


3.3 Tratamiento de la variable Cabin

Los registros vacíos de la variable Cabin se completan con el valor **N** ya que la variable es categórica, los demás datos se van a tratar de forma que solo quede las letras de los camarotes.

```
# Convertimos a carácter la variable para reemplazar los valores.
data$Cabin <- as.character(data$Cabin)

# Reemplazamos los valores vacíos por N
data$Cabin[is.na(data$Cabin) | data$Cabin == ""] <- "N"
data$Cabin[data$Cabin == "T"] <- "N"
# cortamos la primera letra de cada registro.
data$Cabin <- as.factor(substr(data$Cabin, start = 1, stop = 1))
```

A continuación se puede observar la distribución de los camarones en el Titanic.



##

```
table(data$Cabin)
```

```
##
##     A     B     C     D     E     F     G     N
##    22    65    94    46    41    21     5 1015
```

3.4 Tratamiento de la variable Embarked

La variable Embarked tiene dos valores faltantes, al buscar los pasajeros en la Enciclopedia Titánica se verifica que los dos pasajeros embarcaron en Southampton lo cual coincide con el valor más frecuente de embarcado: S.

```
# Verificamos la frecuencia de los valores
summary(data$Embarked)
```

```
##        C   Q   S
##   2 270 123 914
```

```
# Reemplazamos por el valor S los campos vacíos, convertimos en factor
# nuevamente y verificamos los resultados.
data[which(data$Embarked == ""), "Embarked"] <- "S"
```

```r
data$Embarked <- factor(data$Embarked)

summary(data$Embarked)
```

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

3.5 Tratamiento de la variable Name

La variable Name se compone del título de la persona acompañado de su nombre, la variable en el estado actual no brinda mayor información por lo que se limpiará para dejar solo el título de persona con lo que se puede agrupar a las personas y analizar datos. Primero se divide los valores de la variable tomando como separador un espacio y se deja solo los valores que contengan un punto ya que el título de la persona en todos los casos está acompañado de un punto: Mr. Miss., etc.

```r
# Declaramos la función trim para eliminar espacios al inicio o fin de
# un registro
trim <- function(x) gsub("^\\s+|\\s+$", "", x)

# Dividimos la variable Name con un espacio como separador, creamos más
# columnas
data_split = separate(data, "Name", paste("Name", 2:7, sep = ""), sep = " ",
    extra = "drop")

# Eliminamos todo lo que no tenga un punto en su valor o que su
# longitud sea igual a 2
data_split$Name2[!grepl(".", data_split$Name2, fixed = TRUE)] <- ""
data_split$Name3[!grepl(".", data_split$Name3, fixed = TRUE)] <- ""
data_split$Name4[!grepl(".", data_split$Name4, fixed = TRUE)] <- ""
data_split$Name5[!grepl(".", data_split$Name5, fixed = TRUE)] <- ""
data_split$Name6[!grepl(".", data_split$Name6, fixed = TRUE) || length(data_split$Name7) ==
    2] <- ""
data_split$Name7[!grepl(".", data_split$Name7, fixed = TRUE)] <- ""

# Juntamos las columnas antes creadas al dividir Name, eliminados los
# espacios vacíos y eliminamos la columnas extra
data_split$Name <- trim(paste(data_split$Name2, data_split$Name3, data_split$Name4,
    data_split$Name5, data_split$Name6, data_split$Name7))
data_split = subset(data_split, select = -c(Name2, Name3, Name4, Name5,
    Name6, Name7))
data <- data_split
```

Se observa que existen 18 títulos muchos de ellos con uno o dos valores por lo que se procede a reemplazar por "Otros." aquellos valores que tengan menos de 10 repeticiones. El resultado son 5 valores distintos, Master, Miss, Mr, Mrs y Otros.

```r
table(data$Name)
```

```
##
##     Capt.      Col. Countess.      Don.     Dona.       Dr. Jonkheer.     Lady.
##         1         4         1         1         1         8         1         1
```

```
##    Major.   Master.     Miss.     Mlle.      Mme.       Mr.      Mrs.       Ms.
##         2       61       260         2         1       757       197         2
##      Rev.      Sir.
##         8         1
```

```
data$Name <- as.character(data$Name)
data$Name <- with(data, ave(Name, Name, FUN = function(i) replace(i, length(i) <
    10, "Otros.")))
data$Name <- as.factor(data$Name)
summary(data$Name)
```

```
## Master.   Miss.     Mr.    Mrs.  Otros.
##      61     260     757     197      34
```

3.6 Tratamiento de la familia del pasajero

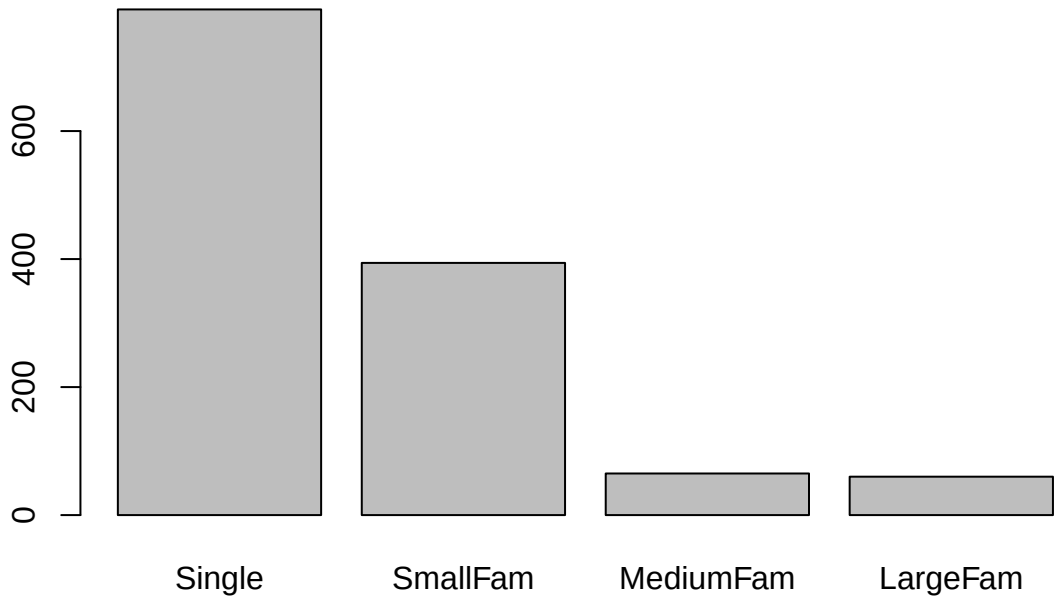
El pasajero cuenta con las variables Parch y SibSp que juntas indican el tamaño de la familia a bordo. A continuación estás variables se juntan y categorizan en los grupos:

- Single (Family_Size = 1),
- SmallFam (Family_Size entre 1 y 3),
- MediumFam (Family_Size entre 4 y 5) y
- LargarFam (Family_Size > 5).

```
data$Family_Size <- data$SibSp + data$Parch + 1
summary(data$Family_Size)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   1.000   1.884   2.000  11.000
```

```
data$Family_Segmented <- cut(data$Family_Size, breaks = c(0, 1, 3, 5, 11),
    labels = c("Single", "SmallFam", "MediumFam", "LargeFam"))
plot(data$Family_Segmented)
```



3.7 Identificación y tratamiento de valores extremos

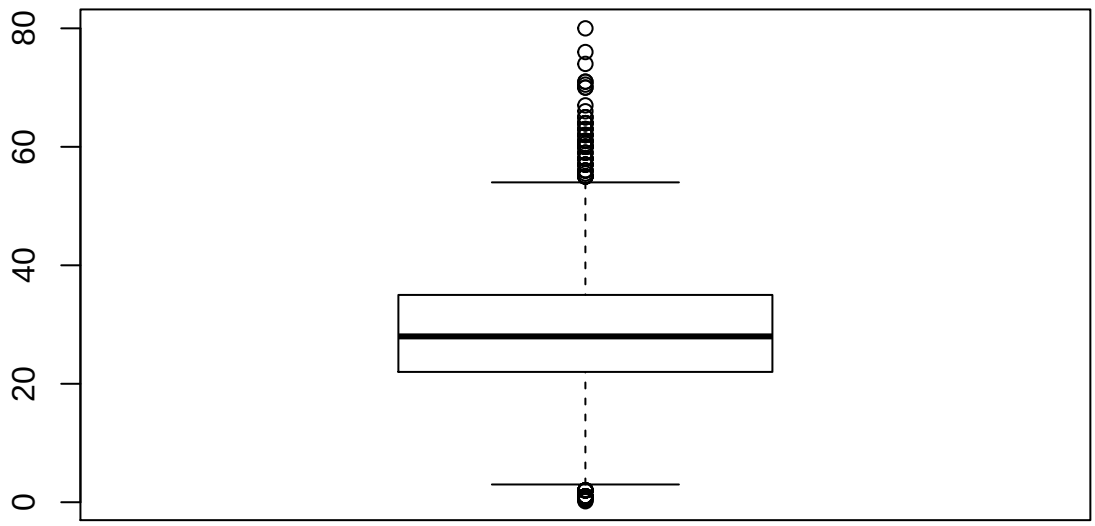
A continuación revisamos las variables numéricas para verificar y tratar los valores extremos.

Iniciamos con la variable Age, obtenemos los valores únicos y los ordenamos para observar los valores que la variable puede tomar, en los resultados a continuación se constata que la edad mínima es de 0.17 años y la máxima de 80. Inicialmente se podría pensar que los valores menores a cero son un error en los datos, pero buscando a los pasajeros por su nombre se comprobó que los datos son correctos.

```
unique(sort(data$Age))
```

```
##  [1]  0.17  0.33  0.42  0.67  0.75  0.83  0.92  1.00  2.00  3.00  4.00  5.00
## [13]  6.00  7.00  8.00  9.00 10.00 11.00 11.50 12.00 13.00 14.00 14.50 15.00
## [25] 16.00 17.00 18.00 18.50 19.00 20.00 20.50 21.00 22.00 22.50 23.00 23.50
## [37] 24.00 24.50 25.00 26.00 26.50 27.00 28.00 28.50 29.00 30.00 30.50 31.00
## [49] 32.00 32.50 33.00 34.00 34.50 35.00 36.00 36.50 37.00 38.00 38.50 39.00
## [61] 40.00 40.50 41.00 42.00 43.00 44.00 45.00 45.50 46.00 47.00 48.00 49.00
## [73] 50.00 51.00 52.00 53.00 54.00 55.00 55.50 56.00 57.00 58.00 59.00 60.00
## [85] 60.50 61.00 62.00 63.00 64.00 65.00 66.00 67.00 70.00 70.50 71.00 74.00
## [97] 76.00 80.00
```

```
bp <- boxplot(data$Age)
```




```
bp$out
```

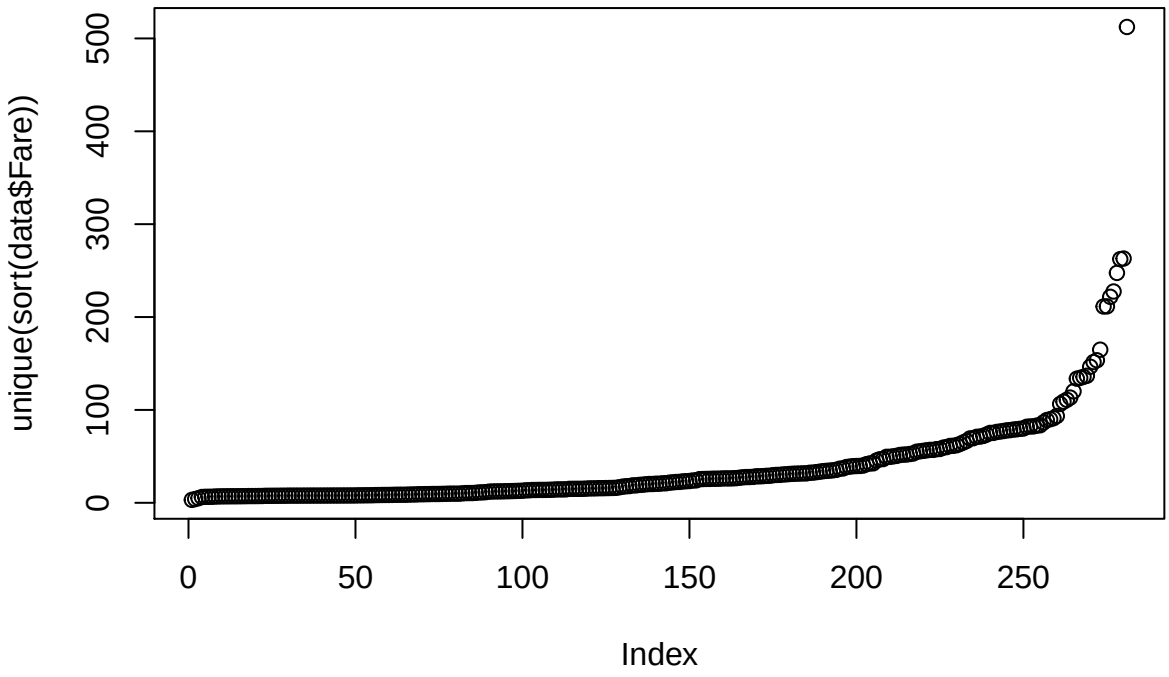
```
##   [1]  2.00 58.00 55.00  2.00 66.00 65.00  0.83 59.00 71.00 70.50  2.00 55.50
##  [13]  1.00 61.00  1.00 56.00  1.00 58.00  2.00 59.00 62.00 58.00 63.00 65.00
##  [25]  2.00  0.92 61.00  2.00 60.00  1.00  1.00 64.00 65.00 56.00  0.75  2.00
##  [37] 63.00 58.00 55.00 71.00  2.00 64.00 62.00 62.00 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 62.00 63.00 55.00 60.00 60.00 55.00
##  [73] 67.00  2.00 76.00 63.00  1.00 61.00 60.50 64.00 61.00  0.33 60.00 57.00
##  [85] 64.00 55.00  0.92  1.00  0.75  2.00  1.00 64.00  0.83 55.00 55.00 57.00
##  [97] 58.00  0.17 59.00 55.00 57.00
```

Siguiente variable a verificar es el precio del pasaje (*Fare*) se observa que el mínimo valor presente es de 3.71 y el máximo de 512 libras británicas. Luego de investigar los valores de los pasajes del Titanic se encontró que estaban entre 870-4350 para una suite de primera clase, entre 30-150 para un camarote de primera clase, entre 12-60 para segunda clase y entre 3-8 para tercera clase. Al contrastar estos valores con los del dataset se verifica que están dentro de los rangos de los precios establecidos, aunque en la gráfica se observa que el valor de 512 está muy alejado de los demás registros se verificó que corresponde a 3 personas de 1era clase que al menos uno de ellos tiene varios camarotes.

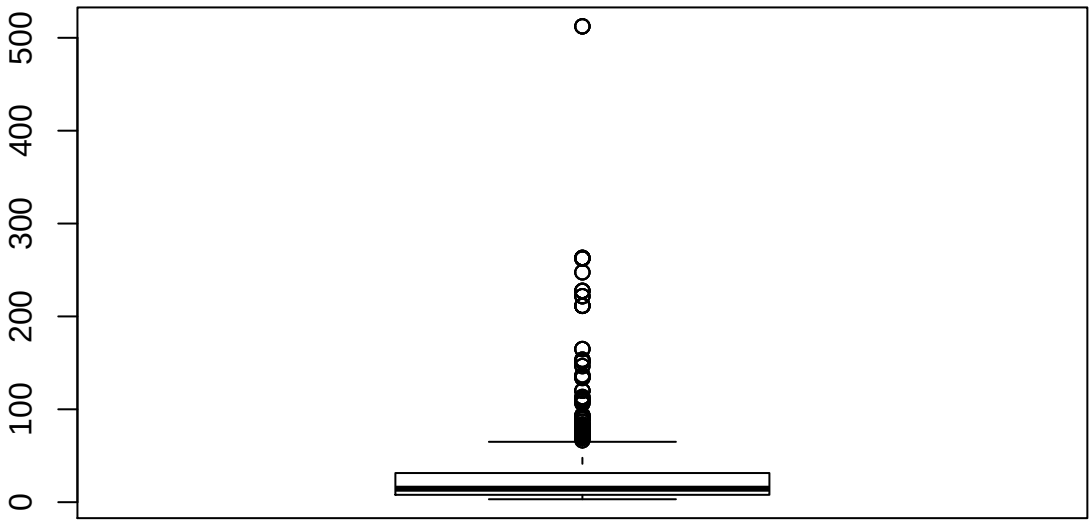
```
summary(data$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.171   7.925  14.500  33.805  31.387 512.329
```

```r
plot(unique(sort(data$Fare)))
```




```r
boxplot(data$Fare)
```

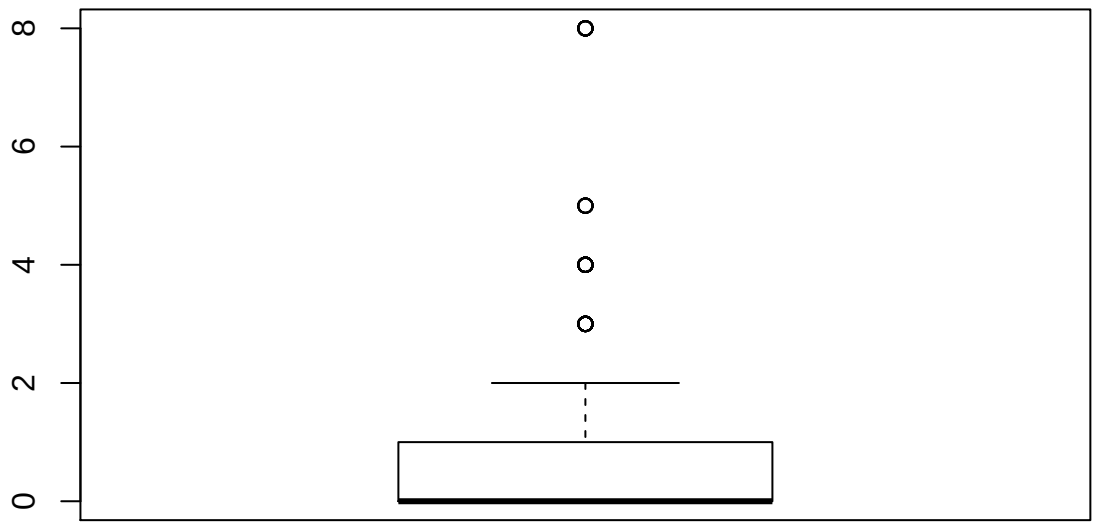


En el número de hermanos y parientes está dentro de un rango razonable y no existen valores que sean llamativos.

```
summary(data$SibSp)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.4989  1.0000  8.0000
```

```
bpsi <- boxplot(data$SibSp)
```




```
bpsi$out
```

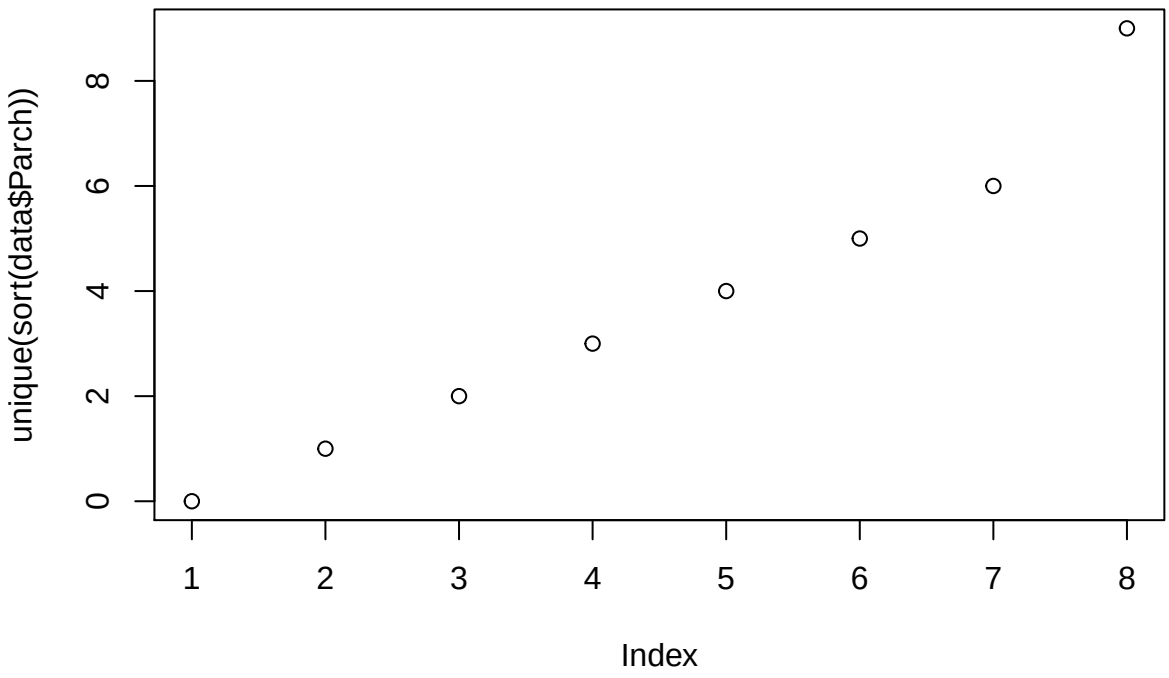
```
##  [1] 3 4 3 3 4 5 3 4 5 3 3 4 8 4 4 3 8 4 8 3 4 4 4 4 8 3 3 5 3 5 3 4 4 3 3 5 4 3
## [39] 4 8 4 3 4 8 4 8 3 4 5 3 4 8 4 8 4 3 3
```

De igual manera el número de padres e hijos se observa que no existen datos fuera de lo normal

```
summary(data$Parch)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.385   0.000   9.000
```

```
plot(unique(sort(data$Parch)))
```



4 Análisis de los datos.

4.1 Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).

En este caso se han eliminado las variables passenge y ticket que no aportaban información, además se han creado nuevas variables apartir de variables existentes en los datos y se han procesado los datos.

En este apartado se tomaran las todas las variables en los datos menos las que se han eliminado para realizar los análisis y comparaciones. Se usará a la variable Survived como variable dependiente y como variable base para comparar con las demás variables, y obtener un perfil de los que sobrevivieron y los que no.

Planificación:

4.2 Comprobación de la normalidad y homogeneidad de la varianza.

- Se probará hipótesis de normalidad y varianza para las variable Age y Fare.

Normalidad

Hipótesis:

H_0 : La variable Age sigue una distribución normal.
H_1 : La variable Age no sigue una distribución normal.

H_0 : La variable Fare sigue una distribución normal.
H_1 : La variable Fare no sigue una distribución normal.

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

```
##
##  Shapiro-Wilk normality test
##
## data:  data$Age
## W = 0.95166, p-value < 2.2e-16
```

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

```
##
##  Shapiro-Wilk normality test
##
## data:  data$Fare
## W = 0.52785, p-value < 2.2e-16
```

Análisis: La Variable Age y Fare no siguen una distribución normal, ya que el p_value es $< \alpha$.

Varianza

Hipótesis:

$H_0 : \sigma^2_{Age0} = \sigma^2_{Age1}$

$H_1 : \sigma^2_{Age0} \neq \sigma^2_{Age1}$

$H_0 : \sigma^2_{Fare0} = \sigma^2_{Fare1}$

$H_1 : \sigma^2_{Fare0} \neq \sigma^2_{Fare1}$

```
var.test(data$Age[data$Survived == "1"], data$Age[data$Survived == "0"])
```

```
##
##  F test to compare two variances
##
## data:  data$Age[data$Survived == "1"] and data$Age[data$Survived == "0"]
## F = 1.2818, num df = 493, denom df = 814, p-value = 0.001876
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  1.095799 1.504049
## sample estimates:
## ratio of variances
##           1.281796
```

```
var.test(data$Fare[data$Survived == "1"], data$Fare[data$Survived == "0"])
```

```
##
##  F test to compare two variances
##
## data:  data$Fare[data$Survived == "1"] and data$Fare[data$Survived == "0"]
## F = 3.8372, num df = 493, denom df = 814, p-value < 2.2e-16
```

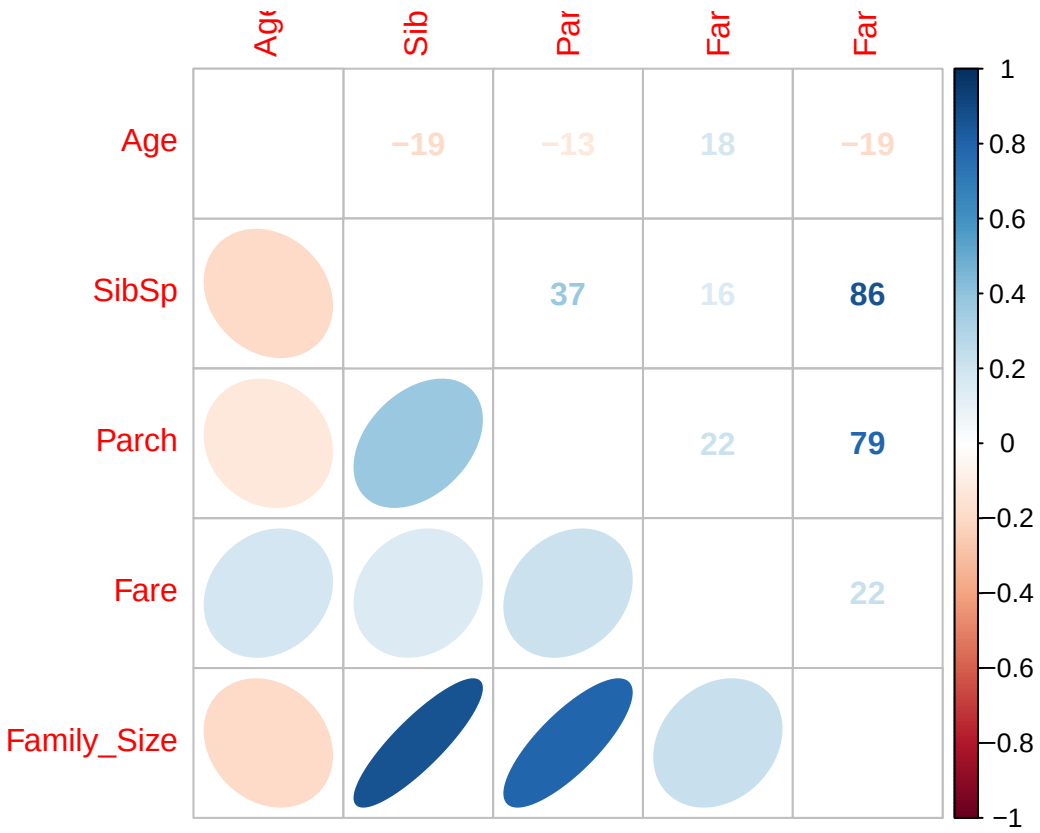
```
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  3.280404 4.502548
## sample estimates:
## ratio of variances
##           3.837208
```

Análisis: en los dos var.test anteriores el p_value es $< \alpha$ entonces se rechaza la hipótesis nula de homocedasticidad en las varianzas, entonces se acepta la hipotesis alternativas existe heterocedasticidad entre las varianzas, son diferentes.

4.3 Aplicación de pruebas estadísticas para comparar los grupos de datos. En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc. Aplicar al menos tres métodos de análisis diferentes.

- Se aplicará correlaciones.

```
correlaciones <- data %>% select(c(4:7, 14)) %>% cor()
corrplot.mixed(correlaciones, lower = "ellipse", addCoefasPercent = T,
    tl.pos = "lt", diag = "n", upper = "number")
```



Se observa que hay correlación fuerte directa entre la variable Family_size con PArch y SibSp, esto tiene sentido ya que Famili_size es una variable calculada a partir de estas dos variables.

- Tes de comparación de medianas

Se aplicará test no paramétrico de Mann-Whitney / Wilcoxon para dos muestras ya que las variables no son normales, aunque podria justificarlo dado el teorema central del límite ya que las muestras son superiores a 30.

Hipótesis: $H_0 : Me_{A0} = Me_{A1}$ $H_1 : Me_{A0} \neq Me_{A1}$

$H_0 : Me_{F0} = Me_{F1}$ $H_1 : Me_{F0} \neq Me_{F1}$

```
wilcox.test(data$Age[data$Survived == "1"], data$Age[data$Survived == "0"])
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  data$Age[data$Survived == "1"] and data$Age[data$Survived == "0"]
## W = 188425, p-value = 0.0514
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(data$Fare[data$Survived == "1"], data$Fare[data$Survived ==
    "0"])
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  data$Fare[data$Survived == "1"] and data$Fare[data$Survived == "0"]
## W = 265530, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Interpretación:

- Se concluye que la mediana de la edad de los sobrevientes y la mediana de la edad de los no sobrevivientes son iguales.
- Se concluye que la mediana del costo del ticket de los sobrevientes y la mediana del costo del ticket de los no sobrevivientes son diferentes.
- Se aplicará pruebas chi cuadrado de independencia.

H_0 : No existe asociación entre la variable dependiente Survived y la variable independiente. No existe dependencia entre las variables.

H_1 : Existe asociación entre la variable dependiente Survived y la variable independiente. Existe dependencia entre las variables.

```
tabla1 <- table(data$Survived, data$Pclass)
chisq.test(tabla1)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla1
## X-squared = 91.724, df = 2, p-value < 2.2e-16
```

```r
tabla2 <- table(data$Survived, data$Sex)
chisq.test(tabla2)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabla2
## X-squared = 617.31, df = 1, p-value < 2.2e-16
```

```r
tabla3 <- table(data$Survived, data$SibSp)
chisq.test(tabla3)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla3
## X-squared = 44.565, df = 6, p-value = 5.711e-08
```

```r
tabla4 <- table(data$Survived, data$Parch)
chisq.test(tabla4)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla4
## X-squared = 45.827, df = 7, p-value = 9.445e-08
```

```r
tabla5 <- table(data$Survived, data$Cabin)
chisq.test(tabla5)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla5
## X-squared = 94.95, df = 7, p-value < 2.2e-16
```

```r
tabla6 <- table(data$Survived, data$Embarked)
chisq.test(tabla6)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla6
## X-squared = 24.194, df = 2, p-value = 5.577e-06
```

```r
tabla7 <- table(data$Survived, data$Age_Segmented)
chisq.test(tabla7)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla7
## X-squared = 23.65, df = 7, p-value = 0.001313
```

```
tabla8 <- table(data$Survived, data$Fare_Segmented)
chisq.test(tabla8)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla8
## X-squared = 97.606, df = 5, p-value < 2.2e-16
```

```
tabla9 <- table(data$Survived, data$Name)
chisq.test(tabla9)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla9
## X-squared = 624.52, df = 4, p-value < 2.2e-16
```

```
tabla10 <- table(data$Survived, data$Family_Size)
chisq.test(tabla10)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla10
## X-squared = 101.97, df = 8, p-value < 2.2e-16
```

```
tabla11 <- table(data$Survived, data$Family_Segmented)
chisq.test(tabla11)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla11
## X-squared = 85.565, df = 3, p-value < 2.2e-16
```

Análisis: Se concluye que hay asociación o dependencia entre la variable dependiente Survived y cada una de las variables independientes categóricas.

- Se aplicará Anova o test de Kruskal Wallis para probar diferencias significativas entre las variables.

H_0 : No existe diferencia en la mediana de las variables (Age y Fare) entre los distintos grupos de las variables categóricas.

H_1 : No existe diferencia en la mediana de las variables (Age y Fare) entre los distintos grupos de las variables categóricas.

```r
kruskal.test(Age ~ Survived, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Survived
## Kruskal-Wallis chi-squared = 3.7953, df = 1, p-value = 0.0514
```

```r
kruskal.test(Fare ~ Survived, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Survived
## Kruskal-Wallis chi-squared = 93.89, df = 1, p-value < 2.2e-16
```

```r
kruskal.test(Age ~ Sex, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Sex
## Kruskal-Wallis chi-squared = 10.058, df = 1, p-value = 0.001517
```

```r
kruskal.test(Fare ~ Sex, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Sex
## Kruskal-Wallis chi-squared = 64.637, df = 1, p-value = 9.006e-16
```

```r
kruskal.test(Age ~ Pclass, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Pclass
## Kruskal-Wallis chi-squared = 168.61, df = 2, p-value < 2.2e-16
```

```r
kruskal.test(Fare ~ Pclass, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Pclass
## Kruskal-Wallis chi-squared = 744.81, df = 2, p-value < 2.2e-16
```

```r
kruskal.test(Age ~ Cabin, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Cabin
## Kruskal-Wallis chi-squared = 125.48, df = 7, p-value < 2.2e-16
```

```r
kruskal.test(Fare ~ Cabin, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Cabin
## Kruskal-Wallis chi-squared = 465.88, df = 7, p-value < 2.2e-16
```

```r
kruskal.test(Age ~ Embarked, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Embarked
## Kruskal-Wallis chi-squared = 7.2843, df = 2, p-value = 0.0262
```

```r
kruskal.test(Fare ~ Embarked, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Embarked
## Kruskal-Wallis chi-squared = 135.72, df = 2, p-value < 2.2e-16
```

```r
kruskal.test(Age ~ Cabin, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Cabin
## Kruskal-Wallis chi-squared = 125.48, df = 7, p-value < 2.2e-16
```

```r
kruskal.test(Fare ~ Cabin, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Cabin
## Kruskal-Wallis chi-squared = 465.88, df = 7, p-value < 2.2e-16
```

```r
kruskal.test(Age ~ Name, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Name
## Kruskal-Wallis chi-squared = 274.42, df = 4, p-value < 2.2e-16
```

```r
kruskal.test(Fare ~ Name, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Name
## Kruskal-Wallis chi-squared = 148.12, df = 4, p-value < 2.2e-16
```

```r
kruskal.test(Age ~ Family_Segmented, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Family_Segmented
## Kruskal-Wallis chi-squared = 47.83, df = 3, p-value = 2.315e-10
```

```r
kruskal.test(Fare ~ Family_Segmented, data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Family_Segmented
## Kruskal-Wallis chi-squared = 347.07, df = 3, p-value < 2.2e-16
```

Análisis: Solo en el primer test se acepta la hipótesis nula, en las demás se acepta la hipotesis alternativa.

- Se aplicará un modelo de clasificación para predecir la variable Survived.

4.4 Exportacion de datos procesados

Ahora ya con los datos limpios y tratados se divide el dataset en un conjunto de entrenamiento y uno de evaluación.

```r
# Se extrae el conjunto de evaluación
test <- data %>% filter(tipo == "test")
test = subset(test, select = -c(tipo))

# Se extrae el conjunto de entrenamiento
train <- data %>% filter(tipo == "entrenamiento")
train = subset(train, select = -c(tipo))

# Se persisten los conjuntos de datos en dos archivos. write.csv(test,
# 'fuentes/test_clean.csv', row.names = T) write.csv(train,
# 'fuentes/train_clean.csv', row.names = T)
```

4.5 Leer archivos limpios

```
# test <- read.csv('fuentes/test_clean.csv') train <-
# read.csv('fuentes/train_clean.csv.')
```

Escalar variables:

```
train$Age <- scale(train$Age, center = T, scale = T)
train$Fare <- scale(train$Fare, center = T, scale = T)
test$Age <- scale(test$Age, center = T, scale = T)
test$Fare <- scale(test$Fare, center = T, scale = T)
```

Random Forest:

```
myControl <- trainControl(method = "cv", number = 5)
model_rf1 <- train(Survived ~ ., train, method = "rf", trControl = myControl,
    importance = TRUE)
model_rf1
```

```
## Random Forest
##
## 891 samples
##  13 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 714, 712, 713, 713, 712
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.8215655  0.6127037
##   19    0.8316781  0.6381697
##   36    0.8283262  0.6318360
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 19.
```

```
varImp(model_rf1)
```

```
## rf variable importance
##
##   only 20 most important variables shown (out of 36)
##
##                           Importance
## Pclass3                       100.00
## NameMr.                        89.10
## Fare                           79.94
## Age                            72.27
## Sexmale                        70.99
## Family_Size                    52.77
```

```
## EmbarkedS                    43.20
## CabinN                       40.25
## CabinE                       40.17
## Fare_Segmented8-15           40.01
## Pclass2                      36.23
## SibSp                        30.01
## Age_Segmented21-30           26.07
## Family_SegmentedSmallFam     22.83
## EmbarkedQ                    22.70
## NameOtros.                   22.35
## Fare_Segmented31-99          22.05
## Age_Segmented31-40           21.95
## Family_SegmentedLargeFam     20.11
## Fare_Segmented200-600        19.58
```

```
pred_rf1 <- predict(model_rf1, newdata = test)
confusionMatrix(pred_rf1, test$Survived)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 243  37
##          1  23 115
##
##                Accuracy : 0.8565
##                  95% CI : (0.8191, 0.8886)
##     No Information Rate : 0.6364
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.6836
##
##  Mcnemar's Test P-Value : 0.09329
##
##             Sensitivity : 0.9135
##             Specificity : 0.7566
##          Pos Pred Value : 0.8679
##          Neg Pred Value : 0.8333
##              Prevalence : 0.6364
##          Detection Rate : 0.5813
##    Detection Prevalence : 0.6699
##       Balanced Accuracy : 0.8351
##
##        'Positive' Class : 0
##
```

Se escogen las 8 variables más importantes paraa hacer un modelo sin variables correlacionadas, es decir se observa si es más importante Age o Age_Segmented, o SibSp o Parch o Family_size o Family_segmented, etc.

```
train1 <- train %>% dplyr::select(c(1:4, 7:9, 12:13))
test1 <- test %>% dplyr::select(c(1:4, 7:9, 12:13))
model_rf2 <- train(Survived ~ ., train1, method = "rf", trControl = myControl,
```

```
    importance = TRUE)
model_rf2
```

```
## Random Forest
##
## 891 samples
##   8 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 714, 713, 713, 712, 712
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.8170463  0.6013156
##   10    0.8316532  0.6379444
##   19    0.8305549  0.6378793
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 10.
```

```
pred_rf2 <- predict(model_rf2, newdata = test1)
confusionMatrix(pred_rf2, test1$Survived)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 244  36
##          1  22 116
##
##                Accuracy : 0.8612
##                  95% CI : (0.8243, 0.8929)
##     No Information Rate : 0.6364
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.6942
##
##  Mcnemar's Test P-Value : 0.08783
##
##             Sensitivity : 0.9173
##             Specificity : 0.7632
##          Pos Pred Value : 0.8714
##          Neg Pred Value : 0.8406
##              Prevalence : 0.6364
##          Detection Rate : 0.5837
##    Detection Prevalence : 0.6699
##       Balanced Accuracy : 0.8402
##
##        'Positive' Class : 0
##
```

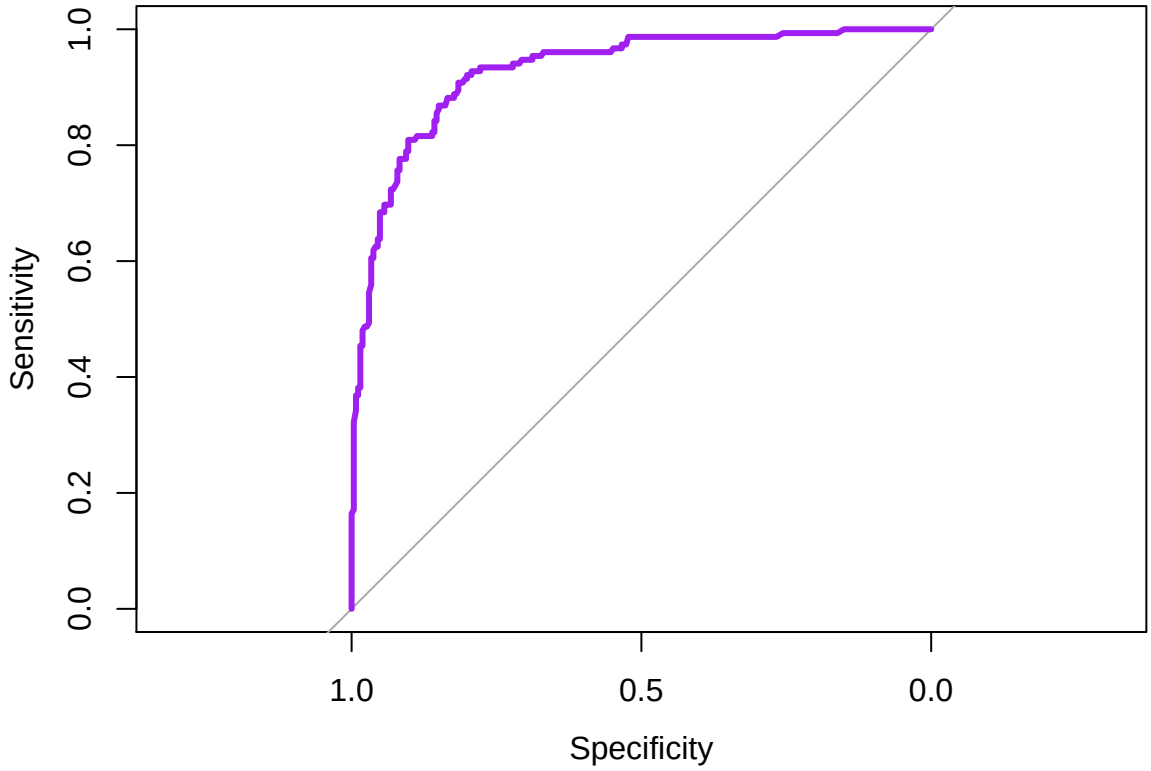
Se observan buenas medidas para la accuracy. Se observa un buen modelo, con un poder de predicción bastante alto.

Curva ROC Y AUC

```
probs_rf <- predict(model_rf2, test1, type = "prob")
ScoreRFauc <- probs_rf[, 2]
rf_roc <- roc(test1$Survived, ScoreRFauc, data = test1)
```

ROC

```
plot(rf_roc, col = "purple", lwd = 3)
```



AUC

```
auc(rf_roc)
```

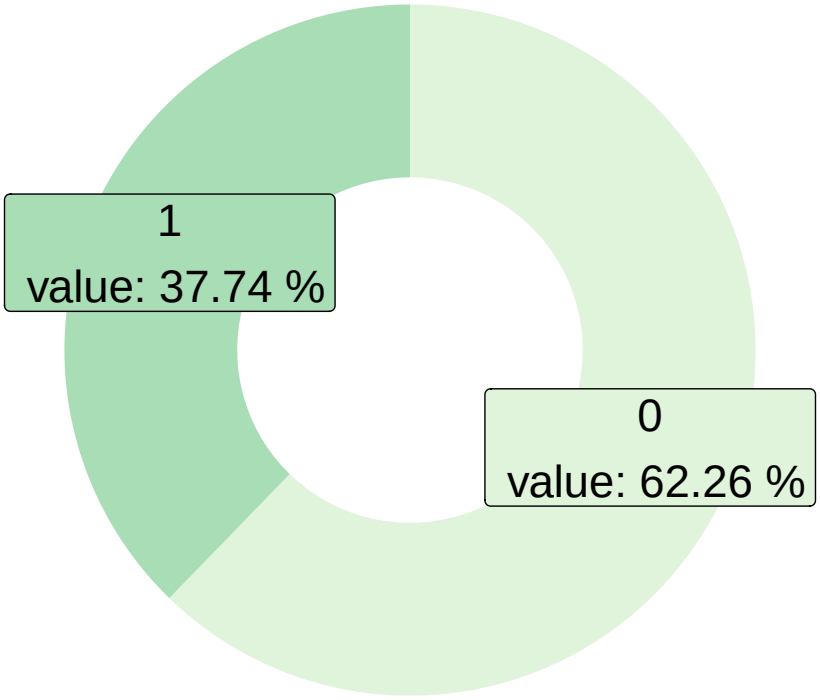
```
## Area under the curve: 0.9287
```

Análisis:

EL AUC es el estadístico que proporciona una medida completa de la capacidad predictiva de un modelo. Como resultado un modelo muy ueno, el modelo discrimina de modo excepcional.

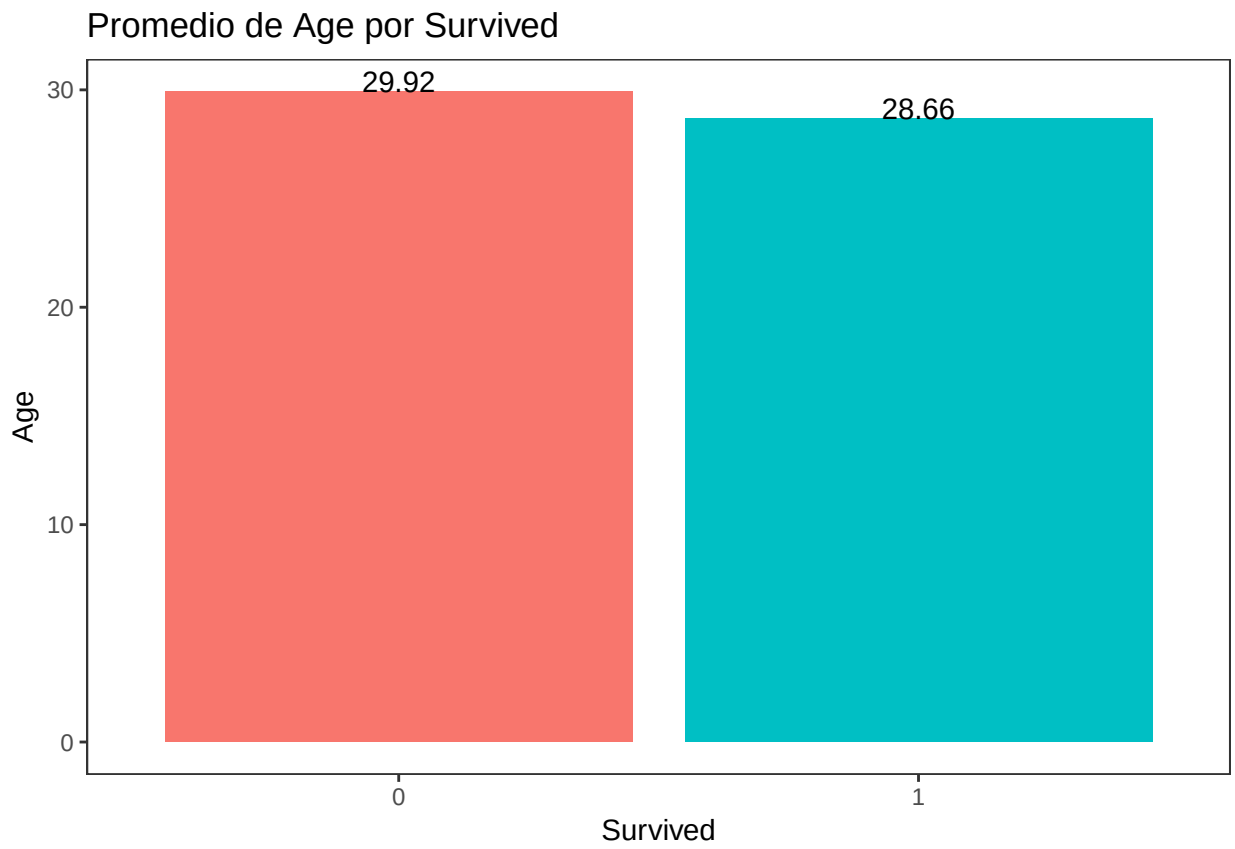
5 Representación de los resultados a partir de tablas y gráficas.

```
data1 <- as.data.frame(round(prop.table(table(data$Survived)) * 100, 2))
data1$fraction <- data1$Freq/sum(data1$Freq)
data1$ymax <- cumsum(data1$fraction)
data1$ymin <- c(0, head(data1$ymax, n = -1))
data1$labelPosition <- (data1$ymax + data1$ymin)/2
data1$label <- paste0(data1$Var1, "\n value: ", data1$Freq)
ggplot(data1, aes(ymax = ymax, ymin = ymin, xmax = 4, xmin = 3, fill = Var1)) +
    geom_rect() + geom_label(x = 3.5, aes(y = labelPosition, label = paste(label,
    "%")), size = 6) + scale_fill_brewer(palette = 4) + coord_polar(theta = "y") +
    xlim(c(2, 4)) + theme_void() + theme(legend.position = "none")
```



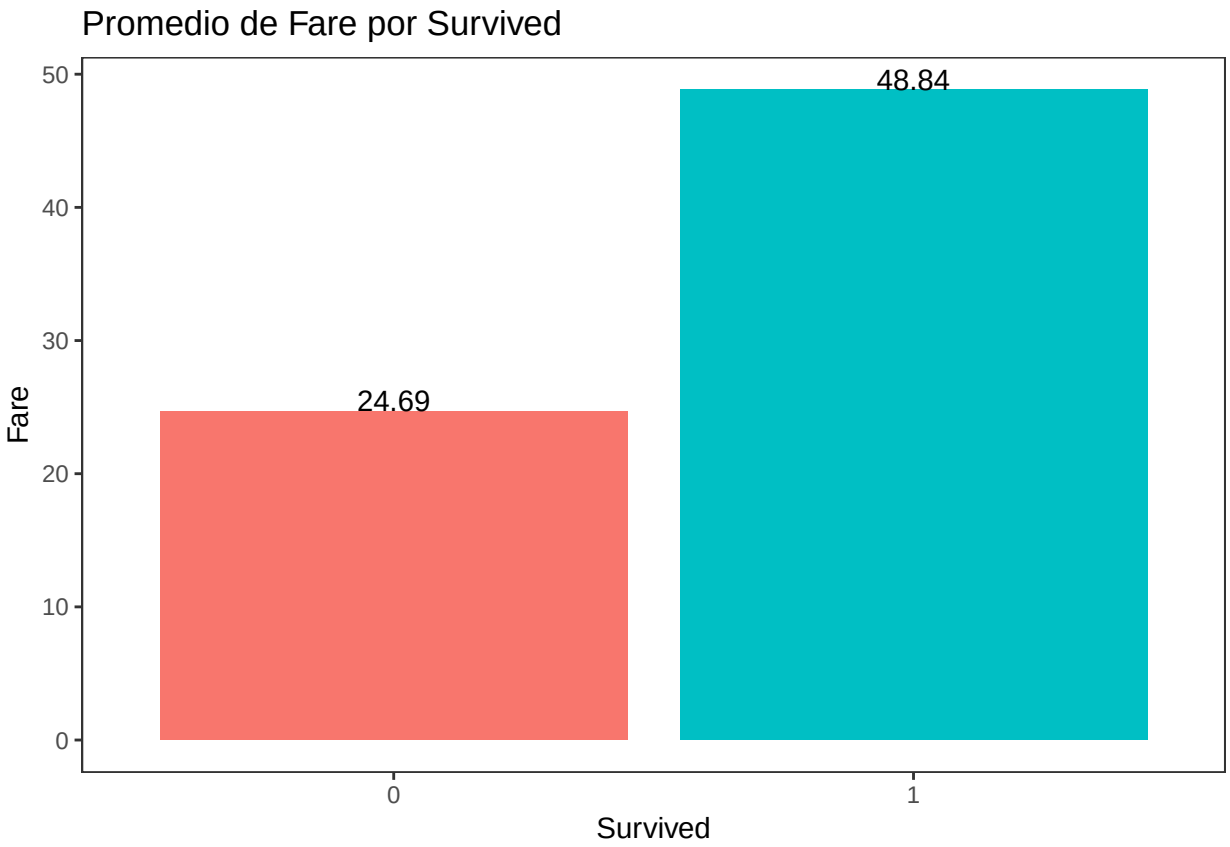
Análisis: El 62.26% de las personas de la data del Titanic no sobrevieron al hundimiento y el 37.74% de las personas si sobrevieron.

```
m_control <- aggregate(Age ~ Survived, FUN = mean, data = data)
ggplot(data = m_control, mapping = aes(x = Survived, y = Age, fill = Survived)) +
    geom_col() + labs(title = "Promedio de Age por Survived", x = "Survived",
    y = "Age", fill = "Survived") + geom_text(aes(label = round(Age, 2)),
    position = position_dodge(1), vjust = 0) + theme_test() + theme(legend.position = "NOne")
```



Análisis: La edad media de los que sobrevivieron aproximadamente es 29 años, la edad media de los que no sobrevivieron es de aproxidamente 30 años.

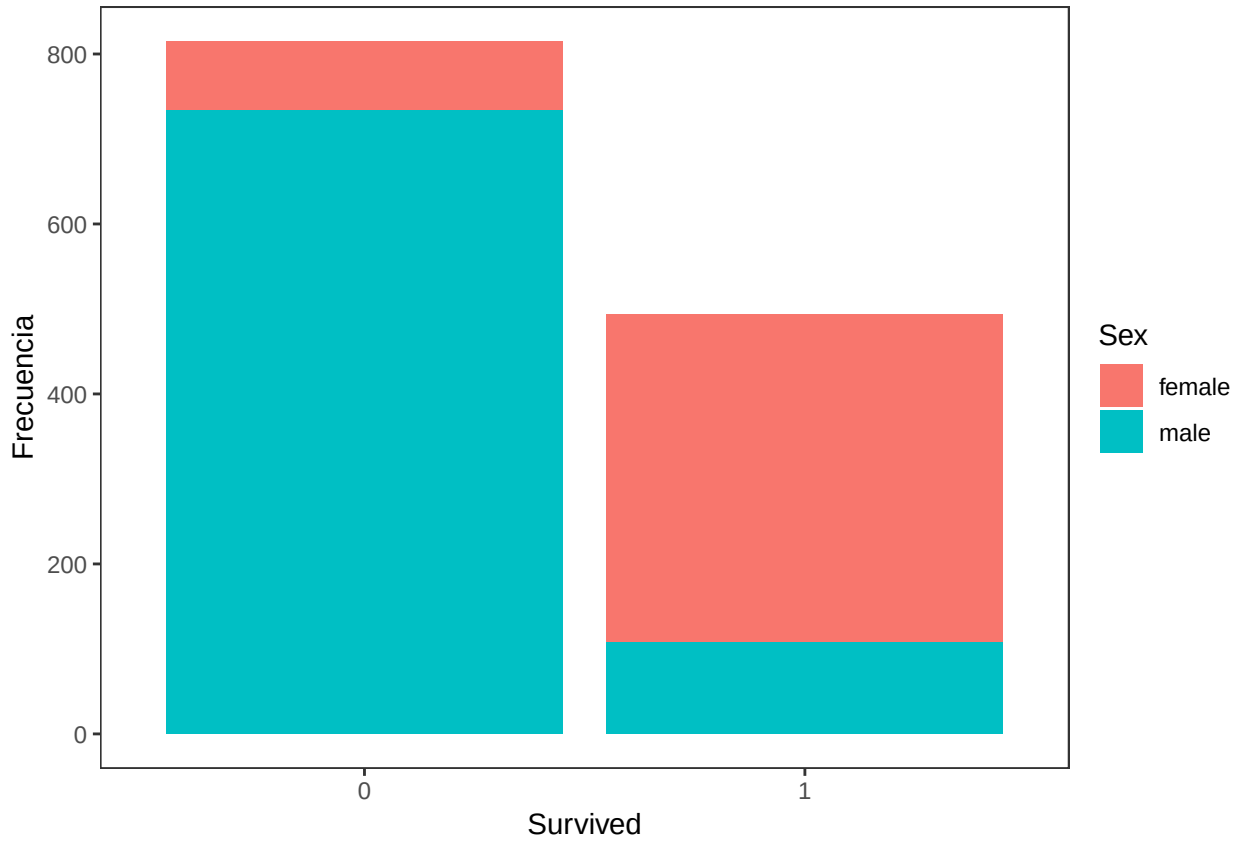
```
m_control2 <- aggregate(Fare ~ Survived, FUN = mean, data = data)
ggplot(data = m_control2, mapping = aes(x = Survived, y = Fare, fill = Survived)) +
    geom_col() + labs(title = "Promedio de Fare por Survived", x = "Survived",
    y = "Fare", fill = "Survived") + geom_text(aes(label = round(Fare,
    2)), position = position_dodge(1), vjust = 0) + theme_test() + theme(legend.position = "NOne")
```



Análisis: El costo promedio del ticket pagado por los que sobrevivieron es de 48.84 libras británicas, El costo promedio del tickets pagado por los que no sobrevivieron es de 24.69 libras británicas.

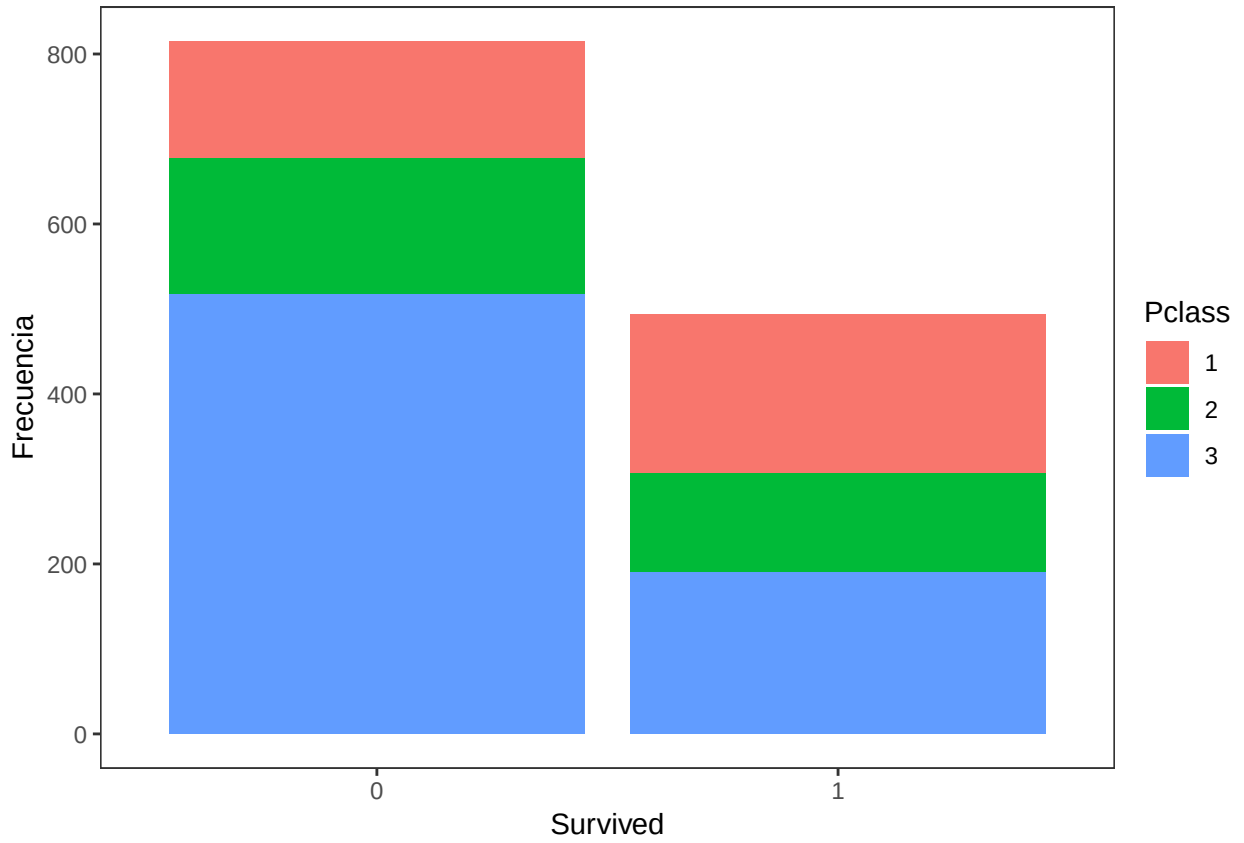
5.0.1

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = Sex)) + labs(x = "Survived",
    y = "Frecuencia", fill = "Sex") + theme_test()
```



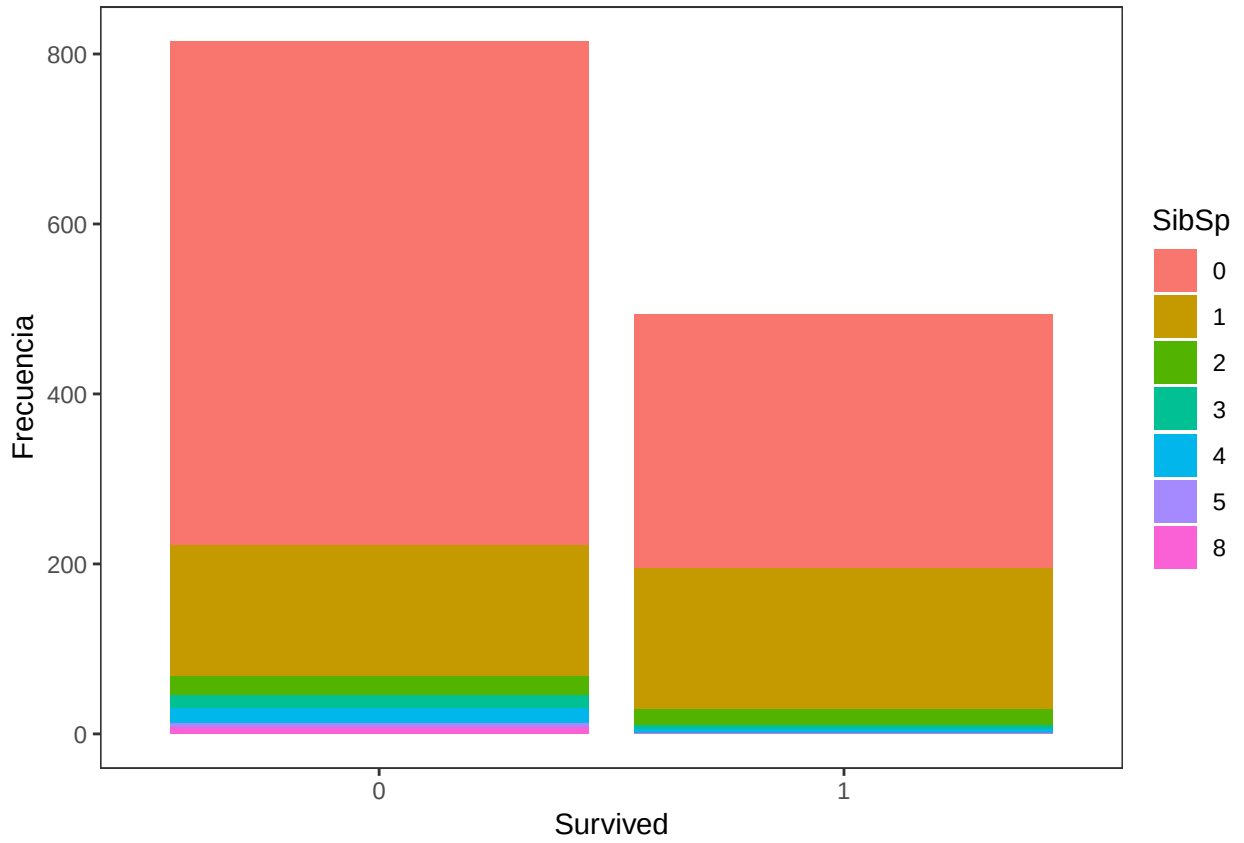
Análisis: El grupo de los que no sobrevivieron estuvo conformado en us mayoría por hombres, y del grupo de los que sobrevivieron estaba conformado más por mujeres, esto tiene sentido ya que le dieron prioridad para subir a los botes a las mujeres.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = Pclass)) + labs(x = "Survived",
    y = "Frecuencia", fill = "Pclass") + theme_test()
```



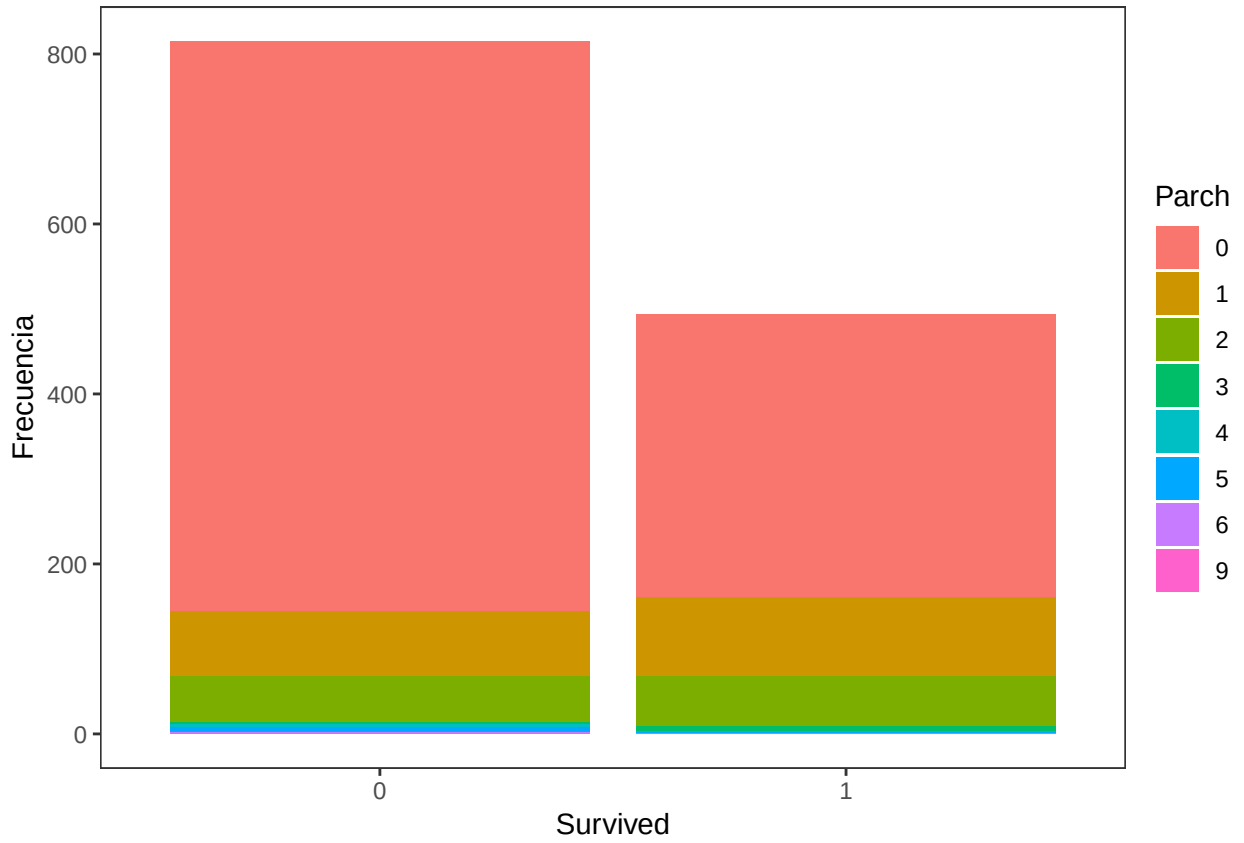
Análisis: El grupo de los que no sobrevivieron aproximadamente el 75% eran de la tercera clase, en el grupo de los que sobrevivieron predominan los de primera clase y tercera clase.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = factor(SibSp))) + labs(x = "Survived",
    y = "Frecuencia", fill = "SibSp") + theme_test()
```



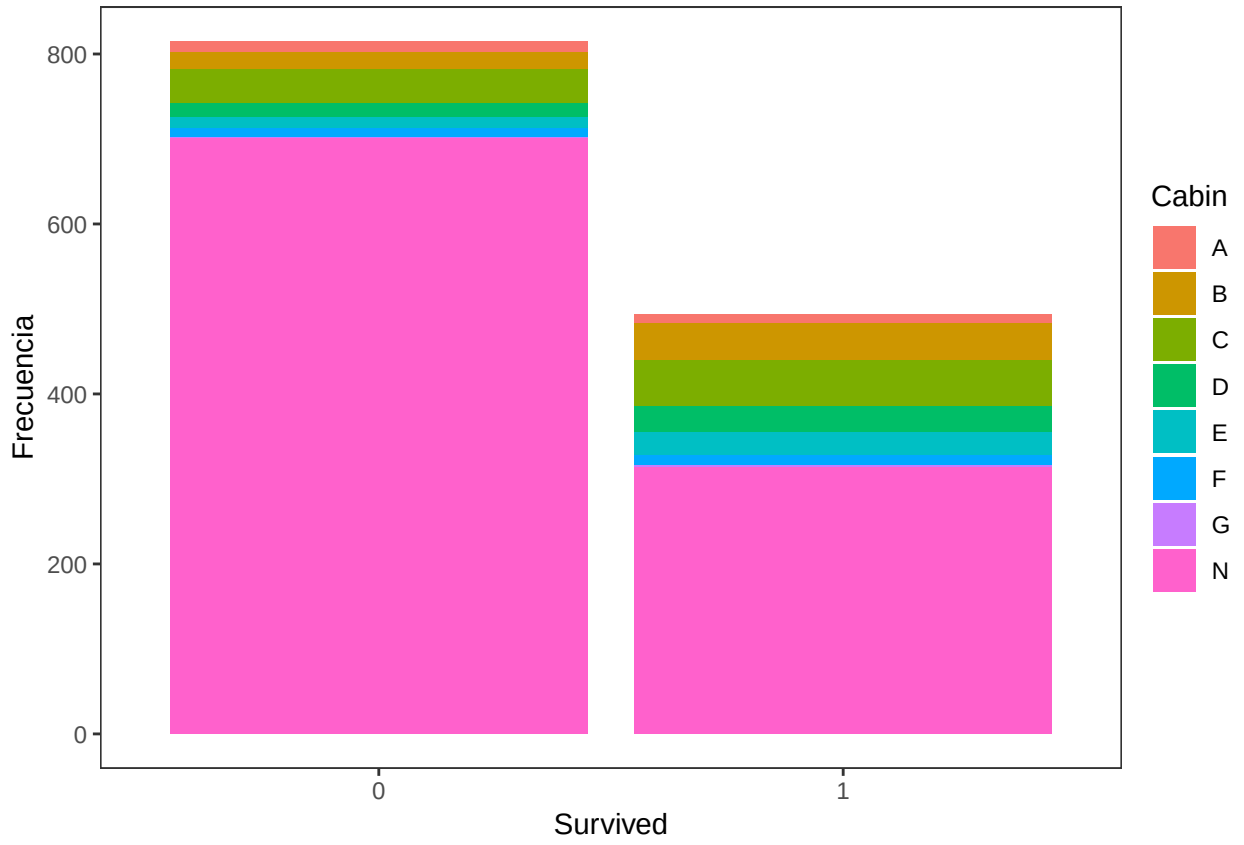
Análisis: En ambos grupos prodominan personas que no tenian hermanos, hermanas, esposos ni esposas a bordo.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = factor(Parch))) + labs(x = "Survived",
    y = "Frecuencia", fill = "Parch") + theme_test()
```

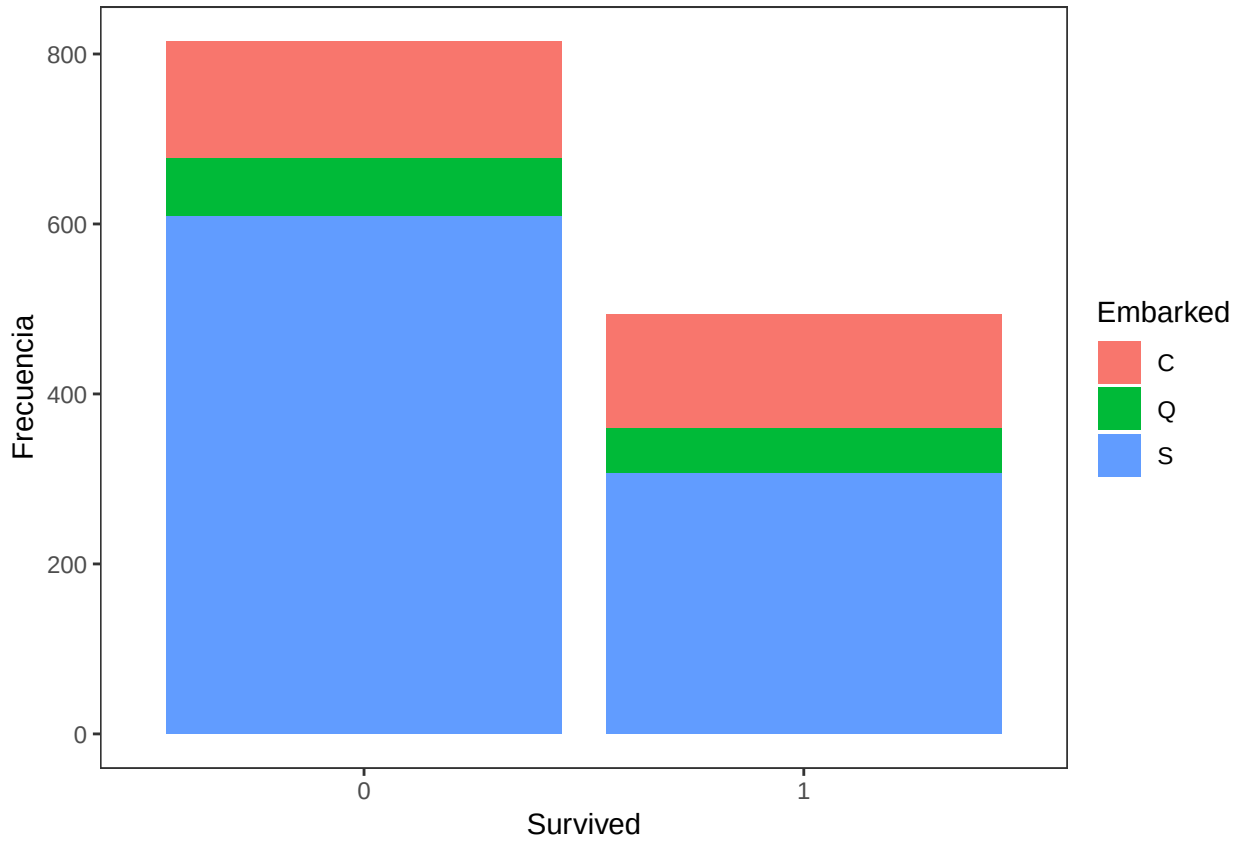


Análisis: En ambos grupos predominan personas que no tenian padres ni hijos a bordo.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = Cabin)) + labs(x = "Survived",
    y = "Frecuencia", fill = "Cabin") + theme_test()
```

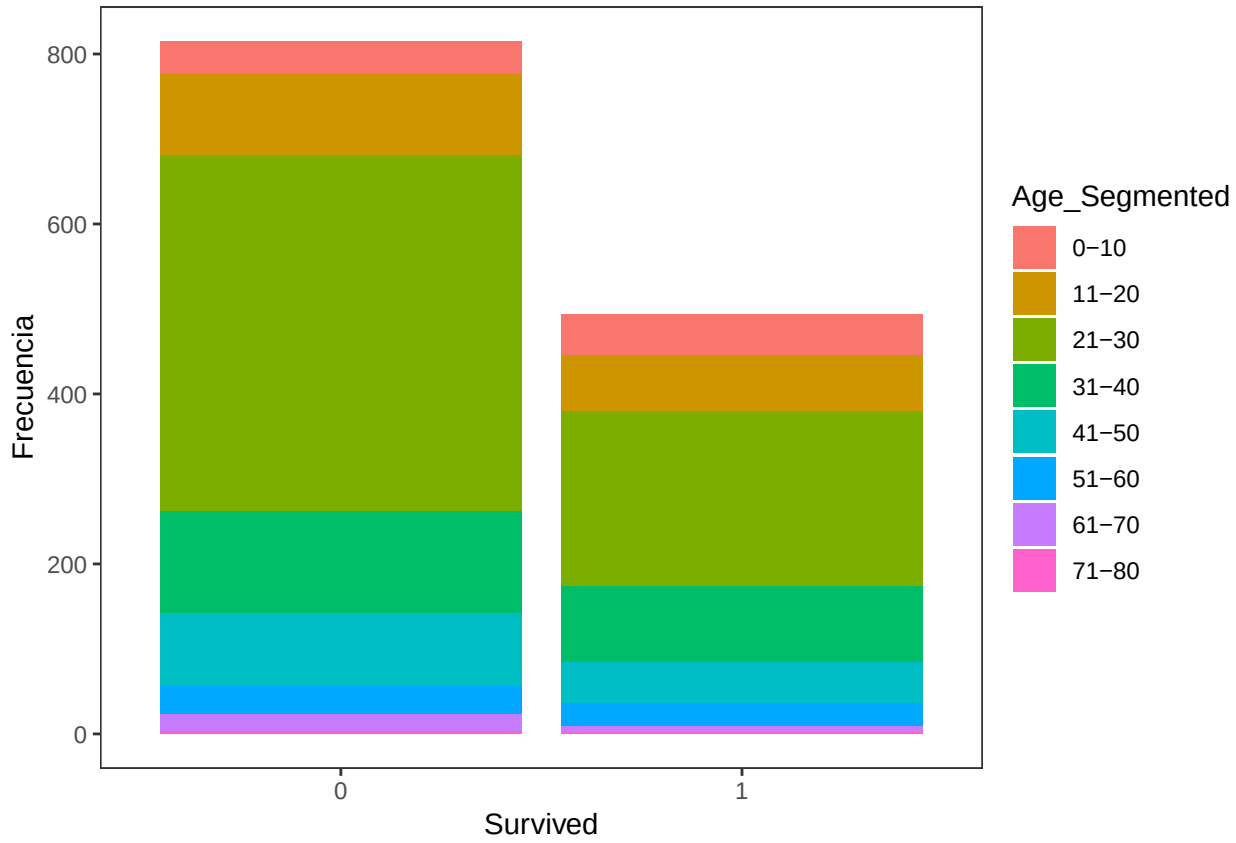



```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = Embarked)) + labs(x = "Survived",
    y = "Frecuencia", fill = "Embarked") + theme_test()
```



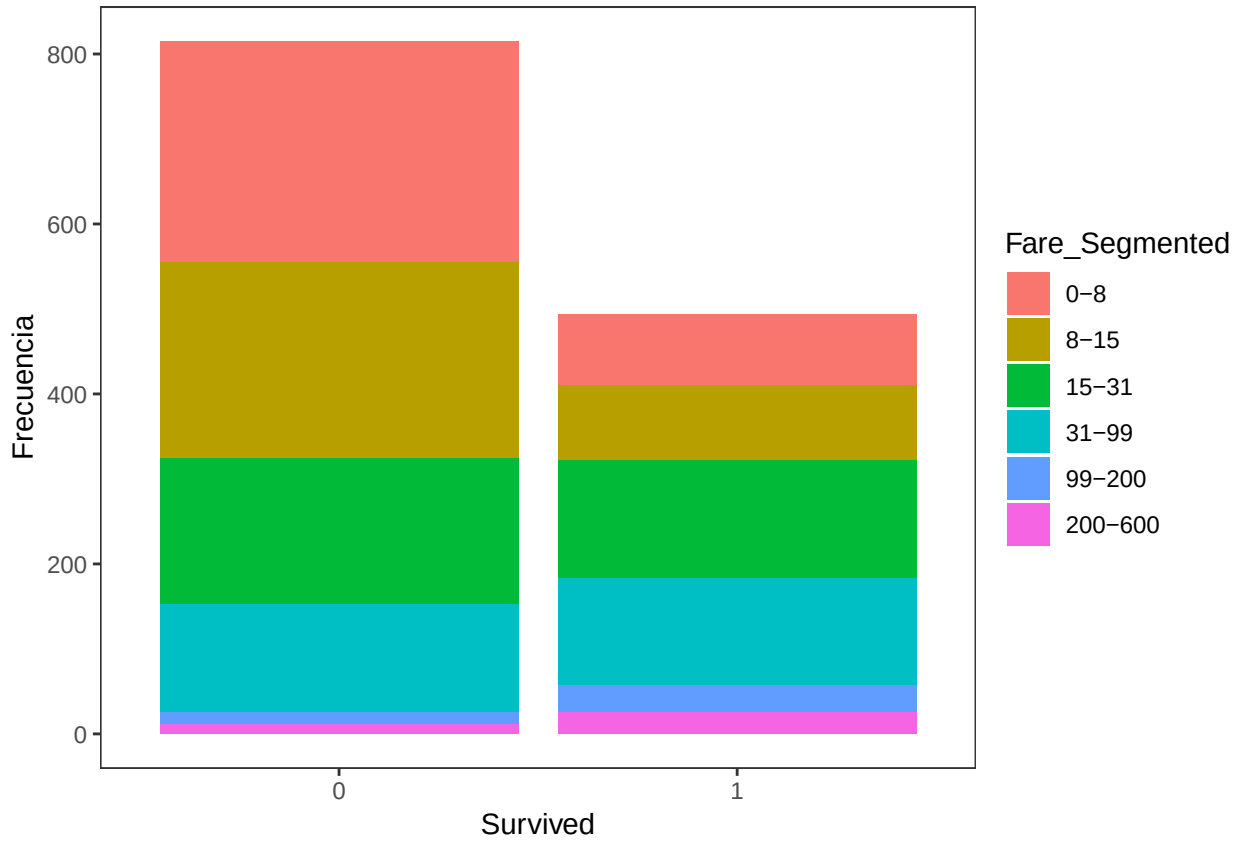
Análisis: En ambos grupos predominan personas que embarcaron en Southampton.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = Age_Segmented)) + labs(x = "Survived",
    y = "Frecuencia", fill = "Age_Segmented") + theme_test()
```



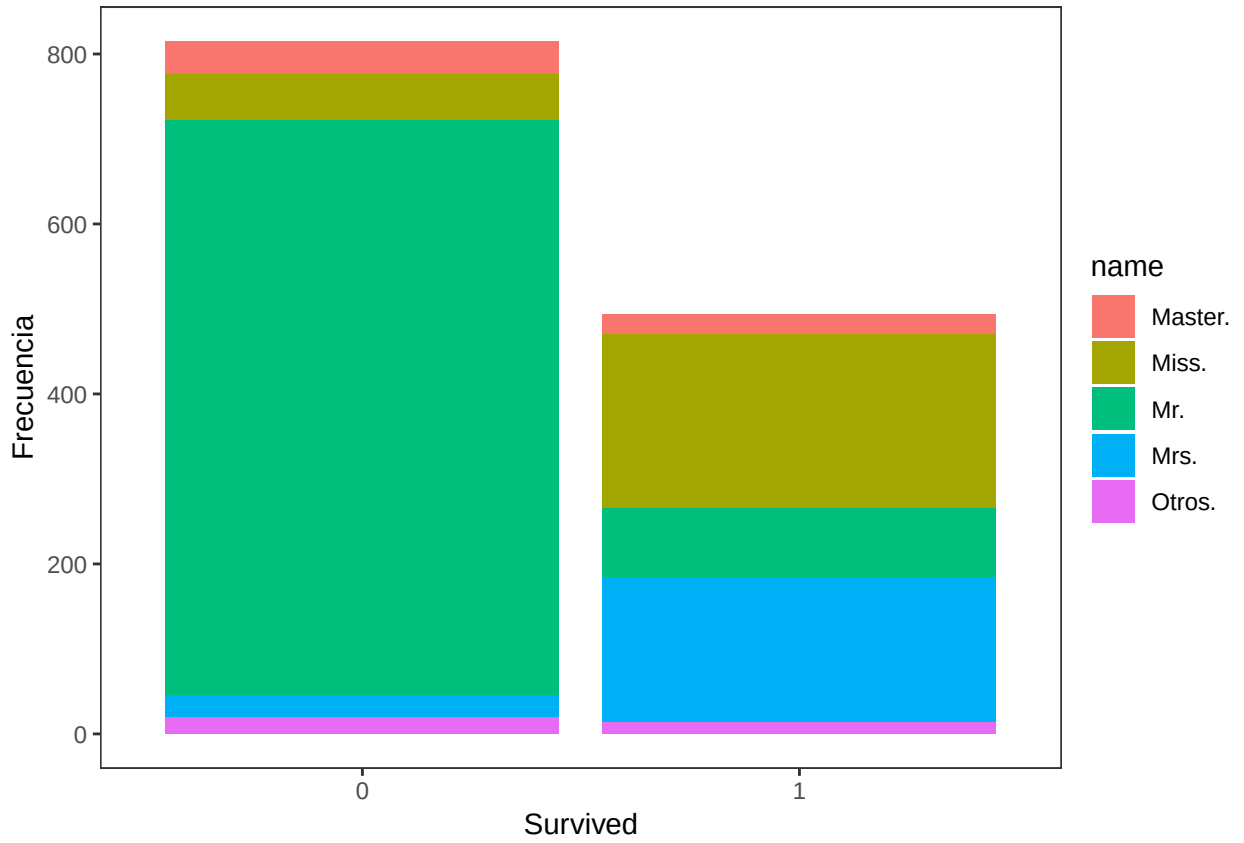
Análisis: En ambos grupos predominan personas entre 21-30 años.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = Fare_Segmented)) + labs(x = "Survived",
    y = "Frecuencia", fill = "Fare_Segmented") + theme_test()
```



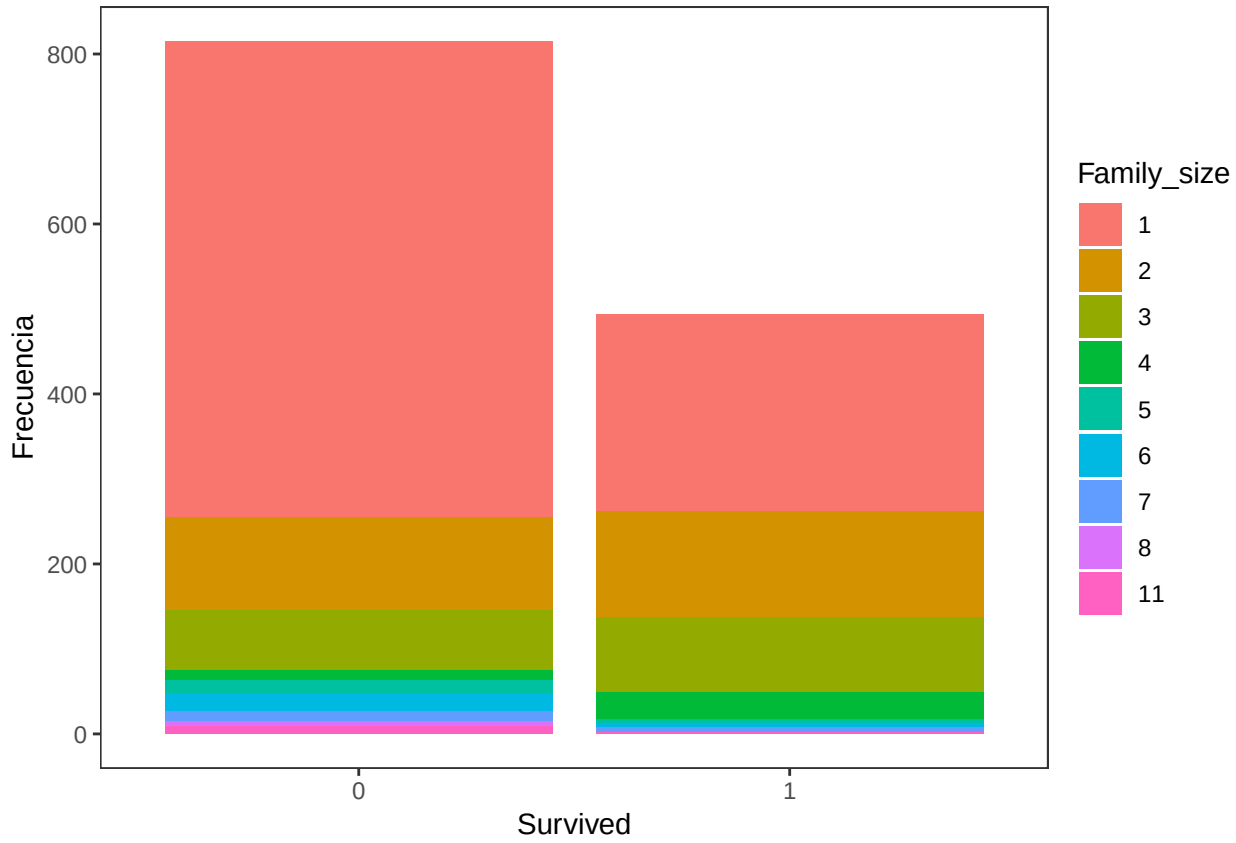
Análisis: En el grupo de los que no sobrevivieron predominan los que pagaron por el ticket entre 0 a 15 libras británicas, y en el grupo de los que sobrevivieron predominan los que pagaron entre 15 y 99 libras británicas.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = Name)) + labs(x = "Survived",
    y = "Frecuencia", fill = "name") + theme_test()
```



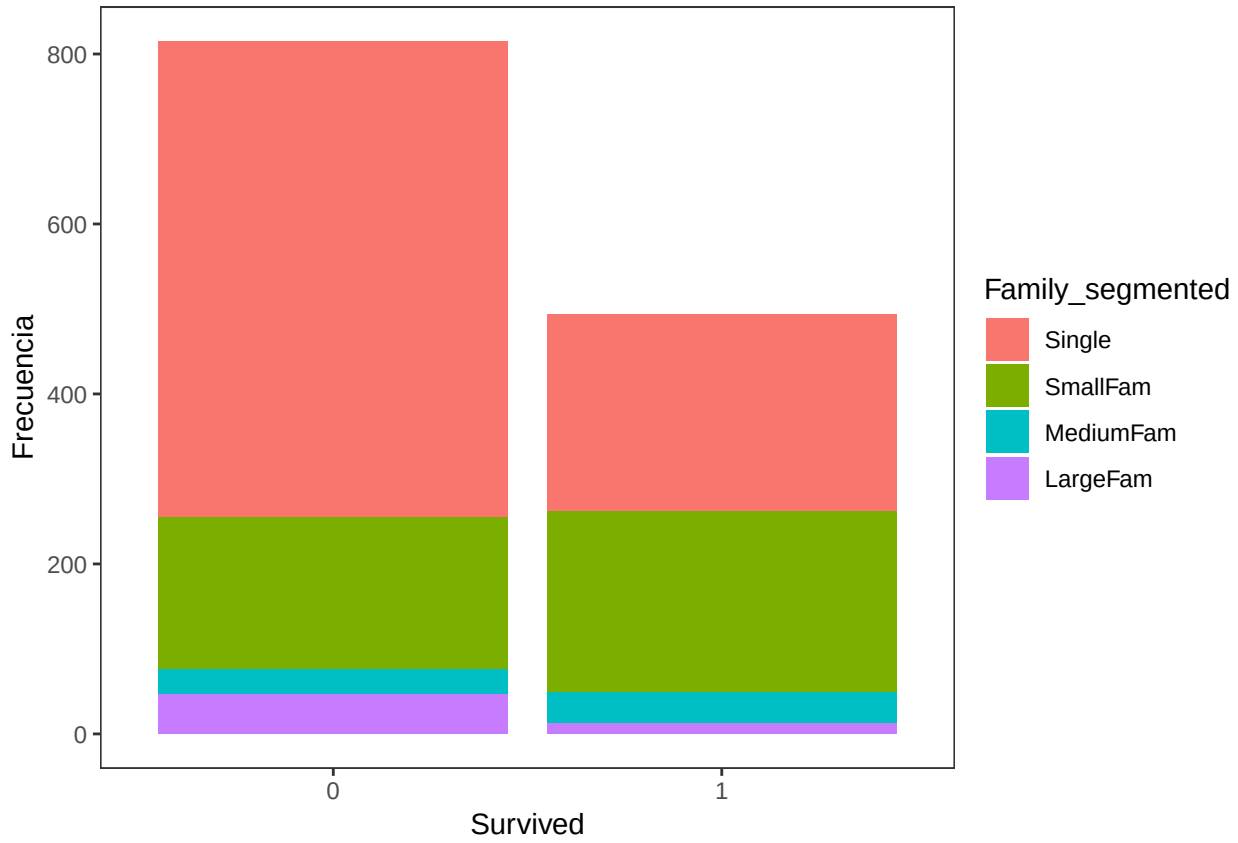
Análisis: En el grupo de los que no sobrevivieron estuvo compuesto en su mayoría por Señores. Y en el grupo de los que sobrevivieron estuvo compuesto por Señoritas y señoras.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = factor(Family_Size))) +
    labs(x = "Survived", y = "Frecuencia", fill = "Family_size") + theme_test()
```



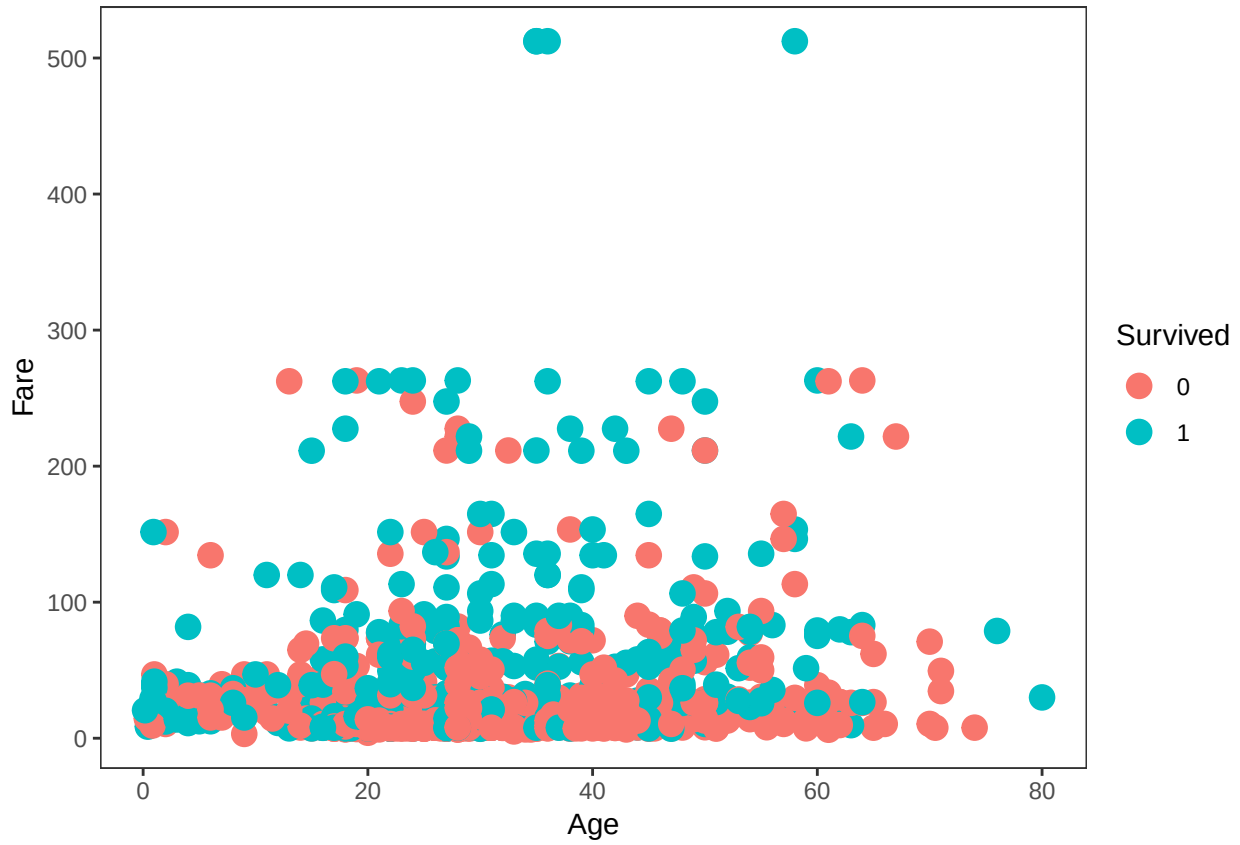
Análisis: En ambos grupos predominan personas que viajaba solas.

```
ggplot(data, aes(Survived)) + geom_bar(aes(fill = Family_Segmented)) +
    labs(x = "Survived", y = "Frecuencia", fill = "Family_segmented") +
    theme_test()
```

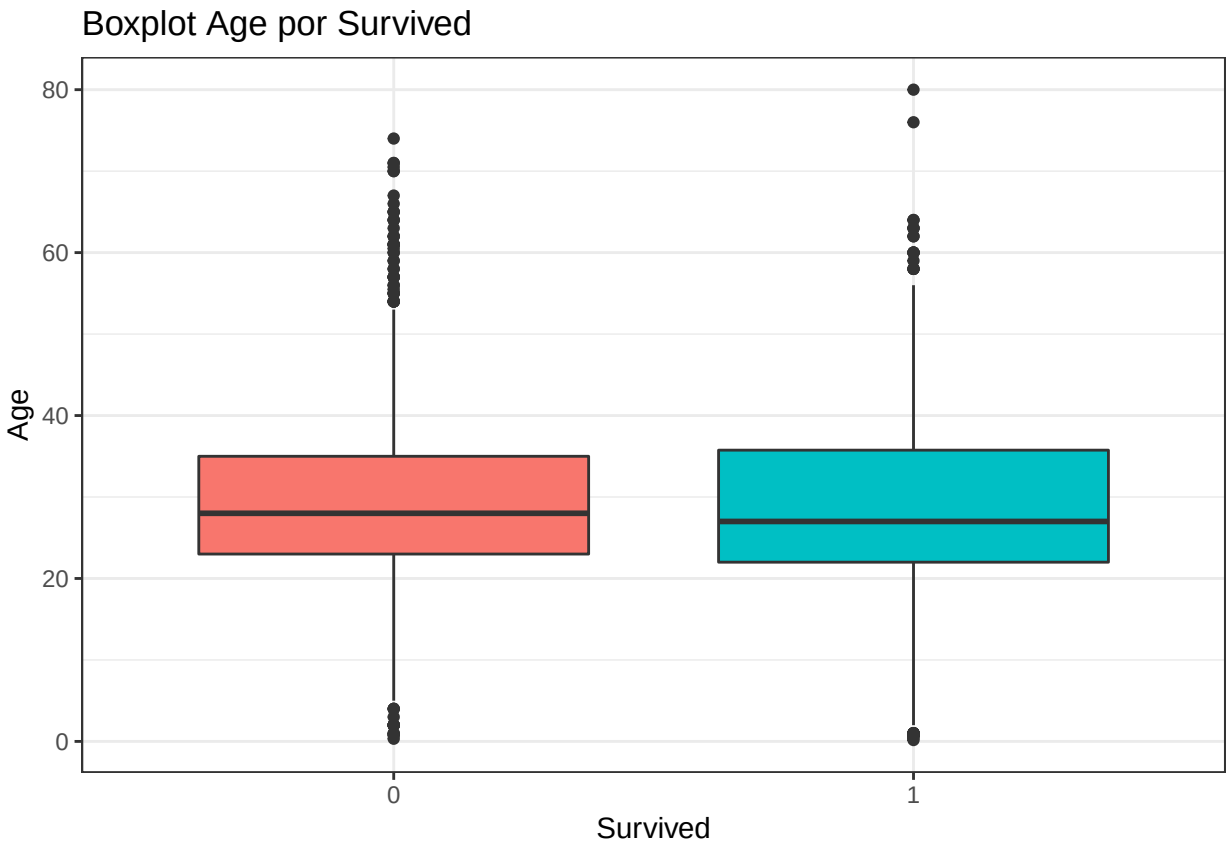


Análisis: En ambos grupos predominan personas personas que viajan solas y de Familia pequeña.

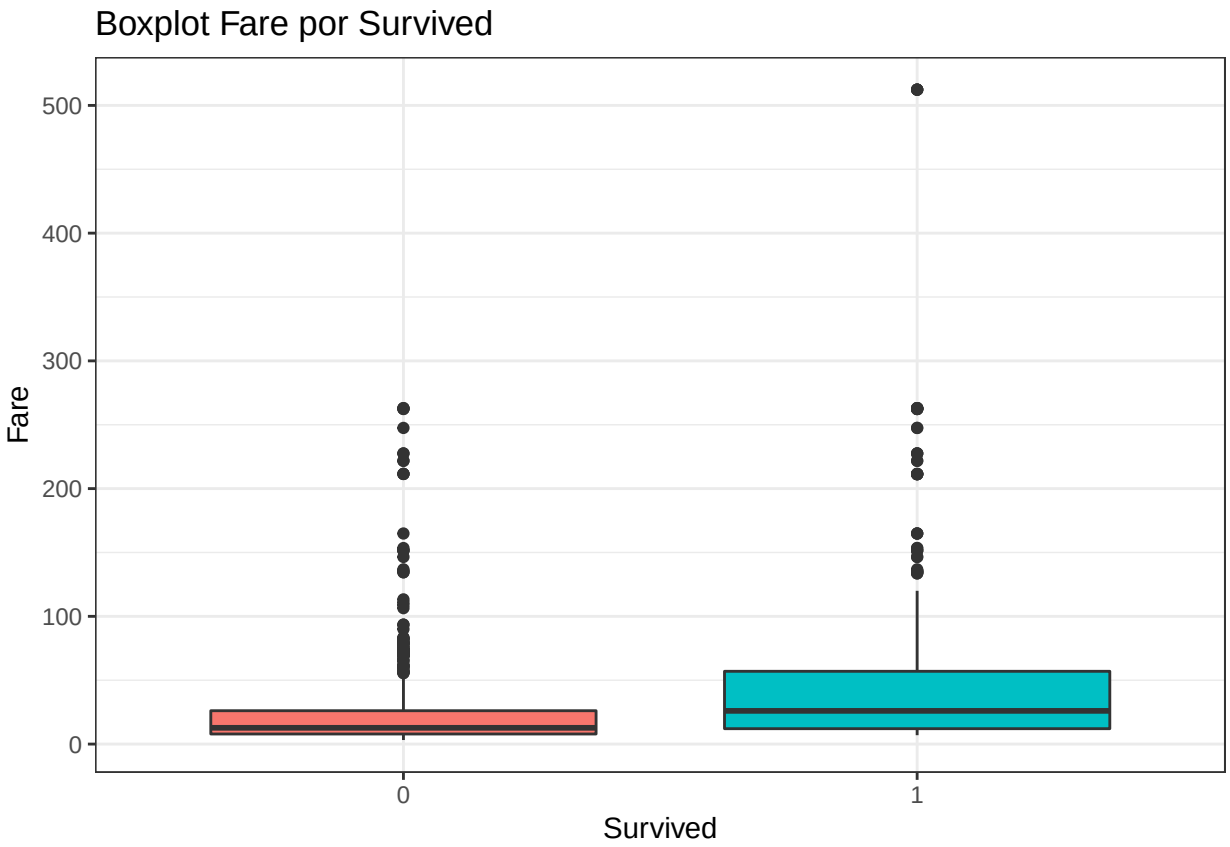
```
ggplot(data, aes(x = Age, y = Fare, color = Survived)) + geom_point(size = 4) +
    theme_test()
```




```
ggplot(data, aes(factor(Survived), Age, fill = Survived)) + geom_boxplot(aes(fill = (Survived))) +
    labs(title = "Boxplot Age por Survived", x = "Survived", y = "Age") +
    theme_bw() + theme(legend.position = "None")
```




```
ggplot(data, aes(factor(Survived), Fare, fill = Survived)) + geom_boxplot(aes(fill = (Survived))) +
    labs(title = "Boxplot Fare por Survived", x = "Survived", y = "Fare") +
    theme_bw() + theme(legend.position = "None")
```



6 Resolución del problema.

¿Cuáles son las conclusiones?

- Se concluye que las variable categóricas muestran dependencia con la variable survived.
- No existe diferencia en las edades de los que sobrevivieron y los que no.
- Los que murieron fueron más hombres y las que sobreviieron fueron más mujeres.
- Viajaban más personas solas, y de la tercera clase.
- La mayoría de las personas que viajaban se embarcaron en Southampton.
- No todas las variables de la data eran relevantes para clasificar a los que si sobrevieron de los que no.

¿Los resultados permiten responder al problema?

Si, el objetivo del trabajo era predecir quienes sobrevivian y quienes no, además de describir ambos grupos y determinar que variables del contexto social y/o económico son importantes o relevantes a la hora de predecir o determinar ambos grupos. Las variables más discriminantes son el título de la persona, la cabina que ocupada, donde se embarcó, si viajaba sola, la clase en la que viajaba, el valor que pago por el ticket del viaje, el Sexo de la persona.