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 del pasajero
- **Sex** Sexo del pasajero
- **Age** Edad del pasajero
- **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*). 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:

```
# install.packages( 'ggplot2', repos =
# c('http://rstudio.org/_packages', 'http://cran.rstudio.com') )
# library('ggplot2')
if (!require("ggplot2")) install.packages("ggplot2")
library("ggplot2")
if (!require("formatR")) install.packages("formatR")
library("formatR")
if (!require("arules")) install.packages("arules")
library(arules)
```

```r
if (!require("missForest")) install.packages("missForest")
library(missForest)

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:

```r
# 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$tipo <- "test"
test$Survived <- NA

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

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.

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

```
## 'data.frame':    1309 obs. of  13 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 1 2 2 ...
##  $ Pclass     : Factor w/ 3 levels "1","2","3": 3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 1307 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 417 58
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : Factor w/ 929 levels "110152","110413",..: 524 597 670 50 473 276 86 396 345 133 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
```

```
##  $ Cabin      : Factor w/ 187 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
##  $ tipo       : Factor w/ 2 levels "entrenamiento",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Al verificar el resumen del data set set observa que en el conjunto de entrenamiento existen 549 personas que no sobrevivieron y 342 que sí, las demás 418 personas corresponden al conjunto de validación que no disponen de este valor. El número de hombres en el dataset es casi el doble que el de mujeres con 843 hombres y 466 mujeres. La edad mínima de los pasajeros es de 0.17 años, la máxima de 80 y la media de 29 años, aunque existen 263 registros sin edad. El número máximo de hermanos y parejas que tiene un pasajero a bordo es 8, mientras que la media es 0.4. Sin embargo, el número máximo de padres e hijos a bordo que tiene un pasajero es de 9 y la media es de 0.3. El valor máximo que han pagado para embarcar es de 512 libras británicas, la media es de 33 y el mínimo de 0. La mayoría de pasajeros embarcaron en Southampton (914), seguido de Queenstown (123) y de Cherbourg (270), existen dos registros sin datos.

```
summary(data)
```

```
##   PassengerId   Survived   Pclass                                    Name
##  Min.   :   1   0   :549   1:323   Connolly, Miss. Kate                  :   2
##  1st Qu.: 328   1   :342   2:277   Kelly, Mr. James                      :   2
##  Median : 655   NA's:418   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
```

Analizado los datos 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 Fare 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 utilizará el algoritmo MissForest para imputar la información faltante.

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

```
## Survived   Pclass     Name      Sex      Age    SibSp    Parch     Fare
##      418        0        0        0      263        0        0        1
##    Cabin Embarked     tipo
##     1014        2        0
```

3.1 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.

```
# 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.

```
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)
```

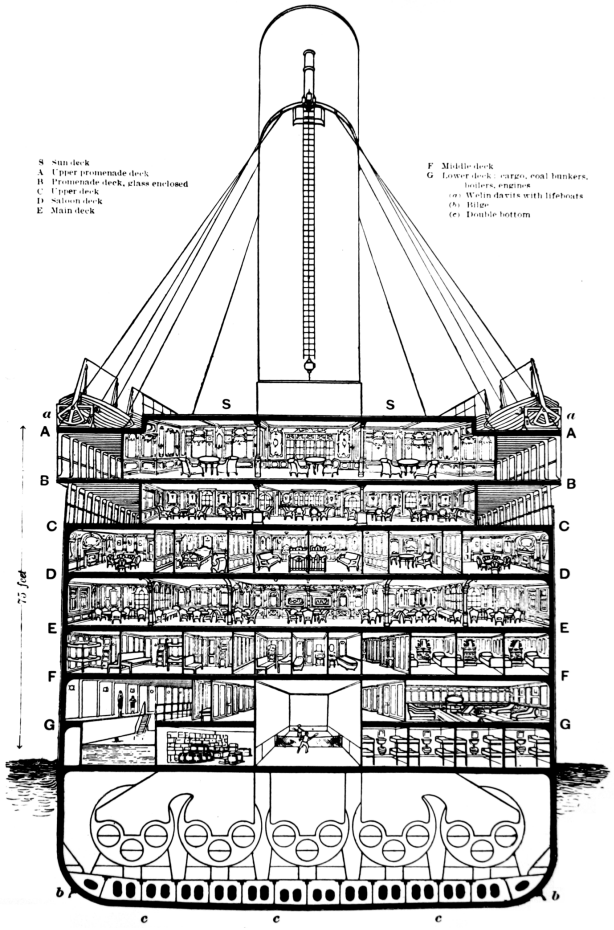
```
## Master.   Miss.     Mr.    Mrs.  Otros.
##      61     260     757     197      34
```

3.2 Tratamiento de la variable Cabin

Cada registro de la variable Cabin está compuesto de la categoría del camarote (primera letra) seguido de un número, para el análisis de los datos solo es necesario la categoría del camarote.

```
# cortamos la primera letra de cada registro.
data$Cabin <- as.character(data$Cabin)
data$Cabin <- as.factor(substr(data$Cabin, start = 1, stop = 1))
```

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



3.3 Preparación de variables con datos faltantes

La preparación de las variables con datos faltantes consiste en reemplazar los valores faltantes y vaciós por NA con el objetivo de indicar al algoritmo de imputación cuales son los valores faltantes.

Variable Fare

En la variable Fare se reemplazan por NA 18 valores vacíos y con valor cero, se incluye el valor cero ya que todos los pasajeros deben haber pagado por su pasaje a bordo.

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

```
##
## FALSE  TRUE
##  1291    18
```

```
data$Fare[is.na(data$Fare) | data$Fare == 0] <- NA
```

Variable Cabin

En la variable Cabin se reemplaza por NA 1019 registros, entre valores vacíos, no existentes y los que corresponden a la categoría T, ya que tiene un solo registro.

```
table(data$Cabin == "T" | is.na(data$Fare) | data$Cabin == "")
```

```
##
## FALSE  TRUE
##   290  1019
```

```
# Reemplazamos los valores vacíos por N
data$Cabin[is.na(data$Cabin) | data$Cabin == "" | data$Cabin == "T"] <- NA
```

Variable Age

En la variable Age se reemplazan 272 valores vacíos por NA

```
table(is.na(data$Age) | data$Age == "")
```

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

```
data$Age[is.na(data$Age)] <- NA
```

Variable Embarked

En la variable Embarked se reemplaza 2 valores vaciós por NA

```
table(data$Embarked == "" | is.na(data$Embarked))
```

```
##
## FALSE  TRUE
##  1307     2
```

```
data$Embarked[data$Embarked == ""] <- NA
```

3.4 Imputar Datos Faltantes

Para completar los datos faltantes de las variables se utiliza MissForest, un algoritmo de imputación para datos faltantes o perdidos. El algoritmo para cada variable con datos faltantes crea un Random Forest y predice los valores faltantes, esto se realiza en varias iteraciones hasta cruzar un umbral de aceptación.

Para el algoritmo se utilizan las todas las variables a excepción de survived y el tipo de registro.

```
datatest <- data
variablesMiss <- missForest(datatest[c(2:9, 11)])
```

```
##   missForest iteration 1 in progress...done!
##   missForest iteration 2 in progress...done!
##   missForest iteration 3 in progress...done!
##   missForest iteration 4 in progress...done!
##   missForest iteration 5 in progress...done!
##   missForest iteration 6 in progress...done!
##   missForest iteration 7 in progress...done!
##   missForest iteration 8 in progress...done!
##   missForest iteration 9 in progress...done!
##   missForest iteration 10 in progress...done!
```

```r
data$Fare <- variablesMiss$ximp$Fare
data$Age <- variablesMiss$ximp$Age
data$Cabin <- variablesMiss$ximp$Cabin
data$Embarked <- variablesMiss$ximp$Embarked
```

Se comprueba nuevamente si existen datos faltantes.

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

```
## Survived   Pclass      Sex      Age    SibSp    Parch     Fare    Cabin
##      418        0        0        0        0        0        0        0
## Embarked     tipo     Name
##        0        0        0
```

Verificamos los datos imputados en la variables Fare, se observa que las personas que menos pagan están en clase 3.

```r
data[which(is.na(datatest$Fare)), ]
```

```
##      Survived Pclass  Sex      Age SibSp Parch      Fare Cabin Embarked
## 180         0      3 male 36.00000     0     0  9.484584     F        S
## 264         0      1 male 40.00000     0     0 46.860049     B        S
## 272         1      3 male 25.00000     0     0  8.857656     F        S
## 278         0      2 male 38.01776     0     0 14.212903     F        S
## 303         0      3 male 19.00000     0     0  8.972886     F        S
## 414         0      2 male 38.01776     0     0 14.212903     F        S
## 467         0      2 male 38.01776     0     0 14.212903     F        S
## 482         0      2 male 38.01776     0     0 14.212903     F        S
## 598         0      3 male 49.00000     0     0  9.277237     F        S
## 634         0      1 male 39.02387     0     0 40.239197     A        S
## 675         0      2 male 38.01776     0     0 14.212903     F        S
## 733         0      2 male 38.01776     0     0 14.212903     F        S
## 807         0      1 male 39.00000     0     0 39.867714     A        S
## 816         0      1 male 43.31445     0     0 42.674592     B        S
## 823         0      1 male 38.00000     0     0 37.280478     A        S
## 1044     <NA>      3 male 60.50000     0     0  9.185228     F        S
## 1158     <NA>      1 male 39.02387     0     0 40.239197     A        S
## 1264     <NA>      1 male 49.00000     0     0 54.526464     B        S
##                tipo   Name
## 180  entrenamiento    Mr.
## 264  entrenamiento    Mr.
## 272  entrenamiento    Mr.
## 278  entrenamiento    Mr.
## 303  entrenamiento    Mr.
## 414  entrenamiento    Mr.
## 467  entrenamiento    Mr.
## 482  entrenamiento    Mr.
## 598  entrenamiento    Mr.
## 634  entrenamiento    Mr.
## 675  entrenamiento    Mr.
## 733  entrenamiento    Mr.
## 807  entrenamiento    Mr.
```

```
## 816  entrenamiento    Mr.
## 823  entrenamiento Otros.
## 1044          test    Mr.
## 1158          test    Mr.
## 1264          test    Mr.
```

Verificamos los datos imputados en la variables Age

```
head(data[which(is.na(datatest$Age)), ])
```

```
##    Survived Pclass    Sex      Age SibSp Parch    Fare Cabin Embarked
## 6         0      3   male 31.54661     0     0  8.4583     F        Q
## 18        1      2   male 32.31483     0     0 13.0000     F        S
## 20        1      3 female 35.45420     0     0  7.2250     F        C
## 27        0      3   male 27.33562     0     0  7.2250     F        C
## 29        1      3 female 22.64430     0     0  7.8792     F        Q
## 30        0      3   male 28.07156     0     0  7.8958     F        S
##             tipo  Name
## 6  entrenamiento   Mr.
## 18 entrenamiento   Mr.
## 20 entrenamiento  Mrs.
## 27 entrenamiento   Mr.
## 29 entrenamiento Miss.
## 30 entrenamiento   Mr.
```

Ahora comprobamos los datos en Cabin, donde más datos faltantes habían y se observa que la mayoría de valores faltantes se ha completado con los camarotes F

```
data$Cabin = droplevels(data$Cabin)
summary(data$Cabin)
```

```
##   A   B   C   D   E   F   G
##  41  73 108  70  70 882  65
```

Verificamos los datos imputados en la variables Embarked, se observa que en los dos casos se ha imputado por C.

```
head(data[which(is.na(datatest$Embarked)), ])
```

```
##     Survived Pclass    Sex Age SibSp Parch Fare Cabin Embarked          tipo
## 62         1      1 female  38     0     0   80     B        C entrenamiento
## 830        1      1 female  62     0     0   80     B        C entrenamiento
##      Name
## 62  Miss.
## 830  Mrs.
```

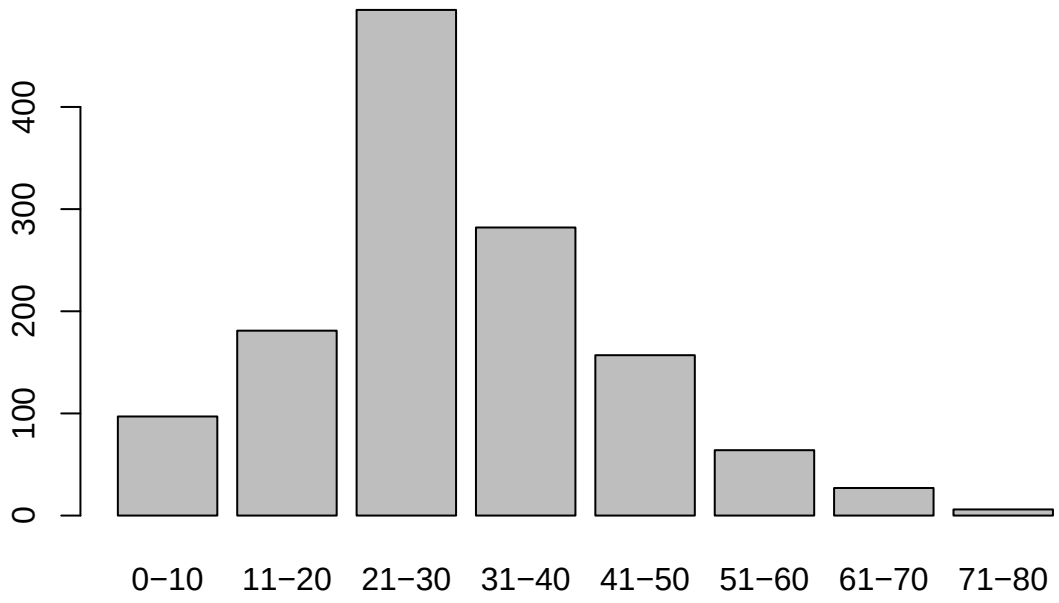
3.5 Segmentación de la variable Age

A continuación se categoriza la edad de los pasajeros 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   28.39   29.80   37.09   80.00
```

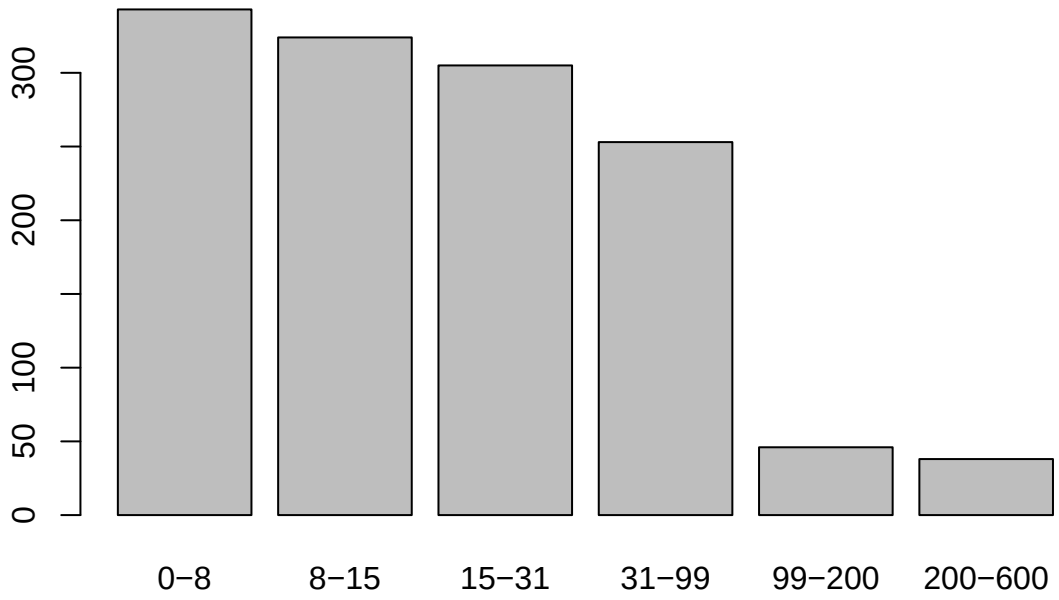
```r
data$Age_Segmented <- cut(data$Age, breaks = c(0, 10, 20, 30, 40, 50, 60,
    70, 80), labels = c("0-10", "11-20", "21-30", "31-40", "41-50", "51-60",
    "61-70", "71-80"))
plot(data$Age_Segmented)
```



3.6 Segmentación variable Fare

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].

```r
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)
```



3.7 Segmentación de la familia del pasajero

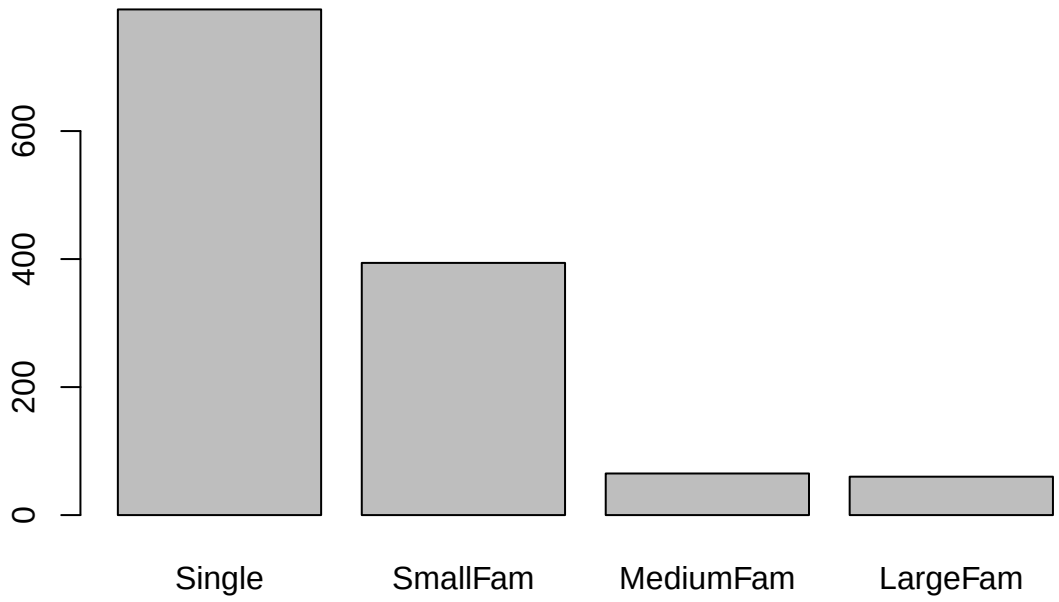
El pasajero cuenta con las variables Parch y SibSp que juntas indican el tamaño de la familia a bordo. A continuación estás variables se juntan y categorizan en los grupos:

- 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)
```



3.8 Identificación y tratamiento de valores extremos

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

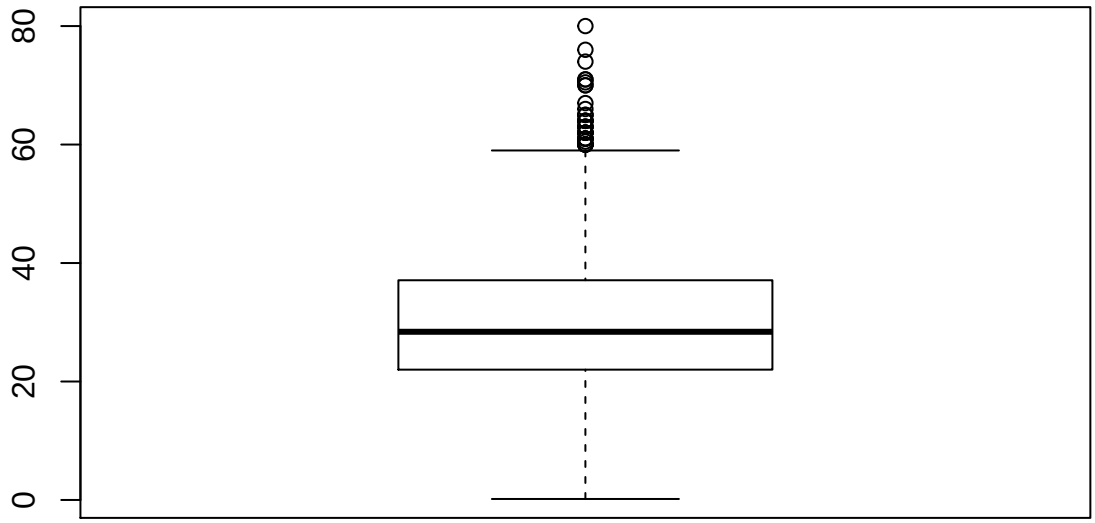
Iniciamos con la variable Age, obtenemos los valores únicos y los ordenamos para observar los valores que la variable puede tomar, en los resultados a continuación se constata que la edad mínima es de 0.17 años y la máxima de 80. Inicialmente se podría pensar que los valores menores a cero son un error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

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

```
##   [1]  0.170000  0.330000  0.420000  0.670000  0.750000  0.830000  0.920000
##   [8]  1.000000  2.000000  3.000000  4.000000  4.454604  5.000000  5.134351
##  [15]  5.434274  6.000000  6.355689  6.599277  6.904976  7.000000  7.226589
##  [22]  8.000000  8.008819  9.000000 10.000000 11.000000 11.500000 11.870444
##  [29] 12.000000 13.000000 14.000000 14.017499 14.500000 15.000000 15.448697
##  [36] 16.000000 16.039633 16.477991 17.000000 17.353559 18.000000 18.500000
##  [43] 19.000000 19.022576 19.202386 19.494641 19.754321 19.879047 20.000000
##  [50] 20.500000 21.000000 21.602842 22.000000 22.145041 22.264766 22.271519
##  [57] 22.500000 22.618394 22.644303 22.647682 23.000000 23.102044 23.223609
##  [64] 23.236668 23.500000 23.626711 24.000000 24.279498 24.447712 24.500000
##  [71] 24.505814 25.000000 25.801199 26.000000 26.024084 26.496420 26.500000
##  [78] 27.000000 27.082276 27.323045 27.335615 27.340019 27.583344 27.725190
##  [85] 27.936781 28.000000 28.071563 28.096965 28.103229 28.393504 28.500000
##  [92] 28.661863 28.815153 28.821358 28.917287 28.992136 29.000000 29.081550
```

```
##  [99] 29.162116 29.203806 29.233595 29.364579 29.381779 29.388872 29.409579
## [106] 29.632530 29.677234 29.677872 30.000000 30.072946 30.137734 30.214103
## [113] 30.500000 30.871028 30.896358 30.988540 31.000000 31.546613 31.634172
## [120] 31.652142 31.671743 31.690093 31.912935 32.000000 32.138453 32.153824
## [127] 32.314834 32.500000 32.709544 33.000000 34.000000 34.064244 34.399850
## [134] 34.500000 34.849621 34.953555 35.000000 35.062857 35.454204 35.807947
## [141] 36.000000 36.500000 36.510255 36.546132 36.672383 37.000000 37.090930
## [148] 37.394077 37.412390 37.601668 37.762799 38.000000 38.017756 38.500000
## [155] 38.581556 38.858687 38.894501 39.000000 39.023873 39.399906 39.459152
## [162] 39.587614 39.698040 39.822461 40.000000 40.500000 40.521290 41.000000
## [169] 41.786859 42.000000 42.210424 42.219630 42.375959 42.612139 43.000000
## [176] 43.227142 43.314447 43.682863 44.000000 44.271671 45.000000 45.500000
## [183] 45.714945 46.000000 46.084000 46.670658 46.791934 47.000000 47.010981
## [190] 47.696539 47.697690 48.000000 48.026941 48.190523 48.317725 48.390129
## [197] 48.426007 49.000000 49.802219 50.000000 50.783856 51.000000 51.053652
## [204] 52.000000 53.000000 54.000000 55.000000 55.500000 56.000000 57.000000
## [211] 58.000000 59.000000 60.000000 60.500000 61.000000 62.000000 63.000000
## [218] 64.000000 65.000000 66.000000 67.000000 70.000000 70.500000 71.000000
## [225] 74.000000 76.000000 80.000000
```

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




```
bp$out
```

```
##  [1] 66.0 65.0 71.0 70.5 61.0 62.0 63.0 65.0 61.0 60.0 64.0 65.0 63.0 71.0 64.0
```

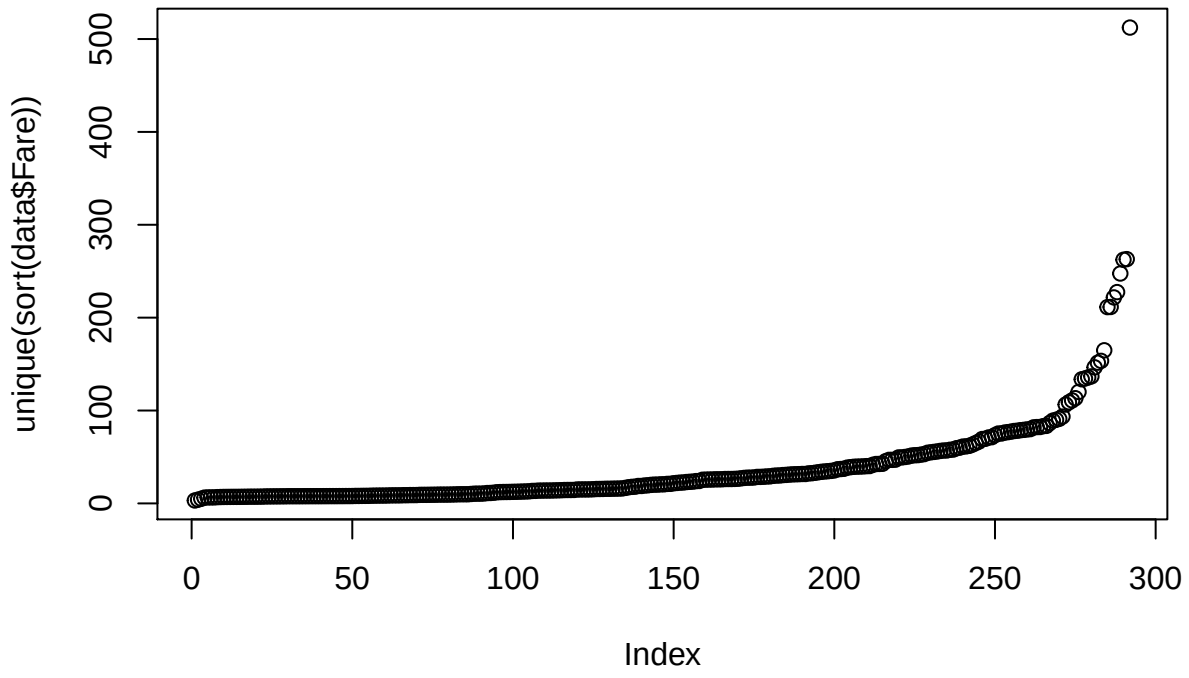
```
## [16] 62.0 62.0 60.0 61.0 80.0 70.0 60.0 60.0 70.0 62.0 74.0 62.0 63.0 60.0 60.0
## [31] 67.0 76.0 63.0 61.0 60.5 64.0 61.0 60.0 64.0 64.0
```

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.

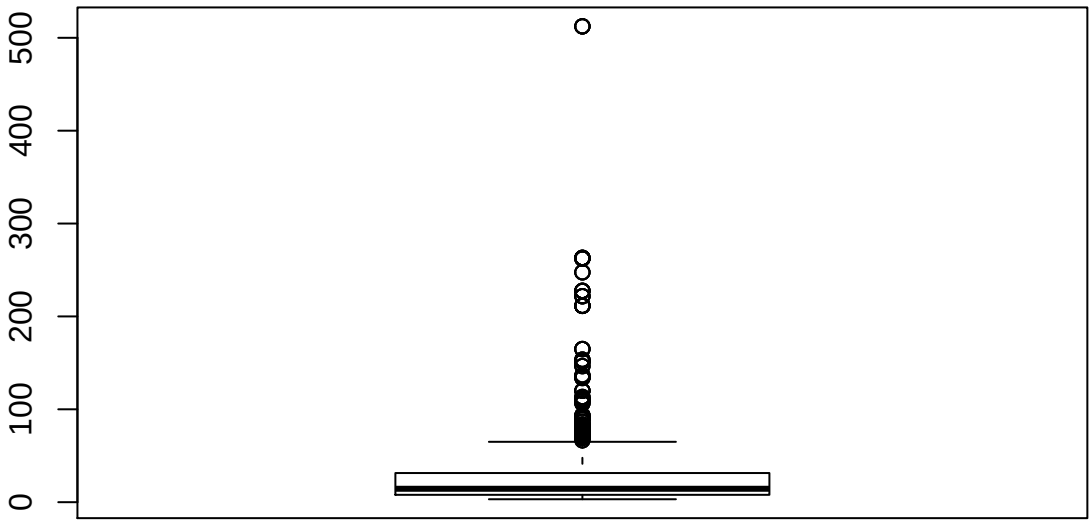
```
summary(data$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.171   7.925  14.458  33.601  31.387 512.329
```

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




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

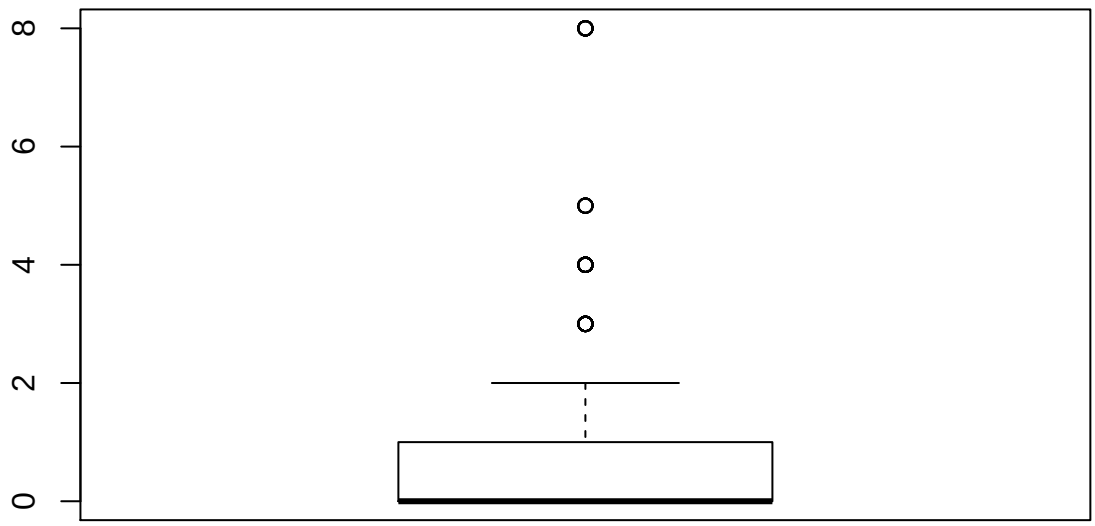


En el número de hermanos y parientes está dentro de un rango razonable y no existen valores que sean llamativos.

```
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
```

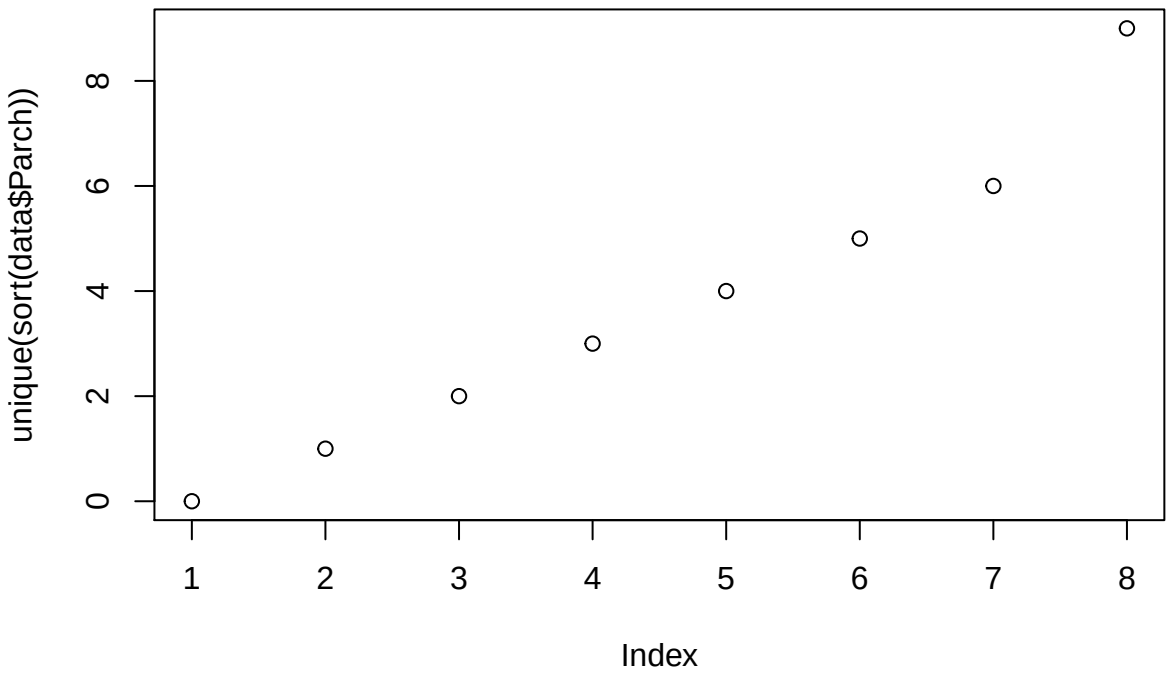
```
##  [1] 3 4 3 3 4 5 3 4 5 3 3 4 8 4 4 3 8 4 8 3 4 4 4 4 8 3 3 5 3 5 3 4 4 3 3 5 4 3
## [39] 4 8 4 3 4 8 4 8 3 4 5 3 4 8 4 8 4 3 3
```

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 = 0.97973, p-value = 1.342e-12
```

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

```
##
##  Shapiro-Wilk normality test
##
## data:  data$Fare
## W = 0.52458, 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 = 1.1891, num df = 341, denom df = 548, p-value = 0.07294
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.9840496 1.4431439
## sample estimates:
## ratio of variances
##           1.189116
```

```
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 = 4.5351, num df = 341, denom df = 548, p-value < 2.2e-16
```

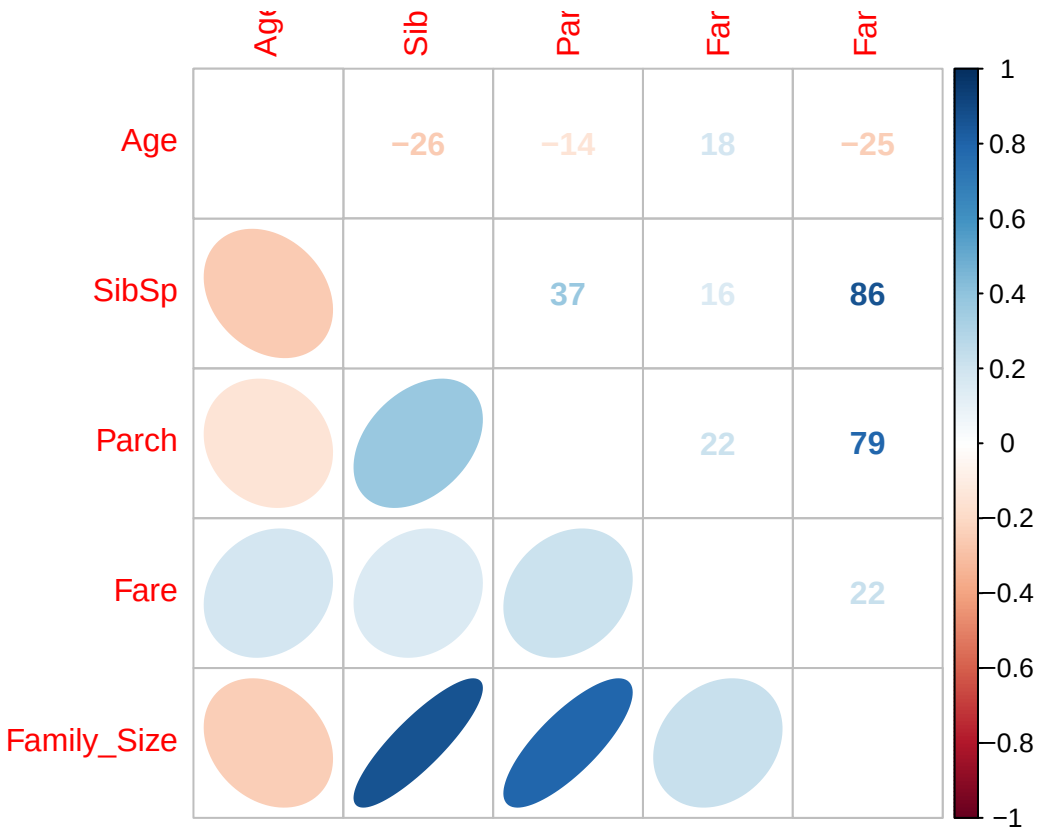
```
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  3.753024 5.503944
## sample estimates:
## ratio of variances
##            4.53512
```

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 = 87852, p-value = 0.1067
## 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 = 128197, 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.

Eliminar niveles que no tienen observaciones en la data

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

Pruebas

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

```
##
##  Pearson's Chi-squared test
##
## data:  tabla1
## X-squared = 102.89, 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 = 260.72, 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 = 37.272, df = 6, p-value = 1.559e-06
```

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

```
##
##  Pearson's Chi-squared test
##
## data:  tabla4
## X-squared = NaN, df = 7, p-value = NA
```

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

```
##
##  Pearson's Chi-squared test
##
## data:  tabla5
## X-squared = 128.02, df = 6, p-value < 2.2e-16
```

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

```
##
##  Pearson's Chi-squared test
##
## data:  tabla6
## X-squared = 28.005, df = 2, p-value = 8.294e-07
```

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

```
##
##  Pearson's Chi-squared test
##
## data:  tabla7
## X-squared = 16.045, df = 7, p-value = 0.02471
```

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

```
##
##  Pearson's Chi-squared test
##
## data:  tabla8
## X-squared = 82.29, df = 5, p-value = 2.783e-16
```

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

```
##
##  Pearson's Chi-squared test
##
## data:  tabla9
## X-squared = 283.31, 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 = 80.672, df = 8, p-value = 3.58e-14
```

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

```
##
##  Pearson's Chi-squared test
##
## data:  tabla11
## X-squared = 66.056, df = 3, p-value = 2.982e-14
```

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 : Existe diferencia en la mediana de las variables (Age y Fare) entre los distintos grupos de las variables categóricas.

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Survived
## Kruskal-Wallis chi-squared = 2.6035, df = 1, p-value = 0.1066
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Survived
## Kruskal-Wallis chi-squared = 84.421, df = 1, p-value < 2.2e-16
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Sex
## Kruskal-Wallis chi-squared = 12.79, df = 1, p-value = 0.0003484
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Sex
## Kruskal-Wallis chi-squared = 67.098, df = 1, p-value = 2.583e-16
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Pclass
## Kruskal-Wallis chi-squared = 231.92, 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 = 742.77, 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 = 263.52, df = 6, p-value < 2.2e-16
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Cabin
## Kruskal-Wallis chi-squared = 597.21, df = 6, p-value < 2.2e-16
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Embarked
## Kruskal-Wallis chi-squared = 5.9622, df = 2, p-value = 0.05074
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Embarked
## Kruskal-Wallis chi-squared = 138.38, 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 = 263.52, df = 6, p-value < 2.2e-16
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Cabin
## Kruskal-Wallis chi-squared = 597.21, df = 6, p-value < 2.2e-16
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Age by Name
## Kruskal-Wallis chi-squared = 369.26, 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 = 151.47, 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 = 70.231, df = 3, p-value = 3.808e-15
```

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

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Fare by Family_Segmented
## Kruskal-Wallis chi-squared = 354.25, df = 3, p-value < 2.2e-16
```

Análisis:

- Solo en el primer test de Age- Embarked y Age-Survived se acepta la hipótesis nula, es decir que no hay diferencias significativas entre la mediana de la edad en los grupos determinados, es decir los grupos dentro de esas variables tienen una mediana igual.
- En las demás se acepta la hipotesis alternativa, es decir que si hay diferencias significativas entre las medianas de los grupos.

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.

```
# 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)
```

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: 712, 712, 713, 714, 713
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.8250314  0.6190666
##   21    0.8194198  0.6119267
##   41    0.8115420  0.5958298
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 2.
```

```
varImp(model_rf1)
```

```
## rf variable importance
##
##   only 20 most important variables shown (out of 41)
##
##                           Importance
## Sexmale                       100.00
## NameMr.                        97.76
## NameMrs.                       85.40
## NameMiss.                      76.33
## Pclass3                        74.70
## CabinF                         74.67
## Fare                           68.69
## CabinE                         63.76
## Age                            56.12
## Family_Size                    50.74
## Pclass2                        47.14
## CabinG                         46.16
## Family_SegmentedLargeFam       46.12
## CabinB                         43.83
## Family_SegmentedSmallFam       42.25
## SibSp                          39.09
## EmbarkedS                      38.22
## Fare_Segmented8-15             34.18
## Age_Segmented21-30             28.13
## Fare_Segmented15-31            26.70
```

```
pred_rf1 <- predict(model_rf1, newdata = test)
pred_rf1_train <- predict(model_rf1, newdata = train)

confusionMatrix(pred_rf1_train, train$Survived)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 519  95
##          1  30 247
##
##                Accuracy : 0.8597
##                  95% CI : (0.8352, 0.8819)
##     No Information Rate : 0.6162
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6924
##
##  Mcnemar's Test P-Value : 1.038e-08
##
##             Sensitivity : 0.9454
##             Specificity : 0.7222
##          Pos Pred Value : 0.8453
##          Neg Pred Value : 0.8917
##              Prevalence : 0.6162
##          Detection Rate : 0.5825
##    Detection Prevalence : 0.6891
```

```
##       Balanced Accuracy : 0.8338
##
##        '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, 12:14))
test1 <- test %>% dplyr::select(c(1:4, 7:9, 12:14))
model_rf2 <- train(Survived ~ ., train1, method = "rf", trControl = myControl,
    importance = TRUE)
model_rf2
```

```
## Random Forest
##
## 891 samples
##   9 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 712, 713, 713, 712, 714
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.7834260  0.5205613
##   12    0.8249176  0.6238261
##   22    0.8125642  0.5992816
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 12.
```

```
pred_rf2 <- predict(model_rf2, newdata = test1)
pred_rf2_train <- predict(model_rf2, newdata = train1)

confusionMatrix(pred_rf2_train, train1$Survived)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 544  19
##          1   5 323
##
##                Accuracy : 0.9731
##                  95% CI : (0.9602, 0.9827)
##     No Information Rate : 0.6162
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9426
```

```
##
##  Mcnemar's Test P-Value : 0.007963
##
##             Sensitivity : 0.9909
##             Specificity : 0.9444
##          Pos Pred Value : 0.9663
##          Neg Pred Value : 0.9848
##              Prevalence : 0.6162
##          Detection Rate : 0.6105
##    Detection Prevalence : 0.6319
##       Balanced Accuracy : 0.9677
##
##        'Positive' Class : 0
##
```

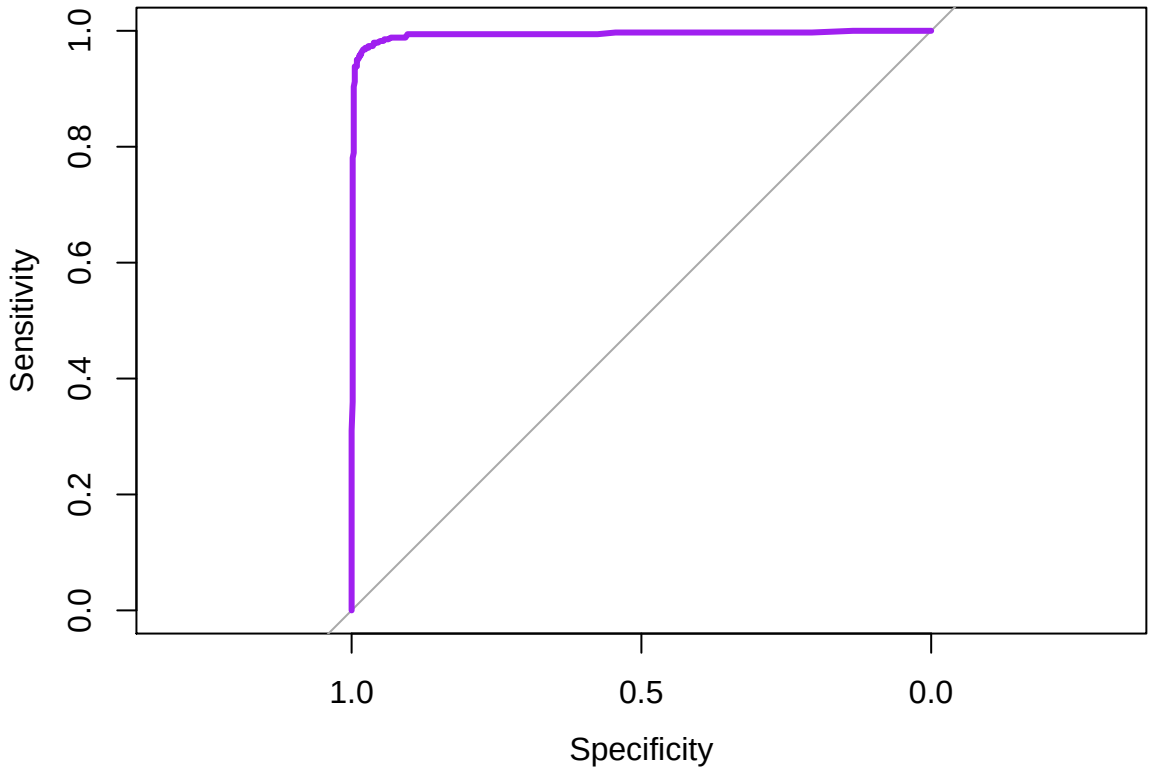
Se observan buenas medidas para la accuracy. Se observa un buen modelo, con un poder de predicción bastante alto.

Curva ROC Y AUC

```
probs_rf <- predict(model_rf2, train1, type = "prob")
ScoreRFauc <- probs_rf[, 2]
rf_roc <- roc(train1$Survived, ScoreRFauc, data = train1)
```

ROC

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



AUC

```
auc(rf_roc)
```

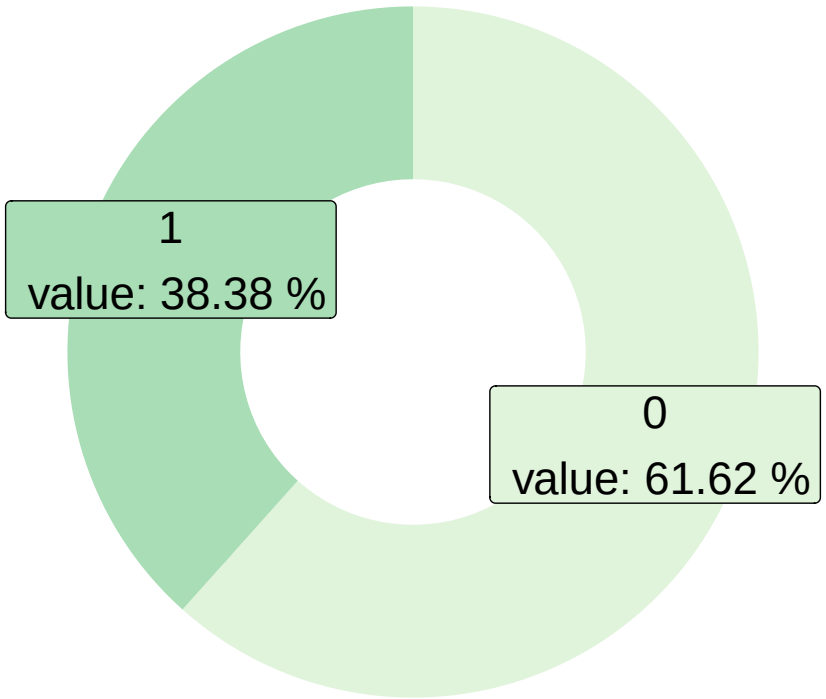
```
## Area under the curve: 0.9931
```

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.

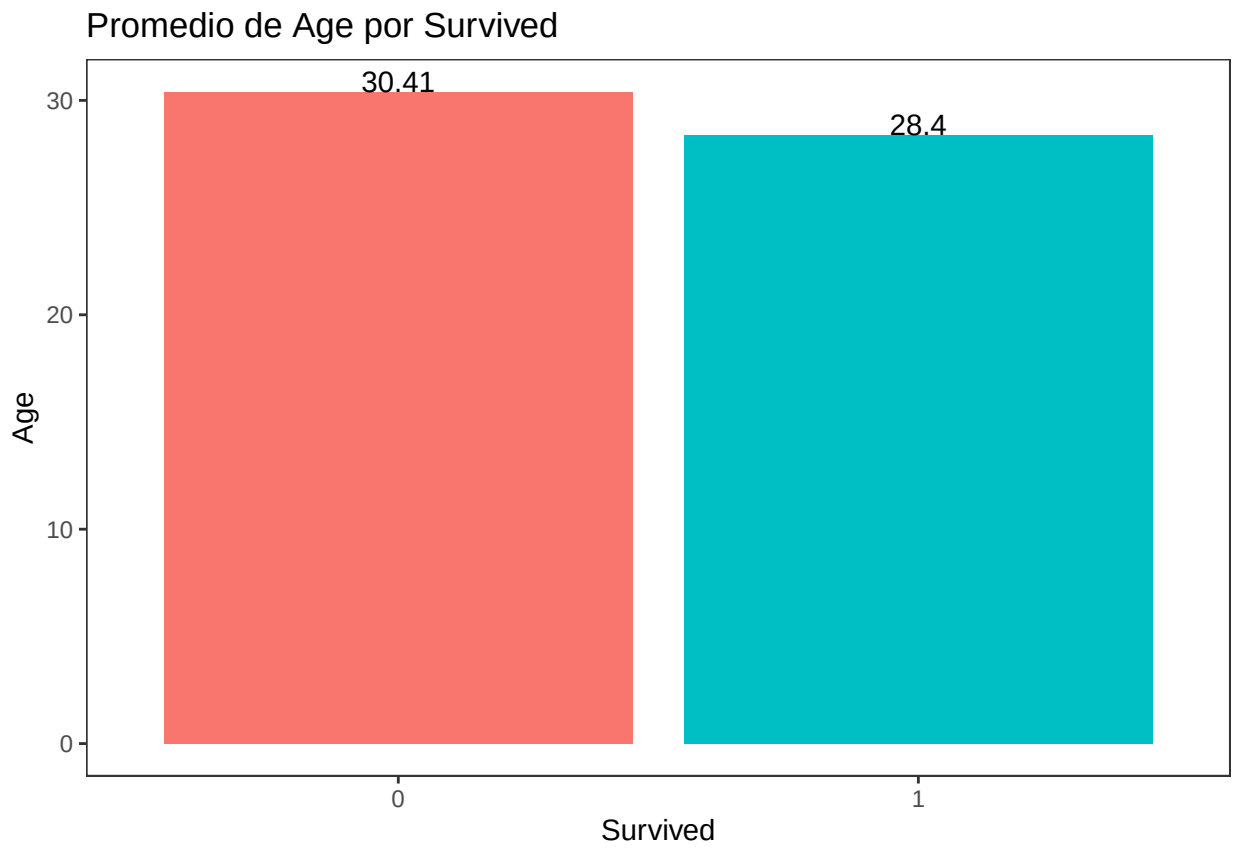
5 Representación de los resultados a partir de tablas y gráficas.

```
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")
```



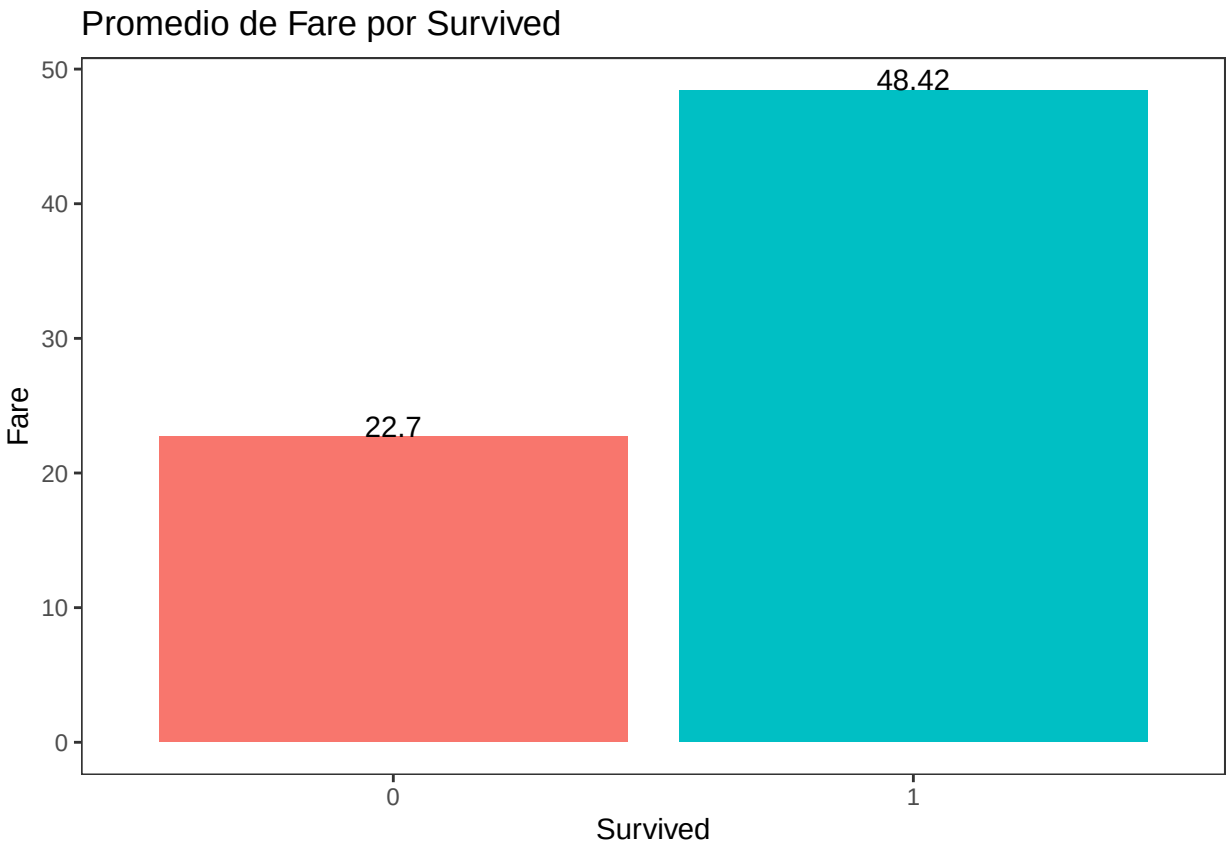
Análisis: El 61.62% de las personas de la data del Titanic no sobrevivieron al hundimiento y el 38.38% de las personas si sobrevivieron.

```
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 28 años, la edad media de los que no sobrevivieron es de aproxidamente 30 años.

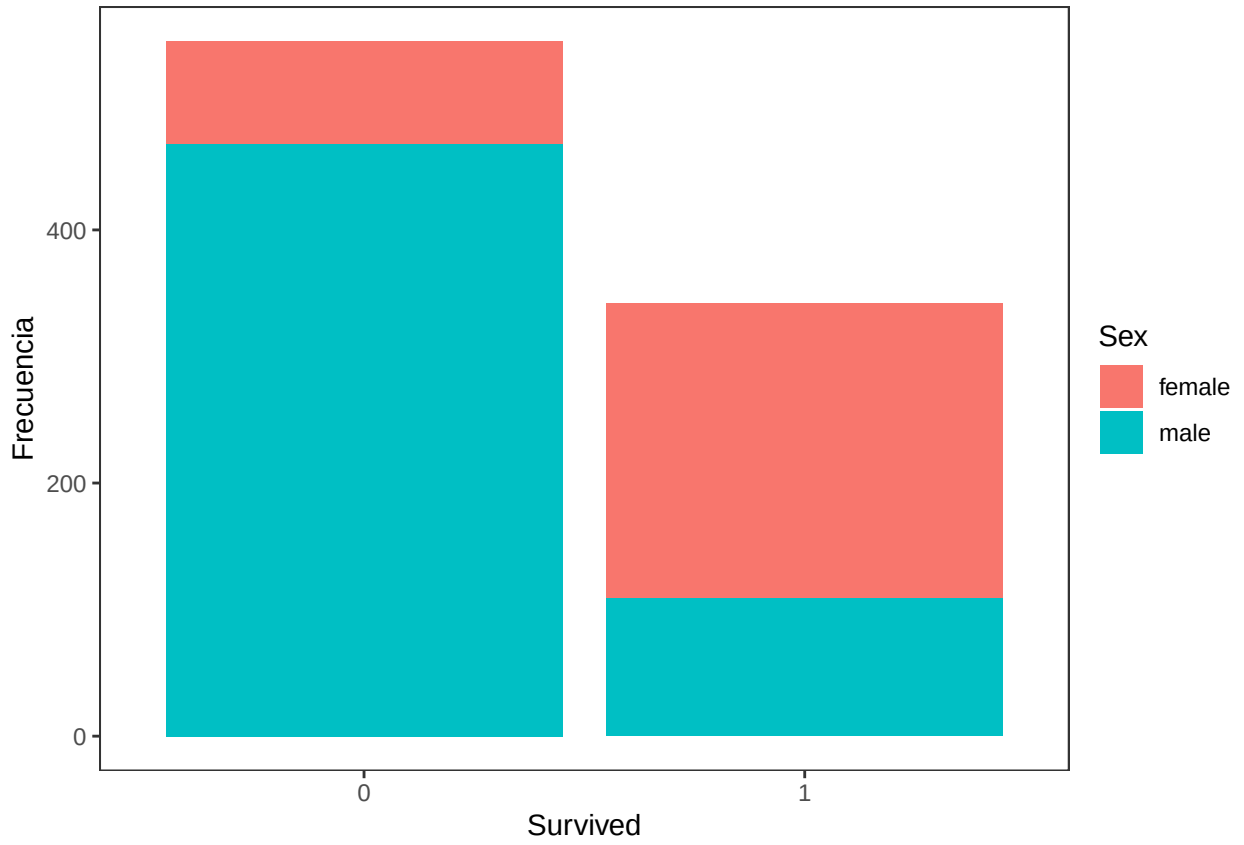
```
m_control2 <- aggregate(Fare ~ Survived, FUN = mean, data = data)
ggplot(data = m_control2, mapping = aes(x = Survived, y = Fare, fill = Survived)) +
    geom_col() + labs(title = "Promedio de Fare por Survived", x = "Survived",
    y = "Fare", fill = "Survived") + geom_text(aes(label = round(Fare,
    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 48.42 libras británicas, El costo promedio del tickets pagado por los que no sobrevivieron es de 22.74 libras británicas.

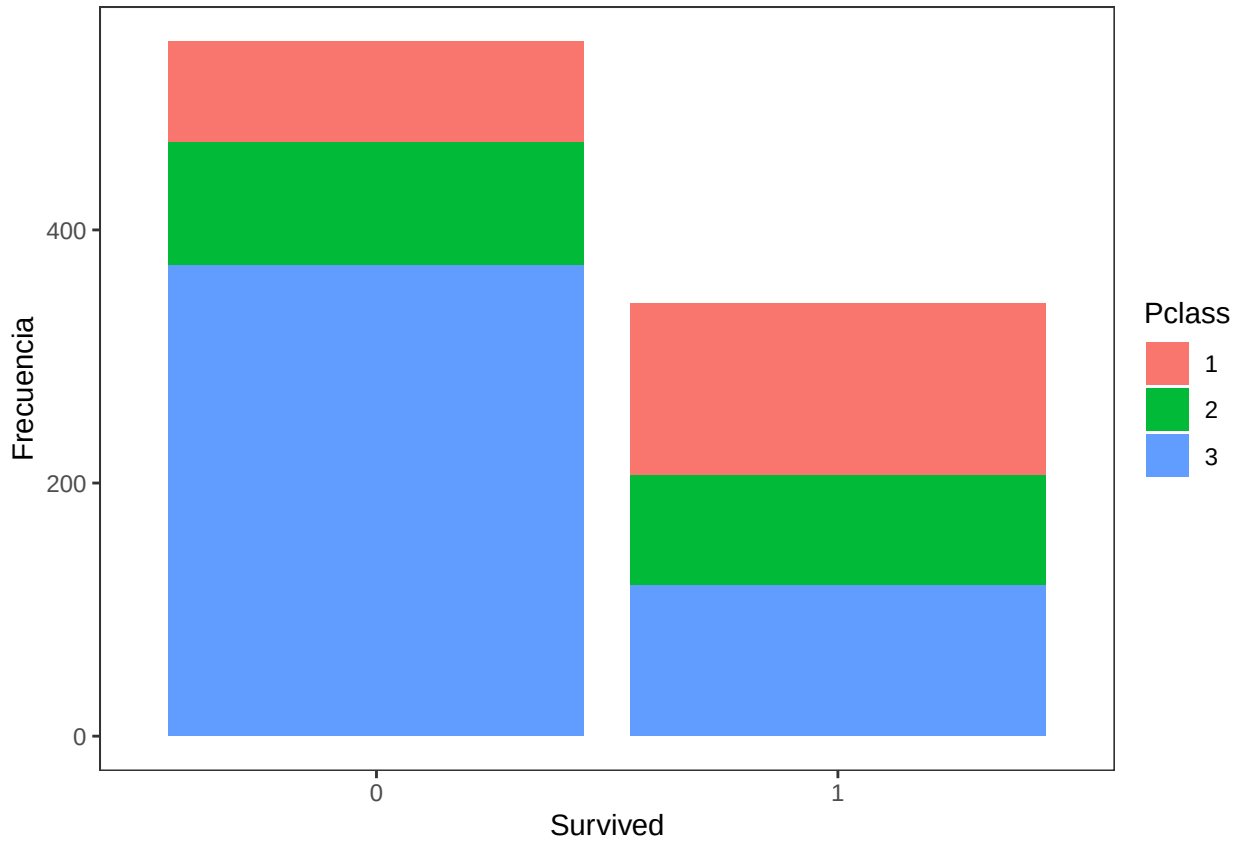
5.0.1

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



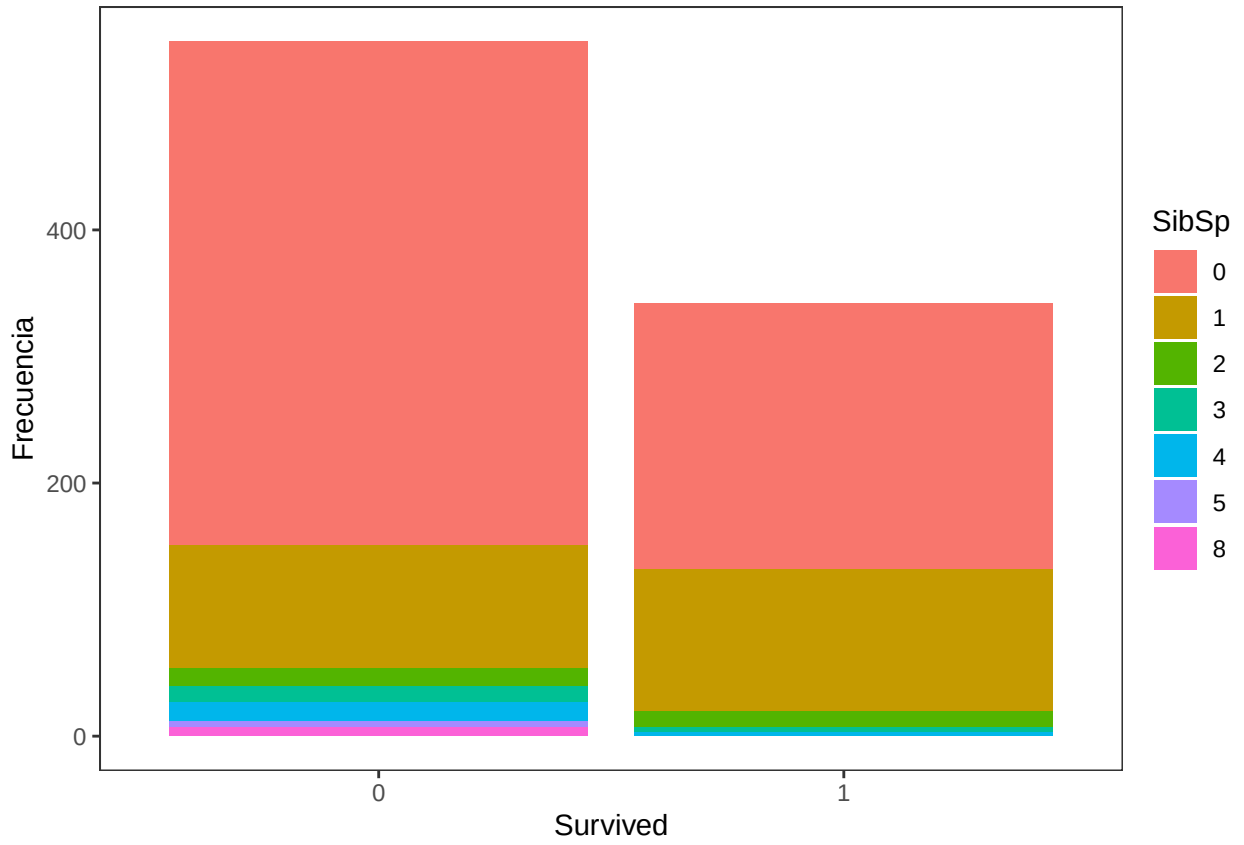
Análisis: El grupo de los que no sobrevivieron estuvo conformado en us mayoría por hombres, y del grupo de los que sobrevivieron estaba conformado más por mujeres, esto tiene sentido ya que le dieron prioridad para subir a los botes a las mujeres.

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



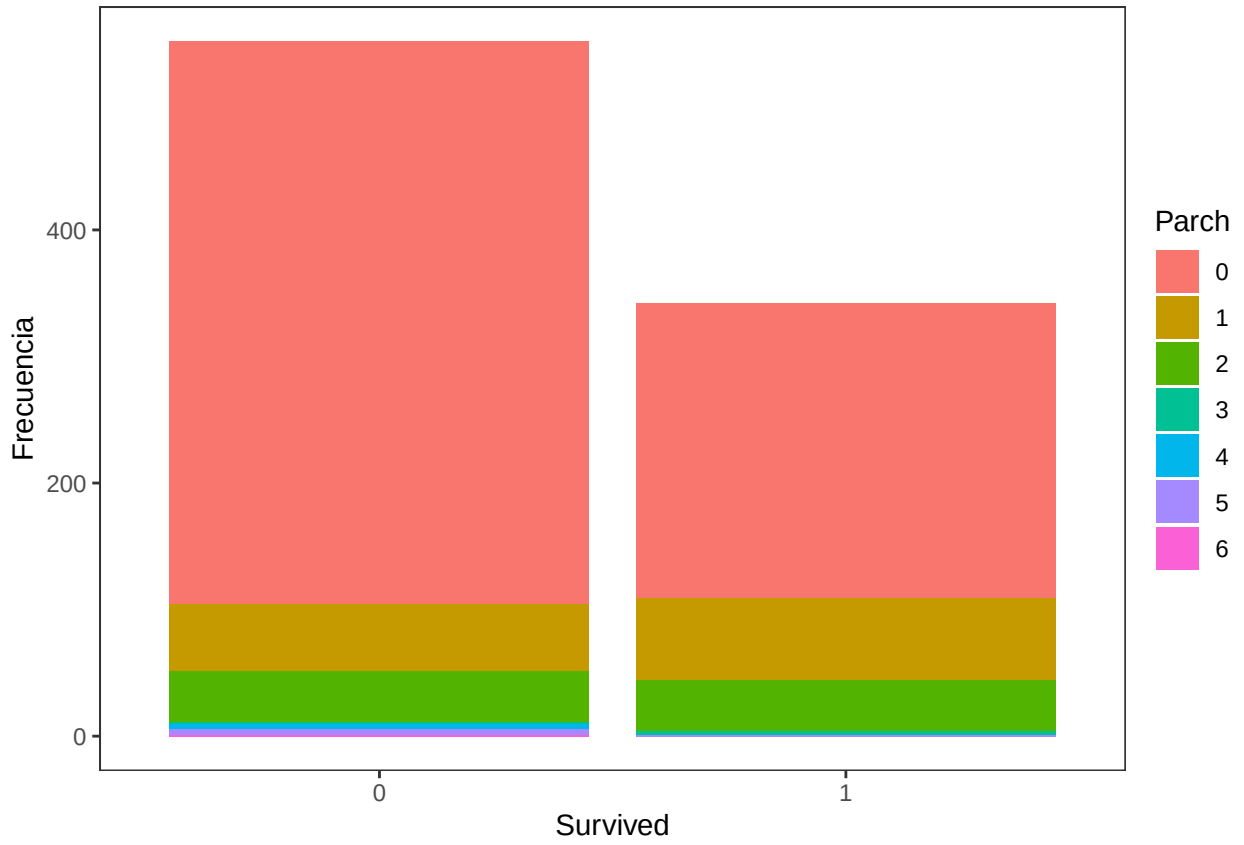
Análisis: El grupo de los que no sobrevivieron aproximadamente el 75% eran de la tercera clase, en el grupo de los que sobrevivieron predominan los de primera clase y tercera clase.

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



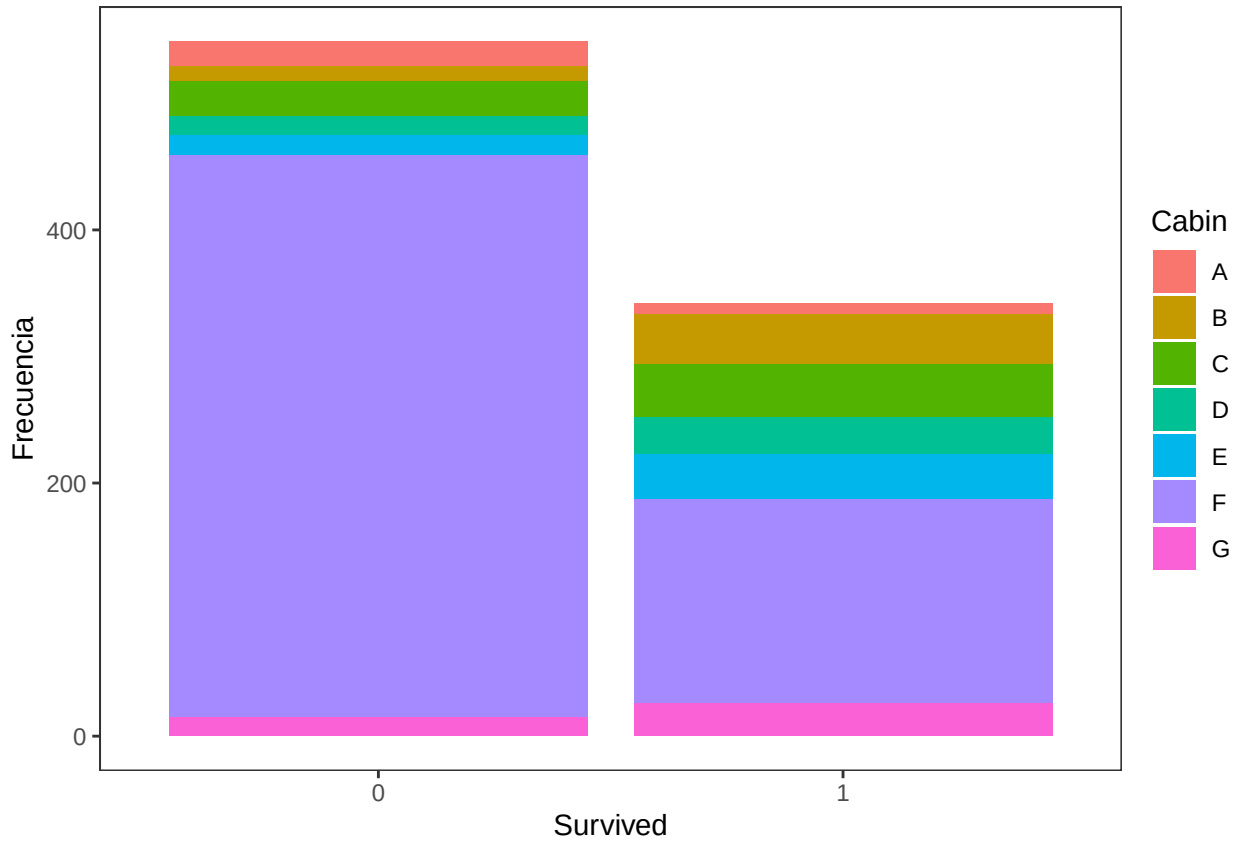
Análisis: En ambos grupos prodominan personas que no tenian hermanos, hermanas, esposos ni esposas a bordo.

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



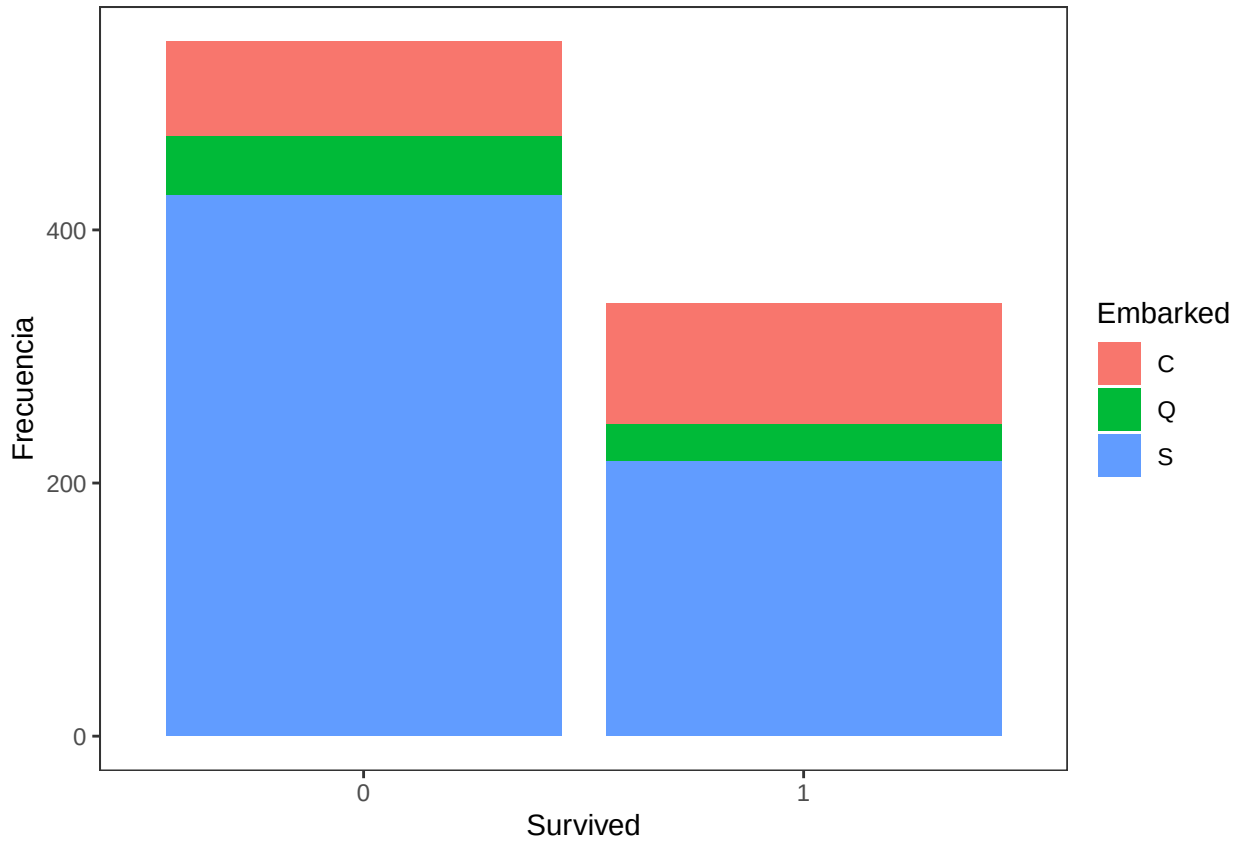
Análisis: En ambos grupos predominan personas que no tenian padres ni hijos a bordo.

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



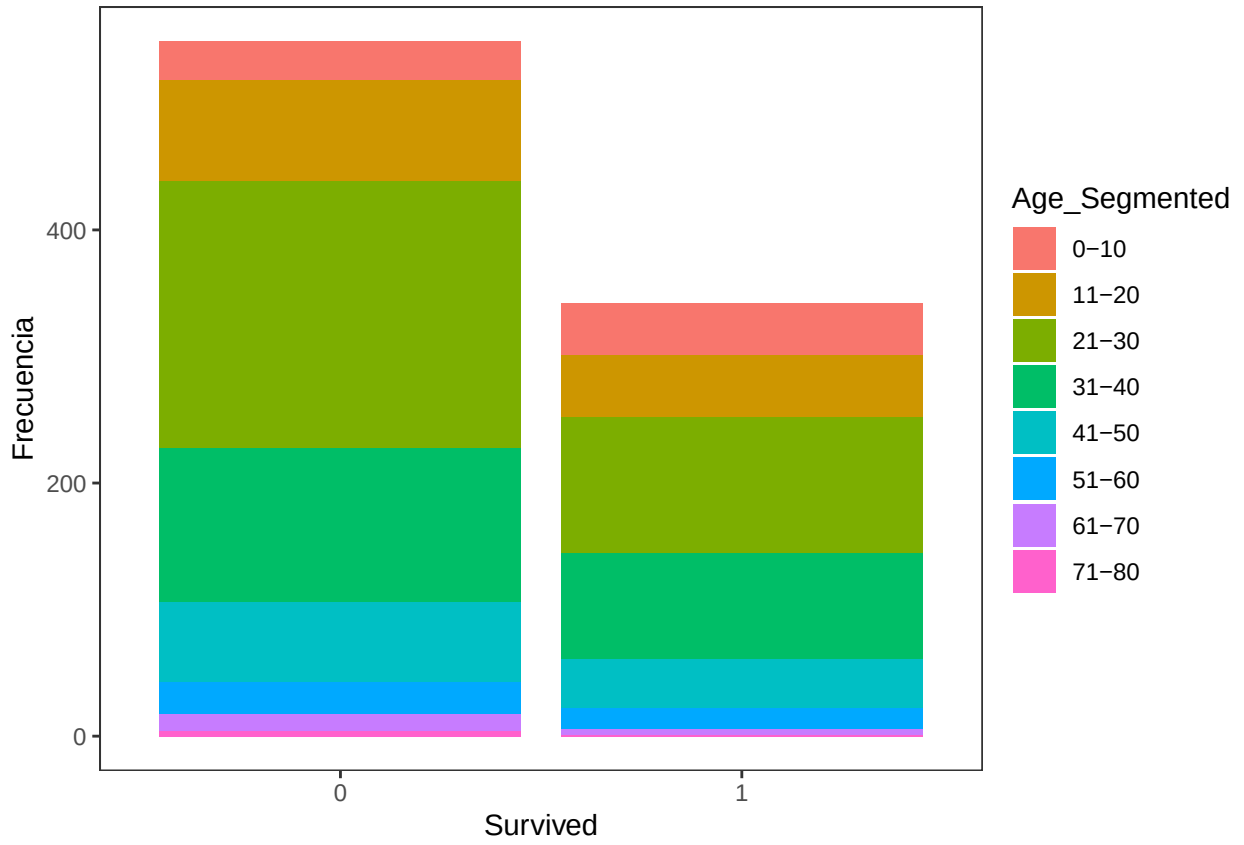
Análisis: En ambos grupos predominan las personas que estaban en la cabina F.

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



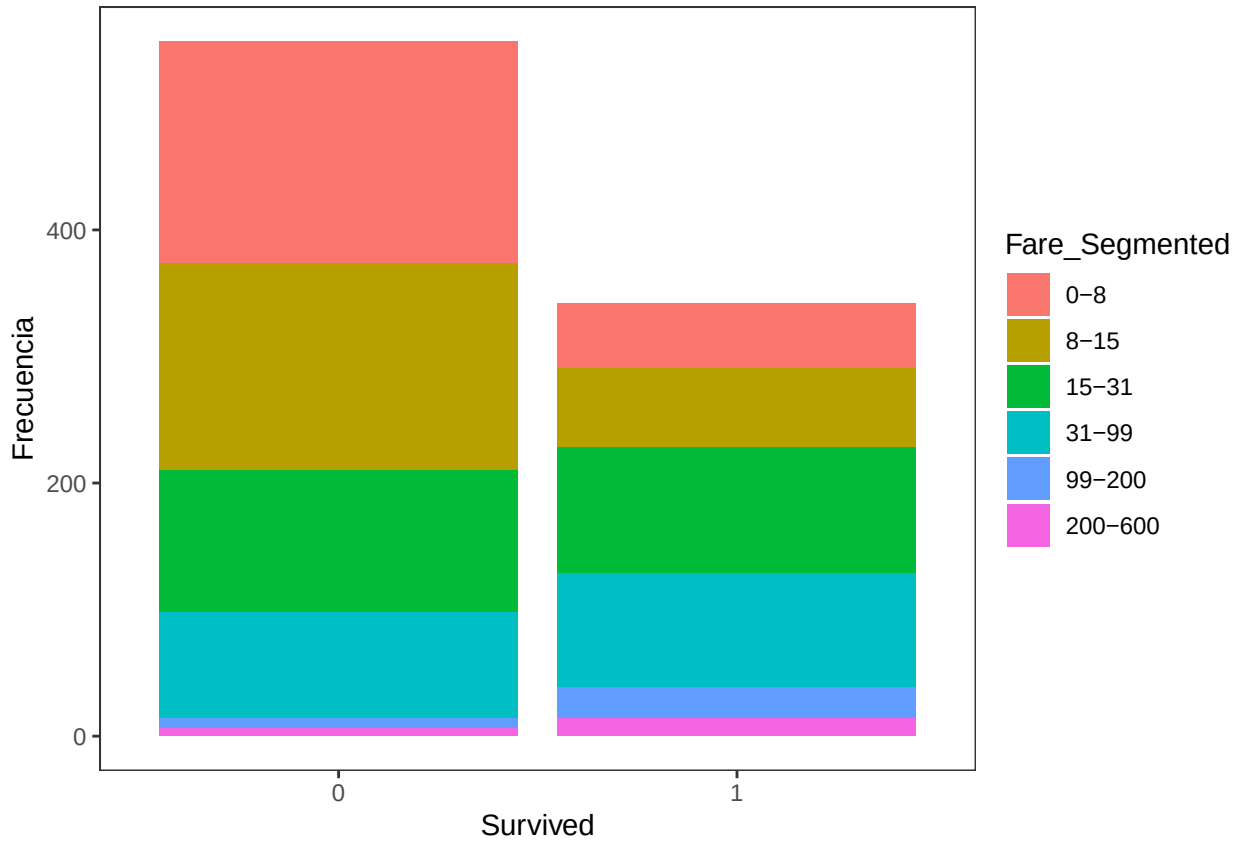
Análisis: En ambos grupos predominan personas que embarcaron en Southampton.

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



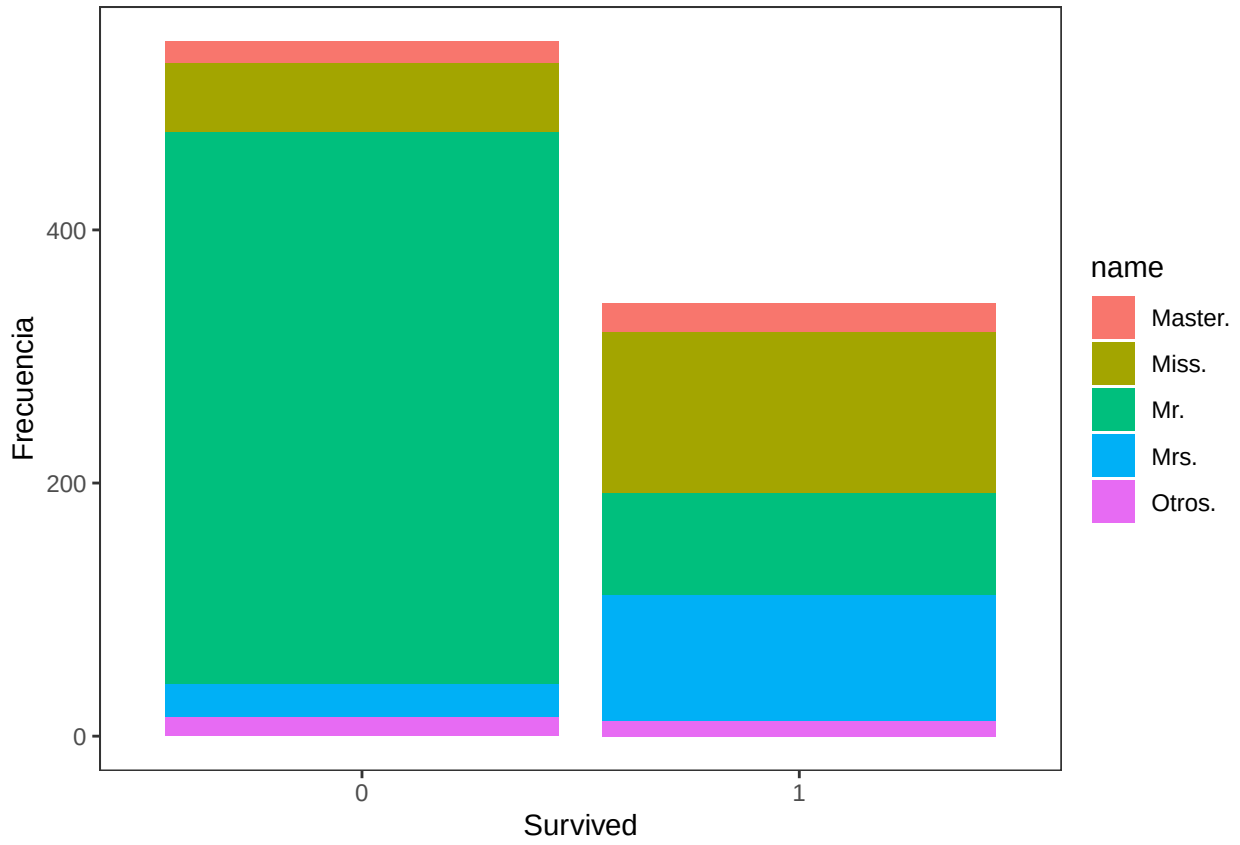
Análisis: En ambos grupos predominan personas entre 21-30 años y de 31-40.

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



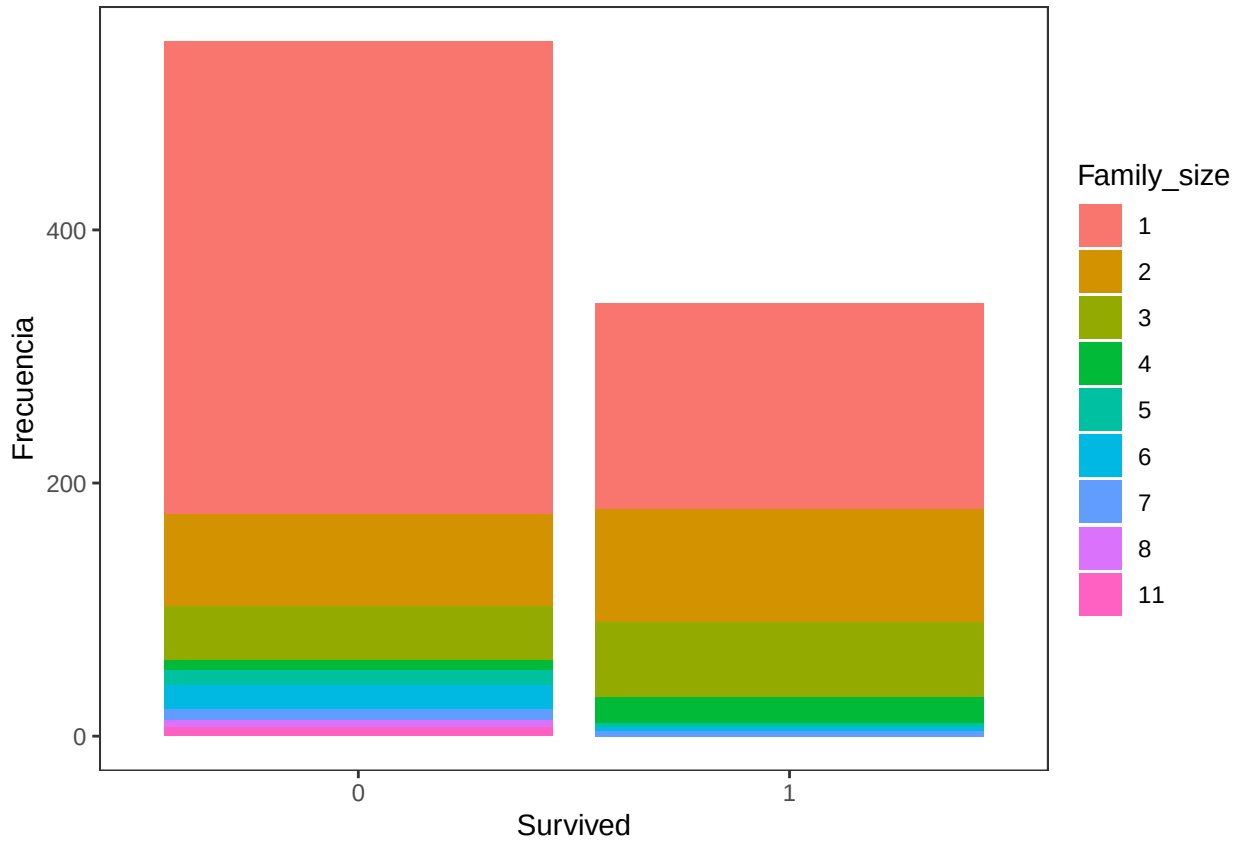
Análisis: En el grupo de los que no sobrevivieron predominan los que pagaron por el ticket entre 0 a 15 libras británicas, y en el grupo de los que sobrevivieron predominan los que pagaron entre 15 y 99 libras británicas.

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



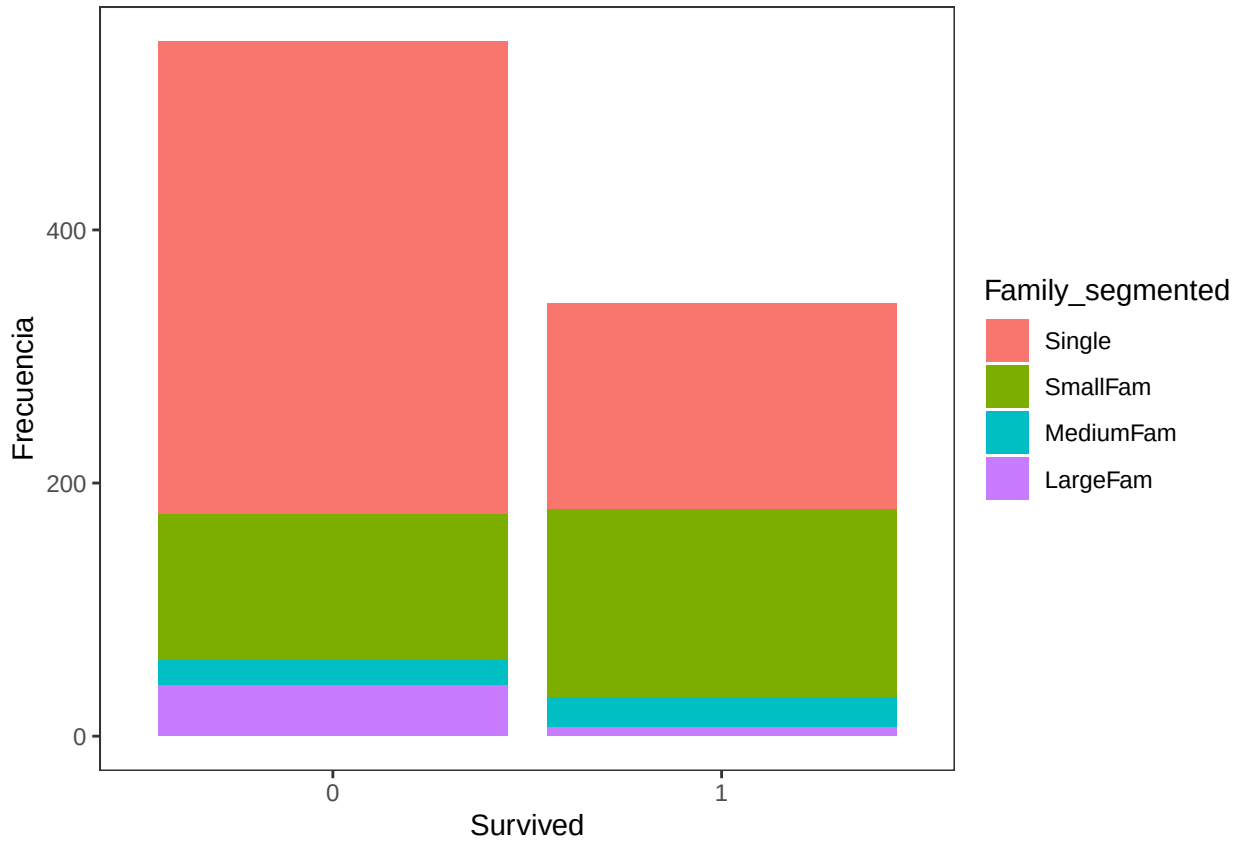
Análisis: En el grupo de los que no sobrevivieron estuvo compuesto en su mayoría por Señores. Y en el grupo de los que sobrevivieron estuvo compuesto por Señoritas y señoras.

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



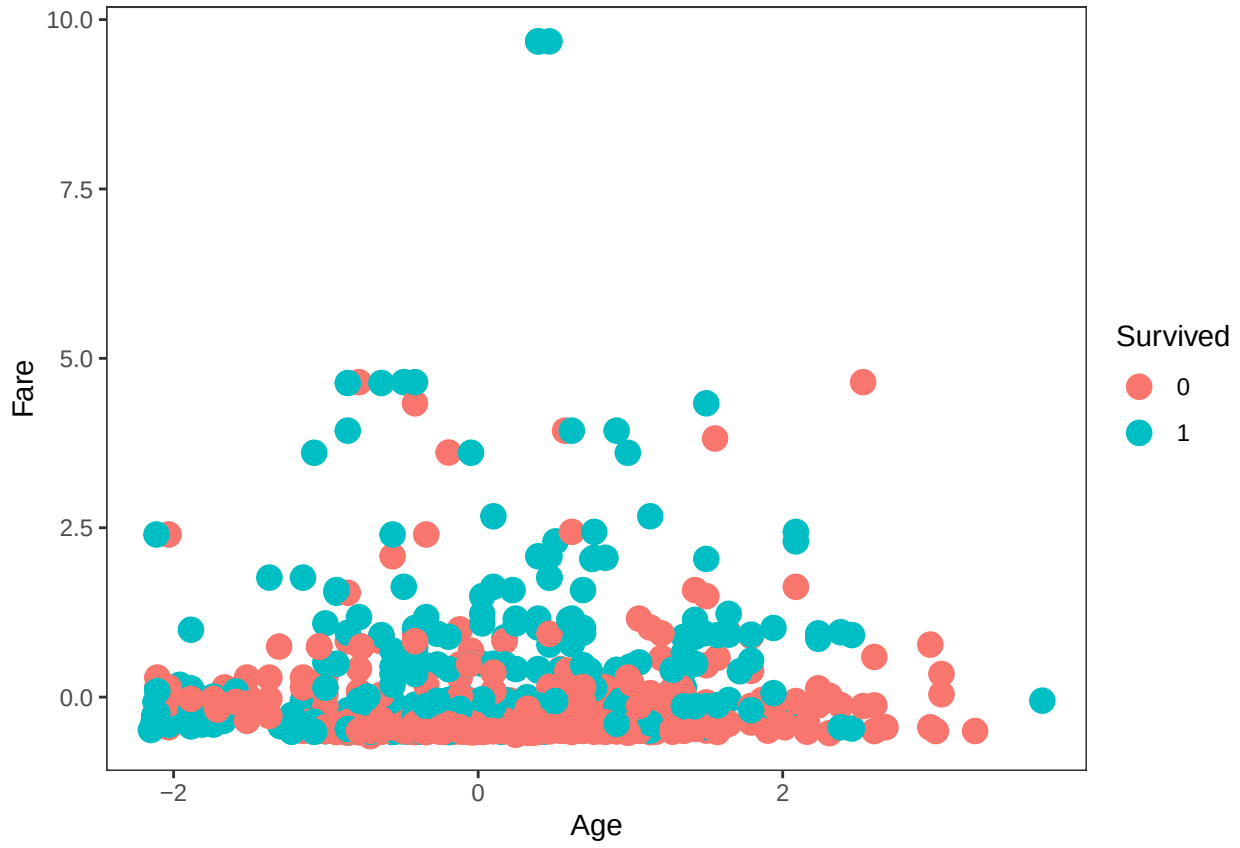
Análisis: En ambos grupos predominan personas que viajaba solas.

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



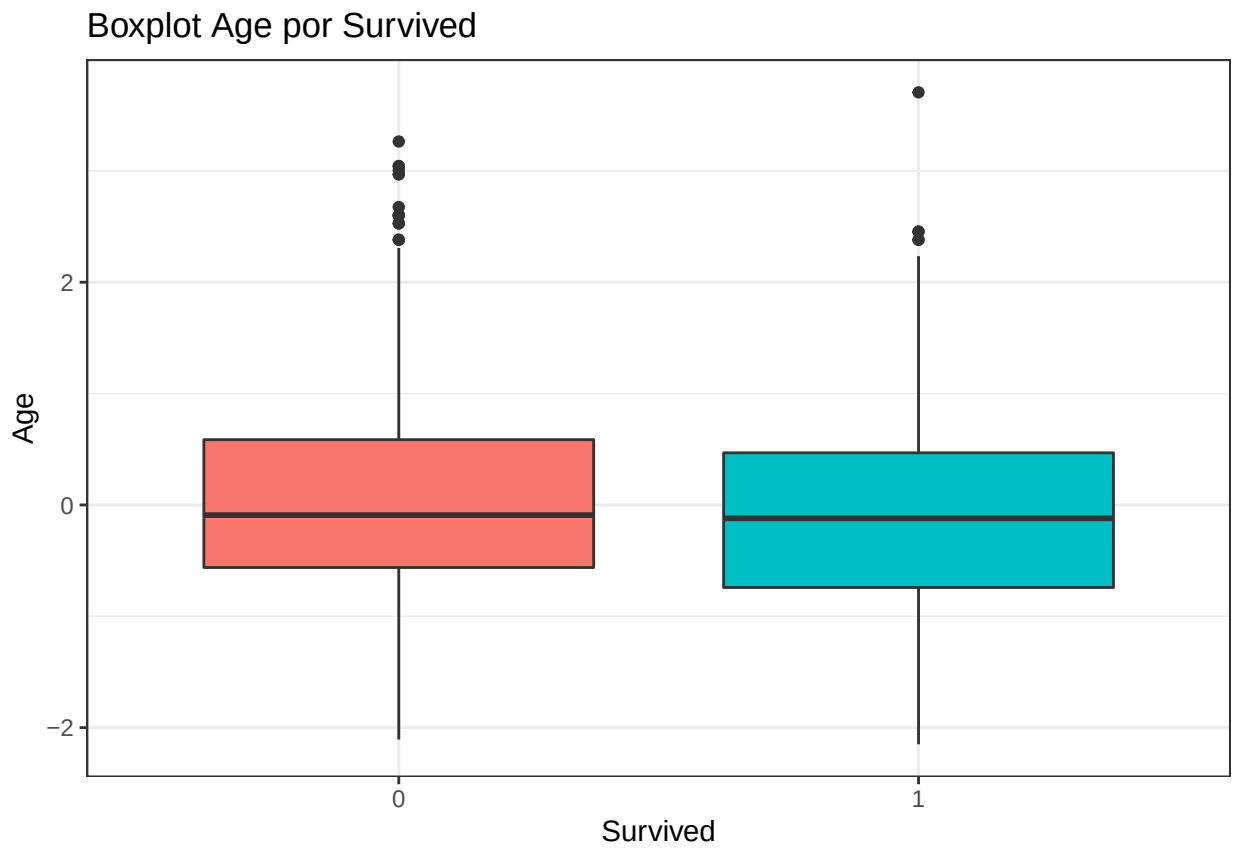
Análisis: En ambos grupos predominan personas personas que viajan solas y de Familia pequeña.

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



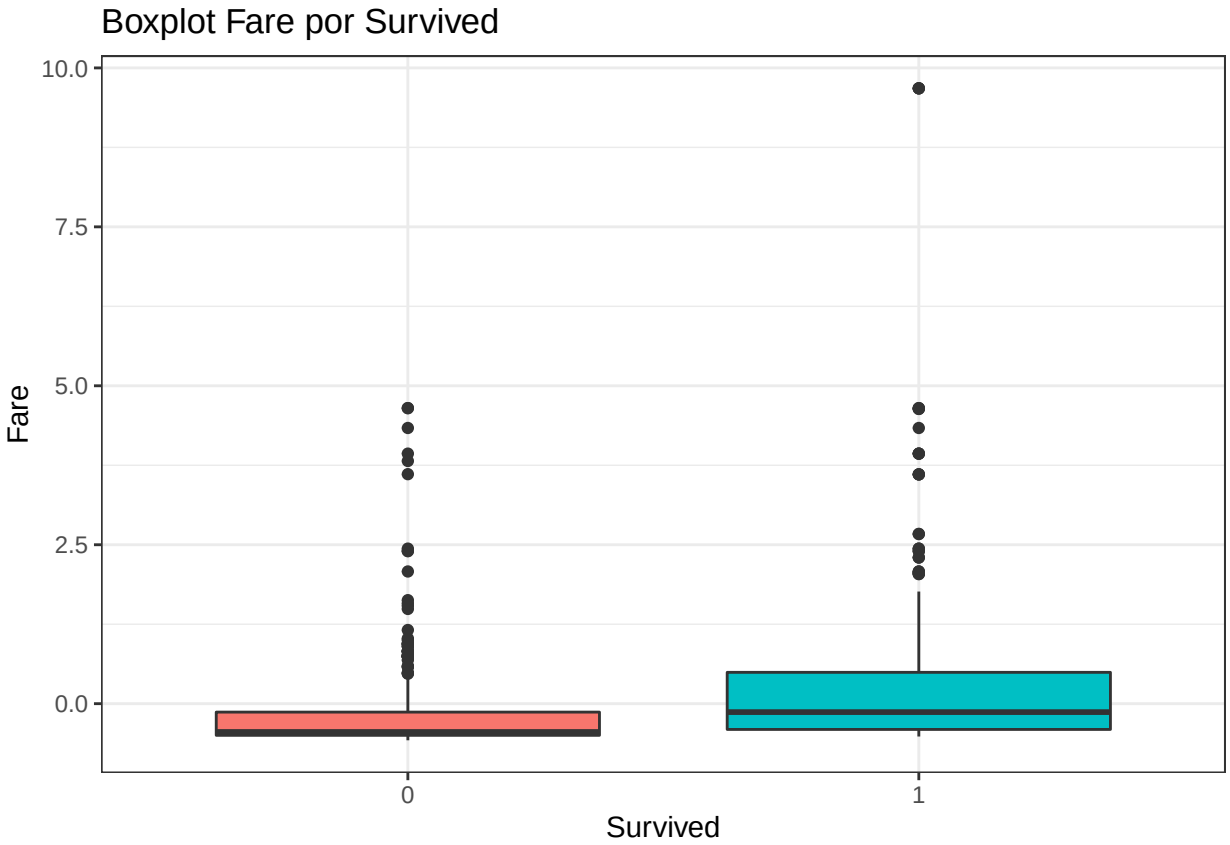
Se corrobora que no hay relación entre las variables.

```
ggplot(train, 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")
```



Hay valores atipicos en ambos grupos, las medias son muy parecidas. Anteriormente se comprobó que no había diferencia entre las medias de la edad en ambos grupos.

```
ggplot(train, 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")
```



Se evidencian valores atípicos, además se evidencia que la media del costo del ticket para el grupo que sobrevivió es mayor que la media del grupo que no sobrevivió.

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.

7 Contribuciones

Contribuciones	Firma
Investigación previa	Joshelyn Intriago, David Morocho
Redacción de las respuestas	Joshelyn Intriago, David Morocho
Desarrollo código	Joshelyn Intriago, David Morocho