TIPOLOGÍA Y CICLO DE VIDA DE LOS DATOS
Práctica 2

JORDI COSTILLA AZNAR
DIEGO GONZÁLEZ MARTÍNEZ

Contenido

1.- Descripción del dataset

Disponemos de un dataset distribuido en 3 ficheros:

- train.cvs
- test.cvs
- gender_submission.cvs

Empezamos por el último, contiene el id de cada pasajero y si sobrevivió al accidente del Titanic, este dato ya está incorporado en el fichero train.csv que además contiene el resto de los campos que ya están asignados en el fichero test, una vez realizada la carga de los ficheros, añadiremos al dataframe de test el campo "Survived" del fichero gender_submission.cvs

Ahora disponemos de dos datasets uno para entrenamiento (train) y otro para comprobar el funcionamiento (test), ambos disponen de la misma estructura:

Variable	Definición
PassengerId	Identificador único del pasajero, tipo integer
Survived	Valor binario si el pasajero sobrevivió o no al accidente
Pclass	Clase asignada al pasajero, tipo numérico, aunque en realidad se trata de un factor
Name	Tipo string, indica el nombre y apellidos del pasajero
Sex	Define el sexo del pasajero, aunque es del tipo string, se trata de un factor
Age	Define la edad del pasajero, tipo numérico
SibSp	Define si tiene hermanos / cónyuges a bordo del Titanic, tipo integer
Parch	Define si tiene padres / niños a bordo del Titanic, tipo integer
Ticket	Identificación del ticket del pasajero, tipo string
Fare	Tarifa aplicada en la compra, tipo numérico
Cabin	Identifica la cabina del pasajero, si éste dispone de una, tipo string
Embarked	Lugar de embarque del pasajero, tipo string

1.1 - Importancia/Resolución del problema

A partir de este conjunto de datos se plantea la problemática de determinar qué variables influyen más sobre el resultado final si el pasajero sobrevivió o no, a partir de la descripción del dataset procederemos a determinar que variables serán las más importantes para predecir si un pasajero del Titanic podría haber sobrevivido al accidente.

Este análisis toma especial relevancia para valorar posibles mejoras en este tipo de buques, para mejorar en la medida de lo posible la tasa de supervivientes en caso de accidente, obviamente el primer esfuerzo debe centrarse en evitar los accidentes, pero este punto no entra dentro del alcance de este estudio.

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

Empezaremos cargando los ficheros a analizar:

```
train <- read.csv('D:/Documentos/UOC/Master/3-TCVD/Practica2/train.csv')
test <- read.csv('D:/Documentos/UOC/Master/3-TCVD/Practica2/test.csv')
gender <- read.csv('D:/Documentos/UOC/Master/3-
TCVD/Practica2/gender_submission.csv')
```

Una vez cargados, procedemos a visualizarlos:

```
summary(train)
```

```
##   PassengerId       Survived          Pclass
##  Min.   :  1.0   Min.   :0.0000   Min.   :1.000
##  1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000
##  Median :446.0   Median :0.0000   Median :3.000
##  Mean   :446.0   Mean   :0.3838   Mean   :2.309
##  3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :891.0   Max.   :1.0000   Max.   :3.000
##
##                                     Name         Sex           Age
##  Abbing, Mr. Anthony                   :  1   female:314   Min.   : 0.42
##  Abbott, Mr. Rossmore Edward           :  1   male  :577   1st Qu.:20.12
##  Abbott, Mrs. Stanton (Rosa Hunt)      :  1                Median :28.00
##  Abelson, Mr. Samuel                   :  1                Mean   :29.70
##  Abelson, Mrs. Samuel (Hannah Wizosky):  1                3rd Qu.:38.00
##  Adahl, Mr. Mauritz Nils Martin        :  1                Max.   :80.00
##  (Other)                               :885                NA's   :177
##      SibSp           Parch             Ticket         Fare
##  Min.   :0.000   Min.   :0.0000   1601    :  7   Min.   :  0.00
##  1st Qu.:0.000   1st Qu.:0.0000   347082  :  7   1st Qu.:  7.91
##  Median :0.000   Median :0.0000   CA. 2343:  7   Median : 14.45
##  Mean   :0.523   Mean   :0.3816   3101295 :  6   Mean   : 32.20
##  3rd Qu.:1.000   3rd Qu.:0.0000   347088  :  6   3rd Qu.: 31.00
##  Max.   :8.000   Max.   :6.0000   CA 2144 :  6   Max.   :512.33
##                                   (Other) :852
##          Cabin     Embarked
##             :687    :  2
```

```
##  B96 B98    :  4   C:168
##  C23 C25 C27:  4   Q: 77
##  G6         :  4   S:644
##  C22 C26    :  3
##  D          :  3
##  (Other)    :186
```

```
summary(test)
```

```
##   PassengerId         Pclass
##  Min.   : 892.0   Min.   :1.000
##  1st Qu.: 996.2   1st Qu.:1.000
##  Median :1100.5   Median :3.000
##  Mean   :1100.5   Mean   :2.266
##  3rd Qu.:1204.8   3rd Qu.:3.000
##  Max.   :1309.0   Max.   :3.000
##
##                                          Name         Sex
##  Abbott, Master. Eugene Joseph            :  1   female:152
##  Abelseth, Miss. Karen Marie              :  1   male  :266
##  Abelseth, Mr. Olaus Jorgensen            :  1
##  Abrahamsson, Mr. Abraham August Johannes :  1
##  Abrahim, Mrs. Joseph (Sophie Halaut Easu):  1
##  Aks, Master. Philip Frank                :  1
##  (Other)                                  :412
##       Age            SibSp            Parch             Ticket
##  Min.   : 0.17   Min.   :0.0000   Min.   :0.0000   PC 17608:  5
##  1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.0000   113503  :  4
##  Median :27.00   Median :0.0000   Median :0.0000   CA. 2343:  4
##  Mean   :30.27   Mean   :0.4474   Mean   :0.3923   16966   :  3
##  3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.0000   220845  :  3
##  Max.   :76.00   Max.   :8.0000   Max.   :9.0000   347077  :  3
##  NA's   :86                                        (Other) :396
##       Fare                    Cabin     Embarked
##  Min.   :  0.000                 :327   C:102
##  1st Qu.:  7.896   B57 B59 B63 B66:  3   Q: 46
##  Median : 14.454   A34           :  2   S:270
##  Mean   : 35.627   B45           :  2
##  3rd Qu.: 31.500   C101          :  2
##  Max.   :512.329   C116          :  2
##  NA's   :1         (Other)       : 80
```

```
summary(gender)
```

```
##   PassengerId        Survived
##  Min.   : 892.0   Min.   :0.0000
##  1st Qu.: 996.2   1st Qu.:0.0000
##  Median :1100.5   Median :0.0000
##  Mean   :1100.5   Mean   :0.3636
##  3rd Qu.:1204.8   3rd Qu.:1.0000
##  Max.   :1309.0   Max.   :1.0000
```

Tras comprobar los resultados observamos que el dataset train está completo, pero el dateset test carece del campo “Survived” que si podemos ver en el dataset gender_submissionl, por tanto, combinaremos ambos dataset para obtener un dataset completo tanto en train como en test:

```
test1 <- merge(test, gender, by="PassengerId")
test1 = test1[,c(1,12,2:11)]
```

Cargaremos los ficheros en dataframes:

```
train <- as.data.frame(train)
test <- as.data.frame(test1)
```

Miraremos que coincidan las columnas de ambos dataframes:

```
colnames(train)
```

```
##  [1] "PassengerId" "Survived"    "Pclass"      "Name"        "Sex"
##  [6] "Age"         "SibSp"       "Parch"       "Ticket"      "Fare"
## [11] "Cabin"       "Embarked"
```

```
str(train)
```

```
## 'data.frame':    891 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 417
581 ...
##  $ 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/ 681 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/ 148 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 ...
```

```
colnames(test)
```

```
##  [1] "PassengerId" "Survived"    "Pclass"      "Name"        "Sex"
##  [6] "Age"         "SibSp"       "Parch"       "Ticket"      "Fare"
## [11] "Cabin"       "Embarked"
```

```
str(test)
```

```
## 'data.frame':    418 obs. of  12 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
##  $ Survived   : int  0 1 0 0 1 0 1 0 1 0 ...
##  $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
##  $ Name       : Factor w/ 418 levels "Abbott, Master. Eugene Joseph",..: 210 409 273 414 182 370 85
58 5 104 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 2 2 1 2 1 2 1 2 ...
##  $ Age        : num  34.5 47 62 27 22 14 30 26 18 21 ...
##  $ SibSp      : int  0 1 0 0 1 0 0 1 0 2 ...
##  $ Parch      : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ Ticket     : Factor w/ 363 levels "110469","110489",..: 153 222 74 148 139 262 159 85 101 270 ...
##  $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
##  $ Cabin      : Factor w/ 77 levels "","A11","A18",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 2 3 2 3 3 3 2 3 1 3 ...
```

3.- Integración y selección de los datos de interés a analizar.

Ahora ya podremos empezar a trabajar con ellos, empezamos aplicando la función summary para los dataframes para disponer una idea general.

```
summary(train)
```

```
##   PassengerId       Survived          Pclass
##  Min.   :  1.0   Min.   :0.0000   Min.   :1.000
##  1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000
##  Median :446.0   Median :0.0000   Median :3.000
##  Mean   :446.0   Mean   :0.3838   Mean   :2.309
##  3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :891.0   Max.   :1.0000   Max.   :3.000
##
##                                     Name         Sex           Age
##  Abbing, Mr. Anthony                   :  1   female:314   Min.   : 0.42
##  Abbott, Mr. Rossmore Edward           :  1   male  :577   1st Qu.:20.12
##  Abbott, Mrs. Stanton (Rosa Hunt)      :  1                Median :28.00
##  Abelson, Mr. Samuel                   :  1                Mean   :29.70
##  Abelson, Mrs. Samuel (Hannah Wizosky):  1                3rd Qu.:38.00
##  Adahl, Mr. Mauritz Nils Martin        :  1                Max.   :80.00
##  (Other)                               :885                NA's   :177
##      SibSp           Parch             Ticket         Fare
##  Min.   :0.000   Min.   :0.0000   1601    :  7   Min.   :  0.00
##  1st Qu.:0.000   1st Qu.:0.0000   347082  :  7   1st Qu.:  7.91
##  Median :0.000   Median :0.0000   CA. 2343:  7   Median : 14.45
##  Mean   :0.523   Mean   :0.3816   3101295 :  6   Mean   : 32.20
##  3rd Qu.:1.000   3rd Qu.:0.0000   347088  :  6   3rd Qu.: 31.00
##  Max.   :8.000   Max.   :6.0000   CA 2144 :  6   Max.   :512.33
##                                   (Other) :852
##          Cabin     Embarked
##              :687    :  2
##  B96 B98     :  4   C:168
##  C23 C25 C27:  4   Q: 77
##  G6          :  4   S:644
##  C22 C26     :  3
##  D           :  3
##  (Other)     :186
```

```
summary(test)
```

```
##   PassengerId        Survived          Pclass
##  Min.   : 892.0   Min.   :0.0000   Min.   :1.000
```

```
##  1st Qu.: 996.2   1st Qu.:0.0000   1st Qu.:1.000
##  Median :1100.5   Median :0.0000   Median :3.000
##  Mean   :1100.5   Mean   :0.3636   Mean   :2.266
##  3rd Qu.:1204.8   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :1309.0   Max.   :1.0000   Max.   :3.000
##
##                                          Name         Sex
##  Abbott, Master. Eugene Joseph             :  1   female:152
##  Abelseth, Miss. Karen Marie               :  1   male  :266
##  Abelseth, Mr. Olaus Jorgensen             :  1
##  Abrahamsson, Mr. Abraham August Johannes :  1
##  Abrahim, Mrs. Joseph (Sophie Halaut Easu):  1
##  Aks, Master. Philip Frank                 :  1
##  (Other)                                   :412
##       Age            SibSp            Parch             Ticket
##  Min.   : 0.17   Min.   :0.0000   Min.   :0.0000   PC 17608:  5
##  1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.0000   113503  :  4
##  Median :27.00   Median :0.0000   Median :0.0000   CA. 2343:  4
##  Mean   :30.27   Mean   :0.4474   Mean   :0.3923   16966   :  3
##  3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.0000   220845  :  3
##  Max.   :76.00   Max.   :8.0000   Max.   :9.0000   347077  :  3
##  NA's   :86                                        (Other) :396
##       Fare                    Cabin     Embarked
##  Min.   :  0.000                 :327   C:102
##  1st Qu.:  7.896   B57 B59 B63 B66:  3   Q: 46
##  Median : 14.454   A34           :  2   S:270
##  Mean   : 35.627   B45           :  2
##  3rd Qu.: 31.500   C101          :  2
##  Max.   :512.329   C116          :  2
##  NA's   :1         (Other)       : 80
```

```
head(train)
```

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

```
head(test)
```

```
##   PassengerId Survived Pclass                                      Name
## 1         892        0      3                          Kelly, Mr. James
## 2         893        1      3          Wilkes, Mrs. James (Ellen Needs)
## 3         894        0      2                 Myles, Mr. Thomas Francis
```

```
## 4         895        0      3                              Wirz, Mr. Albert
## 5         896        1      3 Hirvonen, Mrs. Alexander (Helga E Lindqvist)
## 6         897        0      3                    Svensson, Mr. Johan Cervin
##      Sex  Age SibSp Parch  Ticket    Fare Cabin Embarked
## 1   male 34.5     0     0  330911  7.8292              Q
## 2 female 47.0     1     0  363272  7.0000              S
## 3   male 62.0     0     0  240276  9.6875              Q
## 4   male 27.0     0     0  315154  8.6625              S
## 5 female 22.0     1     1 3101298 12.2875              S
## 6   male 14.0     0     0    7538  9.2250              S
```

Factorizamos variables numéricas Pclass y survived para test y para train;

```
train$Pclass <- as.factor(train$Pclass)
train$Survived <- as.factor(train$Survived)
test$Pclass <- as.factor(test$Pclass)
test$Survived <- as.factor(test$Survived)

sapply(train, function(x) class(x))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##   "integer"    "factor"    "factor"    "factor"    "factor"   "numeric"
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##   "integer"   "integer"    "factor"   "numeric"    "factor"    "factor"
```

```
sapply(test, function(x) class(x))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##   "integer"    "factor"    "factor"    "factor"    "factor"   "numeric"
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##   "integer"   "integer"    "factor"   "numeric"    "factor"    "factor"
```

3.1. Análisis de datos vacíos o nulos

Una vez tenemos las clases de las variables procedemos a estudiar los valores nulos antes de hacer el análisis de estas:

```
sapply(train, function(x) sum(is.na(x)))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0         177
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           0           0           0
```

```
sapply(test, function(x) sum(is.na(x)))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0          86
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           1           0           0
```

Podemos ver que solo presentamos valores nulos en la variable Age y en Fare en caso de test. Para tratarlos podríamos eliminar los registros nulos, pero esto haría que

perdiéramos muchos datos, por lo que procederemos a sustituir estos valores nulos por la media:

```
summary(train$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.42   20.12   28.00   29.70   38.00   80.00     177
```

```
summary(test$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.17   21.00   27.00   30.27   39.00   76.00      86
```

```
round(mean(train$Age, na.rm = TRUE),2)
```

```
## [1] 29.7
```

```
round(mean(test$Age, na.rm = TRUE),2)
```

```
## [1] 30.27
```

Podemos ver que las medias de edades son 29.68 Para train y 30.27 para test, entonces las sustituiremos por estas:

```
train <- train %>% mutate(Age = replace(Age, which(is.na(Age)),
round(mean(train$Age, na.rm = TRUE),2)))
test <- test %>% mutate(Age = replace(Age,
which(is.na(Age)),round(mean(test$Age, na.rm = TRUE),2)))
```

```
head(train)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                 Name    Sex  Age SibSp
## 1                            Braund, Mr. Owen Harris   male 22.0     1
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female 38.0     1
## 3                             Heikkinen, Miss. Laina female 26.0     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female 35.0     1
## 5                           Allen, Mr. William Henry   male 35.0     0
## 6                                   Moran, Mr. James   male 29.7     0
##   Parch           Ticket    Fare Cabin Embarked
## 1     0        A/5 21171  7.2500              S
## 2     0         PC 17599 71.2833   C85        C
## 3     0 STON/O2. 3101282  7.9250              S
## 4     0           113803 53.1000  C123        S
## 5     0           373450  8.0500              S
## 6     0           330877  8.4583              Q
```

```
head(test)
```

```
##   PassengerId Survived Pclass                                     Name
## 1         892        0      3                         Kelly, Mr. James
```

```
## 2         893        1      3             Wilkes, Mrs. James (Ellen Needs)
## 3         894        0      2                    Myles, Mr. Thomas Francis
## 4         895        0      3                             Wirz, Mr. Albert
## 5         896        1      3 Hirvonen, Mrs. Alexander (Helga E Lindqvist)
## 6         897        0      3                   Svensson, Mr. Johan Cervin
##      Sex  Age SibSp Parch  Ticket    Fare Cabin Embarked
## 1   male 34.5     0     0  330911  7.8292              Q
## 2 female 47.0     1     0  363272  7.0000              S
## 3   male 62.0     0     0  240276  9.6875              Q
## 4   male 27.0     0     0  315154  8.6625              S
## 5 female 22.0     1     1 3101298 12.2875              S
## 6   male 14.0     0     0    7538  9.2250              S
```

```
summary(test$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.17   23.00   30.27   30.27   35.75   76.00
```

```
test <- test %>% mutate(Fare = replace(Fare, which(is.na(Fare)),
round(mean(test$Fare, na.rm = TRUE),2)))
```

Y ahora haremos lo mismo con la variable Fare, para el único nulo que presenta:

```
summary(test$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   7.896  14.454  35.627  31.500 512.329
```

```
round(mean(test$Fare, na.rm = TRUE),2)
```

```
## [1] 35.63
```

Miraremos si tiene sentido primero el cambio de Fare, ya que tenemos que ver a qué clase corresponde este pasajero:

```
test[test$Fare == 35.63,]$PassengerId
```

```
## [1] 1044
```

```
test[test$PassengerId == 1044,]
```

```
##     PassengerId Survived Pclass               Name  Sex  Age SibSp Parch
## 153        1044        0      3 Storey, Mr. Thomas male 60.5     0     0
##     Ticket  Fare Cabin Embarked
## 153   3701 35.63              S
```

Podemos ver que el pasajero cuya cuota se ha substituido por la media, podría no corresponder a la 3a clase, entonces miramos cómo se distribuyen las cuotas:

```
summary(test[test$Pclass==3,]$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.171   7.750   7.896  12.566  14.441  69.550
```

```
summary(test[test$Pclass==2,]$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   9.688  13.000  15.750  22.202  26.000  73.500
```

```
summary(test[test$Pclass==1,]$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   30.10   60.00   94.28  134.50  512.33
```

Entonces el pasajero por el que se le ha cambiado el valor de Fare por la media, podría pertenecer perfectamente a la 3a clase, dado que está por debajo del máximo valor de Fare en la tercera clase, como podemos ver en el summary anterior.

```
summary(test$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   7.896  14.454  35.627  31.500 512.329
```

Para acabar con el análisis de Fare, podemos ver lo siguiente:

```
train[train$Fare == 0,]$Sex
```

```
##  [1] male male male male male male male male male male male male male male male
## [15] male
## Levels: female male
```

Sólo los hombres obtuvieron entradas gratuitas para el titanic, por lo que también podríamos substituir por 0 el valor nulo anterior. Pero se ha decidido por lo anterior.

```
sapply(train, function(x) sum(is.na(x)))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0           0
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           0           0           0
```

```
sapply(test, function(x) sum(is.na(x)))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0           0
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           0           0           0
```

Entonces podemos ver que el cambio se ha hecho correctamente.
Eliminaremos la variable PassengerId ya que no nos aporta nada:

```
train <- select(train, -PassengerId )
test <- select(test, -PassengerId )
```

También borraremos el ticket, la cabina, sibsp y parch, ya que sólo aportan información extra que no necesitamos, lo que nos interesa es relacionar la supervivencia con las variables que consideramos que tienen más peso en el estudio, como son Age, Pclass y Sex.

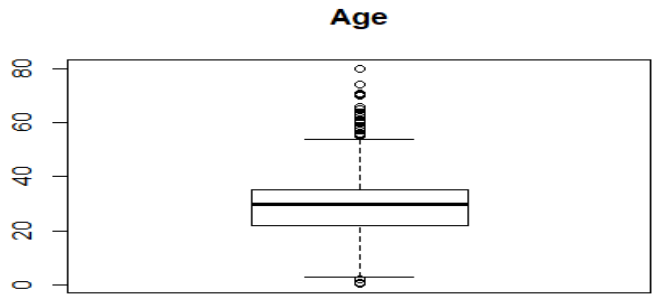
```
train <- select(train, -Ticket )
test <- select(test, -Ticket )
```

```
train <- select(train, -Cabin )
test <- select(test, -Cabin )

train <- select(train, -SibSp )
test <- select(test, -SibSp )

train <- select(train, -Parch )
test <- select(test, -Parch )
```

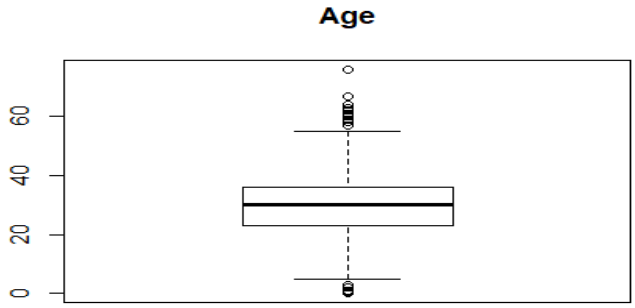
3.2. Tratamiento de valores extremos

Procederemos a analizar las variables para ambos datasets:

```
boxplot( train$Age, main="Age" )
```




```
boxplot( test$Age, main="Age" )
```



Podemos ver los valores extremos:

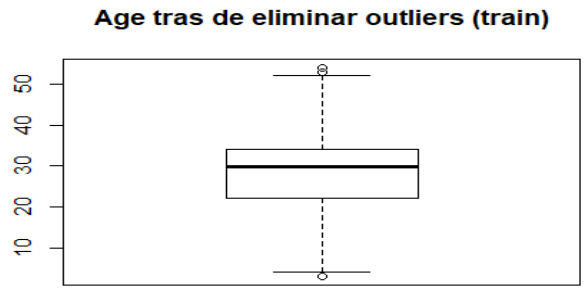
```
valuesX <- boxplot.stats(train$Age)$out
#miramos valores extremos en la variable Age:
cat("Valores extremos en Age de train:", toString(valuesX), "\n" )
```

```
## Valores extremos en Age de train: 2, 58, 55, 2, 66, 65, 0.83, 59, 71, 70.5, 2, 55.5,
1, 61, 1, 56, 1, 58, 2, 59, 62, 58, 63, 65, 2, 0.92, 61, 2, 60, 1, 1, 64, 65, 56, 0.75,
2, 63, 58, 55, 71, 2, 64, 62, 62, 60, 61, 57, 80, 2, 0.75, 56, 58, 70, 60, 60, 70, 0.67,
57, 1, 0.42, 2, 1, 62, 0.83, 74, 56
```

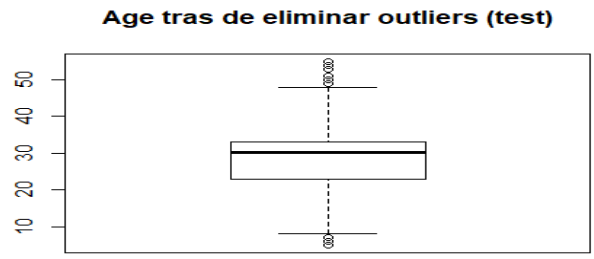
```
valuesY <- boxplot.stats(test$Age)$out
#miramos valores extremos en la variable Age:
cat("Valores extremos en Age de test:", toString(valuesY), "\n" )
```

```
## Valores extremos en Age de test: 62, 63, 60, 60, 67, 2, 76, 63, 1, 61, 60.5, 64, 61,
0.33, 60, 57, 64, 0.92, 1, 0.75, 2, 1, 64, 0.83, 57, 58, 0.17, 59, 57, 3
```

```
idxAgeX <- which( train$Age %in% valuesX)
idxAgeY <- which( test$Age %in% valuesY)

trainA <- train[-idxAgeX, ]
testA <- test[-idxAgeY,]
#Boxplot final:
boxplot( trainA$Age, main="Age tras de eliminar outliers (train)" )
```

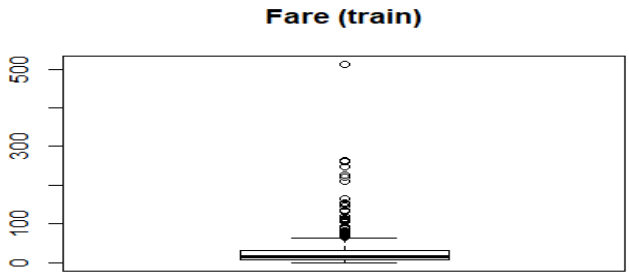



```
boxplot( testA$Age, main="Age tras de eliminar outliers (test)" )
```

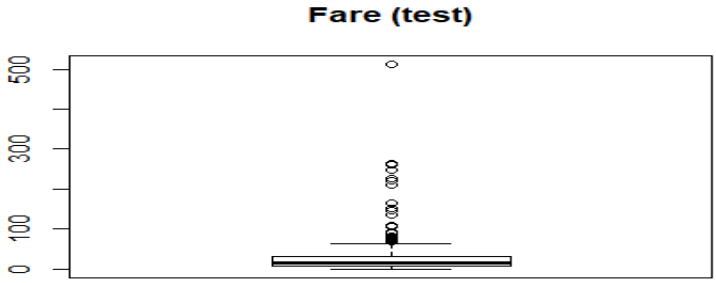


Podemos ver que las medias en ambos casos son muy similares, pero para analizar las variables, no eliminaremos estos valores extremos puesto que en la edad consideramos importantes mantener a estos registros con edades superiores a la media e inferiores para poder determinar la tasa de supervivencia según edades y si eliminamos ésta se podrá ver afectada.

```r
boxplot( train$Fare, main="Fare (train)" )
```




```r
boxplot( test1$Fare, main="Fare (test)" )
```



valores extremos:

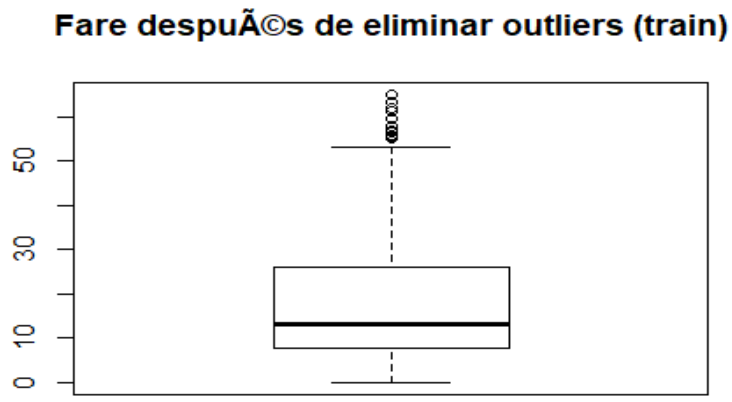
```r
values1X <- boxplot.stats(train$Fare)$out
#miramos valores extremos en la variable Age:
cat("Valores extremos en Fare (train):", toString(values1X), "\n" )
```

```
## Valores extremos en Fare (train): 71.2833, 263, 146.5208, 82.1708, 76.7292, 80,
83.475, 73.5, 263, 77.2875, 247.5208, 73.5, 77.2875, 79.2, 66.6, 69.55, 69.55, 146.5208,
69.55, 113.275, 76.2917, 90, 83.475, 90, 79.2, 86.5, 512.3292, 79.65, 153.4625, 135.6333,
77.9583, 78.85, 91.0792, 151.55, 247.5208, 151.55, 110.8833, 108.9, 83.1583, 262.375,
164.8667, 134.5, 69.55, 135.6333, 153.4625, 133.65, 66.6, 134.5, 263, 75.25, 69.3,
135.6333, 82.1708, 211.5, 227.525, 73.5, 120, 113.275, 90, 120, 263, 81.8583, 89.1042,
91.0792, 90, 78.2667, 151.55, 86.5, 108.9, 93.5, 221.7792, 106.425, 71, 106.425,
110.8833, 227.525, 79.65, 110.8833, 79.65, 79.2, 78.2667, 153.4625, 77.9583, 69.3,
76.7292, 73.5, 113.275, 133.65, 73.5, 512.3292, 76.7292, 211.3375, 110.8833, 227.525,
151.55, 227.525, 211.3375, 512.3292, 78.85, 262.375, 71, 86.5, 120, 77.9583, 211.3375,
79.2, 69.55, 120, 93.5, 80, 83.1583, 69.55, 89.1042, 164.8667, 69.55, 83.1583
```

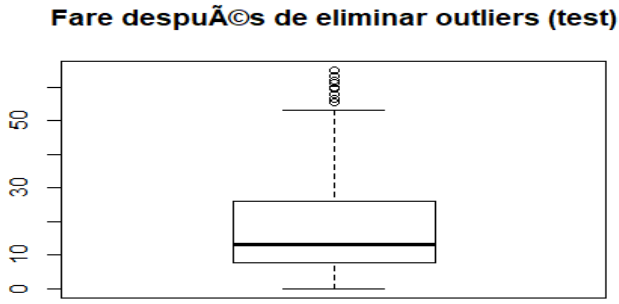
```r
values1Y <- boxplot.stats(test1$Fare)$out
#miramos valores extremos en la variable Age:
cat("Valores extremos en Fare (test):", toString(values1Y), "\n" )
```

```
## Valores extremos en Fare (test): 82.2667, 262.375, 76.2917, 263, 262.375, 262.375,
263, 211.5, 211.5, 221.7792, 78.85, 221.7792, 75.2417, 151.55, 262.375, 83.1583,
221.7792, 83.1583, 83.1583, 247.5208, 69.55, 134.5, 227.525, 73.5, 164.8667, 211.5,
71.2833, 75.25, 106.425, 134.5, 136.7792, 75.2417, 136.7792, 82.2667, 81.8583, 151.55,
```

```
93.5, 135.6333, 146.5208, 211.3375, 79.2, 69.55, 512.3292, 73.5, 69.55, 69.55, 134.5,
81.8583, 262.375, 93.5, 79.2, 164.8667, 211.5, 90, 108.9

idxFareX <- which( train$Fare %in% values1X)
idxFareY <- which( test1$Fare %in% values1Y)

FareX <- train[ -idxFareX, ]
FareY <- test1[ -idxFareY, ]

#Boxplot final:
boxplot( FareX$Fare, main="Fare despuÃ©s de eliminar outliers (train)" )
```




```
boxplot( FareY$Fare, main="Fare despuÃ©s de eliminar outliers (test)" )
```



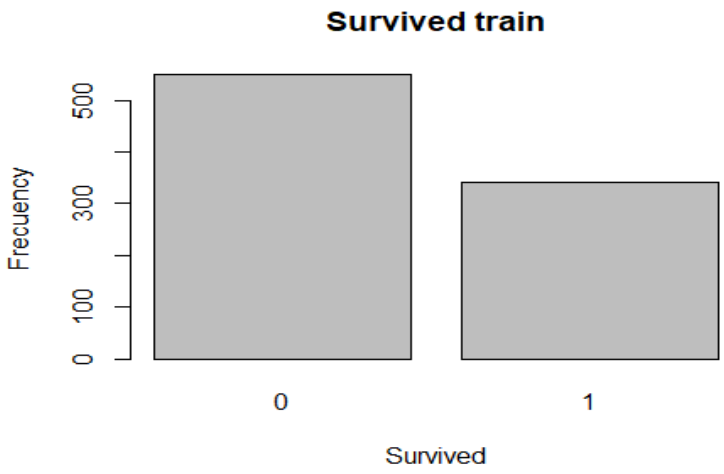
Podemos ver que no tiene sentido eliminar los valores extremos en Fare, igual que en Age, dado que por lógica podemos considerar que las tarifas de los pasajeros pueden variar dado que dependiendo de la clase a la que pertenezca puede ser más alto o bajo.

No nos interesa eliminar estos valores porque perderemos datos y cualquier pérdida de datos consideramos que afectará a nuestro estudio.

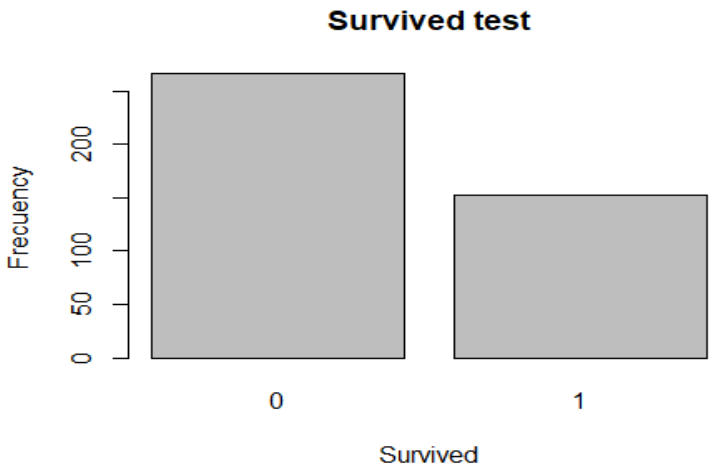
4.- Análisis de los datos.

4.1. Selección de los grupos de datos

```
barplot(table(train$Survived), main = "Survived train", ylab =
"Frecuency", xlab = "Survived")
```

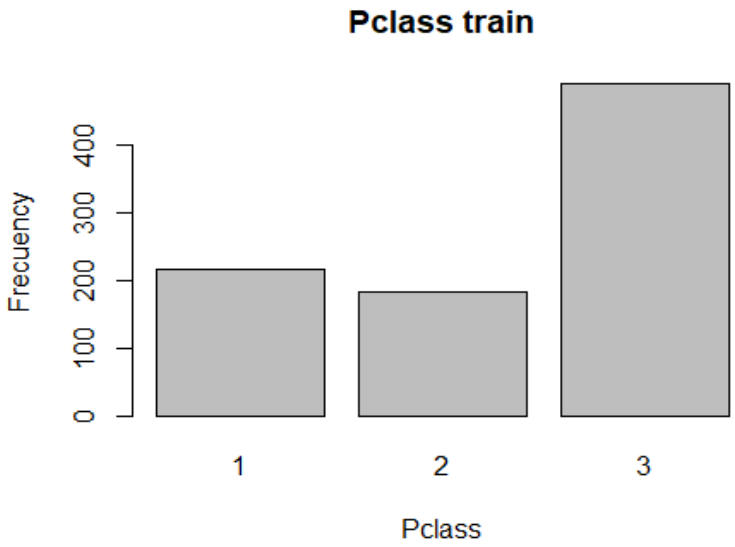



```
barplot(table(test$Survived), main = "Survived test", ylab = "Frecuency",
xlab = "Survived")
```

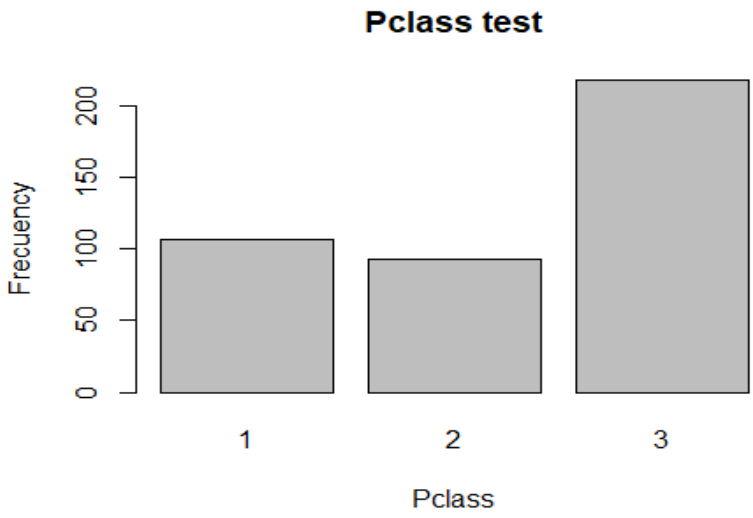


El número de registros para ambos dataframes es diferente, dado que tenemos más registros en train que en test, pero la clase mayoritaria en ambos casos es la que no sobrevivió a la catástrofe del Titanic. Miraremos por clase también, qué clase fue la mayoritaria de pasajeros:

```
barplot(table(train$Pclass), main = "Pclass train", ylab = "Frecuency",
xlab = "Pclass")
```




```
barplot(table(test$Pclass), main = "Pclass test", ylab = "Frecuency",
xlab = "Pclass")
```



La clase mayoritaria de pasajeros en el Titanic, fue la 3ra, como es lógico, la mayoría de clientes que entraron en el Titanic lo hicieron mediante la clase más económica.

4.2. Comprobación de la normalidad i homogeneidad de la variancia.

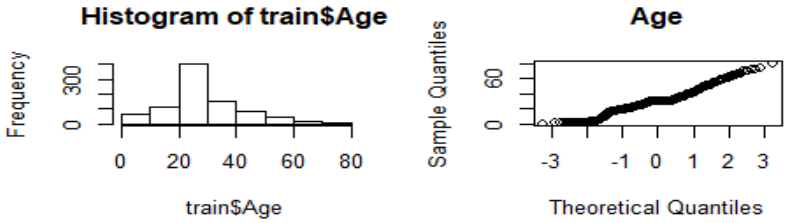
Vemos la distribución de edades para train:

```
summary(train$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.42   22.00   29.70   29.70   35.00   80.00
```

```
par(mfrow=c(2,2))
hist(train$Age)
qqnorm(train$Age, main="Age")
qqline
```

```
## function (y, datax = FALSE, distribution = qnorm, probs = c(0.25,
##     0.75), qtype = 7, ...)
## {
##     stopifnot(length(probs) == 2, is.function(distribution))
##     y <- quantile(y, probs, names = FALSE, type = qtype, na.rm = TRUE)
##     x <- distribution(probs)
##     if (datax) {
##         slope <- diff(x)/diff(y)
##         int <- x[1L] - slope * y[1L]
##     }
##     else {
##         slope <- diff(y)/diff(x)
##         int <- y[1L] - slope * x[1L]
##     }
##     abline(int, slope, ...)
## }
## <bytecode: 0x00000000155b4670>
## <environment: namespace:stats>
```



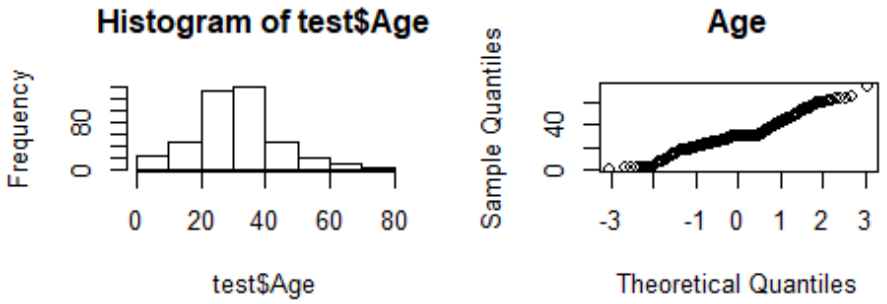
Vemos la distribución de edades para test:

```
summary(test$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.17   23.00   30.27   30.27   35.75   76.00
```

```
par(mfrow=c(2,2))
hist(test$Age)
qqnorm(test$Age, main="Age")
qqline
```

```
## function (y, datax = FALSE, distribution = qnorm, probs = c(0.25,
##     0.75), qtype = 7, ...)
## {
##     stopifnot(length(probs) == 2, is.function(distribution))
##     y <- quantile(y, probs, names = FALSE, type = qtype, na.rm = TRUE)
##     x <- distribution(probs)
##     if (datax) {
##         slope <- diff(x)/diff(y)
##         int <- x[1L] - slope * y[1L]
##     }
##     else {
##         slope <- diff(y)/diff(x)
##         int <- y[1L] - slope * x[1L]
##     }
##     abline(int, slope, ...)
## }
## <bytecode: 0x00000000155b4670>
## <environment: namespace:stats>
```



Valoramos las variables que no siguen una distribución normal para test:

```
library(nortest)
alpha = 0.05
col.names = colnames(test)
for (i in 1:ncol(test)) {
    if (i == 1) cat("Variables que no siguen una distribución normal:\n")
    if (is.integer(test[,i]) | is.numeric(test[,i])) {
      p_val = ad.test(test[,i])$p.value
      if (p_val < alpha) {
        cat(col.names[i])
        # Format output
        if (i < ncol(test) - 1) cat(", ")
        if (i %% 3 == 0) cat("\n")
      }
     }
    }
```

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

Ahora realizamos el test de homogeneidad para **test**(Age ~ Survived)

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

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Age by Survived
```

```
## Fligner-Killeen:med chi-squared = 6.0438, df = 1, p-value =
## 0.01396
```

Valoramos las variables que no siguen una distribución normal para train:

```
library(nortest)
alpha = 0.05
col.names = colnames(train)
for (i in 1:ncol(test)) {
    if (i == 1) cat("Variables que no siguen una distribución normal:\n")
    if (is.integer(train[,i]) | is.numeric(train[,i])) {
      p_val = ad.test(train[,i])$p.value
      if (p_val < alpha) {
        cat(col.names[i])
        # Format output
        if (i < ncol(train) - 1) cat(", ")
        if (i %% 3 == 0) cat("\n")
      }
     }
    }
```

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

Ahora realizamos el test de homogeneidad para train(Age ~ Survived)

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

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

4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos

Pruebas estadísticas, vemos la relación entre Survived con Age y Fare, teniendo en cuenta que Fare de forma numérica podría identificar la clase de cada pasajero en función del coste aplicado a cada tarifa, realizaremos el test de Wilcox:

```
wilcox.test(train$Age~train$Survived)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  train$Age by train$Survived
## W = 98220, p-value = 0.2434
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(train$Fare~train$Survived)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  train$Fare by train$Survived
## W = 57806, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Como resultado obtenemos que efectivamente existen diferencias significativas entre los grupos de edad, tal y como veremos con más detalle en las siguientes pruebas.

Ahora aplicaremos para el test de de Kruskal-Wallis para la variable clase:

```
kruskal.test(train$Pclass~train$Survived)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  train$Pclass by train$Survived
## Kruskal-Wallis chi-squared = 102.68, df = 1, p-value < 2.2e-16
```

Como resultado obtenemos que estadisticamente tenemos diferencias entre los grupos de clase en contrate con la supervivencia de los pasajeros

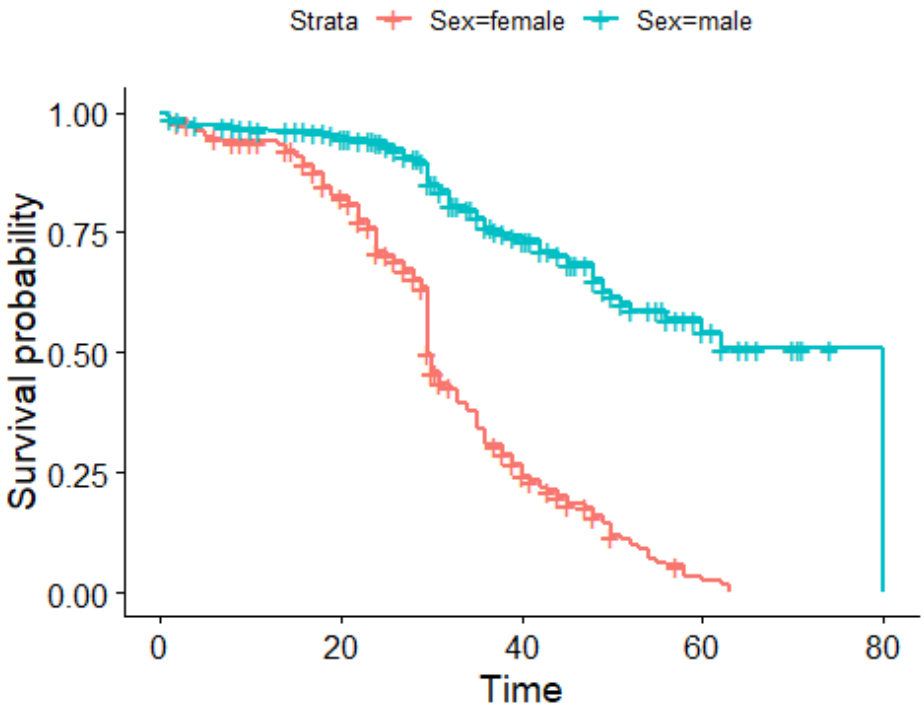
Aplicamos el test de supervivencia para ver la correlación entre Edad y Sexo

```
library(survival)
library("survminer")
```

```
## Loading required package: ggpubr
```

```
## Loading required package: magrittr
```

```
require("survival")

fit <- survfit(Surv(train$Age,train$Survived==1) ~ train$Sex, data =
train)
ggsurvplot(fit, data = train)
```



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

Nos interesa ver la relación de la clase Survived con el resto de las variables, como la de la clase a la que pertenecían, la edad y el sexo:

Para train;

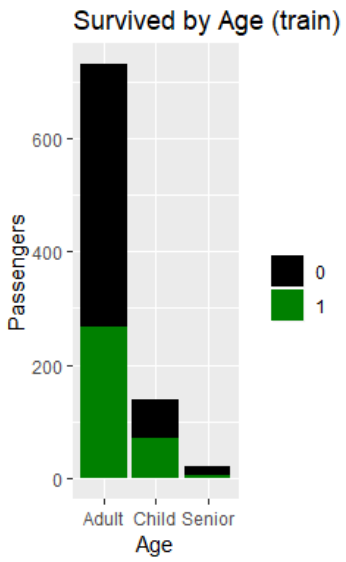
```
grid.newpage()
plotbyClass_train<-ggplot(train,aes(Pclass,fill=Survived))+geom_bar()
+labs(x="Class", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by
PClass(train)")

plotbyAge_train<-ggplot(train,aes(Age,fill=Survived))+geom_bar()
```

```
+labs(x="Age", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by Age
(train)")

plotbySex_train<-ggplot(train,aes(Sex,fill=Survived))+geom_bar()
+labs(x="Sex", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by Sex
(train)")

plotbyEmbarked_train<-
ggplot(train,aes(Embarked,fill=Survived))+geom_bar() +labs(x="Embarked",
y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by
Embarked (train)")

grid.arrange(plotbyClass_train,plotbyAge_train,plotbySex_train,
plotbyEmbarked_train,ncol=2)

## Warning: position_stack requires non-overlapping x intervals
```

Podemos ver que en Pclass, la clase con más víctimas es la mayoritaria, es decir, la clase más económica, pero también podemos ver que la proporción de víctimas y supervivientes es la que más se ve afectada… ya que más del 50% de los pasajeros de la 3a clase murieron. Por lo que hace a la variable Sex, la proporción con más supervivientes es la de mujeres mientras que la proporción de supervivientes masculinos es todo lo

contrario. Por puerto de embarcación, podemos ver que pasa lo mismo que con la variable PClass, que tenemos un puerto, concretamente el de Southampton, con la mayor parte de pasajeros, entonces la proporción se acerca a lo que se ha visto anteriormente con la 3ra Clase. Finalmente tenemos a la variable Age, que tendremos que agrupar en 3 segmentos, separándola por rangos de Edad, ya que como se ha hecho hasta ahora, para visualizarlo es muy complicado de poder determinar la tasa de supervivencia y entonces lo cambiaremos por child, adult y senior (menores o iguales a 18 años serán los jóvenes, seguido de los adultos hasta los 60 años)

```
trainAge <-train
trainAge$Age[trainAge$Age <=18] = "Child"
trainAge$Age[(trainAge$Age > 18) & (trainAge$Age <=60) & (trainAge$Age != "Child")] = "Adult"
trainAge$Age[(trainAge$Age != "Child") & (trainAge$Age != "Adult")] = "Senior"
trainAge$Age = as.factor(trainAge$Age)

grid.newpage()

NewplotbyAge_train<-ggplot(trainAge,aes(Age,fill=Survived))+geom_bar()
+labs(x="Age", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by Age (train)")


grid.arrange(NewplotbyAge_train,ncol=2)
```



Entonces podemos ver perfectamente que la mayoría de los pasajeros eran Adultos y la mayoría de ellos fueron víctimas, esto podemos atribuirlo a que la mayoría de pasajeros no sobrevivieron, mientras que en pasajeros entre 0-18 años, la proporción se ve más

repartida que en Adult. Por último, tenemos a Senior, que son minoría, podemos ver que la proporción de víctimas es mayor a la de supervivientes.

Una vez mostradas las gráficas, podemos ver las proporciones:

```
table_SexSurvived <- table(trainAge$Sex, trainAge$Survived)
prop.table(table_SexSurvived, margin = 1)
```

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

Podemos ver que un 74% de la población de mujeres sobrevivió a la catastrofe mientras que sólo el 18% de la población masculina lo hizo.

Sguiremos con el estudio de la clase:

```
table_PclassSurvived <- table(trainAge$Pclass,train$Survived)
prop.table(table_PclassSurvived, margin = 1)
```

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

Como se ha comentado anteriormente, podemos ver que el 76% de los pasajeros pertenecientes a la 3a clase, fueron víctimas. Podemos ver que mientras subimos de clase (de 3a a 1a) vamos encontrando una mayor tasa de supervivencia que podríamos decir que esta se debe a que cada vez que subimos de clase, nos encontramos con menos pasajeros.

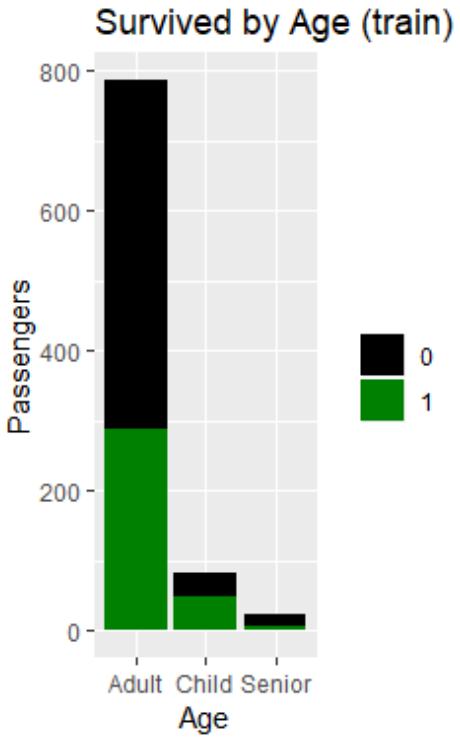
```
table_AgeSurvived <- table(trainAge$Age,train$Survived)
prop.table(table_AgeSurvived, margin = 1)
```

```
##
##                  0         1
##   Adult  0.6342466 0.3657534
##   Child  0.4964029 0.5035971
##   Senior 0.7727273 0.2272727
```

Podemos ver que los niños sobrevivieron en un 50% más o menos, debido a que se ha elegido la edad de 18 años como niño también y podríamos ver, en los siguientes cálculos que, si lo cambiamos a niño hasta la edad de 15, que podría cambiar:

```
trainAge <- train
trainAge$Age[trainAge$Age <=15] = "Child"
trainAge$Age[(trainAge$Age > 15) & (trainAge$Age <=60) & (trainAge$Age !=
"Child")] = "Adult"
trainAge$Age[(trainAge$Age != "Child") & (trainAge$Age != "Adult")] =
"Senior"
trainAge$Age = as.factor(trainAge$Age)
```

```
grid.newpage()

NewplotbyAge_train<-ggplot(trainAge,aes(Age,fill=Survived))+geom_bar()
+labs(x="Age", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by Age
(train)")


grid.arrange(NewplotbyAge_train,ncol=2)
```



Entonces podemos ver perfectamente que la mayoría de los pasajeros eran Adultos y la mayoría de ellos fueron víctimas, esto podemos atribuirlo a que la mayoría de los pasajeros no sobrevivieron, mientras que en pasajeros entre 0-18 años, la proporción se ve más repartida que en Adult. Por último, tenemos a Senior, que son minoría, podemos ver que la proporción de víctimas es mayor a la de supervivientes.

Una vez mostradas las gráficas, podemos ver las proporciones:

```
table_SexSurvived <- table(trainAge$Sex, trainAge$Survived)
prop.table(table_SexSurvived, margin = 1)
```

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

Podemos ver que un 74% de la población de mujeres sobrevivió a la catástrofe mientras que sólo el 18% de la población masculina lo hizo…

Seguiremos con el estudio de la clase:

```
table_PclassSurvived <- table(trainAge$Pclass,train$Survived)
prop.table(table_PclassSurvived, margin = 1)
```

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

Como se ha comentado anteriormente, podemos ver que el 76% de los pasajeros pertenecientes a la 3a clase, fueron víctimas. Podemos ver que mientras subimos de clase (de 3a a 1a) vamos encontrando una mayor tasa de supervivencia que podríamos decir que esta se debe a que cada vez que subimos de clase, nos encontramos con menos pasajeros.

```
table_AgeSurvived <- table(trainAge$Age,train$Survived)
prop.table(table_AgeSurvived, margin = 1)
```

```
##
##                  0         1
##   Adult  0.6335878 0.3664122
##   Child  0.4096386 0.5903614
##   Senior 0.7727273 0.2272727
```

Podemos ver que éste cambio representa un incremento irrelevante, dado que se pensaba que incrementaríamos mucho más la tasa de supervivencia de la clase niños, por lo que sólo nos fijaremos en lo representado anteriormente.

Y lo combinaremos con la clase, la edad (child -> <=18) y la supervivéncia.

```
tablebyAClass <- table(train$Age,train$Survived,train$Pclass)
tablebyAClass
```

```
## , ,  = 1
##
##
##          0   1
##   0.42   0   0
##   0.67   0   0
##   0.75   0   0
##   0.83   0   0
##   0.92   0   1
##   1      0   0
##   2      1   0
##   3      0   0
##   4      0   1
##   5      0   0
##   6      0   0
##   7      0   0
##   8      0   0
##   9      0   0
```

```
##   10     0   0
##   11     0   1
##   12     0   0
##   13     0   0
##   14     0   1
##   14.5   0   0
##   15     0   1
##   16     0   3
##   17     0   3
##   18     1   3
##   19     2   3
##   20     0   0
##   20.5   0   0
##   21     1   2
##   22     1   4
##   23     0   3
##   23.5   0   0
##   24     2   5
##   24.5   0   0
##   25     1   2
##   26     0   2
##   27     1   3
##   28     2   2
##   28.5   0   0
##   29     2   1
##   29.7  16  14
##   30     1   5
##   30.5   0   0
##   31     2   3
##   32     0   2
##   32.5   0   0
##   33     1   3
##   34     0   1
##   34.5   0   0
##   35     0   9
##   36     2   7
##   36.5   0   0
##   37     2   1
##   38     2   4
##   39     1   4
##   40     2   3
##   40.5   0   0
##   41     0   1
##   42     1   3
##   43     0   1
##   44     1   2
##   45     3   2
##   45.5   1   0
##   46     2   0
##   47     4   1
##   48     0   5
##   49     1   4
##   50     3   2
##   51     1   2
##   52     1   3
##   53     0   1
##   54     2   2
##   55     1   0
##   55.5   0   0
##   56     2   2
##   57     0   0
```

```
##   58     2   3
##   59     0   0
##   60     1   2
##   61     2   0
##   62     2   1
##   63     0   1
##   64     2   0
##   65     2   0
##   66     0   0
##   70     1   0
##   70.5   0   0
##   71     2   0
##   74     0   0
##   80     0   1
##
## , ,  = 2
##
##
##          0   1
##   0.42   0   0
##   0.67   0   1
##   0.75   0   0
##   0.83   0   2
##   0.92   0   0
##   1      0   2
##   2      0   2
##   3      0   3
##   4      0   2
##   5      0   1
##   6      0   1
##   7      0   1
##   8      0   2
##   9      0   0
##   10     0   0
##   11     0   0
##   12     0   0
##   13     0   1
##   14     0   1
##   14.5   0   0
##   15     0   0
##   16     2   0
##   17     0   2
##   18     4   2
##   19     3   3
##   20     0   0
##   20.5   0   0
##   21     3   1
##   22     0   2
##   23     6   1
##   23.5   0   0
##   24     4   6
##   24.5   0   0
##   25     5   2
##   26     2   0
##   27     4   2
##   28     4   5
##   28.5   0   0
##   29     3   3
##   29.7   7   4
##   30     5   3
##   30.5   0   0
```

```
##   31     3   2
##   32     2   2
##   32.5   1   1
##   33     1   2
##   34     5   5
##   34.5   0   0
##   35     2   1
##   36     4   3
##   36.5   1   0
##   37     1   0
##   38     1   0
##   39     3   0
##   40     0   3
##   40.5   0   0
##   41     0   1
##   42     2   3
##   43     1   0
##   44     2   0
##   45     0   2
##   45.5   0   0
##   46     1   0
##   47     1   0
##   48     1   1
##   49     0   0
##   50     1   3
##   51     1   0
##   52     2   0
##   53     0   0
##   54     3   1
##   55     0   1
##   55.5   0   0
##   56     0   0
##   57     2   0
##   58     0   0
##   59     1   0
##   60     1   0
##   61     0   0
##   62     0   1
##   63     0   0
##   64     0   0
##   65     0   0
##   66     1   0
##   70     1   0
##   70.5   0   0
##   71     0   0
##   74     0   0
##   80     0   0
##
## , ,  = 3
##
##
##          0   1
##   0.42   0   1
##   0.67   0   0
##   0.75   0   2
##   0.83   0   0
##   0.92   0   0
##   1      2   3
##   2      6   1
##   3      1   2
##   4      3   4
```

```
##   5      0   3
##   6      1   1
##   7      2   0
##   8      2   0
##   9      6   2
##   10     2   0
##   11     3   0
##   12     0   1
##   13     0   1
##   14     3   1
##   14.5   1   0
##   15     1   3
##   16     9   3
##   17     7   1
##   18    12   4
##   19    11   3
##   20    12   3
##   20.5   1   0
##   21    15   2
##   22    15   5
##   23     4   1
##   23.5   1   0
##   24     9   4
##   24.5   1   0
##   25    11   2
##   26    10   4
##   27     2   6
##   28    12   0
##   28.5   2   0
##   29     7   4
##   29.7 102  34
##   30     9   2
##   30.5   2   0
##   31     4   3
##   32     7   5
##   32.5   0   0
##   33     7   1
##   34     4   0
##   34.5   1   0
##   35     5   1
##   36     5   1
##   36.5   0   0
##   37     2   0
##   38     3   1
##   39     5   1
##   40     5   0
##   40.5   2   0
##   41     4   0
##   42     4   0
##   43     3   0
##   44     3   1
##   45     4   1
##   45.5   1   0
##   46     0   0
##   47     3   0
##   48     2   0
##   49     1   0
##   50     1   0
##   51     3   0
##   52     0   0
##   53     0   0
```

```
##   54     0   0
##   55     0   0
##   55.5   1   0
##   56     0   0
##   57     0   0
##   58     0   0
##   59     1   0
##   60     0   0
##   61     1   0
##   62     0   0
##   63     0   1
##   64     0   0
##   65     1   0
##   66     0   0
##   70     0   0
##   70.5   1   0
##   71     0   0
##   74     1   0
##   80     0   0
```

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

```
## , ,  = 1
##
##
##                 0          1
##   0.42 0.00000000 0.00000000
##   0.67 0.00000000 0.00000000
##   0.75 0.00000000 0.00000000
##   0.83 0.00000000 0.00000000
##   0.92 0.00000000 1.00000000
##   1    0.00000000 0.00000000
##   2    0.10000000 0.00000000
##   3    0.00000000 0.00000000
##   4    0.00000000 0.10000000
##   5    0.00000000 0.00000000
##   6    0.00000000 0.00000000
##   7    0.00000000 0.00000000
##   8    0.00000000 0.00000000
##   9    0.00000000 0.00000000
##   10   0.00000000 0.00000000
##   11   0.00000000 0.25000000
##   12   0.00000000 0.00000000
##   13   0.00000000 0.00000000
##   14   0.00000000 0.16666667
##   14.5 0.00000000 0.00000000
##   15   0.00000000 0.20000000
##   16   0.00000000 0.17647059
##   17   0.00000000 0.23076923
##   18   0.03846154 0.11538462
##   19   0.08000000 0.12000000
##   20   0.00000000 0.00000000
##   20.5 0.00000000 0.00000000
##   21   0.04166667 0.08333333
##   22   0.03703704 0.14814815
##   23   0.00000000 0.20000000
##   23.5 0.00000000 0.00000000
##   24   0.06666667 0.16666667
##   24.5 0.00000000 0.00000000
##   25   0.04347826 0.08695652
##   26   0.00000000 0.11111111
```

```
##   27   0.05555556 0.16666667
##   28   0.08000000 0.08000000
##   28.5 0.00000000 0.00000000
##   29   0.10000000 0.05000000
##   29.7 0.09039548 0.07909605
##   30   0.04000000 0.20000000
##   30.5 0.00000000 0.00000000
##   31   0.11764706 0.17647059
##   32   0.00000000 0.11111111
##   32.5 0.00000000 0.00000000
##   33   0.06666667 0.20000000
##   34   0.00000000 0.06666667
##   34.5 0.00000000 0.00000000
##   35   0.00000000 0.50000000
##   36   0.09090909 0.31818182
##   36.5 0.00000000 0.00000000
##   37   0.33333333 0.16666667
##   38   0.18181818 0.36363636
##   39   0.07142857 0.28571429
##   40   0.15384615 0.23076923
##   40.5 0.00000000 0.00000000
##   41   0.00000000 0.16666667
##   42   0.07692308 0.23076923
##   43   0.00000000 0.20000000
##   44   0.11111111 0.22222222
##   45   0.25000000 0.16666667
##   45.5 0.50000000 0.00000000
##   46   0.66666667 0.00000000
##   47   0.44444444 0.11111111
##   48   0.00000000 0.55555556
##   49   0.16666667 0.66666667
##   50   0.30000000 0.20000000
##   51   0.14285714 0.28571429
##   52   0.16666667 0.50000000
##   53   0.00000000 1.00000000
##   54   0.25000000 0.25000000
##   55   0.50000000 0.00000000
##   55.5 0.00000000 0.00000000
##   56   0.50000000 0.50000000
##   57   0.00000000 0.00000000
##   58   0.40000000 0.60000000
##   59   0.00000000 0.00000000
##   60   0.25000000 0.50000000
##   61   0.66666667 0.00000000
##   62   0.50000000 0.25000000
##   63   0.00000000 0.50000000
##   64   1.00000000 0.00000000
##   65   0.66666667 0.00000000
##   66   0.00000000 0.00000000
##   70   0.50000000 0.00000000
##   70.5 0.00000000 0.00000000
##   71   1.00000000 0.00000000
##   74   0.00000000 0.00000000
##   80   0.00000000 1.00000000
##
## , ,  = 2
##
##
##                0          1
##   0.42 0.00000000 0.00000000
##   0.67 0.00000000 1.00000000
```

```
##   0.75 0.00000000 0.00000000
##   0.83 0.00000000 1.00000000
##   0.92 0.00000000 0.00000000
##   1    0.00000000 0.28571429
##   2    0.00000000 0.20000000
##   3    0.00000000 0.50000000
##   4    0.00000000 0.20000000
##   5    0.00000000 0.25000000
##   6    0.00000000 0.33333333
##   7    0.00000000 0.33333333
##   8    0.00000000 0.50000000
##   9    0.00000000 0.00000000
##   10   0.00000000 0.00000000
##   11   0.00000000 0.00000000
##   12   0.00000000 0.00000000
##   13   0.00000000 0.50000000
##   14   0.00000000 0.16666667
##   14.5 0.00000000 0.00000000
##   15   0.00000000 0.00000000
##   16   0.11764706 0.00000000
##   17   0.00000000 0.15384615
##   18   0.15384615 0.07692308
##   19   0.12000000 0.12000000
##   20   0.00000000 0.00000000
##   20.5 0.00000000 0.00000000
##   21   0.12500000 0.04166667
##   22   0.00000000 0.07407407
##   23   0.40000000 0.06666667
##   23.5 0.00000000 0.00000000
##   24   0.13333333 0.20000000
##   24.5 0.00000000 0.00000000
##   25   0.21739130 0.08695652
##   26   0.11111111 0.00000000
##   27   0.22222222 0.11111111
##   28   0.16000000 0.20000000
##   28.5 0.00000000 0.00000000
##   29   0.15000000 0.15000000
##   29.7 0.03954802 0.02259887
##   30   0.20000000 0.12000000
##   30.5 0.00000000 0.00000000
##   31   0.17647059 0.11764706
##   32   0.11111111 0.11111111
##   32.5 0.50000000 0.50000000
##   33   0.06666667 0.13333333
##   34   0.33333333 0.33333333
##   34.5 0.00000000 0.00000000
##   35   0.11111111 0.05555556
##   36   0.18181818 0.13636364
##   36.5 1.00000000 0.00000000
##   37   0.16666667 0.00000000
##   38   0.09090909 0.00000000
##   39   0.21428571 0.00000000
##   40   0.00000000 0.23076923
##   40.5 0.00000000 0.00000000
##   41   0.00000000 0.16666667
##   42   0.15384615 0.23076923
##   43   0.20000000 0.00000000
##   44   0.22222222 0.00000000
##   45   0.00000000 0.16666667
##   45.5 0.00000000 0.00000000
##   46   0.33333333 0.00000000
```

```
##   47   0.11111111 0.00000000
##   48   0.11111111 0.11111111
##   49   0.00000000 0.00000000
##   50   0.10000000 0.30000000
##   51   0.14285714 0.00000000
##   52   0.33333333 0.00000000
##   53   0.00000000 0.00000000
##   54   0.37500000 0.12500000
##   55   0.00000000 0.50000000
##   55.5 0.00000000 0.00000000
##   56   0.00000000 0.00000000
##   57   1.00000000 0.00000000
##   58   0.00000000 0.00000000
##   59   0.50000000 0.00000000
##   60   0.25000000 0.00000000
##   61   0.00000000 0.00000000
##   62   0.00000000 0.25000000
##   63   0.00000000 0.00000000
##   64   0.00000000 0.00000000
##   65   0.00000000 0.00000000
##   66   1.00000000 0.00000000
##   70   0.50000000 0.00000000
##   70.5 0.00000000 0.00000000
##   71   0.00000000 0.00000000
##   74   0.00000000 0.00000000
##   80   0.00000000 0.00000000
##
## , ,  = 3
##
##
##                 0          1
##   0.42 0.00000000 1.00000000
##   0.67 0.00000000 0.00000000
##   0.75 0.00000000 1.00000000
##   0.83 0.00000000 0.00000000
##   0.92 0.00000000 0.00000000
##   1    0.28571429 0.42857143
##   2    0.60000000 0.10000000
##   3    0.16666667 0.33333333
##   4    0.30000000 0.40000000
##   5    0.00000000 0.75000000
##   6    0.33333333 0.33333333
##   7    0.66666667 0.00000000
##   8    0.50000000 0.00000000
##   9    0.75000000 0.25000000
##   10   1.00000000 0.00000000
##   11   0.75000000 0.00000000
##   12   0.00000000 1.00000000
##   13   0.00000000 0.50000000
##   14   0.50000000 0.16666667
##   14.5 1.00000000 0.00000000
##   15   0.20000000 0.60000000
##   16   0.52941176 0.17647059
##   17   0.53846154 0.07692308
##   18   0.46153846 0.15384615
##   19   0.44000000 0.12000000
##   20   0.80000000 0.20000000
##   20.5 1.00000000 0.00000000
##   21   0.62500000 0.08333333
##   22   0.55555556 0.18518519
##   23   0.26666667 0.06666667
```

```
##   23.5 1.00000000 0.00000000
##   24   0.30000000 0.13333333
##   24.5 1.00000000 0.00000000
##   25   0.47826087 0.08695652
##   26   0.55555556 0.22222222
##   27   0.11111111 0.33333333
##   28   0.48000000 0.00000000
##   28.5 1.00000000 0.00000000
##   29   0.35000000 0.20000000
##   29.7 0.57627119 0.19209040
##   30   0.36000000 0.08000000
##   30.5 1.00000000 0.00000000
##   31   0.23529412 0.17647059
##   32   0.38888889 0.27777778
##   32.5 0.00000000 0.00000000
##   33   0.46666667 0.06666667
##   34   0.26666667 0.00000000
##   34.5 1.00000000 0.00000000
##   35   0.27777778 0.05555556
##   36   0.22727273 0.04545455
##   36.5 0.00000000 0.00000000
##   37   0.33333333 0.00000000
##   38   0.27272727 0.09090909
##   39   0.35714286 0.07142857
##   40   0.38461538 0.00000000
##   40.5 1.00000000 0.00000000
##   41   0.66666667 0.00000000
##   42   0.30769231 0.00000000
##   43   0.60000000 0.00000000
##   44   0.33333333 0.11111111
##   45   0.33333333 0.08333333
##   45.5 0.50000000 0.00000000
##   46   0.00000000 0.00000000
##   47   0.33333333 0.00000000
##   48   0.22222222 0.00000000
##   49   0.16666667 0.00000000
##   50   0.10000000 0.00000000
##   51   0.42857143 0.00000000
##   52   0.00000000 0.00000000
##   53   0.00000000 0.00000000
##   54   0.00000000 0.00000000
##   55   0.00000000 0.00000000
##   55.5 1.00000000 0.00000000
##   56   0.00000000 0.00000000
##   57   0.00000000 0.00000000
##   58   0.00000000 0.00000000
##   59   0.50000000 0.00000000
##   60   0.00000000 0.00000000
##   61   0.33333333 0.00000000
##   62   0.00000000 0.00000000
##   63   0.00000000 0.50000000
##   64   0.00000000 0.00000000
##   65   0.33333333 0.00000000
##   66   0.00000000 0.00000000
##   70   0.00000000 0.00000000
##   70.5 1.00000000 0.00000000
##   71   0.00000000 0.00000000
##   74   1.00000000 0.00000000
##   80   0.00000000 0.00000000
```

Como se ha visto anteriormente, la 3a clase es la que más víctimas tiene, por lo que, si nos fijamos en los niños, podemos ver que cada vez que aumentamos la clase, tenemos que la supervivencia de los incrementa de manera significativa.

```
tablebyBClass <- table(train$Age,train$Survived,train$Sex)
tablebyBClass
```

```
## , ,  = female
##
##
##          0   1
##   0.42   0   0
##   0.67   0   0
##   0.75   0   2
##   0.83   0   0
##   0.92   0   0
##   1      0   2
##   2      4   2
##   3      1   1
##   4      0   5
##   5      0   4
##   6      1   1
##   7      0   1
##   8      1   1
##   9      4   0
##   10     1   0
##   11     1   0
##   12     0   0
##   13     0   2
##   14     1   3
##   14.5   1   0
##   15     0   4
##   16     1   5
##   17     1   5
##   18     5   8
##   19     0   7
##   20     2   0
##   20.5   0   0
##   21     3   4
##   22     2  10
##   23     1   4
##   23.5   0   0
##   24     2  14
##   24.5   0   0
##   25     3   2
##   26     2   3
##   27     1   5
##   28     2   5
##   28.5   0   0
##   29     2   5
##   29.7  17  36
##   30     2   9
##   30.5   1   0
##   31     2   5
##   32     1   2
##   32.5   0   1
##   33     0   6
##   34     0   4
##   34.5   0   0
##   35     0   8
```

```
##   36     0   7
##   36.5   0   0
##   37     1   0
##   38     1   4
##   39     2   4
##   40     1   5
##   40.5   0   0
##   41     2   2
##   42     0   3
##   43     1   1
##   44     1   2
##   45     3   3
##   45.5   0   0
##   46     0   0
##   47     1   1
##   48     1   3
##   49     0   2
##   50     1   4
##   51     0   1
##   52     0   2
##   53     0   1
##   54     0   3
##   55     0   1
##   55.5   0   0
##   56     0   1
##   57     1   0
##   58     0   3
##   59     0   0
##   60     0   1
##   61     0   0
##   62     0   1
##   63     0   2
##   64     0   0
##   65     0   0
##   66     0   0
##   70     0   0
##   70.5   0   0
##   71     0   0
##   74     0   0
##   80     0   0
##
## , ,  = male
##
##
##          0   1
##   0.42   0   1
##   0.67   0   1
##   0.75   0   0
##   0.83   0   2
##   0.92   0   1
##   1      2   3
##   2      3   1
##   3      0   4
##   4      3   2
##   5      0   0
##   6      0   1
##   7      2   0
##   8      1   1
##   9      2   2
##   10     1   0
##   11     2   1
```

```
##   12     0   1
##   13     0   0
##   14     2   0
##   14.5   0   0
##   15     1   0
##   16    10   1
##   17     6   1
##   18    12   1
##   19    16   2
##   20    10   3
##   20.5   1   0
##   21    16   1
##   22    14   1
##   23     9   1
##   23.5   1   0
##   24    13   1
##   24.5   1   0
##   25    14   4
##   26    10   3
##   27     6   6
##   28    16   2
##   28.5   2   0
##   29    10   3
##   29.7 108  16
##   30    13   1
##   30.5   1   0
##   31     7   3
##   32     8   7
##   32.5   1   0
##   33     9   0
##   34     9   2
##   34.5   1   0
##   35     7   3
##   36    11   4
##   36.5   1   0
##   37     4   1
##   38     5   1
##   39     7   1
##   40     6   1
##   40.5   2   0
##   41     2   0
##   42     7   3
##   43     3   0
##   44     5   1
##   45     4   2
##   45.5   2   0
##   46     3   0
##   47     7   0
##   48     2   3
##   49     2   2
##   50     4   1
##   51     5   1
##   52     3   1
##   53     0   0
##   54     5   0
##   55     1   0
##   55.5   1   0
##   56     2   1
##   57     1   0
##   58     2   0
##   59     2   0
```

```
##   60     2   1
##   61     3   0
##   62     2   1
##   63     0   0
##   64     2   0
##   65     3   0
##   66     1   0
##   70     2   0
##   70.5   1   0
##   71     2   0
##   74     1   0
##   80     0   1
```

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

```
## , ,  = female
##
##
##                 0          1
##   0.42 0.00000000 0.00000000
##   0.67 0.00000000 0.00000000
##   0.75 0.00000000 1.00000000
##   0.83 0.00000000 0.00000000
##   0.92 0.00000000 0.00000000
##   1    0.00000000 0.28571429
##   2    0.40000000 0.20000000
##   3    0.16666667 0.16666667
##   4    0.00000000 0.50000000
##   5    0.00000000 1.00000000
##   6    0.33333333 0.33333333
##   7    0.00000000 0.33333333
##   8    0.25000000 0.25000000
##   9    0.50000000 0.00000000
##   10   0.50000000 0.00000000
##   11   0.25000000 0.00000000
##   12   0.00000000 0.00000000
##   13   0.00000000 1.00000000
##   14   0.16666667 0.50000000
##   14.5 1.00000000 0.00000000
##   15   0.00000000 0.80000000
##   16   0.05882353 0.29411765
##   17   0.07692308 0.38461538
##   18   0.19230769 0.30769231
##   19   0.00000000 0.28000000
##   20   0.13333333 0.00000000
##   20.5 0.00000000 0.00000000
##   21   0.12500000 0.16666667
##   22   0.07407407 0.37037037
##   23   0.06666667 0.26666667
##   23.5 0.00000000 0.00000000
##   24   0.06666667 0.46666667
##   24.5 0.00000000 0.00000000
##   25   0.13043478 0.08695652
##   26   0.11111111 0.16666667
##   27   0.05555556 0.27777778
##   28   0.08000000 0.20000000
##   28.5 0.00000000 0.00000000
```

```
##   29   0.10000000 0.25000000
##   29.7 0.09604520 0.20338983
##   30   0.08000000 0.36000000
##   30.5 0.50000000 0.00000000
##   31   0.11764706 0.29411765
##   32   0.05555556 0.11111111
##   32.5 0.00000000 0.50000000
##   33   0.00000000 0.40000000
##   34   0.00000000 0.26666667
##   34.5 0.00000000 0.00000000
##   35   0.00000000 0.44444444
##   36   0.00000000 0.31818182
##   36.5 0.00000000 0.00000000
##   37   0.16666667 0.00000000
##   38   0.09090909 0.36363636
##   39   0.14285714 0.28571429
##   40   0.07692308 0.38461538
##   40.5 0.00000000 0.00000000
##   41   0.33333333 0.33333333
##   42   0.00000000 0.23076923
##   43   0.20000000 0.20000000
##   44   0.11111111 0.22222222
##   45   0.25000000 0.25000000
##   45.5 0.00000000 0.00000000
##   46   0.00000000 0.00000000
##   47   0.11111111 0.11111111
##   48   0.11111111 0.33333333
##   49   0.00000000 0.33333333
##   50   0.10000000 0.40000000
##   51   0.00000000 0.14285714
##   52   0.00000000 0.33333333
##   53   0.00000000 1.00000000
##   54   0.00000000 0.37500000
##   55   0.00000000 0.50000000
##   55.5 0.00000000 0.00000000
##   56   0.00000000 0.25000000
##   57   0.50000000 0.00000000
##   58   0.00000000 0.60000000
##   59   0.00000000 0.00000000
##   60   0.00000000 0.25000000
##   61   0.00000000 0.00000000
##   62   0.00000000 0.25000000
##   63   0.00000000 1.00000000
##   64   0.00000000 0.00000000
##   65   0.00000000 0.00000000
##   66   0.00000000 0.00000000
##   70   0.00000000 0.00000000
##   70.5 0.00000000 0.00000000
##   71   0.00000000 0.00000000
##   74   0.00000000 0.00000000
##   80   0.00000000 0.00000000
##
## , ,  = male
##
##
##                0          1
##   0.42 0.00000000 1.00000000
##   0.67 0.00000000 1.00000000
##   0.75 0.00000000 0.00000000
##   0.83 0.00000000 1.00000000
##   0.92 0.00000000 1.00000000
```

```
##   1    0.28571429 0.42857143
##   2    0.30000000 0.10000000
##   3    0.00000000 0.66666667
##   4    0.30000000 0.20000000
##   5    0.00000000 0.00000000
##   6    0.00000000 0.33333333
##   7    0.66666667 0.00000000
##   8    0.25000000 0.25000000
##   9    0.25000000 0.25000000
##   10   0.50000000 0.00000000
##   11   0.50000000 0.25000000
##   12   0.00000000 1.00000000
##   13   0.00000000 0.00000000
##   14   0.33333333 0.00000000
##   14.5 0.00000000 0.00000000
##   15   0.20000000 0.00000000
##   16   0.58823529 0.05882353
##   17   0.46153846 0.07692308
##   18   0.46153846 0.03846154
##   19   0.64000000 0.08000000
##   20   0.66666667 0.20000000
##   20.5 1.00000000 0.00000000
##   21   0.66666667 0.04166667
##   22   0.51851852 0.03703704
##   23   0.60000000 0.06666667
##   23.5 1.00000000 0.00000000
##   24   0.43333333 0.03333333
##   24.5 1.00000000 0.00000000
##   25   0.60869565 0.17391304
##   26   0.55555556 0.16666667
##   27   0.33333333 0.33333333
##   28   0.64000000 0.08000000
##   28.5 1.00000000 0.00000000
##   29   0.50000000 0.15000000
##   29.7 0.61016949 0.09039548
##   30   0.52000000 0.04000000
##   30.5 0.50000000 0.00000000
##   31   0.41176471 0.17647059
##   32   0.44444444 0.38888889
##   32.5 0.50000000 0.00000000
##   33   0.60000000 0.00000000
##   34   0.60000000 0.13333333
##   34.5 1.00000000 0.00000000
##   35   0.38888889 0.16666667
##   36   0.50000000 0.18181818
##   36.5 1.00000000 0.00000000
##   37   0.66666667 0.16666667
##   38   0.45454545 0.09090909
##   39   0.50000000 0.07142857
##   40   0.46153846 0.07692308
##   40.5 1.00000000 0.00000000
##   41   0.33333333 0.00000000
##   42   0.53846154 0.23076923
##   43   0.60000000 0.00000000
##   44   0.55555556 0.11111111
##   45   0.33333333 0.16666667
##   45.5 1.00000000 0.00000000
##   46   1.00000000 0.00000000
##   47   0.77777778 0.00000000
##   48   0.22222222 0.33333333
##   49   0.33333333 0.33333333
```

```
##   50   0.40000000 0.10000000
##   51   0.71428571 0.14285714
##   52   0.50000000 0.16666667
##   53   0.00000000 0.00000000
##   54   0.62500000 0.00000000
##   55   0.50000000 0.00000000
##   55.5 1.00000000 0.00000000
##   56   0.50000000 0.25000000
##   57   0.50000000 0.00000000
##   58   0.40000000 0.00000000
##   59   1.00000000 0.00000000
##   60   0.50000000 0.25000000
##   61   1.00000000 0.00000000
##   62   0.50000000 0.25000000
##   63   0.00000000 0.00000000
##   64   1.00000000 0.00000000
##   65   1.00000000 0.00000000
##   66   1.00000000 0.00000000
##   70   1.00000000 0.00000000
##   70.5 1.00000000 0.00000000
##   71   1.00000000 0.00000000
##   74   1.00000000 0.00000000
##   80   0.00000000 1.00000000
```

Podemos ver que la tasa de mujeres supervivientes en senior es del 100%, mientras que en hombres es mucho menor, ya que sólo sobrevivieron 2 hombres mayores a 60 años... Igual que la proporción de mujeres supervivientes respecto a los hombres supervivientes es mayor en Adult y en Child.

Para test:

```
grid.newpage()
plotbyClass_test<-ggplot(test,aes(Pclass,fill=Survived))+geom_bar()
+labs(x="Class", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by
PClass(train)")

plotbyAge_test<-ggplot(test,aes(Age,fill=Survived))+geom_bar()
+labs(x="Age", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by Age
(train)")

plotbySex_test<-ggplot(test,aes(Sex,fill=Survived))+geom_bar()
+labs(x="Sex", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by Sex
(train)")

plotbyEmbarked_test<-ggplot(test,aes(Embarked,fill=Survived))+geom_bar()
+labs(x="Embarked", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by
Embarked (train)")

grid.arrange(plotbyClass_test,plotbyAge_test,plotbySex_test,
plotbyEmbarked_test,ncol=2)
```


```
test$Age[test$Age <=18] = "Child"
test$Age[(test$Age > 18) & (test$Age <=60) & (test$Age != "Child")] =
"Adult"
test$Age[(test$Age != "Child") & (test$Age != "Adult")] = "Senior"
test$Age = as.factor(test$Age)
```

Volvemos a hacer lo mismo para ver los gráficos, con las Clases de edades actualizadas:

```
grid.newpage()
plotbyClass_test<-ggplot(test,aes(Pclass,fill=Survived))+geom_bar()
+labs(x="Class", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by
PClass(train)")

plotbyAge_test<-ggplot(test,aes(Age,fill=Survived))+geom_bar()
+labs(x="Age", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by Age
(train)")

plotbySex_test<-ggplot(test,aes(Sex,fill=Survived))+geom_bar()
+labs(x="Sex", y="Passengers")+ guides(fill=guide_legend(title=""))+
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by Sex
(train)")

plotbyEmbarked_test<-ggplot(test,aes(Embarked,fill=Survived))+geom_bar()
+labs(x="Embarked", y="Passengers")+ guides(fill=guide_legend(title=""))+
```

```
scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived by
Embarked (train)")

grid.arrange(plotbyClass_test,plotbyAge_test,plotbySex_test,
plotbyEmbarked_test,ncol=2)
```

Podemos ver que es similar al dataset de train, pero con la diferencia de que los hombres fueron víctima en su totalidad.

Veremos ahora las probabilidades:

```
table_SexSurvivedTest <- table(test$Sex, test$Survived)
prop.table(table_SexSurvivedTest, margin = 1)
```

```
##
##          0 1
##   female 0 1
##   male   1 0
```

Podemos ver que la totalidad de mujeres sobrevivió en el dataset de test y la totalidad de hombres murió.

```
table_PclassSurvivedTest <- table(test$Pclass,test$Survived)
prop.table(table_PclassSurvivedTest, margin = 1)
```

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

```
##   1 0.5327103 0.4672897
##   2 0.6774194 0.3225806
##   3 0.6697248 0.3302752
```

En test, podemos ver que siempre tenemos mayoría de víctimas y una proporción muy similar entre la 3a y 2a clase. Pero siempre tenemos mayoría de víctimas, exceptuando la primera clase, que la tasa de víctimas es muy poco superior a la de supervivientes.

```
table_AgeSurvivedTest <- table(test$Age,test$Survived)
prop.table(table_AgeSurvivedTest, margin = 1)
```

```
##
##                 0         1
##   Adult  0.6487252 0.3512748
##   Child  0.5555556 0.4444444
##   Senior 0.6363636 0.3636364
```

Podemos ver que en test las proporciones son similares a lo que podemos ver en train

```
tablebyAClassTest <- table(test$Age,test$Survived,test$Pclass)

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

```
## , ,  = 1
##
##
##                   0          1
##   Adult  0.14447592 0.12464589
##   Child  0.05555556 0.03703704
##   Senior 0.27272727 0.36363636
##
## , ,  = 2
##
##
##                   0          1
##   Adult  0.15297450 0.06515581
##   Child  0.11111111 0.12962963
##   Senior 0.27272727 0.00000000
##
## , ,  = 3
##
##
##                   0          1
##   Adult  0.35127479 0.16147309
##   Child  0.38888889 0.27777778
##   Senior 0.09090909 0.00000000
```

```
tableTestbyBClass <- table(test$Age,test$Survived,test$Sex)
tableTestbyBClass
```

```
## , ,  = female
##
##
##            0   1
##   Adult    0 124
##   Child    0  24
##   Senior   0   4
##
## , ,  = male
```

```
##
##
##            0   1
##   Adult  229   0
##   Child   30   0
##   Senior   7   0
```

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

```
## , ,  = female
##
##
##                  0         1
##   Adult  0.0000000 0.3512748
##   Child  0.0000000 0.4444444
##   Senior 0.0000000 0.3636364
##
## , ,  = male
##
##
##                  0         1
##   Adult  0.6487252 0.0000000
##   Child  0.5555556 0.0000000
##   Senior 0.6363636 0.0000000
```

Podemos ver que en test todas las mujeres sobrevivieron, mientras que todos los hombres murieron, sin tener que depender de la edad, la variable que tiene más peso en test para decidir la supervivencia de un pasajero es la del género.

Ahora empezaremos a hacer algunas predicciones:

```
predictTrainSex <- glm(Survived ~ Sex,family = "binomial" , data = train)
summary(predictTrainSex)
```

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

```
predictTestSex <- glm(Survived ~ Sex,family = "binomial" , data = test)
```

```
## Warning: glm.fit: algorithm did not converge
```

```
summary(predictTestSex)
```

```
##
## Call:
## glm(formula = Survived ~ Sex, family = "binomial", data = test)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -2.409e-06  -2.409e-06  -2.409e-06   2.409e-06   2.409e-06
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)    26.57   28885.47   0.001    0.999
## Sexmale       -53.13   36209.86  -0.001    0.999
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5.4798e+02  on 417  degrees of freedom
## Residual deviance: 2.4251e-09  on 416  degrees of freedom
## AIC: 4
##
## Number of Fisher Scoring iterations: 25
```

Comprobamos que la probabilidad de supervivencia disminuye cuando el sexo del pasajero es hombre, en ambos dataframes.

```
predictTrainAge <- glm(Survived ~ Age,family = "binomial" , data = train)
summary(predictTrainAge)
```

```
##
## Call:
## glm(formula = Survived ~ Age, family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1126  -0.9862  -0.9430   1.3616   1.6383
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.14325    0.17209  -0.832   0.4052
## Age         -0.01120    0.00539  -2.077   0.0378 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.7  on 890  degrees of freedom
## Residual deviance: 1182.3  on 889  degrees of freedom
## AIC: 1186.3
##
## Number of Fisher Scoring iterations: 4
```

```
predictTestAge <- glm(Survived ~ Age,family = "binomial" , data = test)
summary(predictTestAge)
```

```
##
## Call:
## glm(formula = Survived ~ Age, family = "binomial", data = test)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0842  -0.9303  -0.9303   1.4465   1.4465
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.61344    0.11150  -5.502 3.76e-08 ***
## AgeChild     0.39030    0.29569   1.320    0.187
## AgeSenior    0.05382    0.63662   0.085    0.933
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 547.98  on 417  degrees of freedom
## Residual deviance: 546.26  on 415  degrees of freedom
## AIC: 552.26
##
## Number of Fisher Scoring iterations: 4
```

Podemos ver que cuando la edad del pasajero es niño la probabilidad de supervivencia aumenta, mientras que disminuye en caso de senior, para el caso de train. En caso de test, podemos ver que la probabilidad de supervivencia disminuye cuando el pasajero es niño o senior.

```
predictTrainPclass <- glm(Survived ~ Pclass,family = "binomial" , data = train)
summary(predictTrainPclass)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass, family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4094  -0.7450  -0.7450   0.9619   1.6836
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.5306     0.1409   3.766 0.000166 ***
## Pclass2      -0.6394     0.2041  -3.133 0.001731 **
## Pclass3      -1.6704     0.1759  -9.496  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.7  on 890  degrees of freedom
## Residual deviance: 1083.1  on 888  degrees of freedom
## AIC: 1089.1
##
## Number of Fisher Scoring iterations: 4
```

```
predictTestPclass <- glm(Survived ~ Pclass,family = "binomial" , data = test)
summary(predictTestPclass)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass, family = "binomial", data = test)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1223  -0.8954  -0.8826   1.2335   1.5043
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.1310     0.1938  -0.676   0.4989
## Pclass2      -0.6109     0.2945  -2.074   0.0381 *
## Pclass3      -0.5759     0.2414  -2.386   0.0171 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 547.98  on 417  degrees of freedom
## Residual deviance: 541.42  on 415  degrees of freedom
## AIC: 547.42
##
## Number of Fisher Scoring iterations: 4
```

Podemos ver que para ambos dataframes, la probabilidad de sobrevivir siendo pasajero de 2a y 3a clase disminuye notablemente.

```
predictTrainTotal <- glm(Survived ~ Sex + Pclass + Age,family =
"binomial", data = train)
summary(predictTrainTotal)
```

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

```
predictTestTotal <- glm(Survived ~ Sex + Pclass + Age,family =
"binomial", data = test)
```

```
## Warning: glm.fit: algorithm did not converge
```

```
summary(predictTestTotal)
```

```
##
## Call:
## glm(formula = Survived ~ Sex + Pclass + Age, family = "binomial",
##     data = test)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -2.409e-06  -2.409e-06  -2.409e-06   2.409e-06   2.409e-06
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.657e+01  4.037e+04   0.001    0.999
## Sexmale     -5.313e+01  3.664e+04  -0.001    0.999
## Pclass2     -1.489e-09  5.117e+04   0.000    1.000
## Pclass3     -9.180e-10  4.335e+04   0.000    1.000
## AgeChild    -4.915e-09  5.274e+04   0.000    1.000
## AgeSenior    3.074e-10  1.103e+05   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5.4798e+02  on 417  degrees of freedom
## Residual deviance: 2.4251e-09  on 412  degrees of freedom
## AIC: 12
##
## Number of Fisher Scoring iterations: 25
```

Podemos ver que para train, el hecho de ser niño aumenta la probabilidad de sobrevivir, mientras que pertenecer a la 2a o 3a clase disminuye las probabilidades de supervivencia, al igual que ser hombre, que también disminuye notablemente. En train podemos ver que sólo aumentaremos las probabilidades de supervivencia si somos ancianos, pero las disminuiremos si pertenecemos a la 2a o 3a clase, al igual que siendo hombres o niños.

Entonces miraremos el modelo C50;

```
nrow(train)
```

```
## [1] 891
```

```
nrow(test)
```

```
## [1] 418
```

Creamos otro dataset para train para ejecutar el modelo C50;

```
mTrain <- select(train, -Embarked, -Fare, -Name)
mTrain$Survived <- ifelse(mTrain$Survived == 0, "Dies", "Lives")
mTrain$Survived <- as.factor(mTrain$Survived)
head(mTrain)
```

```
##   Survived Pclass    Sex  Age
## 1     Dies      3   male 22.0
## 2    Lives      1 female 38.0
## 3    Lives      3 female 26.0
## 4    Lives      1 female 35.0
```

```
## 5     Dies      3   male 35.0
## 6     Dies      3   male 29.7
```

Hacemos lo mismo para test:

```
mTest <- select(test, -Embarked, -Fare, -Name)
mTest$Survived <- ifelse(mTest$Survived == 0, "Dies", "Lives")
mTest$Survived <- as.factor(mTest$Survived)
head(mTest)
```

```
##   Survived Pclass    Sex    Age
## 1     Dies      3   male  Adult
## 2    Lives      3 female  Adult
## 3     Dies      2   male Senior
## 4     Dies      3   male  Adult
## 5    Lives      3 female  Adult
## 6     Dies      3   male  Child
```

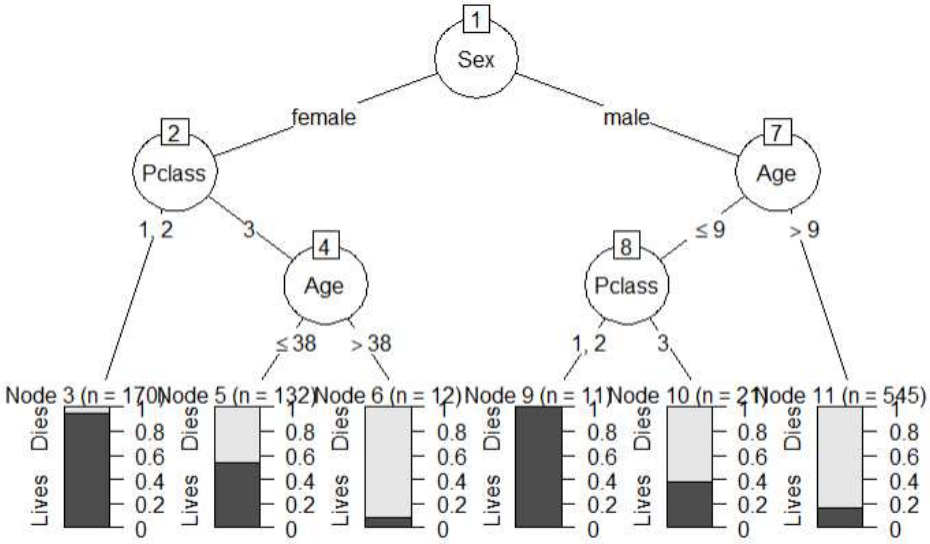
```
yTR <- mTrain[,1]
XTR <- mTrain[,2:4]
yTS <- mTrain[,1]
XTS <- mTrain[,2:4]

trainX <- XTR[1:891,]
trainy <- yTR[1:891]
testX <- XTS[1:418,]
testy <- yTS[1:418]

modelTR <- C50::C5.0(trainX, trainy, rules=TRUE )
summary(modelTR)
```

```
##
## Call:
## C5.0.default(x = trainX, y = trainy, rules = TRUE)
## C5.0 [Release 2.07 GPL Edition]      Mon Jan 06 22:09:31 2020
## -----------------------------
##
## Class specified by attribute `outcome'
##
## Read 891 cases (4 attributes) from undefined.data
##
## Rules:
##
## Rule 1: (51/4, lift 1.5)
##  Pclass = 3
##  Age > 38
##  ->  class Dies  [0.906]
##
## Rule 2: (577/109, lift 1.3)
##  Sex = male
##  ->  class Dies  [0.810]
##
## Rule 3: (11, lift 2.4)
##  Pclass in {1, 2}
##  Sex = male
##  Age <= 9
##  ->  class Lives  [0.923]
##
```

```
## Rule 4: (314/81, lift 1.9)
##  Sex = female
##  ->  class Lives  [0.741]
##
## Default class: Dies
## Evaluation on training data (891 cases):
##
##          Rules
##    ----------------
##      No      Errors
##
##       4  169(19.0%)   <<
##
##
##     (a)   (b)    <-classified as
##    ----  ----
##     479    70    (a): class Dies
##      99   243    (b): class Lives
##
##
##  Attribute usage:
##
##  100.00% Sex
##    6.96% Pclass
##    6.96% Age
##
##
## Time: 0.0 secs
```

Podemos ver ahora que si la es hombre tiene un 81% de probabilidades de morir mientras que una mujer tiene un porcentaje de supervivencia de 0,74%, no nos aparece ninguna Rule que nos lo relacione con Age o Pclass, la clase que tiene más peso para decidir el destino de la vida de un pasajero en el Titanic es la variable Sex;

```
model1 <- C50::C5.0(trainX, trainy)
plot(model1)
```



Podemos ver en el Árbol de decisión cómo se distribuyen según género, que es la variable que tiene más peso en relación con la supervivencia, seguido de la clase y la edad para las mujeres y la edad y la clase para los hombres.

Finalmente comprobamos su cualidad prediciendo la clase por los datos de prueba con los datos de test.

```
predicted_model <- predict(model1, testX, type="class")
print(sprintf("La precisión del Árbol es: %.4f
%%",100*sum(predicted_model == testy) / length(predicted_model)))

## [1] "La precisión del Árbol es: 83.0144 %"
```

Podemos ver que la precisión del Árbol es del 83%.

Para finalizar, crearemos los nuevos archivos csv de para train y test, que estos estarán sin valores nulos y sin las variables que no consideremos relevantes para el estudio.

```
write.csv(train, file = "D:/Documentos/UOC/Master/3-
TCVD/Practica2/trainFinal.csv")
write.csv(test, file = "D:/Documentos/UOC/Master/3-
TCVD/Practica2/testFinal.csv")
```

6.- Resolución del problema. A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

A partir de los resultados obtenidos podemos indicar que, de todos los modelos analizados, el modelo utilizado de árbol de decisión permite acercarse a un porcentaje de resolución del 83% en el acierto de los resultados de supervivencia de los pasajeros del Titanic. Según todas las pruebas estadísticas realizadas, el modelo de árbol de decisión se acerca a un porcentaje de acierto aceptable.

Los resultados permiten responder al planteamiento inicial del problema, ya que con los datos aportados se pueden extraer conclusiones sobre la posibilidad de supervivencia de los pasajeros en función de las variables analizadas.

Pero también debemos tener en cuenta el punto de vista sociocultural del momento del accidente del Titanic principios del siglo XX, donde las diferencias entre clases sociales se reflejaban en todos los ámbitos de la sociedad, hemos podido observar que la variable *Pclass,* junto a la variable *Sex* y la variable *Age* fueron determinantes para sobrevivir al accidente, la primera debido a que la a mayor disponibilidad económica, primera clase, mayor probabilidad de sobrevivir esto era debido a que primera clase ocupaba las plantas más cercanas a la cubierta y en consecuencia el recorrido hasta las barcas salvavidas era menor y que no disponían de barcas salvavidas para todos los pasajeros, posteriormente y siguiendo este mismo criterio, se encontraba la segunda clase y por último nos encontramos con la tercera clase. Pero también hemos observado que el sexo de la

persona también era determinante junto con la edad, esto es debido a que se consideraba a las mujeres y a los niños más débiles y tenían prioridad a la hora de ser evacuados.

En definitiva, la variable más determinante es Age tal y como ya hemos mencionado durante el análisis de los resultados obtenidos.

7.- Código, ficheros y contribuciones

El código se encuentra en el siguiente link:

https://github.com/gonmard/UOC_PRAT2_TITANIC/blob/master/code/PRA2TIPOLOGIA_DG_JC_V3.Rmd

Los ficheros originales y los definitivos para realizar todas las pruebas se encuentran en el repositorio generado en la presentación de esta segunda práctica de la asignatura, ver el siguiente link:

https://github.com/gonmard/UOC_PRAT2_TITANIC/tree/master/files

Contribuciones:

Contribuciones	Firmas
Investigación previa	Jordi Costilla / Diego González
Redacción de las respuestas	Jordi Costilla / Diego González
Desarrollo del código	Jordi Costilla / Diego González