PRAC2: Tipologia i cicle de vida de les dades

Mila Ramírez Guevara

13/12/2020

1. Descripción de los datos.

Para este proyecto contamos con los datos de los pasajeros del Titanic. El estudio que queremos realizar es un pequeño proyecto de Machine Learning, para intentar predecir la supervivencia de los pasajeros del titanic en base a determinadas características conocidas de los mismos que tenemos registradas en el dataset.

Además de esta predicción general, como objetivo adicional, me gustaría conocer si el pasajero número 892 sobrevivió al Titanic y que predicción hace sobre esto el modelo que plantearé en las siguientes páginas.

Sobre la importancia de un proyecto de este tipo, esta en que no solo abarca la predicción de la supervivencia a un accidente. Además extrapolando este tipo de estudios a otras situaciones, un modelo similar nos podría permitir determinar si en función de ciertas características un usuario de una plataforma compraría determinado producto. O un paciente con determinadas patologías o características respondería bien a x tratamiento. Por lo que el desarrollar un proyecto de este tipo puede tener aplicaciones en campos diversos.

Volviendo al proyecto del títanic y a la descripción de los datos es importante tener en cuenta las notas adicionales que se hacen en la fuente de datos y el significado de las columnas que recogen las características de cada pasajero.

Variable	Definition	Key
survival	Survival	0 = No, 1 = Yes
pclass	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
sex	Sex	
Age	Age in years	
sibsp	# of siblings / spouses aboard the Titanic	
parch	# of parents / children aboard the Titanic	
ticket	Ticket number	
fare	Passenger fare	
cabin	Cabin number	
embarked	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

- pclass: A proxy for socio-economic status (SES) 1st = Upper 2nd = Middle 3rd = Lower
- age: Age is fractional if less than 1. If the age is estimated, is it in the form of xx.5
- sibsp: The dataset defines family relations in this way... Sibling = brother, sister, stepbrother, stepsister Spouse = husband, wife (mistresses and fiancés were ignored)
- parch: The dataset defines family relations in this way... Parent = mother, father Child = daughter, son, stepdaughter, stepson Some children travelled only with a nanny, therefore parch=0 for them.

Así pues, contamos con 891 registros de 12 variables para el set de entrenamiento y 418 registros de 11 variables para el set de test. El total de registros será de 1309 y más adelante los veremos en detalle.

Siendo el pasajero 892, el primero del que desconocemos la supervivencia y el que voy a intentar descubrir a lo largo de estas páginas.

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

Lo primero para comenzar con este apartado sería revisar los datos de los que disponemos.

En mi caso tengo tres tipos de datos Train,test,gender_submision. En la descripción de los archivos que se puede encontrar en Kaggle (https://www.kaggle.com/c/titanic/data) se indica que los datos se han dividido en datos de entrenamiento y de test para el modelo de machine learning que queremos crear

- Train: csv con información sobre los pasajeros del Titanic para entrenar el modelo. Para este set de datos se proporciona el resultado final para cada pasajero indicando si este sobrevive o no. Y se espera que el modelo esté basado en las características de cada pasajero.
- Test: csv con información de los pasajeros a modo de test para probar el modelo. Este set de datos se debe usar para ver que tan bien funciona nuestro modelo con datos a ciegas de los que no conocemos el resultado en cuanto a si un pasajero sobrevive o no.
- gender_submission (es el resultado del modelo en caso de que se plantee que solo las mujeres sobreviven al accidente. y se proporciona como modelo del resultado que debemos obtener después de usar el modelo)

En base a la descripción de estos archivos es claro que los que debo usar para trabajar son el train y el test.

Para crear el modelo debo usar los datos del archivo train.csv, pero para testear el modelo tendré que usar los datos del archivo test.csv. Así que el primer paso será hacer una lectura de ambos archivos y verificar los datos con los que cuento.

```
#librerias necesarias

library(dplyr)
```

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

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

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

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v stringr 1.4.0
## v tidyr   1.1.2     v forcats 0.5.0
## v readr   1.4.0
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(hrbrthemes)
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.

##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and

##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
#Cargar datos y primera revisión

Titanic_train<- read.csv(
  paste("C:/Users/mila_/Documents/Master ciencia de dades",
  "/Tipología y ciclo de vida de los datos/PRAC 2/train.csv",sep=""),
  header=TRUE)

Titanic_test<- read.csv(
  paste("C:/Users/mila_/Documents/Master ciencia de dades",
  "/Tipología y ciclo de vida de los datos/PRAC 2/test.csv",sep = ""),
  header=TRUE)


str(Titanic_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       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
##  $ Sex        : chr  "male" "female" "female" "female" ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  "" "C85" "" "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
```

```
str(Titanic_test)
```

```
## 'data.frame':    418 obs. of  11 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
```

```
##  $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
##  $ Name       : chr  "Kelly, Mr. James" "Wilkes, Mrs. James (Ellen Needs)" "Myles, Mr. Thomas Francis
##  $ Sex        : chr  "male" "female" "male" "male" ...
##  $ 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     : chr  "330911" "363272" "240276" "315154" ...
##  $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
##  $ Cabin      : chr  "" "" "" "" ...
##  $ Embarked   : chr  "Q" "S" "Q" "S" ...
```

```
# En este apartado además quiero conocer la identidad del pasajero 892
Titanic_test[Titanic_test$PassengerId==892,]
```

```
##   PassengerId Pclass             Name  Sex  Age SibSp Parch Ticket   Fare Cabin
## 1         892      3 Kelly, Mr. James male 34.5     0     0 330911 7.8292
##   Embarked
## 1        Q
```

De esta primera revisión podemos ver que los datos con los que contamos en ambos archivos son similares para verificar que en efecto podemos usar train.csv como set de entrenamiento compararé las columnas con las que contamos en ambos dataframe, lo esperable es que train.csv tenga un campo referido a la supervivencia y test.csv no cuente con el. ya lo hemos visto con la función str, pero dado que para combinar ambos archivos debemos verificar que los nombres de las variables son identicos he considerado oportuno revisar solo el nombre de las columnas, para ver si hay diferencias en la escritura(Mayúsculas, minúsculas,errores tipograficos...) Además en esta primera exploración hemos descubierto la identidad de nuestro pasajero 892 Mr James Kelly pasajero de tercera clase, de 34 años que viajaba solo pago 7,8 libras por el billete y embarcó en el puerto de Queensland.

```
#Revisión de columnas en dataframes
colnames(Titanic_train)
```

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

```
colnames(Titanic_test)
```

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

De la revisión de columnas en efecto vemos que el archivo train.csv cuenta con una columna referida a la supervivencia.

En este caso la integración de datos no sería obligatoria ya que para hacer el modelo puedo trabajar con los datos que me proporciona el archivo train csv, pero deberé hacer la limpieza de ambos datasets y dado que cuentan con prácticamente las mismas variables he considerado mejor hacer la limpieza en conjunto de todos los datos.

Por lo tanto tendré que integrar los datos de train con test, y combinar ambos dataframes para más adelante volver a separarlos.

La variable survived, que solo está presente en el grupo de entrenamiento se debe añadir también al grupo test, por lo que creo una variable en el dataframe test que se llame "Survived" y tenga valores NA. Quiero además comprobar antes de combinar los dataframes que en el dataframe "train" no hay valores NA para Survived, y en efecto es así.

Además para más adelante separar los datos tendré la opción de separar por el PassangerID, teniendo en cuenta de que se trata de números consecutivos y podría hacer el corte en la fila 891 para el set de entrenamiento, u otra opción sería usar el campo "Survived", pero la finalidad última del proyecto es que estos valores dejen de ser NA, y se puedan predecir, por lo que no sería un buen separador, así que he decidido añadir una nueva columna en ambos dataset para identificar si son registros de entrenamiento o test.

```
paste("Los valores NA para variable Survived rn train son:",sum(is.na(Titanic_train$Survived)))
```

```
## [1] "Los valores NA para variable Survived rn train son: 0"
```

```
#nueva columna en test dataframe

Titanic_test$Survived <- NA

#nueva columna Set_type para separar dataframes más adelante
Titanic_test$Set_type<- "test"
Titanic_train$Set_type<- "train"

#combinación de datasets

Titanic_complete<- rbind(Titanic_train, Titanic_test)

#compruebo las filas del nuevo dataframe para ver si es correcto.
paste("Número de filas de Dataframe Titanic_complete:",nrow(Titanic_complete))
```

```
## [1] "Número de filas de Dataframe Titanic_complete: 1309"
```

3. Limpieza de datos.

Antes de empezar la limpieza de datos propiamente dicha, quiero revisar la cantidad de datos de que dispongo, de manera más formal ya que esta información si está facilmente accesible en el apartado "global enviroment" de RStudio, pero para la presentación del trabajo considero que es importante tenerla visible. Así que haré un estudio muy preeliminar para determinar las dimensiones del dataframe con el que estoy trabajando y algunas características de las variables.

```
#preliminar analysis of dataframe variables and dimensions

str(Titanic_complete)
```

```
## 'data.frame':    1309 obs. of  13 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
##  $ Sex        : chr  "male" "female" "female" "female" ...
```

```
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  "" "C85" "" "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
##  $ Set_type   : chr  "train" "train" "train" "train" ...
```

Aquí comprobamos que el tipo de objeto es en efecto un Dataframe que contiene 1309 observaciones y 13 variables, los nombres de cada variable y la interpretación que hace R de cada una de ellas.

En la interpretación que hace R por defecto, se considera:

- variables numéricas discretas (int) : PassengerId, Survived, Pclass, Age, SibSp, Parch
- variables numéricas continuas (num):Fare
- variables tipo texto: Name, Sex, Ticket, Cabin, Embarked, set_type

De la interpretación de datos que ha hecho R podemos detectar que hay algunas diferencias con lo que podríamos interpretar nosotros. Ya que las variables numéricas que interpreto serían (Age, SibSp,Parch y Fare) Mientras que las variables Survived, Pclass,Sex,Cabin, Embarked y set_type deberían considerarse variables categóricas.

PassagerID, que sirve para identificar a los pasajeros la mantendré como variable numérica, y la variable Name tal y como está planteada tampoco es representativa pero podemos hacer alguna transformación con ella y plantearla como un factor.

Interpretado esto lo que haré será cambiar la interpretación de variables que ha hecho R y establecer la mía.

```
#cambio de variables texto a variables factor.

for (i in seq_along(Titanic_complete[,c(2,3,5,9,11,12,13)])) {
  Titanic_complete[,c(2,3,5,9,11,12,13)][[i]] <- as.factor(Titanic_complete[,c(2,3,5,9,11,12,13)][[i]])
}
#comprobació de que el cambio es efectivo
unlist(lapply(Titanic_complete[,c(2,3,5,9,11,12,13)], class), use.names = TRUE)
```

```
## Survived   Pclass      Sex   Ticket    Cabin Embarked Set_type
## "factor" "factor" "factor" "factor" "factor" "factor" "factor"
```

3.1 Gestión de Zeros y elementos vacíos.

Para iniciar esta sección, utilizo la función summary para tener un resumen y visión general de las variables categóricas, numéricas y también para poder detectar el número de missing values.

```
#summary of variables
summary(Titanic_complete)
```

```
##   PassengerId   Survived   Pclass      Name              Sex
##  Min.   :   1   0   :549   1:323   Length:1309        female:466
##  1st Qu.: 328   1   :342   2:277   Class :character   male  :843
##  Median : 655   NA's:418   3:709   Mode  :character
##  Mean   : 655
```

```
##  3rd Qu.: 982
##  Max.   :1309
##
##       Age            SibSp            Parch            Ticket
##  Min.   : 0.17   Min.   :0.0000   Min.   :0.000   CA. 2343:  11
##  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  Set_type
##  Min.   :  0.000                 :1014    :  2    test :418
##  1st Qu.:  7.896   C23 C25 C27   :   6   C:270    train:891
##  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
```

```
#hecha la comprobación la única variable en la que es extraño encontrar valores 0 es en Fare.
#ya que esto implica una tarifa gratuita, lo cual es extraño
#contabilizo el número de 0 para esta variable con el siguiente comando.

paste("número de registros con valor zero en la tarifa:",length(Titanic_complete[Titanic_complete$Fare==
```

```
## [1] "número de registros con valor zero en la tarifa: 13"
```

Con esta primera conversión podemos detectar que hay valores missing en la variable Age se reflejan como NA al igual que en Fare, por otro lado en Cabin y embarked se reflejan como una variable vacía (null). Además hay valores zero en SibSp, Parch y Fare.

No es extraño que haya valores zero en SibSp, Parch pero sí en Fare como se ha comentado.

La mayor cantidad devalores missing esta en Cabin y en Age por lo que habrá que tratarlos, Los otros valores missing corresponden a Embarked y Fare, aunque estos son solo 3 registros por lo que se imputen o se eliminen no deberían tener una gran repercusión.

De la función Summary también deduzco que las variables categoricas están normalizadas. Es decir, no hay variables que estén por ejemplo escritas en diferentes formas(Mayúsculas, minúsculas) y que representen la misma categoría sino que las nomenclaturas son homogéneas. Donde podría sospechar que podría existir este problema sería en las variables Ticket y Cabin ya que como vemos contienen varios valores que se clasifican como "otros". Para poder acabar de comprobar esto he usado la función Table para hacer un conteo de los registros de cada variable

Pero de esta comprobación extraigo muy poca información ya que como es esperable hay una gran cantidad de tickets y de cabinas. Si quiero usar estas variables para el modelo tendré que tratarlas de alguna manera.

```
#comprobación de variables categóricas Cabin y TIcket,el primer dataframe corresponde
#a Ticket y el segundo a Cabin

id.ticket_Cabin<-c(9,11)

var_ticket_Cabin<-colnames(Titanic_complete)[id.ticket_Cabin]
```

```
for (i in var_ticket_Cabin){

  print(tail(as.data.frame(table(Titanic_complete[i]))))

}
```

```
##            Var1 Freq
## 924  W./C. 6607    4
## 925  W./C. 6608    5
## 926  W./C. 6609    1
## 927 W.E.P. 5734    2
## 928   W/C 14208    1
## 929   WE/P 5735    2
##     Var1 Freq
## 182   F2    4
## 183  F33    4
## 184  F38    1
## 185   F4    4
## 186   G6    5
## 187    T    1
```

Por el momento he decidido considerar que las cabinas y tickets estan normalizados.Aunque muy posiblemente estas variables no lleguen a la fase de análisis El siguiente paso es tratar los valores perdidos

missing values en variable Embarked

Antes de optar por un método de imputación de variable he decidido revisar los pasajeros concretos que tienen estos valores perdidos para verificar si hay algo que nos pudiera dar una pista clara del puerto de Embarque, sabiendo que la mayoría de pasajeros embarcaron en el puerto S(South Hampton)una opción sería también asumir que para estos pasajeros el emarque fue en el puerto S.

```
#Búsqueda de los pasajeros con valores missing

Titanic_complete[Titanic_complete$Embarked=="",]
```

```
##     PassengerId Survived Pclass                                      Name
## 62           62        1      1                    Icard, Miss. Amelie
## 830         830        1      1 Stone, Mrs. George Nelson (Martha Evelyn)
##        Sex Age SibSp Parch Ticket Fare Cabin Embarked Set_type
## 62  female  38     0     0 113572   80   B28             train
## 830 female  62     0     0 113572   80   B28             train
```

las pasajeras coinciden en número ticket, tarifa, clase y cabina, es posible que embarcaran en el mismo puerto. No consta que viajaran con esposos,hermanos o hijos, por lo que descarto poder obtener el puerto de embarque a partir de datos de posibles familiares que viajaran con ellas.

otra opción es encontrar pasajeros que tengan la misma cabina o el mismo ticket es posible que si eso se diera embarcaran en el mismo puerto que estas dos pasajeras, aplico el filtro correspondiente para detectar si hay otros pasajeros en la misma cabina o que tengan el mismo ticket

```
#Busqueda de otros pasajeros con misma cabina o número de ticket.
Titanic.embarked<-Titanic_complete[Titanic_complete$PassengerId!=62 &
                                   Titanic_complete$PassengerId!=830,]
Titanic.embarked[Titanic.embarked$Cabin=="B28",]
```

```
##  [1] PassengerId Survived    Pclass      Name        Sex         Age
##  [7] SibSp       Parch       Ticket      Fare        Cabin       Embarked
## [13] Set_type
## <0 rows> (or 0-length row.names)
```

```
Titanic.embarked[Titanic.embarked$Ticket==113572,]
```

```
##  [1] PassengerId Survived    Pclass      Name        Sex         Age
##  [7] SibSp       Parch       Ticket      Fare        Cabin       Embarked
## [13] Set_type
## <0 rows> (or 0-length row.names)
```

No tengo ningún resultado, y en este punto considero que al tratarse de únicamente 2 registros podríamos eliminarlos, en una situación de real seguramente los eliminaría, pero al tratarse de un proyecto para estudio y para una competición de Kaggle decido optar por buscar algún método de imputación.

Una opción es encontrar los puertos de embarque considerando las clases y las tarifas.

Utilizo la función table para verificar en que puertos han subido mayoritariamente los pasajeros de primera clase, ya que si hay una clara mayoría en este punto se resolvería el problema.

```
#Puertos de embarque segun clase

table(Titanic.embarked$Pclass,Titanic.embarked$Embarked)
```

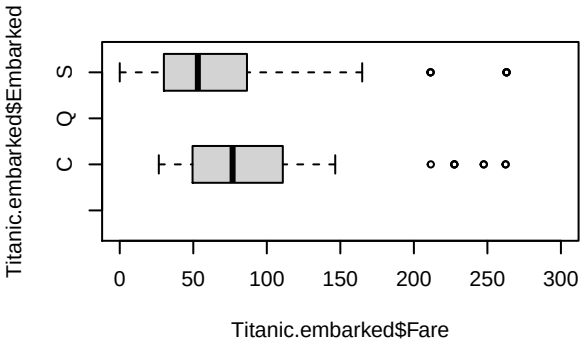
```
##
##           C   Q   S
##   1   0 141   3 177
##   2   0  28   7 242
##   3   0 101 113 495
```

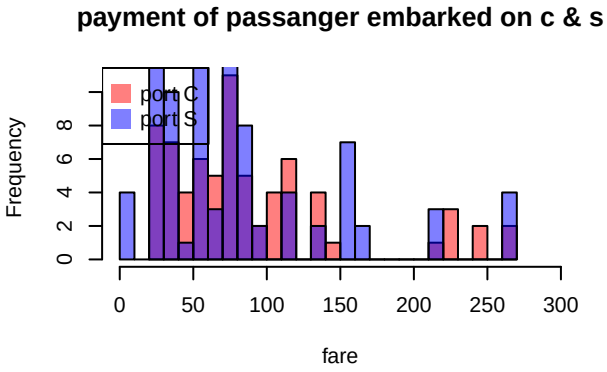
De esto puedo deducir que es más probable que las pasajeras embarcaran o bien en South Hampton o bien en Cherburgo. Considerando además de la clase las tarifas de los tickets puedo plantear dos graficos, un boxplot y un histograma

```
#creo un dataframe accesorio que no incluya a los pasajeros con variabales missing
Titanic.embarked<-filter(Titanic.embarked[Titanic.embarked$Pclass==1
                                          & Titanic.embarked$Embarked!="Q"
                                          & Titanic.embarked$Fare<300,])
Titanic.embarked<-Titanic.embarked %>%
  filter_all(~ !is.na(.))


nf <- layout(matrix(c(1,2,1,2,1,2),ncol=4), widths=c(10,10), heights=c(15,15), TRUE)
```

```
## Warning in matrix(c(1, 2, 1, 2, 1, 2), ncol = 4): la longitud de los datos [6]
## no es un submúltiplo o múltiplo del número de columnas [4] en la matriz
```

```r
boxplot(Titanic.embarked$Fare ~ Titanic.embarked$Embarked, horizontal = TRUE,
        ylim = c(0,300))

#representación de tarifas en relación al puerto de embarque
hist(Titanic.embarked[Titanic.embarked$Embarked=="C","Fare"],
     breaks=30,xlim=c(0,300), col=rgb(1,0,0,0.5),
     xlab="fare", main="payment of passanger embarked on c & s")
hist(Titanic.embarked[Titanic.embarked$Embarked=="S","Fare"],
     breaks=30,xlim=c(0,300), col=rgb(0,0,1,0.5), add=T)
legend("topleft", legend=c("port C","port S"),
       col=c(rgb(1,0,0,0.5),
     rgb(0,0,1,0.5)), pt.cex=2, pch=15 )
```





En ambos llego a la misma conclusión. Lo más probable es que las pasajeras embarcaran en el puerto S, ya que apróximadament el 70% de los pasajeros de primera clase que pagaron por sus tickets 80 libras o menos embarcaron por la puerta S mientras que en el caso de la puerta C solo lo hicieron un 50%.En el histograma veo información similar, hay más probabilidad de que los pasajeros embarcaran por la puerta S que por la C.

En este caso voy a optar por considerar que las pasajeras embarcaron en el puerto de South Hampton que además es donde más pasajeros embarcaron, aún así todo apunta a que son missing values completely at Random Para comprobar que el cambio se ha realizado correctamente hago el recuento otra vez con la función "table"

```r
#Recuento en dataframe original Titanic_complete
Titanic_complete[Titanic_complete$Embarked=="","Embarked"]<-"S"
```

```
table(Titanic_complete$Embarked)
```

```
##
##       C   Q   S
##   0 270 123 916
```

missing values en variable Fare

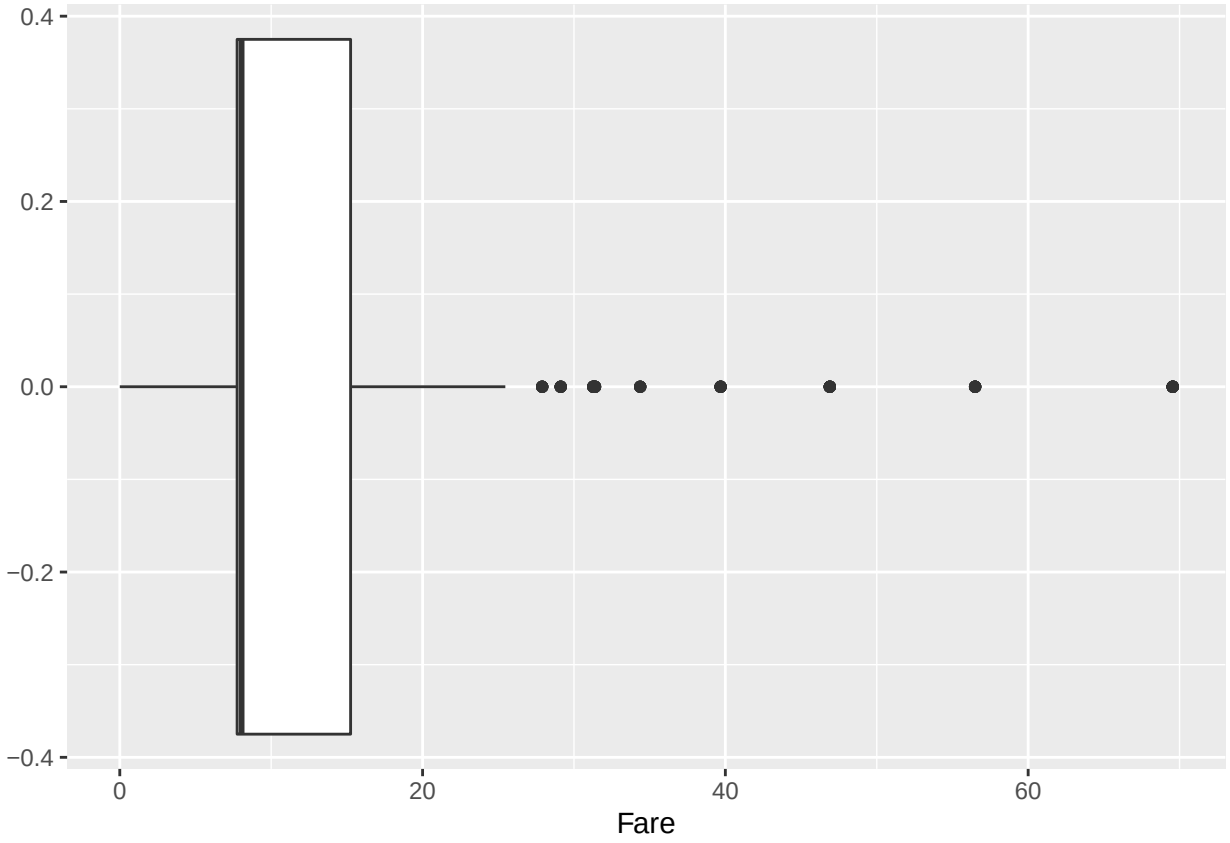
A priori este valor faltante se podría considerar de tipo MAR(Missing at random), es decir a priori se podría explicar esta variable a partir de la classe y tal vez también por el puerto de embarque pero solo con la clase deberíamos poder tener una aproximación al dato.

```
#compruebo el registro completo que corresponde al dato perdido

Titanic_complete[is.na(Titanic_complete$Fare),]
```

```
##      PassengerId Survived Pclass               Name  Sex  Age SibSp Parch
## 1044        1044     <NA>      3 Storey, Mr. Thomas male 60.5     0     0
##      Ticket Fare Cabin Embarked Set_type
## 1044   3701   NA              S     test
```

```
#compruebo en que valores de fare se concentran las tarifas para la tercera clase
Titanic.class<-filter(Titanic_complete[Titanic_complete$Pclass==3,])
ggplot(Titanic.class,aes(x=Fare))+geom_boxplot()
```

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




```
summary(Titanic.class$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    7.75    8.05   13.30   15.25   69.55       1
```

No disponemos de mucha información en el registro del pasajero pero si hacemos un grafico de caja para la tarifa veremos que hay varios valores un tanto extraños que se alejan de la mayoría, este tipo de valores suele afectar a la media, por lo que en este caso para la imputación, será mejor reemplazar el valor de la variable por la mediana que suele verse menos afectada por valores extremos, así pues reemplazo el valor. Aunque es cierto que hay más valores extremos de los que esperaría para esta variable y tendré que hacer más comprobaciones al respecto.

```
#calculo de la mediana y reemplazo del valor missing
median.fare<-median(Titanic_complete$Fare, na.rm = TRUE)
Titanic_complete[is.na(Titanic_complete$Fare),"Fare"]<-median.fare
#comprobación de que ya no hay valores missing para Fare
sum(is.na(Titanic_complete$Fare))
```

```
## [1] 0
```

missing values variable Age

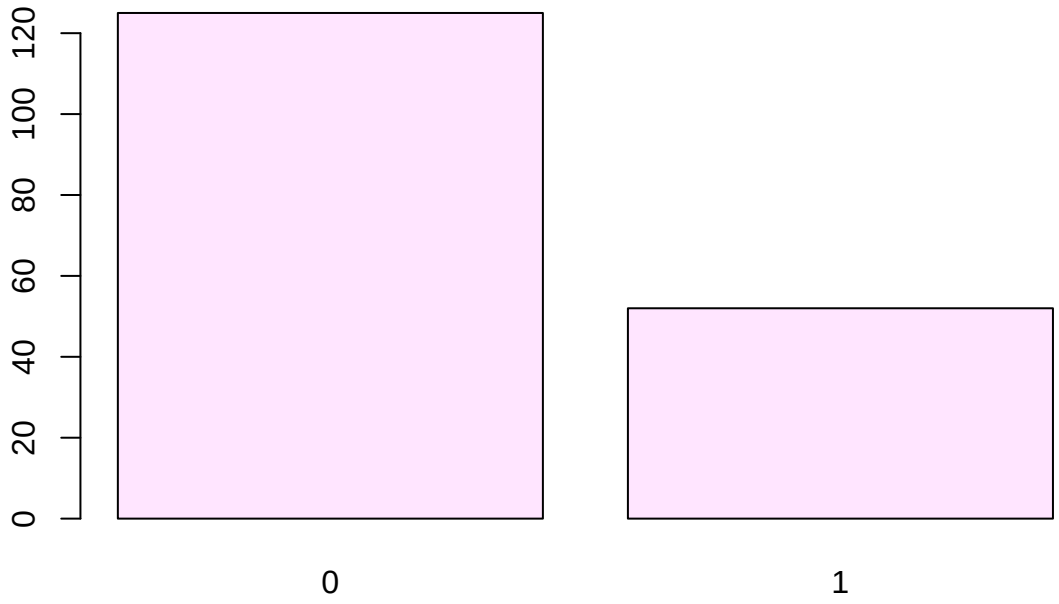
La imputación que haga de esta variable tendrá una repercusión mayor en el modelo por lo tanto quiero primero hacer una evaluación de los valores que si tenemos para esta variable y también de los registros que no tienen esta variable missing para ver cual sería la mejor manera de imputar estos valores.

En este caso interpreto que las variables missing son también del tipo MAR(Missing at random), es decir que alguna otra variable tal vez pueda explicar la ausencia de estos valores

Por otra parte, una posible causa que se me ocurre es que tal vez los pasajeros de los que no tenemos datos sobre la edad no sobrevivieran al accidente.

Hago un gráfico para comprobarlo.

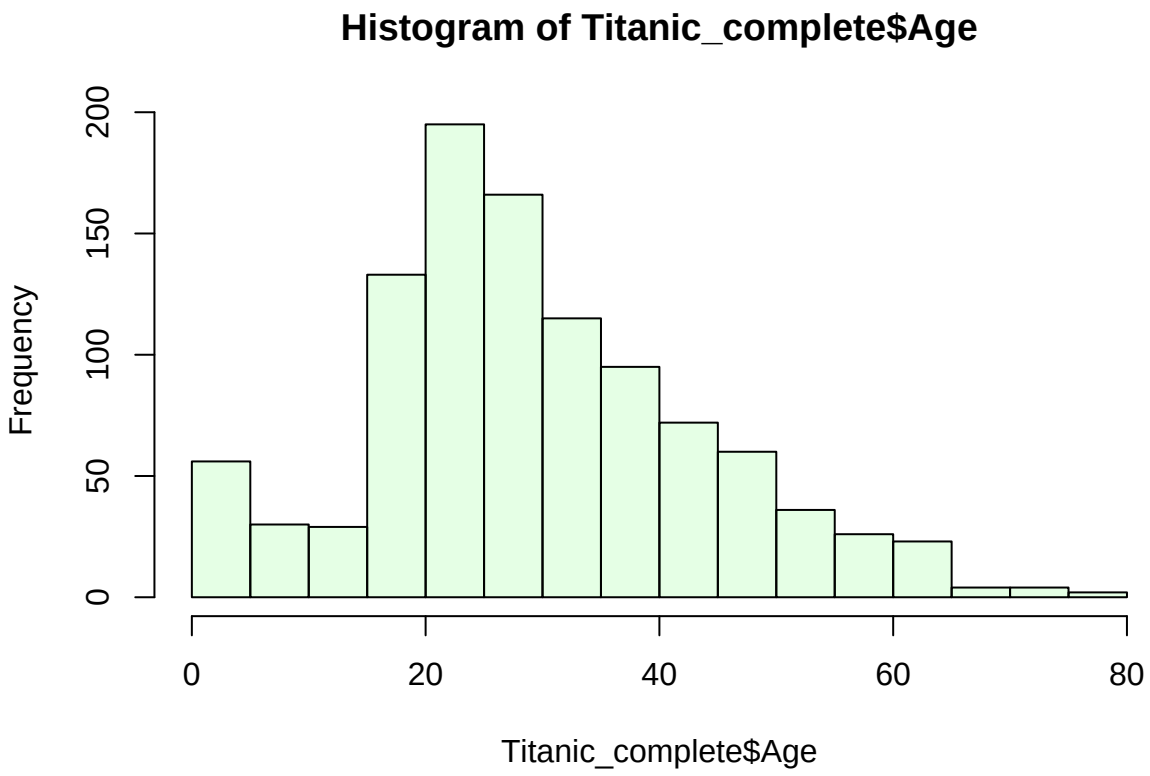
```
#identificadores de los registros con missing data para la variable Age
id.mis.age <- which( is.na(Titanic_complete$Age))
plot(Titanic_complete[id.mis.age,"Survived"],col = rgb(1, 0, 1, 0.10))
```



Del gráfico puedo ver que en efecto la mayoría de los datos perdidos sobre la edad se corresponden a pasajeros que no sobrevivieron al accidente.

Hago un histograma y reviso otra vez el resumen de los datos

```
#Distribución de las edades y resumen de los datos.

hist(Titanic_complete$Age,col = rgb(0, 1, 0, 0.10))
```




```
summary(Titanic_complete$Age,na.rm=TRUE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.17   21.00   28.00   29.88   39.00   80.00     263
```

Del gráfico también deduzco que la distribución de las edades en un histograma se aproxima a una normal (esto se verá en detalle en puntos posteriores)y con el resumen de datos obtengo que la mediana y la media son bastante similares la media es de 29,88 y la mediana es de 28,00. Normalmente si hay muchos valores extremos o outliers reemplazaría los datos por la mediana pero realmente no tenemos muchos valores extremos y todos entran dentro de lo que podemos considerar valores normales para la edad siendo la persona mayor de unos 80 años y las menores bebes de meses.

En este caso podría imputar los valores directamente con la media, pero los valores missing son de hasta el 20% por lo que realmente son muchos valores como para imputarlos todos con el mismo valor, una alternativa es generar valores random que se encuentren dentro del rango interquartilico, que constituye el 50% de los registros.

```
#Calculo de valores random que se encuentren entr Q1 Y Q3.

Q1<-quantile(Titanic_complete$Age,na.rm = TRUE)[[2]]
Q3<-quantile(Titanic_complete$Age,na.rm = TRUE)[[4]]
n.row<-nrow(Titanic_complete[id.mis.age,])
valores<- sample(Q1:Q3,n.row,replace = TRUE)

#reemplazo los valores missing por el conjunto de valores random generados
```

```
Titanic_complete[id.mis.age,"Age"]<-valores

#compruebo que los  valores se han substituido correctamentamente contando los valores NA, para la vari
sum(is.na(Titanic_complete$Age))
```

```
## [1] 0
```

```
summary(Titanic_complete$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.17   22.00   29.00   29.84   36.50   80.00
```

Hacer esta imputación provoca que el rango intercuartílico se mantenga similar al inicial y también que la media se altere menos. Aunque no es un método e imputación ideal. Tal vez cabría considerar otro tipo de imputación como KNN de cara a mejoras en el modelo.

missing values variable cabin

En el caso de los valores missing para la variable Cabin representan hasta el 77% de los valores que tenemos para esta variable, por lo tanto más de la mitad de la muestra, a priori no creo que estos valores puedan ser imputables no obstante, si separamos la letra de los números que forman cada cabina podemos tener categorías más claras, y ver si hay algún tratamiento posible.

```
#Creo una nueva columna resumen de las cabinas solo con la letra.
#Buscando información sobre las cabinas la letra indica la cubierta en la que se encontraban.
#y nos puede servir para agrupar las cabinas.

Titanic_complete$CabinG <- substring(Titanic_complete$Cabin, 1, 1)
#compruebo que la nueva variable se ha creado correctamente con la función head
head(Titanic_complete)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                 Name    Sex Age SibSp Parch
## 1                            Braund, Mr. Owen Harris   male  22     1     0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
## 3                             Heikkinen, Miss. Laina female  26     0     0
## 4       Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
## 5                           Allen, Mr. William Henry   male  35     0     0
## 6                                   Moran, Mr. James   male  27     0     0
##            Ticket    Fare Cabin Embarked Set_type CabinG
## 1        A/5 21171  7.2500             S    train
## 2         PC 17599 71.2833   C85        C    train      C
## 3 STON/O2. 3101282  7.9250             S    train
## 4           113803 53.1000  C123        S    train      C
## 5           373450  8.0500             S    train
## 6           330877  8.4583             Q    train
```

```r
#con las funciones table y addmargins compruebo la cantidad de registros
#que tengo para cada variable.

addmargins(addmargins(table(Titanic_complete$Pclass,Titanic_complete$CabinG),2),1)
```

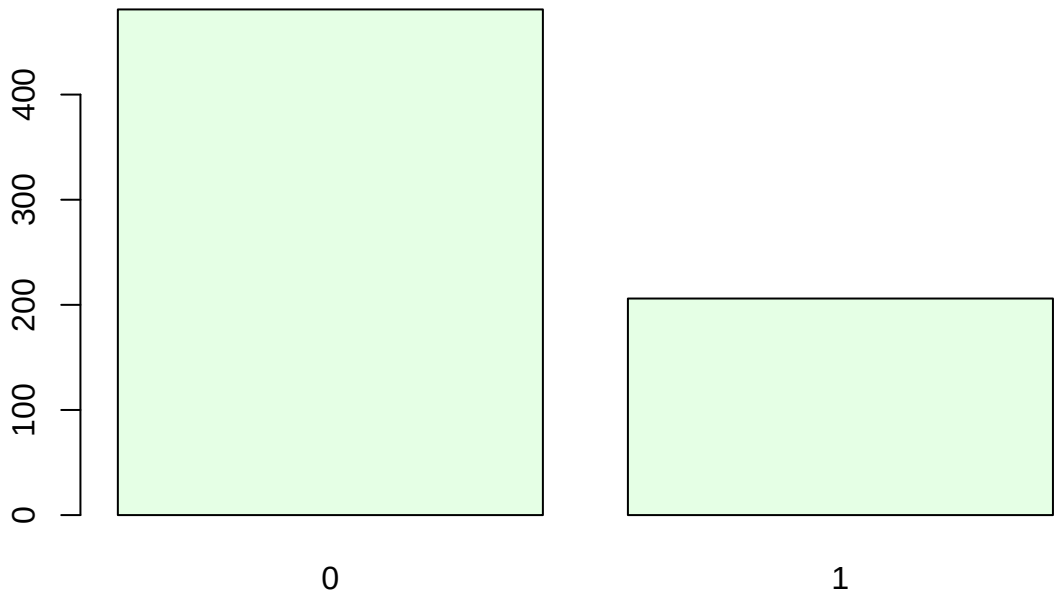
```
##
##              A    B    C    D    E    F    G    T  Sum
##   1     67   22   65   94   40   34    0    0    1  323
##   2    254    0    0    0    6    4   13    0    0  277
##   3    693    0    0    0    0    3    8    5    0  709
##   Sum 1014   22   65   94   46   41   21    5    1 1309
```

De aquí lo más relevante es la suma de valores faltante 77%(1014) respecto al total de valores que podemos ver en la última columna (sum) con el valor de 1309, Además no vemos una relación clara entre la clase y la cabina, podemos decir que. *primera clase: A B C D E con mayoría en la cabina C>B>D*
segunda clase: D E F (no contamos con suficientes registros para establecer mayorías claras entre los 3 grupos) *Tercera clase: E F G (no contamos con suficientes registros para establecer mayorías claras entre los 3 grupos)

Aún a pesar tener las cabinas que se asignan más o menos a cada clase considero que en este caso por el volumen tan alto de valores missing es mejor no imputarlos.

Pero si podemos hacer una última comprobación que me resulta interesante y es el contrastar si los datos missing se corresponden mayoritariamente con los pasajeros que no han sobrevivido, de ser así es posible que haya una causa para esto y una opción puede ser asignar una cubierta ficticia a las cabinas desconocidas, y así poder trabajar con los datos restantes, y ver si los datos de la cabina ficticia se comportan igual aunque, es posible que esto requiera la creación de variables adicionales, y por la limitación de tiempo no se si podré abarcar este aspecto.

```r
#quiero verificar si los datos faltantes corresponden mayoritariamente a los pasajeros
#que no sobrevivieron al accidente por lo que creo una gráfica

plot(Titanic_complete[Titanic_complete$Cabin=="","Survived"],col = rgb(0, 1, 0, 0.10))
```




```
Titanic_complete$Cabin<-as.character(Titanic_complete$Cabin)
#substituyo los valores missing por una N
#De este modo se podrá trabajar con los demás y tal vez incluirlos en el modelo.
Titanic_complete[Titanic_complete$Cabin=="","Cabin"]<-"N"
Titanic_complete[Titanic_complete$CabinG=="","CabinG"]<-"N"
```

Como se esperaba la mayoria de datos faltantes corresponden a pasajeros que no han sobrevivido. Compruebo que los cambios se han hecho correctamente.

```
sum(is.na(Titanic_complete["Cabin"]))
```

```
## [1] 0
```

```
sum(is.na(Titanic_complete["CabinG"]))
```

```
## [1] 0
```

```
Titanic_complete$Cabin<-as.factor(Titanic_complete$Cabin)
Titanic_complete$CabinG<-as.factor(Titanic_complete$CabinG)

head(Titanic_complete)
```

```
##   PassengerId Survived Pclass
```

```
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                  Name    Sex Age SibSp Parch
## 1                             Braund, Mr. Owen Harris   male  22     1     0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
## 3                              Heikkinen, Miss. Laina female  26     0     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
## 5                            Allen, Mr. William Henry   male  35     0     0
## 6                                    Moran, Mr. James   male  27     0     0
##             Ticket    Fare Cabin Embarked Set_type CabinG
## 1        A/5 21171  7.2500     N        S    train      N
## 2         PC 17599 71.2833   C85        C    train      C
## 3 STON/O2. 3101282  7.9250     N        S    train      N
## 4           113803 53.1000  C123        S    train      C
## 5           373450  8.0500     N        S    train      N
## 6           330877  8.4583     N        Q    train      N
```

Gestion de zeros para la variable Fare

Como hemos visto en el analisis preeliminar hay 15 valores zero para la tarifa. Las tarifas deberían estar relacionadas con los tickets, así que compruebo si buscando los mismos números de tickets tenemos el mismo número de registros que cuando buscamos los registros que tienen una tarifa de 0 ya que si un ticket tiene un número igual pero con un valor diferente para la tarifa, tendremos más registros, y es posible que el valor faltante sea igual.

```
#comparación del número de registros de variable fare=0 y
#número de registros correspondientes a los tickets
Titanic_complete[Titanic_complete$Fare==0,]
```

```
##      PassengerId Survived Pclass                                Name  Sex Age
## 180          180        0      3                 Leonard, Mr. Lionel male  36
## 264          264        0      1                Harrison, Mr. William male  40
## 272          272        1      3          Tornquist, Mr. William Henry male  25
## 278          278        0      2           Parkes, Mr. Francis "Frank" male  32
## 303          303        0      3       Johnson, Mr. William Cahoone Jr male  19
## 414          414        0      2        Cunningham, Mr. Alfred Fleming male  26
## 467          467        0      2                Campbell, Mr. William male  24
## 482          482        0      2      Frost, Mr. Anthony Wood "Archie" male  30
## 598          598        0      3                  Johnson, Mr. Alfred male  49
## 634          634        0      1         Parr, Mr. William Henry Marsh male  33
## 675          675        0      2            Watson, Mr. Ennis Hastings male  27
## 733          733        0      2                 Knight, Mr. Robert J male  22
## 807          807        0      1               Andrews, Mr. Thomas Jr male  39
## 816          816        0      1                    Fry, Mr. Richard male  30
## 823          823        0      1       Reuchlin, Jonkheer. John George male  38
## 1158        1158     <NA>      1 Chisholm, Mr. Roderick Robert Crispin male  35
## 1264        1264     <NA>      1               Ismay, Mr. Joseph Bruce male  49
##      SibSp Parch Ticket Fare       Cabin Embarked Set_type CabinG
## 180      0     0   LINE    0           N        S    train      N
```

```
## 264      0     0 112059    0         B94        S    train      B
## 272      0     0   LINE    0           N        S    train      N
## 278      0     0 239853    0           N        S    train      N
## 303      0     0   LINE    0           N        S    train      N
## 414      0     0 239853    0           N        S    train      N
## 467      0     0 239853    0           N        S    train      N
## 482      0     0 239854    0           N        S    train      N
## 598      0     0   LINE    0           N        S    train      N
## 634      0     0 112052    0           N        S    train      N
## 675      0     0 239856    0           N        S    train      N
## 733      0     0 239855    0           N        S    train      N
## 807      0     0 112050    0         A36        S    train      A
## 816      0     0 112058    0        B102        S    train      B
## 823      0     0  19972    0           N        S    train      N
## 1158     0     0 112051    0           N        S     test      N
## 1264     0     0 112058    0 B52 B54 B56        S     test      B
```

```
nrow(Titanic_complete[Titanic_complete$Fare==0,])
```

```
## [1] 17
```

```
a<-unique(Titanic_complete[Titanic_complete$Fare==0,"Ticket"])

sum_final=0
for (i in a){
  filas= nrow(Titanic_complete[Titanic_complete$Ticket==i,])
  sum_final=sum_final+filas
}
sum_final
```

```
## [1] 17
```

El número de registros es el mismo, por lo que o bien los valores de las tarifas son correctos para esos número de tickets o tenemos que encontrar los valores de otra manera.

Después de esto he buscado información sobre la variable Fare. Por lo que he encontrado es una variable compuesta con varios factores que influyen en la misma, había tarifas especiales en función de la edad y de si se adquirían en grupo o si se compraban en algún "pack" que incluyese acceso al barco y a algún tren, además el precio cambiaba en función del país de compra y habían algunos pasajeros trabajadores de los dueños de la compañia que viajaron gratis por lo que he optado por mantener los zeros y no tratarlos ya que seria plausible que este número reducido de valores fuera correcto aunque no habitual, después de todo representa cerca del 1% de todos los registros con los que contamos.

3.2 Identificación y tratamiento de valores extremos(Outliers o valors atípicos).

Para comprobar los valores extremos haré visualizaciones de las variables numéricas, Age, SibSp, Parch y Fare, y con la función boxplot.stats extraeré los valores extremos en cuestión.

Para la variable Age, los datos perdidos han sido tratados y fuera de eso en la revisión inicial que he realizado no he detectado valores atípicos, ya que todos los datos encontrados corresponden a valores normales que podríamos esperar para edades, no obstante si consideramos la representación del boxplot de la muestra podemos contar algunos valores "outliers", pero que no trataré, porque aunque atípicos son valores que forman parte de la muestra.

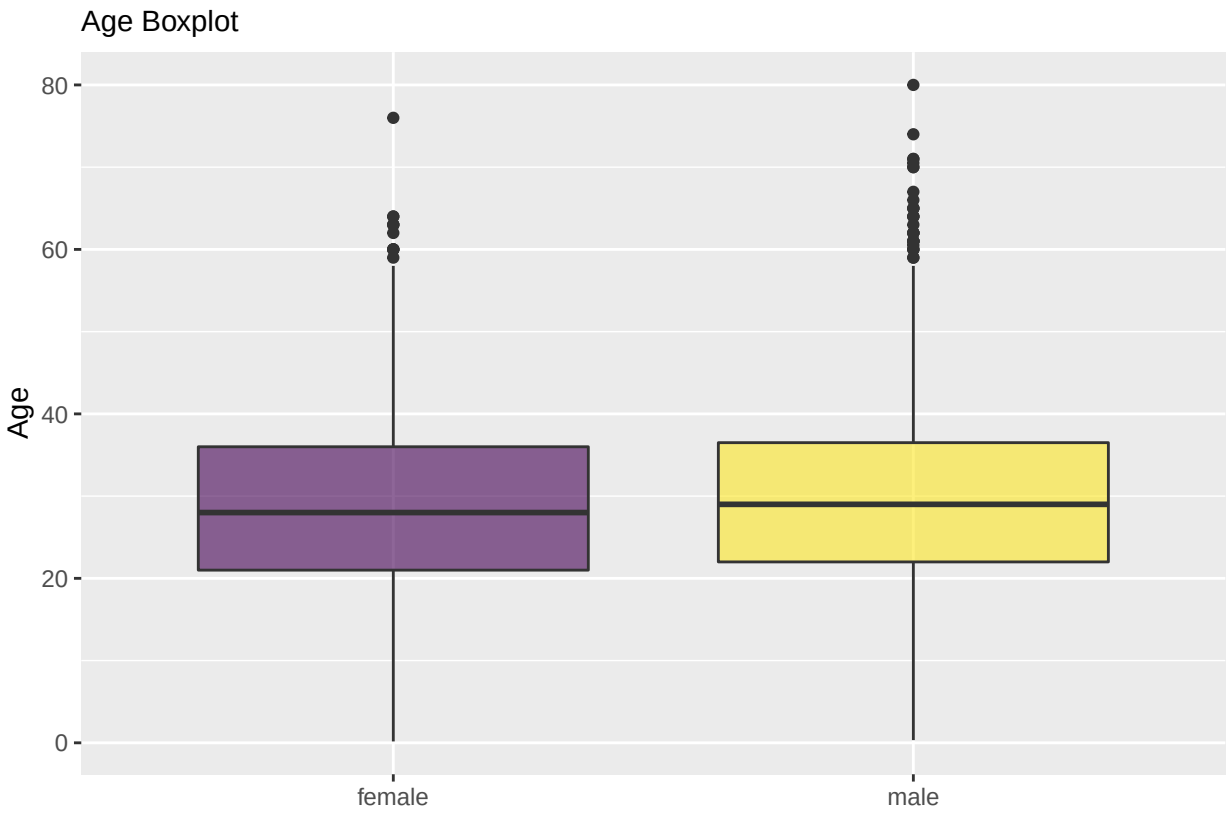
Hago un gráfico de densidades para tener una representación gráfica de apoyo. Además me ha resultado interesante plantear un gráfico boxplot para ver los valores atípicos en cuanto a edad y sexo.

El resultado del análisis es el siguiente:

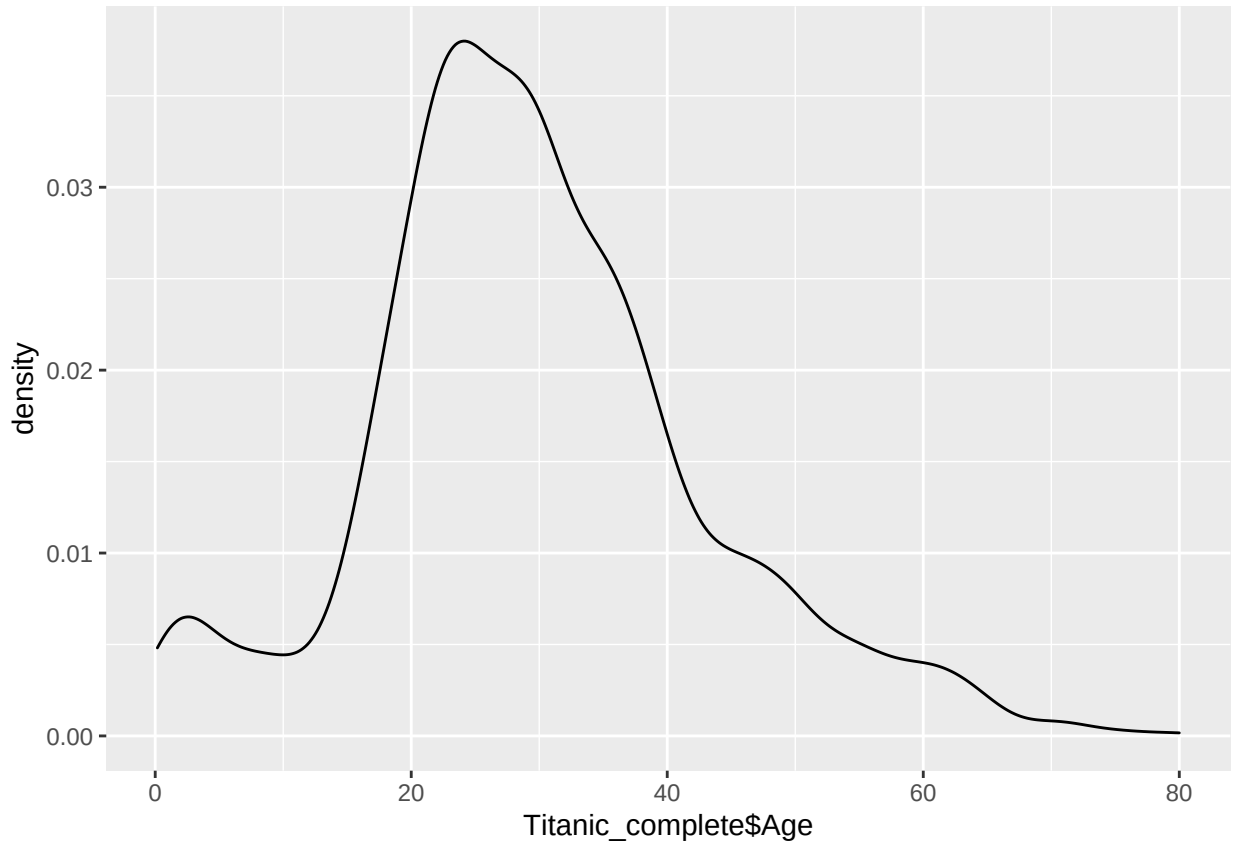
```
#recuento de posibles valores outliers

paste ("los valores atípicos encontrados de acuerdo a la función boxplot.stats:",
       length(boxplot.stats(Titanic_complete$Age)$out))
```

```
## [1] "los valores atípicos encontrados de acuerdo a la función boxplot.stats: 44"
```

```
#Gráfico boxplot y de densidades

Titanic_complete %>%
  ggplot( aes(x=Sex,y=Age, fill=Sex)) +
    geom_boxplot() +
    scale_fill_viridis(discrete = TRUE, alpha=0.6) +
    theme(
      legend.position="none",
      plot.title = element_text(size=11)
    ) +
    ggtitle("Age Boxplot") +
    xlab("")
```



```
ggplot(mapping = aes(x=Titanic_complete$Age))+geom_density()
```



Del gráfico de densidades deduzco que la distribución de la variable sigue una normal y pueden haber valores "atípicos", pero deben considerarse como una parte de la muestra . En cuanto a los boxplots planteados la información que tenemos es que los hombres tienen una mediana de edad más elevada y además también habían hombres mayores viajanndo de hasta 80 años, es posible que esto también afectara a su supervivencia en cuanto a que los desplazamientos en un barco en esa situación posiblemente fueron más difíciles para personas mayores.

Para las variables SibSp y Parch. He hecho representaciones en una tabla donde he contado el número de registros que tienen cada valor de SibSp o Parch.

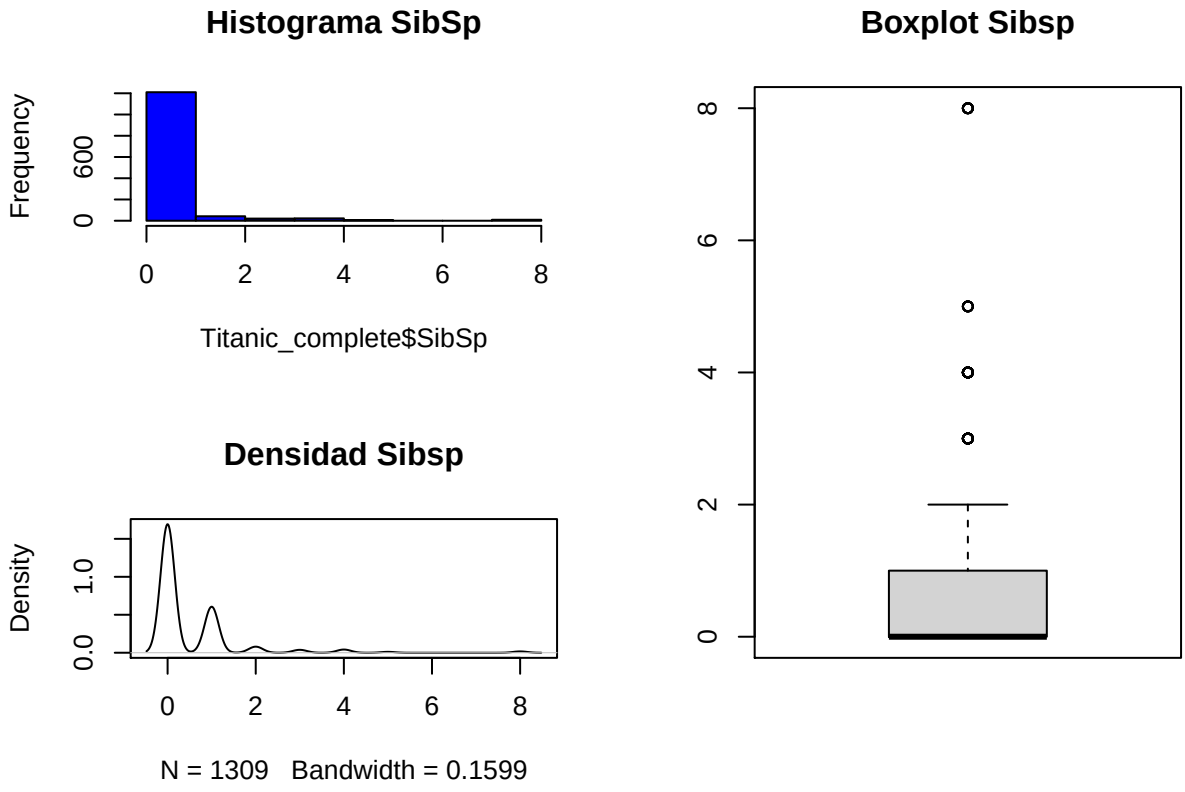
Los valores entran dentro de lo que ya esperaba tras haber hecho anteriormente uso de la función summary y revisando el contexto de la época, no es descabellado pensar que hubieran personas con 8 o incluso 9 hijos. En el siglo XX hubieron grandes avances médicos por los que la supervivencia de los recién nacidos era más elevada y el uso de anticonceptivos aún no estaba demasiado extendido. Voy a considerar por tanto validos todos los valores.

Además teniendo en cuenta la función Boxplot.stats hago un recuento de los valores que se considerarían atípicos, para ver en que proporción afectan al total de registros.

```
table(Titanic_complete$SibSp)
```

```
##
##   0   1   2   3   4   5   8
## 891 319  42  20  22   6   9
```

```
layout(matrix(c(1,3,2,3), 2, 2, byrow = TRUE))
hist(Titanic_complete$SibSp ,col="blue",main=" Histograma SibSp ",breaks =10)
plot(density(Titanic_complete$SibSp),main="Densidad Sibsp")
boxplot(Titanic_complete$SibSp ,main="Boxplot Sibsp")
```



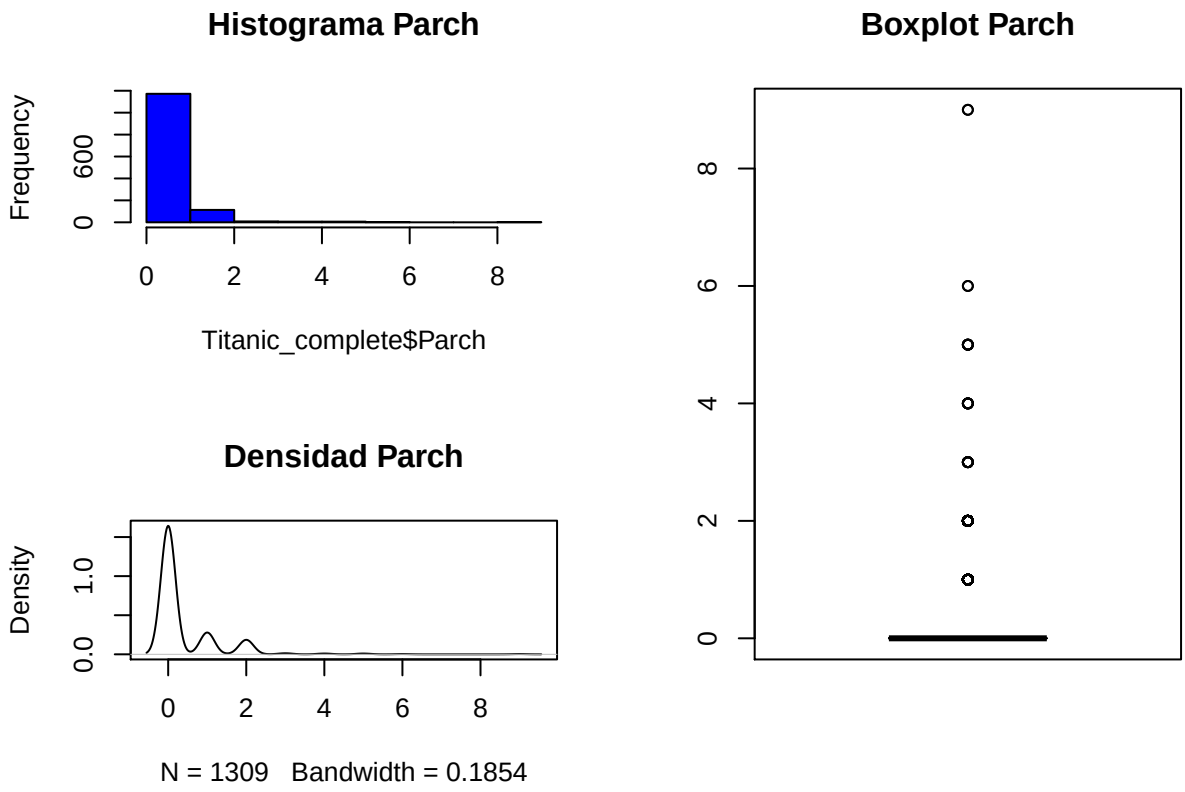

```
SibSp.outlier <-length(boxplot.stats(Titanic_complete$SibSp)$out)
paste("los valores atípicos de acuerdo a la función boxplot.stats: ",SibSp.outlier)
```

```
## [1] "los valores atípicos de acuerdo a la función boxplot.stats:  57"
```

```
table(Titanic_complete$Parch)
```

```
##
##    0    1    2    3    4    5    6    9
## 1002  170  113    8    6    6    2    2
```

```
layout(matrix(c(1,3,2,3), 2, 2, byrow = TRUE))
hist(Titanic_complete$Parch ,col="blue",main=" Histograma Parch ",breaks =10)
plot(density(Titanic_complete$Parch),main="Densidad Parch")
boxplot(Titanic_complete$Parch ,main="Boxplot Parch")
```




```
Parch.outlier <-length(boxplot.stats(Titanic_complete$Parch)$out)
paste("los valores atípicos de acuerdo a la función boxplot.stats: ",Parch.outlier)
```

```
## [1] "los valores atípicos de acuerdo a la función boxplot.stats:  307"
```

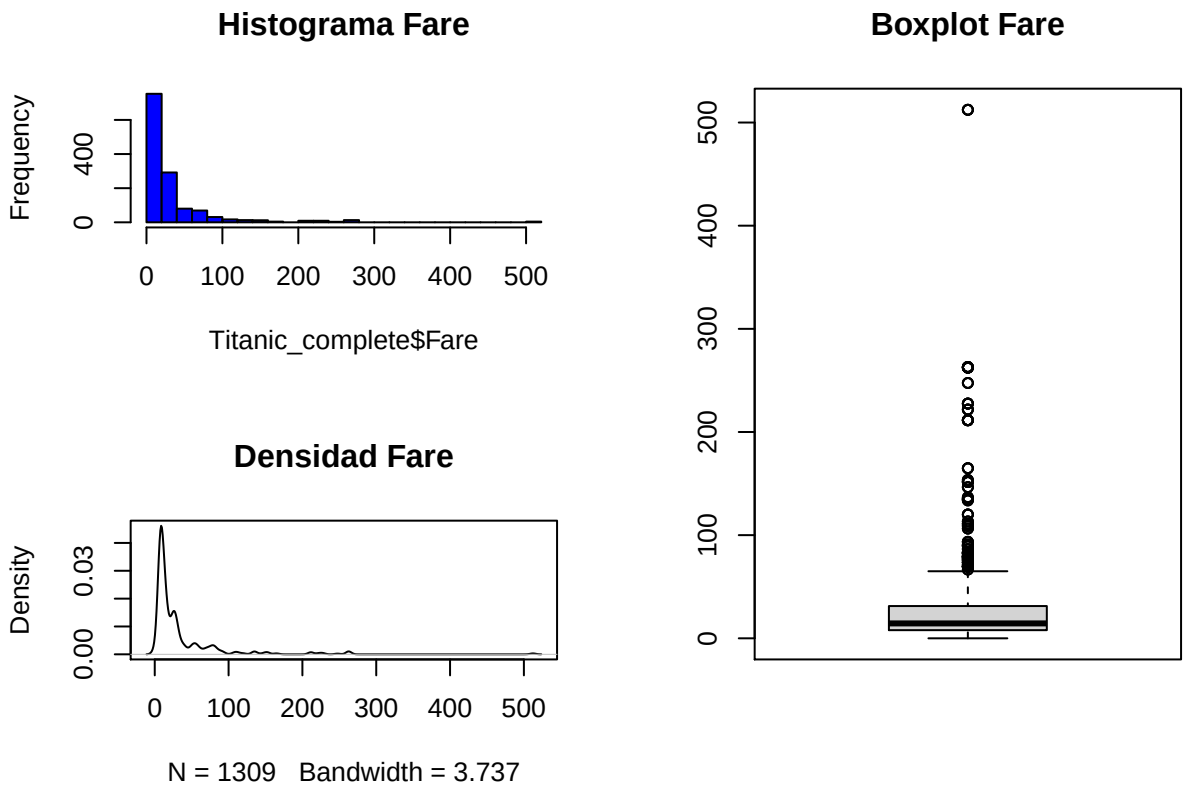
Los gráficos muestran distribuciones similares y concuerdan con lo que he podido comprobar con la función Table.

Para la variable Fare, igual que en los casos anteriores planteo un boxplot, un histograma y un grafico de densidad y hago un recuento de los valores suceptibles de considerarse outliers

```
Fare.outlier <-length(boxplot.stats(Titanic_complete$Fare)$out)
paste("los valores atípicos son: ",
      Fare.outlier)
```

```
## [1] "los valores atípicos son:  171"
```

```
layout(matrix(c(1,3,2,3), 2, 2, byrow = TRUE))
hist(Titanic_complete$Fare ,col="blue",main=" Histograma Fare ",breaks =30)
plot(density(Titanic_complete$Fare),main="Densidad Fare")
boxplot(Titanic_complete$Fare ,main="Boxplot Fare")
```



De el resultado veo que hay valores que en efecto parecen atípicos, sobre todo los de precios superiores a las 500 libras. Voy a comprobar los registros para verificar que la tarifa corresponde a un ticket de primera clase ya que, si no fuera así podemos asumir que en efecto se trata de un valor atípico.

```
Titanic_complete[Titanic_complete$Fare>400,]
```

```
##      PassengerId Survived Pclass
## 259          259        1      1
## 680          680        1      1
## 738          738        1      1
## 1235        1235     <NA>      1
##                                                                    Name    Sex Age
## 259                                                    Ward, Miss. Anna female  35
## 680                                       Cardeza, Mr. Thomas Drake Martinez   male  36
## 738                                                  Lesurer, Mr. Gustave J   male  35
## 1235 Cardeza, Mrs. James Warburton Martinez (Charlotte Wardle Drake) female  58
##      SibSp Parch   Ticket     Fare       Cabin Embarked Set_type CabinG
## 259      0     0 PC 17755 512.3292           N        C    train      N
## 680      0     1 PC 17755 512.3292 B51 B53 B55        C    train      B
## 738      0     0 PC 17755 512.3292        B101        C    train      B
## 1235     0     1 PC 17755 512.3292 B51 B53 B55        C     test      B
```

Vemos que hay cuatro registros con la misma tarifa superior a 500 libras y todos corresponden al mismo número de ticket y todos a primera clase, así que sería muy posible que este número de ticket tenga esa tarifa y que el valor no sea atípico o Outlier. Para comprobarlo aplico el filtro por número de ticket y cuando hago esto compruebo que el ticket PC 17755 tiene siempre el mismo precio 512.3292.

```
Titanic_complete[Titanic_complete$Ticket=="PC 17755",]
```

```
##      PassengerId Survived Pclass
## 259          259        1      1
## 680          680        1      1
## 738          738        1      1
## 1235        1235     <NA>      1
##                                                                  Name    Sex Age
## 259                                                Ward, Miss. Anna female  35
## 680                                 Cardeza, Mr. Thomas Drake Martinez   male  36
## 738                                             Lesurer, Mr. Gustave J   male  35
## 1235 Cardeza, Mrs. James Warburton Martinez (Charlotte Wardle Drake) female  58
##      SibSp Parch   Ticket     Fare       Cabin Embarked Set_type CabinG
## 259      0     0 PC 17755 512.3292           N        C    train      N
## 680      0     1 PC 17755 512.3292 B51 B53 B55        C    train      B
## 738      0     0 PC 17755 512.3292        B101        C    train      B
## 1235     0     1 PC 17755 512.3292 B51 B53 B55        C     test      B
```

Interpretando estos resultados la conclusión a la que llego es que a pesar de que los valores son atipicos no se deben a errores y no se deben corregir ya que forman parte de la muestra. Los valores muy reducidos y 0 también hemos visto tienen una explicación, por lo que consideraremos que son valores que forman parte de la muestra y representan la propia variación de la misma

Otras comprobaciones en la limpieza de datos.

Fuera del tratamiento de valores perdidos, de los outliers y del cambio de formato haré algunas comprobaciones más para tener datos con los que sea más fácil trabajar y que aporten más valor al futuro modelo.

- Para la variable PassengerId comprobare que no hayan registros repetidos con el mismo passengerId

```
length(unique(Titanic_complete$PassengerId))
```

```
## [1] 1309
```

La conclusión que obtengo de esto es que ningun PassengerId esta repetido. Si alguno estuviera repetido el número de registros únicos sería menor.

- Para la variable Fare

Inicialmente plantee redondearla para simplificar los datos, pero según he avanzado con el análisis he visto que realmente no comportaba un cambio sustancial, así que mantendré la variable tal y como está en lugar de usar la función round como tenía pensado.

- Para la variable Name

Esta variable tal y como esta expresada no aporta demasiado valor ni podemos extraer datos relevantes de la misma, pero considerando los apellidos tal vez podrían servirnos para establecer relaciones familiares, y nos dan información sobre los títulos (Miss, Mr, Master, etc)

```r
#divido la variable name en dos nuevas columnas una para el apellido
#y la otra para el título

vars <- c("Surname","Name2")
Titanic_complete<- separate(Titanic_complete, Name, into = vars,
                            sep = c(","), remove=FALSE,extra ="drop")%>%
  separate(Name2, into = c("Title","namerest"), sep = c(". "),
           extra="warn")
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 845 rows [1, 2, 4, 5,
## 7, 8, 9, 10, 11, 13, 14, 15, 16, 18, 19, 21, 23, 24, 25, 26, ...].
```

```r
#elimino la columna residual del nombre que se ha generado al crear dos columnas
#más una para el apellido y otra para el título
Titanic_complete$namerest<-NULL
# compruebo el dataframe actualizado con los cambios
head(Titanic_complete)
```

```
##   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   Surname Title    Sex
## 1                            Braund, Mr. Owen Harris    Braund    Mr   male
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer)   Cumings   Mrs female
## 3                             Heikkinen, Miss. Laina Heikkinen  Miss female
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel)  Futrelle   Mrs female
## 5                           Allen, Mr. William Henry     Allen    Mr   male
## 6                                   Moran, Mr. James     Moran    Mr   male
##   Age SibSp Parch           Ticket    Fare Cabin Embarked Set_type CabinG
## 1  22     1     0        A/5 21171  7.2500     N        S    train      N
## 2  38     1     0         PC 17599 71.2833   C85        C    train      C
## 3  26     0     0 STON/O2. 3101282  7.9250     N        S    train      N
## 4  35     1     0           113803 53.1000  C123        S    train      C
## 5  35     0     0           373450  8.0500     N        S    train      N
## 6  27     0     0           330877  8.4583     N        Q    train      N
```

```r
Titanic_complete$Title<- str_trim(Titanic_complete$Title)
Titanic_complete$Surname<- as.factor(str_trim(Titanic_complete$Surname))

#revisión de registro con titulo no reconocible
Titanic_complete[Titanic_complete$Title=="th",]
```

```
##     PassengerId Survived Pclass
## 760         760        1      1
##                                                  Name Surname Title
## 760 Rothes, the Countess. of (Lucy Noel Martha Dyer-Edwards)  Rothes    th
##        Sex Age SibSp Parch Ticket Fare Cabin Embarked Set_type CabinG
## 760 female  33     0     0 110152 86.5   B77        S    train      B
```

El valor no reconocible corresponde a “the countes” al tener un espacio entre “the”, “countess” no se ha separado correctamente, como solo es un registro no considero necesario revisar el código ya que con la función table me ha permitido revisar que era el único título atípico, pero intentaré optimizarlo, en la revisión final.

Después de esto agrupamos los Titulos de acuerdo a los siguientes grupos “elite” (que incluye trabajos de altas categorias sociales como reverendos o doctores o titulos heredados de nobleza que otorgan una categoría de “elite”),non_elite (que incluye Miss, Mr,Ms,Mlle,Mme y Mrs), a pesar de que creo que de este grupo podríamos crear nuevas variables sobre mujeres casadas o solteras y hombres que no tienen ningún título ni profesion relevante, además de separar totalmente el grupo de nobles voy a dejar la primera clasificación de este modo, pero no lo descarto para futuras mejoras del modelo.

```
#agrupación de categorías.

Titanic_complete$Title[Titanic_complete$Title %in%
                         c('Capt','Col','Dr','Major',
                           'Rev','Master','Don','Jonkheer',
                           'Sir','Lady','Dona','th')] <-"elite"
Titanic_complete$Title[Titanic_complete$Title %in%
                         c('Miss', 'Ms', 'Mlle','Mr',
                           'Mrs','Mme')]<- 'non_elite'

Titanic_complete$Title<-as.factor(Titanic_complete$Title)

table(Titanic_complete$Title)
```

```
##
##     elite non_elite
##        90      1219
```

- Para las variables Sibsp y Parch

Para estas variables no haré cambios. Pero si considero interesante generar una nueva que incluya los integrantes totales de la familia de un pasajero. para determinar si hay una relación entre e tamaño de una familia y la supervivencia. Esta nueva variable será redundante con las otras dos variables pero creo que puede aportar valor al modelo.

```
#creo la variable Family.unit
Titanic_complete$Family.unit<-Titanic_complete$SibSp+
  Titanic_complete$Parch+1
#compruebo que la variaable se ha creado correctamente
head(Titanic_complete)
```

```
##   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   Surname     Title
## 1                             Braund, Mr. Owen Harris    Braund non_elite
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer)   Cumings non_elite
## 3                              Heikkinen, Miss. Laina Heikkinen non_elite
```

```
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel)  Futrelle non_elite
## 5                             Allen, Mr. William Henry     Allen non_elite
## 6                                     Moran, Mr. James     Moran non_elite
##      Sex Age SibSp Parch           Ticket    Fare Cabin Embarked Set_type
## 1   male  22     1     0        A/5 21171  7.2500     N        S    train
## 2 female  38     1     0         PC 17599 71.2833   C85        C    train
## 3 female  26     0     0 STON/O2. 3101282  7.9250     N        S    train
## 4 female  35     1     0           113803 53.1000  C123        S    train
## 5   male  35     0     0           373450  8.0500     N        S    train
## 6   male  27     0     0           330877  8.4583     N        Q    train
##   CabinG Family.unit
## 1      N           2
## 2      C           2
## 3      N           1
## 4      C           2
## 5      N           1
## 6      N           1
```

Una vez limpiados los datos y creadas las nuevas variables vuelvo a revisar los datos de los que dispongo con las funciones str y summary

```
str(Titanic_complete)
```

```
## 'data.frame':    1309 obs. of  17 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       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
##  $ Surname    : Factor w/ 875 levels "Abbing","Abbott",..: 101 183 335 273 16 544 506 614 388 565 ..
##  $ Title      : Factor w/ 2 levels "elite","non_elite": 2 2 2 2 2 2 2 1 2 2 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 27 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",..: 721 817 915 66 650 374 110 542 478 175 ..
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : Factor w/ 187 levels "A10","A11","A14",..: 186 107 186 71 186 186 164 186 186 186 ..
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
##  $ Set_type   : Factor w/ 2 levels "test","train": 2 2 2 2 2 2 2 2 2 2 ...
##  $ CabinG     : Factor w/ 9 levels "A","B","C","D",..: 8 3 8 3 8 8 5 8 8 8 ...
##  $ Family.unit: num  2 2 1 2 1 1 1 5 3 2 ...
```

```
summary(Titanic_complete)
```

```
##   PassengerId   Survived   Pclass      Name              Surname
##  Min.   :   1   0   :549   1:323   Length:1309        Andersson:  11
##  1st Qu.: 328   1   :342   2:277   Class :character   Sage     :  11
##  Median : 655   NA's:418   3:709   Mode  :character   Asplund  :   8
##  Mean   : 655                                         Goodwin  :   8
##  3rd Qu.: 982                                         Davies   :   7
##  Max.   :1309                                         Brown    :   6
##                                                       (Other)  :1258
```

```
##        Title          Sex           Age            SibSp           Parch
##  elite    :  90   female:466   Min.   : 0.17   Min.   :0.0000   Min.   :0.000
##  non_elite:1219   male  :843   1st Qu.:22.00   1st Qu.:0.0000   1st Qu.:0.000
##                                Median :29.00   Median :0.0000   Median :0.000
##                                Mean   :29.84   Mean   :0.4989   Mean   :0.385
##                                3rd Qu.:36.50   3rd Qu.:1.0000   3rd Qu.:0.000
##                                Max.   :80.00   Max.   :8.0000   Max.   :9.000
##
##       Ticket          Fare                      Cabin      Embarked  Set_type
##  CA. 2343:  11   Min.   :  0.000   N              :1014    :  0    test :418
##  1601    :   8   1st Qu.:  7.896   C23 C25 C27    :   6   C:270    train:891
##  CA 2144 :   8   Median : 14.454   B57 B59 B63 B66:   5   Q:123
##  3101295 :   7   Mean   : 33.281   G6             :   5   S:916
##  347077  :   7   3rd Qu.: 31.275   B96 B98        :   4
##  347082  :   7   Max.   :512.329   C22 C26        :   4
##  (Other) :1261                     (Other)        : 271
##      CabinG      Family.unit
##  N      :1014   Min.   : 1.000
##  C      :  94   1st Qu.: 1.000
##  B      :  65   Median : 1.000
##  D      :  46   Mean   : 1.884
##  E      :  41   3rd Qu.: 2.000
##  A      :  22   Max.   :11.000
##  (Other):  27
```

El resultado es que tengo más variables pero no todas pasarán a la fase de análisis. Además ya no hay missing values (excepto en el apartado "Survived" donde es normal ya que hemos combinado los datos de train y test).

Y todos los datos están en formatos adecuados para su tratamiento (númerico o factor), exceptuando la variable Name, pero es una variable que no pasará a la fase de análisis.

Por otra parte indicar que a pesar de que hasta aquí he realizado la mayor parte de tareas relativas a la limpieza de los datos es posible que en el siguiente apartado con la selección del modelo y la selección de variables realice alguna tarea más que se considere de limpieza de datos.

4. Analisis de datos.

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

El objetivo del análisis es predecir la supervivencia de los pasajeros del Titanic en el grupo Test.

Los análisis se harán en base al grupo de datos de Entrenamiento, por lo que antes de iniciar las pruebas estadisticas del punto 4.3 hare la separación de datos otra vez en grupos "Train" y "Test" En cuanto a las variables que utilizaré a partir de ahora serán:

PassangerID (solo con finalidad de identificación de registro) set_type (solo con finalidad de volver a separar datos train/test) *Pclass* Title *Sex* Age *SibSp/Parch/Family.unit (la información proporcionada por estas variables puede ser redundante, por lo que es posible que solo utilice Family.unit, pero quiero hacer algunas comprobaciones antes de decidirlo y valorar si tal vez el uso de PCA podría ser útil)* Fare *CabinG* Embarked.

Selección de variables.

- Separación de variables en numéricas o categóricas
  - Para las variables numéricas (histogramas, boxplot (ya realizado en apartado "limpieza de datos")), correlación entre variables)
    - Decisión sobre las variables cuantitativas a utilizar en el análisis
  - Para variables categóricas (gráficos de barras)
    - Decisión sobre las variables categóricas a utilizar en el análisis

Análisis exploratorio

- Relación entre variables y supervivencia.
  - ¿Qué incrementa la supervivencia?¿edad, sexo,clase,tarifa,puerto de embarque?
  - los títulos nobiliarios garantizan una mayor supervivencia?¿los miembros de famílias numerosas tenían menos posibilidades de salvarse?¿la ubicación de las cabinas en relación al punto de colapso del barco nos indica cuál es la mejor cubierta para sobrevivir al accidente?¿que factores tiene el pasajero 892 a favor y en contra?
- Decisión sobre las variables finales a utilizar en el análisis, reducción de dataframe y binarización de variables

————————*Desarrollo del punto 4.1*————————-

Para seleccionar las variables primero quiero hacer un estudio preeliminar independiente de cada variable, el tratamiento de las variables categóricas será diferente del que de a las variables numéricas por lo que lo primero que haré será separar las variables en 2 grupos.

```
#identificación de las variables factor y variables numericas
id.factor<- c(2,3,5,6,7,14,16)
id.numeric2<- c(8,9,10,12,17)
id.numeric<-c(8,9,10,12)
var.factor<-colnames(Titanic_complete)[id.factor]
var.numeric2<-colnames(Titanic_complete)[id.numeric2]#incluye family.unit
var.numeric<-colnames(Titanic_complete)[id.numeric]#no incluye family.unit
head(Titanic_complete[var.factor])
```

```
##   Survived Pclass   Surname     Title    Sex Embarked CabinG
## 1        0      3    Braund non_elite   male        S      N
## 2        1      1   Cumings non_elite female        C      C
## 3        1      3 Heikkinen non_elite female        S      N
## 4        1      1  Futrelle non_elite female        S      C
## 5        0      3     Allen non_elite   male        S      N
## 6        0      3     Moran non_elite   male        Q      N
```

```
head(Titanic_complete[var.numeric])
```

```
##   Age SibSp Parch    Fare
## 1  22     1     0  7.2500
## 2  38     1     0 71.2833
## 3  26     0     0  7.9250
## 4  35     1     0 53.1000
## 5  35     0     0  8.0500
## 6  27     0     0  8.4583
```

Selección de variables: Variables numéricas A continuación hare comprobaciones sobre las variables numéricas para decidir las que participarán en el modelo.

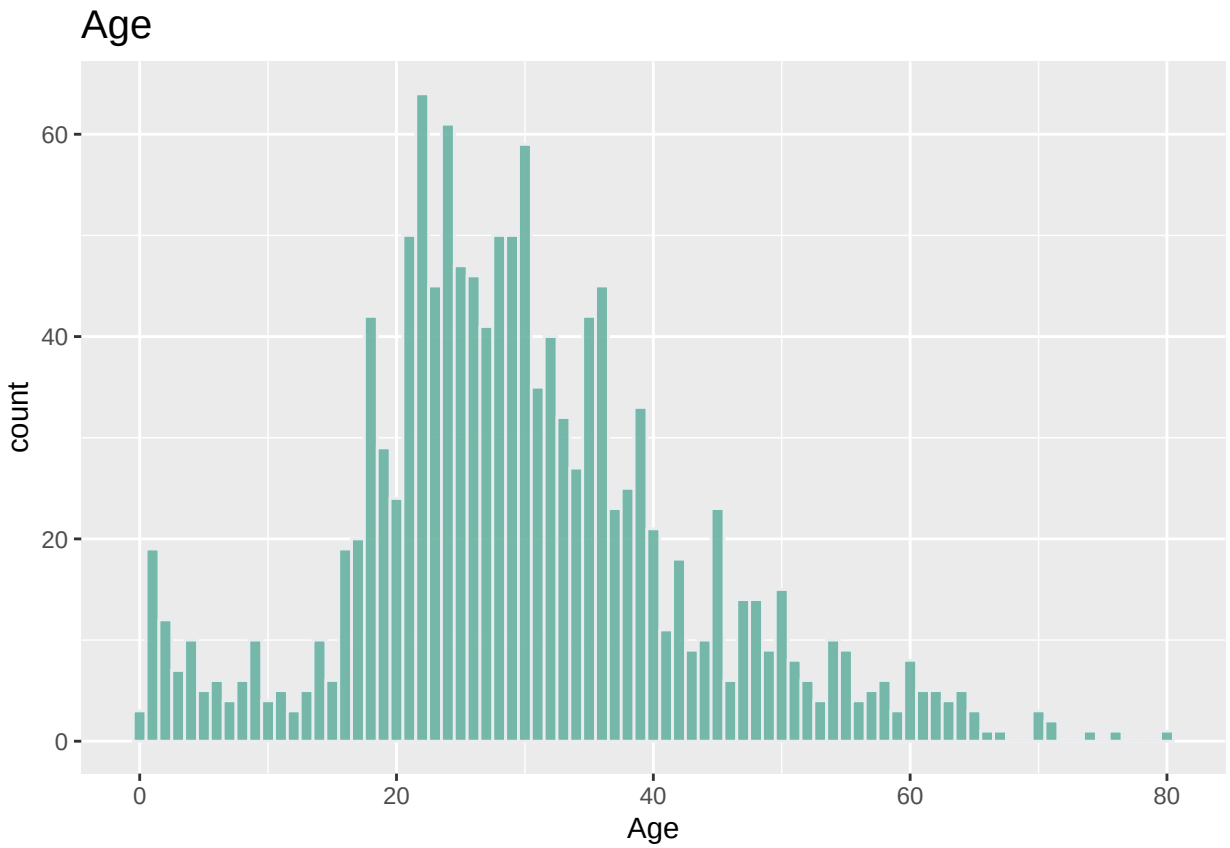
```
#Histogramas para variables numéricas(cuantitativas), esta revisión nos servirá también
#para evaluar la normalidad de las variables, lo que se verá al detalle en el punto 4.2
require(gridExtra)
```

```
## Loading required package: gridExtra
```

```
##
## Attaching package: 'gridExtra'
```

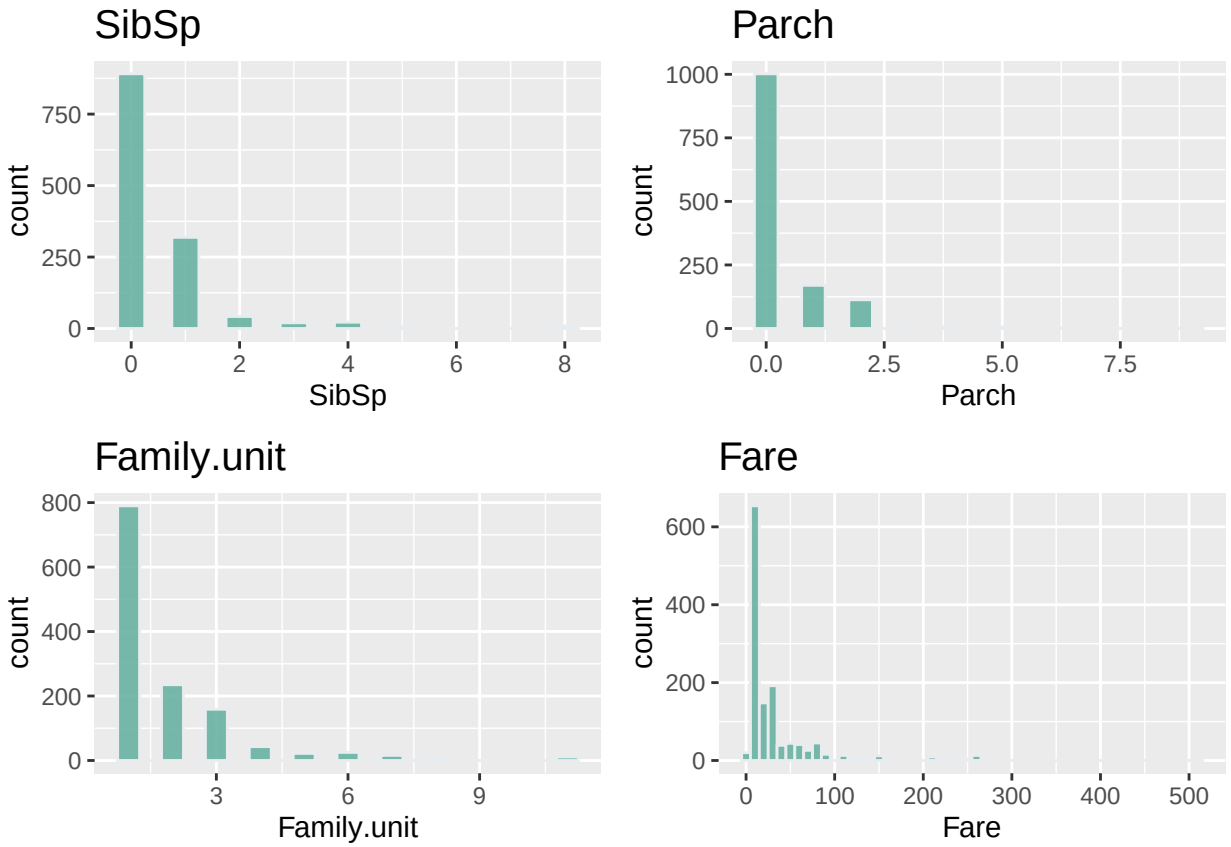
```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
ggplot(Titanic_complete,aes(x=Age)) +
  geom_histogram( binwidth=1, fill="#69b3a2", color="#e9ecef", alpha=0.9) +
  ggtitle("Age") +theme(plot.title = element_text(size=15))
```




```
plot1<-ggplot(Titanic_complete,aes(x=SibSp)) +
  geom_histogram( binwidth=0.5, fill="#69b3a2", color="#e9ecef", alpha=0.9) +
  ggtitle("SibSp") +theme(plot.title = element_text(size=15))
```

```
plot2<-ggplot(Titanic_complete,aes(x=Parch)) +
  geom_histogram( binwidth=0.5, fill="#69b3a2", color="#e9ecef", alpha=0.9) +
  ggtitle("Parch") +theme(plot.title = element_text(size=15))

plot3<-ggplot(Titanic_complete,aes(x=Family.unit)) +
  geom_histogram( binwidth=0.5, fill="#69b3a2", color="#e9ecef", alpha=0.9) +
  ggtitle("Family.unit") +theme(plot.title = element_text(size=15))

plot4<-ggplot(Titanic_complete,aes(x=Fare)) +
  geom_histogram( binwidth=10, fill="#69b3a2", color="#e9ecef", alpha=0.9) +ggtitle("Fare") +
  theme(plot.title = element_text(size=15))

grid.arrange(plot1,plot2,plot3,plot4, ncol=2)
```



De las variables numéricas podemos ver que la única que tiene una distribución similar a la normal es la variable Age. Por los gráficos que obtenemos de las demás variables podemos tal vez considerar aproximarlas a una distreibución normal creo que esto es especialmente interesante para la variable Fare, en la que el rango de valores es muy amplio y va de 0 a 500 libras.

Por otra parte datos generales que extaremos de este análisis preeliminar y que será interesante evaluar en base a la supervivencia son: * La edad de la mayoría de los pasajeros va de 18 a 40 años. Lo esperable es que sobrevivan más pasajeros entre estas edades, si solo se considerase esta variable. La media de edad es de aproximadamente 30 años * La mayoría de pasajeros viajaban solos * La mayoría han pagado tarifas entre 10 y 40 libras aproximadamente con una media de 33.

De este análisis inicial y también por la manera en que he creado la variable "Family.unit" quiero ver si hay correlación entre variables ya que si la correlación es exacta esto podría afectar al modelo en análisis

posteriores.

Es importante sobre todo de cara a las variables SibSp y Parch que son las que podríamos sospechar como altamente relacionadas (esto sin considerar Family.unit que es en la que más correlación espero, pero de esta última estoy evaluandola para substiruir SibSp y Parch)

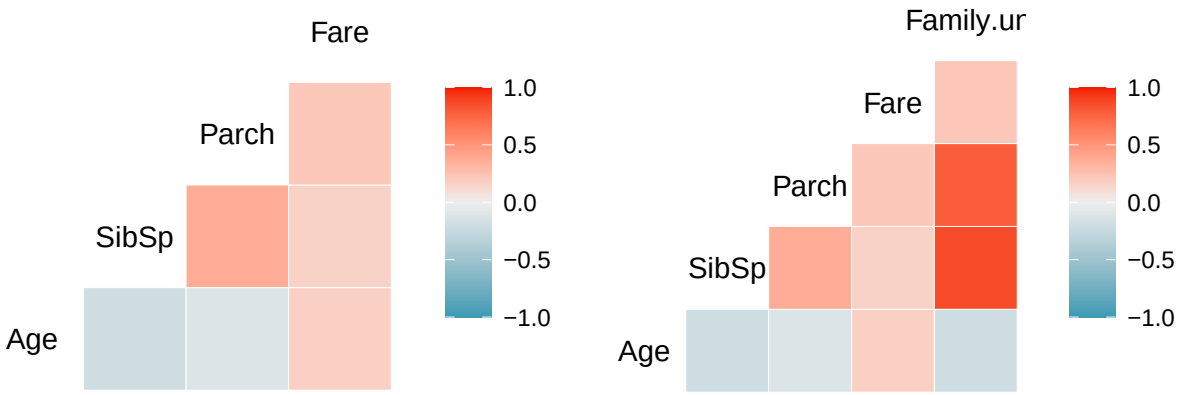
```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
require(gridExtra)
#matriz de correlaciones
cor(Titanic_complete[var.numeric])
```

```
##              Age      SibSp      Parch      Fare
## Age    1.0000000 -0.1915834 -0.1198805 0.1716041
## SibSp -0.1915834  1.0000000  0.3735872 0.1603494
## Parch -0.1198805  0.3735872  1.0000000 0.2216345
## Fare   0.1716041  0.1603494  0.2216345 1.0000000
```

```
#gráfico de correlaciones para todas las variables numéricas

plot5<-ggcorr(Titanic_complete[var.numeric])

#gráfico de correlaciones excluyendo nueva variable "Family.unit"
plot6<-ggcorr(Titanic_complete[var.numeric2])

grid.arrange(plot5,plot6, ncol=2)
```




```
#correlación entre Age y Fare
cor.test(Titanic_complete$Age,Titanic_complete$Fare, method="spearman")
```

```
## Warning in cor.test.default(Titanic_complete$Age, Titanic_complete$Fare, :
## Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  Titanic_complete$Age and Titanic_complete$Fare
## S = 316081305, p-value = 1.943e-08
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.1544654
```

```
#correlación entre SibSp y Parch
cor.test(Titanic_complete$SibSp,Titanic_complete$Parch, method="spearman")
```

```
## Warning in cor.test.default(Titanic_complete$SibSp, Titanic_complete$Parch, :
## Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
```

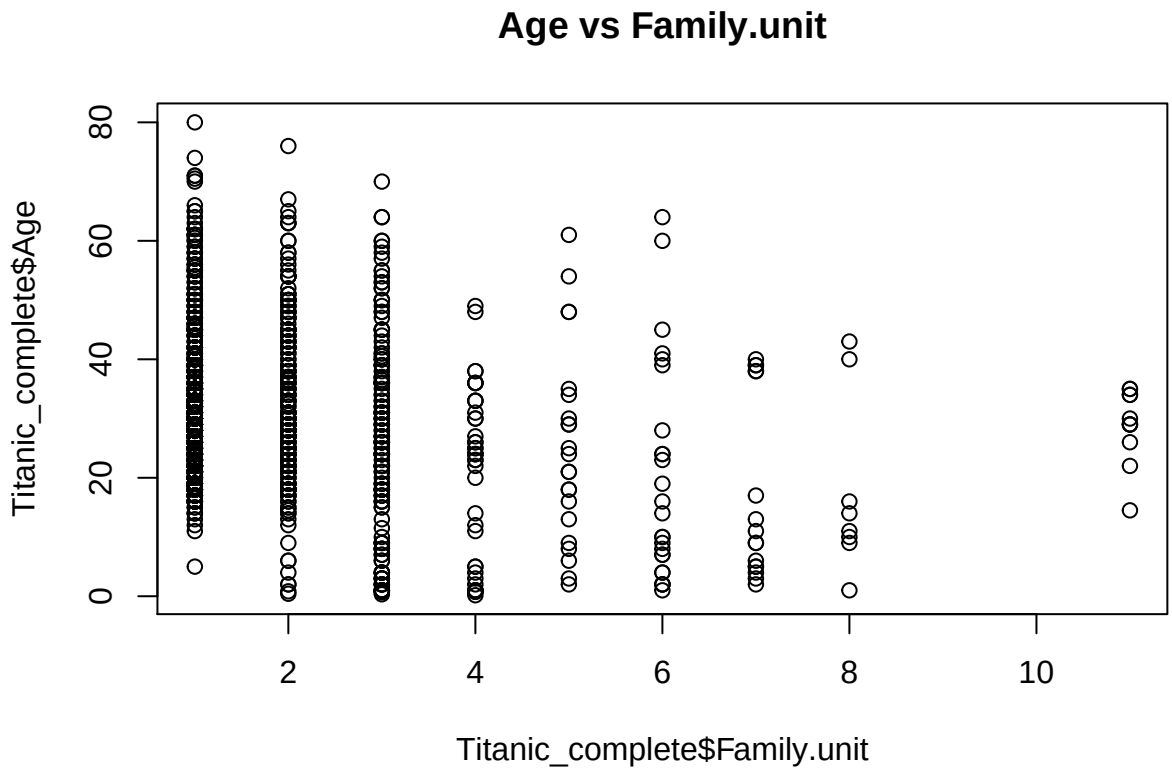
```
##
## data:  Titanic_complete$SibSp and Titanic_complete$Parch
## S = 209949776, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.438373
```

```
cor.test(Titanic_complete$SibSp,Titanic_complete$Parch, method="spearman")
```

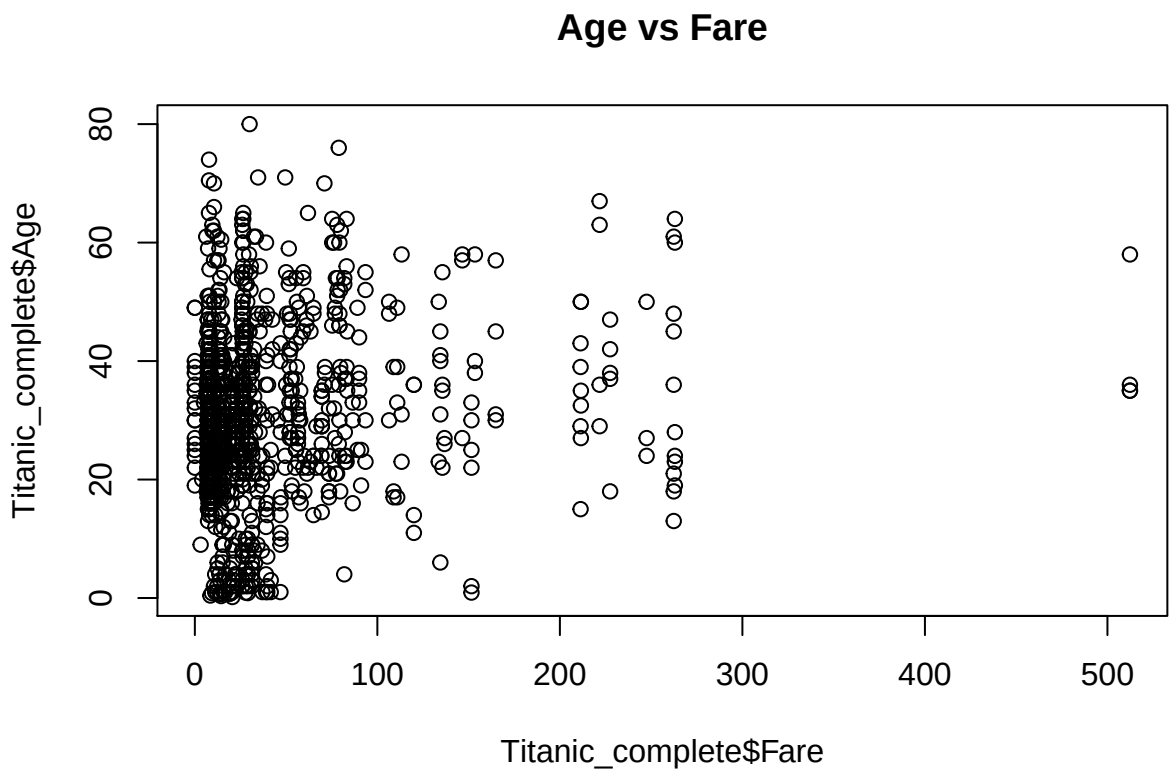
```
## Warning in cor.test.default(Titanic_complete$SibSp, Titanic_complete$Parch, :
## Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  Titanic_complete$SibSp and Titanic_complete$Parch
## S = 209949776, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.438373
```

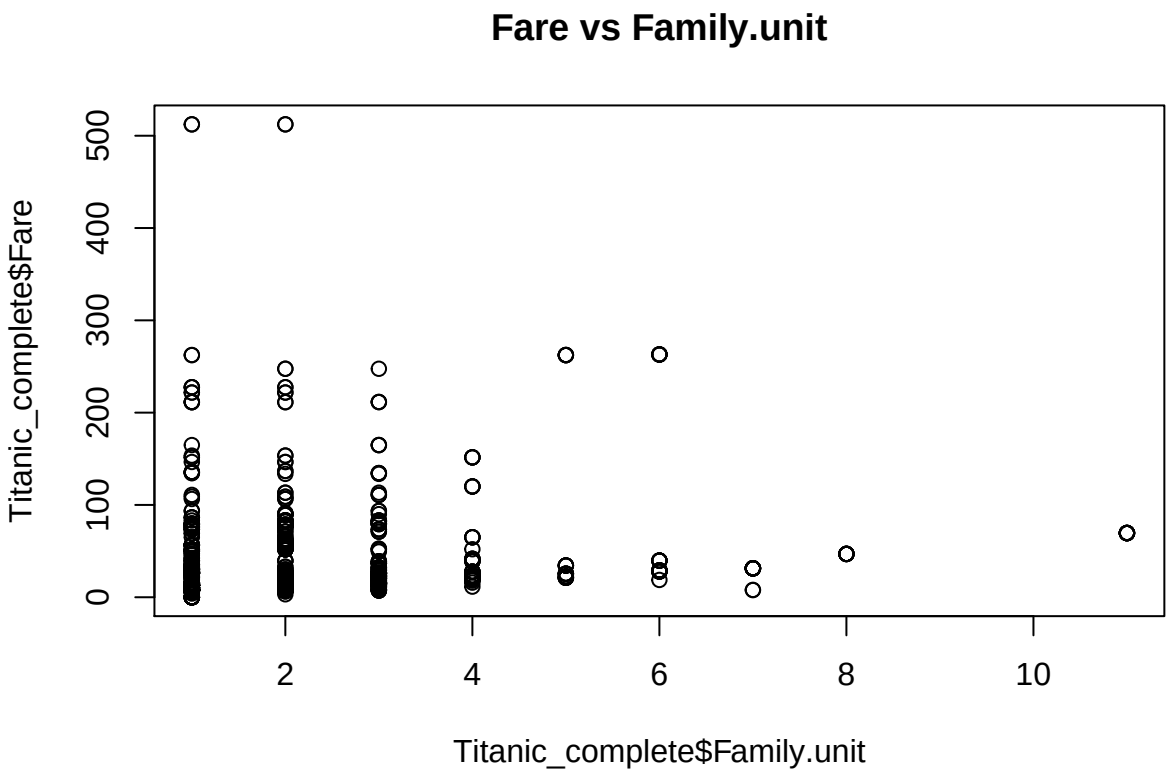
```
#representación de las relaciones entre variables que estamos considerando de cara al modelo

plot( Titanic_complete$Age ~ Titanic_complete$Family.unit, main="Age vs Family.unit" )
```



```
plot( Titanic_complete$Age ~ Titanic_complete$Fare, main="Age vs Fare" )
```




```
plot( Titanic_complete$Fare ~ Titanic_complete$Family.unit, main="Fare vs Family.unit" )
```



De los gráficos y tablas de correlación interpretamos que como era esperable la mayor relación está entre la variable Family.unit y las variables Parch y Sibsp, la variable Age no está relacionada con ninguna otra practicamente y la variable Fare está relacionada con todas las otras variables pero muy ligeramente.

Para asegurarme respecto a este punto he planteado un test de correlación entre las dos variables continuas Age y Fare, y detecto que la correlación es baja, de 0.15, por lo tanto no hay peligro de colinealidad en el uso de estas dos variables, en Parch y SibSp el valor es más alto por lo que opto por plantear un analisis de PCA para reducir la dimensionalidad.

El análisis de componentes principales permite, determinar nuevas variables numéricas que podría utilizar en lugar de las actuales y que expliquen la variabilidad, usualmente funciona mejor con variables continuas, así que no sería el método ideal, pero al contar con dos variables continuas, dos discretas y una (family.unit) que he creado que esta muy correlacionada con SibSp y Parch, pero que no sustituye del todo ambas variables he pensado que podía ser una opción viable, la otra alternativa que veo para la reducción de variables en este sentido es descartar las variables SibSp y Parch. De hacer eso me quedaría con 3 variables numéricas por lo que no lo descarto tampoco y dependerá del resultado que obtenga en el análisis de componentes principales que realizaré a continuación.

El análisis de componentes principales lo realizaré solo sobre las variables Parch,SibSp,Family unit, Age y Fare por lo que utilizaré el identificador de columnas [var.numeric]

```
#Utilizo el modelo de componentes principales PCA
pca<-prcomp(as.matrix(Titanic_complete[var.numeric]),scale=TRUE,center = TRUE)

#Con la función summary puedo ver la variabilidad acumulada con las variables
#y de ahí es claro determinar que con 4 PCA explicamos la variabilidad completa
#(recordemos que la variable Family Unit era una combinación de otras dos variables)
#y con 3 PCA  explicamos el 84% de la variabilidad
```

```r
summary(pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4
## Standard deviation     1.2429 1.0851 0.8122 0.7863
## Proportion of Variance 0.3862 0.2944 0.1649 0.1545
## Cumulative Proportion  0.3862 0.6805 0.8455 1.0000
```

```r
#Añado los 3 componentes principales al dataframe.
#Esto reduciría nuestras 5 variables numéricas a solo 3.

Titanic_complete<-cbind(Titanic_complete,pca$x[,1:3])
head(Titanic_complete)
```

```
##   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   Surname     Title
## 1                             Braund, Mr. Owen Harris    Braund non_elite
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer)   Cumings non_elite
## 3                              Heikkinen, Miss. Laina Heikkinen non_elite
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel)  Futrelle non_elite
## 5                            Allen, Mr. William Henry     Allen non_elite
## 6                                    Moran, Mr. James     Moran non_elite
##      Sex Age SibSp Parch           Ticket    Fare Cabin Embarked Set_type
## 1   male  22     1     0        A/5 21171  7.2500     N        S    train
## 2 female  38     1     0         PC 17599 71.2833   C85        C    train
## 3 female  26     0     0 STON/O2. 3101282  7.9250     N        S    train
## 4 female  35     1     0           113803 53.1000  C123        S    train
## 5   male  35     0     0           373450  8.0500     N        S    train
## 6   male  27     0     0           330877  8.4583     N        Q    train
##   CabinG Family.unit         PC1         PC2         PC3
## 1      N           2  0.01680051  0.85424885  0.06075622
## 2      C           2 -0.13687205 -0.86338264  0.13825309
## 3      N           1  0.69349594  0.49405263  0.13780883
## 4      C           2 -0.06420803 -0.46333097  0.04482026
## 5      N           1  0.86186180 -0.01889609 -0.27708508
## 6      N           1  0.70851600  0.43049989  0.09834417
```

```r
#De  cara a facilitar analisis posteriores y la interpretación de  resultados
#En caso de que opte por continuar el análisis con PCA,
#voy a revisar las relaciones que hay entre PCA y las variables de partida
#Para ello debo estudiar las posibles correlaciones entre PCA y las variables de Partida.
cor(Titanic_complete[var.numeric],pca$x[,1:3])
```

```
##              PC1         PC2        PC3
## Age    0.3064006 -0.80820743 -0.4927019
```

```
## SibSp -0.7849034  0.14060505 -0.2144980
## Parch -0.7914458 -0.04341845 -0.2876489
## Fare  -0.4565209 -0.70891234  0.5367872
```

Del analisis de componentes principales por tanto las conclusiones que podemos sacar y que posteriormente nos serán necesarias para interpretar los resultados son que:

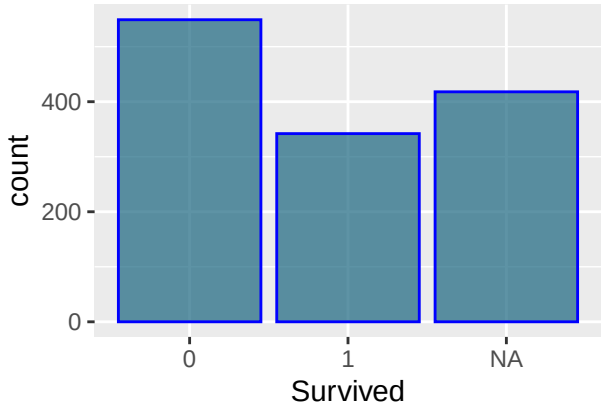
- PC1, está altamente relacionado negativamente con Family.unit>SibSp>Parch (cuanto más altos son estos valores menor será PC1), el factor edad se relaciona favorablemente aunque el impacto es mínimo.
- PC2 esta correlacionada con Age y Fare. Es decir, cuando aumentan Age y Fare también lo hace el PC2 los efectos de SibSp,Parch y Family.unit serían poco destacables en comparación
- En PC3 la relación más estrecha se da con los factores Age positivamente (Si Age aumenta también lo hace PC3) y con fare.d (pero una relación negativa), si Fare.d aumenta PC3 disminuye y a la inversa, SibSp y Family.unit también tendrían cierta relación positiva aunque en comparación a la de Fare y Age, no es relevante el impacto.

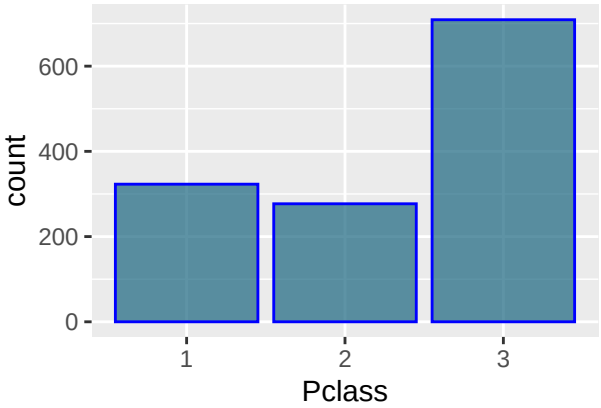
Conclusión

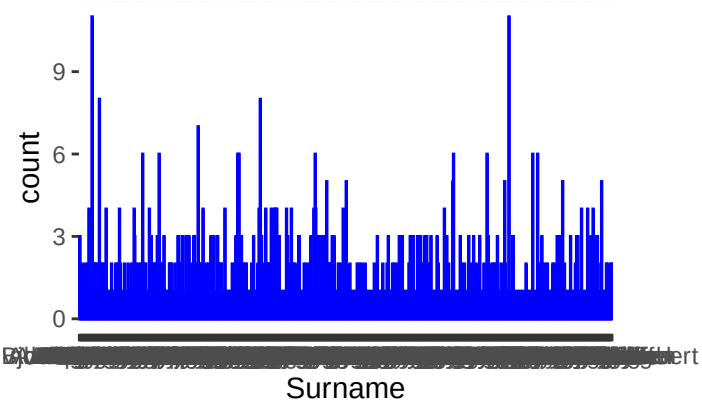
Voy a optar por descartar las variables Parch y SibSp, en su lugar utilizaré la variable Family.unit que combina las dos anteriores. A pesar de hacer el análisis de componentes principales creo que las variables no son tan numerosas en este caso y la variable creada sobre la unidad familiar recoge los datos de las otras dos variables por lo que, la perdidad de variabilidad debería ser mínima.Además con 3 componentes principales explico solo un 84% de la variabilidad y teniendo un número reducido de variables creo que no compensa el uso de este método.

Selección de variables: Variables categóricas Al igual que con las variables numéricas haré un estudio de las variables categóricas para revisar cuales participarán en el modelo.

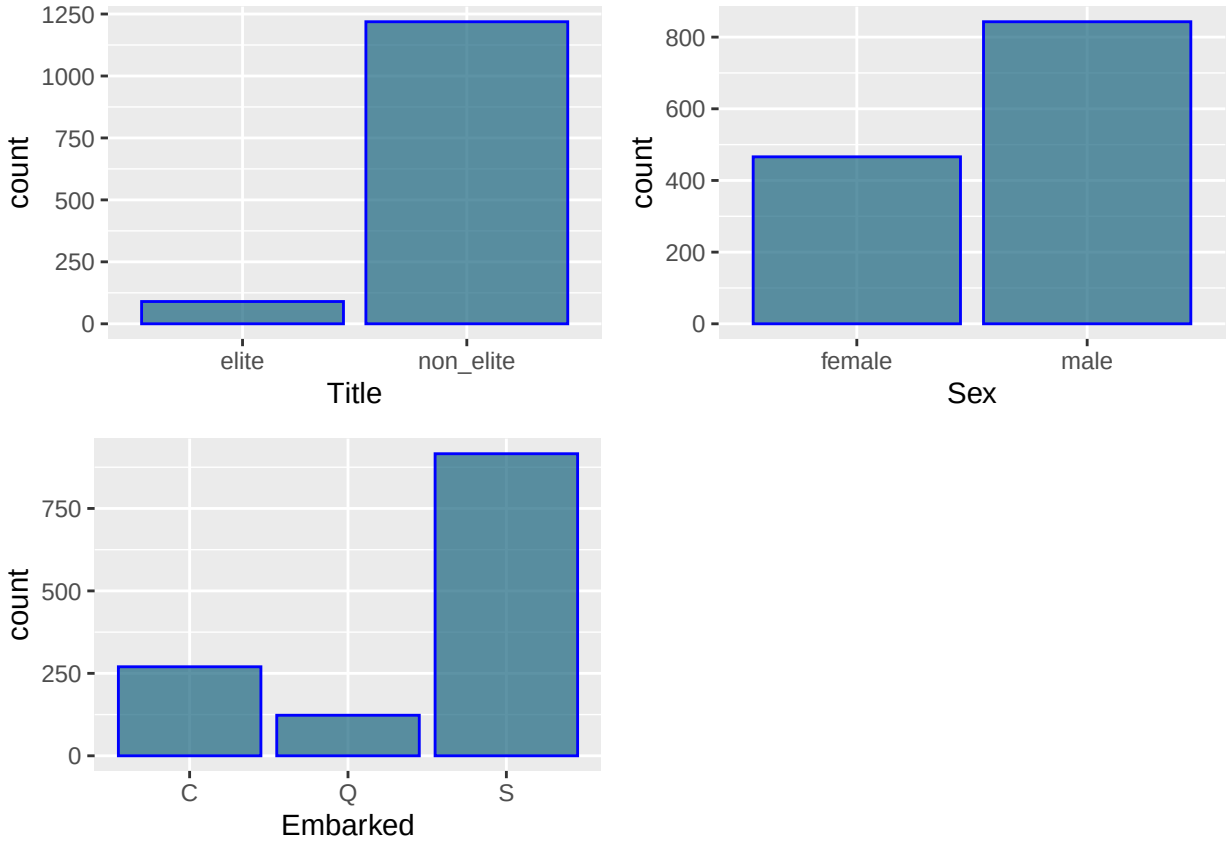
```
#Barplot per variables categóricas
plot7<-ggplot(Titanic_complete,
              aes(x=Survived))+geom_bar(color="blue",
                                        fill=rgb(0.1,0.4,0.5,0.7))
plot8<-ggplot(Titanic_complete,
              aes(x=Pclass))+geom_bar(color="blue",
                                      fill=rgb(0.1,0.4,0.5,0.7))
plot9<-ggplot(Titanic_complete,
              aes(x=Surname))+geom_bar(color="blue",
                                       fill=rgb(0.1,0.4,0.5,0.7))
grid.arrange(plot7,plot8,plot9, ncol=2)
```



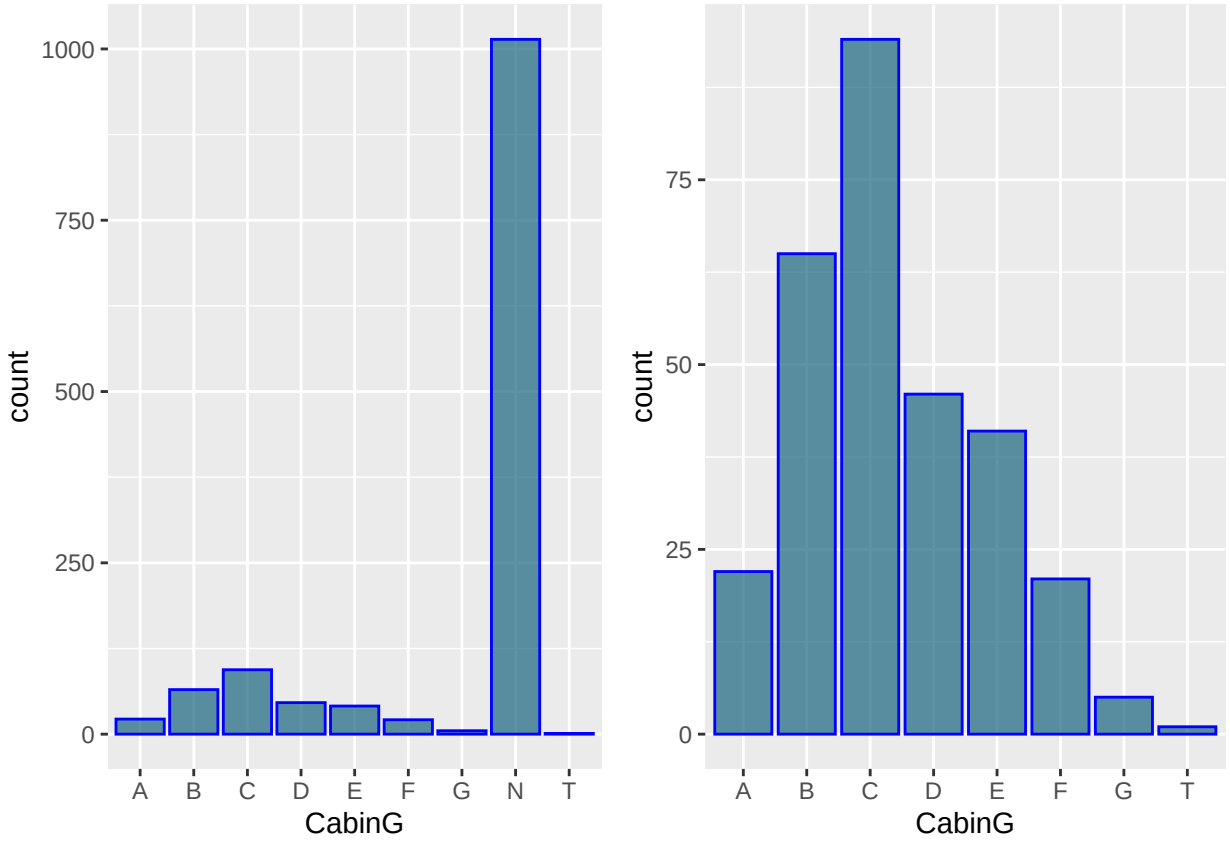





```
plot10<-ggplot(Titanic_complete,
               aes(x=Title))+geom_bar(color="blue",
                                      fill=rgb(0.1,0.4,0.5,0.7))
plot11<-ggplot(Titanic_complete,
               aes(x=Sex))+geom_bar(color="blue",
                                    fill=rgb(0.1,0.4,0.5,0.7))
plot12<-ggplot(Titanic_complete,
               aes(x=Embarked))+geom_bar(color="blue",
                                         fill=rgb(0.1,0.4,0.5,0.7))

grid.arrange(plot10,plot11,plot12, ncol=2)
```




```
plot13<-ggplot(Titanic_complete,
               aes(x=CabinG))+geom_bar(color="blue",
                                       fill=rgb(0.1,0.4,0.5,0.7))
plot14<-ggplot(Titanic_complete[Titanic_complete$CabinG!="N",],
               aes(x=CabinG))+geom_bar(color="blue",
                                       fill=rgb(0.1,0.4,0.5,0.7))
grid.arrange(plot13,plot14, ncol=2)
```



Del estudio individual de las variables categóricas lo que interpretamos es lo siguiente

- De los 900 aprox pasajeros de los que tenemos datos solo 300aprox sobrevivieron

*viajaron más hombres que mujeres

- Los pasajeros de tercera clase consituyen una amplia mayoría se esperaría que si solo se considera esta variable la mayoría de los supervivientes fueran de tercera clase.
- La variable surname por si sola no nos aporta demasiado valor, si bien es cierto que algunos apellidos se agrupan la información combinada que nos aporta es bastante similar a la que nos podría dar la nueva variable Family,unit.
- Los títulos nobiliarios y personas pertenecientes a la élite constituyen una minoria.
- La mayoría de pasajeros embarcan en South hampton que es también el punto de partida del barco.

*En cuanto a la cubierta de las cabinas he planteado dos graficos en el segundo sin considerar los valores que desconocemos y en el primero tendríamos claramente que disponemos más datos de los pasajeros de las cabinas de la cubierta C

Conclusión: De entre las variables categóricas las que descartaríamos en este punto serían "surname" y Cabin, esta última variable me resulta interesante pero considero que necesita procesamiento adicional y busqueda de información adicional para crear una variable compuesta. Por la limitación de tiempo, no entraré en mayor profundidad en cuanto al estudio de esa variable.

Análisis exploratorio: relación entre variable supervivencia y otras variables En este punto ya hemos determinado que la variable supervivencia es la variable dependiente que quiero predecir, de cara a hacer la última selección de varibles es interesante ver la posible relación que pueda haber entre esta variables y las variables categóricas y numéricas

```
#comparaciónes en formato tabla entre supervivencia y variables categóricas.
table(Titanic_complete$Survived,Titanic_complete$Pclass)
```

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

```
table(Titanic_complete$Survived,Titanic_complete$Title)
```

```
##
##     elite non_elite
##   0    32       517
##   1    31       311
```

```
table(Titanic_complete$Survived,Titanic_complete$Sex)
```

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

```
table(Titanic_complete$Survived,Titanic_complete$Embarked)
```

```
##
##           C   Q   S
##   0   0  75  47 427
##   1   0  93  30 219
```

```
table(Titanic_complete$Survived,Titanic_complete$CabinG)
```

```
##
##       A   B   C   D   E   F   G   N   T
##   0   8  12  24   8   8   5   2 481   1
##   1   7  35  35  25  24   8   2 206   0
```

De la comparación entre la supervivencia y las variables categóricas, lo que veo reflejado es lo siguiente:

- Segun la clase parece haber más supervivencia para los pasajeros de primera clase
- Sobre los títulos o cargos de Elite, sí parece haber más supervivencia de estas clases sociales aunque la idea principal que sacamos de esto es que más mujeres sobrevivieron
- El punto anterior sobre la supervivencia de más mujeres en relación a los hombres se compara en la tabla de contraste entre Sexo y supervivencia
- En cuanto al puerto de embarque sobreviven más pasajeros que embarcan en el puerto S aunque si tenemos en cuenta la supervivencia de los pasajeros que embarcan en cada puerto vemos claramente que entre los pasajeros que embarcaron por el puerto C la supervivencia es algo mayor al 50%,

- Por último sobre las cabinas vemos que la mayoría de los supervivientes tenían las cabinas en las cubiertas B y C. Pero es más destacable el hecho de los supervivientes de las cabinas D E en las cuales solo murieron 8 pasajeros respectivamente (de los que contamos con información)
- En cuanto a nuestro pasajero James Kelly 892, tiene muchos de los factores que a priori son desfavorables para su supervivencia (tercera clase, sin títulos, sexo masculino)

Hacemos un análisis similar para las variables numéricas comparandolas con la variable "Survived".

```
aggregate(Titanic_complete[var.numeric2],list(Titanic_complete$Survived), mean)
```

```
##   Group.1      Age     SibSp     Parch     Fare Family.unit
## 1       0 30.32332 0.5537341 0.3296903 22.11789    1.883424
## 2       1 28.63939 0.4736842 0.4649123 48.39541    1.938596
```

```
head(Titanic_complete)
```

```
##   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   Surname     Title
## 1                             Braund, Mr. Owen Harris    Braund non_elite
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer)   Cumings non_elite
## 3                              Heikkinen, Miss. Laina Heikkinen non_elite
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel)  Futrelle non_elite
## 5                            Allen, Mr. William Henry     Allen non_elite
## 6                                    Moran, Mr. James     Moran non_elite
##      Sex Age SibSp Parch           Ticket    Fare Cabin Embarked Set_type
## 1   male  22     1     0        A/5 21171  7.2500     N        S    train
## 2 female  38     1     0         PC 17599 71.2833   C85        C    train
## 3 female  26     0     0 STON/O2. 3101282  7.9250     N        S    train
## 4 female  35     1     0           113803 53.1000  C123        S    train
## 5   male  35     0     0           373450  8.0500     N        S    train
## 6   male  27     0     0           330877  8.4583     N        Q    train
##   CabinG Family.unit         PC1         PC2         PC3
## 1      N           2  0.01680051  0.85424885  0.06075622
## 2      C           2 -0.13687205 -0.86338264  0.13825309
## 3      N           1  0.69349594  0.49405263  0.13780883
## 4      C           2 -0.06420803 -0.46333097  0.04482026
## 5      N           1  0.86186180 -0.01889609 -0.27708508
## 6      N           1  0.70851600  0.43049989  0.09834417
```

Y lo más destacable es que este análisis preeliminar ya nos da una idea de que los pasajeros que pagaron tickets más caros tal vez tuvieron más probabilidades de supervivencia, también se puede intuir que factores como la edad (más jovenes) eran más favorables de cara a la supervivencia y también el sexo del pasajero aunque no son valores significativos.

Conclusión:

Las variables con las que plantearé el modelo serán: Survived, como variable dependiente a predecir. Pclass,Title,Sex,Age,Fare y Family.unit

El modelo que aplicaré para la predicción es un modelo de regresión logística ya que intento predecir una variable dicotómica, el pasajero sobrevive (sí o no), por lo que aunque a grandes rasgos la limpieza de datos ya está realizada también tendré que realizar otras tareas de limpieza como será escalar y centrar los datos y binarizar las variables categóricas.

4.2 Comprobación de la normalidad y homogeneidad de la variancia.

Para comprobar la normalidad utilizaré el test de Shapiro-Wilk.

La hipotesis nula será que la població està distribuïda normalmente y la alternativa que no lo está. Utilizando un nivel de significación de 0,05 si el pvalor es menor que el nivel de significación se rechazaría la hipotesis nula.

```
#Todos los test que planteo me dan el mismo resultado que las muestras no siguen una normal, a modo de

shapiro.test(Titanic_complete$Age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Titanic_complete$Age
## W = 0.97478, p-value = 2.359e-14
```

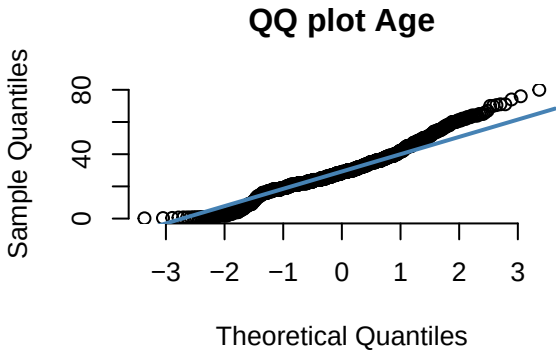
```
#shapiro.test(Titanic_complete$Family.unit)
#shapiro.test(Titanic_complete$Fare)
```

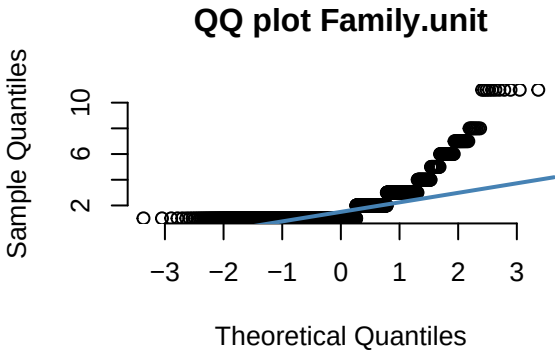
En todos los casos se esta rechazando la hipotesis de normalidad, pero hay que tener en cuenta que el test de Shapiro-Wilk a pesar de ser un método muy robusto.

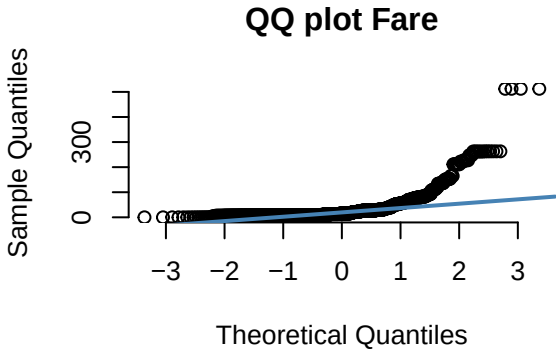
Para muestras muy grandes no resulta efectivo muchas veces por lo que la normalidad se debe evaluar de otra maneras. En cuanto a la variable Age por el histograma sí podemos identificar que se aproxima a la forma que tendría una normal, mientras que las demás variables no se aproximamn a la normal, por lo que con esa comprobación podríamos decir que las muestras no estan normalmente distribuidas para las otras variables.

Una última comprobación que podemos hacer para determinar la normalidad es un qqplot

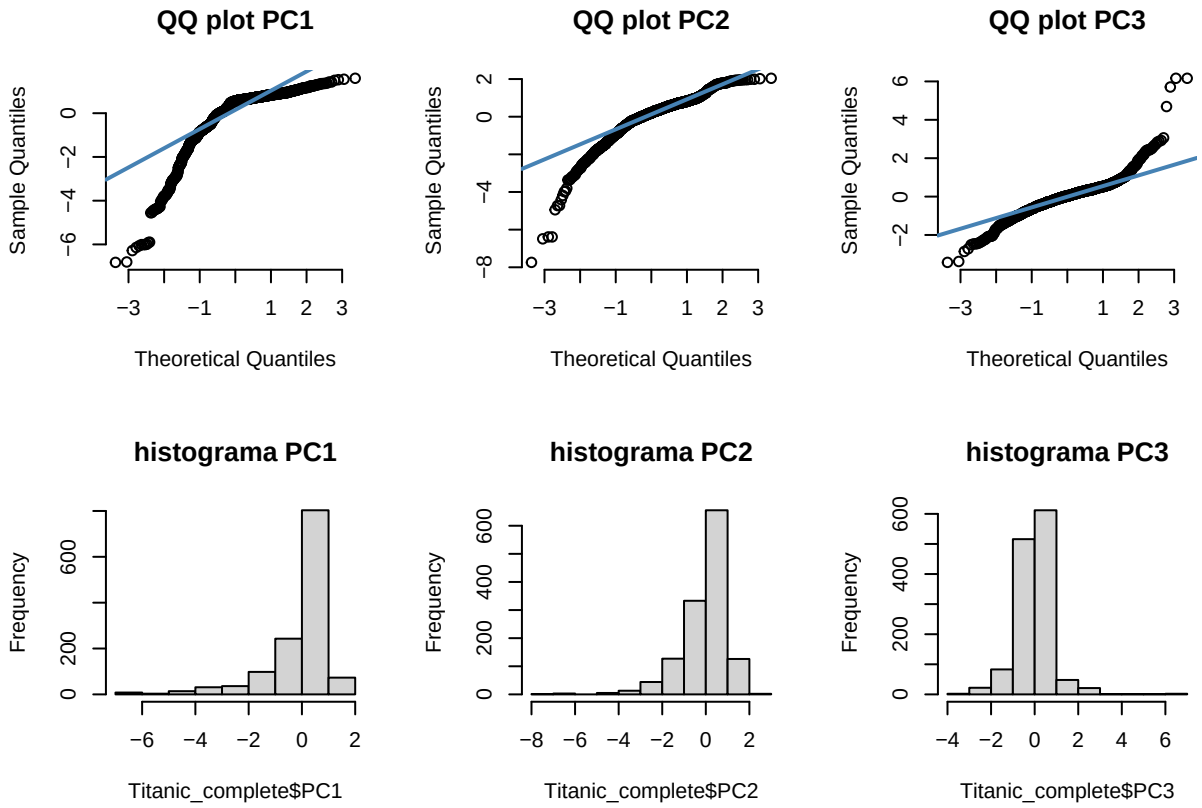
```
par(mfrow=c(2,2))
qqnorm(Titanic_complete$Age, pch = 1, frame = FALSE, main="QQ plot Age")
qqline(Titanic_complete$Age, col = "steelblue", lwd = 2)

qqnorm(Titanic_complete$Family.unit, pch = 1, frame = FALSE, main="QQ plot Family.unit")
qqline(Titanic_complete$Family.unit, col = "steelblue", lwd = 2)

qqnorm(Titanic_complete$Fare, pch = 1, frame = FALSE, main="QQ plot Fare")
qqline(Titanic_complete$Fare, col = "steelblue", lwd = 2)



#Aunque finalmente he descartado continuar el modelo con variables PCA
#He hecho pruebas de normalidad para PCA
#(estas no serían necesarias pero he hecho algunas comprobaciones extras
#antes de descartar la opción de usar PCA).

par(mfrow=c(2,3))
```








```
qqnorm(Titanic_complete$PC1, pch = 1, frame = FALSE,main="QQ plot PC1")
qqline(Titanic_complete$PC1, col = "steelblue", lwd = 2)
qqnorm(Titanic_complete$PC2, pch = 1, frame = FALSE,main="QQ plot PC2")
qqline(Titanic_complete$PC2, col = "steelblue", lwd = 2)
qqnorm(Titanic_complete$PC3, pch = 1, frame = FALSE,main="QQ plot PC3")
qqline(Titanic_complete$PC3, col = "steelblue", lwd = 2)
hist(Titanic_complete$PC1, main="histograma PC1")
hist(Titanic_complete$PC2, main="histograma PC2")
hist(Titanic_complete$PC3, main="histograma PC3")
```




```
#En este caso se ve que el tercer componente principal
#e incluso podríamos decir que con el segundo hay aproximaciones a la normal
#Esto es posible que se de porque ambos componentes estan muy
#correlacionados con la edad que sí sigue una normal
```

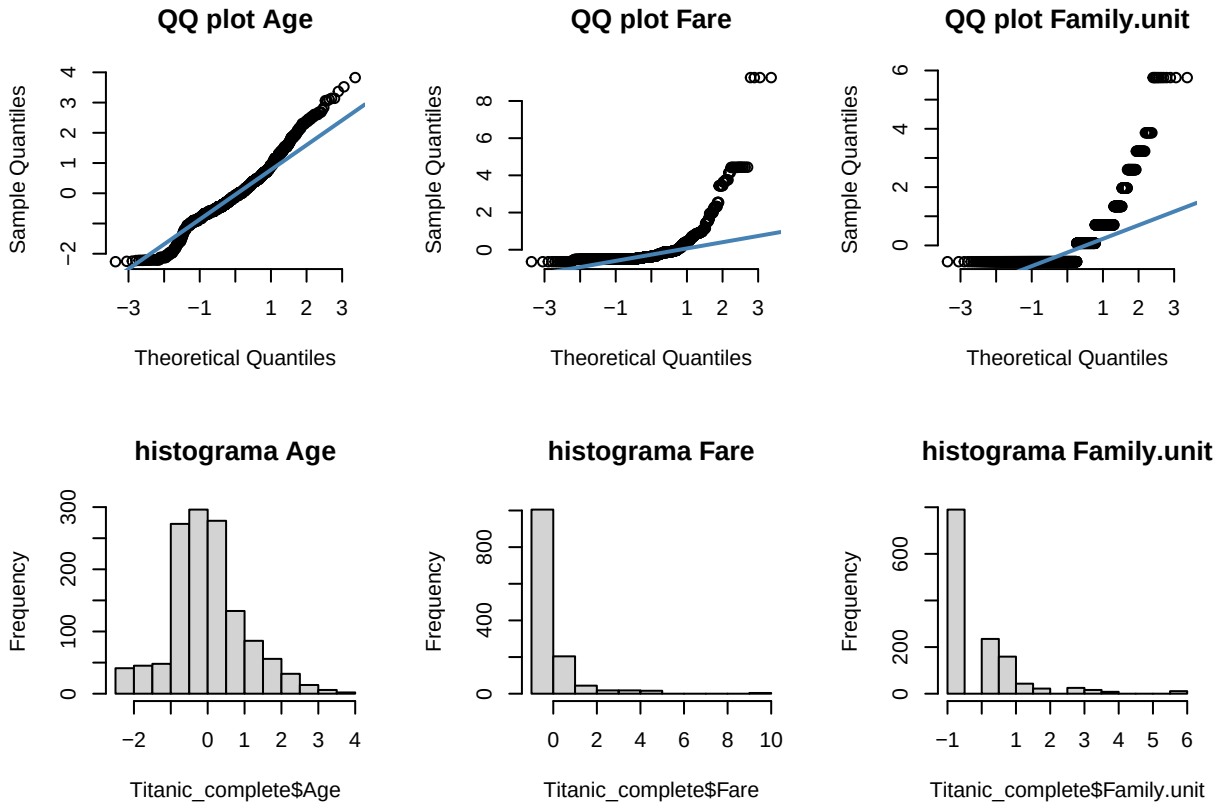
De estas representaciones gráficas deducimos lo mismo que hemos detectado antes con los histogramas, la variable edad tiene una distribución que se aproxima bastante a la normal.

Por lo que se refiere a las demás variables numéricas, por las represenntaciones gráficas no vemos tampoco una aproximación clara a la normal, pero al disponer de un número tan grande de registros o de muestra por el teorema del límite central podríamos aproximar todas las muestras a la distribución normal y tratarlas como si estuvieran normalmente distribuidas.

El cambio que si será necesario es normalizar las variables y centrarlas ya que hay modelos sensibles a las diferencias de variancias y de escalas, de forma que, si no se igualan de alguna forma los predictores, aquellos que se midan en una escala mayor o que tengan más varianza, dominarán el modelo aunque no sean los que más relación tienen con la variable respuesta.Por lo tanto conviene normalizar las variables y centrarlas, en este caso lo haré utilizando las funciones scale y center de R.

```
#Antes de escalar y centrar los valores quiero mantener los datos
#originales en un dataframe accesorio
#para poder tenerlos disponibles de cara a la creación de visualizaciones
#del apartado 5, por lo que genero un dataframe accesorio al que llamaré #Titanic_complete_raw

Titanic_complete_raw<- Titanic_complete
```

```r
#escalo y centro las variables
Titanic_complete$Age<-scale(Titanic_complete$Age,center = TRUE)
Titanic_complete$Fare<-scale(Titanic_complete$Fare,center = TRUE)
Titanic_complete$Family.unit<-scale(Titanic_complete$Family.unit,
                                    center = TRUE)

par(mfrow=c(2,3))
qqnorm(Titanic_complete$Age, pch = 1, frame = FALSE,main="QQ plot Age")
qqline(Titanic_complete$Age, col = "steelblue", lwd = 2)
qqnorm(Titanic_complete$Fare, pch = 1, frame = FALSE,main="QQ plot Fare")
qqline(Titanic_complete$Fare, col = "steelblue", lwd = 2)
qqnorm(Titanic_complete$Family.unit, pch = 1, frame = FALSE,main="QQ plot Family.unit")
qqline(Titanic_complete$Family.unit, col = "steelblue", lwd = 2)
hist(Titanic_complete$Age, main="histograma Age")
hist(Titanic_complete$Fare, main="histograma Fare")
hist(Titanic_complete$Family.unit, main="histograma Family.unit")
```




```r
library(car)
```

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

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:purrr':
##
##     some

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

```
fligner.test(Age ~ Set_type, data = Titanic_complete)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Age by Set_type
## Fligner-Killeen:med chi-squared = 0.25654, df = 1, p-value = 0.6125
```

```
fligner.test(Fare ~ Set_type, data = Titanic_complete)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Fare by Set_type
## Fligner-Killeen:med chi-squared = 0.10421, df = 1, p-value = 0.7468
```

```
fligner.test(Family.unit ~ Set_type, data = Titanic_complete)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Family.unit by Set_type
## Fligner-Killeen:med chi-squared = 0.22387, df = 1, p-value = 0.6361
```

Hecho este proceso tendremos las variables normalizadas y en cuanto a la homogeneidad de varianzas vemos que para los grupos de entrenamiento y test si se cumple.

Limpieza de datos últimos pasos

Dado que como he indicado utilizaré un modelo de regresión logística, es necesario binarizar las variables categóricas. Pero antes de realizar este paso quiero eliminar de los datasets las variables que no continuaran en el modelo.

```
#Selección de variables predictoras a incluir en el modelo
Titanic_complete<-Titanic_complete[, c(1,2,3,6,7,8,12,15,17)]
#guardo el dataset con variables seleccionadas y limpieza
write.csv(Titanic_complete,paste("C:/Users/mila_/Documents/Master ciencia de dades",
  "/Tipología y ciclo de vida de los datos/PRAC 2/titanic_complete.csv",sep=""))

#compruebo los datos
head(Titanic_complete)
```

```
##   PassengerId Survived Pclass     Title    Sex        Age       Fare Set_type
## 1           1        0      3 non_elite   male -0.5980158 -0.5030988    train
## 2           2        1      1 non_elite female  0.6225304  0.7344629    train
## 3           3        1      3 non_elite female -0.2928792 -0.4900532    train
## 4           4        1      1 non_elite female  0.3936780  0.3830371    train
## 5           5        0      3 non_elite   male  0.3936780 -0.4876373    train
## 6           6        0      3 non_elite   male -0.2165951 -0.4797462    train
##   Family.unit
## 1  0.07332427
## 2  0.07332427
## 3 -0.55813274
## 4  0.07332427
## 5 -0.55813274
## 6 -0.55813274
```

Por otra parte en este punto he decidido separar de nuevo los datos en test y entrenamiento, aunque tendré más adelante que binarizar las variables categoricas y tendré que hacer esa transformación sobre todos los grupos (test y train)

```
#Separación de dataset Train para analisis y de test para la prueba del modelo

Train.f<-Titanic_complete%>%
  select(PassengerId,Survived,
         Pclass,Title,Sex,Age,Fare,Set_type,Family.unit)%>%
  filter(Set_type=="train")

Test.f<-Titanic_complete%>%
  select(PassengerId,Survived,Pclass,Title,
         Sex,Age,Fare,Set_type,Family.unit)%>%
  filter(Set_type=="test")

str(Train.f)
```

```
## 'data.frame':    891 obs. of  9 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 ...
##  $ Title      : Factor w/ 2 levels "elite","non_elite": 2 2 2 2 2 2 2 1 2 2 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num [1:891, 1] -0.598 0.623 -0.293 0.394 0.394 ...
##   ..- attr(*, "scaled:center")= num 29.8
##   ..- attr(*, "scaled:scale")= num 13.1
##  $ Fare       : num [1:891, 1] -0.503 0.734 -0.49 0.383 -0.488 ...
##   ..- attr(*, "scaled:center")= num 33.3
##   ..- attr(*, "scaled:scale")= num 51.7
##  $ Set_type   : Factor w/ 2 levels "test","train": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Family.unit: num [1:891, 1] 0.0733 0.0733 -0.5581 0.0733 -0.5581 ...
##   ..- attr(*, "scaled:center")= num 1.88
##   ..- attr(*, "scaled:scale")= num 1.58
```

Una vez separados los grupos continuamos con el siguiente apartado relativo al análisis con test stadísticos

4.3 Aplicación de pruebas estadísticas para comparar los grupos de datos. En función de los datos i del objetivo del estudio, aplicar pruebas de contraste de hipotesis, correlaciones, regressiones, etc. Aplicar al menos tres metodos de análisis diferentes

Correlación:

He aplicado pruebas de correlación para las variables cuantitativas y a raiz de eso un análisis de componentes principales que se puede ver en el punto 4.1

Contraste de Hipotesis: contraste de proporciones

De las cuestiones planteadas en el planteamiento del análisis voy a hacer un contraste de hipotesis sobre proporciones, la pregunta que pretendo responder con esto es algo que llama mi atención inicialmente quería plantear si la probabilidad de supervivencia de los nobles era más alta que la de personas sin títulos nobiliarios, pero la muestra de personas con título es demasiado pequeña así que en lugar de esa cuestión la que intentaré responder con una contraste de hipotesis sobre las proporciones es si ¿ pertenecer a una élite da más posibilidades de supervivencia en comparación a si no se pertenece a una élite

```
#hago una comprobación rápida de los supervivientes
#según el título que tengan
table(Train.f$Title,Train.f$Survived)
```

```
##
##                0   1
##   elite       32  31
##   non_elite 517 311
```

como el tamaño de la muestra es superior a 30 registros podemos asumir normalidad por lo que podemos continuar con el planteamiento de la hipótesis:

```
#Calculo de supervivientes nobles  y tamaño n muestra nobles
x.p.sup.elite<-length(Train.f[Train.f$Title=="elite"&
                                Train.f$Survived==1,"Title"])
n.p.sup.elite<-length(Train.f[Train.f$Title=="elite","Title"])

#Calculo de supervivientes no nobles y tamaño muestra "no nobles"
x.p.sup.n.elite<-length(Train.f[Train.f$Title=="non_elite"&
                                  Train.f$Survived==1,"Title"])
n.p.sup.n.elite<-length(Train.f[Train.f$Title=="non_elite","Title"])
```

Planteo las Hipotesis:

H0: P.sup.elite = p.sup.n.elite H1: P.Sup.elite > p.sup.n.elite

```
prop.test(x=c(x.p.sup.elite,x.p.sup.n.elite), n=c(n.p.sup.elite,n.p.sup.n.elite), p = NULL,
          alternative = "greater",
          conf.level = 0.95)
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
```

```
## data:  c(x.p.sup.elite, x.p.sup.n.elite) out of c(n.p.sup.elite, n.p.sup.n.elite)
## X-squared = 2.883, df = 1, p-value = 0.04476
## alternative hypothesis: greater
## 95 percent confidence interval:
##  0.0006816397 1.0000000000
## sample estimates:
##    prop 1    prop 2
## 0.4920635 0.3756039
```

El p-valor es menor que el nivel de significación por lo tanto se acepta la hipotesis alternativa.

En efecto miembros de la aristocracia o con títulos nobiliarios tenían más probabilidades de supervivencia en el titanic, aunque la diferencia no es tan pronunciada como podríamos pensar, es posible que eso se explique en parte por que la mayoría de la aristocrácia que se tiene en cuenta en esta comparación son hombres y el acceso a los botes salvavidas se produjo en base a "mujeres y niños primero".

A raiz de este análisis se me ha ocurrido que para evaluar mejor la idoniedad de las variables cualitativas que usaré en el modelo de regresión lineal una mejor opción habria sido hacer contrastes de propociones en relación a la proporción de supervivencia global. Es decir, si tengo una proporción de supervivenvia de 38,38%. Las variables cualitativas en las que puedo ver una supervivencia mayor a esta es muy posible que aporten valor al modelo y que si no mejoran esta proporción en cambio no aporten al modelo. Es una posibilidad que me parece interesannte de explorar y ciertamente más precisa que la que he usado, pero dada la limitación de tiempo optaré por mantener las variables que ya elegí en su momento. Aún así hago un cálculo rápido de las proporciones muestrales para comprobar esta "teoría" y en efecto es posible que sea interesante investigar esa vía.

```
prop.table(table(Train.f$Survived))
```

```
##
##         0         1
## 0.6161616 0.3838384
```

```
prop.table(table(Train.f[Train.f$Sex=="female","Sex"],
                 Train.f[Train.f$Sex=="female","Survived"]))
```

```
##
##                   0         1
##   female 0.2579618 0.7420382
##   male   0.0000000 0.0000000
```

Regressión logística

La variable que intentamos predecir con el modelo es de tipo categórica y dicotómica (sobrevive o no sobrevive), por lo que en este caso lo más acertado es usar un modelo de regresión logística (logit).

Para poder aplicar este modelo de regresión tenemos que binarizar las variables cualitativas como llevamos comentando en apartados anteriores. Pero además en este punto tenemos que separar los datos de train.f en grupos otra vez de entrenamiento y test.

```
library(lattice)
library(caret)
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
#defino la proporción en que quiero dividir la muestra de entrenamiento y la muestra test.
size_sample <- floor(0.80 * nrow(Train.f))

#plantamos la semilla para que siempre se genere la misma división aleatoria de los registros en pasos
set.seed(123)

#separación de los datos en entrenamiento y test

train.selection <- sample(seq_len(nrow(Train.f)), size = size_sample)

train.sample <- Train.f[train.selection, ]
test.sample <- Train.f[-train.selection, ]

#compruebo que la proporción de supervivientes es similar respecto a la de la muestra completa inicial

prop.table(table(Train.f$Survived))
```

```
##
##         0         1
## 0.6161616 0.3838384
```

```
prop.table(table(train.sample$Survived))
```

```
##
##        0        1
## 0.616573 0.383427
```

```
prop.table(table(test.sample$Survived))
```

```
##
##         0         1
## 0.6145251 0.3854749
```

```
#La variación es mínima e interpreto que la separación de datos se ha hecho de manera correcta.
```

```
#Binarización de los datos cualitativos.
library(recipes)
```

```
##
## Attaching package: 'recipes'

## The following object is masked from 'package:stringr':
##
##     fixed

## The following object is masked from 'package:stats':
##
##     step
```

```
receta<- recipe( Survived ~ Pclass + Title + Sex + Age + Fare + Family.unit,
                        data =  train.sample)


receta <- receta %>% step_dummy(all_nominal(), -all_outcomes())

#Se entrena el objeto receta

receta_trained<- prep(receta, training = train.sample)
receta_trained
```

```
## Data Recipe
##
## Inputs:
##
##       role #variables
##    outcome          1
##  predictor          6
##
## Training data contained 712 data points and no missing data.
##
## Operations:
##
## Dummy variables from Pclass, Title, Sex [trained]
```

```
#se aplican las transformaciones a train y test de registros conocidos

train.sample.ok <- bake(receta_trained, new_data = train.sample)
test.sample.ok  <- bake(receta_trained, new_data = test.sample)



#Reviso como queda el set de entrenamiento cuando ya he binarizado las variables cualitativas
glimpse(train.sample.ok)
```

```
## Rows: 712
## Columns: 8
## $ Age             <dbl[,1]> <matrix[26 x 1]>
## $ Fare            <dbl[,1]> <matrix[26 x 1]>
## $ Family.unit     <dbl[,1]> <matrix[26 x 1]>
## $ Survived        <fct> 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, ...
## $ Pclass_X2       <dbl> 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ Pclass_X3       <dbl> 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, ...
## $ Title_non_elite <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ Sex_male        <dbl> 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, ...
```

En este punto ya se puede aplicar el modelo de regresión logística pero dado que tenemos que crear un modelo predictivo y no solo de regresión logística tengo que pasar por dos pasos más uno de entrenamiento del modelo y la predicción(que se hará aplicando el modelo de regresión logística.)

Para esto he planteado 3 modelos la diferencia entre ellos son los predictores que se utilizan, para el primero y el segundo, he utilizado todas las variables que había seleccionado y las variables categóricas binarizadas,

pero he expresado la fórmula de manera distinta mi intención era ver si el resultado era el mismo y así ha sido.

Y para el tercero las mismas variables que en el segundo pero eliminando la variable Fare

```
#planteamiento de modelos de regresión logística múltiple
set.seed(123)
modelo_reg.logistic <- train(Survived ~ ., data = train.sample.ok,
                         method = "glm",
                         metric = "Accuracy",
                         family = "binomial")

equation_2<-"Survived ~ Pclass_X2 + Pclass_X3 + Title_non_elite + Sex_male + Age + Fare + Family.unit"
formula_2<- as.formula(equation_2)
set.seed(123)
modelo_reg.logistic_2 <- train(formula_2, data = train.sample.ok,
                         method = "glm",
                         metric = "Accuracy",
                         family = "binomial")

equation_3<-"Survived ~ Pclass_X2 + Pclass_X3 + Title_non_elite + Sex_male + Age + Family.unit"
formula_3<- as.formula(equation_3)
set.seed(123)
modelo_reg.logistic_3 <- train(formula_3, data = train.sample.ok,
                         method = "glm",
                         metric = "Accuracy",
                         family = "binomial")

#Resumen de los resultados de los 3 modelos.

modelo_reg.logistic
```

```
## Generalized Linear Model
##
## 712 samples
##   7 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 712, 712, 712, 712, 712, 712, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.8065235  0.5852163
```

```
summary(modelo_reg.logistic$finalModel)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -2.4688  -0.5908  -0.4062   0.6130   2.4397
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)       4.2896     0.4915   8.728  < 2e-16 ***
## Age              -0.3740     0.1133  -3.301 0.000963 ***
## Fare              0.2269     0.1397   1.624 0.104464
## Family.unit      -0.5627     0.1275  -4.414 1.02e-05 ***
## Pclass_X2        -0.9565     0.3362  -2.845 0.004439 **
## Pclass_X3        -1.7728     0.3256  -5.445 5.17e-08 ***
## Title_non_elite  -1.8516     0.3875  -4.778 1.77e-06 ***
## Sex_male         -3.1722     0.2398 -13.226  < 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: 947.98  on 711  degrees of freedom
## Residual deviance: 612.22  on 704  degrees of freedom
## AIC: 628.22
##
## Number of Fisher Scoring iterations: 5
```

```
modelo_reg.logistic_2
```

```
## Generalized Linear Model
##
## 712 samples
##   7 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 712, 712, 712, 712, 712, 712, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.8065235  0.5852163
```

```
summary(modelo_reg.logistic_2$finalModel)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4688  -0.5908  -0.4062   0.6130   2.4397
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)       4.2896     0.4915   8.728  < 2e-16 ***
```

```
## Pclass_X2        -0.9565     0.3362  -2.845 0.004439 **
## Pclass_X3        -1.7728     0.3256  -5.445 5.17e-08 ***
## Title_non_elite  -1.8516     0.3875  -4.778 1.77e-06 ***
## Sex_male         -3.1722     0.2398 -13.226  < 2e-16 ***
## Age              -0.3740     0.1133  -3.301 0.000963 ***
## Fare              0.2269     0.1397   1.624 0.104464
## Family.unit      -0.5627     0.1275  -4.414 1.02e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 947.98  on 711  degrees of freedom
## Residual deviance: 612.22  on 704  degrees of freedom
## AIC: 628.22
##
## Number of Fisher Scoring iterations: 5
```

```
modelo_reg.logistic_3
```

```
## Generalized Linear Model
##
## 712 samples
##   6 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 712, 712, 712, 712, 712, 712, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.8068102  0.5853155
```

```
summary(modelo_reg.logistic_3$finalModel)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3836  -0.5765  -0.4028   0.6234   2.4449
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)       4.4366     0.4855   9.138  < 2e-16 ***
## Pclass_X2        -1.2067     0.3012  -4.006 6.17e-05 ***
## Pclass_X3        -2.0716     0.2725  -7.602 2.91e-14 ***
## Title_non_elite  -1.7932     0.3875  -4.628 3.69e-06 ***
## Sex_male         -3.1752     0.2394 -13.265  < 2e-16 ***
## Age              -0.3845     0.1130  -3.403 0.000668 ***
## Family.unit      -0.5083     0.1220  -4.168 3.07e-05 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 947.98  on 711  degrees of freedom
## Residual deviance: 615.29  on 705  degrees of freedom
## AIC: 629.29
##
## Number of Fisher Scoring iterations: 5
```

Las diferencias entre modelos no son representativas no comportan que el valor AIC disminuya, al contrario aumenta y la precisión se mantiene al rededor del 80%, por lo que voy a optar por mantener el modelo original para la predicción.

```
library(vcd)
```

```
## Loading required package: grid
```

```
#predicciones de los valores de supervivencia.
predicciones_raw <- predict(modelo_reg.logistic, newdata = test.sample.ok,
                            type = "raw")

#predicciones  de la probabilidad de que lla supervivencia sea 0 u 1
predicciones_prob<- predict(modelo_reg.logistic, newdata = test.sample.ok,
                            type = "prob")
```

5. Representación de los resultados a partir de tablas i graficas

Al tratarse de un modelo con 5 predictores no es posible hacer una representación gráfica que los incluya a todos ya que necesitaríamos un espacio con 5 dimensiones. No obstante se pueden hacer representaciones de la supervivencia en relación a las variables más representativas para el modelo predictivo en este caso la clase, el sexo y la pertenencia a un grupo que se pueda considerar "elite"

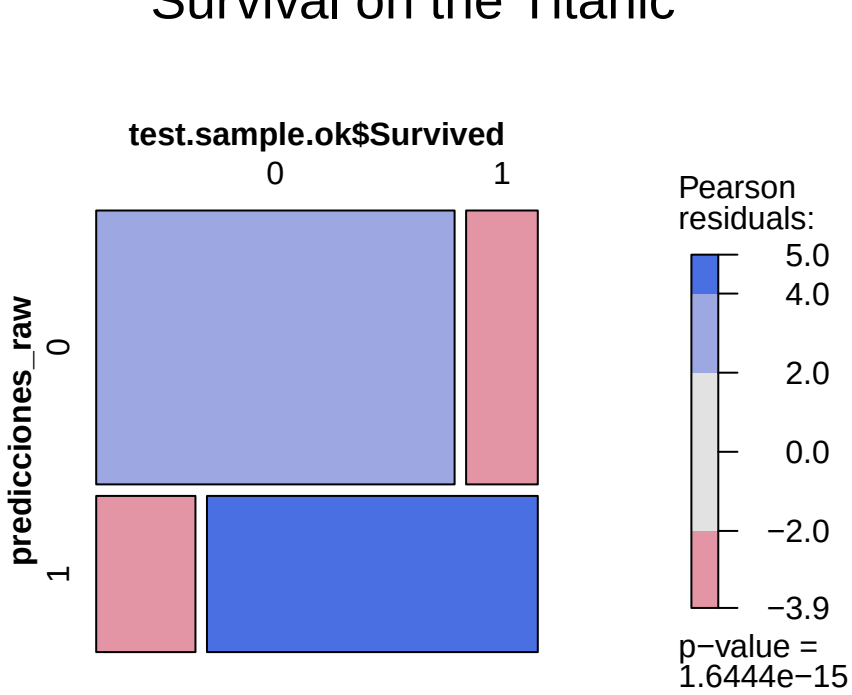
Además también podemos representar el grado de aciertos que tiene nuestro modelo en relación a las observaciones conocidas.

```
#Para evaluar los aciertos

confusionMatrix(data = predicciones_raw, reference = test.sample.ok$Survived,positive="1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 95 19
##          1 15 50
##
##                Accuracy : 0.8101
##                  95% CI : (0.7448, 0.8647)
##     No Information Rate : 0.6145
##     P-Value [Acc > NIR] : 1.35e-08
```

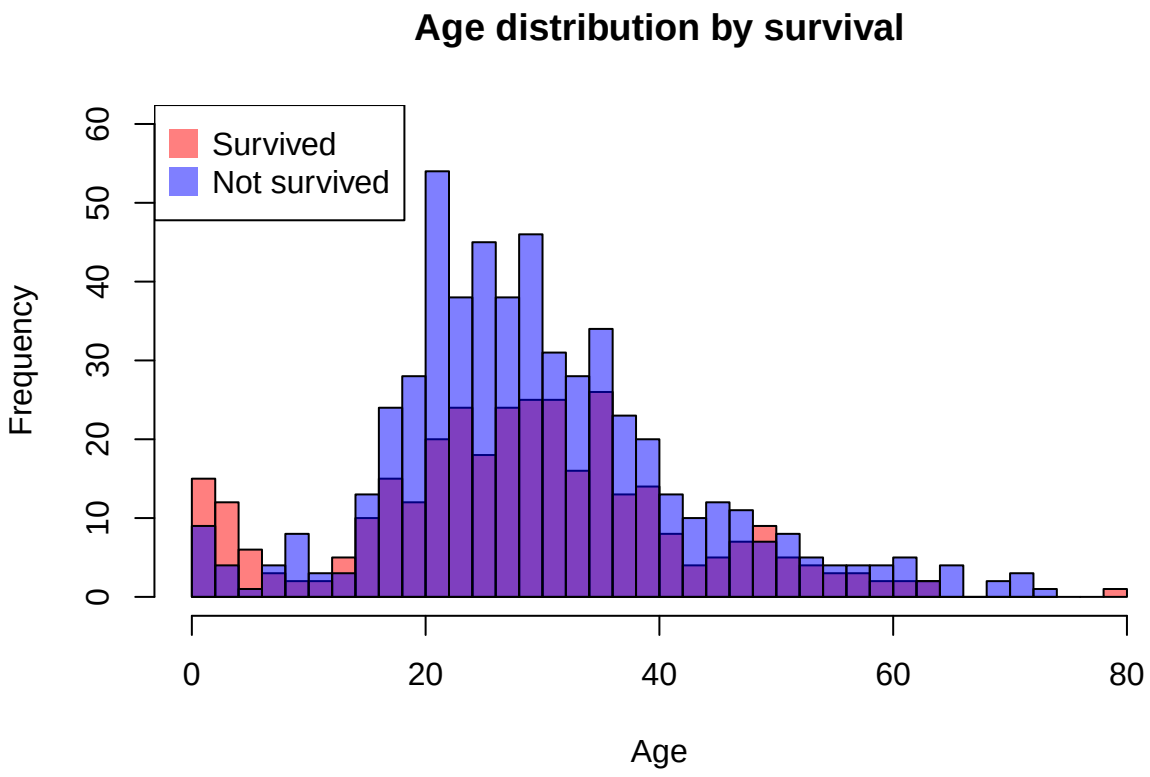
```
##
##                  Kappa : 0.5947
##
##  Mcnemar's Test P-Value : 0.6069
##
##             Sensitivity : 0.7246
##             Specificity : 0.8636
##          Pos Pred Value : 0.7692
##          Neg Pred Value : 0.8333
##              Prevalence : 0.3855
##          Detection Rate : 0.2793
##    Detection Prevalence : 0.3631
##       Balanced Accuracy : 0.7941
##
##        'Positive' Class : 1
##
```

```
mosaic(~ predicciones_raw + test.sample.ok$Survived,
main = "Survival on the Titanic", shade = TRUE, colorize=T ,legend = TRUE)
```

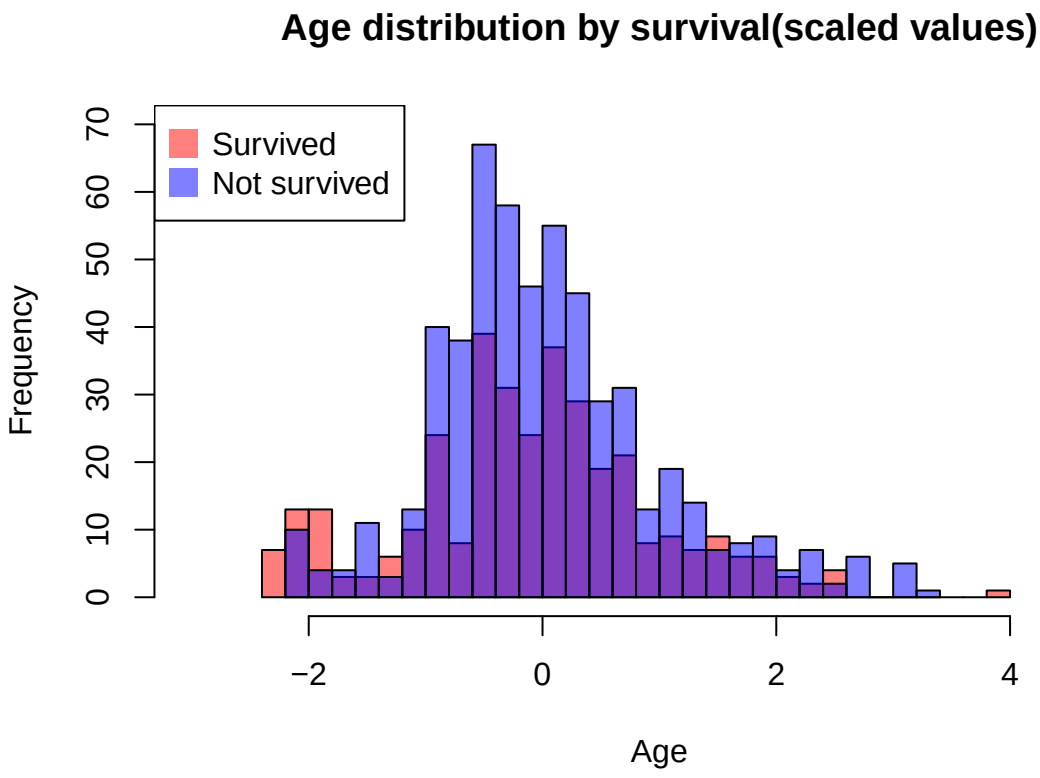


Tanto en la tabla como en la matriz de confusión podemos ver el nivel de aciertos del modelo ya que estamos comparando los valores originales del grupo test con valores conocidos con la predicción que hace el modelo en rosa se representan los valores para los que no se ha acertado.

```r
#Para la representación de la edad quiero utilizar los datos sin escalar así que utilizaré el dataframe

hist(Titanic_complete_raw[Titanic_complete_raw$Survived==1,"Age"], breaks=30,xlim=c(0,80), ylim=c(0,60)
hist(Titanic_complete_raw[Titanic_complete_raw$Survived==0,"Age"], breaks=30,xlim=c(0,80), ylim=c(0,60)
legend("topleft", legend=c("Survived","Not survived"), col=c(rgb(1,0,0,0.5),
     rgb(0,0,1,0.5)), pt.cex=2, pch=15 )
```




```r
#Adicionalmente hago también una representación de los datos escalados y centrados

hist(Titanic_complete[Titanic_complete$Survived==1,"Age"], breaks=30,xlim=c(-3,5), ylim=c(0,70),col=rgb
hist(Titanic_complete[Titanic_complete$Survived==0,"Age"], breaks=30,xlim=c(-3,5), ylim=c(0,70), col=rgb
legend("topleft", legend=c("Survived","Not survived"), col=c(rgb(1,0,0,0.5),
     rgb(0,0,1,0.5)), pt.cex=2, pch=15 )
```




```
length(Titanic_complete_raw[Titanic_complete_raw$Age<10,"Age"])
```

```
## [1] 82
```

```
(82/1309)*100
```

```
## [1] 6.264324
```

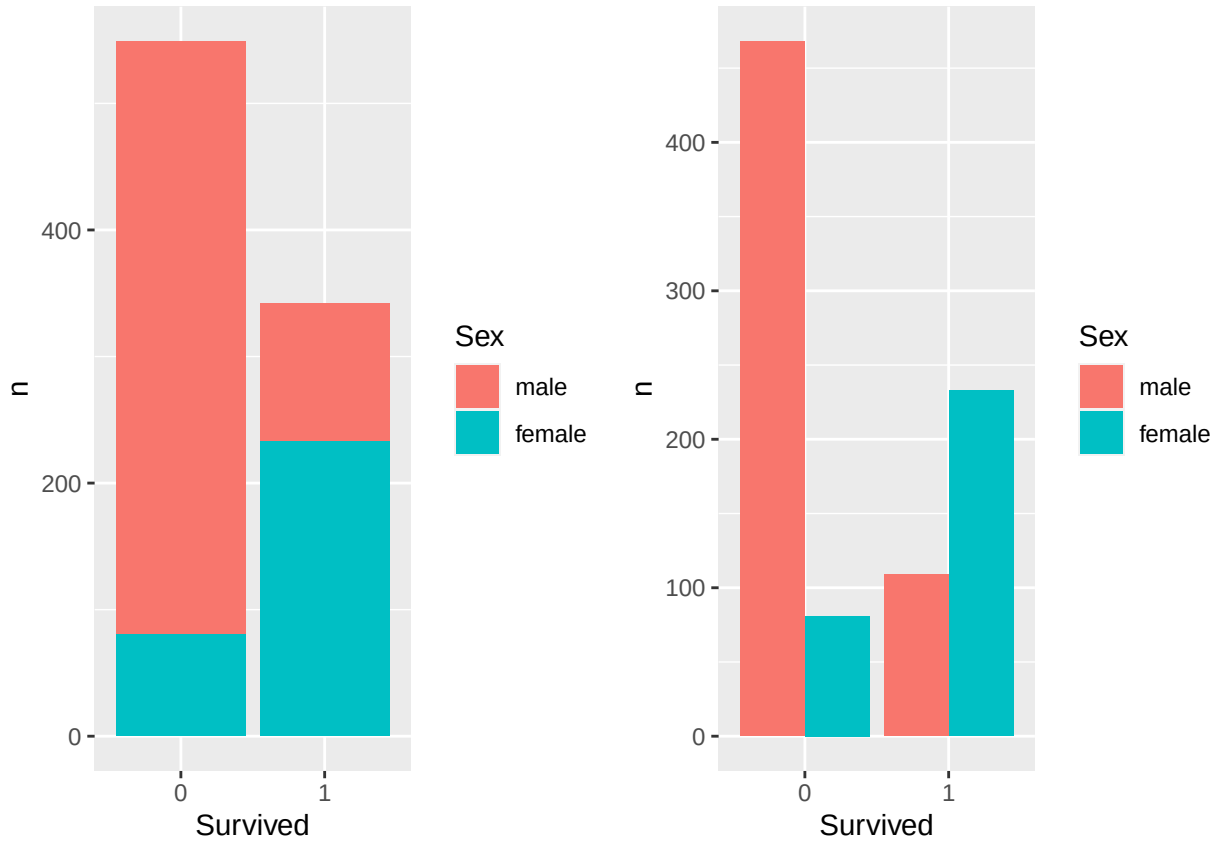
De este último gráfico la conclusión que obtengo es de que pese a que los valores son diferentes(en el segundo grupo se han normalizado) la distribución es muy similar y el único rango en que la supervivencia es mayor es en el de los niños menores de 10 años, esto explica porque en nuestro modelo la variable age no es tan significativa como esperaríamos si contabilizamos el número de pasajeros con menos de 10 años podremos ver que no representan más del 6% de los pasajeros totales por lo que la supervivencia que aumenta en este rango de edad tiene poco efecto en la muestra general y esto explicaria que no se trate de un factor tan importante para el modelo. Por otra parte otras variables que representaremos a continuación si tendrían un peso más importante en el modelo. Será el caso de la variable Sex, Class y title.

Para hacer la representación de estas variables utilizaré el dataset Titanic_complete final donde ya no hay valores missing pero sin tener aún las variables binarizadas. He considerado que era más facil hacer la representación de este modo siendo que la intención de estas visualizaciones es explicar los resultados que extraemos de la interpretación del modelo, y que tanto los datos iniciales como finales si el modelo es correcto tendrán un comportamiento similar al que se ve en el modelo.

```
library(dplyr)
library(forcats)

#Grafico de la variable Survived en relación a la variable sex
agg_1 <-  na.omit(count(Titanic_complete, Survived,Sex))
head(agg_1)
```

```
##   Survived    Sex   n
## 1        0 female  81
## 2        0   male 468
## 3        1 female 233
## 4        1   male 109
```

```
agg_ord_1 <- mutate(agg_1,
                    Survived = reorder(Survived, -n, sum),
                    Sex = reorder(Sex, -n, sum))
p1 <- ggplot(agg_ord_1) +
      geom_col(aes(x = Survived, y = n, fill = Sex))
p2 <- ggplot(agg_ord_1) +
      geom_col(aes(x = Survived, y = n, fill = Sex), position = "dodge")
grid.arrange(p1, p2, nrow = 1)
```

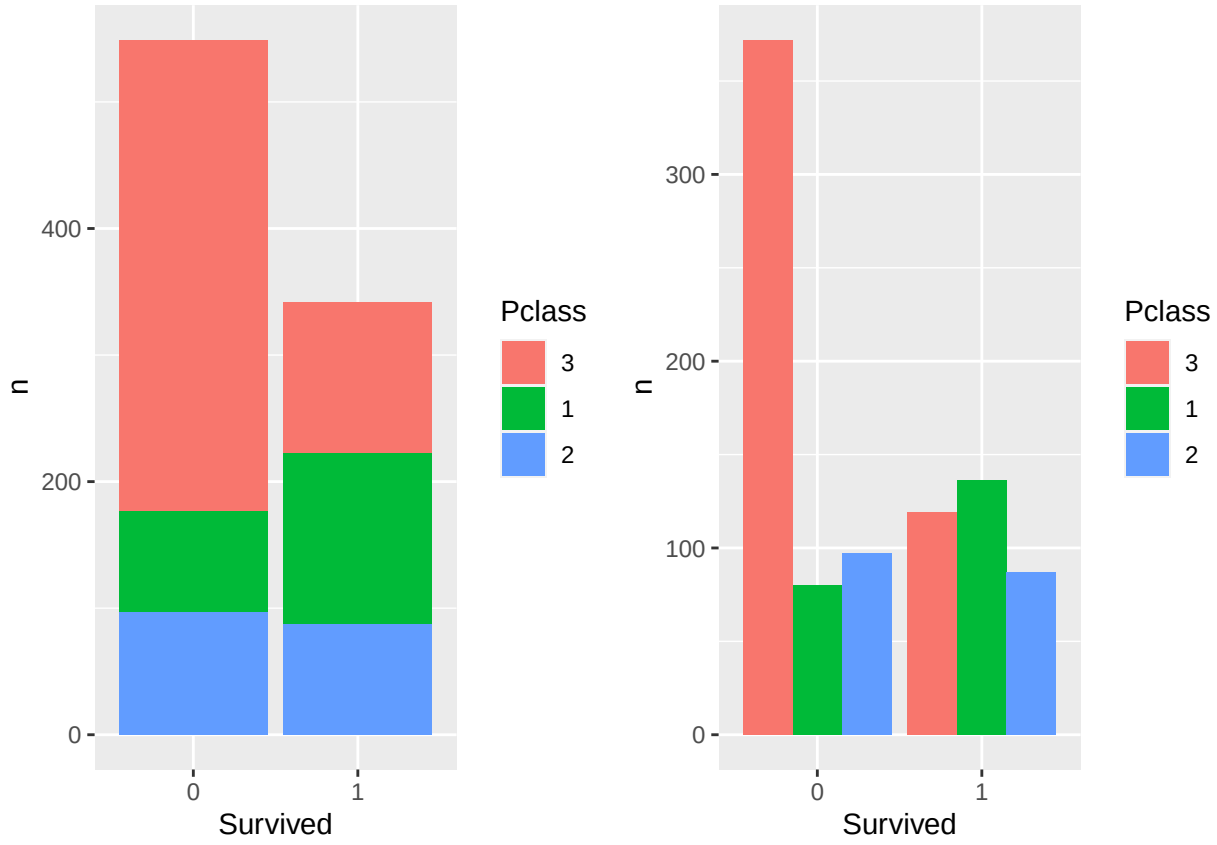



```
#Grafico de la variable Survived en relación a la variable Pclass
```

```r
agg_2 <-  na.omit(count(Titanic_complete, Survived,Pclass))
head(agg_2)
```

```
##   Survived Pclass   n
## 1        0      1  80
## 2        0      2  97
## 3        0      3 372
## 4        1      1 136
## 5        1      2  87
## 6        1      3 119
```

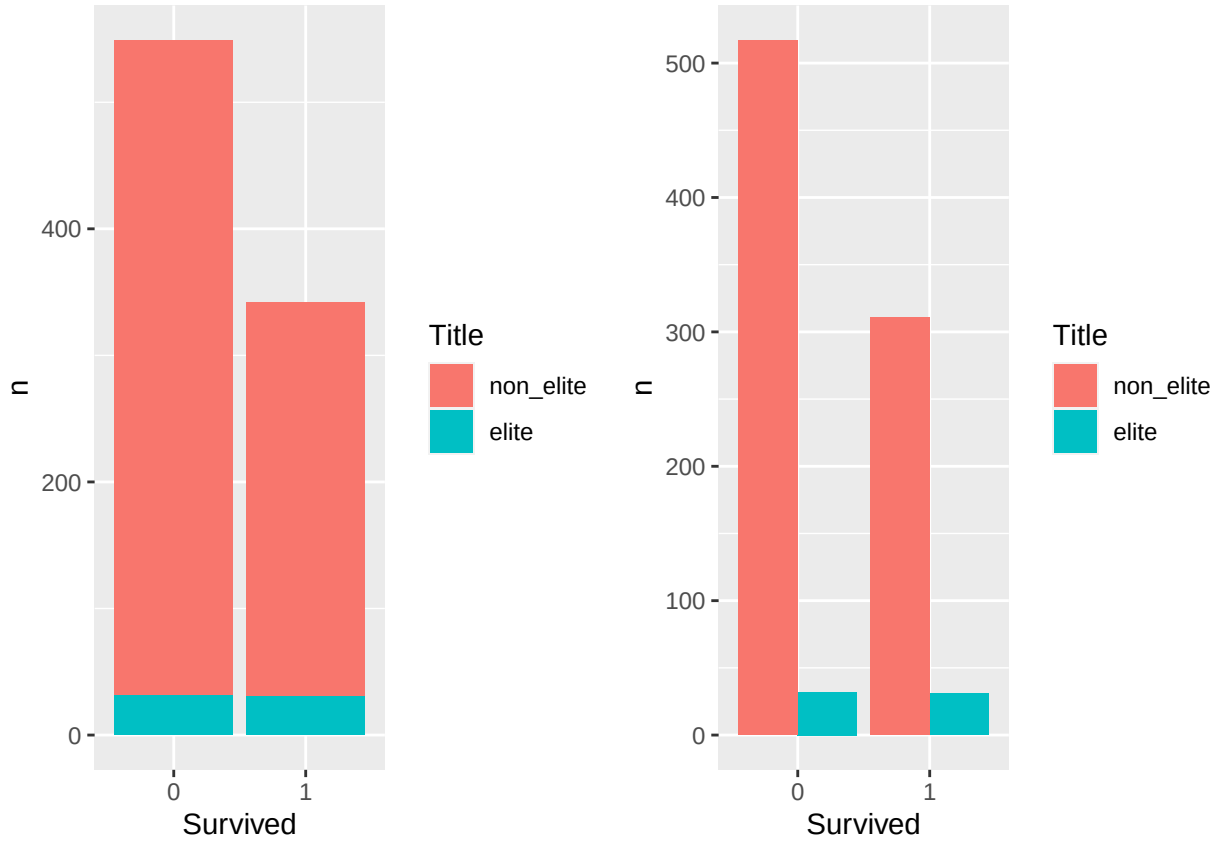
```r
agg_ord_2 <- mutate(agg_2,
                  Survived = reorder(Survived, -n, sum),
                  Pclass = reorder(Pclass, -n, sum))
p3 <- ggplot(agg_ord_2) +
      geom_col(aes(x = Survived, y = n, fill = Pclass))
p4 <- ggplot(agg_ord_2) +
      geom_col(aes(x = Survived, y = n, fill = Pclass), position = "dodge")
grid.arrange(p3, p4, nrow = 1)
```




```r
#Grafico de la variable Survived en relación a la variable Title

agg_3 <-  na.omit(count(Titanic_complete, Survived,Title))
head(agg_3)
```

```
##   Survived     Title   n
## 1        0     elite  32
## 2        0 non_elite 517
## 3        1     elite  31
## 4        1 non_elite 311
```

```
agg_ord_3 <- mutate(agg_3,
                    Survived = reorder(Survived, -n, sum),
                    Title = reorder(Title, -n, sum))
p3 <- ggplot(agg_ord_3) +
      geom_col(aes(x = Survived, y = n, fill = Title))
p4 <- ggplot(agg_ord_3) +
      geom_col(aes(x = Survived, y = n, fill = Title), position = "dodge")
grid.arrange(p3, p4, nrow = 1)
```



La interpretación de estos gráficos es que en efecto el sexo es una variable que influye en la supervivencia, los hombres sobreviven menos, lo mismo con la clase, un pasajero de 3ª clase tenía menos posibilidades de sobrevivir y en cuanto a la pertenencia a una élite es un factor desfavorable en caso de no pertenecer a la Elite

Lo cierto es que lo que nos sugieren todos estos datos no dan un buen pronostico para el pasajero James Kelly al que estoy haciendo seguimiento en el análisis

Sería interesante también calcular los logaritmos de los coeficientes para ver en que proporciones afecta cada factor, pero dado que no es el objetivo final del proyecto no me adentraré en este aspecto.

6.Resolución del problema.

```
# Aplicamos el modelo predictivo a la muestra test en blanco que tenemos
Test.f.ok<-bake(receta_trained, new_data = Test.f)

Predicted <- predict(modelo_reg.logistic, newdata = Test.f.ok,
                             type = "raw")
Test.f$Survived<-Predicted

#Compruebo también la predicción que ha hecho el modelo sobre el pasajero 892 Kelly, Mr. James hombre d

Test.f[Test.f$PassengerId==892]
```

```
##     PassengerId
## 1           892
## 2           893
## 3           894
## 4           895
## 5           896
## 6           897
## 7           898
## 8           899
## 9           900
## 10          901
## 11          902
## 12          903
## 13          904
## 14          905
## 15          906
## 16          907
## 17          908
## 18          909
## 19          910
## 20          911
## 21          912
## 22          913
## 23          914
## 24          915
## 25          916
## 26          917
## 27          918
## 28          919
## 29          920
## 30          921
## 31          922
## 32          923
## 33          924
## 34          925
## 35          926
## 36          927
## 37          928
## 38          929
## 39          930
```

```
## 40          931
## 41          932
## 42          933
## 43          934
## 44          935
## 45          936
## 46          937
## 47          938
## 48          939
## 49          940
## 50          941
## 51          942
## 52          943
## 53          944
## 54          945
## 55          946
## 56          947
## 57          948
## 58          949
## 59          950
## 60          951
## 61          952
## 62          953
## 63          954
## 64          955
## 65          956
## 66          957
## 67          958
## 68          959
## 69          960
## 70          961
## 71          962
## 72          963
## 73          964
## 74          965
## 75          966
## 76          967
## 77          968
## 78          969
## 79          970
## 80          971
## 81          972
## 82          973
## 83          974
## 84          975
## 85          976
## 86          977
## 87          978
## 88          979
## 89          980
## 90          981
## 91          982
## 92          983
## 93          984
```

```
## 94          985
## 95          986
## 96          987
## 97          988
## 98          989
## 99          990
## 100         991
## 101         992
## 102         993
## 103         994
## 104         995
## 105         996
## 106         997
## 107         998
## 108         999
## 109        1000
## 110        1001
## 111        1002
## 112        1003
## 113        1004
## 114        1005
## 115        1006
## 116        1007
## 117        1008
## 118        1009
## 119        1010
## 120        1011
## 121        1012
## 122        1013
## 123        1014
## 124        1015
## 125        1016
## 126        1017
## 127        1018
## 128        1019
## 129        1020
## 130        1021
## 131        1022
## 132        1023
## 133        1024
## 134        1025
## 135        1026
## 136        1027
## 137        1028
## 138        1029
## 139        1030
## 140        1031
## 141        1032
## 142        1033
## 143        1034
## 144        1035
## 145        1036
## 146        1037
## 147        1038
```

```
## 148        1039
## 149        1040
## 150        1041
## 151        1042
## 152        1043
## 153        1044
## 154        1045
## 155        1046
## 156        1047
## 157        1048
## 158        1049
## 159        1050
## 160        1051
## 161        1052
## 162        1053
## 163        1054
## 164        1055
## 165        1056
## 166        1057
## 167        1058
## 168        1059
## 169        1060
## 170        1061
## 171        1062
## 172        1063
## 173        1064
## 174        1065
## 175        1066
## 176        1067
## 177        1068
## 178        1069
## 179        1070
## 180        1071
## 181        1072
## 182        1073
## 183        1074
## 184        1075
## 185        1076
## 186        1077
## 187        1078
## 188        1079
## 189        1080
## 190        1081
## 191        1082
## 192        1083
## 193        1084
## 194        1085
## 195        1086
## 196        1087
## 197        1088
## 198        1089
## 199        1090
## 200        1091
## 201        1092
```

```
## 202        1093
## 203        1094
## 204        1095
## 205        1096
## 206        1097
## 207        1098
## 208        1099
## 209        1100
## 210        1101
## 211        1102
## 212        1103
## 213        1104
## 214        1105
## 215        1106
## 216        1107
## 217        1108
## 218        1109
## 219        1110
## 220        1111
## 221        1112
## 222        1113
## 223        1114
## 224        1115
## 225        1116
## 226        1117
## 227        1118
## 228        1119
## 229        1120
## 230        1121
## 231        1122
## 232        1123
## 233        1124
## 234        1125
## 235        1126
## 236        1127
## 237        1128
## 238        1129
## 239        1130
## 240        1131
## 241        1132
## 242        1133
## 243        1134
## 244        1135
## 245        1136
## 246        1137
## 247        1138
## 248        1139
## 249        1140
## 250        1141
## 251        1142
## 252        1143
## 253        1144
## 254        1145
## 255        1146
```

```
## 256        1147
## 257        1148
## 258        1149
## 259        1150
## 260        1151
## 261        1152
## 262        1153
## 263        1154
## 264        1155
## 265        1156
## 266        1157
## 267        1158
## 268        1159
## 269        1160
## 270        1161
## 271        1162
## 272        1163
## 273        1164
## 274        1165
## 275        1166
## 276        1167
## 277        1168
## 278        1169
## 279        1170
## 280        1171
## 281        1172
## 282        1173
## 283        1174
## 284        1175
## 285        1176
## 286        1177
## 287        1178
## 288        1179
## 289        1180
## 290        1181
## 291        1182
## 292        1183
## 293        1184
## 294        1185
## 295        1186
## 296        1187
## 297        1188
## 298        1189
## 299        1190
## 300        1191
## 301        1192
## 302        1193
## 303        1194
## 304        1195
## 305        1196
## 306        1197
## 307        1198
## 308        1199
## 309        1200
```

```
## 310        1201
## 311        1202
## 312        1203
## 313        1204
## 314        1205
## 315        1206
## 316        1207
## 317        1208
## 318        1209
## 319        1210
## 320        1211
## 321        1212
## 322        1213
## 323        1214
## 324        1215
## 325        1216
## 326        1217
## 327        1218
## 328        1219
## 329        1220
## 330        1221
## 331        1222
## 332        1223
## 333        1224
## 334        1225
## 335        1226
## 336        1227
## 337        1228
## 338        1229
## 339        1230
## 340        1231
## 341        1232
## 342        1233
## 343        1234
## 344        1235
## 345        1236
## 346        1237
## 347        1238
## 348        1239
## 349        1240
## 350        1241
## 351        1242
## 352        1243
## 353        1244
## 354        1245
## 355        1246
## 356        1247
## 357        1248
## 358        1249
## 359        1250
## 360        1251
## 361        1252
## 362        1253
## 363        1254
```

```
## 364        1255
## 365        1256
## 366        1257
## 367        1258
## 368        1259
## 369        1260
## 370        1261
## 371        1262
## 372        1263
## 373        1264
## 374        1265
## 375        1266
## 376        1267
## 377        1268
## 378        1269
## 379        1270
## 380        1271
## 381        1272
## 382        1273
## 383        1274
## 384        1275
## 385        1276
## 386        1277
## 387        1278
## 388        1279
## 389        1280
## 390        1281
## 391        1282
## 392        1283
## 393        1284
## 394        1285
## 395        1286
## 396        1287
## 397        1288
## 398        1289
## 399        1290
## 400        1291
## 401        1292
## 402        1293
## 403        1294
## 404        1295
## 405        1296
## 406        1297
## 407        1298
## 408        1299
## 409        1300
## 410        1301
## 411        1302
## 412        1303
## 413        1304
## 414        1305
## 415        1306
## 416        1307
## 417        1308
```

```
## 418        1309
```

##7. A partir de los resultados obtenidos, cuales són las conclusions? Los resultados permiten responder al problema.

A partir del modelo de regresión logística podemos predecir con un 80% de precisión si un pasajero sobrevivirá al naufragio considerando factores como la edad, el sexo, la clase y el número de familiares con el que viajaba, o su nivel de influencia (título).

En terminos generales y saliendo de la predicción propiammente, en el modelo el factor que tiene más influencia es el género, las mujeres tenían más probabilidad de supervivencia en el Titánic. Esto va en la linea de lo que ya hemos visto desde la exploración inicial Otras variables que afectarían serían los títulos (entendidos como la pertenencia a una élite por herencia o por trabajos privilegiados), la clase, y el tamaño de la unidad familiar con la que se viajaba, también sería significativa la edad aunque en menor medida y lo que no influiria tanto sería la tarifa.

En el momento de analizar la tarifa ya indicamos que es una variable compuesta por lo que es dificil determinar hasta que punto es relevante sin conocer todos los factores de las que esta compuesta.

Aún así el modelo si nos permite hacer una predicción aunque esta sería una primera iteración del proyecto hay varias variables que tienen potencial para seguir siendo exploradas y además se pueden hacer pruebas adicionales en cuanto a comparación de modelos por ejemplo, que sería lo más habitual y recomendable, en mi caso por el límite de tiempo de dedicación no he podido indagar más en algunos aspectos como sería el de plantear más modelos o pararme más en las variables, o el ver la influencia que tendría cada factor en el modelo.

La variable cabin y embarked considero que se podrían haber desarrollado más e incluso ticket es posible que las letras de los tickets tuvieran alguna relación con el puerto de embarque o con las cabinas pero, son variables que requieren más tiempo de preprocesado y de investigación, y sería también interesante contraponer esos datos a los datos técnicos que se pueden encontrar en cuanto a como se produjo el hundimiento, por donde se produjo el choque, etc.

Además a pesar de que la precisión del modelo parece estar dentro de un rango bueno del 80% el valor del valor del criterio de AIC es de 628.83 que resulta algo elevado por lo que seria posible implementar modelos diferentes para disminuir ese valor y que la valoración del modelo sea mejor.

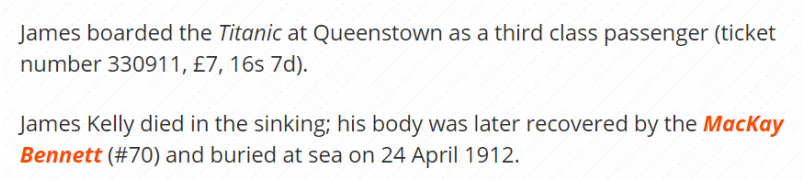
Aún así en terminos generales creo que las conclusiones serían:

- El modelo de regresión logística múltiple nos permite predecir los pasajeros que sobreviven y mueren en el Titánic tratandose de una variable dicotómica.
- Las variables que tienen mayor impacto en el modelo son el sexo y la clase.
- La tarifa no esta necesariamente relacionada con la clase, y es una variable compleja que no se puede asociar directamente con la clase alta.

*Las personas con influencia (títulos o trabajos reconocidos)tenían más probabilidades de sobrevivir que personas que no tuvieran esto.

*La edad no es un factor tan determinante por lo que el ser menor de edad no garantizaba la supervivencia.

*El pasajero 892 Lo que he visto y analizado es que tenía varios factores importantes en contra, era hombre y viajaba en tercera clase. La predicción del modelo es que no sobrevivió y buscando información en internet se confirma esto.



Estas serían las conclusiones finales que tengo por el momento. La bibliografia que he utilizado y el detalle sobre el contenido del repositorio se puede consultar en https://github.com/MilaRG/Titanic_with_R