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 en base a determinadas características de los mismos que tenemos registradas en el dataset.

La importancia que tiene el proyecto es que nos permite responder a la pregunta en base a factores que tengamos de "nuevos" pasajeros. Es decir, en base a ciertos parametros habría sobrevivido el pasajero X. Supongamos que el pasajero x es un hombre, con un pasaje de segunda clase, sin hijos y 45 años de edad, ¿qué probabilidad tiene de sobrevivir a este accidente?

Y extrapolando este tipo de estudios a otras situaciones, un modelo similar nos podría permitir determinar si en función de ciertas características por ejemplo un usuario de una plataforma compraría determinado producto. O un paciente con determinadas patologías o características respondería bien a x tratamiento.

También se debe tener en cuenta las siguientes notas sobre los datos que utilizaremos 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.

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

Lo primer que hago para determinar los datos de interés y la integración es revisar los datos de los que estoy partiendo.

En mi caso tengo tres tipos de datos Train,test,gender_submision. En la descripción de los archivos que se puede encontrar en la fuente de Kaggle 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()
```

```r
library(ggplot2)
library(tidyverse)
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
```

```r
library(viridis)
```

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

```r
#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" ...
```

```r
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" ...
```

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. . .)

```
#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 y 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 las variables categóricas que he considerado.

```
#change character to factor object

Titanic_complete$Survived<-as.factor(Titanic_complete$Survived)
Titanic_complete$Pclass<-as.factor(Titanic_complete$Pclass)
Titanic_complete$Sex<-as.factor(Titanic_complete$Sex)
Titanic_complete$Cabin<-as.factor(Titanic_complete$Cabin)
Titanic_complete$Embarked<-as.factor(Titanic_complete$Embarked)
Titanic_complete$Ticket<-as.factor(Titanic_complete$Ticket)
Titanic_complete$Set_type<-as.factor(Titanic_complete$Set_type)
#Titanic_complete$Name<-as.factor(Titanic_complete$Name)
```

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 est

paste("número de registros con valor zero en la tarifa:",sum(Titanic_train$Fare==0))
```

```
## [1] "número de registros con valor zero en la tarifa: 15"
```

Con esta primera conversión ya 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.

Los valores missing más relevantes los tenemos en Cabin y en Age por lo que habrá que tratarlos, Los otros valores missing corresponden a Embarked y Fare, pero en total 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, me hace pensar que si quiero usar estas variables para el modelo tendré que tratarlas de alguna manera.

```
#comprobación de variables categóricas Cabin y TIcket
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]),header=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. Así que 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.

```
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
```

En esta primera vista, me llama la atención que las pasajeras coinciden en número ticket, tarifa, clase y cabina, así que 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.

Si encuentro pasajeros que tengan la misma cabina o el mismo ticket es muy posible que embarcaran en el mismo puerto que estas dos pasajeras así que 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)
```

```
#valorar usar knn???
#library(VIM)
#selected.vars<-c("Embarked","Pclass","Fare")
#output <- kNN( Titanic_complete[,selected.vars], variable=c("Embarked"), k=3 )
#output[output$BPD_imp==TRUE,]
```

No tengo ningún resultado, y a priori en este punto considero que al tratarse de únicamente 2 registros podríamos eliminarlos, y en un caso 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.

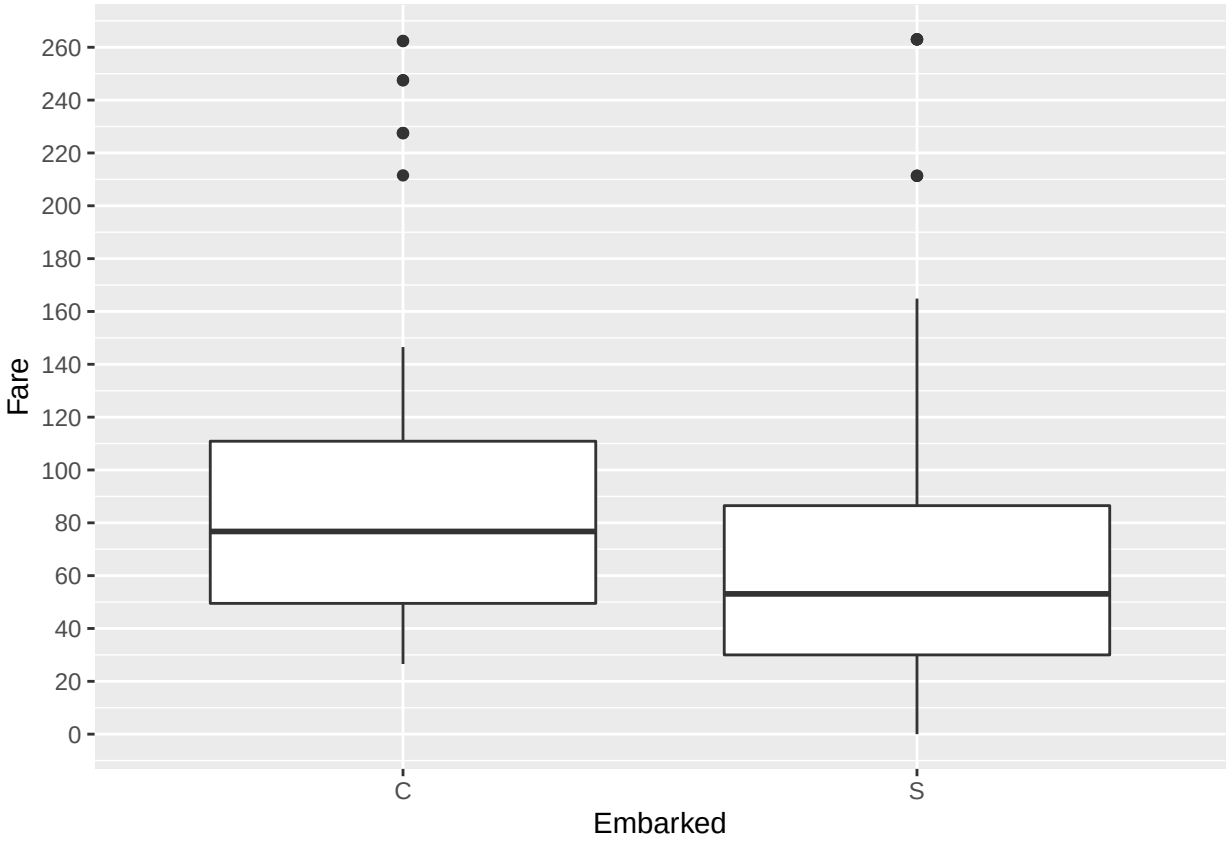
Utilizo la función table para verificar en que puertos han subido mayoritariamente los pasajeros de primera clase, ya que si hay una clara mayoría en este punto se resolvería el problema.

```
#Puertos de embarque segun clase

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

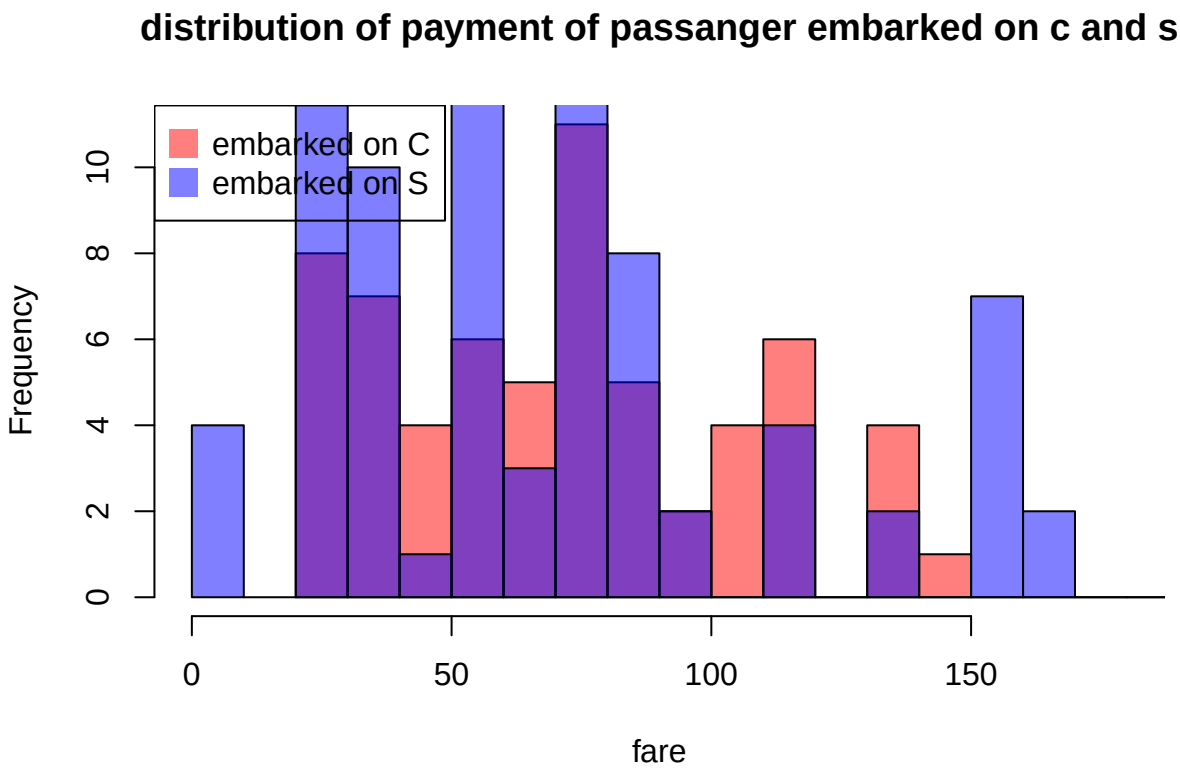
```
##
##           C   Q   S
##   1   0 141   3 177
##   2   0  28   7 242
##   3   0 101 113 495
```

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<-Titanic.embarked %>%
  filter_all(~ !is.na(.))

#boxplot
ggplot(Titanic.embarked,aes(x=Embarked, y=Fare))+geom_boxplot()+ scale_y_continuous(breaks = seq(0,300,1
```




```
#quiero comparar los pasajeros que embarcaron por C y por S, creo dos subsets para hacer una representa

Titanic.embarkedC<-Titanic.embarked[Titanic.embarked$Embarked=="C",]
Titanic.embarkedS<-Titanic.embarked[Titanic.embarked$Embarked=="S",]

hist(Titanic.embarkedC$Fare, breaks=30,xlim=c(0,180), col=rgb(1,0,0,0.5), xlab="fare", main="distributi
hist(Titanic.embarkedS$Fare, breaks=30,xlim=c(0,180), col=rgb(0,0,1,0.5), add=T)
legend("topleft", legend=c("embarked on C","embarked on 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. Que 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"

```
#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, y zero values

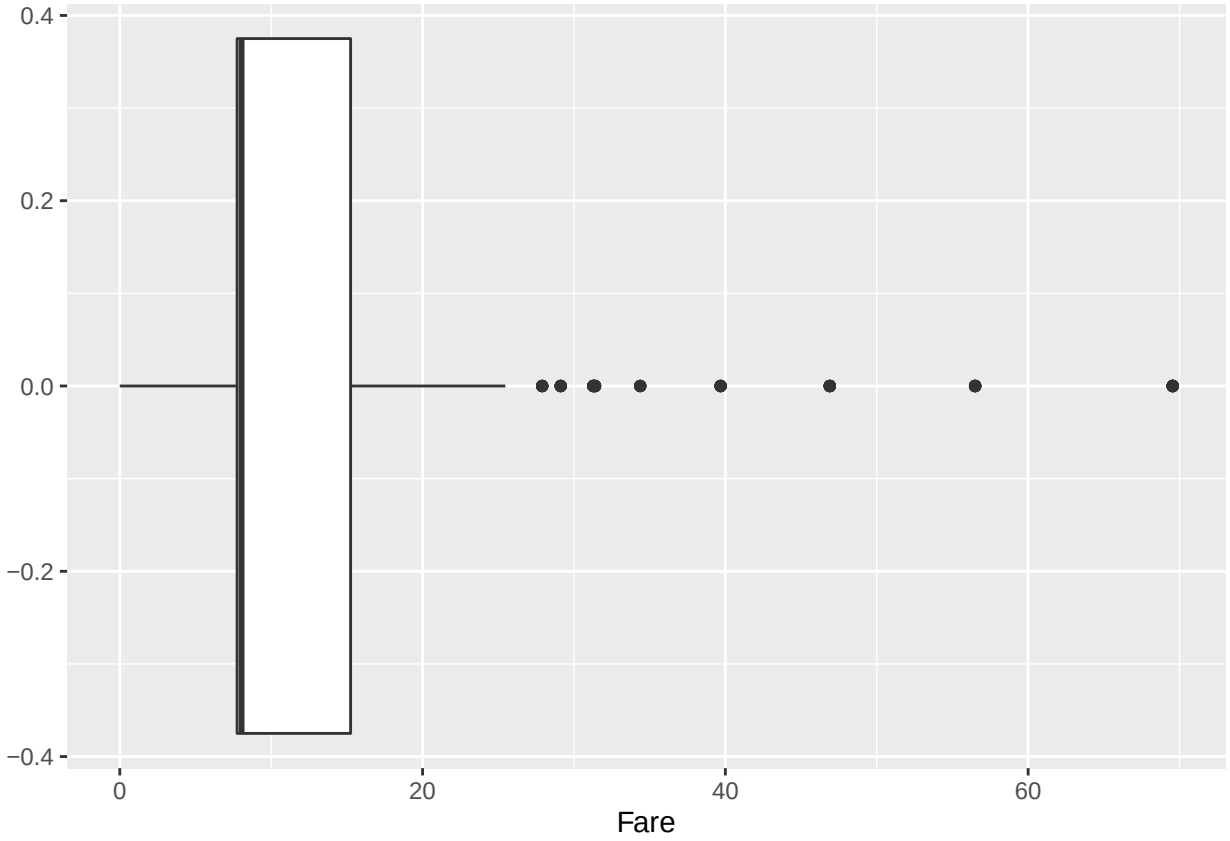
A priori este valor faltante se podría considerar de tipo MAR(Missing at random), es decir a priori se podría explicar esta variable a partir de la classe y tal vez también por el puerto de embarque pero solo con la clase deberíamos poder tener una aproximación al dato.

```
#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 será mejor reemplazar el valor de la variable por la mediana que suele verse menos afectada por valores extremos, así pues reemplazo el valor.

```r
#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

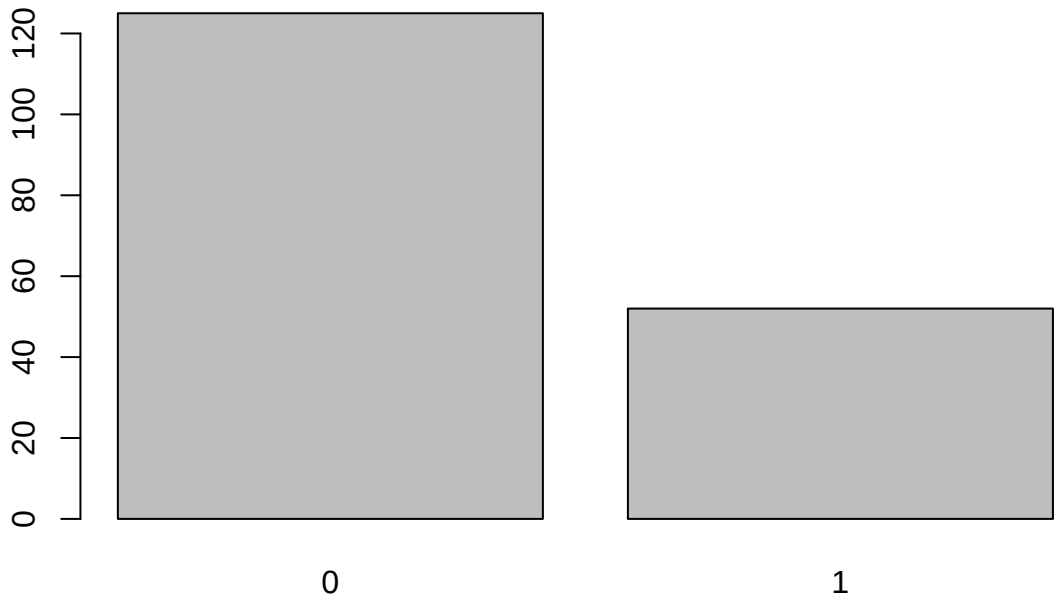
La imputación que haga de esta variable tendrá una repercusión mayor en el modelo por lo tanto quiero primero hacer una evaluación de los valores que si tenemos para esta variable y también de los registros que no tienen esta variable missing para ver cual sería la mejor manera de imputar estos valores.

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.

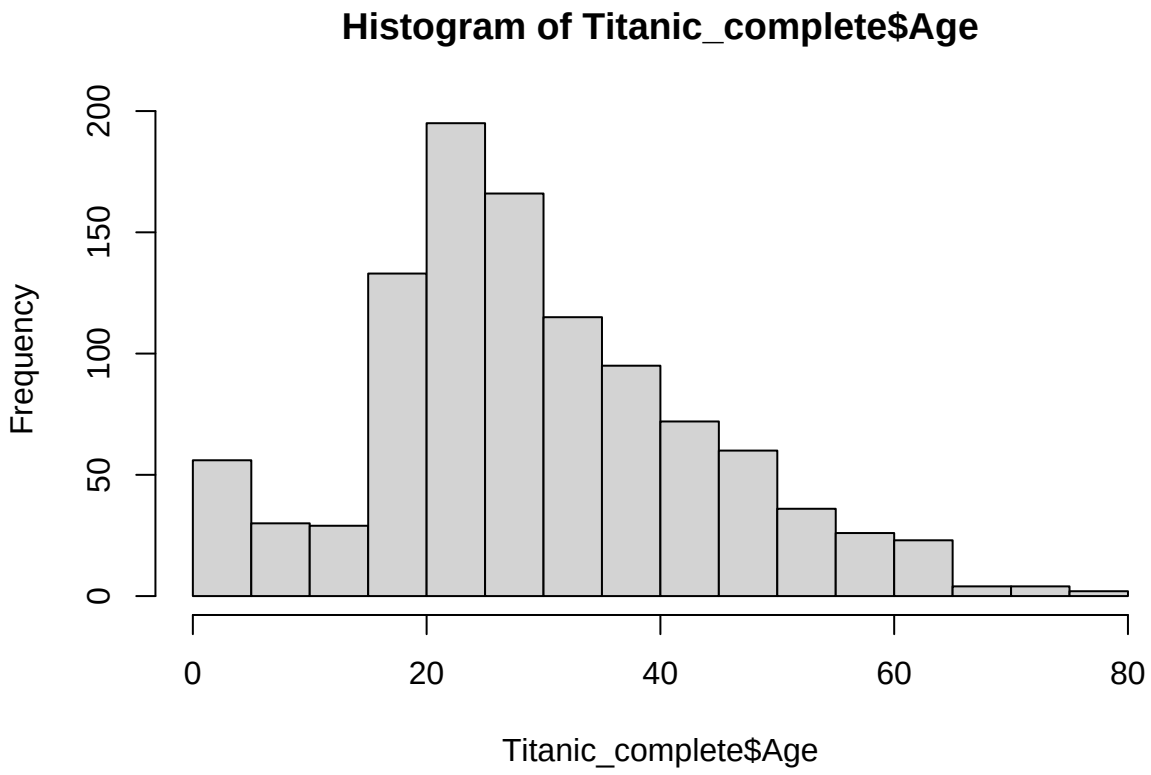
```r
#identificadores de los registros con missing data para la variable Age
id.mis.age <- which( is.na(Titanic_complete$Age))
plot(Titanic_complete[id.mis.age,"Survived"])
```



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. Por lo tanto en efecto si se trata de valores MAR.

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)
```




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

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

La distribución de las edades en un histograma se aproximan 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 realmente constituye el 50% de los registros.

```r
#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
```

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

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.17   22.00   29.00   29.91   37.00   80.00
```

Hacer esta imputación provoca que el rango intercuartílico se mantenga más similar al inicial y también que la media se altere menos. Aunque no es un método e imputación ideal.

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 creo que si separamos la letra de los números que forman cada cabina podemos tener categorías más claras, y ver si hay algún tratamiento que podemos hacer

```r
#Creo una nueva columna resumen de las cabinas solo con la letra, buscando información sobre las cabina

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  29     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
```

```
#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)
```

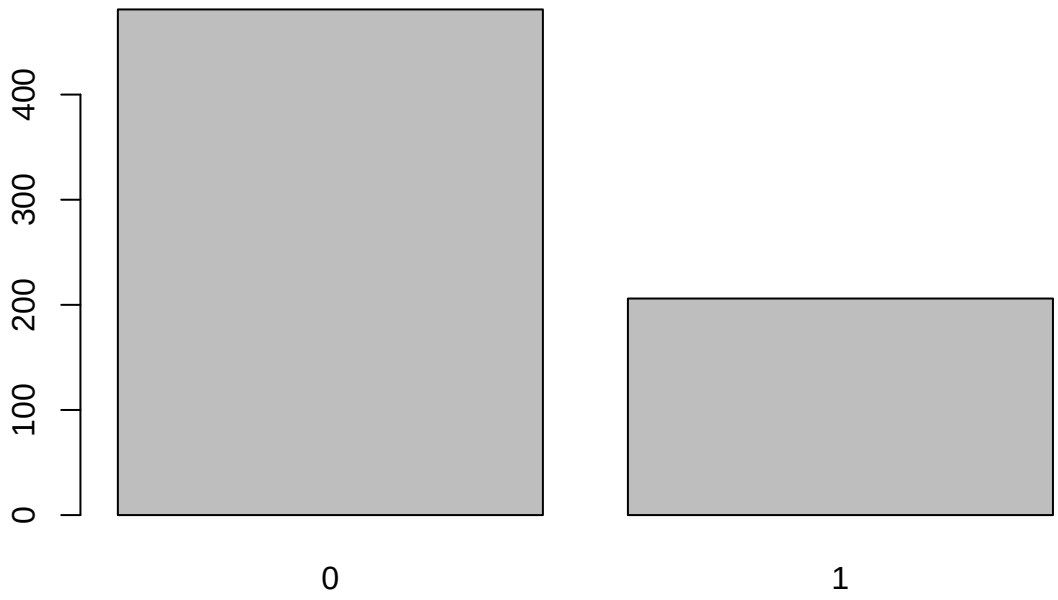
```
##
##                A    B    C    D    E    F    G    T  Sum
##   1     67   22   65   94   40   34    0    0    1  323
##   2    254    0    0    0    6    4   13    0    0  277
##   3    693    0    0    0    0    3    8    5    0  709
##   Sum 1014   22   65   94   46   41   21    5    1 1309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

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.

```
#quiero verificar si los datos faltantes corresponden mayoritariamente a los pasajeros que no sobrevivi

plot(Titanic_complete[Titanic_complete$Cabin=="","Survived"])
```




```
Titanic_complete$Cabin<-as.character(Titanic_complete$Cabin)
#substituyo los valores missing por una N para poder trabajar con los demás e 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  29     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 registres correspondientes a los ti
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  31
## 303          303        0      3       Johnson, Mr. William Cahoone Jr male  19
## 414          414        0      2        Cunningham, Mr. Alfred Fleming male  22
## 467          467        0      2                 Campbell, Mr. William male  22
## 482          482        0      2      Frost, Mr. Anthony Wood "Archie" male  24
## 598          598        0      3                   Johnson, Mr. Alfred male  49
## 634          634        0      1         Parr, Mr. William Henry Marsh male  37
## 675          675        0      2            Watson, Mr. Ennis Hastings male  33
## 733          733        0      2                  Knight, Mr. Robert J male  37
## 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  31
## 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
```

Pero el resultado es que tenemos el mismo número de registros por lo que o bien los valores de las tarifas son correctos para ese 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 es solo el 1% de todos los registros.

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.

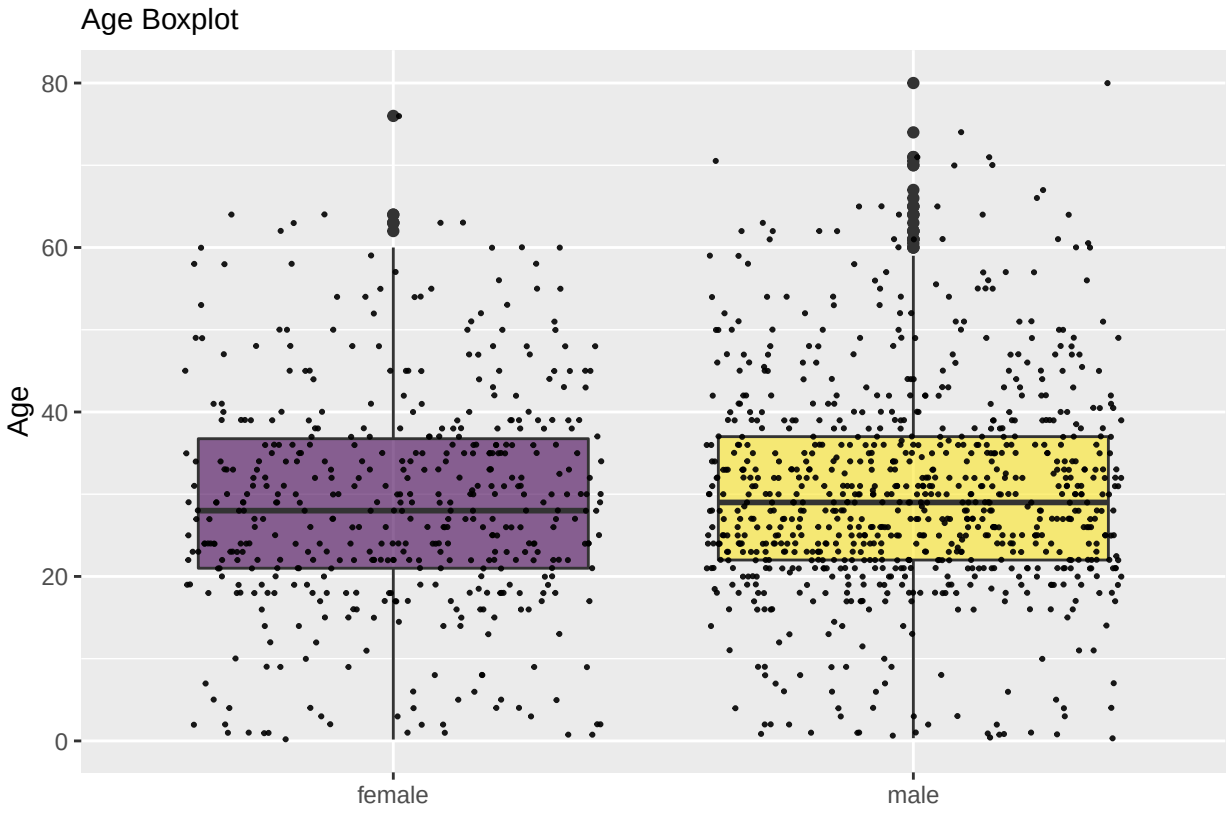
Para la variable Age, los datos perdidos han sido tratados y fuera de eso en la revisión inicial que he realizado no he detectado valores atípicos, ya que todos los datos encontrados corresponden a valores normales que podríamos esperar para edades.

De todos modos hago un gráfico de densidades para comprobar que no hay valores sentinelas u outliers que pudieran salirse de lo esperado y que hubieran pasado desapercibidos. Además me ha resultado interesante

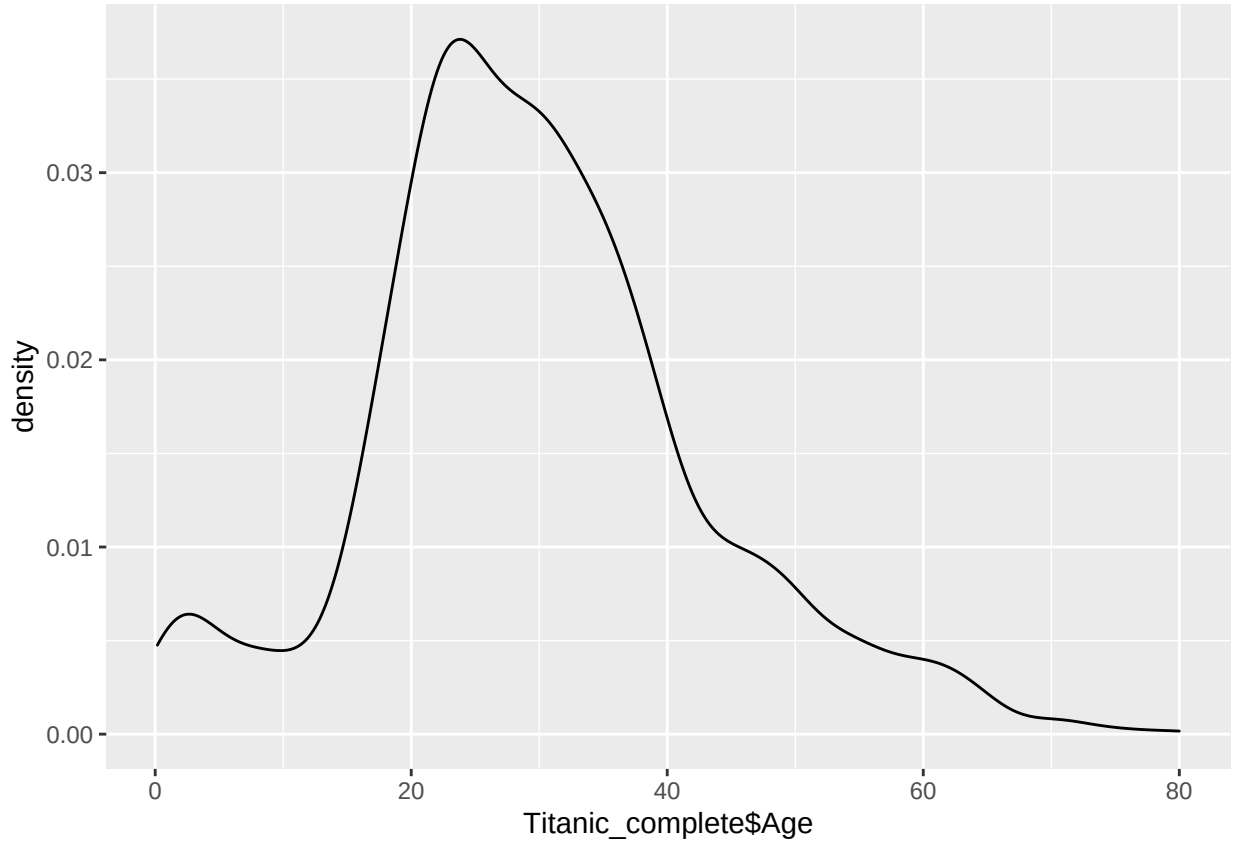
plantear un gráfico boxplot para ver si hay también valores sentinelas u outliers considerando una separación por sexos.

El resultado de los gráfico es el siguiente:

```
Titanic_complete %>%
  ggplot( aes(x=Sex,y=Age, fill=Sex)) +
    geom_boxplot() +
    scale_fill_viridis(discrete = TRUE, alpha=0.6) +
    geom_jitter(color="black", size=0.4, alpha=0.9)+
    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 no hay valores sentinelas u outliers. En cuanto a los boxplots planteados realmente no nos dan demasiada información solo que los hombres tienen una mediana de edad más elevada porque hay hombes mayores que afectan a la distribución (de hasta 80 años), cosa que no ocurre con las mujeres.

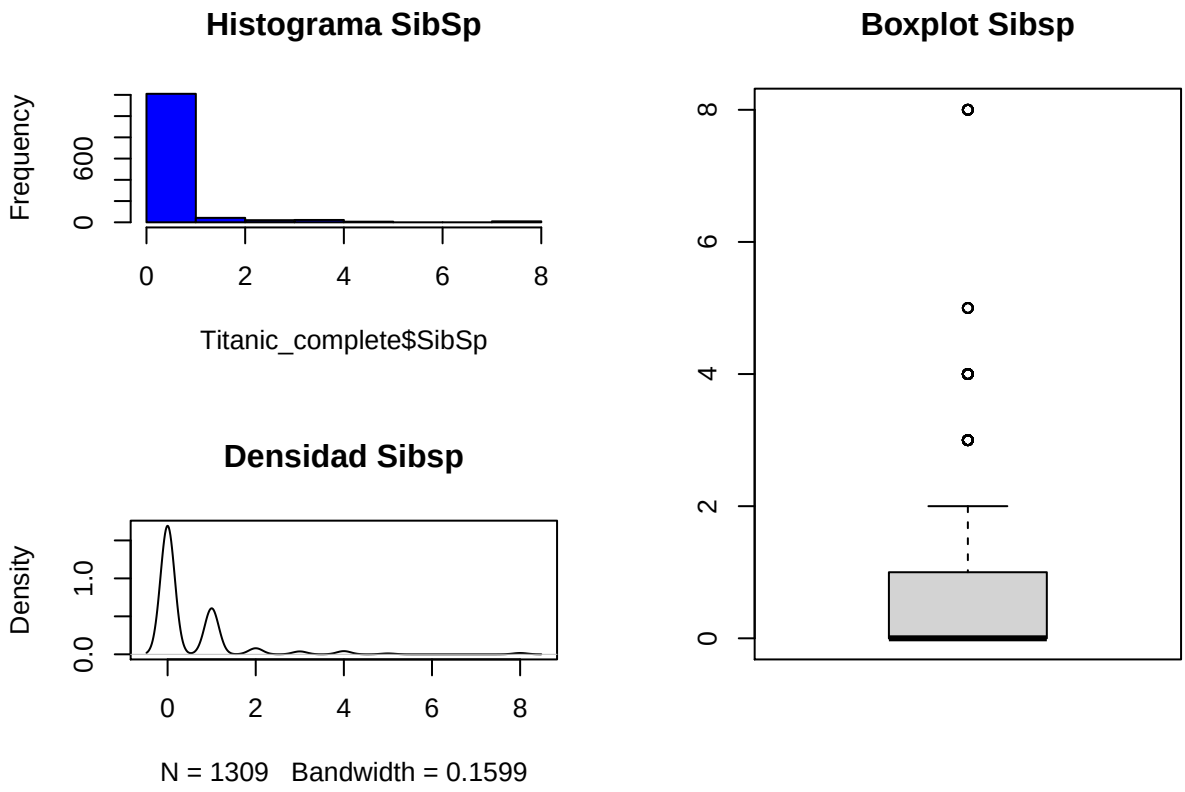
Las siguientes variables que quiero estudiar son SibSp, Parch. He hecho representaciones en una tabla donde he contado el número de registros que tienen cada valor de SibSp o Parch.

Los valores entran dentro de lo que ya esperaba tras haber hecho ya 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 y que no hay valores extremos.

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

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

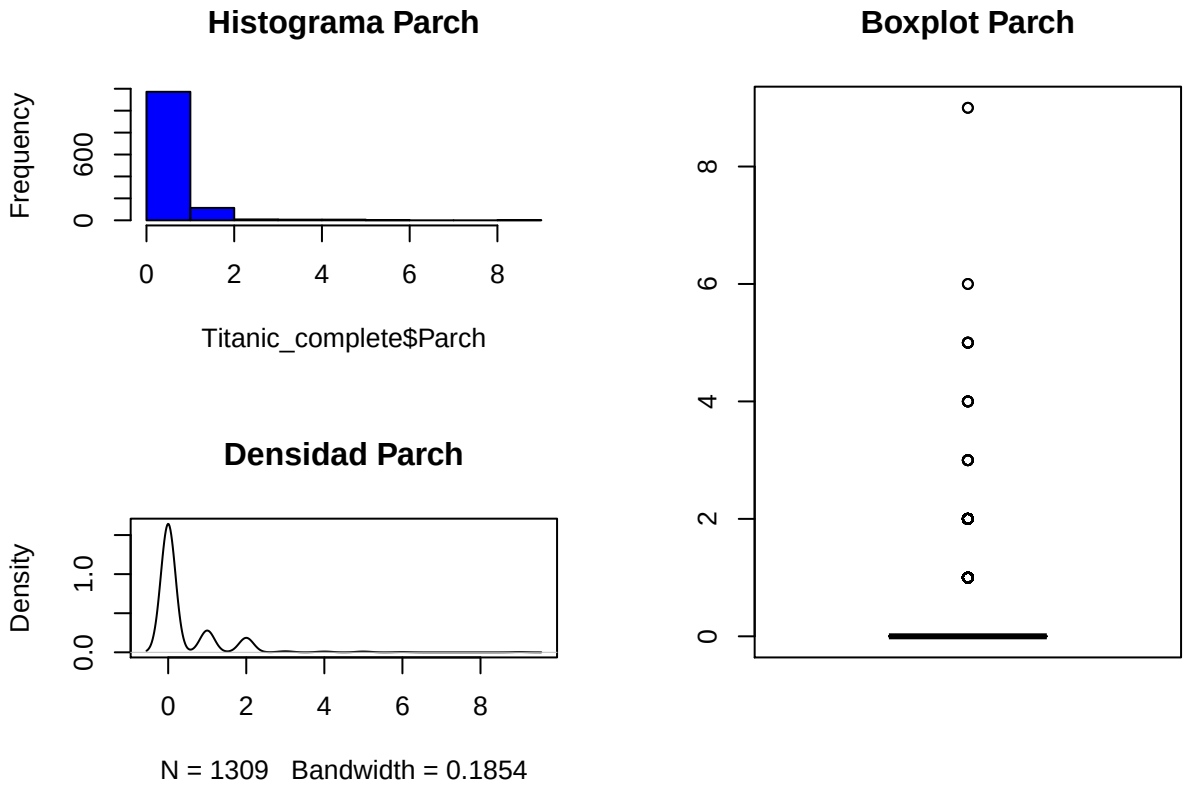
```
layout(matrix(c(1,3,2,3), 2, 2, byrow = TRUE))
hist(Titanic_complete$SibSp ,col="blue",main=" Histograma SibSp ",breaks =10)
plot(density(Titanic_complete$SibSp),main="Densidad Sibsp")
boxplot(Titanic_complete$SibSp ,main="Boxplot Sibsp")
```




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

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

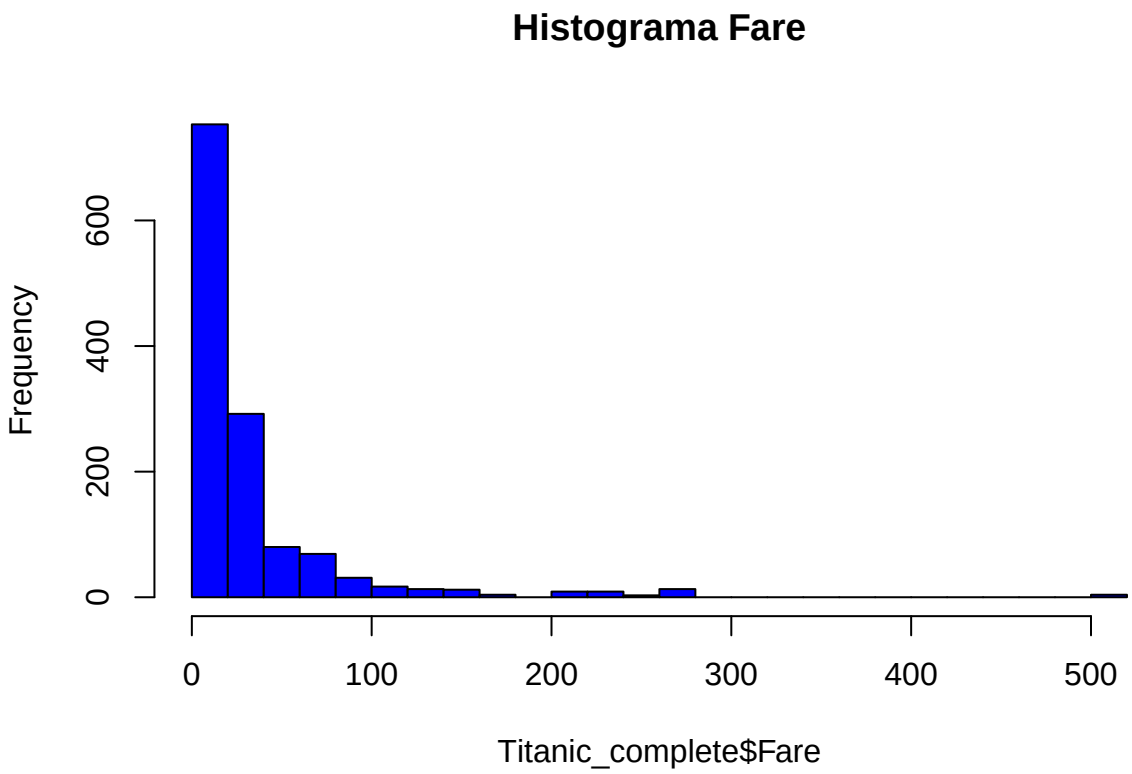
```
layout(matrix(c(1,3,2,3), 2, 2, byrow = TRUE))
hist(Titanic_complete$Parch ,col="blue",main=" Histograma Parch ",breaks =10)
plot(density(Titanic_complete$Parch),main="Densidad Parch")
boxplot(Titanic_complete$Parch ,main="Boxplot Parch")
```



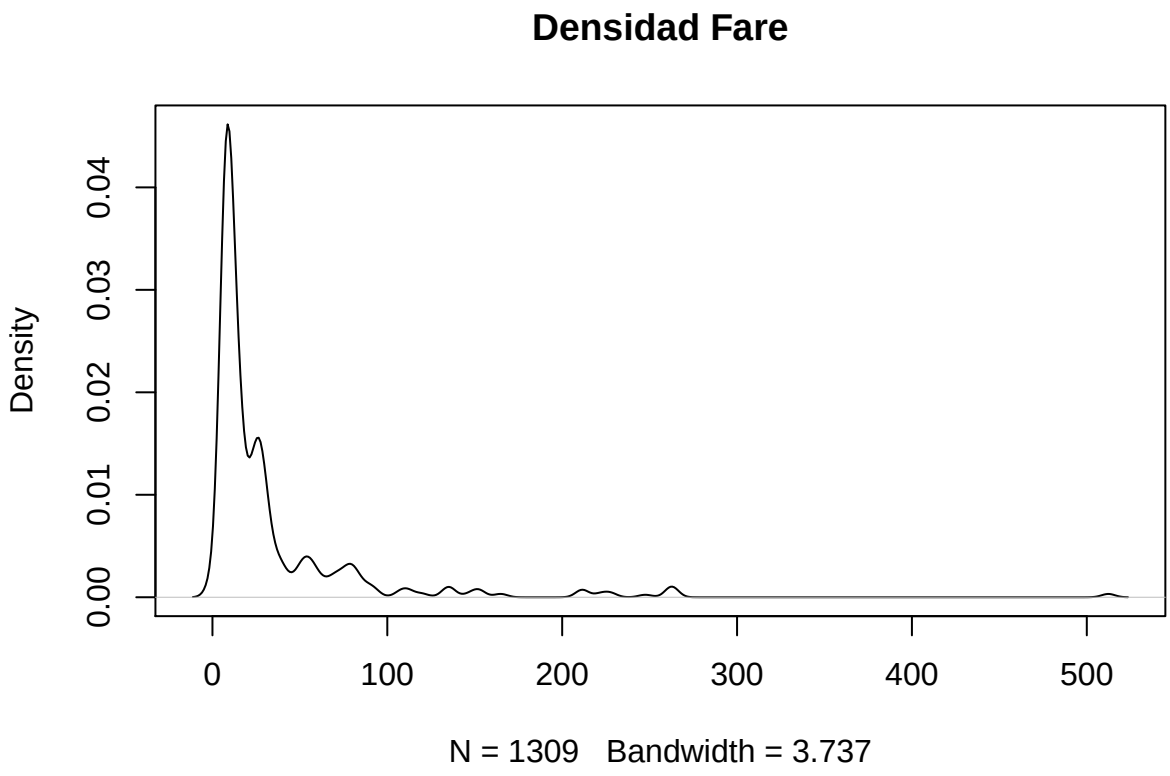
Los gráficos muestran distribuciones similares y concuerdan con lo que he podido comprobar con la función Table.

Finalmente la última variable en la que tenemos que evaluar los outliers es en Fare, igual que en los casos anteriores planteo un boxplot, un histograma y un grafico de densidad

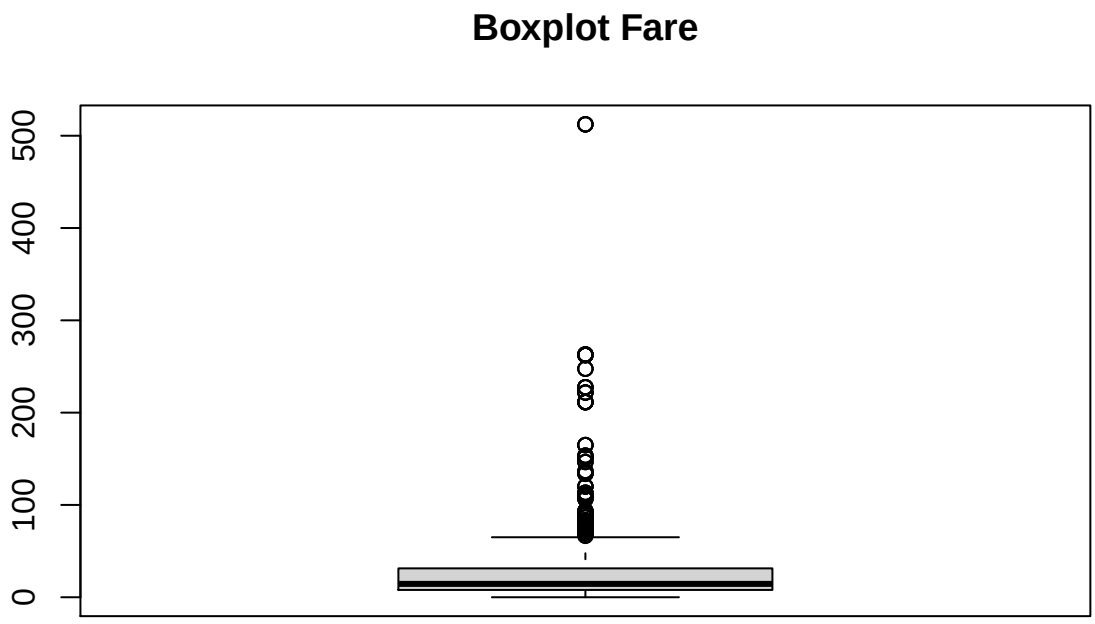
```
#layout(matrix(c(1,1,1,1), 2, 2, byrow = TRUE))
hist(Titanic_complete$Fare ,col="blue",main=" Histograma Fare ",breaks =30)
```




```
plot(density(Titanic_complete$Fare),main="Densidad Fare")
```




```r
boxplot(Titanic_complete$Fare ,main="Boxplot Fare")
```



De el resultado veo que hay valores que en efecto parecen atípicos, concretamente 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
```

Otras comprobaciones en la limpieza de datos.

Fuera deñ 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 modelo.

*Per 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

Otro cambio que considero necesario es discretizar esta variable, esto se puede hacer con la función round por ejemplo o cut. de momento solo usaré la función round

```
#discretización con función round, he preferido  realizarla en una nueva columna (nueva variable que añ

fare_d<-round(Titanic_complete$Fare,0)

#hasta pasar a la fase de planteamiento y analisis del proyecto mantendré ambas variables.
Titanic_complete$Fare.d<-fare_d

# compruebo que se ha creado correctamente la variable
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  29     0     0
##             Ticket    Fare Cabin Embarked Set_type CabinG Fare.d
## 1        A/5 21171  7.2500     N        S    train      N      7
## 2         PC 17599 71.2833   C85        C    train      C     71
## 3 STON/O2. 3101282  7.9250     N        S    train      N      8
## 4           113803 53.1000  C123        S    train      C     53
## 5           373450  8.0500     N        S    train      N      8
## 6           330877  8.4583     N        Q    train      N      8
```

*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)

```
#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 = "dr
  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, ...].
```

```
#elimino la columna residual del nombre que se ha generado al crear dos columnas más una para el apelli
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  29     0     0           330877  8.4583     N        Q    train      N
##   Fare.d
## 1      7
## 2     71
## 3      8
## 4     53
## 5      8
## 6      8
```

```
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 Fare.d
## 760 female  33     0     0 110152 86.5   B77        S    train      B     86
```

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_other” (que incluye trabajos de altas categorias sociales como reverendos o doctores o titulos heredados que sin ser nobles otorga una categoría de “elite”, como la de maestro), Miss, Mr y Mrs (no he agrupado estos atributos porque nos puede aportar información sobre mujeres casadas o solteras y hombres que no tienen ningún título ni profesiones que se puedan considerar d e“elite”), por último la categoría final sería “Noble” en este grupo se consideran a todos los pasajeros con títulos nobiliarios.

```
#agrupación de categorías.

Titanic_complete$Title[Titanic_complete$Title %in% c('Capt','Col', 'Dr','Major','Rev','Master')] <-"eli
Titanic_complete$Title[Titanic_complete$Title %in% c('Miss', 'Ms', 'Mlle')]<- 'Miss'
Titanic_complete$Title[Titanic_complete$Title %in% c('Mme')] <-'Mrs'
Titanic_complete$Title[Titanic_complete$Title %in% c('Don','Jonkheer', 'Sir','Lady','Dona','th')] <-'No

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

table(Titanic_complete$Title)
```

```
##
## elite_other        Miss          Mr         Mrs       Noble
##          84         264         757         198           6
```

- 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.

```
#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    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  29     0     0           330877  8.4583     N        Q    train      N
##   Fare.d Family.unit
## 1      7           2
## 2     71           2
## 3      8           1
## 4     53           2
## 5      8           1
## 6      8           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  18 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/ 5 levels "elite_other",..: 3 4 2 4 3 3 3 1 4 4 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 29 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 ...
##  $ Fare.d     : num  7 71 8 53 8 8 52 21 11 30 ...
##  $ 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
##  elite_other: 84   female:466   Min.   : 0.17   Min.   :0.0000
##  Miss       :264   male  :843   1st Qu.:22.00   1st Qu.:0.0000
##  Mr         :757                Median :29.00   Median :0.0000
##  Mrs        :198                Mean   :29.91   Mean   :0.4989
##  Noble      :  6                3rd Qu.:37.00   3rd Qu.:1.0000
##                                 Max.   :80.00   Max.   :8.0000
##
##      Parch            Ticket          Fare                     Cabin
##  Min.   :0.000   CA. 2343:  11   Min.   :  0.000   N              :1014
##  1st Qu.:0.000   1601    :   8   1st Qu.:  7.896   C23 C25 C27    :   6
##  Median :0.000   CA 2144 :   8   Median : 14.454   B57 B59 B63 B66:   5
##  Mean   :0.385   3101295 :   7   Mean   : 33.281   G6             :   5
##  3rd Qu.:0.000   347077  :   7   3rd Qu.: 31.275   B96 B98        :   4
##  Max.   :9.000   347082  :   7   Max.   :512.329   C22 C26        :   4
##                  (Other) :1261                     (Other)        : 271
##  Embarked  Set_type       CabinG         Fare.d       Family.unit
##   :  0    test :418   N      :1014   Min.   :  0.0   Min.   : 1.000
##  C:270    train:891   C      :  94   1st Qu.:  8.0   1st Qu.: 1.000
##  Q:123                B      :  65   Median : 14.0   Median : 1.000
##  S:916                D      :  46   Mean   : 33.3   Mean   : 1.884
##                       E      :  41   3rd Qu.: 31.0   3rd Qu.: 2.000
##                       A      :  22   Max.   :512.0   Max.   :11.000
##                       (Other):  27
```

El resultado es que tengo muchas 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.

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)* Fare.d *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)
  - Para variables categóricas (gráficos de barras)
- Decisión sobre las variables finales 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?¿las famílias numerosas tenían menos posibilidades de salvar a alguno de sus miembros?¿la ubicación de las cabinas en relación al punto de colapso del barco nos indica cuál es la mejor cubierta para sobrevivir al accidente?

- Decisión sobre las variables finales a utilizar en el análisis

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.numeric<-c(8,9,10,17,18)

var.factor<-colnames(Titanic_complete)[id.factor]
var.numeric<-colnames(Titanic_complete)[id.numeric]

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

```
##   Survived Pclass   Surname Title    Sex Embarked CabinG
## 1        0      3    Braund    Mr   male        S      N
## 2        1      1   Cumings   Mrs female        C      C
```

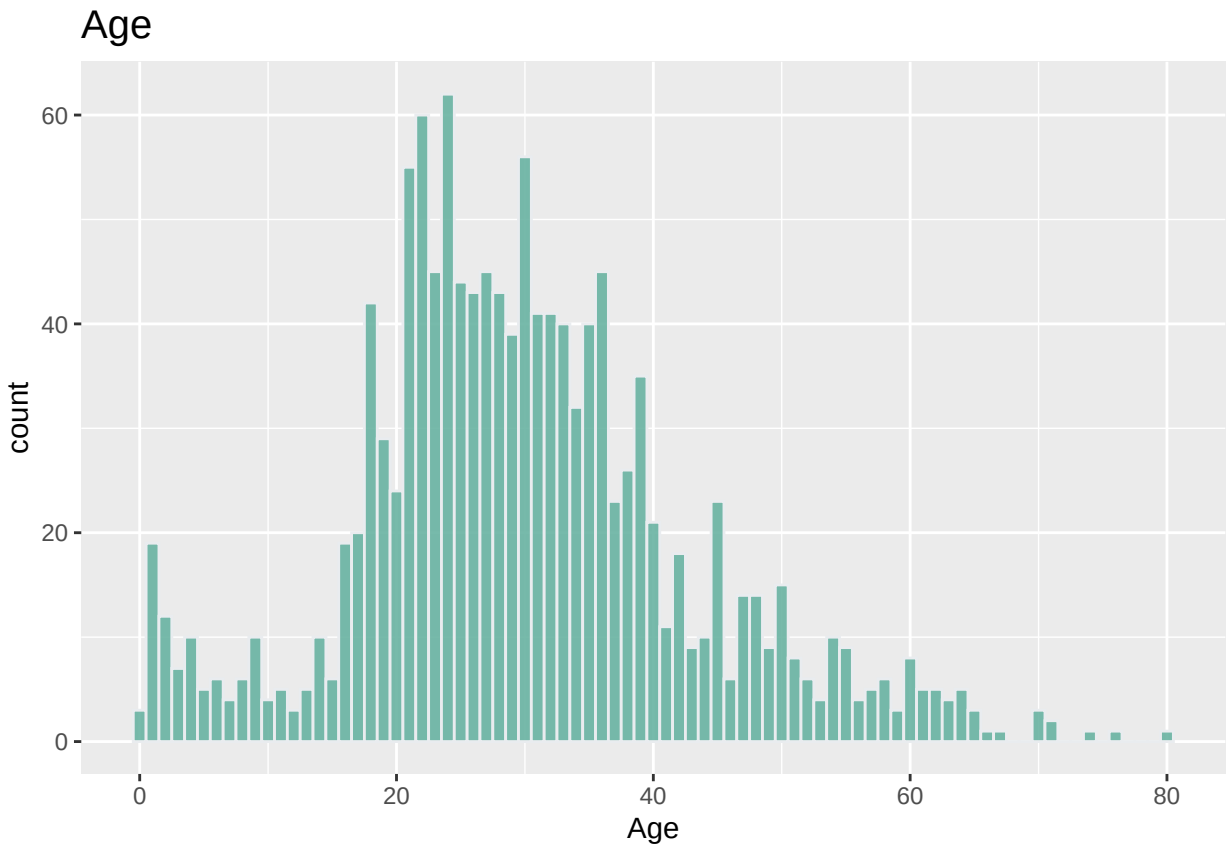
```
## 3        1      3 Heikkinen  Miss female        S      N
## 4        1      1  Futrelle   Mrs female        S      C
## 5        0      3     Allen    Mr   male        S      N
## 6        0      3     Moran    Mr   male        Q      N
```

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

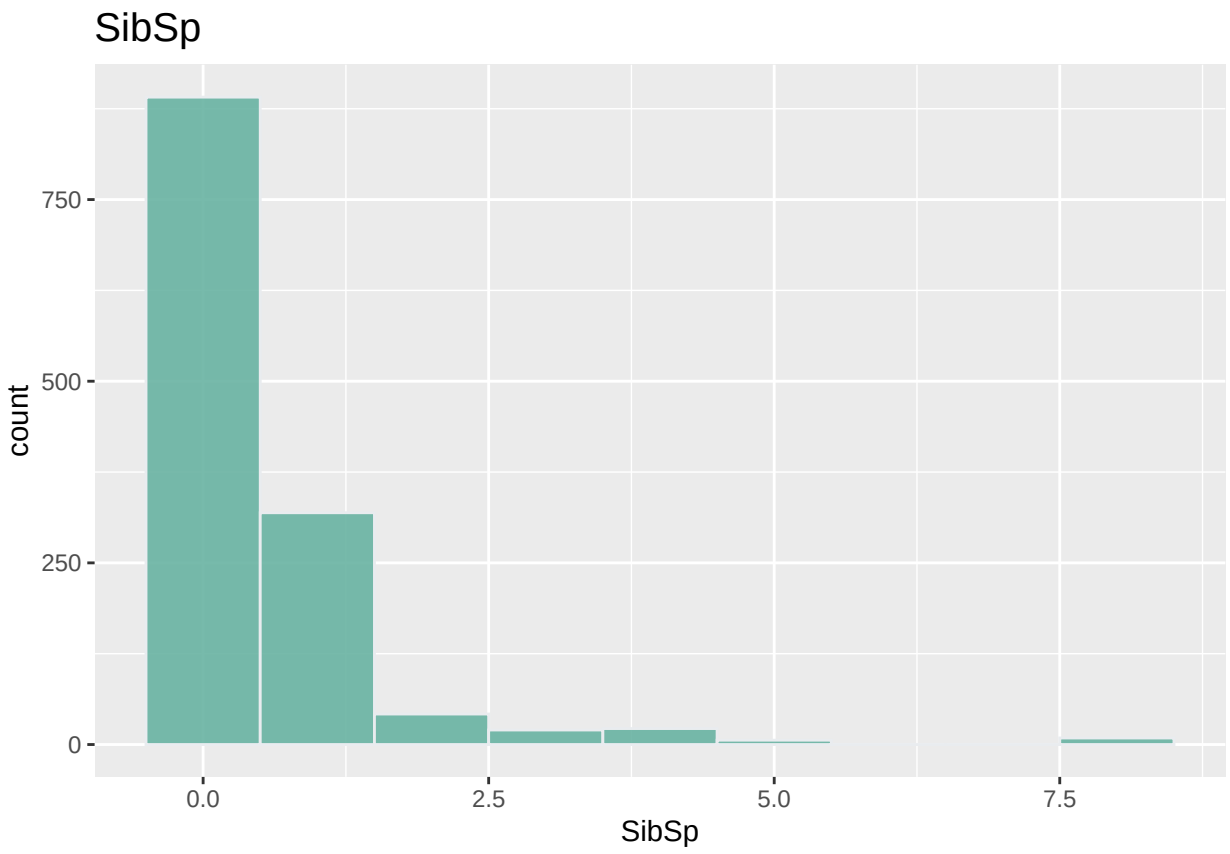
```
##   Age SibSp Parch Fare.d Family.unit
## 1  22     1     0      7           2
## 2  38     1     0     71           2
## 3  26     0     0      8           1
## 4  35     1     0     53           2
## 5  35     0     0      8           1
## 6  29     0     0      8           1
```

```
#pensar implementar gráficos en bucle
```

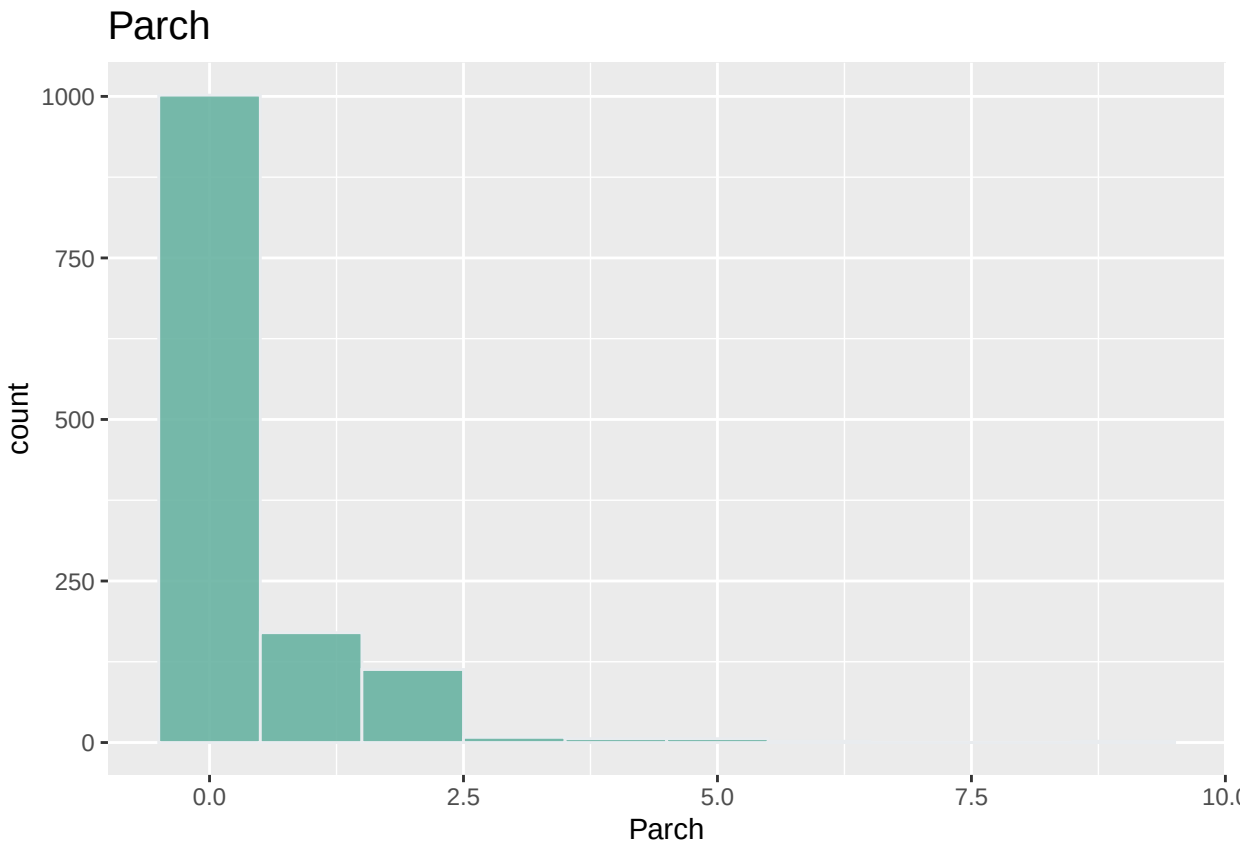
```
#Histogramas para variables numéricas(cuantitativas), esta revisión nos servirá también para evaluar la

ggplot(Titanic_complete,aes(x=Age)) + geom_histogram( binwidth=1, fill="#69b3a2", color="#e9ecef", alpha
```



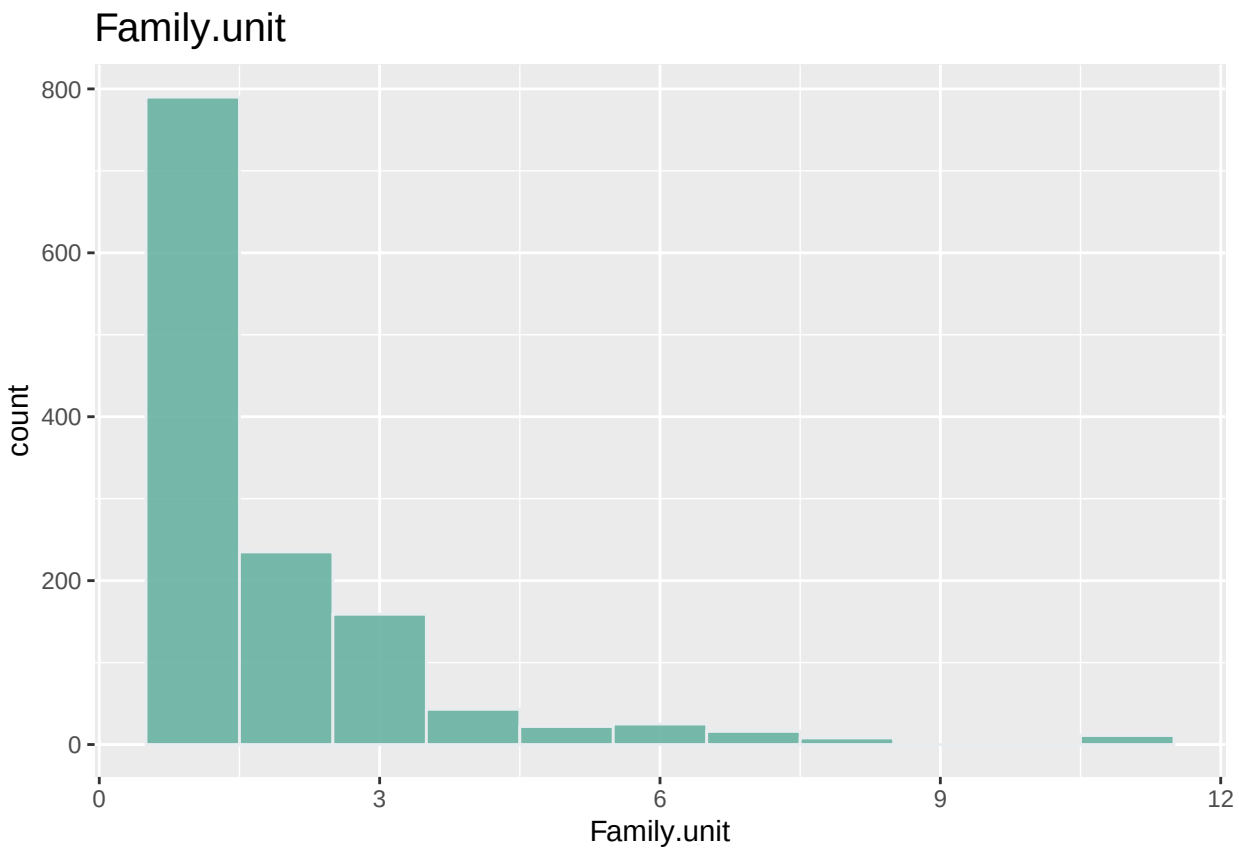

```
ggplot(Titanic_complete,aes(x=SibSp)) + geom_histogram( binwidth=1, fill="#69b3a2", color="#e9ecef", al
```



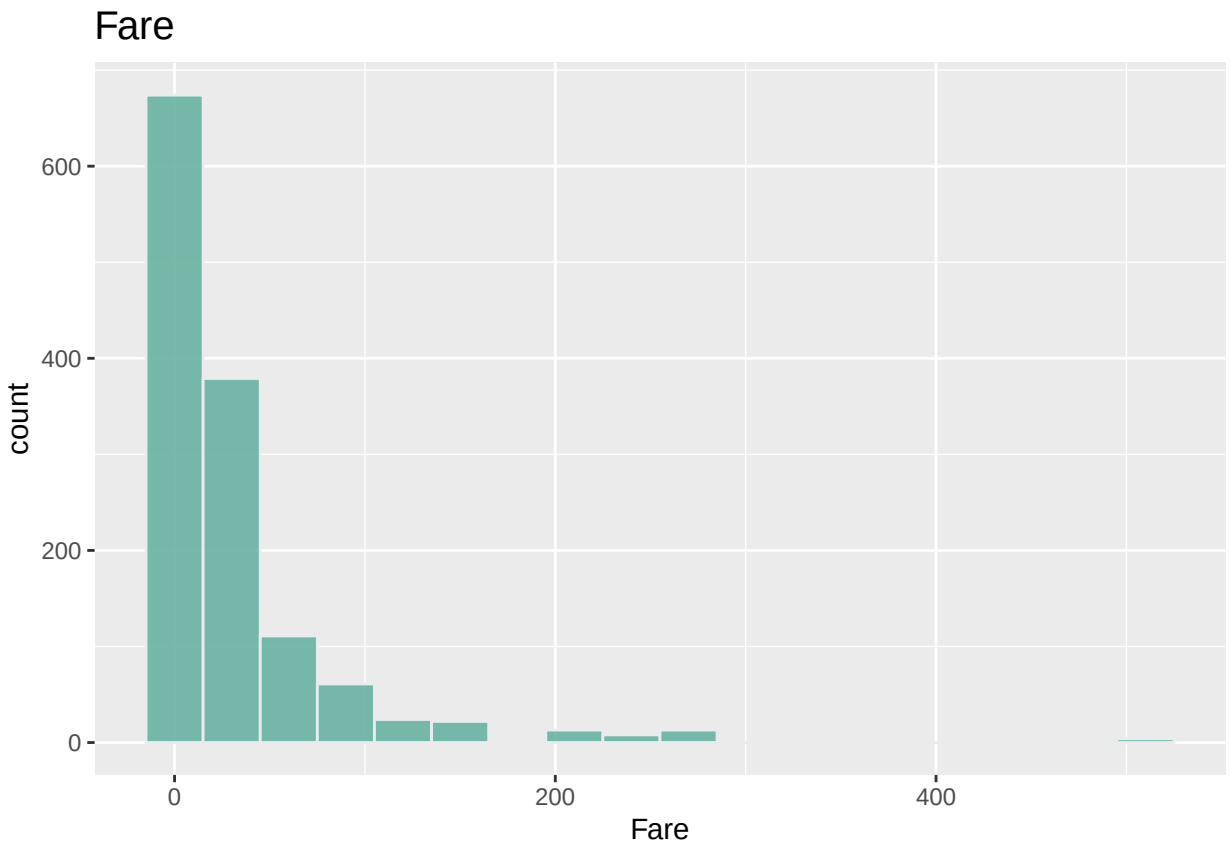

```
ggplot(Titanic_complete,aes(x=Parch)) + geom_histogram( binwidth=1, fill="#69b3a2", color="#e9ecef", al
```




```
ggplot(Titanic_complete,aes(x=Family.unit)) + geom_histogram( binwidth=1, fill="#69b3a2", color="#e9ece
```

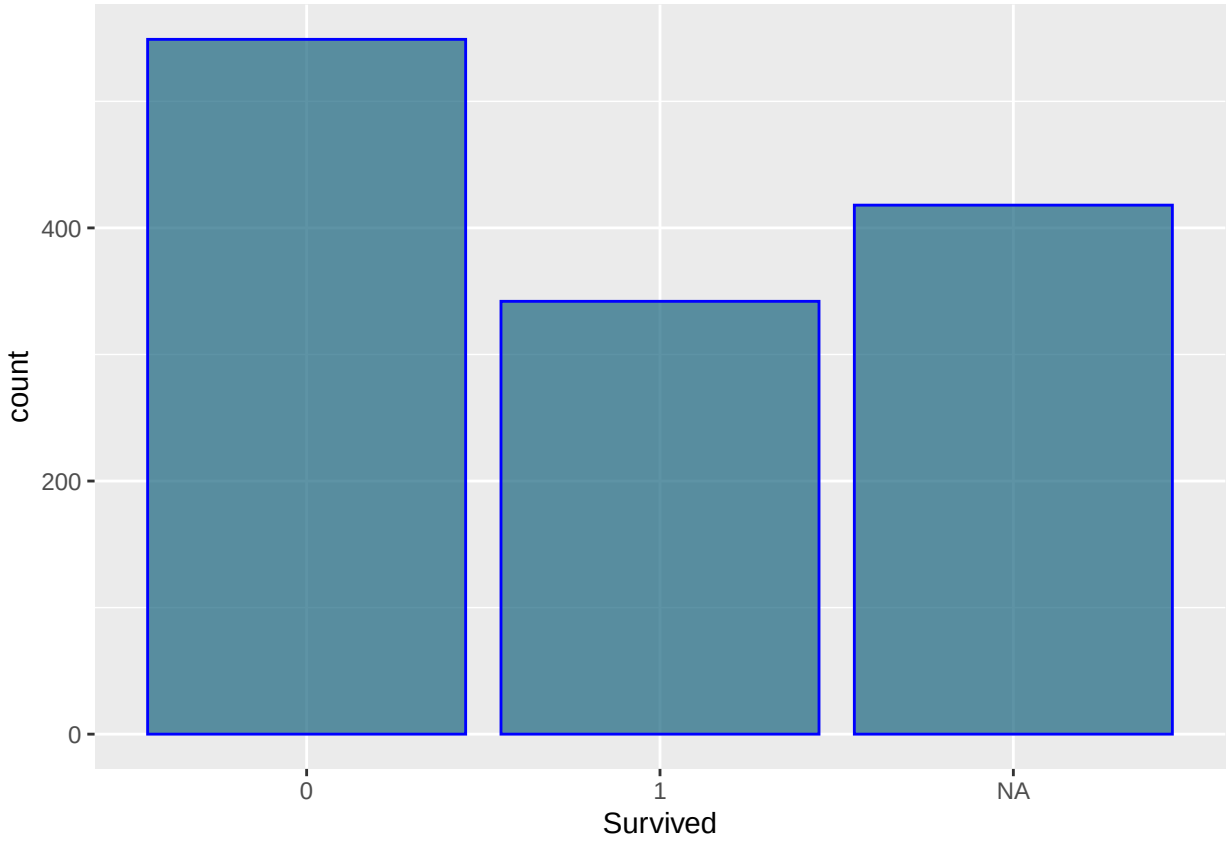



```
ggplot(Titanic_complete,aes(x=Fare)) + geom_histogram( binwidth=30, fill="#69b3a2", color="#e9ecef", al
```

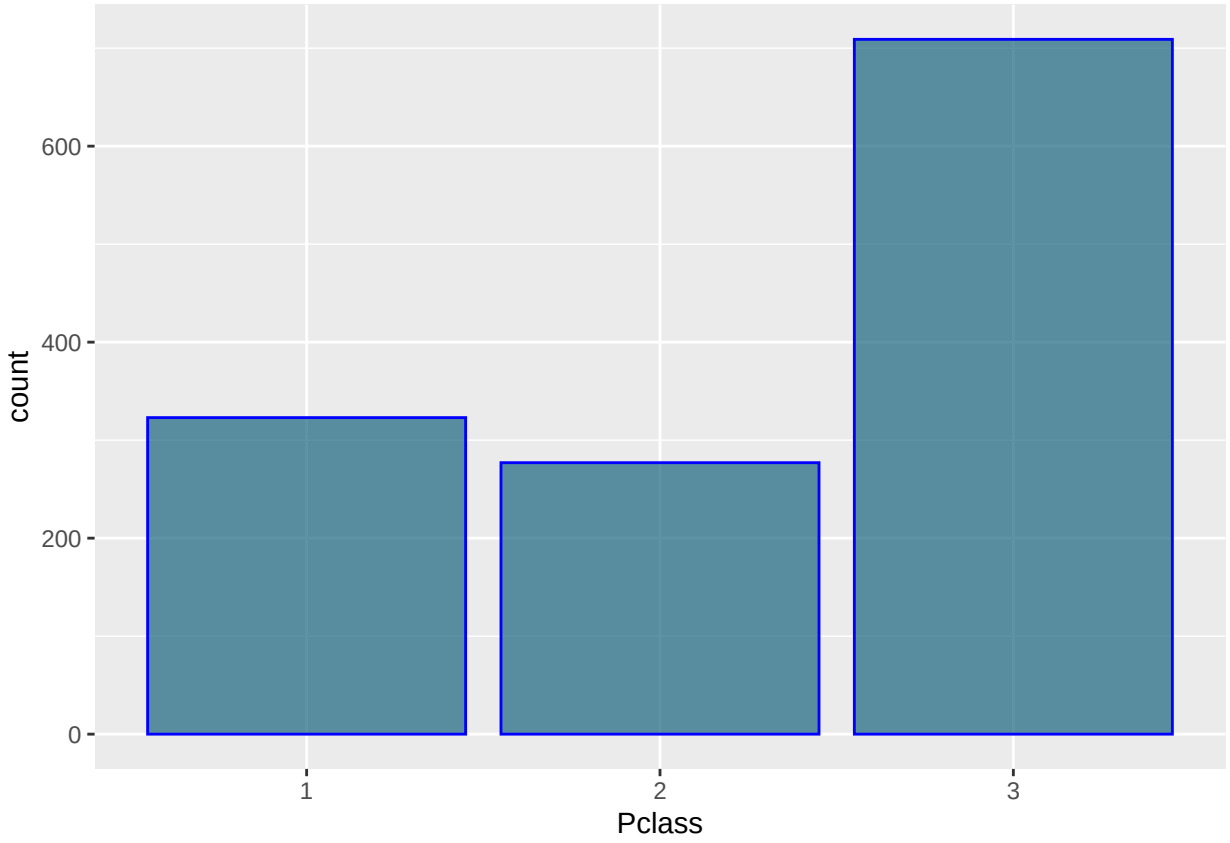


De las variables numéricas podemos ver que la única que tiene una distribución similar a la normal es la variable Age. Por los gráficos que obtenemos de las demás variables podemos tal vez considerar normalizarlas, y ver si con eso se aproximan más a una distreibución normal creo que esto es especialmente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

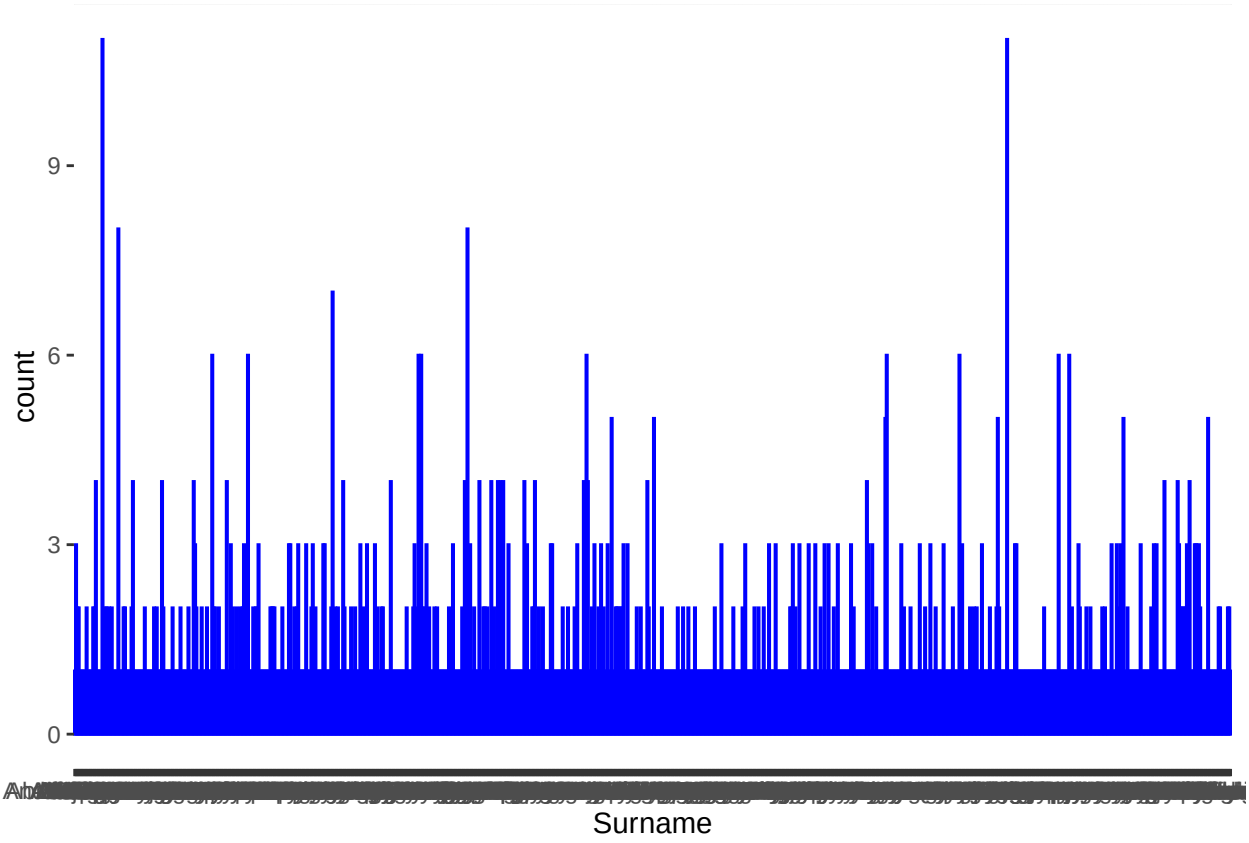
```
#Barplot per variables categóricas
ggplot(Titanic_complete, aes(x=Survived))+geom_bar(color="blue",fill=rgb(0.1,0.4,0.5,0.7))
```



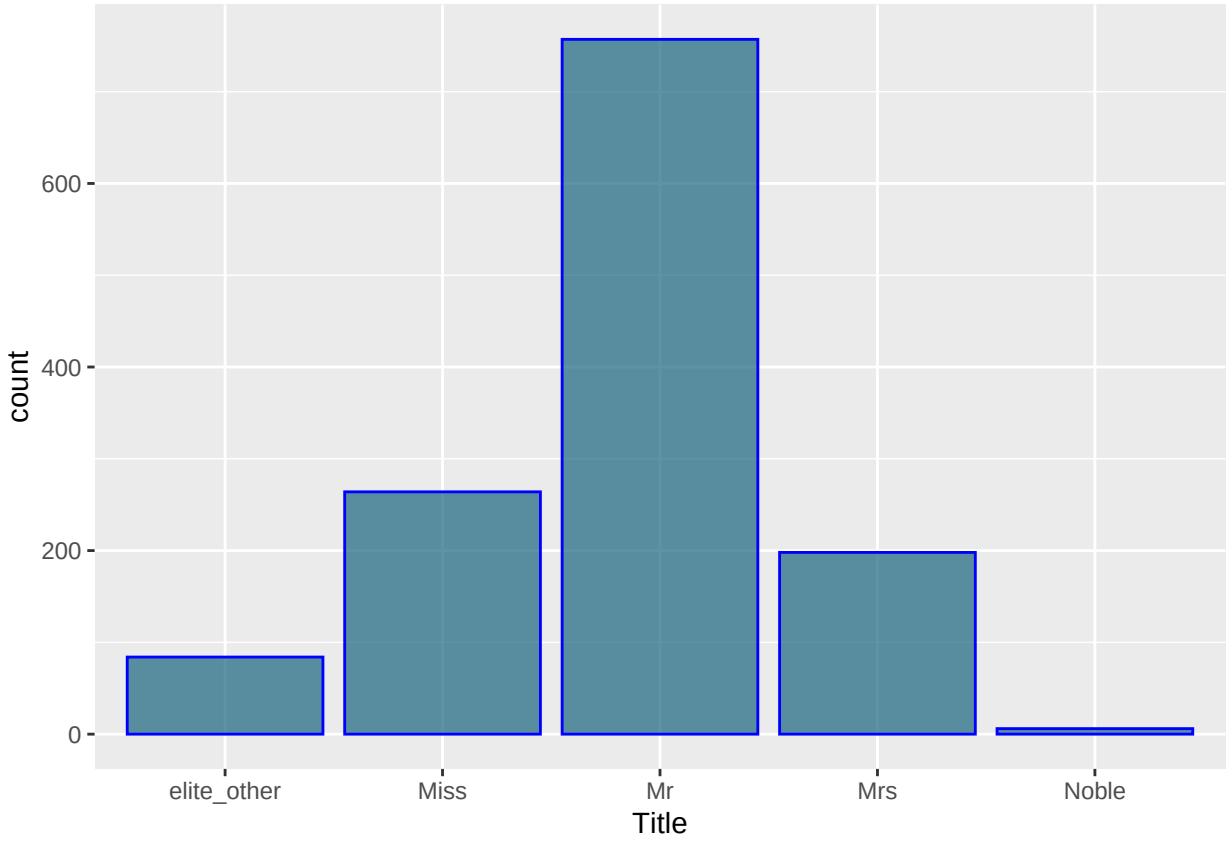

```
ggplot(Titanic_complete, aes(x=Pclass))+geom_bar(color="blue",fill=rgb(0.1,0.4,0.5,0.7))
```



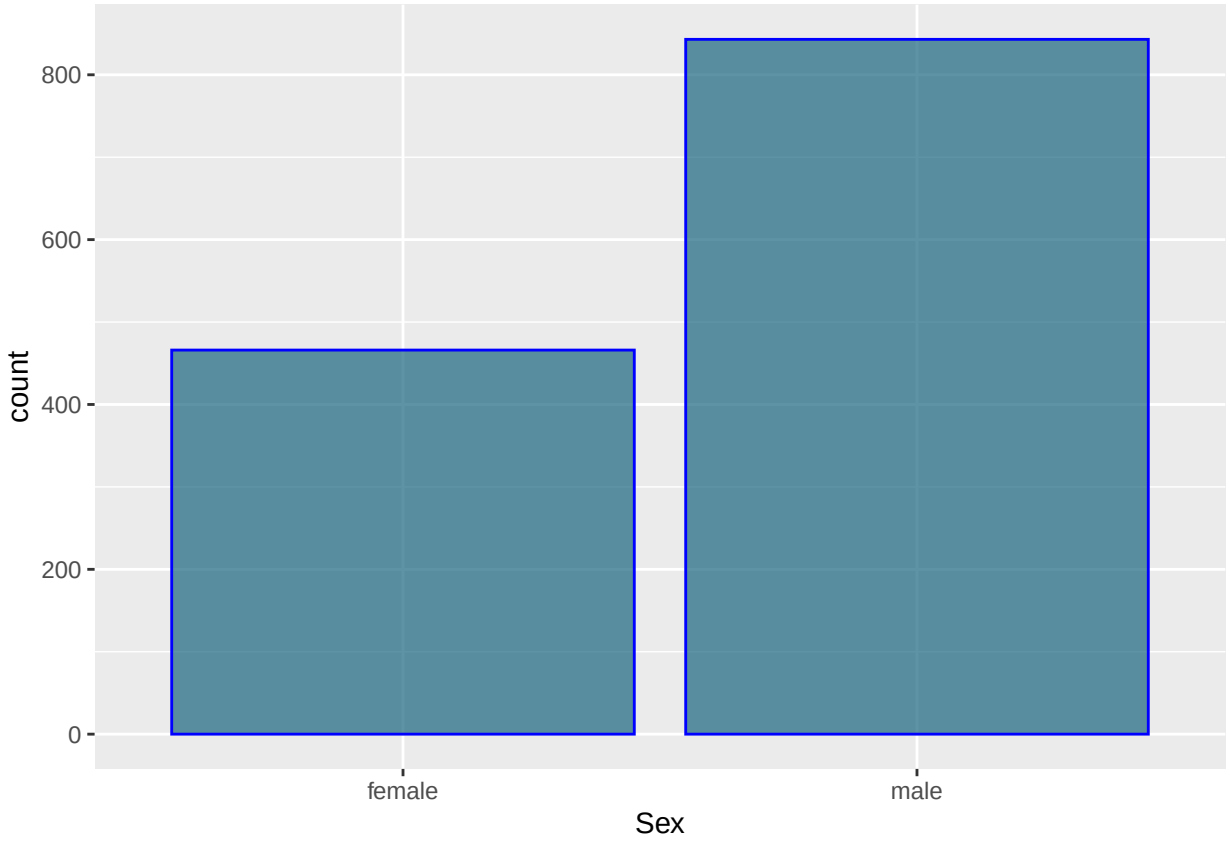

```
ggplot(Titanic_complete, aes(x=Surname))+geom_bar(color="blue",fill=rgb(0.1,0.4,0.5,0.7))
```



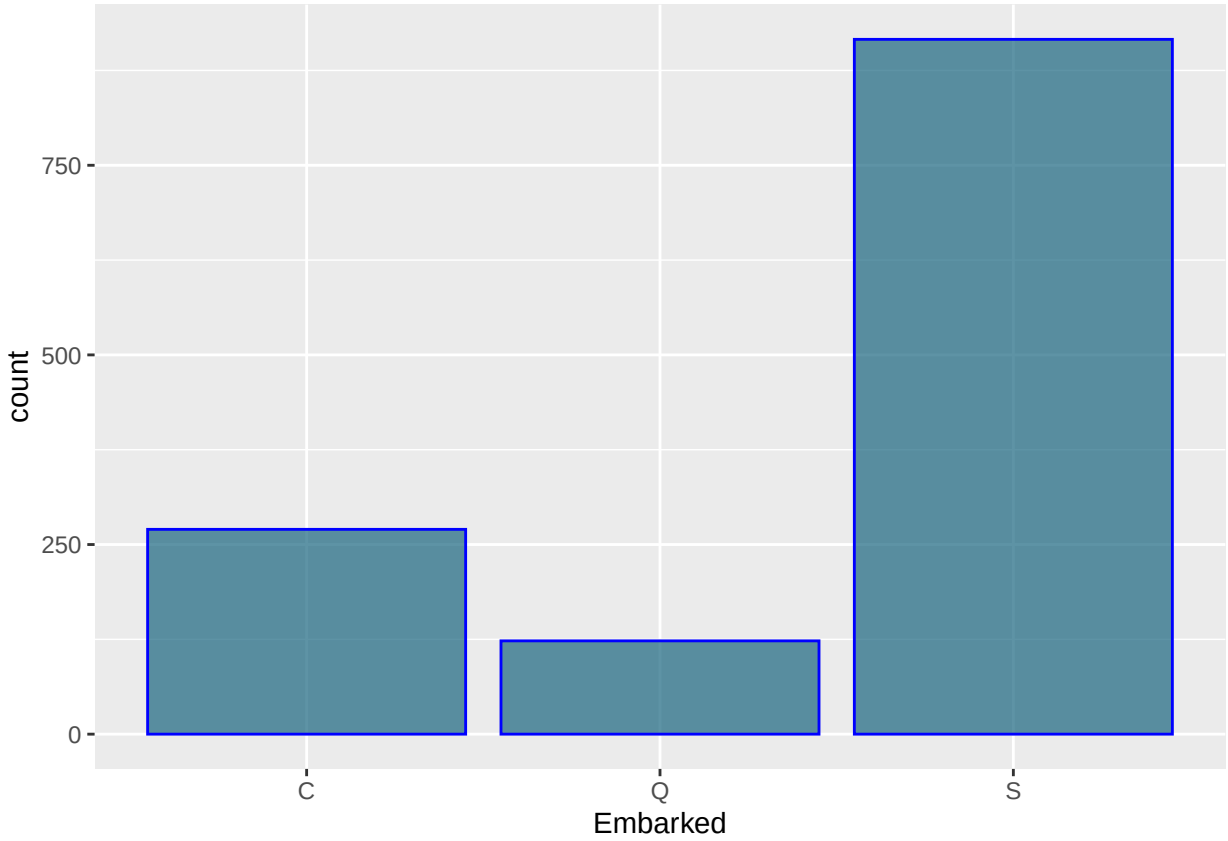

```
ggplot(Titanic_complete, aes(x=Title))+geom_bar(color="blue",fill=rgb(0.1,0.4,0.5,0.7))
```



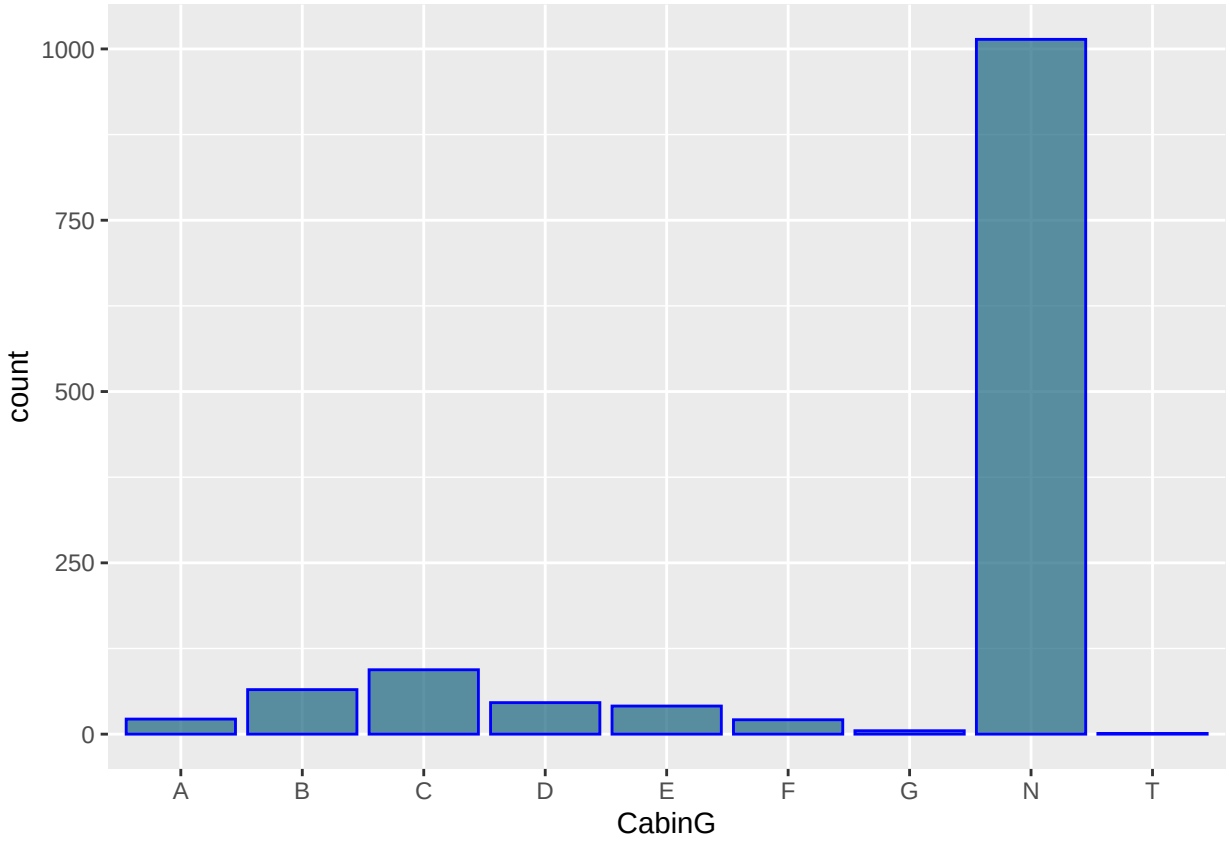

```
ggplot(Titanic_complete, aes(x=Sex))+geom_bar(color="blue",fill=rgb(0.1,0.4,0.5,0.7))
```



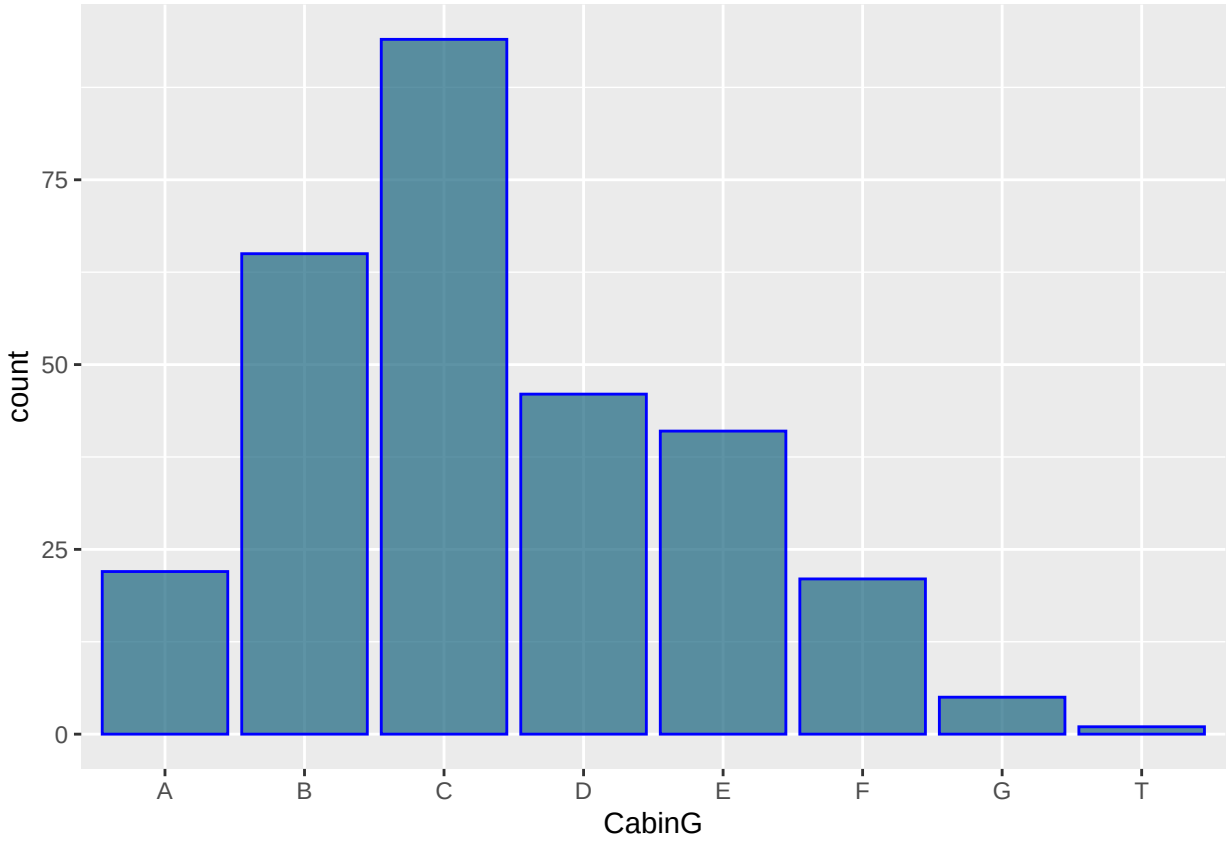

```
ggplot(Titanic_complete, aes(x=Embarked))+geom_bar(color="blue",fill=rgb(0.1,0.4,0.5,0.7))
```




```
ggplot(Titanic_complete,aes(x=CabinG))+geom_bar(color="blue",fill=rgb(0.1,0.4,0.5,0.7))
```




```
ggplot(Titanic_complete[Titanic_complete$CabinG!="N",],aes(x=CabinG))+geom_bar(color="blue",fill=rgb(0.
```




```
#pensar plantear for loop?
```

4.2 Comprobación de la normalidad i homogeneidad de la variancia.

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

En funció de les dades i de l'objectiu de l'estudi, aplicar proves de contrast d'hipòtesis, correlacions, regressions, etc. Aplicar almenys tres mètodes d'anàlisi diferents

```
#Separación de dataset Train para analisis

Train.f<-Titanic_complete%>%
  select(PassengerId,Survived,Pclass,Title,Sex,Age,SibSp,Parch,Embarked,Set_type,CabinG,Fare.d,Family.u
  filter(Set_type=="train")
str(Train.f)
```

```
## 'data.frame':    891 obs. of  13 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 1 2 2 ...
##  $ Pclass     : Factor w/ 3 levels "1","2","3": 3 1 3 1 3 3 1 3 3 2 ...
##  $ Title      : Factor w/ 5 levels "elite_other",..: 3 4 2 4 3 3 3 1 4 4 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 29 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 ...
##  $ 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 ...
##  $ Fare.d     : num  7 71 8 53 8 8 52 21 11 30 ...
##  $ Family.unit: num  2 2 1 2 1 1 1 5 3 2 ...
```

El número de registros y de variables coinciden con lo que esperabamos además la interpretación (variable numérica/factor) es también la que esperabamos.

hist(Titanic_complete[Titanic_complete$Pclass==2,“Fare”], breaks=30, xlim=c(0,20), col=rgb(1,0,0,0.5), xlab=“Fare”, ylab=“frequency”, main=“Fare payed class1”) hist(Titanic_complete[Titanic_complete$Pclass==3,“Fare”], breaks=30, xlim=c(0,20), col=rgb(0,0,1,0.5),add=T) #hist(Titanic_complete[Titanic_complete$Pclass==3,“Fare”], breaks=30, xlim=c(0,300), col=rgb(0,1,0,0.5), add=T) legend(“topright”, legend=c(“Ixos”,“Primadur”), col=c(rgb(1,0,0,0.5), rgb(0,0,1,0.5)), pt.cex=2, pch=15)

El tratamiento de variables numéricas zeros o missing values, será diferente del tratamiento de las variables categóricas missing, por lo que el siguiente paso que haré es separar las variables por numéricas y categóricas y haré representaciones gráficas que me ayuden a tener más información y poder enfocar mejor el estudio.

4.1. Selecció dels grups de dades que es volen analitzar/comparar (planificació dels anàlisis a aplicar). 4.2. Comprovació de la normalitat i homogeneïtat de la variància.

4.3. Aplicació de proves estadístiques per comparar els grups de dades. En funció de les dades i de l’objectiu de l’estudi, aplicar proves de contrast d’hipòtesis,correlacions, regressions, etc. Aplicar almenys tres mètodes d’anàlisi diferents. 5. Representació dels resultats a partir de taules i gràfiques.