PRAC2: Limpieza y análisis de datos

Xiaolin Ye, Javier Galan

Mayo 2022

Índice

---

1. Descripción del dataset

El dataset elegidos proviene de Kaggle, describe en él la información de los pasajeros de Titanic (género, edad...), las condiciones de embargue (tipo de clase, cabina..) y si sobrevivió el accidente.

El principal objetivo para analizar estos datos es la predicción de la supervivencia de los pasajeros, para ello nos facilita un juego de datos de entrenamiento "train.csv" como la base para el aprendizaje supervisado y otro juego de "test.csv" que sirve para evaluar la precisión de la predicción.

Esta tarea también forma parte de una competición que celebra Kaggle donde los participantes suben sus trabajos y compiten por ser la mejor implementación que resuelve la pregunta clave:

¿Qué tipo de pasajero tiene mayor probabilidad de sobrevivir?

##Lectura de los datos:

Para crear nuestro modelo, primero leemos los datos de entrenamiento.

```
titanic_train <-read.csv("train.csv",header=T,sep=",")
```

```
titanic_test <-read.csv("test.csv",header=T,sep=",")
```

Hacemos un breve análisis de los datos ya que nos interesa tener una idea general de los datos que disponemos.
Exploración de la base de datos

Primero calcularemos las dimensiones de nuestra base de datos y analizaremos qué tipos de atributos tenemos.

Calculamos las dimensiones de la base de datos mediante la función dim(). Obtenemos que disponemos de 891 registros o pasajeros (filas) y 12 variables (columnas) para el juego de entrenamiento, 418 registros y 11 variables para el juego de validación.

```
dim(titanic_train)
```

```
## [1] 891  12
```

```
dim(titanic_test)
```

```
## [1] 418  11
```

¿Cuáles son esas variables?

```
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" ...
```

Gracias a la función str() sabemos si las varibales son categoricas o numéricas:

- *PassengerId*: int. variable cuantitativo.
  Numéro entero ordinal que identifica a cada uno de los pasajeros.
- *Survived* : int. variable cualitativo.
  Código numérico que identifica el estado de supervivencia de los pasajeros tras el accidente
- *Pclass* : int. variable cuantitativo.
  Clase del ticket que posee el pasajero, 1 = primera, 2 = segunda y 3 = tercera
- *Name* : chr. variable cualitativo.
  Nombre del pasajero
- *Sex* : chr. variable cualitativo.
  Género del pasajero
- *Age* : num. variable cuantitativo.
  Edad del pasajero
- *SibSp* : int. variable cuantitativo.
  Numéro de hermanos/pareja a bordo
- *Parch* : int. variable cuantitativo.
  Numéro de padres/hijos a bordo
- *Ticket* : chr. vairable cualitativo
  Identificador del ticket
- *Fare* : num. variable cuantitativo.
  Precio del ticket
- *Cabin* : chr. variable cualitativo.
  Identificador de la cabina
- *Embarked* : chr. variable cualitativo.
  Puerto de embargue. C = Cherbourg, Q = Queenstown, S = Southampton

---

2. Limpieza de datos

Para poder generar un modelo preciso y adecuado, es de vital importancia seleccionar, limpiar y transformar los datos en el caso de que sea necesario.

Empezamos este apartado seleccionando aquellos atributos/registros que nos van a ser útiles a la hora de construir el modelo.

2.1. Selección de datos

2.1.1. Registros duplicados

Comprobamos que no hay registros duplicados que eliminar

```
sum(duplicated(titanic_train))
```

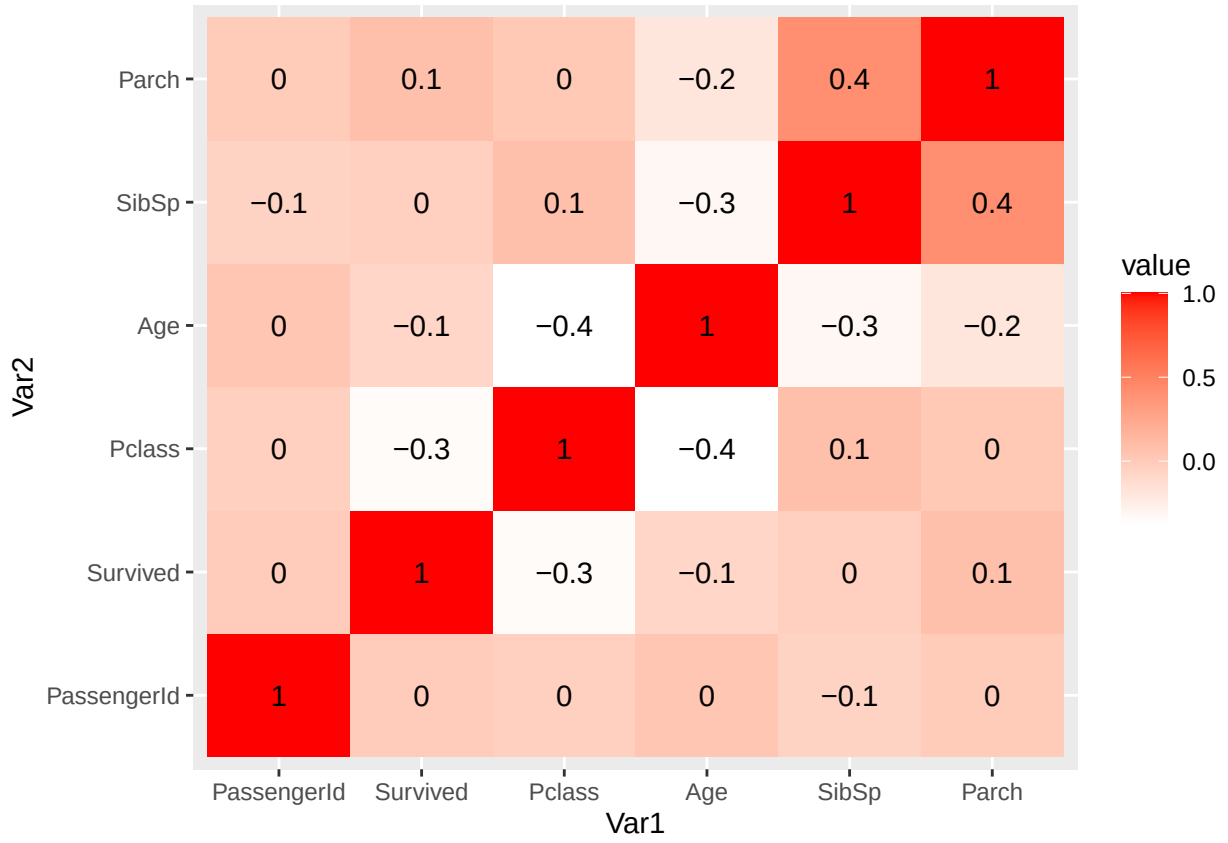
```
## [1] 0
```

2.1.2. correlación entre variables

2.1.2.1. variables cuantitativas Para poder identificar qué variables cuantitativas podemos descartar, podemos calcular la correlación entre las variables.

Por un lado, podes identificar y descartar las variables que no contribuyen apenas a la variable objetivo, por otro lado podemos reconocer variables predictivos que están fuertemente relacionados y usar solo una de ellas.

```
numeric_cor <- cor(titanic_train[,c('PassengerId','Survived','Pclass', 'Age', 'SibSp', 'Parch')],
    titanic_train[,c('PassengerId','Survived','Pclass', 'Age', 'SibSp', 'Parch')], method = 'pearson',us
melted_cormat <- melt(numeric_cor)
ggplot(data = melted_cormat, aes(x=Var1, y=Var2)) +
  geom_tile(aes(fill = value))+
  geom_text(aes(label = round(value,1 ))) +
  scale_fill_gradient(low = "white", high = "red")
```



De aquí podemos observar que, a pesar de que PassengerId es un número, es meramente un identificador ordinal que no aporta información particular de los pasajeros y por lo tanto, afecta de forma escasa en todas las demás variables.

Por otro lado, se observa algo de correlación entre las variables SibSp y Parch, que por nuestra intuición, sabemos que una persona con pareja es más probable tener hijos por ejemplo. No obstante, como no se relacionan igual a la variable target, no podemos descartar ninguna de ellas.

2.1.2.2. variables cualitativas Sabemos que el nombre de una persona viene de fuentes variadas y suele condicionar en los sucesos cotidianos de una forma determinista, por lo que no la incluiremos en nuestro modelo.

De igual manera a *PassengerId*, *Ticket*, este identificador vienen condicionados por otros factores que no suele relacionarse con los individuos directamente (como por ejemplo la empresa comercializadora, el puerto de embargue, la fecha de compra...), también lo descartaremos.

Puesto que no tenemos más suposiciones lógicas sobre las demás variables, las vamos a mantener para los siguientes pasos de procesado/análisis.

```
titanic_train <- titanic_train[,c(2,3,5,6,7,8,10,11,12)]
```

2.2. Valores nulos

Es de gran interés saber si tenemos muchos valores nulos (campos vacíos) y la distribución de valores por variables. Es por ello recomendable empezar el análisis con una visión general de las variables.

Mostraremos para cada atributo la cantidad de valores perdidos mediante la función is.na.

```
colSums(is.na(titanic_train))
```

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

Aunque una práctica común para completar los valores numéricos perdidos es poner la media, en este caso al sospechar que puede afectar significativamente en el grado de supervivencia debemos de tratarlo con precaución ya que aumentar el valor de la tendencia central 177 veces en un juego de datos de 900 podemos crear una falsa conclusión.

Por lo que se ha considerado que en este caso es preferible imputar estos valores basando en una serie de factores como por ejemplo clase, género, número de hermanos/pareja, padres/hijos y el precio del ticket.

```
titanic_train = kNN(titanic_train, variable = "Age",
         dist_var = c("Pclass","Sex", "SibSp", "Parch", "Fare"), imp_var = FALSE)
colSums(is.na(titanic_train))
```

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

Por otro lado, también necesitamos encontrar los valores vacíos, podemos usar colSums para saber la suma en cada columna.

```
colSums(titanic_train=="")
```

```
## Survived   Pclass      Sex      Age    SibSp    Parch     Fare    Cabin
##        0        0        0        0        0        0        0      687
## Embarked
##        2
```

Como se puede ver, la variable *Cabin* tiene un porcentaje de 77 % de valores vacíos, con un número tan alto de valores desconocidos no nos va a ser útil para construir el modelo. Es más, usar esta variable nos introducirá errores. Por lo que decidimos descartar esta variable dell dataset.

```
titanic_train <- titanic_train[-c(8)]
```

Por otro lado, el puerto de embargue contiene una cantidad reducida de campos desconocidos, le asignaremos la categoría de *U (Unkown)*.

```
table(titanic_train$Embarked)
```

```
##
##       C   Q   S
##   2 168  77 644
```

```
titanic_train[titanic_train == ""] <- "U"
table(titanic_train$Embarked)
```

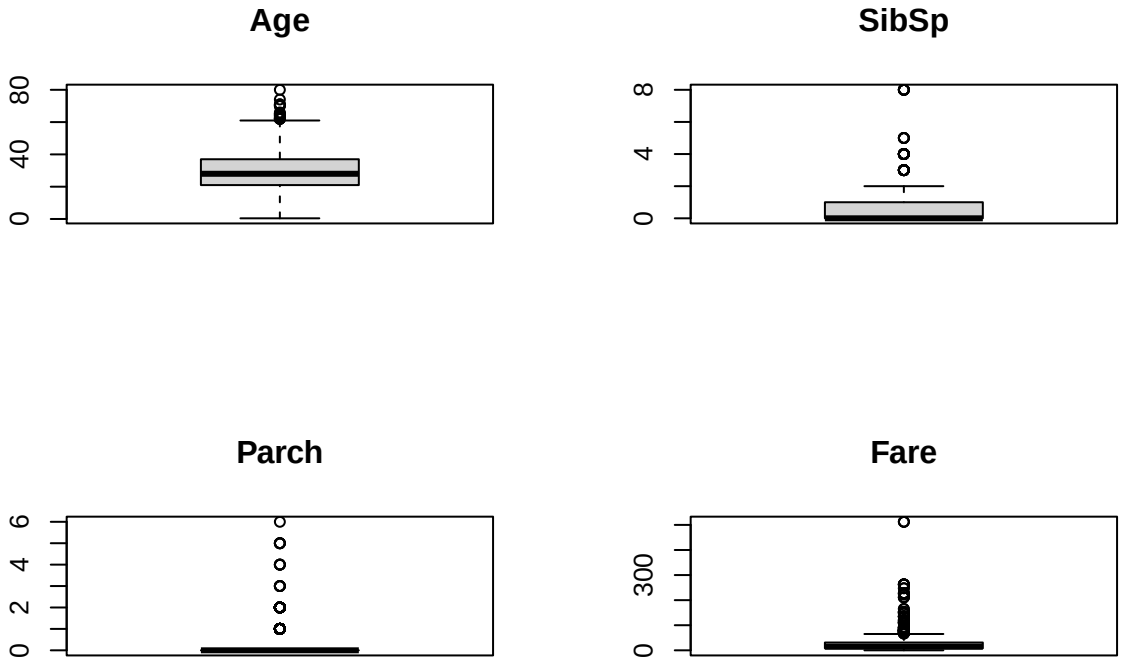
```
##
##   C   Q   S   U
## 168  77 644   2
```

2.3. Valores extremos

2.3.1. Valores numéricos

Para los valores numéricos, lo más rápido para visualizar los outliers son los boxplots.

```
par(mfrow=c(2,2))
age_box <- boxplot(titanic_train[,4], main = colnames(titanic_train[4]))
sibSp_box <- boxplot(titanic_train[,5], main = colnames(titanic_train[5]))
parch_box <- boxplot(titanic_train[,6], main = colnames(titanic_train[6]))
fare_box <- boxplot(titanic_train[,7], main = colnames(titanic_train[7]))
```

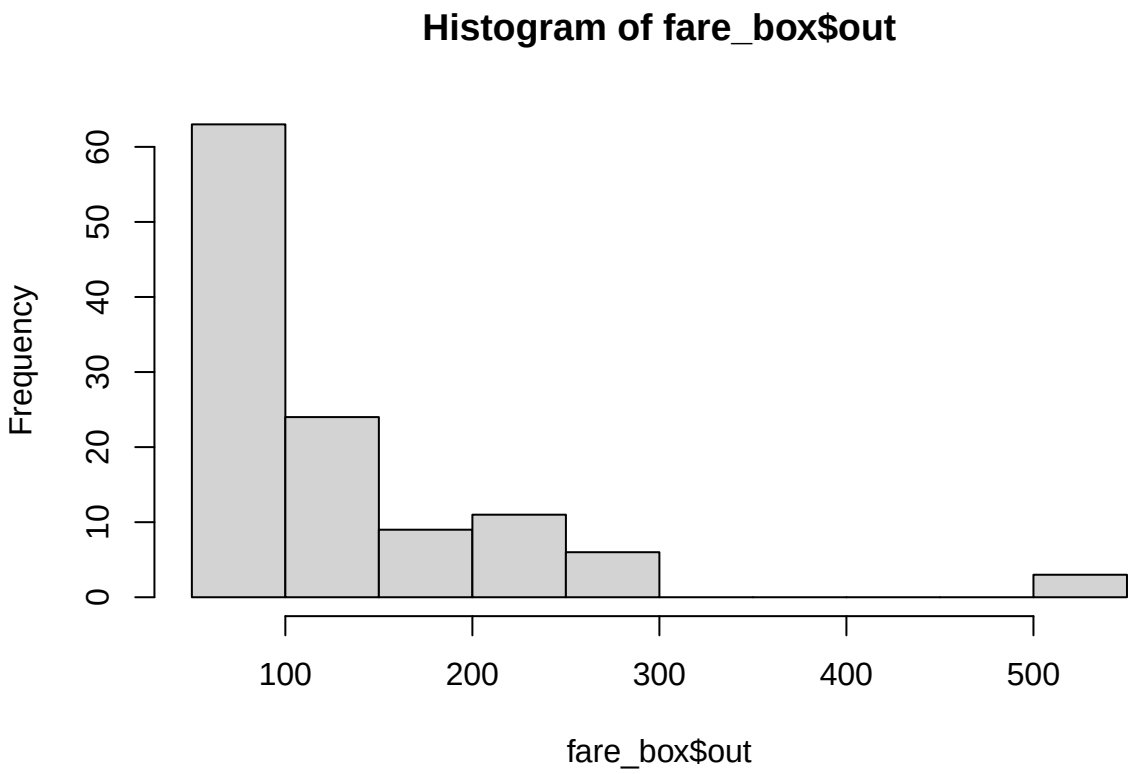


A pesar de que podemos ver muchos puntos en las 4 gráficas, la gran mayoría de ellas están dentro de las posibilidades reales, con lo que se consideran valores anormales pero válidos para el estudio.

No obstante, los únicos datos con los que podemos tener alguna duda es el precio de ticket alrededor de 500, muy alejados de los otros valores del resto.

Para visualizar los outliers detectados, grafiamos un histograma para ver su distribución.

```
hist_counts <- hist(fare_box$out)
```



A pesar de ser outliers, hay bastante cantidad de ellos, esto puede deberse que en un trayecto hay categorías muy diversas, con cantidades cada vez menores en clases crecientes y/o otros servicios que puede afectar al precio final.

Si contamos la cantidad de estos tickes en cada bin, vemos que incluso en el rango de 500 hay más de 1 ticket vendido a ese precio, con lo que es poco probable que se trate de un error de escritura o recopilación de datos.

```
hist_counts$counts
```

```
##  [1] 63 24  9 11  6  0  0  0  0  3
```

Como resumen, en este apartado no se hará tratamiento de ningún valor extremo ya que se pueden considerar válidos todos a pesar de presentar diferencia con la tendencia central de cada variable.

3. Análisis de los datos

primero sacamos dimensión y factorizamos con un nueva variable para poder hacer comparaciones con el grupo superviviente.

3.1. Análisis descriptivo

3.1.1. summary con las 5 estadísticos más comunes

```
summary(titanic_train)
```

```
##     Survived          Pclass          Sex                 Age       
##  Min.   :0.0000   Min.   :1.000   Length:891         Min.   : 0.42  
##  1st Qu.:0.0000   1st Qu.:2.000   Class :character   1st Qu.:21.00  
##  Median :0.0000   Median :3.000   Mode  :character   Median :28.00  
##  Mean   :0.3838   Mean   :2.309                      Mean   :29.64  
##  3rd Qu.:1.0000   3rd Qu.:3.000                      3rd Qu.:37.00  
##  Max.   :1.0000   Max.   :3.000                      Max.   :80.00  
##      SibSp           Parch             Fare          Embarked        
##  Min.   :0.000   Min.   :0.0000   Min.   :  0.00   Length:891        
##  1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:  7.91   Class :character  
##  Median :0.000   Median :0.0000   Median : 14.45   Mode  :character  
##  Mean   :0.523   Mean   :0.3816   Mean   : 32.20                     
##  3rd Qu.:1.000   3rd Qu.:0.0000   3rd Qu.: 31.00                     
##  Max.   :8.000   Max.   :6.0000   Max.   :512.33
```

Ahora añadiremos un campo nuevo a los datos. Este campos contendrá el valor cualitativo de la Supervivencia con un método simple asignadno los valores de la varibale Survived si es 0 le asinamos “NO” y si es 1 le asignamos “SI”.

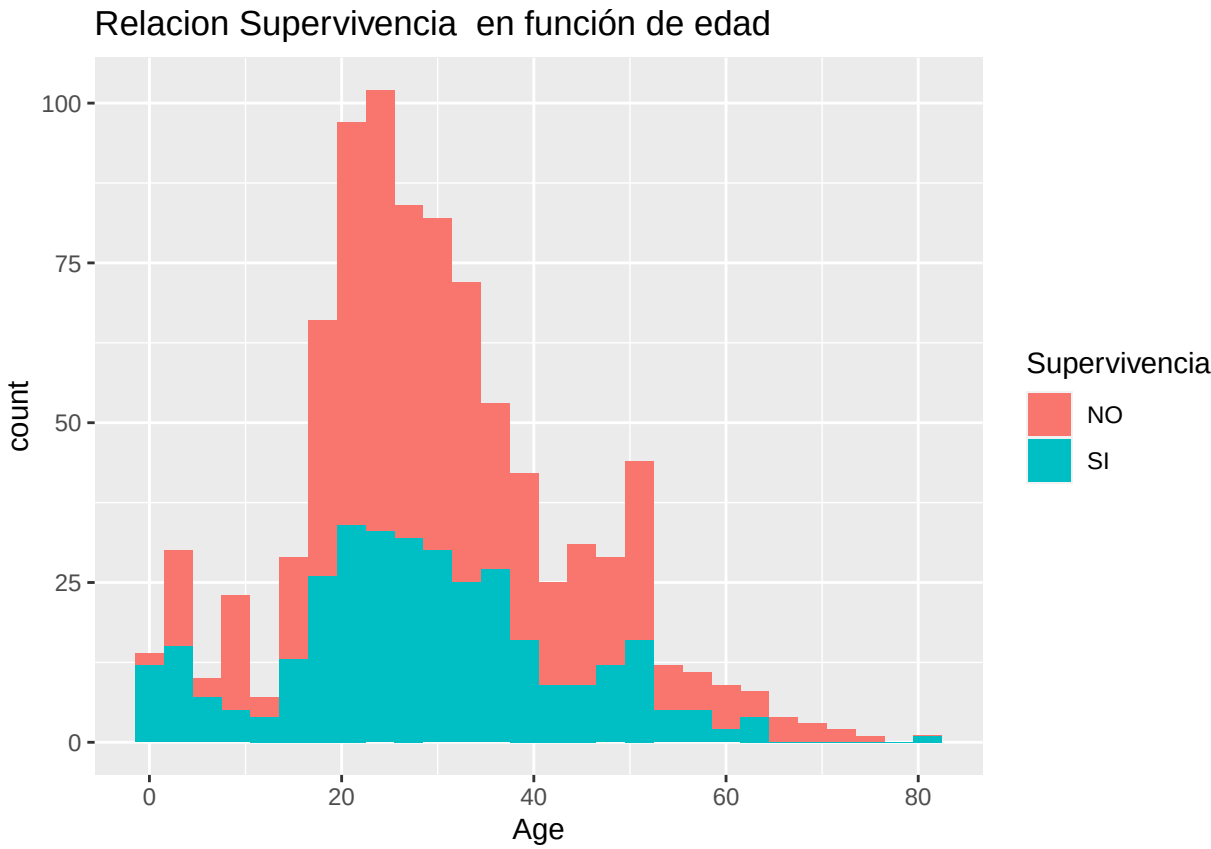
```
titanic_train["Supervivencia"]<-  as.factor(titanic_train$Survived)
titanic_train["Supervivencia"]<- ifelse(titanic_train$Supervivencia == '0' ,'NO','SI')
```

Tambien crearemos un campo nuevo para la Clase del ticket, ya que aunque el datast sea numerico, para nuestors estudios nos vendra bien categorizarlo

```
titanic_train["Clase"]<-  cut(titanic_train$Pclass, breaks = c(0,1,2,3), labels = c("primera", "segunda
```

3.1.2. histogramas/supervivencia de todas las variables numéricas

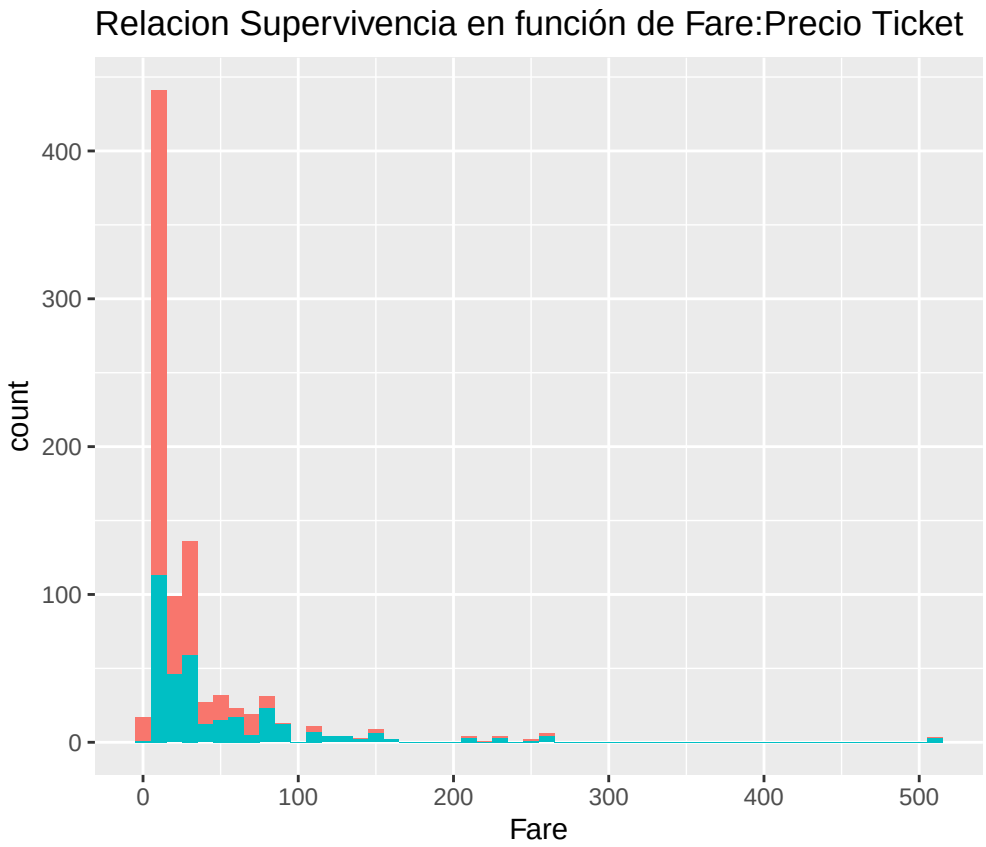
```
ggplot(data = titanic_train, aes(x=Age,fill=Supervivencia))+
  geom_histogram(binwidth =3)+
  ggtitle("Relacion Supervivencia  en función de edad")
```



3.1.2.1. Edad

Observamos como el parámetro position="hijo" nos da la proporción acumulada de un atributo dentro de otro. Parece que los niños tuvieron más posibilidad de salvarse.

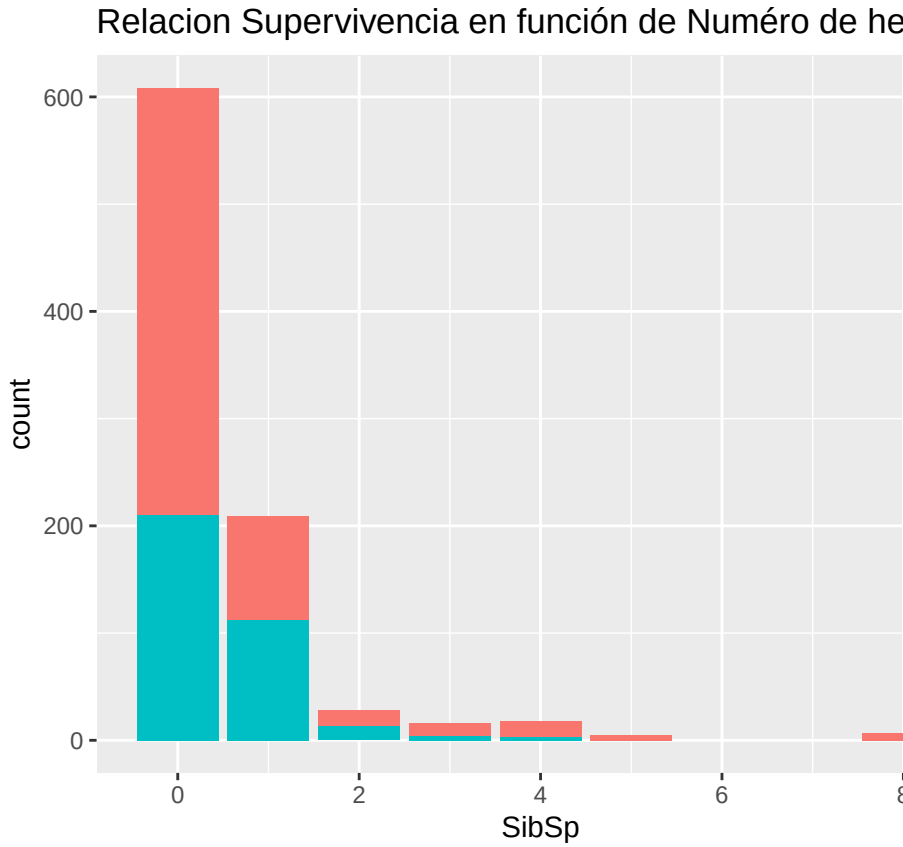
```
ggplot(data = titanic_train, aes(x=Fare,fill=Supervivencia))+
  geom_histogram(binwidth = 10)+
  ggtitle("Relacion Supervivencia en función de Fare:Precio Ticket")
```



3.1.2.2. Precio del ticket

Aunque a simple vista parece que los que poseen un ticket barato tiene menor probabilidad de sobrevivir, lo cierto es que hay muy pocas muestras con precios mucho más elevados.

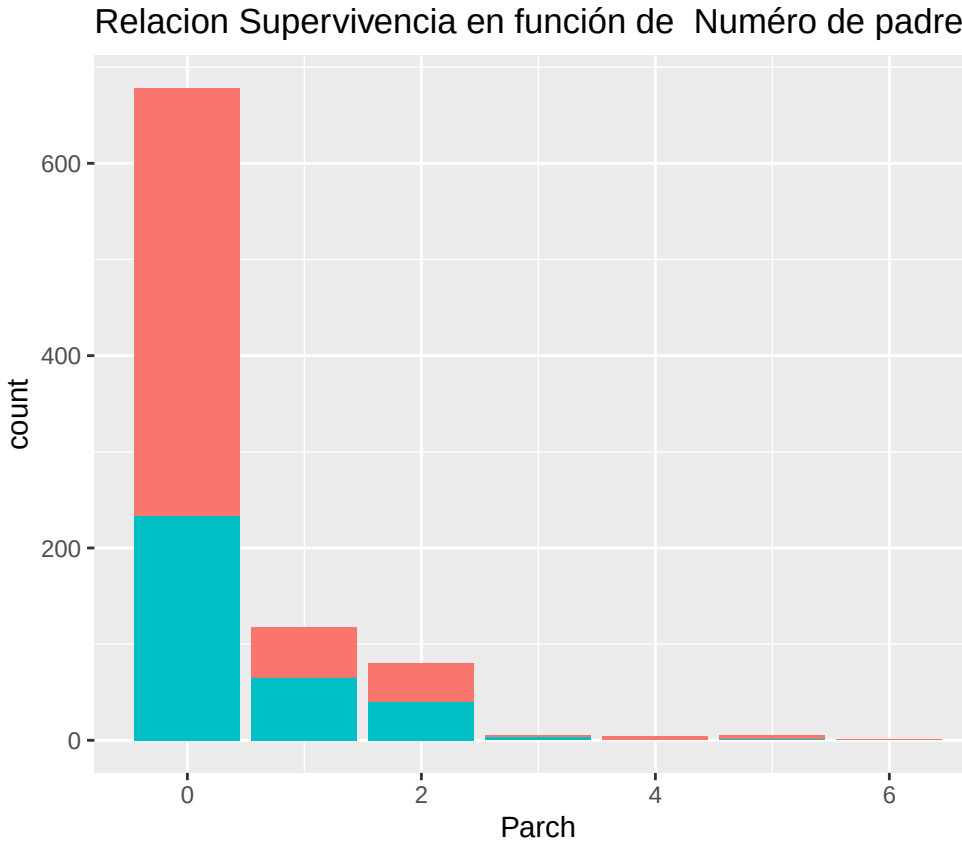
```
ggplot(data = titanic_train, aes(x=SibSp,fill=Supervivencia))+
  geom_bar()+
  ggtitle("Relacion Supervivencia en función de Numéro de hermanos/pareja a bordo")
```



3.1.2.3. Número de hermanos/pareja

La mayoría de los viajeros a bordo no tenían ningún hermano/pareja acompañado, tampoco se ve de forma evidente ninguna relación con la probabilidad de supervivencia.

```
ggplot(data = titanic_train, aes(x=Parch,fill=Supervivencia))+
  geom_bar()+
  ggtitle("Relacion Supervivencia en función de  Numéro de padres/hijos a bord")
```



3.1.2.4. Número de padres/hijos

De la misma manera, la mayoría de los viajeros a bordo no tenían ningún pariente/hijo acompañado, tampoco se ve de forma evidente ninguna relación con la probabilidad de supervivencia.

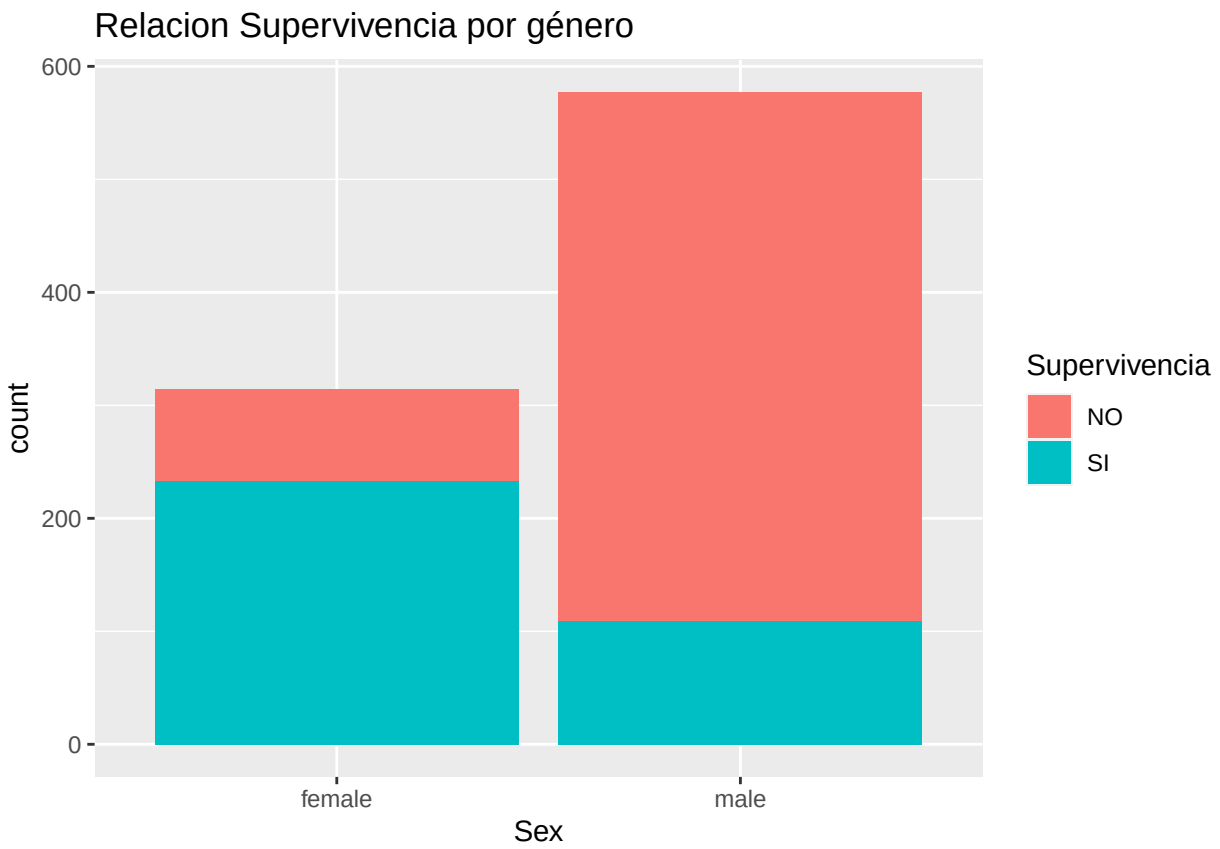
3.1.3. tabla de frecuencias /supervivencia de todas las variables categóricas

Nos proponemos analizar las relaciones entre las diferentes variables del juego de datos para ver si se relacionan y cómo.

3.1.3.1. Género

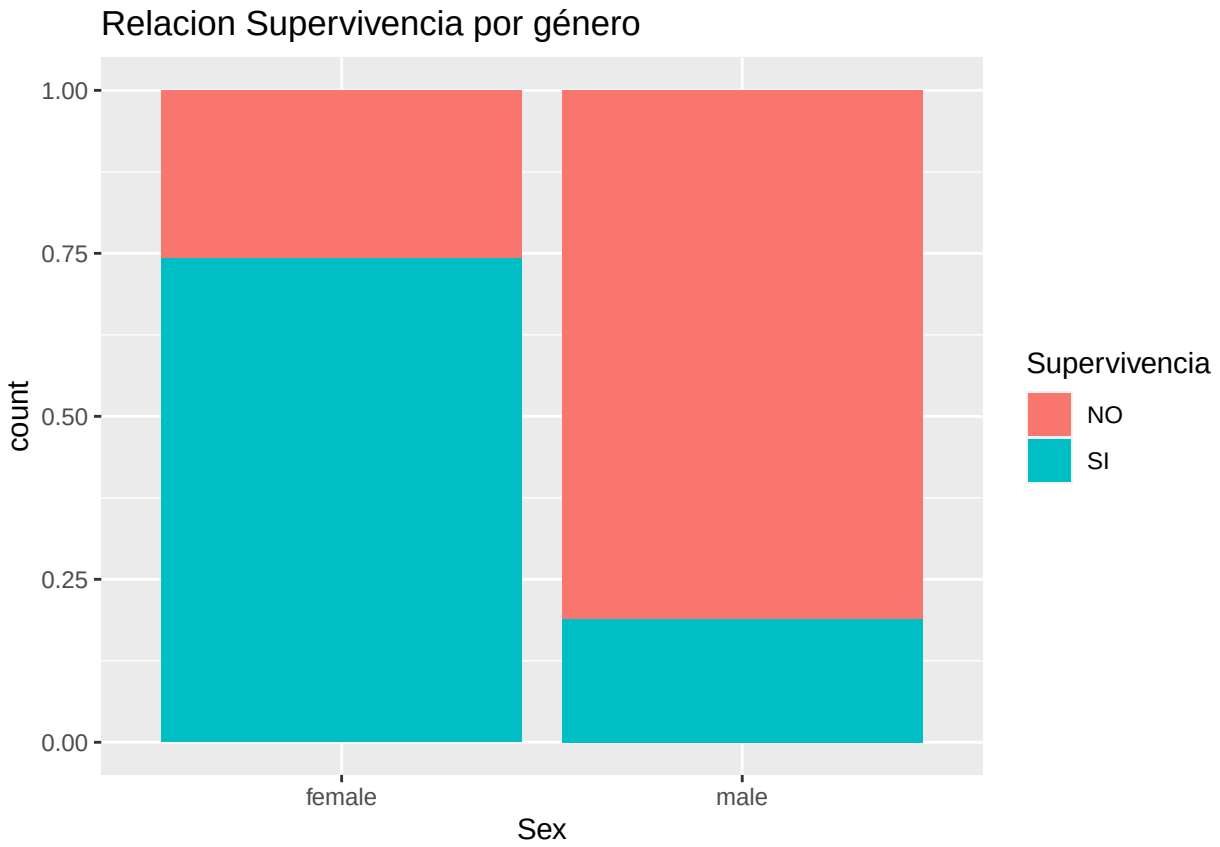
Visualizamos la relación entre las variables "género" y "supervivencia":

```
ggplot(data=titanic_train, aes(x=Sex,fill=Supervivencia))+
  geom_bar()+
  ggtitle("Relacion Supervivencia por género")
```



Otro punto de vista. Survived como función de Sexo:

```
ggplot(data=titanic_train,aes(x=Sex,fill=Supervivencia))+
  geom_bar(position="fill")+
  ggtitle("Relacion Supervivencia por género")
```



En la primera gráfica podemos observar fácilmente la cantidad de mujeres que viajaban respecto hombres y los que no sobrevivieron. En la segunda gráfica de forma porcentual observamos el sexo y los porcentajes de supervivencia en función de ser mujer/hombre.

En ambas gráficas confirmamos la suposición de que la probabilidad de supervivencia para mujeres es mayor que la de hombres.

Podemos obtener numéricamente la matriz de porcentajes de frecuencia:

```
t<-table(titanic_train$Sex,titanic_train$Supervivencia)
for (i in 1:dim(t)[1]){
    t[i,]<-t[i,]/sum(t[i,])*100
}
t
```

```
##
##                NO       SI
##   female 25.79618 74.20382
##   male   81.10919 18.89081
```

Vemos que la posiblidiad de sobrevivir si eran mujeres es de un 74,20 % sin embargo solo del 18,89 % para los hombres

3.1.3.2. Clase Visualizamos la relación entre las variables “clase” y “supervivencia”:

```
ggplot(data=titanic_train,aes(x=Clase,fill=Supervivencia))+
  geom_bar(position="fill")+
  ggtitle("Relacion Supervivencia por clase")
```

Si obtenemos su matriz de porcentajes de frecuencia

```
t<-table(titanic_train$Clase,titanic_train$Supervivencia)
for (i in 1:dim(t)[1]){
    t[i,]<-t[i,]/sum(t[i,])*100
}
t
```

```
##
##                 NO       SI
##   primera 37.03704 62.96296
##   segunda 52.71739 47.28261
##   tercera 75.76375 24.23625
```

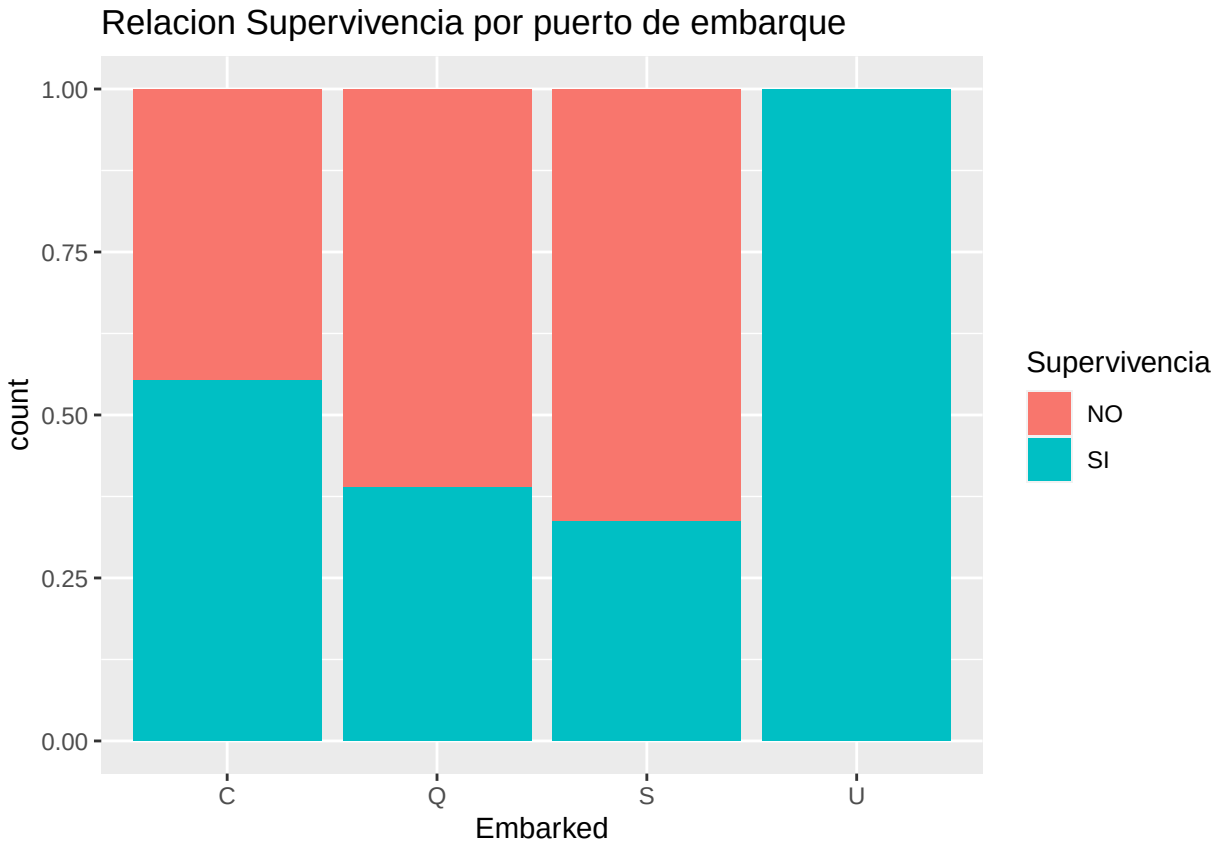
La probabilidadde sobevivir dependiendo de la clase es:

Primera Clase –> 62,96 % Segunda Clsae –> 47,28 % Tercer Clase –> 24,23 %

Parece que los viajeros de primera clase tuvieron mayores probabilidades de sobrevivir que los de la segunda, y estos a su vez que los de la tercera.

3.1.3.3. Puerto Visualizamos la relación entre las variables "Embarked" y "supervivencia":

```
ggplot(data=titanic_train,aes(x=Embarked,fill=Supervivencia))+
  geom_bar(position="fill")+
  ggtitle("Relacion Supervivencia por puerto de embarque")
```



Si obtenemos su matriz de porcentajes de frecuencia

```
t<-table(titanic_train$Embarked,titanic_train$Supervivencia)
for (i in 1:dim(t)[1]){
    t[i,]<-t[i,]/sum(t[i,])*100
}
t
```

```
##
##            NO        SI
##   C  44.64286  55.35714
##   Q  61.03896  38.96104
##   S  66.30435  33.69565
##   U   0.00000 100.00000
```

La probabilidad de sobrevivir dependiendo del puerto de embarque es :

Puerto C –> 55,35 % Puerto Q –> 38,96 % Puerto S –> 33,69 % Puerto U –> 100 %

C = Cherbourg, Q = Queenstown, S = Southampton, U = unkown

Aunque en el puerto U la probabilidad de supervivencia es 100 %, se ha de tener en cuenta que son los dos casos donde no se conocía el puerto de embarque, con lo que no es adecuado derivar conclusiones a partir de estas muestras.

No obstante, sí se observa que los viajeros que embarcan en el puerto de Cherbourg tienen ligeramente mayores probabilidades de sobrevivir que los demás.

3.2. Selección de grupos

Despúes de ver los análisis preliminares, vemos que las variables que influyen en la probabilidad de supervivencia son la edad, la clase, el género y el puerto. Por lo que vamos a conducir más pruebas estadísticas sobre ellas.

En concreto, nos proponemos a comparar los siguientes aspectos entre diferentes grupos:

- La diferencia de edad en el grupo superviviente y y el no superviviente
- La diferenica de la proporción de hombres y mujeres supervivientes
- La diferencia de la proporción de supervivientes en cada clase
- La diferencia de la proporción de supervivientes dependiendo del puerto

A continuación, se seleccionan los grupos dentro de nuestro conjunto de datos que pueden resultar interesantes para analizar y/o comparar.

3.3. Test de normalidad y heteroscedasticidad

Para la comprobación de que los valores que toman nuestras variables cuantitativas provienen de una población distribuida normalmente, utilizaremos la prueba de normalidad de el test de Shapiro Wilk en cada variables numérica.

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

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Age
## W = 0.97844, p-value = 3.333e-10
```

```
shapiro.test(titanic_train$Pclass)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Pclass
## W = 0.71833, p-value < 2.2e-16
```

```
shapiro.test(titanic_train$SibSp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$SibSp
## W = 0.51297, p-value < 2.2e-16
```

```
shapiro.test(titanic_train$Parch)
```

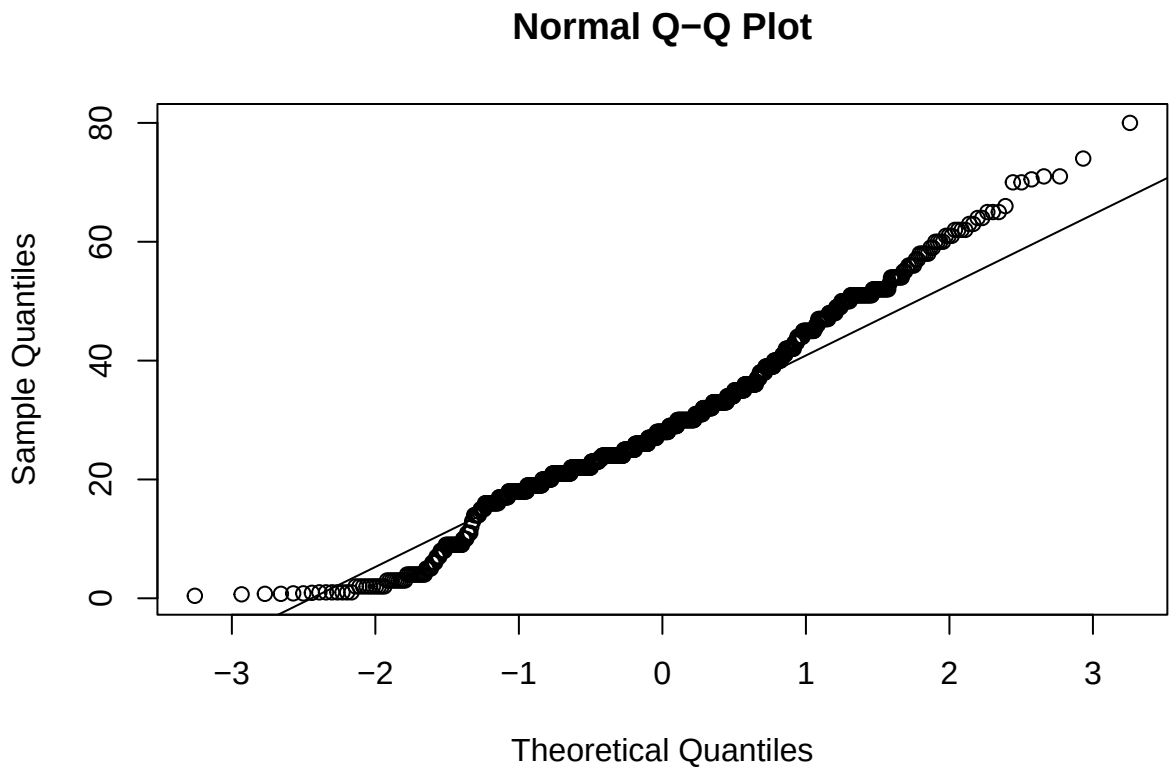
```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Parch
## W = 0.53281, p-value < 2.2e-16
```

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

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Fare
## W = 0.52189, p-value < 2.2e-16
```

De una forma más visual podemos hacer plot QQ por ejemplo sobre la variable cuantitativa que más nos interesa.

```
qqnorm(titanic_train$Age)
qqline(titanic_train$Age)
```



El test nos indica que ninguna variable esta normalizada, ya que el p-valor es inferior al coeficiente 0.05, por lo que se puede rechazar la hipotesis nula y entender que no es normal. Que no sea normal no quiere decir

que no pueda ser normalizable, ya que segun el teorema del limite central al tener mas de 30 elementos en las observaciones podemos aproximarla como una distribución normal de media 0 y desviación estandard 1.

Seguidamente, pasamos a estudiar la homogeneidad de varianzas mediante la aplicación un test de Fligner-Killeen

En este caso, estudiaremos esta homogeneidad en cuanto los grupos conformados por los supervivientes y el sexo. En el siguiente test, la hipótesis nula consiste en que ambas varianzas son iguales.

```
fligner.test(Survived ~ Sex , data =titanic_train)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Survived by Sex
## Fligner-Killeen:med chi-squared = 5.7729, df = 1, p-value = 0.01627
```

Puesto que obtenemos un p-valor menor a 0,05, aceptamos la hipótesis de que las varianzas de ambas muestras NO son homogéneas.

De una manera smilar, estudiamos la homoscedaticidad de la edad en los dos grupos supervivencia

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

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Age by Survived
## Fligner-Killeen:med chi-squared = 1.0004, df = 1, p-value = 0.3172
```

Dado que p-value es superior a nivel de significacia, no podemos rechazar la hipótesis nula de homogeneidad y asumimos que las varianzas de edad entre el grupo superviviente y no superviviente son iguales.

3.4. Pruebas estadísticas

Vamos a proceder a realizar un analisis de correlación entre la variables cuantitivas para ver cuales influyen mas en la supervivencia.

Los resultados anteriores muestran los datos de forma descriptiva, podemos añadir algún test estadístico para validar el grado de significancia de la relación. La librería "DescTools" nos permite instalarlo fácilmente.

```
if(!require(DescTools)){
    install.packages('DescTools', repos='http://cran.us.r-project.org')
    library(DescTools)
}
```

```
## Loading required package: DescTools
```

```
## Warning: package 'DescTools' was built under R version 4.1.2
```

3.4.1. Tests de correlación

```
tabla_SST <-table(titanic_train$Sex, titanic_train$Supervivencia)
Phi(tabla_SST)
```

```
## [1] 0.5433514
```

```
CramerV(tabla_SST)
```

```
## [1] 0.5433514
```

```
tabla_SAT <- table(titanic_train$Age, titanic_train$Supervivencia)
Phi(tabla_SAT)
```

```
## [1] 0.3780396
```

```
CramerV(tabla_SAT)
```

```
## [1] 0.3780396
```

```
tabla_SCT <- table(titanic_train$Clase, titanic_train$Supervivencia)
Phi(tabla_SCT)
```

```
## [1] 0.3398174
```

```
CramerV(tabla_SCT)
```

```
## [1] 0.3398174
```

```
tabla_SPT <- table(titanic_train$Embarked, titanic_train$Supervivencia)
Phi(tabla_SPT)
```

```
## [1] 0.1824838
```

```
CramerV(tabla_SPT)
```

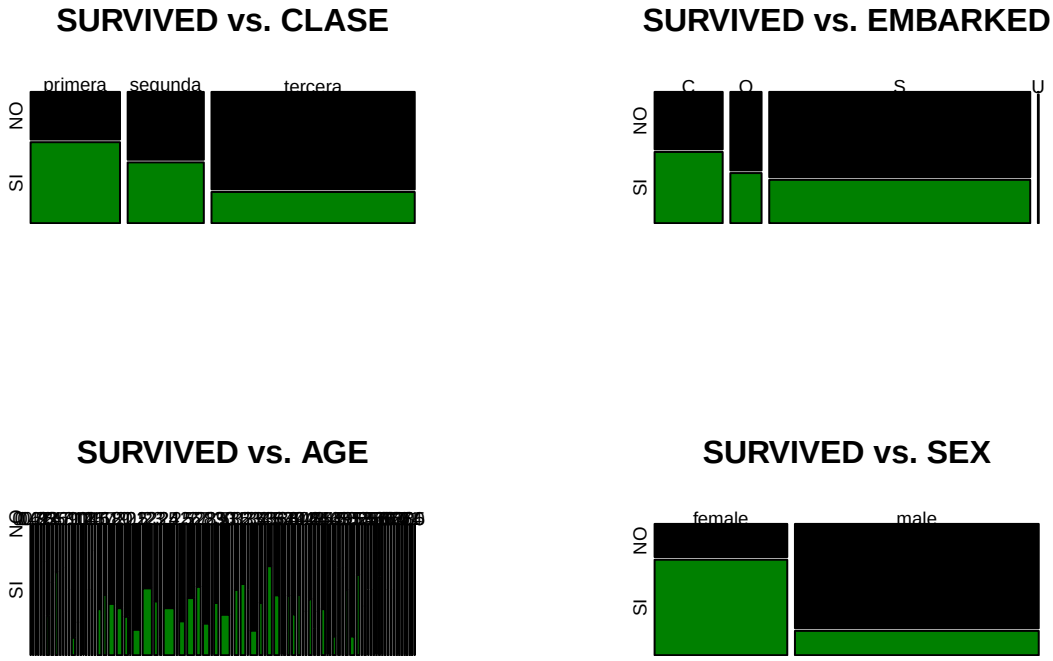
```
## [1] 0.1824838
```

Valores de la V de Cramér (https://en.wikipedia.org/wiki/Cramér%27s_V) y Phi (https://en.wikipedia.org/wiki/Phi_coefficient) entre 0.1 y 0.3 nos indican que la asociación estadística es baja, y entre 0.3 y 0.5 se puede considerar una asociación media.

Finalmente, los valores fueran superiores a 0.5,como en variable Sex , la asociación estadística entre las variables sería alta. Como se puede apreciar, los valores de Phi y V coinciden. Esto ocurre en el contexto de analizar tablas de contingencia 2x2.

Una alternativa interesante a las barras de diagramas, es el plot de las tablas de contingencia. Obtenemos la misma información pero para algunos receptores puede resultar más visual.

```
par(mfrow=c(2,2))
plot(tabla_SCT, col = c("black","#008000"), main = "SURVIVED vs. CLASE")
plot(tabla_SPT, col = c("black","#008000"), main = "SURVIVED vs. EMBARKED")
plot(tabla_SAT, col = c("black","#008000"), main = "SURVIVED vs. AGE")
plot(tabla_SST, col = c("black","#008000"), main = "SURVIVED vs. SEX")
```



En los siguientes apartados, haremos los contrastes de hipótesis para seguir comparando los grupos.

3.4.2. diferencia de supervivencia por edad

En este caso, queremos comparar las medias de las muestras de los dos grupos, por lo que estamos ante un contraste de hipótesis bilateral de dos muestras independientes sobre la media.

Sabemos que la distribución de la edad no es normal, pero puede ser aproximada a ella dado que tenemos una cantidad de muestra superior a lo establecido por el TLC. Además, hemos visto que podemos asumir las varianzas de los dos grupos son iguales.

Las hipótesis que planteamos serían:

H0: La media de edad para el grupo superviviente es igual que el grupo no superviviente. x1 = x2 H1: La media de edad para el grupo superviviente es distinto al de grupo no superviviente. x1 != x2

Usamos el test de Student.

```
x1 = titanic_train$Age[titanic_train$Survived == 0]
x2 = titanic_train$Age[titanic_train$Survived == 1]
```

```
t.test(x1, x2, alternative = "two.sided",  var.equal=FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  x1 and x2
## t = 2.0083, df = 698.19, p-value = 0.04499
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.04367376 3.85850532
## sample estimates:
## mean of x mean of y
##  30.38434  28.43325
```

Dado de p-value es menor que 0.05, rechazamos la hipótesis nula, no son iguales la media de edad en estos grupos. Si observamos la media de los dos grupos podemos ver que en el grupo superviviente la edad es ligeramente menor.

3.4.3. diferencia en proporción de supervivencia por género

Para estudiar el intervalo de confianza para la diferencia de proporciones en dos poblaciones independientes utilizamos la función de r propt.test

comparamos los grupos supervivientes masculinos y femeninos , con el grupo hombre y mujeres.

Las hipótesis que planteamos serían:

H0: La proporción de supervivientes en hombres es igual que en mujeres. p1 = p2 H1: La proporción de supervivientes en hombres no es igual que en mujeres. p1 != p2

```
x1 <- titanic_train[titanic_train$Sex=="male",]
x2 <- titanic_train[titanic_train$Sex=="female",]

n1 <- nrow(x1)
n2 <- nrow(x2)

p1 <- sum(x1$Survived == 1)
p2 <- sum(x2$Survived == 1)
```

```
prop.test (x = c(p1, p2), n= c(n1,n2), alternative="two.side", conf.level=0.95 , correct=FALSE)
```

```
##
##  2-sample test for equality of proportions without continuity
##  correction
##
## data:  c(p1, p2) out of c(n1, n2)
## X-squared = 263.05, df = 1, p-value < 2.2e-16
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.6111119 -0.4951483
## sample estimates:
##    prop 1    prop 2
## 0.1889081 0.7420382
```

Como se observa en la íltima línea del programa, la proporción de supervivientes en los hombre es del 18,89 % y de las mujeres un 74,2 %, mismos resultados que en el apartado donde calculamos la matriz de porcentaje de frecuencia.

Además, el valor p dado es menor que 0.05, lo que implica rechazar la hipótesis nula y confirmar que las proporciones de supervivencia por género son distintas.

3.4.4. Diferencia de la proporción de supervivientes en cada clase

De la misma manera, nos interesan estudiar las diferencias estadísticas según la clase. De manera que planteamos de nuevo un test de proprociones con estas hipótesis:

H0: La proporción de supervivientes en las 3 clases son iguales. p1 = p2 = p3 H0: La proporción de supervivientes en las 3 clases no son todos iguales. p1 != p2 != p3

```
x1 <- titanic_train[titanic_train$Pclass==1,]
x2 <- titanic_train[titanic_train$Pclass==2,]
x3 <- titanic_train[titanic_train$Pclass==3,]


n1 <- nrow(x1)
n2 <- nrow(x2)
n3 <- nrow(x3)

p1 <- sum(x1$Survived == 1)
p2 <- sum(x2$Survived == 1)
p3 <- sum(x3$Survived == 1)
```

```
prop.test (x = c(p1, p2, p3), n= c(n1,n2, n3), alternative="two.side", conf.level=0.95 , correct=FALSE)
```

```
##
##  3-sample test for equality of proportions without continuity
##  correction
##
## data:  c(p1, p2, p3) out of c(n1, n2, n3)
## X-squared = 102.89, df = 2, p-value < 2.2e-16
## alternative hypothesis: two.sided
## sample estimates:
##    prop 1    prop 2    prop 3
## 0.6296296 0.4728261 0.2423625
```

Como se puede ver, volvemos a rechazar la hipótesis nula ya que las proporciones de supervivencia entre las diferentes clases no son iguales siendo la primera clase la que tiene mayor probabilidad de sobrevivir.

3.4.5. Diferencia de la proporción de supervivientes dependiendo del puerto

Por último, analizamos si la supervivencia se relaciona con el puerto de embargue. Se ha de tener en cuenta que tenemos muestras con puerto desconocido y no lo incluiremos para el análisis.

Nos planeamos las siguientes hipótesis:

H0: La proporción de supervivientes en los 3 puertos son iguales. p1 = p2 = p3 H0: La proporción de supervivientes en los 3 puertos son no son todos iguales. p1 != p2 != p3

```r
x1 <- titanic_train[titanic_train$Embarked=="S",]
x2 <- titanic_train[titanic_train$Embarked=="C",]
x3 <- titanic_train[titanic_train$Embarked=="Q",]


n1 <- nrow(x1)
n2 <- nrow(x2)
n3 <- nrow(x3)

p1 <- sum(x1$Survived == 1)
p2 <- sum(x2$Survived == 1)
p3 <- sum(x3$Survived == 1)
```

```r
prop.test (x = c(p1, p2, p3), n= c(n1,n2, n3), alternative="two.side", conf.level=0.95 , correct=FALSE)
```

```
##
##  3-sample test for equality of proportions without continuity
##  correction
##
## data:  c(p1, p2, p3) out of c(n1, n2, n3)
## X-squared = 26.489, df = 2, p-value = 1.77e-06
## alternative hypothesis: two.sided
## sample estimates:
##    prop 1    prop 2    prop 3
## 0.3369565 0.5535714 0.3896104
```

Una vez más demostramos que las proporciones son significativamente diferentes y que los viajeros del puerto "C" tiene mayor probabilidad de supervivencia.

4. regresión logistica

Tal y como hemos visto en el trabajo con los datos nos resulta de gran interes poder predecir si un pasajero sobrevivió o no al hundimiento del Titanic.De hecho existe una competicion en marcha en la plataforma Kaggle para "predecir" o clasificar este dataset.

El modelo que vamos a implementar es la regresión logística, puesto que la regresión linear predice valores continuos y en este caso nuestro objetivo es una variable categórica que indicará sí o no en la supervivencia.

Para obtener un modelo de regresión considerablemente eficiente, lo que haremos obtener varios modelos de regresión utilizando las variables que estén más correladas respecto a la superviiencia Así, de entre todos los modelos que tengamos, escogeremos el mejor utilizando como criterio aquel que presente un menor criterio de información de Akaike (AIC).

Definimos las variables dependiente e independientes.

```r
# Variable a predecir, si se salvo el pasajero
Salvado = titanic_train$Survived
# Regresores   con mayor coeficiente  de correlación con respecto a la superviviencia

Genero =titanic_train$Sex
TicketClase = titanic_train$Pclass
Edad = titanic_train$Age
PuertoE= titanic_train$Embarked
```

Aplicaremos la estrategia de adicción de variables para ir reduciendo el índice de AIC.

```
modelo1<- glm(Salvado ~ Genero, data = titanic_train, family = binomial(link=logit))
summary(modelo1)
```

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

Como se puede ver, el género resulta ser significativo para la predicción del modelo y el hecho de ser hombre contribuye negativamente a la supervivencia.

Añadimos la variable de clase.

```
modelo2<- glm(Salvado ~ Genero + TicketClase, data = titanic_train, family = binomial(link=logit))
summary(modelo2)
```

```
##
## Call:
## glm(formula = Salvado ~ Genero + TicketClase, family = binomial(link = logit),
##     data = titanic_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2030  -0.7036  -0.4519   0.6719   2.1599
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   3.2946     0.2974  11.077   <2e-16 ***
## Generomale   -2.6434     0.1838 -14.380   <2e-16 ***
## TicketClase  -0.9606     0.1061  -9.057   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.7  on 890  degrees of freedom
## Residual deviance:  827.2  on 888  degrees of freedom
## AIC: 833.2
##
## Number of Fisher Scoring iterations: 4
```

Vemos que reduce el índice de AIC de notablemente y la clase en la que se viaja también resulta significativo para la supervivencia. Cuanto mayor es su número, más contribuye a la supervivencia de forma negativa.

Ahora sumamos el efecto de la edad

```
modelo3<- glm(Salvado ~ Genero + TicketClase + Edad, data = titanic_train, family = binomial(link=logit)
summary(modelo3)
```

```
##
## Call:
## glm(formula = Salvado ~ Genero + TicketClase + Edad, family = binomial(link = logit),
##     data = titanic_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6143  -0.6715  -0.4195   0.6408   2.4737
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.633512   0.452491  10.240  < 2e-16 ***
## Generomale  -2.555991   0.186033 -13.739  < 2e-16 ***
## TicketClase -1.190024   0.123230  -9.657  < 2e-16 ***
## Edad        -0.029782   0.007101  -4.194 2.74e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.66  on 890  degrees of freedom
## Residual deviance:  808.78  on 887  degrees of freedom
## AIC: 816.78
##
## Number of Fisher Scoring iterations: 5
```

El incremento de la edad contribuye también negativamente a la supervivencia, aunque su efecto es bastante menor comparado con las demás variables.

Por último, incluimos el puerto del embargue.

```
modelo4<- glm(Salvado ~ Genero + TicketClase + Edad + PuertoE, data = titanic_train, family = binomial(
summary(modelo4)
```

```
##
## Call:
```

```
## glm(formula = Salvado ~ Genero + TicketClase + Edad + PuertoE,
##     family = binomial(link = logit), data = titanic_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5482  -0.6491  -0.4072   0.6746   2.4543
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   4.956840   0.475281  10.429  < 2e-16 ***
## Generomale   -2.513090   0.187620 -13.395  < 2e-16 ***
## TicketClase  -1.179162   0.128652  -9.166  < 2e-16 ***
## Edad         -0.030157   0.007126  -4.232 2.32e-05 ***
## PuertoEQ      0.017722   0.373849   0.047   0.9622
## PuertoES     -0.510434   0.228425  -2.235   0.0254 *
## PuertoEU     12.333769 615.955849   0.020   0.9840
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.66  on 890  degrees of freedom
## Residual deviance:  801.23  on 884  degrees of freedom
## AIC: 815.23
##
## Number of Fisher Scoring iterations: 13
```

A pesar de que el puerto de embargue reduce muy poco el valor de AIC también aporta cierta información a la hora de predicción. Cabe destacar que en este caso las muestras con puerto desconocido han sido evaluados como una categoría extra y el hecho de tener solo 2 casos que sobreviven está confundiendo el algoritmo a entender como factor de éxito para sobrevivir.

Dado que el modelo 4 da el mejor resultado de AIC, lo usaremos para las predicciones.

Una prueba de predicción podría ser como el siguiente caso propuesto:

```
newdata <- data.frame(
Genero = "female",
PuertoE = "S",
Edad = 38,
TicketClase =1)
```

Prediccion de si Sobrbevive o no

```
predict(modelo4, newdata, type='response')
```

```
##        1
## 0.892954
```

El resultado nos indica que con probabilidad de 89 % el individuo del ejemplo sobrevivirá.

4.1. validación con conjunto de test

En el conjunto de test también existen valores de NA en la edad, si no se encuentran disponible darán lugar a predicciones de NA, por lo que seguimos la misma estrategia de imputación para completar estos casos primero.

```
titanic_test = kNN(titanic_test, variable = "Age",
         dist_var = c("Pclass","Sex", "SibSp", "Parch", "Fare"), imp_var = FALSE)
colSums(is.na(titanic_test))
```

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

La columna de Fare no está incluido en nuestro modelo, por lo que lo podemos dejar sin tratar.

Seleccionamos los datos de validación.

```
newdata <- titanic_test[,c("Pclass", "Sex", "Age", "Embarked")]
colnames(newdata) <- c('TicketClase','Genero','Edad', 'PuertoE')

result <- predict(modelo4, newdata, type='response')
```

El resultado que nos proporciona la regresión son probabilidades, tenemos que establecer criterios para categorizar estos en predicciones de supervivencia.

Por lo que para una probabilidad igual o encima de 0.5 consideramos que sobrevive, y consideramos que no si es por debajo de 0.5

```
prediceted_survive <- result >= 0.5

prediceted_survive[prediceted_survive == TRUE] <- 1
prediceted_survive[prediceted_survive == FALSE] <- 0
```

4.2. bondad de ajuste

Una forma para evaluar la bondad de ajuste en las GLM es el test de el test de Hosmer-Lemeshow.

```
if(!require(ResourceSelection)){
    install.packages('ResourceSelection', repos='http://cran.us.r-project.org')
    library(ResourceSelection)
}
```

```
## Loading required package: ResourceSelection
```

```
## Warning: package 'ResourceSelection' was built under R version 4.1.2
```

```
## ResourceSelection 0.3-5   2019-07-22
```

```
hoslem.test(titanic_train$Survived, fitted(modelo4), g=3)
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  titanic_train$Survived, fitted(modelo4)
## X-squared = 2.6883, df = 1, p-value = 0.1011
```

Obtenemos un p-value mayor que 0.05, lo cuál nos indica que tenemos un modelo que ajusta a los datos reales.

4.2.1. ROC

Otra manera visual/numérica para evaluar el modelo es a través de la curva de ROC.

```
if(!require(pROC)){
    install.packages('pROC', repos='http://cran.us.r-project.org')
    library(pROC)
}
```

```
## Loading required package: pROC
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following object is masked from 'package:colorspace':
##
##     coords
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

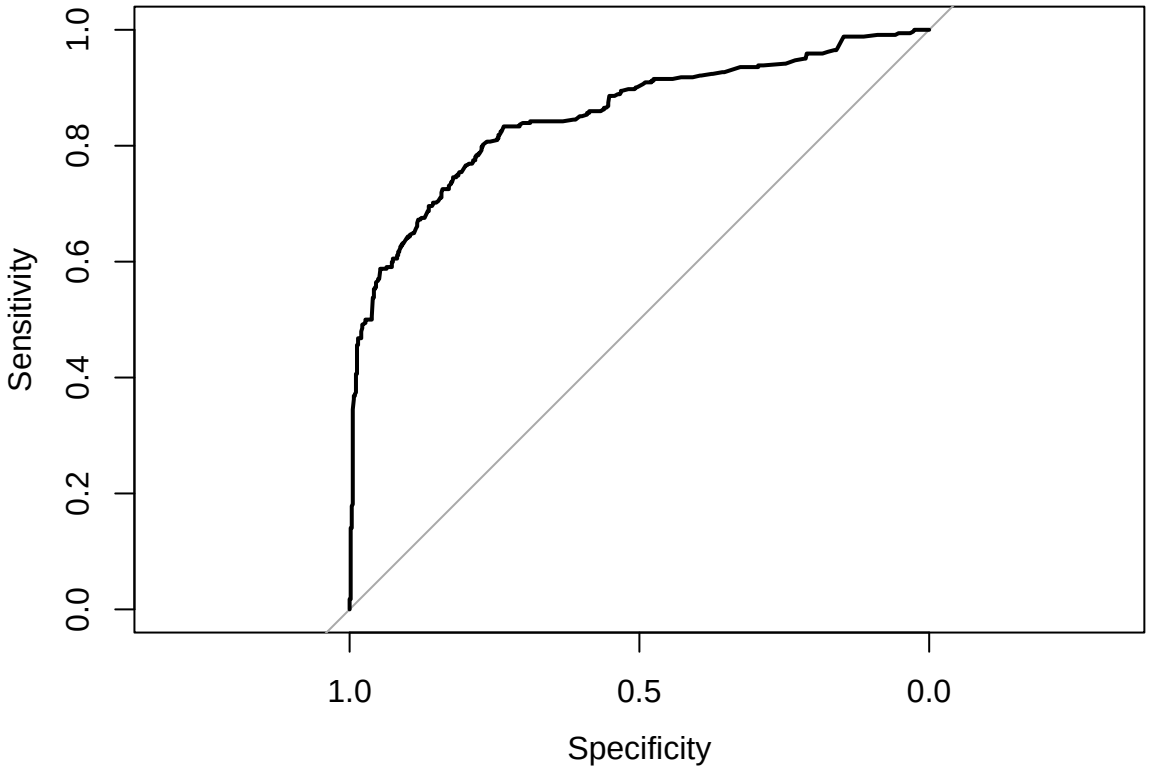
```
traindata <- titanic_train[,c("Pclass", "Sex", "Age", "Embarked")]
colnames(traindata) <- c('TicketClase','Genero','Edad', 'PuertoE')

prob = predict(modelo4, traindata, type="response")
r = roc(titanic_train$Survived, prob, data = titanic_train)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(r)
```




```
auc(r)
```

```
## Area under the curve: 0.8518
```

Tenemos un AUROC de 0.8518, con lo que el modelo discrimina de forma adecuada la variable dependiente.

5. Conclusiones

Como se ha visto con este conjunto de datos se trataba de responder la probabilidad de superviviencia de los pasajeros.

Previamente, se han sometido los datos a un preprocesamiento para manejar los casos de ceros o elementos vacíos y valores extremos (outliers). Para el caso del primero, se ha hecho uso de un método de imputación de valores de tal forma que no tengamos que eliminar registros del conjunto de datos inicial y que la ausencia de valores no implique llegar a resultados poco certeros en los análisis. Para el caso de los valores extremos, se ha optado por incluir los valores en los análisis dado que se pueden considerar válidos todos a pesar de presentar diferencia con la tendencia central de cada variable.

Una vez analizado los datos se han realizado pruebas estadística como es el análisis de correlación sobe el conjunto de datos para ver cuales de las variables del dataset titanic tenia mas influencia sobre la supervivencia. Para cada una de las variables, hemos podido ver cuáles son los resultados que arrojan (entre otros, mediante tablas y proporciones)

Así, el análisis de correlación nos ha permitido conocer cuáles de estas variables ejercen una mayor influencia sobre el la probabilidad de supervivencia del pasajero que embarco.

Mientras que el modelo de regresión logistica obtenido resulta de utilidad a la hora de realizar predicciones sobre la supervivencia de un pasajero con unas caracteristicas dadas.

Asi hemos podido llegar a la conclusion que el genero, el puerto de embarque, la edad y el tipo de billete son las varibales que mas influyen a la hora de clasificar si un pasajero sobrevivio o no dadas unas características concretas.

6. Código y exportación de datos

Exportamos los dos juegos de datos limpios que hemos procesado a partir de los datos originales.

```
write.csv(titanic_train, "titanic_train_clean.csv")
write.csv(titanic_test, "titanic_test_clean.csv")
```

Estos archivos junto con el código fuente están subidos en el repositorio Github en esta dirección:

Repositorio Titanic

7. Recursos básicos

Material didáctico de: Tipología y Ciclo de Datos

Complementarios:

- Los descritos para la anterior PEC.
- Fichero :train.csv :Titanic: Machine Learning from Disaster (https://www.kaggle.com/c/titanic)