

Práctica 2 - Limpieza y análisis de datos

Pablo López Ladrón de Guevara

Rafael Corvillo Alonso

28 de mayo, 2021

Contents

1. Descripción del dataset.	2
2. Integración y selección de los datos de interés a analizar.	3
2.1. Selección de los datos interés	4
2.2. Conversión de variables categóricas a factor	5
3. Limpieza de los datos	6
3.1. Datos Perdidos	6
3.2. Valores extremos.	7
4. Análisis de los datos	12
4.1. Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar)	12
4.2. Comprobación de la normalidad y homogeneidad de la varianza	12
4.3. Aplicación de pruebas estadísticas	19
5. Representación de los resultados a partir de tablas y gráficas.	30
6. Resolución del problema y conclusiones	35

1. Descripción del dataset.

El conjunto de datos utilizado es el correspondiente a la competición Kaggle del hundimiento del Titanic. El objetivo es predecir si un pasajero sobrevivió o no al hundimiento del Titanic. El dataset contiene los siguientes atributos:

- **PassengerId:** Identificador único del pasajero.
- **Survived:** Es la variable objetivo que se pretende predecir. Si el pasajero sobrevivió al naufragio, codificada como 0 (no) y 1 (sí).
- **Pclass:** Clase en la que viaja el pasajero: Primera segunda o tercera (1, 2, 3)
- **Name:** Nombre del pasajero.
- **Sex:** Sexo del pasajero.
- **Age:** Edad del pasajero. Pueden tener números decimales, refiriéndose con ellos a los meses.
- **SibSp:** Número de los siguientes tipos de familiares que viajan en el barco con el pasajero:
 - Hermanos/as
 - Hermanastros/as
 - Marido/esposa (no se tiene en cuenta amantes o prometidas)
- **Parch:** Número de los siguientes tipos de familiares que viajan en el barco con el pasajero:
 - Padres/madres
 - Hijos/as
 - Hijastros/as
 - En el caso de que el niño viaje con una niñera, no se contará a la niñera como familiar.
- **Ticket:** Identificador del billete.
- **Fare:** Precio pagado por el billete.
- **Cabin:** Número de camarote.
- **Embarked:** Puerto de embarque el pasajero.

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

Vamos a proceder a cargar el conjunto de datos y a realizar un primer análisis visual de los datos cargados y sus tipos. Durante toda la práctica trabajaremos únicamente con el subconjunto de datos de entrenamiento (`train.csv`), ya que es el que contiene información de la variable objetivo `Survived`. El subconjunto de datos de test (`test.csv`) se utilizará al final para predecir la variable objetivo a partir de un modelo supervisado, y el resultado será presentado a la competición de Kaggle a la que pertenecen estos conjuntos de datos.

Si abrimos el archivo con un editor de texto como Notepad++ vemos se utiliza codificación UTF-8 sin BOM. Así que lo leemos con codificación 'UTF-8'.

```
# Carga del conjunto de datos
titanic_data <- read.csv('train.csv')

# Comprobamos que se carga correctamente
head(titanic_data)
```

```
##   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  NA     0     0
##
##   Ticket    Fare Cabin Embarked
## 1   A/5 21171  7.2500      S
## 2   PC 17599 71.2833   C85      C
## 3 STON/O2. 3101282  7.9250      S
## 4   113803 53.1000  C123      S
## 5   373450  8.0500      S
## 6   330877  8.4583      Q
```

```
# Dimensiones de los datos
dim(titanic_data)
```

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

Comprobamos que la información se ha cargado correctamente y que el subconjunto de datos de entrenamiento tiene 891 registros y 12 variables, donde una de ellas es la variable dependiente `Survived`.

Vamos a ver a continuación los valores resumen de este conjunto de datos y los tipos con los que se han cargado las variables para empezar a intuir qué tipo de técnicas de preprocesado vamos a tener que aplicar.

```
# Valores resumen
summary(titanic_data)
```

```
##   PassengerId    Survived  Pclass         Name
```

```
## Min.      : 1.0    Min.      :0.0000    Min.      :1.000    Length:891
## 1st Qu.:223.5    1st Qu.:0.0000    1st Qu.:2.000    Class :character
## Median :446.0    Median :0.0000    Median :3.000    Mode  :character
## Mean    :446.0    Mean    :0.3838    Mean    :2.309
## 3rd Qu.:668.5    3rd Qu.:1.0000    3rd Qu.:3.000
## Max.    :891.0    Max.    :1.0000    Max.    :3.000
##
##      Sex      Age      SibSp      Parch
## Length:891    Min.      : 0.42    Min.      :0.000    Min.      :0.0000
## Class :character 1st Qu.:20.12    1st Qu.:0.000    1st Qu.:0.0000
## Mode  :character Median :28.00    Median :0.000    Median :0.0000
##                      Mean  :29.70    Mean  :0.523    Mean  :0.3816
##                      3rd Qu.:38.00    3rd Qu.:1.000    3rd Qu.:0.0000
##                      Max.   :80.00    Max.   :8.000    Max.   :6.0000
##                      NA's    :177
##      Ticket      Fare      Cabin      Embarked
## Length:891      Min.      : 0.00    Length:891      Length:891
## Class :character 1st Qu.: 7.91    Class :character Class :character
## Mode  :character Median :14.45    Mode  :character Mode  :character
##                      Mean  :32.20
##                      3rd Qu.:31.00
##                      Max.   :512.33
##
```

```
# Tipos con los que se han cargado las variables
str(titanic_data)
```

```
## '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" ...
```

Vemos que vamos a tener que realizar un tratamiento de valores ausentes, por ejemplo a la variable **Age** (177 observaciones con valor NA), además de analizar los valores extremos en otras variables con **Fare**. También podemos ver que algunas variables que parecen ser de tipo factor han sido cargadas con tipo texto o entero.

2.1. Selección de los datos interés

Las variables **Name** y **Ticket** no aportarán ninguna información de interés de cara a saber si un pasajero sobrevive o no. Además, la información que nos puede aportar **Ticket** ya la tenemos disponible en la variable **Fare**, ya que los tickets tienen asociados un precio, y la que nos puede aportar el apellido la tenemos de forma más completa en las variables **SibSp** y **Parch**.

```
# Eliminamos las variables que no vamos a utilizar
drop_data <- names(titanic_data) %in% c("Name", "Ticket")
titanic_data <- titanic_data[, !drop_data]
```

2.2. Conversión de variables categóricas a factor

Convertimos los atributos categóricos a tipo factor. La variable PClass fue clasificada como tipo int al leer el dataset, pero en realidad solo toma tres valores (Las tres clases antes comentadas).

```
# Convertimos las variables categóricas a factor
titanic_data$Survived <- as.factor(titanic_data$Survived)
titanic_data$Pclass <- as.factor(titanic_data$Pclass)
titanic_data$Sex <- as.factor(titanic_data$Sex)
titanic_data$Embarked <- as.factor(titanic_data$Embarked)
```

3. Limpieza de los datos

Ahora vamos realizar el preprocesado de los datos para tener el conjunto de datos listo para los análisis que realizaremos posteriormente. Empezaremos estudiando los datos perdidos y después analizaremos los valores extremos de algunas variables.

3.1. Datos Perdidos

Mostramos el número de valores ausentes en cada uno de los atributos del conjunto de datos.

```
# Valores ausentes  
colSums(is.na(titanic_data))
```

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

```
# Cadenas de caracteres vacías  
colSums(titanic_data=="")
```

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

Como podemos ver, la variable **Age** contiene 177 observaciones con valor NA y tanto la variable **Cabin** como **Embarked** tienen observaciones con cadena de caracteres vacías. En el atributo **Cabin** existen 687 cadenas de caracteres vacías y **Embarked** contiene 2.

Los datos perdidos encontrados en las variables comentadas se tratarán de forma diferente según las características de la variable o la cantidad de datos perdidos. Si las aproximaciones utilizadas para limpiar los datos son los correctos o no, no se sabe hasta que se implementan los modelos. Si no se consiguen los resultados deseados, una posibilidad para intentar mejorar el modelo es volver a este apartado y gestionar de forma diferente el tratamiento de elementos vacíos.

Primero tratamos los datos vacíos de la variable **Embarked**. En este caso solo existen 2 elementos vacíos. Una primera posibilidad sería eliminar completamente esos dos registros, pero con el objetivo de no perder información, imputaremos estos dos registros con la moda estadística de la variable. Para ello creamos la función `getmode()` que nos devuelve el valor más frecuente de una variable.

```
# Función para calcular la moda  
getmode <- function(v) {  
  uniqv <- unique(v)  
  uniqv[which.max(tabulate(match(v, uniqv)))]  
}  
  
# Calculamos la moda de la variable Embarked  
Embarked_mode <- getmode(titanic_data$Embarked)  
  
# Imputamos la moda a las observaciones vacías  
titanic_data$Embarked[titanic_data$Embarked == ""] <- Embarked_mode
```

En la variable `Cabin` el número de elementos vacíos es muy elevado, 687 de un total de 891 registros (77,1%). Debido a este alto porcentaje de elementos vacíos, una primera posibilidad sería sustituir las cadenas vacías por una constante como “Desconocido”. Pero en lugar de ello optaremos por eliminar por completo la variable. Tiene sentido debido al elevado número de valores vacíos y a que el atributo `Pclass` nos puede dar una información similar de cara a saber si un pasajero sobrevive o no. Los camarotes de primera clase estarían situados cerca de cubierta, siendo la probabilidad de sobrevivir mayor que la de un pasajero que viaje en tercera clase y cuyo camarote se encuentre en una planta muy inferior.

```
# Eliminamos la variable Cabin
titanic_data <- titanic_data[, -9]
```

Por último, vamos a tratar los valores ausentes de la variable `Age`. Para imputar estos valores perdidos de las edades de los pasajeros dividiremos el conjunto de datos en seis grupos y le asignaremos el valor de la mediana de cada grupo. Los atributos elegidos para crear los grupos serán la clase en la que viaja el pasajero y el sexo. Como veremos en el apartado de análisis, estas dos variables serán importantes a la hora de decidir si un pasajero sobrevive o no.

Creamos con ayuda de la función `tapply()` la matriz con las medianas de las edades en los grupos comentados.

```
# Calculamos la mediana de Age por grupos de Pclass y Sex
Age_median_matrix <-
  tapply(titanic_data$Age, list(titanic_data$Pclass, titanic_data$Sex),
        median, na.rm = TRUE)
Age_median_matrix

##   female male
## 1   35.0   40
## 2   28.0   30
## 3   21.5   25
```

Se observa que hay diferencia entre las medianas de los diferentes grupos. Con esto corroboramos que ha sido una aproximación acertada hacerlo de esta manera.

Sustituimos los registros vacíos de edad con los valores correspondientes de la matriz de medianas calculadas.

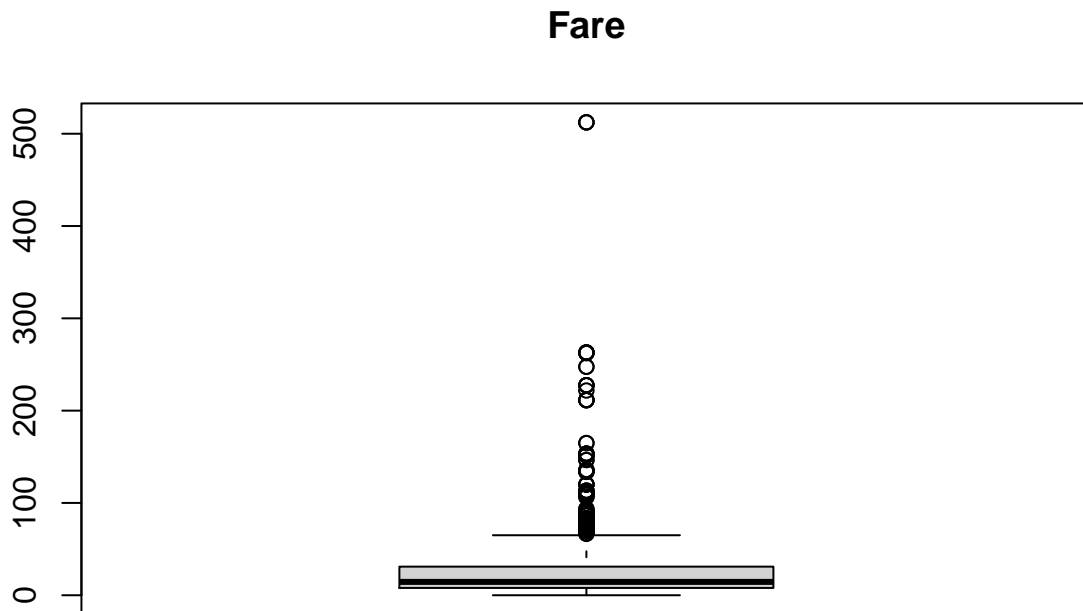
```
# Imputamos los valores ausentes de la variable Age
titanic_data$Age[is.na(titanic_data$Age) & titanic_data$Sex == "female" &
  titanic_data$Pclass == "1"] <- Age_median_matrix[1,1]
titanic_data$Age[is.na(titanic_data$Age) & titanic_data$Sex == "female" &
  titanic_data$Pclass == "2"] <- Age_median_matrix[2,1]
titanic_data$Age[is.na(titanic_data$Age) & titanic_data$Sex == "female" &
  titanic_data$Pclass == "3"] <- Age_median_matrix[3,1]
titanic_data$Age[is.na(titanic_data$Age) & titanic_data$Sex == "male" &
  titanic_data$Pclass == "1"] <- Age_median_matrix[1,2]
titanic_data$Age[is.na(titanic_data$Age) & titanic_data$Sex == "male" &
  titanic_data$Pclass == "2"] <- Age_median_matrix[2,2]
titanic_data$Age[is.na(titanic_data$Age) & titanic_data$Sex == "male" &
  titanic_data$Pclass == "3"] <- Age_median_matrix[3,2]
```

3.2. Valores extremos.

Mostramos los diagramas de cajas de las variables continuas del dataset (`Age` y `Fare`) para comprobar si existen outliers.

Comenzamos observando el atributo correspondiente al precio de los billetes.

```
# Diagrama de caja de Fare
boxplot(titanic_data$Fare, main = "Fare")
```



Utilizamos la función `boxplot.stats()` para obtener las estadísticas de forma numérica.

```
boxplot.stats(titanic_data$Fare)$stats
```

```
## [1] 0.0000 7.9104 14.4542 31.0000 65.0000
```

Existen valores extremos por encima del bigote superior del boxplot (65). La distribución de esta variable está muy desplazada a la izquierda. Esto es debido a que el 50% de los datos se encuentran entre 7,89 y 31,27. Pero se vendieron billetes muy por encima de esos precios. Esta desigualdad en los precios es plausible, ya que al tratarse del viaje inaugural de un transatlántico de lujo, pudo haber diferentes paquetes de precios con diferentes características. Los paquetes más lujosos estarían destinados para los pasajeros más selectos. Un ejemplo sería el billete con un precio cercano a los 500\$. Se diferencian mucho del resto, pero pueden corresponder a la suite más lujosa del barco.

Vemos cuántos outliers son identificados mediante el diagrama de caja.

```
length(titanic_data$Fare[titanic_data$Fare > 65])
```

```
## [1] 116
```

Aunque estos 166 registros son posibles como ya hemos comentado, los trataremos de cara a mejorar el funcionamiento de los modelos que se elaborarán. A la hora de tratar los outliers, se pueden realizar imputación de datos como en el caso de los datos perdidos. En este caso realizaremos la técnica llamada Capping, que consiste en imputar los outliers por encima del bigote superior con dicho valor.

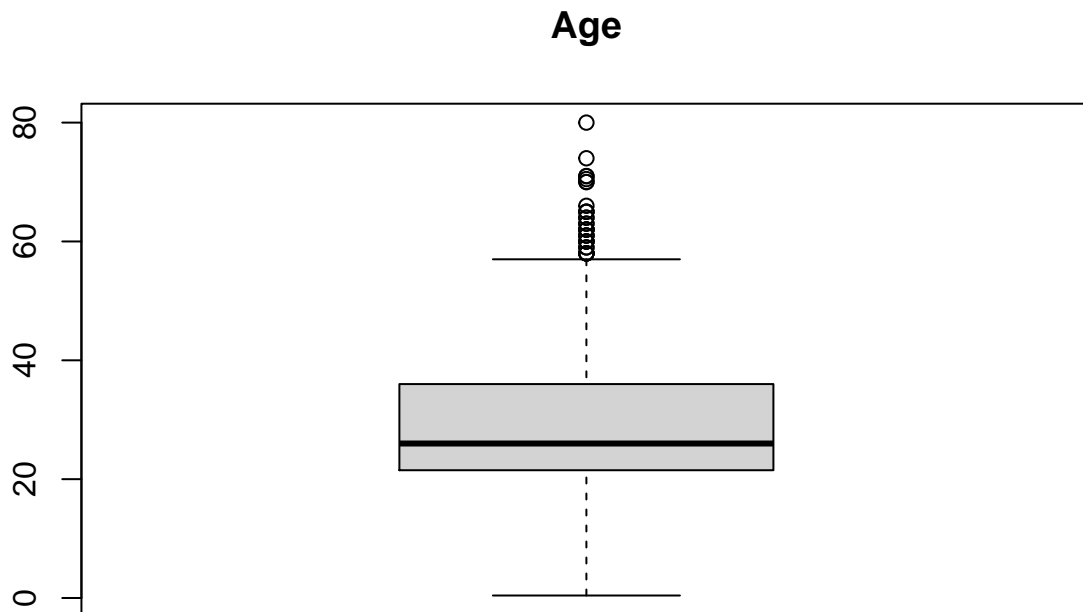
```
# Corrección de outliers de Fare
titanic_data$Fare[titanic_data$Fare > 65] <- 65
```

Mostramos el nuevo diagrama boxplot sin valores extremos.


```
# Diagrama de caja de Fare con los outliers corregidos
#boxplot(titanic_data$Fare, main = "Fare")
```

Analizamos a continuación los valores extremos del atributo correspondiente a la edad de los pasajeros.

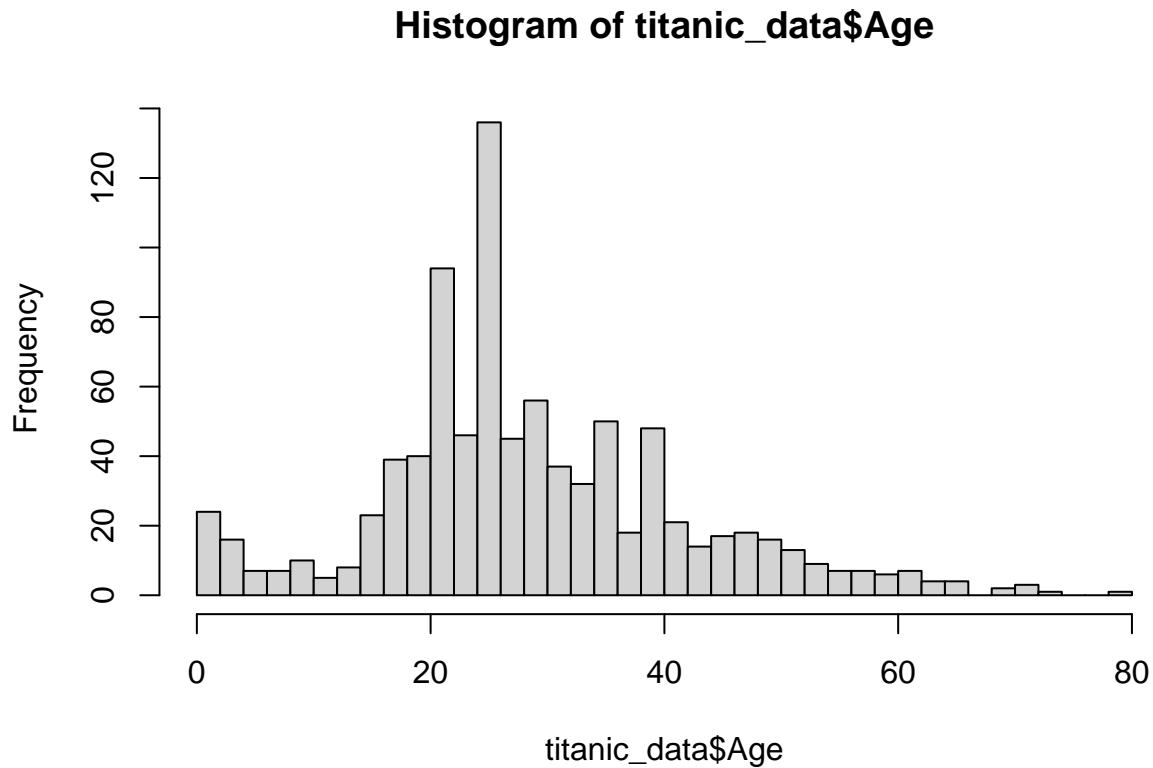
```
# Diagrama de caja de Age
boxplot(titanic_data$Age, main = "Age")
```



Se vuelven a observar valores extremos por encima del bigotes superior del gráfico. Como en el caso anterior es totalmente posible que viajen personas por encima de 60 años. Una posible solución sería no hacer nada con estos outliers para no perder información. En su lugar, optaremos por discretizar la variable. Aunque decidimos mantener la variable original, para poder utilizarla en algunos subapartados del análisis.

Para la discretización hemos decididos crear cinco niveles de frecuencia parecida. Mostramos el histograma del atributo para ver cómo se distribuye. Mostramos agrupaciones de dos en dos, para observarlo con más detalle.

```
hist(titanic_data$Age,breaks = 40)
```



Lo primero que se observa es que gran parte de los datos se concentran entre 20 y 40. Por lo que en ese rango habrá más niveles.

Al existir un total de 891 registros en el conjunto de datos y querer discretizar la edad en 5 niveles, el objetivo será obtener aproximadamente 178 registros en cada nivel. observando el histograma y probando diferentes límites, los cinco niveles de discretización serán los siguientes:

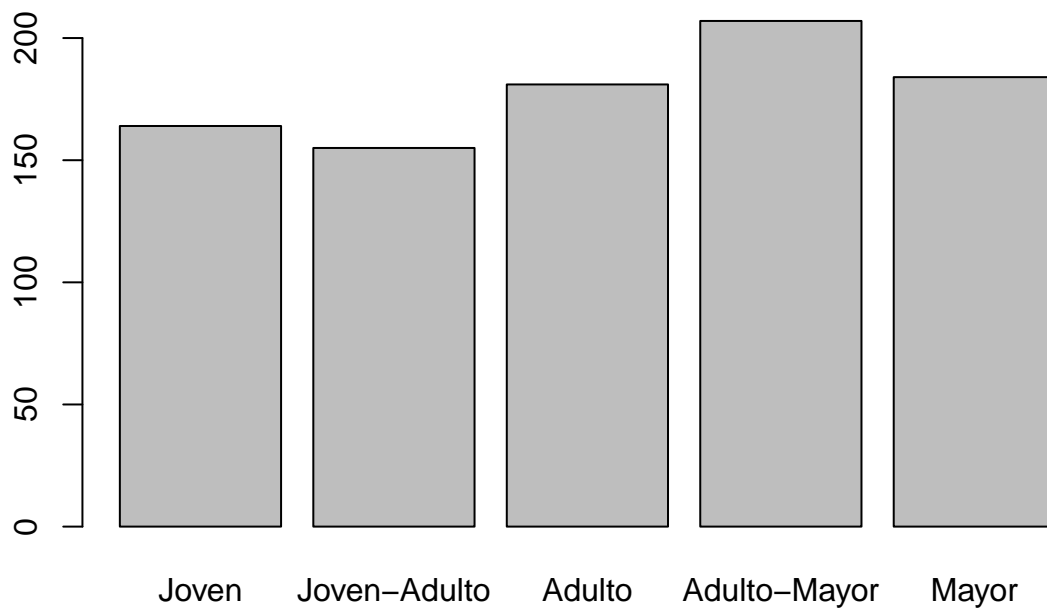
- Joven: [0,19]
- Joven-Adulto: [20,24]
- Adulto: [25,28]
- Adulto-Mayor: [29,39]
- Mayor: [40,80]

Para ello utilizamos la función `cut()`. Activamos la opción `ordered_result` ya que existe un orden de menor a mayor entre los niveles establecidos.

```
titanic_data$Age_d <- cut(titanic_data$Age, breaks = c(0,19,24,28,39,80),  
                          ordered_result = TRUE,  
                          labels = c("Joven", "Joven-Adulto",  
                                     "Adulto", "Adulto-Mayor", "Mayor"))
```

Comprobamos gráficamente que la variable discretizada se distribuye de manera similar en cada uno de sus niveles.

```
plot(titanic_data$Age_d)
```



4. Análisis de los datos

Ahora vamos a realizar diferentes análisis de los datos, haciendo una primera planificación de los grupos de datos y de las pruebas estadísticas que vamos a realizar para ver la relación entre las variables del conjunto de datos.

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

Durante este apartado vamos a realizar diferentes pruebas estadísticas sobre las variables de interés del conjunto de datos. Empezaremos haciendo un análisis de la normalidad y la homocedasticidad de las variables cuantitativas del conjunto de datos (**Age**, **SibSp**, **Parch** y **Fare**). Con los resultados de estos análisis podremos saber qué tipos de pruebas estadísticas (paramétricas o no paramétricas) vamos a poder realizar en los contrastes de hipótesis. Gracias a estos contrastes de hipótesis podremos analizar las relaciones entre las variables del conjunto de datos y más concretamente con la variable objetivo **Survived**, para saber qué variables tuvieron más influencia en el hecho de sobrevivir o no al accidente. También haremos un análisis de correlaciones de las variables cuantitativas para comprobar si existe relación entre ellas.

Una vez estudiadas las relaciones entre las variables mediante los contrastes de hipótesis y las correlaciones calcularemos diferentes modelos de regresión logística para comprobar qué regresores son realmente significativos y que efecto tuvieron sobre la variable dependiente dicotómica **Survived**. Con este modelo podremos dar respuesta a diferentes preguntas que nos podemos plantear como son las siguientes:

- ¿Los pasajeros de primera clase tuvieron más probabilidad de sobrevivir que los de tercera clase?
- ¿Se dio preferencia a las mujeres antes que a los hombres para ser salvados?
- ¿Influyó el puerto de embarque de cada pasajero en el hecho de que sobreviviera al accidente?

Finalmente calcularemos unos árboles de decisión (modelos supervisados) para predecir la variable objetivo **Survived** a partir del resto de variables del conjunto de datos. Una vez tengamos el mejor modelo, lo utilizaremos para predecir la variable objetivo en el conjunto de datos de tests y presentar los resultados a la competición de Kaggle.

4.2. Comprobación de la normalidad y homogeneidad de la varianza

Para comprobar la normalidad de las variable cuantitativas vamos a utilizar el test de *Shapiro-Wilk*, ya que se considera uno de los métodos más potentes para contrastar la normalidad. Este método se basa en el contraste de hipótesis, asumiendo como hipótesis nula que la población sigue una distribución normal. Por tanto, nos basaremos en el *p-valor* para determinar si aceptamos o rechazamos la hipótesis nula de normalidad.

```
# Test de normalidad de las variables cuantitativas
shapiro.test(titanic_data$Age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_data$Age
## W = 0.96548, p-value = 1.118e-13
```

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

```
##
##  Shapiro-Wilk normality test
```

```
##  
## data:  titanic_data$SibSp  
## W = 0.51297, p-value < 2.2e-16
```

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

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

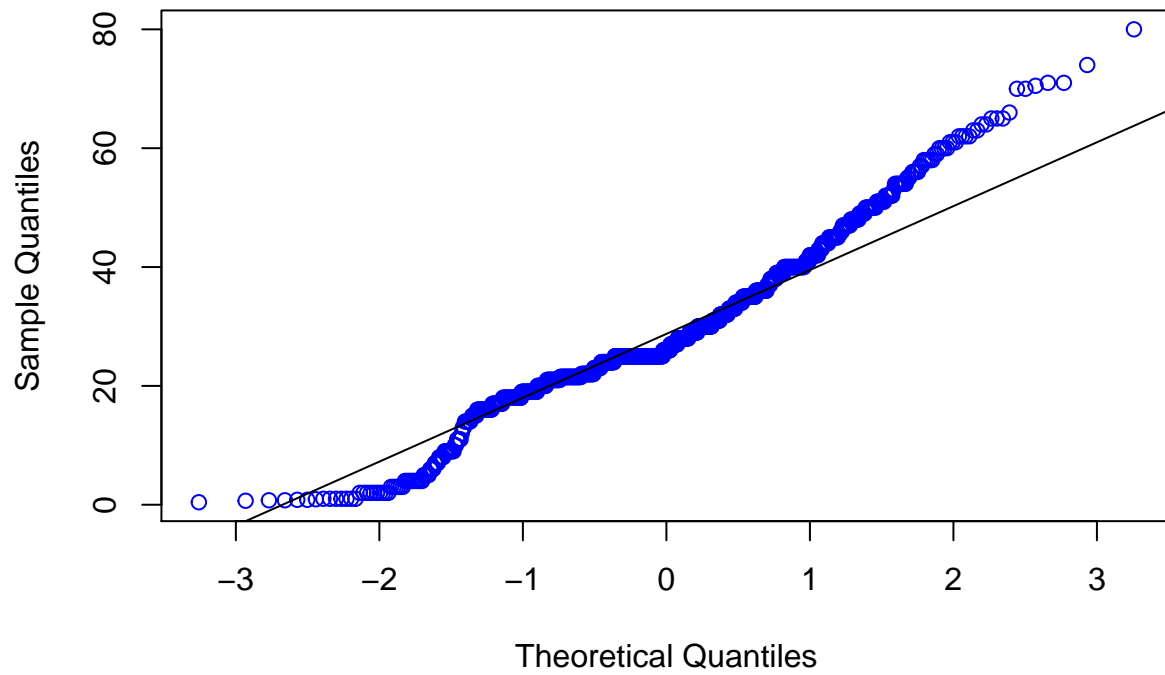
```
shapiro.test(titanic_data$Fare)
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  titanic_data$Fare  
## W = 0.78834, p-value < 2.2e-16
```

Si usamos un nivel de significancia de $\alpha = 0.05$, podemos ver que en todos los tests anteriores se rechaza la hipótesis nula con un nivel de confianza del 95%, ya que $p_valor < 0.05$ en todos los casos. Por tanto, podemos decir que las variables cuantitativas de este conjunto de datos (**Age**, **SibSp**, **Parch** y **Fare**) no siguen una distribución normal. Vamos a comparar las distribuciones de estas variables con la de una normal de forma visual mediante un gráfico QQ.

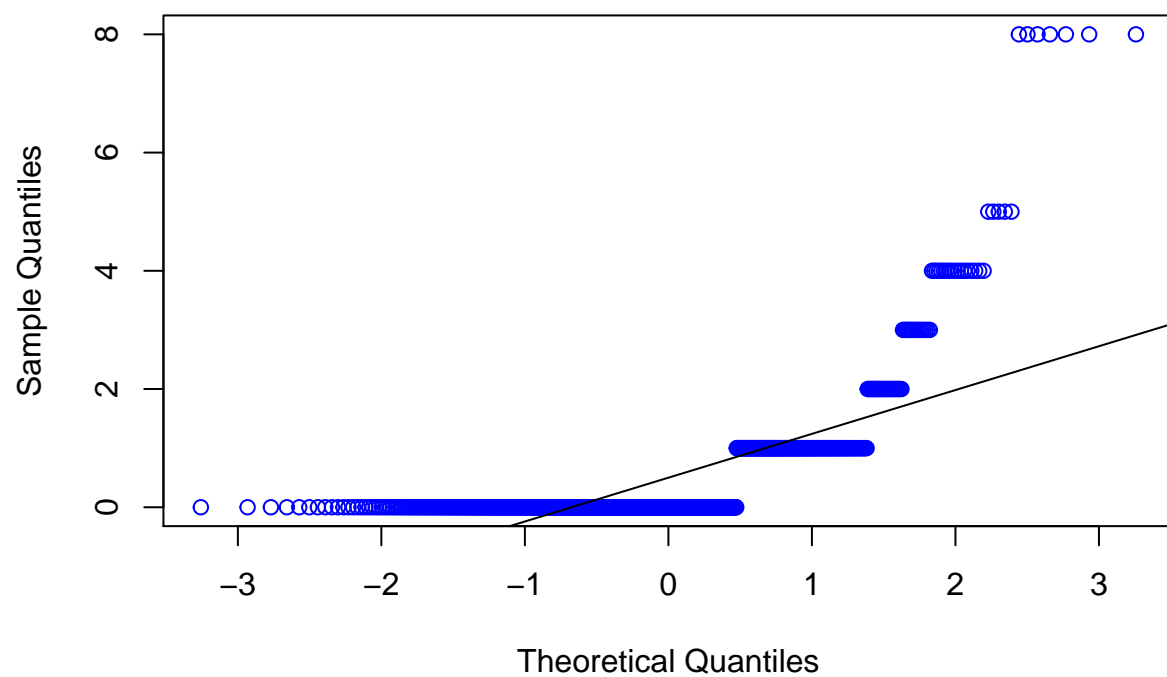
```
# Gráficos QQ de las variables cuantitativas  
qqnorm(titanic_data$Age, main = "QQ plot Age", col = 'blue')  
qqline(titanic_data$Age)
```

QQ plot Age



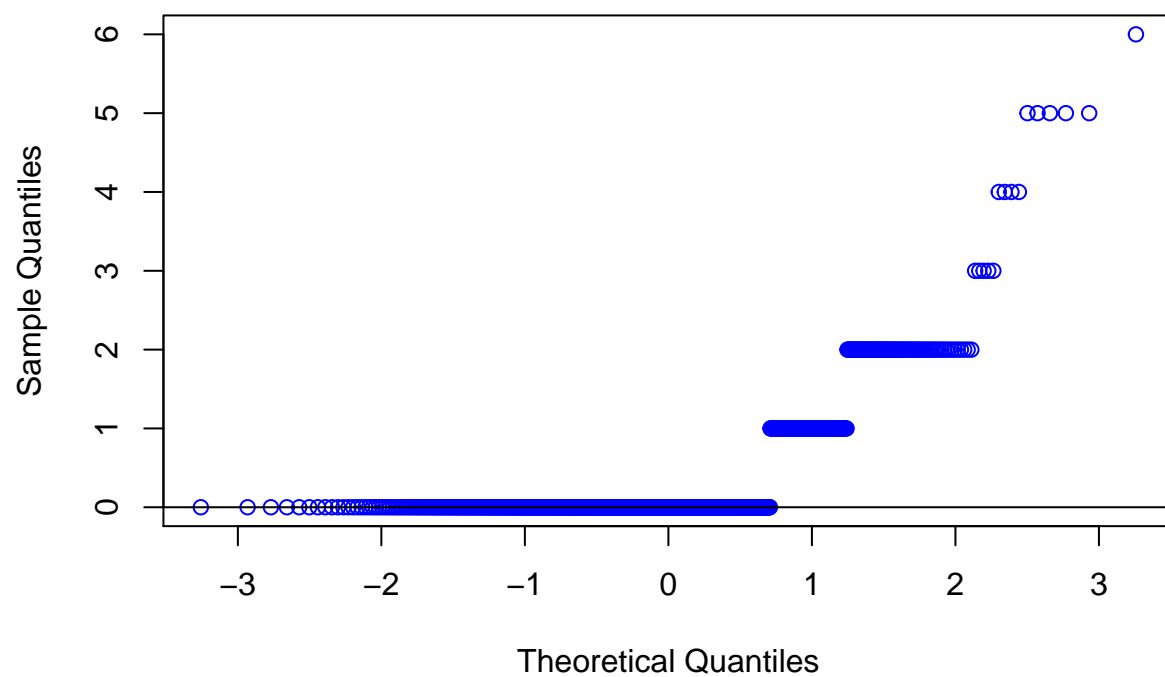
```
qqnorm(titanic_data$SibSp, main = "QQ plot SibSp", col = 'blue')  
qqline(titanic_data$SibSp)
```

QQ plot SibSp

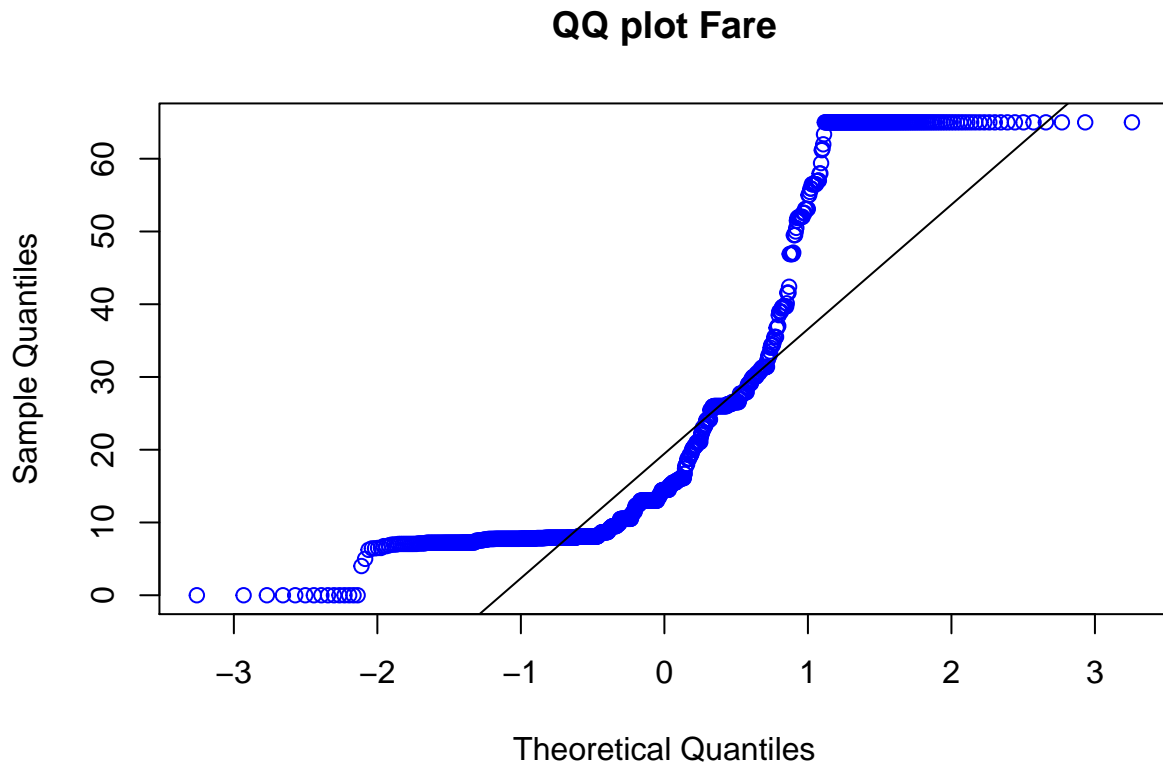


```
qqnorm(titanic_data$Parch, main = "QQ plot Parch", col = 'blue')  
qqline(titanic_data$Parch)
```

QQ plot Parch



```
qqnorm(titanic_data$Fare, main = "QQ plot Fare", col = 'blue')  
qqline(titanic_data$Fare)
```

Comprobamos que, como hemos obtenido con los test de *Shapiro-Wilk*, las variables cuantitativas no siguen una distribución normal. Vemos que la variable **Age** se encuentra cerca de la normalidad, por tanto, como tenemos una cantidad de observaciones suficientemente grande podemos asumir que esta variable sigue una distribución normal basándonos en el teorema central del límite.

Homocedasticidad

Ahora vamos a comprobar la homocedasticidad para las variables anteriores diferenciando en distintos grupos. Para la variable **Age**, como hemos supuesto normalidad, usaremos el test de Levene (paramétrico) y para el resto de variables usaremos el test de Fligner-Killeen (no paramétrico). Ambas pruebas realizan un contraste de hipótesis donde la hipótesis nula asume igualdad de varianzas en los diferentes grupos de datos.

```
# Comprobación de la homocedasticidad para Age
```

```
leveneTest(Age ~ Pclass, data = titanic_data)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value    Pr(>F)
## group  2  16.561 8.684e-08 ***
##      888
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
leveneTest(Age ~ Sex, data = titanic_data)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group  1  0.4543 0.5005
##      889
```

```
leveneTest(Age ~ Embarked, data = titanic_data)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value    Pr(>F)
## group  2  8.6963 0.0001819 ***
##      888
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
leveneTest(Age ~ Survived, data = titanic_data)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value    Pr(>F)
## group  1  5.2188 0.02258 *
##      889
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

De los resultados de los tests de Levene podemos decir que la variable **Age** presenta heterocedasticidad con las variables **Pclass**, **Embarked** y **Survived**, y homocedasticidad con la variable **Sex**. Por tanto, la variable **Age** tendrá varianzas iguales entre los pasajeros de distintos sexo, en cambio, tendrás distinta varianza entre los pasajeros que sobrevivieron y los que no, entre las distintas clases y entre los distintos puertos de embarque.

Para comprobar la homocedasticidad en las variables **SibSp** y **Parch** vamos a sumarlas y usar una variable que indique el número de familiares que iban a bordo del Titanic.

```
# Comprobación de la homocedasticidad para SibSp+Parch
fligner.test(SibSp+Parch ~ Pclass, data = titanic_data)
```

```
##
## Fligner-Killeen test of homogeneity of variances
##
## data:  SibSp + Parch by Pclass
## Fligner-Killeen:med chi-squared = 0.041221, df = 2, p-value = 0.9796
```

```
fligner.test(SibSp+Parch ~ Sex, data = titanic_data)
```

```
##
## Fligner-Killeen test of homogeneity of variances
##
## data:  SibSp + Parch by Sex
## Fligner-Killeen:med chi-squared = 58.958, df = 1, p-value = 1.61e-14
```

```
fligner.test(SibSp+Parch ~ Embarked, data = titanic_data)
```

```
##
## Fligner-Killeen test of homogeneity of variances
##
## data:  SibSp + Parch by Embarked
## Fligner-Killeen:med chi-squared = 5.0719, df = 2, p-value = 0.07919
```

```
fligner.test(SibSp+Parch ~ Survived, data = titanic_data)
```

```
##
## Fligner-Killeen test of homogeneity of variances
##
## data:  SibSp + Parch by Survived
## Fligner-Killeen:med chi-squared = 19.647, df = 1, p-value = 9.317e-06
```

Como podemos ver el tamaño de la familia tendrá varianzas iguales (homocedasticidad) en las diferentes clases (Pclass) y en el puerto de embarque (Embarked). En cambio, presenta varianzas distintas (heterocedasticidad) entre hombres y mujeres (Sex) y entre los pasajeros que sobrevivieron y los que no (Survived).

Para terminar con la comprobación de la homocedasticidad vamos a hacerlo para la variable Fare.

```
# Comprobación de la homocedasticidad para Fare
fligner.test(Fare ~ Pclass, data = titanic_data)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: Fare by Pclass
## Fligner-Killeen:med chi-squared = 130.07, df = 2, p-value < 2.2e-16

fligner.test(Fare ~ Sex, data = titanic_data)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: Fare by Sex
## Fligner-Killeen:med chi-squared = 26.388, df = 1, p-value = 2.793e-07

fligner.test(Fare ~ Embarked, data = titanic_data)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: Fare by Embarked
## Fligner-Killeen:med chi-squared = 81.595, df = 2, p-value < 2.2e-16

fligner.test(Fare ~ Survived, data = titanic_data)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: Fare by Survived
## Fligner-Killeen:med chi-squared = 55.128, df = 1, p-value = 1.129e-13
```

Vemos que la variable Fare presenta heterocedasticidad entre las distintas clases (Pclass), entre hombre y mujeres (Sex), entre los distintos puertos de embarque (Embarked) y entre los pasajeros que sobrevivieron y los que no (Survived).

4.3. Aplicación de pruebas estadísticas

4.3.1. Contraste de hipótesis sobre Survived

A continuación vamos a ver la relación de la variable Survived con el resto de variables que estamos analizando, tanto cuantitativas como categóricas. Hemos visto que la variable Age, aunque es la única para la que se ha asumido normalidad, presenta heterocedasticidad con los distintos grupos de la variable Survived. Por tanto, para todas las variables cuantitativas tendremos que utilizar pruebas no paramétricas como Wilcoxon (datos dependientes) o Mann-Whitney (datos independientes). Estas dos pruebas se aplican indistintamente con la función wilcox.test().

```
# Relación de Survived con las variable cuantitativas
wilcox.test(Age ~ Survived, data = titanic_data)
```

```
##
## Wilcoxon rank sum test with continuity correction
```

```
##
## data: Age by Survived
## W = 98172, p-value = 0.25
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(SibSp+Parch ~ Survived, data = titanic_data)
```

```
##
## Wilcoxon rank sum test with continuity correction
##
## data: SibSp + Parch by Survived
## W = 77659, p-value = 7.971e-07
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(Fare ~ Survived, data = titanic_data)
```

```
##
## Wilcoxon rank sum test with continuity correction
##
## data: Fare by Survived
## W = 58171, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

De acuerdo a los resultados anteriores podemos decir que se observan diferencias estadísticamente significativas de los pasajeros que sobrevivieron y los que no en el tamaño de la familia (**SibSp+Parch**) y en el precio del ticket (**Fare**). En cambio, no se presentan diferencias significativas en la edad (**Age**).

Para comparar si existen diferencias significativas en las variable categóricas entre los pasajeros que sobrevivieron y los que no vamos a aplicar el test de χ^2 .

```
# Relación de Survived con las variable categóricas
table.Pclass <- table(titanic_data$Pclass, titanic_data$Survived)
table.Pclass
```

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

```
chisq.test(table.Pclass)
```

```
##
## Pearson's Chi-squared test
##
## data: table.Pclass
## X-squared = 102.89, df = 2, p-value < 2.2e-16
```

```
table.Sex <- table(titanic_data$Sex, titanic_data$Survived)
table.Sex
```

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

```
chisq.test(table.Sex)
```

```
##
## Pearson's Chi-squared test with Yates' continuity correction
```

```
##
## data: table.Sex
## X-squared = 260.72, df = 1, p-value < 2.2e-16
table.Embarked <- table(titanic_data$Embarked, titanic_data$Survived)
table.Embarked

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

chisq.test(table.Embarked)

## Warning in chisq.test(table.Embarked): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: table.Embarked
## X-squared = NaN, df = 3, p-value = NA
```

Con estos resultados comprobamos que se rechaza la hipótesis nula para los tres casos, por tanto, observamos que tanto el sexo (**Sex**), como la clase (**Pclass**) y el puerto de embarque (**Embarked**) tuvieron repercusión en si un pasajero finalmente sobrevivió al accidente o no.

4.3.2. Análisis de correlaciones

Vamos a calcular los coeficientes de correlación entre las variables cuantitativas para comprobar cuales están relacionadas linealmente entre sí. Para ello vamos a utilizar la correlación de *Spearman* (no paramétrica) para medir el grados de dependencia entre las variables.

```
# Correlación de Spearman entre variables cuantitativas
cor.test(titanic_data$Age, titanic_data$SibSp, method = 'spearman')

## Warning in cor.test.default(titanic_data$Age, titanic_data$SibSp, method =
## "spearman"): Cannot compute exact p-value with ties

##
## Spearman's rank correlation rho
##
## data: titanic_data$Age and titanic_data$SibSp
## S = 137482216, p-value = 6.096e-07
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## -0.166179

cor.test(titanic_data$Age, titanic_data$Parch, method = 'spearman')

## Warning in cor.test.default(titanic_data$Age, titanic_data$Parch, method =
## "spearman"): Cannot compute exact p-value with ties

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

```

## data:  titanic_data$Age and titanic_data$Parch
## S = 145651575, p-value = 1.085e-12
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## -0.2354747

cor.test(titanic_data$Age, titanic_data$Fare, method = 'spearman')

## Warning in cor.test.default(titanic_data$Age, titanic_data$Fare, method =
## "spearman"): Cannot compute exact p-value with ties
##
## Spearman's rank correlation rho
##
## data:  titanic_data$Age and titanic_data$Fare
## S = 97357228, p-value = 1.678e-07
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.1741772

cor.test(titanic_data$SibSp, titanic_data$Parch, method = 'spearman')

## Warning in cor.test.default(titanic_data$SibSp, titanic_data$Parch, method =
## "spearman"): Cannot compute exact p-value with ties
##
## Spearman's rank correlation rho
##
## data:  titanic_data$SibSp and titanic_data$Parch
## S = 64838502, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.450014

cor.test(titanic_data$SibSp, titanic_data$Fare, method = 'spearman')

## Warning in cor.test.default(titanic_data$SibSp, titanic_data$Fare, method =
## "spearman"): Cannot compute exact p-value with ties
##
## Spearman's rank correlation rho
##
## data:  titanic_data$SibSp and titanic_data$Fare
## S = 64642686, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.451675

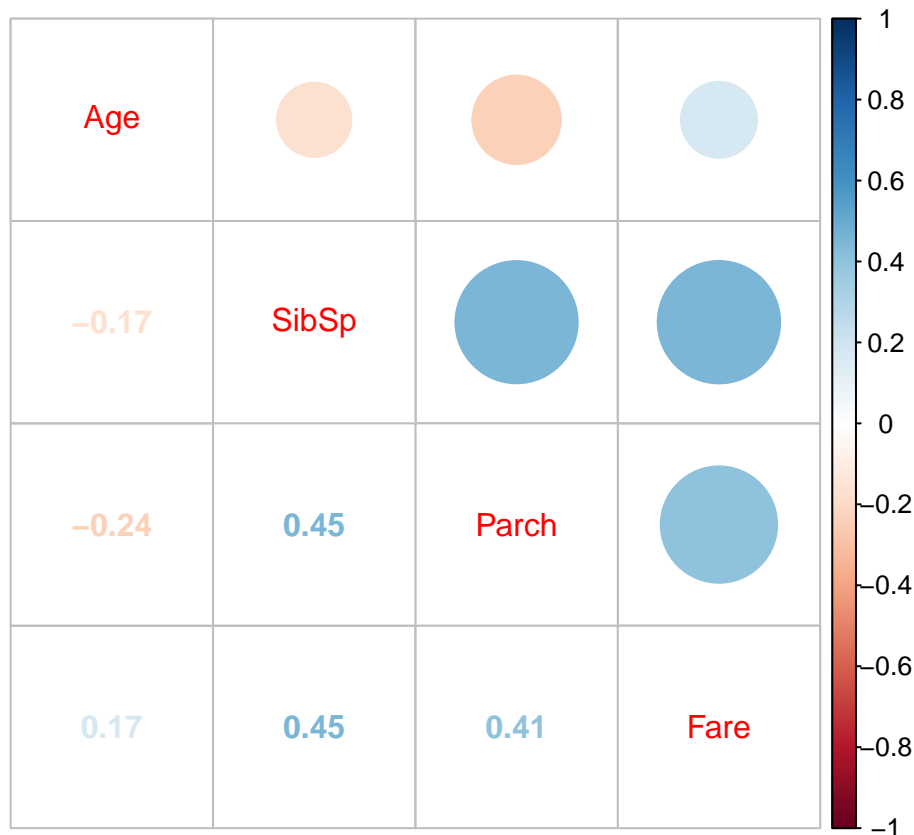
cor.test(titanic_data$Parch, titanic_data$Fare, method = 'spearman')

## Warning in cor.test.default(titanic_data$Parch, titanic_data$Fare, method =
## "spearman"): Cannot compute exact p-value with ties
##
## Spearman's rank correlation rho

```

```
##
## data:  titanic_data$Parch and titanic_data$Fare
## S = 69798308, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.4079429
```

```
corrplot.mixed(cor(titanic_data[, c('Age', 'SibSp', 'Parch', 'Fare')],
                  method = 'spearman'))
```



Como podemos ver, ninguna de las variable está relacionada con las demás ya que el coeficiente de correlación es menor de $|0.5|$ en todos los casos. Por tanto, podemos decir que las variables cuantitativas son independientes entre ellas. Además, el p-valor nos indica que los resultados obtenidos son significativos.

4.3.3. Regresión logística

A continuación vamos a calcular un modelo de regresión logística para comprobar qué variables tienen un efecto significativo sobre la variable dicotómica dependiente **Survived**, es decir, sobre la probabilidad de sobrevivir o no al accidente.

```
# Regresión logística con todas las variables
regression.logist <- glm(Survived ~ Pclass+Sex+Age+SibSp+Parch+Fare+Embarked,
                        data = titanic_data, family = binomial(link="logit"))
summary(regression.logist)
```

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

```
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp + Parch +
##      Fare + Embarked, family = binomial(link = "logit"), data = titanic_data)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.7057  -0.5975  -0.4054   0.6037   2.4930
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.704565   0.585065   6.332 2.42e-10 ***
## Pclass2      -0.727659   0.349697  -2.081 0.037450 *
## Pclass3     -1.892273   0.390078  -4.851 1.23e-06 ***
## Sexmale     -2.691545   0.202947 -13.262 < 2e-16 ***
## Age         -0.043498   0.008278  -5.254 1.48e-07 ***
## SibSp       -0.433336   0.119689  -3.621 0.000294 ***
## Parch       -0.148356   0.123292  -1.203 0.228865
## Fare         0.017067   0.008158   2.092 0.036427 *
## EmbarkedQ   -0.102773   0.385207  -0.267 0.789624
## EmbarkedS   -0.362447   0.241639  -1.500 0.133627
## ---
## 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:  777.19  on 881  degrees of freedom
## AIC: 797.19
##
## Number of Fisher Scoring iterations: 5
```

Como podemos ver en el modelo resultante, las variables independientes (regresores) más significativas son Pclass, Sex, Age y SibSp. Por tanto, vamos a utilizar estas variables para ir construyendo modelos de regresión logística y analizaremos la bondad de cada modelo con la medida AIC.

```
# Regresión logística con las variables independientes más significativas
regression.1 <- glm(Survived ~ Pclass, data = titanic_data,
                    family = binomial(link="logit"))
regression.2 <- glm(Survived ~ Pclass+Sex, data = titanic_data,
                    family = binomial(link="logit"))
regression.3 <- glm(Survived ~ Pclass+Sex+Age, data = titanic_data,
                    family = binomial(link="logit"))
regression.4 <- glm(Survived ~ Pclass+Sex+Age+SibSp, data = titanic_data,
                    family = binomial(link="logit"))

out <- data.frame("Regression model" = c(1:4),
                  "AIC" = c(regression.1$aic, regression.2$aic,
                             regression.3$aic, regression.4$aic),
                  check.names = FALSE)
out %>% kable() %>% kable_styling(latex_options = 'hold_position')
```

Para evaluar los modelos usando el AIC debemos saber que cuanto menor es el valor de esta medida mejor será el modelo. Por tanto, el mejor modelo de regresión logística es el que utiliza todos los regresores significativos y que obtiene un $AIC = 797.52$. Vamos a ver a continuación los coeficientes que se obtienen con este modelo y si los regresores siguen siendo significativos.

Regression model	AIC
1	1089.1080
2	834.8884
3	811.5976
4	797.5173

```
# Mejor modelo de regresión logística
summary(regression.4)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp, family = binomial(link = "logit"),
##      data = titanic_data)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.7722  -0.5946  -0.4018   0.6190   2.4760
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.288748   0.422592  10.149 < 2e-16 ***
## Pclass2     -1.307289   0.268038  -4.877 1.08e-06 ***
## Pclass3     -2.562796   0.260370  -9.843 < 2e-16 ***
## Sexmale     -2.712718   0.194733 -13.930 < 2e-16 ***
## Age         -0.045127   0.008172  -5.522 3.35e-08 ***
## SibSp       -0.377545   0.104500  -3.613 0.000303 ***
## ---
## 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:  785.52  on 885  degrees of freedom
## AIC: 797.52
##
## Number of Fisher Scoring iterations: 5
```

```
# Cálculo de los odd-ratios
exp(coefficients(regression.4))
```

```
## (Intercept)      Pclass2      Pclass3      Sexmale      Age      SibSp
## 72.87516762  0.27055243  0.07708892  0.06635624  0.95587640  0.68554243
```

Vemos que los regresores presentan un p-valor que indica que siguen siendo estadísticamente significativos para el modelo. Si nos fijamos en los *odd-ratios* obtenidos a partir de los coeficientes vemos que la variable Age tiene un OR muy próximo a 1, por tanto podemos decir que no existe asociación entre la edad y la probabilidad de sobrevivir. En cambio, las otras variables independientes presentan un OR menor que 1 por lo que los consideramos factores de riesgo para sobrevivir al accidente. Esto quiere decir que viajar en segunda y tercera clase influye negativamente en la probabilidad de sobrevivir (coeficientes con signo negativo, -1.31 y -2.56, respectivamente), al igual que sucede con los pasajeros de sexo masculino (-2.71). Por último, la variable SibSp (tiene hermanos/as y marido/esposa en el barco) también tiene una influencia negativa en el hecho de sobrevivir, aunque menor que las anteriores (-0.38).

4.3.4. Modelo supervisado

Preparación de los datos de entrenamiento y test

Como se comentó en el primer apartado, los datos originales obtenidos de Kaggle ya estaban divididos en datos de entrenamiento y test. El dataset de entrenamiento es el único que contiene la variable objetivo `Survived`. Por eso vamos a trabajar únicamente con este conjunto de datos para poder observar la precisión del modelo creado.

Por tanto, dividimos el conjunto de datos de entrenamiento en dos: 2/3 de los datos se enfocarán a entrenamiento y 1/3 para test. se realiza la división mediante el método de exclusión con partición estratificada para que la proporción de pasajeros que sobreviven frente a los que no sea similar en ambos conjunto de datos. Utilizamos la función `holdout` de la librería `rminer`.

```
# Hacemos uso de seed para que el resultado sea reproducible
set.seed(10)

# Método de exclusión para particionar el conjunto de datos
h <- holdout(titanic_data$Survived, ratio=2/3, mode="stratified")
titanic_train <- titanic_data[h$str, ]
titanic_test <- titanic_data[h$ts, ]
```

Comprobamos que la variable objetivo se ha repartido de manera similar en ambos conjuntos de datos.

```
# Repartición de la variable Survived en ambos subconjuntos
prop.table(table(titanic_train$Survived))
```

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

```
prop.table(table(titanic_test$Survived))
```

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

Comprobamos que la proporción es exactamente la misma.

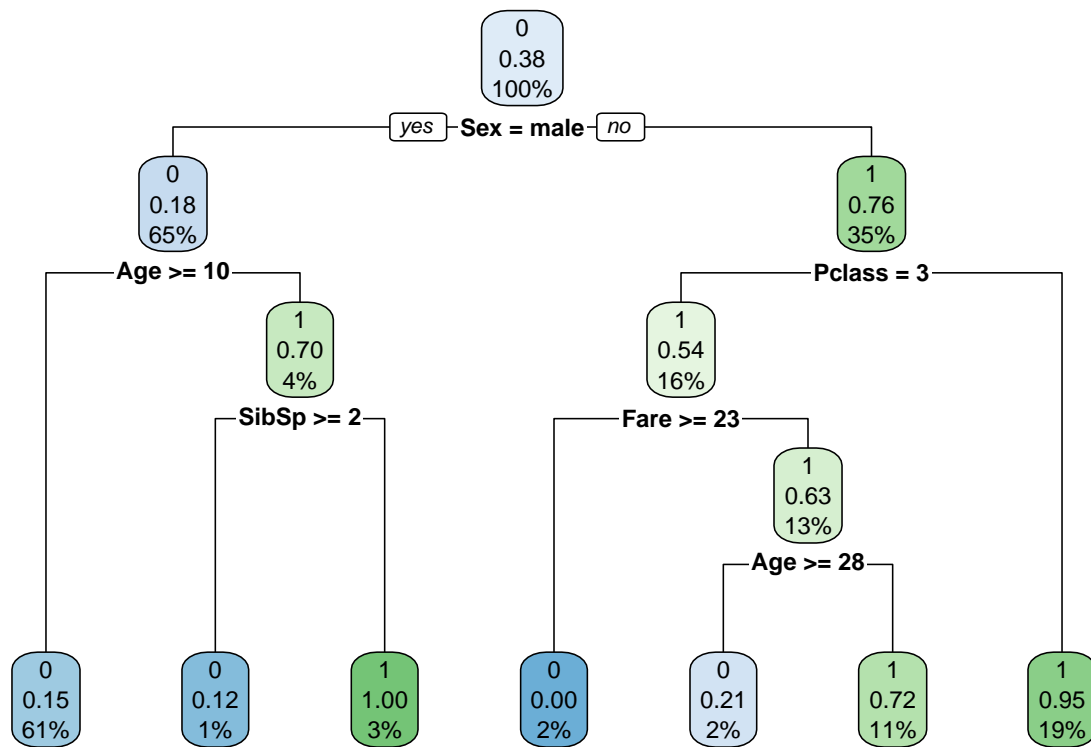
Árbol de decisión

Entrenamos el árbol de decisión utilizando la función `rpart()`.

```
# Calculamos el árbol de decisión con todas las variables
set.seed(20)
mod_dt <- rpart(Survived ~ ., data = titanic_train, method = 'class')
```

Mostramos el árbol de decisión calculado.

```
rpart.plot(mod_dt)
```



Las variables Age, SibSp, Fare, Pclass y Sex son utilizadas para generar el árbol de decisión.

Para saber cómo de bueno es el modelo, predecimos la supervivencia de los pasajeros del conjunto de datos de test.

```
set.seed(30)
pred_dt <- predict(mod_dt, newdata = titanic_test, type = "class")
```

Mostramos la matriz de confusión.

```
confusionMatrix(pred_dt, titanic_test$Survived, positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0    1
##           0 165  45
##           1  18  69
##
##               Accuracy : 0.7879
##               95% CI : (0.7369, 0.833)
##       No Information Rate : 0.6162
##       P-Value [Acc > NIR] : 1.721e-10
##
##               Kappa : 0.5306
##
##  Mcnemar's Test P-Value : 0.001054
##
```

```
##           Sensitivity : 0.6053
##           Specificity : 0.9016
##           Pos Pred Value : 0.7931
##           Neg Pred Value : 0.7857
##           Prevalence : 0.3838
##           Detection Rate : 0.2323
##           Detection Prevalence : 0.2929
##           Balanced Accuracy : 0.7535
##
##           'Positive' Class : 1
##
```

La precisión que nos da el modelo es de 78,19 %. El 78,19% de los registros han sido correctamente clasificados.

La función `confusionMatrix()` también nos facilita otros valores como la sensibilidad (tasa de verdaderos positivos) o la especificidad (tasa de verdaderos negativos). Estos valores son 60,53% y 90,16% respectivamente. Es decir, el modelo es mejor prediciendo correctamente registros negativos que positivos.

Random Forest

Calculamos un modelo Random Forest con una validación cruzada con 4 *folds*.

Mediante la función `train_control`, especificamos las características comentadas del proceso de entrenamiento para crear el modelo.

```
train_control<- trainControl(method="cv", number=4)
```

Creamos el modelo.

```
set.seed(31)
mod_rf<-train(Survived~., data=titanic_train, method="rf", trControl = train_control)
```

Predecimos la variable objetivo con los datos del conjunto de test.

```
set.seed(32)
pred_rf <- predict(mod_rf, newdata=titanic_test)
```

Observamos la matriz de confusión.

```
confusionMatrix(pred_rf,titanic_test$Survived,positive="1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0    1
##           0 161  37
##           1  22  77
##
##           Accuracy : 0.8013
##           95% CI : (0.7514, 0.8452)
##           No Information Rate : 0.6162
##           P-Value [Acc > NIR] : 4.938e-12
##
##           Kappa : 0.5693
##
##           McNemar's Test P-Value : 0.06836
##
##           Sensitivity : 0.6754
##           Specificity : 0.8798
```

```
##          Pos Pred Value : 0.7778
##          Neg Pred Value : 0.8131
##          Prevalence : 0.3838
##          Detection Rate : 0.2593
##          Detection Prevalence : 0.3333
##          Balanced Accuracy : 0.7776
##
##          'Positive' Class : 1
##
```

Con el modelo Random Forest el 80,13 % de los registros de test fueron correctamente clasificados. La precisión de este modelo ha aumentado casi 2% respecto al árbol de decisión.

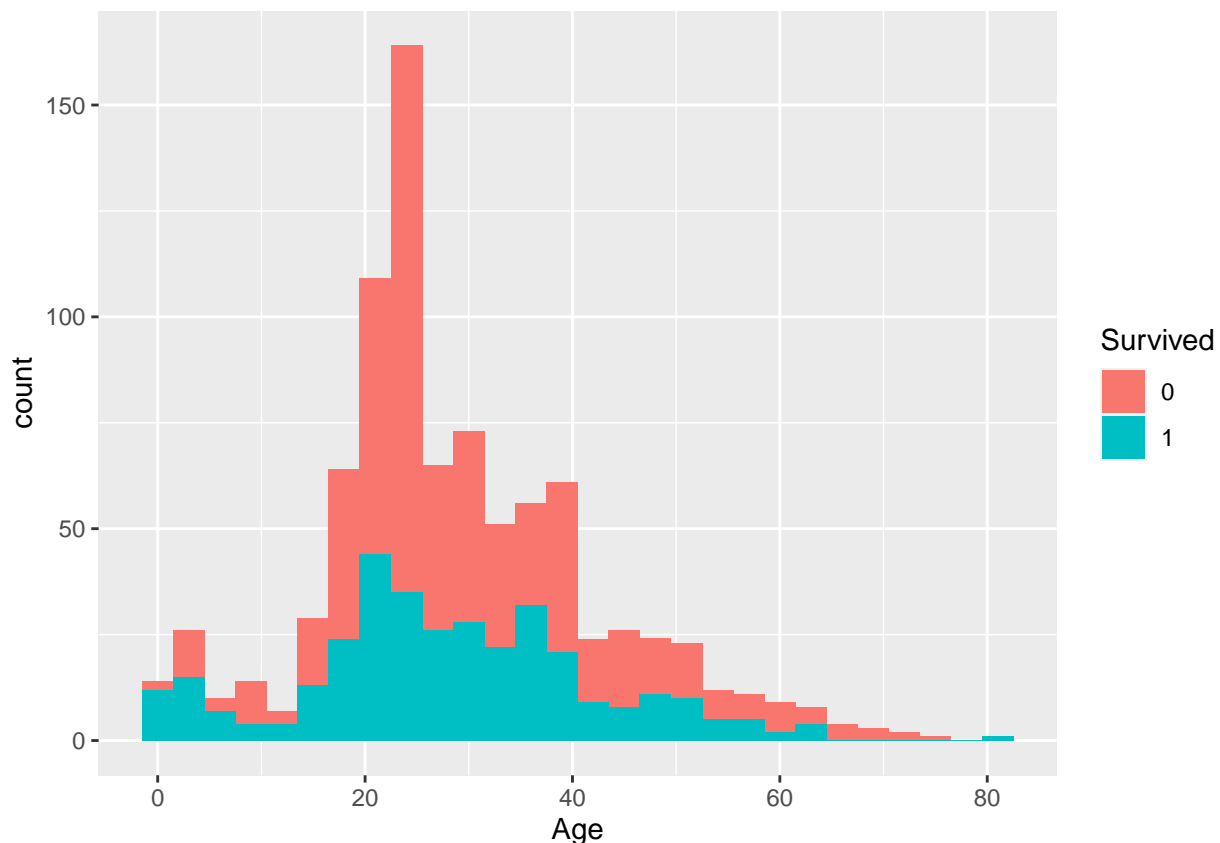
La sensibilidad de este modelo es 67,54% y su especificidad 87,98%. Si comparamos estos valores con los obtenidos mediante el árbol de decisión, se obtiene que el modelo Random Forest clasifica algo peor los pasajeros que no sobrevivieron, pero en cambio, los pasajeros que sí sobrevivieron son clasificados correctamente con un porcentaje bastante mayor.

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

A lo largo de la práctica se ha hecho uso de diferentes tipos de gráficas para comprender mejor los datos y analizarlos. En el apartado anterior por ejemplo, se mostró el árbol de decisión gráficamente para saber que variables utilizaba en la toma de decisiones. También hemos usado gráficos QQ para comprobar la normalidad de algunas variables, diagramas de caja para observar los valores extremos o histogramas para ver la distribución de la variable Age. En este apartado mostraremos algunos ejemplos más para entender mejor el conjunto de datos.

Primeramente se vuelve a mostrar el histograma de la edad de los pasajeros (sin discretizar), pero esta vez añadimos la información si sobreviven o no. Esta vez utilizaremos la función ggplot.

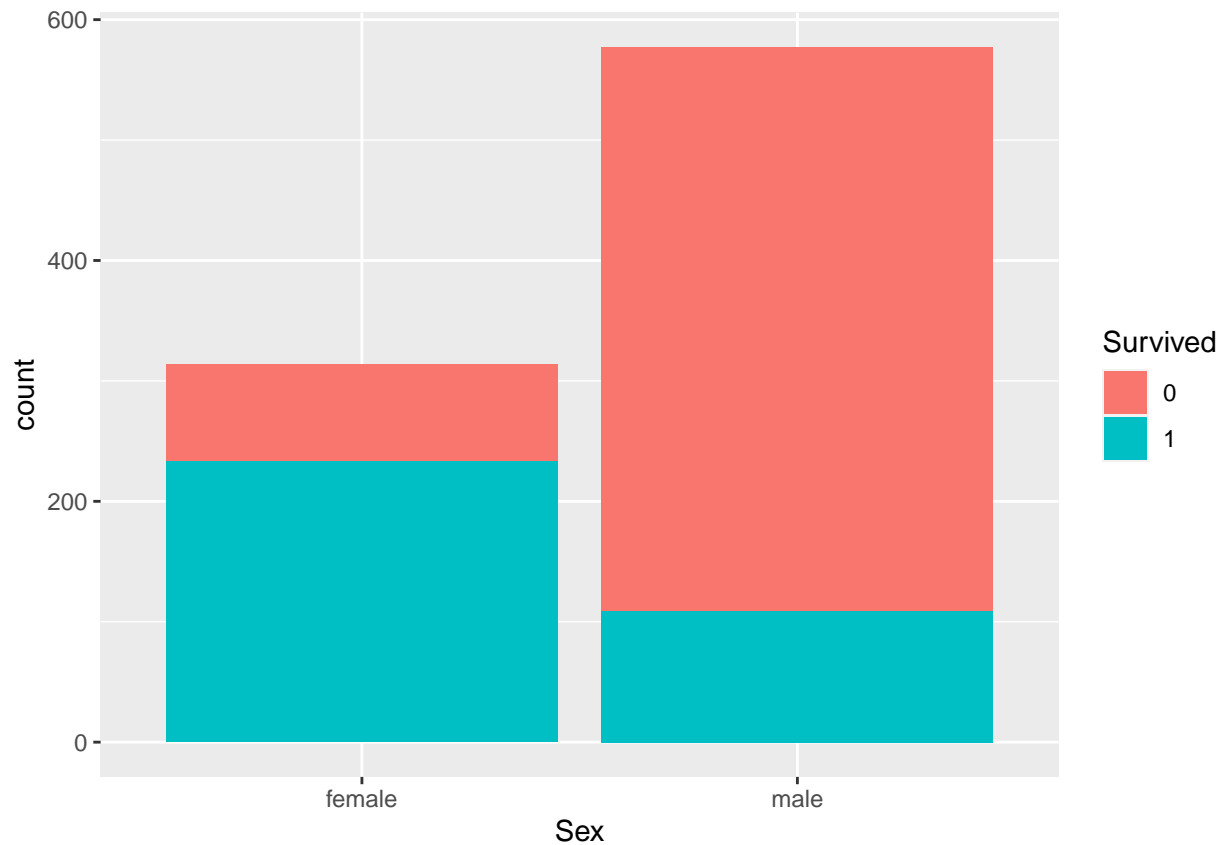
```
ggplot(data = titanic_data,aes(x=Age,fill=Survived))+geom_histogram(binwidth =3)
```



Como ya se observó en su momento, la mayoría de pasajeros se encuentran entre los 20 y cuarenta años. También se observa que la tasa de supervivencia entre los pasajeros menores es bastante alta. Mientras que en los pasajeros entorno a los 25 años sobrevivieron pocos en relación al número total de personas de esa edad (unos 30 de más de 150).

Mostramos ahora un gráfico de barras con el número de pasajeros por edad y la información de si sobrevivieron al accidente.

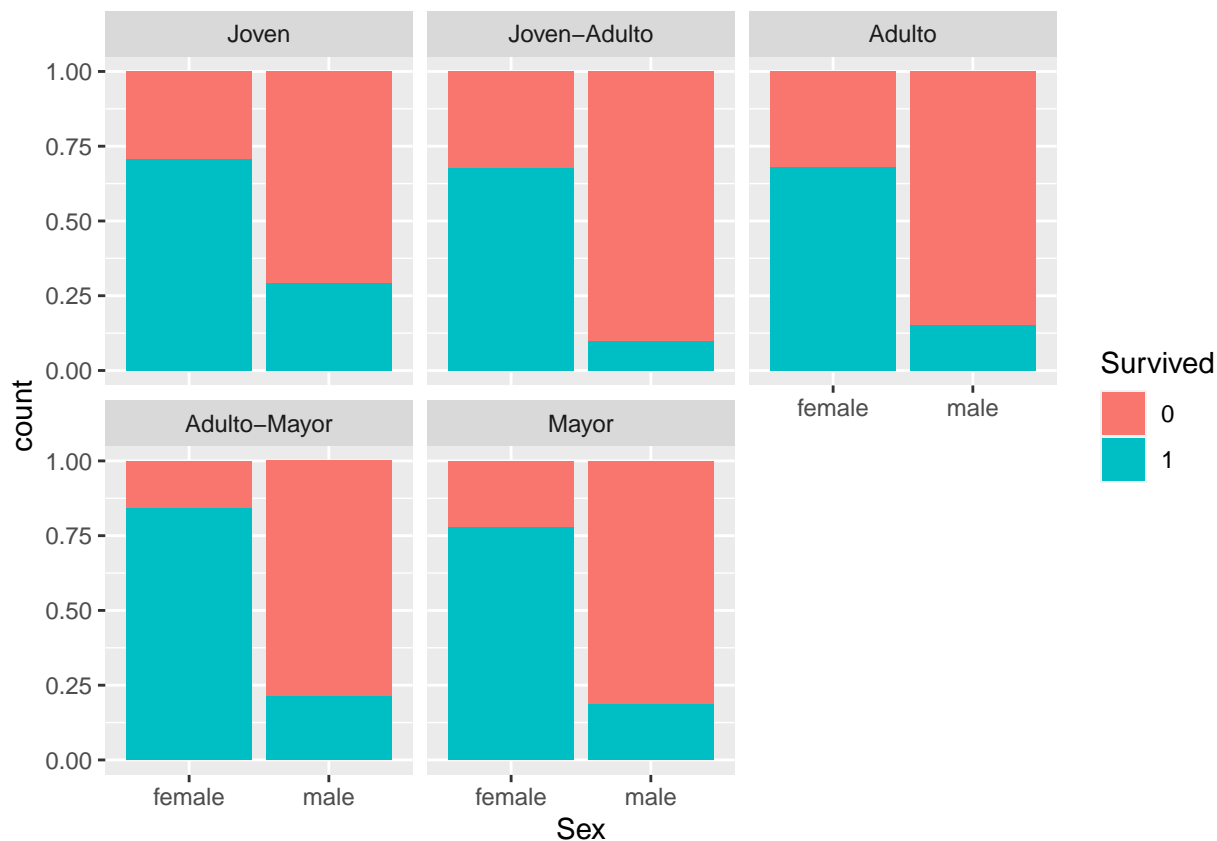
```
ggplot(data=titanic_data,aes(x=Sex,fill=Survived))+geom_bar()
```



El número de pasajeros varones es casi el doble que el de mujeres. Pero la proporción de mujeres que sobrevivieron es mucho mayor que la de hombres.

Para observar con más detalle las proporciones de las gráficas anteriores, mostramos a continuación las tres variables en una serie de gráficas de frecuencia. Pero en este caso utilizaremos la variable **Age** discretizada.

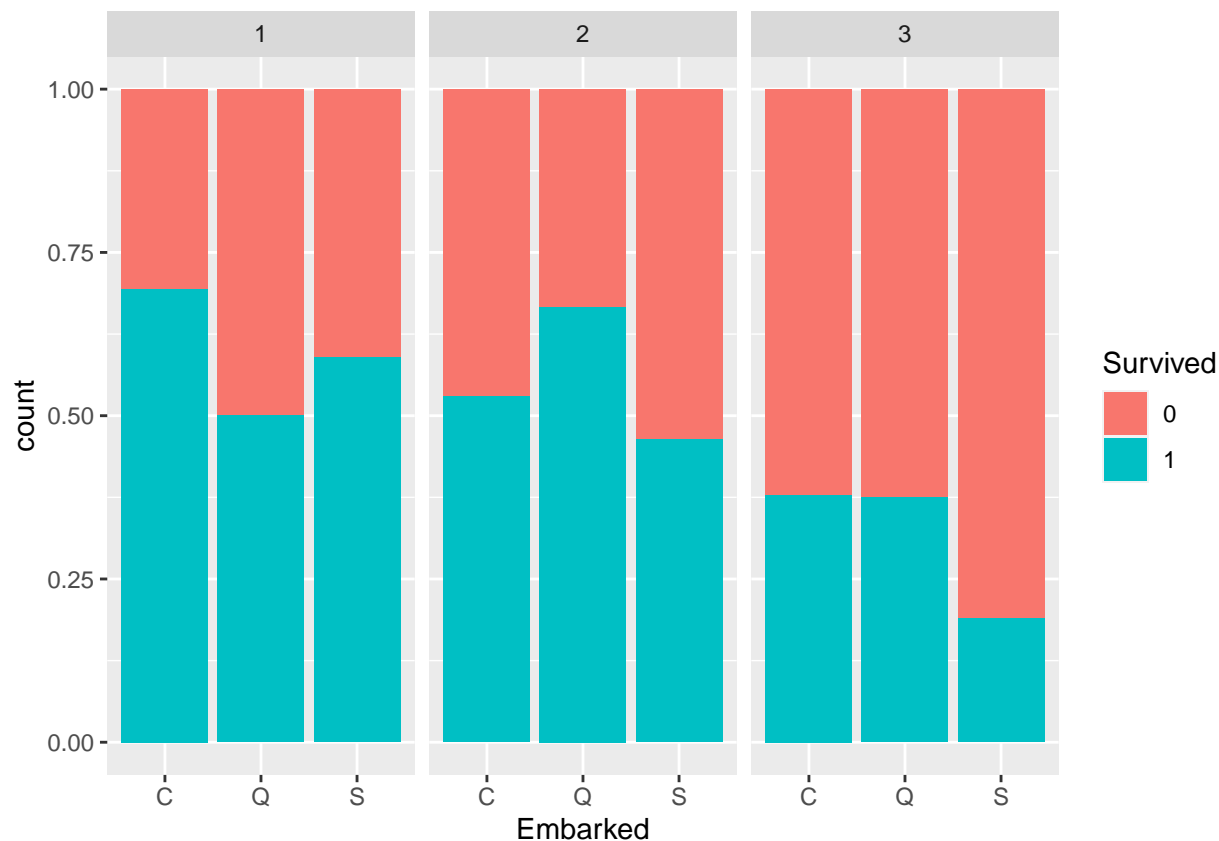
```
ggplot(data = titanic_data, aes(x=Sex, fill=Survived)) + geom_bar(position="fill") + facet_wrap(~Age_d)
```



En todas las edades se observa una gran desigualdad entre el índice de supervivencia si se comparan hombres con mujeres. Las mujeres se mantienen entorno al 75%, mientras que los hombres no llegan casi ni al 25%. Dentro de la edad joven, que comprende pasajeros por debajo de 20 años, es la única en la que la proporción de pasajeros masculinos que sobreviven sobrepasa (por poco) ese 25%. La menor frecuencia de pasajeros salvados tanto en mujeres como en hombres, la encontramos entre la categoría de edad Joven-Adulto. Este nivel de Age abarca pasajeros entre los 20 y los 24 años.

Mostramos una gráfica similar pero este caso nos interesa analizar la supervivencia de los pasajeros respecto a la clase en la que viajaron y el muelle donde embarcaron.

```
ggplot(data = titanic_data, aes(x=Embarked, fill=Survived)) + geom_bar(position="fill") + facet_wrap(~Pclass)
```

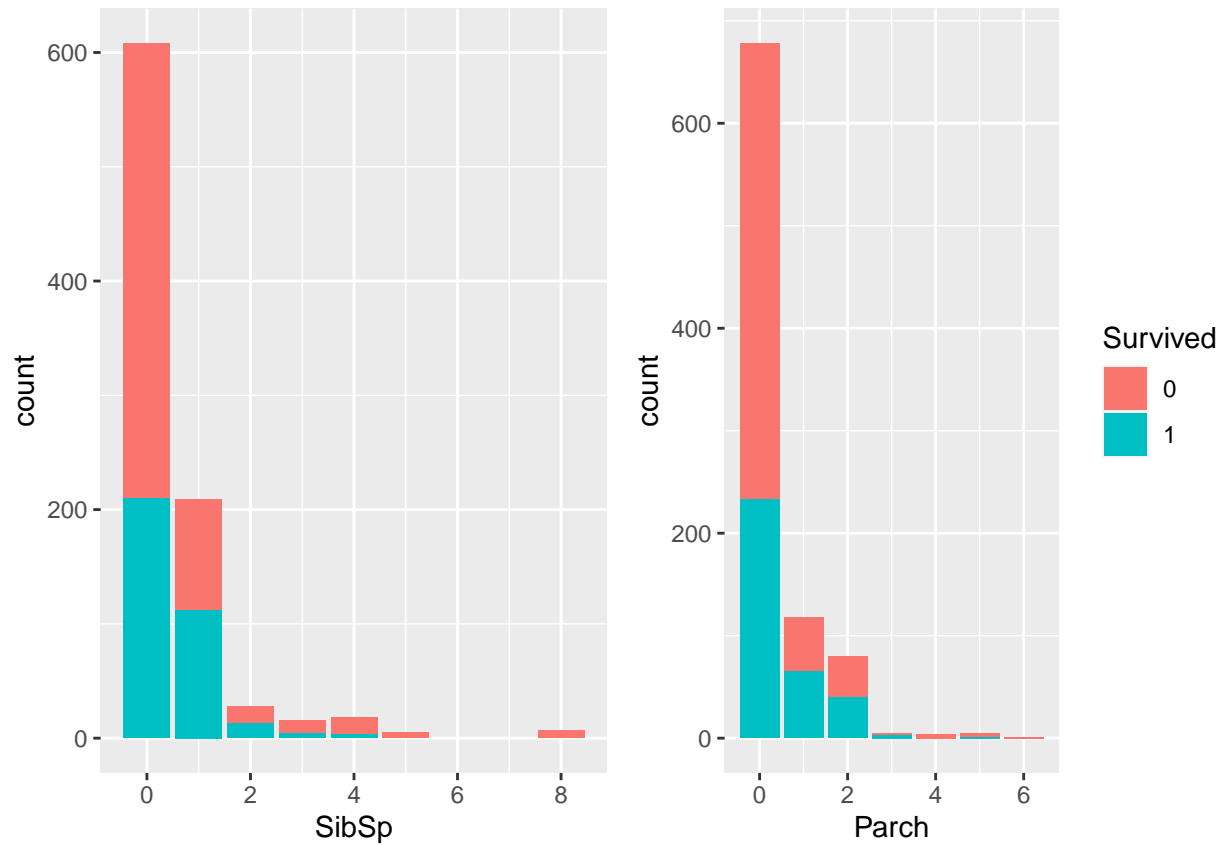



Lo primero que se observa es que la tasa de pasajeros salvados es mucho menor en los pasajeros de tercera clase. En cambio, no existe una gran diferencia entre los de segunda y tercera clase, aún así el índice de supervivencia es algo mayor dentro de primera clase. Otro dato curioso a destacar es que la frecuencia de supervivencia varía bastante dependiendo del muelle de embarque.

comparamos mediante dos diagramas de barras colocados uno junto al otro, la variables que indican la información familiar SibSp y Parch.

```
Graf1 <- ggplot(data = titanic_data,aes(x=SibSp,fill=Survived))+geom_bar()+theme(legend.position="none")
Graf2 <- ggplot(data = titanic_data,aes(x=Parch,fill=Survived))+geom_bar()

grid.arrange(Graf1, Graf2, ncol=2)
```



La mayor parte de los pasajeros viajó sin familiares, pero se observa que la forma de ambas gráficas es similar. Esto podría indicar la presencia de correlación entre ambas variables, pero como vimos en el análisis de correlaciones no existe una correlación significativa entre ninguna variable cuantitativa.

6. Resolución del problema y conclusiones

Con todos los resultados obtenidos ya tenemos la información suficiente para responder a las cuestiones planteadas y predecir el valor de la variable objetivo con una baja tasa de error. A continuación vamos a responder a las preguntas que nos habíamos planteado al inicio del análisis:

- En los contrastes de hipótesis sobre la variable **Survived** hemos visto que se observan diferencias significativas en la variable **Pclass** entre los pasajeros que sobrevivieron y los que no. Después en el modelo de regresión logística hemos comprobado que esta variable tiene significancia en el modelo y que el hecho de ser un pasajero de tercera clase era un factor de riesgo respecto a ser un pasajero de primera clase. Por tanto, podemos afirmar que los pasajeros de primera clase tuvieron más probabilidades de sobrevivir que los de tercera clase.
- Al igual que en la pregunta anterior, el contraste de hipótesis nos dio información de que la variable **Sex** presentaba diferencias significativas entre los pasajeros que sobrevivieron y los que no, y el modelo de regresión logística ha demostrado la significancia de esta variable y que los hombres tuvieron una probabilidad menor de sobrevivir frente a las mujeres.
- En caso del puerto de embarque (**Embarked**), aunque el contraste de hipótesis nos mostraba que había diferencias significativas entre los pasajeros que sobrevivieron y los que no, el modelo de regresión nos ha indicado que esta variable no tiene significancia en el hecho de sobrevivir o no y, por tanto, no influyó en la probabilidad de sobrevivir. Además, el árbol de decisión que hemos calculado posteriormente no ha tenido esta variable en cuenta, por lo que se refuerza esta conclusión.

Con los análisis realizados anteriormente hemos podido dar respuesta a todas las preguntas que nos habíamos planteado y se podría responder a otras muchas. También hemos podido ver la relación entre las distintas variables disponibles en el conjunto de datos y, además, podemos predecir si un pasajero sobrevive o no en función del valor de las variables independientes gracias a un modelo de clasificación supervisado.

Contribuciones	Firma
Investigación previa	Pablo López Ladrón de Guevara, Rafael Corvillo Alonso
Redacción de respuestas	Pablo López Ladrón de Guevara, Rafael Corvillo Alonso
Desarrollo código	Pablo López Ladrón de Guevara, Rafael Corvillo Alonso