Tipologia y ciclo de vida de los datos: Práctica 2

Autor: Sandra Garrido Romero

Diciembre 2019

Table of Contents

Presentación

En esta práctica se elabora un caso práctico orientado a aprender a identificar los datos relevantes para un proyecto analítico y usar las herramientas de integración, limpieza, validación y análisis de las mismas. Para hacer esta práctica tendréis que trabajar en grupos de 2 personas. Tendréis que entregar un solo archivo con el enlace Github (https://github.com) donde se encuentren las soluciones incluyendo los nombres de los componentes del equipo. Podéis utilizar la Wiki de Github para describir vuestro equipo y los diferentes archivos que corresponden a vuestra entrega. Cada miembro del equipo tendrá que contribuir con su usuario Github. Aunque no se trata del mismo enunciado, los siguientes ejemplos de ediciones anteriores os pueden servir como guía: * Ejemplo: https://github.com/Bengis/nba-gap-cleaning * Ejemplo complejo (archivo adjunto).

Competencias

En esta práctica se desarrollan las siguientes competencias del Máster de Data Science:

- Capacidad de analizar un problema en el nivel de abstracción adecuado a cada situación y aplicar las habilidades y conocimientos adquiridos para abordarlo y resolverlo.
- Capacidad para aplicar las técnicas específicas de tratamiento de datos (integración, transformación, limpieza y validación) para su posterior análisis.

Objetivos

Los objetivos concretos de esta práctica son:

- Aprender a aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios o multidisciplinares.
- Saber identificar los datos relevantes y los tratamientos necesarios (integración, limpieza y validación) para llevar a cabo un proyecto analítico.
- Aprender a analizar los datos adecuadamente para abordar la información contenida en los datos.
- Identificar la mejor representación de los resultados para aportar conclusiones sobre el problema planteado en el proceso analítico.
- Actuar con los principios éticos y legales relacionados con la manipulación de datos en función del ámbito de aplicación.
- Desarrollar las habilidades de aprendizaje que les permitan continuar estudiando de un modo que tendrá que ser en gran medida autodirigido o autónomo.
- Desarrollar la capacidad de búsqueda, gestión y uso de información y recursos en el ámbito de la ciencia de datos.

Resolución

Descripción del dataset

El conjunto de datos a utilizar se ha descargado desde este enlace https://www.kaggle.com/c/titanic/data de la página web de *Kaggle*. Este dataset se conforma de 12 variables (columnas) y 1309 pasajeros (filas). El conjunto de datos se ha conseguido uniendo los conjuntos de entrenamiento y test. Los campos del conjunto de datos son los siguientes:

- **PassengerId**: Identificador de cada uno de los pasajeros.
- **Survived**: Indica si el pasajero sobrevivió (1) o no (0).
- **Pclass**: Clase que figuraba en el ticket: primera clase (1), segunda clase (2) o tercera clase (3).

- **Name** : Nombre del pasajero.
- **Sex** : Sexo del pasajero (male, female).
- **Age** : Edad del pasajero.
- **SibSp** : Relaciones familiares del siguiente tipo: hermanos, hermanastros, esposos o esposas.
- **Parch** : Relaciones familiares del siguiente tipo: madre, padre, hijos o hijastros.
- **Ticket** : Número de ticket.
- **Fare** : Precio del ticket.
- **Cabin** : Número de cabina.
- **Embarked** : Puerto de embarque: Cherbourg (C), Queenstown (Q) o Southampton (S).

Importancia y objetivos de los análisis

Este conjunto de datos puede ayudar a resolver qué variables son las que influyen en si un pasajero sobrevivió al hundimiento del Titanic o no.

Integración y selección de los datos

En primer lugar, se va a proceder a leer el fichero de datos obtenido de internet. Se cuenta con dos ficheros CSV, por un lado, el fichero de datos de entrenamiento y por otro, el fichero de datos test. Se van a importar los dos archivos y se van a unir en un sólo *data.frame*:

```
#Lectura del fichero test
test <- read.csv('test.csv',stringsAsFactors = FALSE)
#Letura del fichero entrenamiento
train <- read.csv('train.csv',stringsAsFactors = FALSE)
# Unión de los dos conjuntos importados en uno solo
library(dplyr)
datos <- bind_rows(train,test)
```

Se observa una parte de los datos y el tipo de estos:

```
head(datos)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                 Name    Sex Age SibSp
## 1                             Braund, Mr. Owen Harris   male  22     1
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
## 3                              Heikkinen, Miss. Laina female  26     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1
```

```
## 5                            Allen, Mr. William Henry   male  35     0
## 6                                    Moran, Mr. James   male  NA     0
##   Parch           Ticket    Fare Cabin Embarked
## 1     0        A/5 21171  7.2500              S
## 2     0         PC 17599 71.2833   C85        C
## 3     0 STON/O2. 3101282  7.9250              S
## 4     0           113803 53.1000  C123        S
## 5     0           373450  8.0500              S
## 6     0           330877  8.4583              Q
```

```
str(datos)
```

```
## 'data.frame':    1309 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 Bra
dley (Florence Briggs Thayer)" "Heikkinen, Miss. Laina" "Futrelle, Mrs. J
acques Heath (Lily May Peel)" ...
##  $ 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" ...
```

Más adelante se tratará el tipo de los datos.

Se obtiene un resumen de cada una de las variables. En el que se especifican media, valor minimo, valor máximo, mediana, entre otros.

```
summary(datos)
```

```
##   PassengerId      Survived          Pclass          Name
##  Min.   :   1   Min.   :0.0000   Min.   :1.000   Length:1309
##  1st Qu.: 328   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median : 655   Median :0.0000   Median :3.000   Mode  :character
##  Mean   : 655   Mean   :0.3838   Mean   :2.295
##  3rd Qu.: 982   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :1309   Max.   :1.0000   Max.   :3.000
##                 NA's   :418
##      Sex                 Age            SibSp            Parch
##  Length:1309        Min.   : 0.17   Min.   :0.0000   Min.   :0.000
##  Class :character   1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.000
##  Mode  :character   Median :28.00   Median :0.0000   Median :0.000
##                     Mean   :29.88   Mean   :0.4989   Mean   :0.385
##                     3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.000
##                     Max.   :80.00   Max.   :8.0000   Max.   :9.000
```

```
##                     NA's   :263
##     Ticket             Fare            Cabin
##  Length:1309        Min.   :  0.000   Length:1309
##  Class :character   1st Qu.:  7.896   Class :character
##  Mode  :character   Median : 14.454   Mode  :character
##                     Mean   : 33.295
##                     3rd Qu.: 31.275
##                     Max.   :512.329
##                     NA's   :1
##    Embarked
##  Length:1309
##  Class :character
##  Mode  :character
##
##
##
##
```

Limpieza de los datos

En este apartado se va a proceder a limpiar los datos de manera que no contengan valores vacíos o valores extremos.

Ceros y elementos vacíos

En primer lugar, se procede a comprobar si existen valores nulos en el conjunto:

```
colSums(is.na(datos))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Ag
e
##           0         418           0           0           0         26
3
##       SibSp       Parch      Ticket        Fare       Cabin    Embarke
d
##           0           0           0           1           0
0
```

Existen valores nulos en las variables *Survived*, *Age* y *Fare*.

Se buscan valores vacíos:

```
colSums(datos=="")
```

```
## PassengerId    Survived      Pclass        Name         Sex         Ag
e
##           0          NA           0           0           0          N
A
##       SibSp       Parch      Ticket        Fare       Cabin    Embarke
d
```

```
##           0           0           0          NA        1014
2
```

Las siguientes columnas presentan valores vacíos: *Cabin* y *Embarked*.

Los valores de la variable *Survived* nulos corresponden a los registros procedentes del archivo de datos test.

Los valores vacíos de *Cabin* se van a dejar sin ningún valor ya que no se puede realizar una imputación porque faltan la mayoría de valores de esa variable. Y esta variable no se va a utilizar en el análisis.

Los valores de la variable *Embarked* vacíos se van a sustituir por "S" que representa el puerto de Southampton que fue del que partió el Titanic al comienzo del viaje. Por ello. se supone que la mayoría de viajeros embarcaron en dicho puerto.

```
datos$Embarked[datos$Embarked==""]="S"
```

Algunos de los tipos que R ha asignado a los datos se deben cambiar de tipo *int* a tipo *factor* ya que son variables que clasifican a los datos.

```
datos$Survived<-as.factor(datos$Survived)
datos$Pclass<-as.factor(datos$Pclass)
datos$Sex<-as.factor(datos$Sex)
datos$Embarked<-as.factor(datos$Embarked)
```

Se comprueba que la conversión se ha realizado correctamente:

```
str(datos)
```

```
## 'data.frame':    1309 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 1 2 2 ...
##  $ Pclass     : Factor w/ 3 levels "1","2","3": 3 1 3 1 3 3 1 3 3 2 ..
.
##  $ Name       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bra
dley (Florence Briggs Thayer)" "Heikkinen, Miss. Laina" "Futrelle, Mrs. J
acques Heath (Lily May Peel)" ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1
1 ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : 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   : Factor w/ 3 levels "C","Q","S": 3 1 3 3 3 2 3 3 3 1 ..
.
```

Las variables *Ticket, Name* y *Cabin* se van a eliminar del conjunto de datos porque no se van a utilizar en el análisis.

```
datos<-select(datos,PassengerId,Survived,Pclass,Sex,Age,SibSp,Parch,Fare,
Embarked)
```

Los valores nulos de la variable *Edad* y *Fare* se van a rellenar utilizando el método probabilistico de *missforest*.

```
library(randomForest)

## Warning: package 'randomForest' was built under R version 3.6.2

datos<-missForest::missForest(datos)$ximp

##   missForest iteration 1 in progress...done!
##   missForest iteration 2 in progress...done!
##   missForest iteration 3 in progress...done!
##   missForest iteration 4 in progress...done!
##   missForest iteration 5 in progress...done!
```

Se comprueba que los valores nulos han desaparecido.

```
colSums(is.na(datos))

## PassengerId    Survived      Pclass         Sex         Age       SibS
p
##           0           0           0           0           0
0
##       Parch        Fare    Embarked
##           0           0           0
```

Se muestran los estadisticos más importantes para comprobar que no han variado demasiado tras sustituir los valores nulos.

```
summary(datos)

##   PassengerId   Survived Pclass      Sex           Age
##  Min.   :   1   0:829    1:323   female:466   Min.   : 0.17
##  1st Qu.: 328   1:480    2:277   male  :843   1st Qu.:22.00
##  Median : 655            3:709                Median :28.00
##  Mean   : 655                                 Mean   :29.77
##  3rd Qu.: 982                                 3rd Qu.:37.00
##  Max.   :1309                                 Max.   :80.00
##      SibSp            Parch            Fare         Embarked
##  Min.   :0.0000   Min.   :0.000   Min.   :  0.000   C:270
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:  7.896   Q:123
##  Median :0.0000   Median :0.000   Median : 14.454   S:916
##  Mean   :0.4989   Mean   :0.385   Mean   : 33.277
##  3rd Qu.:1.0000   3rd Qu.:0.000   3rd Qu.: 31.275
##  Max.   :8.0000   Max.   :9.000   Max.   :512.329
```

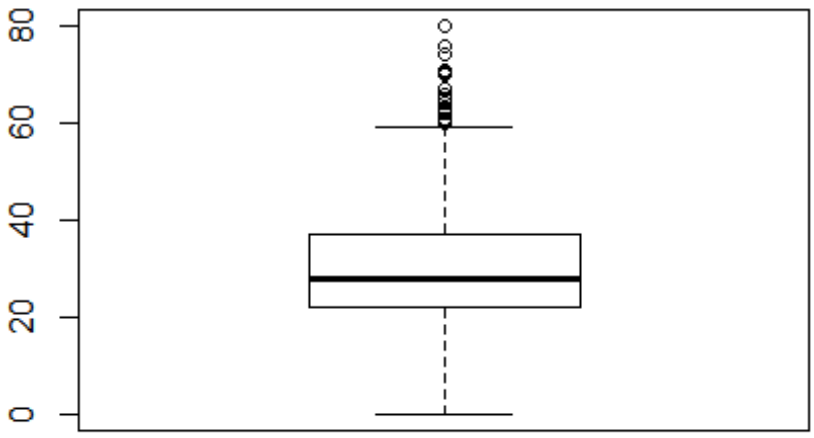
Valores extremos

Los valores extremos son aquellos que distan mucho del resto de valores del conjunto de datos. Se va a utilizar la función *boxplots.stats()* de R para determinar qué valores extremos contiene cada variable. Los valores extremos se van a buscar en las variables numéricas. Además, se van a obtener diagramas de caja de cada una de las variables para comprobar cuánto distan los valores extremos del resto de valores.

```
boxplot.stats(datos$Age)$out
```

```
##  [1] 66.0 65.0 71.0 70.5 61.0 62.0 63.0 65.0 61.0 60.0 64.0 65.0 63.0
71.0
## [15] 64.0 62.0 62.0 60.0 61.0 80.0 70.0 60.0 60.0 70.0 62.0 74.0 62.0
63.0
## [29] 60.0 60.0 67.0 76.0 63.0 61.0 60.5 64.0 61.0 60.0 64.0 64.0
```

```
boxplot(datos$Age)
```

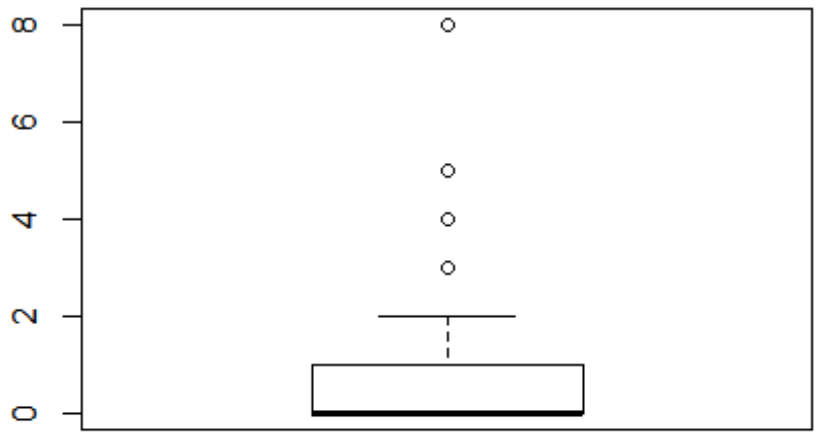


Los valores extremos que aparecen en la variable *Age* se van a dejar como están ya que aparecen debido a que las edades de los pasajeros son muy variadas dado la existencia de bebés y personas mayores a bordo del barco.

```
boxplot.stats(datos$SibSp)$out
```

```
##  [1] 3 4 3 3 4 5 3 4 5 3 3 4 8 4 4 3 8 4 8 3 4 4 4 4 8 3 3 5 3 5 3 4 4
3 3
## [36] 5 4 3 4 8 4 3 4 8 4 8 3 4 5 3 4 8 4 8 4 3 3
```

```
boxplot(datos$SibSp)
```

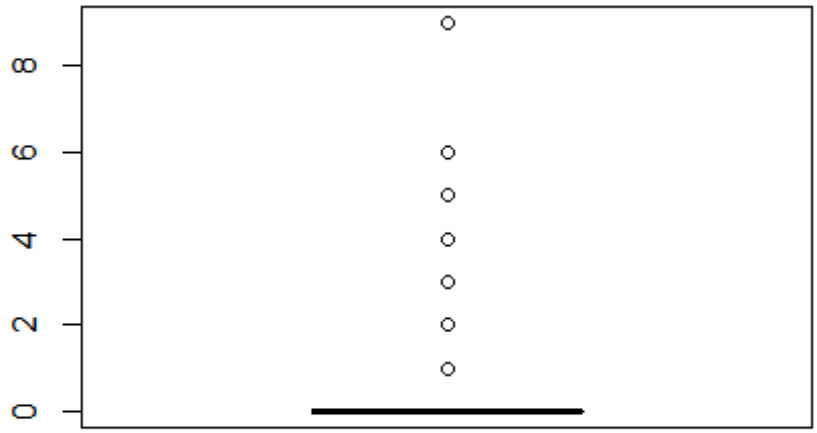


Los valores extremos de *SibSp* tampoco se van a tratar ya que es perfectamente válido que haya personas con muchos familiares a bordo y otras, que no cuenten con familiares en el barco.

```
boxplot.stats(datos$Parch)$out
```

```
##   [1] 1 2 1 5 1 1 5 2 2 1 1 2 2 2 1 2 2 2 3 2 2 1 1 1 1 2 1 1 2 2 1 2
2 2 1
##  [36] 2 1 1 2 1 4 1 1 1 1 2 2 1 2 1 1 1 2 1 1 2 2 2 1 1 2 2 1 2 1 1 1
1 1 1
##  [71] 1 2 1 2 2 1 1 2 1 1 2 1 1 1 1 2 1 1 1 4 1 1 2 2 2 2 2 1 1 1 2 2
1 1 2
## [106] 2 3 4 1 2 1 1 2 1 2 1 2 1 1 2 2 1 1 1 1 2 2 2 2 2 2 1 1 2 1 4 1
1 2 1
## [141] 2 1 1 2 5 2 1 1 1 2 1 5 2 1 1 1 2 1 6 1 2 1 2 1 1 1 1 1 1 1 3 2
1 1 1
## [176] 1 2 1 2 3 1 2 1 2 2 1 1 2 1 2 1 2 1 1 1 2 1 1 2 1 2 1 1 1 1 3 2
1 1 1
## [211] 1 5 2 1 1 1 1 3 1 2 2 1 2 1 2 1 2 4 1 1 2 1 1 1 4 6 2 3 1 1 2 2
2 1 1
## [246] 2 5 2 3 2 1 1 1 2 1 2 2 2 1 2 1 1 2 1 2 1 2 1 2 2 1 1 1 1 1 2 1
1 2 1
## [281] 1 1 2 1 2 9 1 1 1 2 2 2 1 9 1 1 2 2 1 1 2 1 1 1 1 1 1
```

```
boxplot(datos$Parch)
```



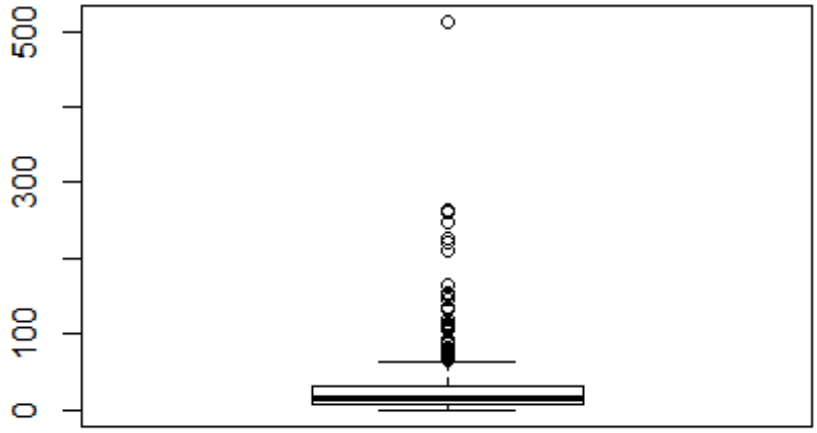
Con la variable *Parch* ocurre lo mismo que con la anterior.

```
boxplot.stats(datos$Fare)$out
```

```
##   [1]  71.2833 263.0000 146.5208  82.1708  76.7292  80.0000  83.4750
##   [8]  73.5000 263.0000  77.2875 247.5208  73.5000  77.2875  79.2000
##  [15]  66.6000  69.5500  69.5500 146.5208  69.5500 113.2750  76.2917
##  [22]  90.0000  83.4750  90.0000  79.2000  86.5000 512.3292  79.6500
##  [29] 153.4625 135.6333  77.9583  78.8500  91.0792 151.5500 247.5208
##  [36] 151.5500 110.8833 108.9000  83.1583 262.3750 164.8667 134.5000
##  [43]  69.5500 135.6333 153.4625 133.6500  66.6000 134.5000 263.0000
##  [50]  75.2500  69.3000 135.6333  82.1708 211.5000 227.5250  73.5000
##  [57] 120.0000 113.2750  90.0000 120.0000 263.0000  81.8583  89.1042
##  [64]  91.0792  90.0000  78.2667 151.5500  86.5000 108.9000  93.5000
##  [71] 221.7792 106.4250  71.0000 106.4250 110.8833 227.5250  79.6500
##  [78] 110.8833  79.6500  79.2000  78.2667 153.4625  77.9583  69.3000
##  [85]  76.7292  73.5000 113.2750 133.6500  73.5000 512.3292  76.7292
##  [92] 211.3375 110.8833 227.5250 151.5500 227.5250 211.3375 512.3292
##  [99]  78.8500 262.3750  71.0000  86.5000 120.0000  77.9583 211.3375
## [106]  79.2000  69.5500 120.0000  93.5000  80.0000  83.1583  69.5500
## [113]  89.1042 164.8667  69.5500  83.1583  82.2667 262.3750  76.2917
## [120] 263.0000 262.3750 262.3750 263.0000 211.5000 211.5000 221.7792
## [127]  78.8500 221.7792  75.2417 151.5500 262.3750  83.1583 221.7792
## [134]  83.1583  83.1583 247.5208  69.5500 134.5000 227.5250  73.5000
## [141] 164.8667 211.5000  71.2833  75.2500 106.4250 134.5000 136.7792
## [148]  75.2417 136.7792  82.2667  81.8583 151.5500  93.5000 135.6333
## [155] 146.5208 211.3375  79.2000  69.5500 512.3292  73.5000  69.5500
```

```
## [162]  69.5500 134.5000  81.8583 262.3750  93.5000  79.2000 164.8667
## [169] 211.5000  90.0000 108.9000
```

```
boxplot(datos$Fare)
```



Por último, los valores extremos de la variable *Fare* también se van a dejar como están porque el precio entre los billetes de primera y tercera clase varían mucho.

Se exporta el conjunto de datos preprocesado a un nuevo fichero: Se exporta el conjunto de datos a un nuevo archivo.

```
write.csv(datos,"datos_clean.csv")
```

Análisis de los datos

Selección de los grupos de datos a analizar

En primer lugar, se tienen los grupos de datos divididos en datos de entrenamiento y datos test tal cual se han descargado de la página web.

Para obtener más agrupaciones de los datos, se van a dividir según el valor de distintas variables.

```
# Agrupación por supervivientes
datos.survived<-datos[datos$Survived=="1",]
datos.nosurvived<-datos[datos$Survived=="1",]
```

```r
# Agrupación por clase
datos.primera<-datos[datos$Pclass=="1",]
datos.segunda<-datos[datos$Pclass=="2",]
datos.tercera<-datos[datos$Pclass=="3",]

# Agrupación por sexo
datos.mujer<-datos[datos$Sex=="female",]
datos.hombre<-datos[datos$Sex=="male",]
```

Comprobación de la normalidad y homogeneidad de la varianza

La comprobación de la normalidad de las variables numéricas se llevará a cabo mediante la prueba de *Pearson-Wilk*. En este caso, las variables que se estudiarán son *Age*, *SibSp*, *Parch* y *Fare*.

Se parte de la hipótesis nula de que las muestras provienen de una distribución normal, y de la hipótesis alternativa de que las muestras no provienen de distribuciones normales.

El nivel de significancia que se utilizará es 0,05. Si el p-valor obtenido en la prueba es menor que 0,05 entonces se rechara la hipótesis nula.

```r
shapiro.test(datos[,5])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos[, 5]
## W = 0.97758, p-value = 2.164e-13
```

```r
shapiro.test(datos[,6])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos[, 6]
## W = 0.51108, p-value < 2.2e-16
```

```r
shapiro.test(datos[,7])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos[, 7]
## W = 0.49797, p-value < 2.2e-16
```

```r
shapiro.test(datos[,8])
```

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

```
## data:  datos[, 8]
## W = 0.52765, p-value < 2.2e-16
```

Además, se va a aplicar la prueba de *Pearson* para comprobar si se obtienen los mismos resultados.

```
library("nortest")
pearson.test(datos[,5])

##
##  Pearson chi-square normality test
##
## data:  datos[, 5]
## P = 277.17, p-value < 2.2e-16

pearson.test(datos[,6])

##
##  Pearson chi-square normality test
##
## data:  datos[, 6]
## P = 23461, p-value < 2.2e-16

pearson.test(datos[,7])

##
##  Pearson chi-square normality test
##
## data:  datos[, 7]
## P = 27466, p-value < 2.2e-16

pearson.test(datos[,8])

##
##  Pearson chi-square normality test
##
## data:  datos[, 8]
## P = 6974, p-value < 2.2e-16
```

Todas las pruebas realizadas han obtenido un p-valor menor al nivel de significancia prefijado, por lo que se rechaza la hipótesis nula en todos los casos. Ninguna de las variables estudiadas sigue una distribución normal.

Aún así, al contar con más de 30 registros, por el teorema central del límite, los datos se pueden normalizar.

Se va a estudiar la homogeneidad de varianzas en la variable *Age* en función de los grupos que se han creado anteriormente. Se parte de la hipótesis nula de que las varianzas de las muestras son iguales, y de la hipótesis alternativa de que no son iguales. El nivel de significancia utilizado será 0,05.

```r
# En función del sexo
fligner.test(datos$Age ~ datos$Sex, data = datos)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  datos$Age by datos$Sex
## Fligner-Killeen:med chi-squared = 3.115, df = 1, p-value = 0.07757
```

```r
# En función de si sobrevivió o no
fligner.test(datos$Age ~ datos$Survived, data = datos)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  datos$Age by datos$Survived
## Fligner-Killeen:med chi-squared = 29.152, df = 1, p-value =
## 6.693e-08
```

```r
# En función de la clase en la que viajaban
fligner.test(datos$Age ~ datos$Pclass, data = datos)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  datos$Age by datos$Pclass
## Fligner-Killeen:med chi-squared = 48.356, df = 2, p-value =
## 3.16e-11
```

Los p-valor obtenidos en los datos agrupados por la supervivencia y la clase son menores a 0,05 por lo que se rechaza la hipótesis de que las varianzas son homogéneas.

En cambio, el p-valor obtenido en el test realizado mediante la agrupación por sexo el p-valor obtenido es mayor al nivel de significancia por lo que se acepta la hipótesis nula. Se puede determinar que las varianzas de las muestras son iguales.

Como la variable *Fare* también es continua, se va a realizar el mismo estudio de homogeneidad de varianzas con cada uno de los grupos formados.

```r
# En función del sexo
fligner.test(datos$Fare ~ datos$Sex, data = datos)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  datos$Fare by datos$Sex
## Fligner-Killeen:med chi-squared = 83.935, df = 1, p-value <
## 2.2e-16
```

```r
# En función de si sobrevivió o no
fligner.test(datos$Fare ~ datos$Survived, data = datos)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  datos$Fare by datos$Survived
## Fligner-Killeen:med chi-squared = 164.39, df = 1, p-value <
## 2.2e-16
```

```
# En función de la clase en la que viajaban
fligner.test(datos$Fare ~ datos$Pclass, data = datos)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  datos$Fare by datos$Pclass
## Fligner-Killeen:med chi-squared = 552.16, df = 2, p-value <
## 2.2e-16
```

En el análisis de cada uno de los grupos se obtiene un p-valor menor al nivel de significancia de 0.05 por lo que se rechaza la hipótesis nula y se acepta la hipótesis de que las varianzas no son iguales.

Pruebas estadisticas

Se obtiene una matriz de porcentajes de frecuencias de la variable *Sex* en función de si el pasajero sobrevivió o no.

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

```
##
##                 0        1
##   female 28.75536 71.24464
##   male   82.44365 17.55635
```

En la tabla se puede ver que las mujeres tenían más probabilidad de sobrevivir que los hombres.

Como se dispone de un conjunto de datos de entrenamiento y otro de test, se va a crear un árbol de decisión para comprobar si se predicen los datos correctamente.

```
y <- datos[,2]
X <- select(datos,Pclass,Sex,Embarked)
trainX <- X[1:891,]
trainy <- y[1:891]
model <- C50::C5.0(trainX, trainy,rules=TRUE )
summary(model)
```

```
##
## Call:
## C5.0.default(x = trainX, y = trainy, rules = TRUE)
##
##
## C5.0 [Release 2.07 GPL Edition]      Tue Jan 07 20:41:53 2020
## -------------------------------
##
## Class specified by attribute `outcome'
##
## Read 891 cases (4 attributes) from undefined.data
##
## Rules:
##
## Rule 1: (577/109, lift 1.3)
##  Sex = male
##  ->  class 0  [0.810]
##
## Rule 2: (491/119, lift 1.2)
##  Pclass = 3
##  ->  class 0  [0.757]
##
## Rule 3: (170/9, lift 2.5)
##  Pclass in {1, 2}
##  Sex = female
##  ->  class 1  [0.942]
##
## Rule 4: (109/18, lift 2.2)
##  Sex = female
##  Embarked in {C, Q}
##  ->  class 1  [0.829]
##
## Default class: 0
##
##
## Evaluation on training data (891 cases):
##
##          Rules
##    ----------------
##      No      Errors
##
##       4  168(18.9%)   <<
##
##
##     (a)   (b)    <-classified as
##    ----  ----
##     523    26    (a): class 0
##     142   200    (b): class 1
##
##
```

```
##  Attribute usage:
##
##   90.12% Sex
##   74.19% Pclass
##   12.23% Embarked
##
##
## Time: 0.0 secs
```

Se utilizan los datos pertenecientes al conjunto test y se comparan con los que se encuentran en el archivo *gender_submission.csv*. La matriz de confusión servirá para comprobar si el modelo creado clasifica correctamente los datos.

```
testX <- X[892:1309,]
testy <- read.csv('gender_submission.csv',stringsAsFactors = FALSE)
testy<-testy$Survived
predicted_model <- predict( model, testX, type="class" )
mat_conf<-table(testy,Predicted=predicted_model)
mat_conf
```

```
##      Predicted
## testy   0   1
##     0 266   0
##     1  41 111
```

Los no supervivientes se predicen correctamente pero, hay fallos en los supervivientes.

Se va a realizar un modelo de regresión logística con las variables *Survived* y *Sex* para comprobar la relación que existe entre ellas.

```
modelo <- glm(datos$Survived ~ datos$Sex, data = datos, family = "binomia
l")
summary(modelo)
```

```
##
## Call:
## glm(formula = datos$Survived ~ datos$Sex, family = "binomial",
##     data = datos)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5788  -0.6214  -0.6214   0.8235   1.8653
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)     0.9073     0.1023   8.865   <2e-16 ***
## datos$Sexmale  -2.4540     0.1366 -17.960   <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: 1720.5  on 1308  degrees of freedom
## Residual deviance: 1342.5  on 1307  degrees of freedom
## AIC: 1346.5
##
## Number of Fisher Scoring iterations: 4
```

El valor obtenido de la estimación es -2.5826 lo que significa que las dos variables tienen una relación inversa. Es decir, si la persona ha sobrevivido está inversamente relacionado con que la persona sea hombre.

También se va a realizar un modelo de regresión logística entre las variables *Survived* y *Pclass*.

```
modelo <- glm(datos$Survived ~ datos$Pclass, data = datos, family = "bino
mial")
summary(modelo)
```

```
##
## Call:
## glm(formula = datos$Survived ~ datos$Pclass, family = "binomial",
##     data = datos)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4740  -0.6817  -0.6817   0.9074   1.7739
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)     0.6746     0.1177   5.733 9.87e-09 ***
## datos$Pclass2  -0.9581     0.1691  -5.667 1.45e-08 ***
## datos$Pclass3  -2.0157     0.1498 -13.459  < 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: 1720.5  on 1308  degrees of freedom
## Residual deviance: 1515.3  on 1306  degrees of freedom
## AIC: 1521.3
##
## Number of Fisher Scoring iterations: 4
```

Los estimadores obtenidos son -0.9581 y -1.9733. Eso implica que si un pasajero ha sobrevivido está inversamente relacionado con la pertenencia a la segunda o tercera clase.

A continuación, se va a realizar un test chi-cuadrado para comprobar si realmente existe esta relación entre las variables *Sex* y *Survived*. La hipótesis nula dice que no

existe asociación entre las dos varables, mientras que la hipótesis alternativa dice que sí existe asociación entre ellas.

```
chisq.test(datos$Survived, datos$Sex)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  datos$Survived and datos$Sex
## X-squared = 370.18, df = 1, p-value < 2.2e-16
```

Partiendo de una confianza, por ejemplo, del 95% se observa que el p-valor obtenido es mucho menor que 0.05 por lo que se puede decir que las dos variables están relacionadas. Este era el resultado esperado después del análisis realizado previamente.

Además, se va a comprobar si la variable *Pclass* y *Survived* se encuentran relacionadas mediante un test chi-cuadrado.

```
chisq.test(datos$Survived, datos$Pclass)
```

```
##
##  Pearson's Chi-squared test
##
## data:  datos$Survived and datos$Pclass
## X-squared = 203.99, df = 2, p-value < 2.2e-16
```

Se vuelve a tener un p-valor ínifimo por lo que las dos variables presentan una relación importante.

Se repite el estudio para comprobar si existe relación entre la variable *Survived* y *Embarked*

```
chisq.test(datos$Survived, datos$Embarked)
```

```
##
##  Pearson's Chi-squared test
##
## data:  datos$Survived and datos$Embarked
## X-squared = 54.353, df = 2, p-value = 1.576e-12
```

Se vuelve a obtener una relación directa entre las dos variables.

Se realiza un análisis entre el puerto de embarque y la clase.

```
chisq.test(datos$Embarked, datos$Pclass)
```

```
##
##  Pearson's Chi-squared test
##
## data:  datos$Embarked and datos$Pclass
## X-squared = 204.48, df = 4, p-value < 2.2e-16
```

Se realiza un test chi-cuadrado entre las variables *Sex* y *Embarked*.

```
chisq.test(datos$Sex, datos$Embarked)
```

```
##
##  Pearson's Chi-squared test
##
## data:  datos$Sex and datos$Embarked
## X-squared = 19.139, df = 2, p-value = 6.982e-05
```

En los dos casos se han obtenido p-valores menores a 0.05 por lo que las variables tienen relación directa.

Por último, se realiza un test Wilcoxon para comprobar la relación entre variables numéricas. Este test comprueba si la mediana de las diferencias de ambas variables es 0. La hipótesis nula determina que la mediana de las diferencias es 0. La hipótesis alternativa determina que la mediana de las diferencias es distinta de 0.

```
wilcox.test(datos$Fare,datos$Age)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  datos$Fare and datos$Age
## W = 580838, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Como el p-valor obtenido es menor al nivel de significancia determinado, se rechaza la hipótesis nula por lo que se acepta la hipótesis alternativa.

Representación de resultados

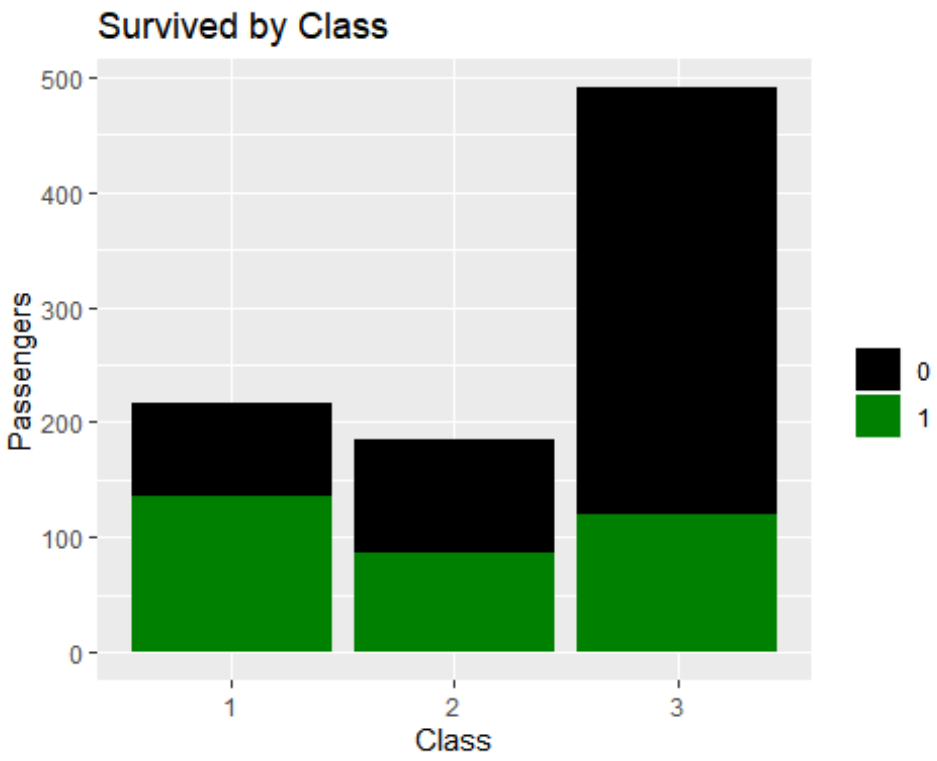
Se van a representar las variables de tipo *factor* en función de si el personaje sobrevivió o no, para ver si concuerda con los resultados obtenidos en el apartado anterior.

```
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'
```

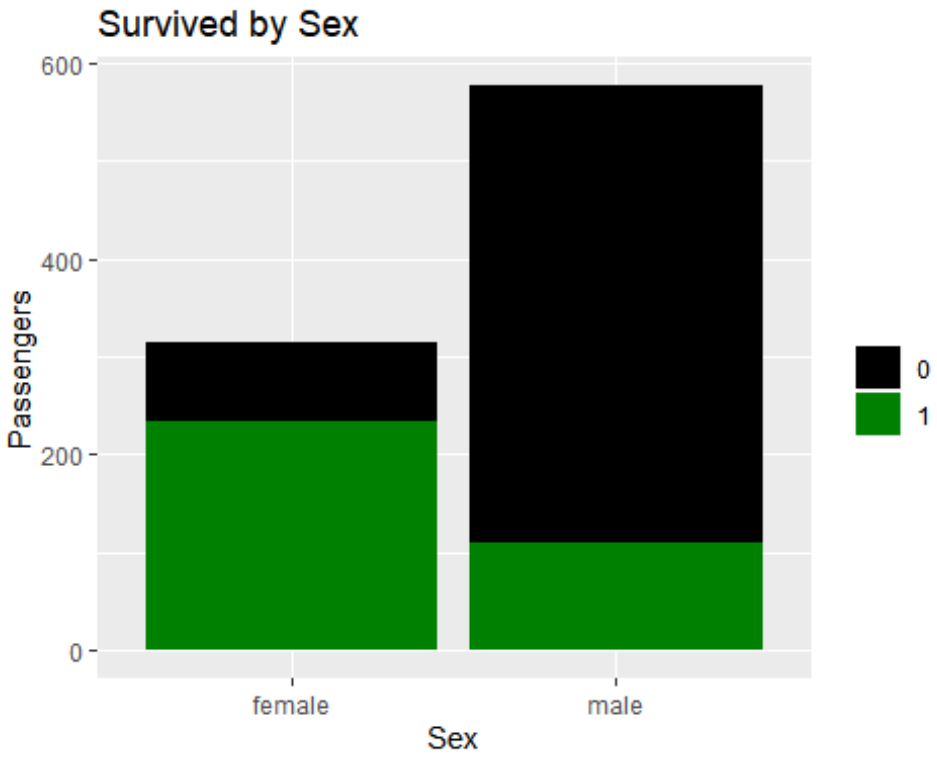
```
## The following object is masked from 'package:randomForest':
##
##     margin
```

```
train<-datos[1:891,]
ggplot(train,aes(Pclass,fill=Survived))+geom_bar() +labs(x="Class", y="Pa
ssengers")+ guides(fill=guide_legend(title=""))+ scale_fill_manual(values
=c("black","#008000"))+ggtitle("Survived by Class")
```



La mayor parte de los pasajeros pertenecientes a la tercera clase no sobrevivieron. Sin embargo en las otras dos clases existentes los pasajeros que sobrevivieron representan más o menos la mitad. Además, en la gráfica se puede ver que más de la mitad de los pasajeros del barco se alojaban en tercera clase. Estos resultados tienen sentido si se tiene en cuenta que las primeras familias en acceder a los botes salvavidas fueron familias de las clases con una categoría superior.

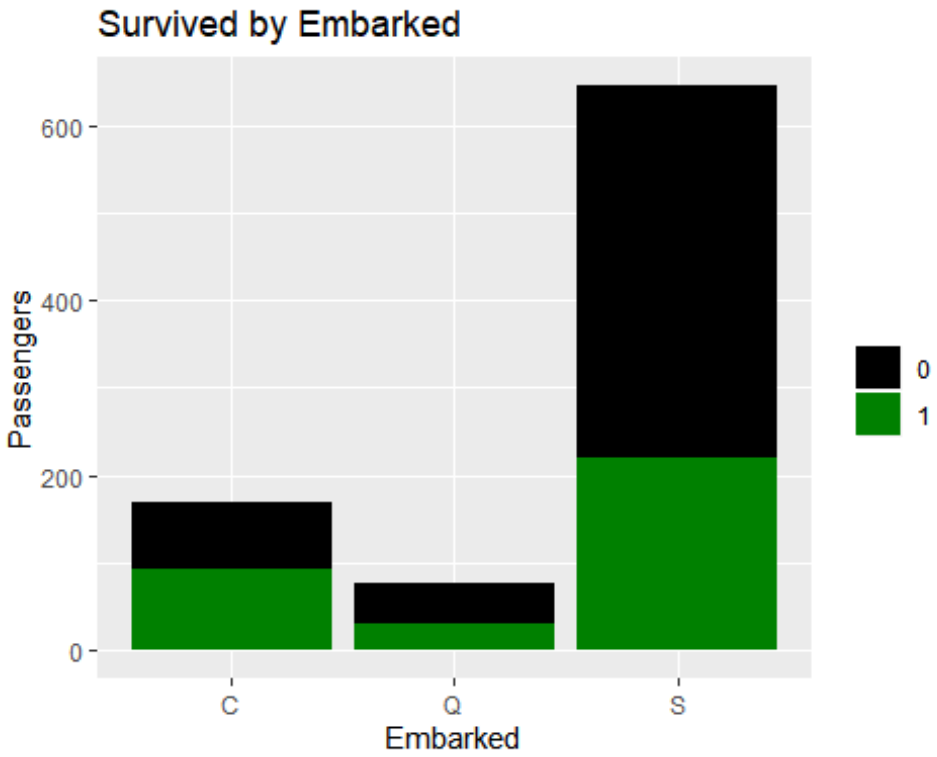
```
ggplot(train,aes(Sex,fill=Survived))+geom_bar() +labs(x="Sex", y="Passeng
ers")+ guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("b
lack","#008000"))+ggtitle("Survived by Sex")
```



El gráfico representa los pasajeros que sobrevivieron o no en función de su sexo. Casi todos los hombres que se encontraban en el barco murieron en comparación con las mujeres. De estas, casi la totalidad sobrevivió a la tragedia. Esta situación puede explicarse por el orden de la subida a los botes salvavidas, las mujeres subieron antes que los hombres.

Si se comparaba con el gráfico anterior, probablemente las mujeres que aparecen muertas correspondan a mujeres de tercera clase.

```
ggplot(train,aes(Embarked,fill=Survived))+geom_bar() +labs(x="Embarked",
y="Passengers")+ guides(fill=guide_legend(title=""))+ scale_fill_manual(v
alues=c("black","#008000"))+ggtitle("Survived by Embarked")
```



Por último, se ha representado la supervivencia de los pasajeros en función del puerto en el que embarcaron. La mayor parte de los pasajeros pertenecían al puerto de salida (Southampton) y la mayor parte de pasajeros que no sobrevivieron pertenecen a este puerto de embarque.

La única explicación posible es que los pasajeros que embarcaron en el resto de puertos perteneciesen a primera o segunda clase.

Conclusiones

La principal conclusión del análisis es que los pasajeros que sobrevivieron pertenecían a la primera o segunda clase y que eran mujeres. Son variables que han presentado una gran relación en el análisis. Esto tiene sentido ya que los primeros pasajeros en embarcar en los botes salvavidas fueron mujeres y niños de clases altas como ya se ha mencionado en apartados anteriores.

Contribuciones	Firma
Investigación previa	SGR
Redacción de las respuestas	SGR
Desarrollo código	SGR