Práctica 2: Limpieza y validación de los datos

Héctor Hernández Membiela

11/06/2019

Índice

1. Detalles de la actividad

1.1. Descripción

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.

1.2. 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.

1.3. 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.

2. Resolución

2.1. Descripción del dataset.

El conjunto de datos objeto de análisis es el dataset Titanic, el cual se ha obtenido a partir de este enlace en Kaggle (https://www.kaggle.com/c/titanic).

El hundimiento del RMS Titanic es, probablemente, el naufragio más famoso de la historia. El 15 de Abril de 1912, durante su viaje inaugural, el Titanic se hundió tras colisionar con un iceberg, muriendo 1502 personas de un total de 2224, contabilizando pasajeros y tripulación.

Una de las razones por las que el naufragio se cobró tantas vidas fue el no disponer de suficientes botes salvavidas para todos los pasajeros y la tripulación. Aunque la suerte también tuvo su influencia en sobrevivir a la catastrofe, dado que algunas personas tenían más probabilidades de sobrevivir que otras, como mujeres, niños o los pasajeros de primera categoría.

Así pues, la importancia de este conjunto de datos radica en la capacidad de analizar qué tipo de personas sobrevivieron. Aplicando técnicas de *machine learning* se busca predecir qué pasajeros sobrevivieron al naufragio.

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

Comenzaremos por la carga del conjunto de datos. Kaggle proporciona los datos divididos en un conjunto de **training** y otro de **test**. Uniremos ambos conjuntos para revisar los datos en su totalidad:

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

train <- read.csv( '../data/titanic-train.csv', stringsAsFactors = F )
test <- read.csv( '../data/titanic-test.csv', stringsAsFactors = F )

full <- bind_rows( train, test )

str( full )
```

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

Como puede verse en la salida del bloque anterior, el conjunto de datos está constituido por 12 variables (columnas) que presentan 1309 observaciones (filas o registros).

Entre los campos de este conjunto de datos, encontramos los siguientes:

Variable	Descripción
PassengerId	Identificador del pasajero

Variable	Descripción
Survived	Indica si el pasajero sobrevivió (1) o murió (0)
Pclass	Categoría en la que viajaba el pasajero
Name	Nombre del pasajero
Sex	Sexo del pasajero
Age	Edad del pasajero
SibSp	Número de hermanos o esposas que viajaban a bordo con el pasajero
Parch	Número de padres o hijos que viajaban a bordo con el pasajero
Ticket	Ticket de embarque
Fare	Tarifa
Cabin	Cabina asignada
Embarked	Puerto donde embarcó el pasajero

2.3. Limpieza de los datos

2.3.1. ¿Los datos contienen ceros o elementos vacíos?

Comúnmente, se utilizan los ceros como valor centinela para indicar la ausencia de ciertos valores. Sin embargo, no es el caso de este conjunto de datos puesto que se ha utilizado una combinación del caracter vacío y el valor especial 'NA' para denotar los valores desconocidos.

Procedemos a conocer a continuación qué campos contienen elementos vacíos. La función **describe** del paquete **Hmisc** nos indica para cada variable del conjunto de datos cuántos valores desconocidos tenemos (campo **missing**):

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

describe( full )
```

```
## full
##
##  12  Variables      1309  Observations
## ---------------------------------------------------------------------------
## PassengerId
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1309        0     1309        1      655    436.7     66.4    131.8
##      .25      .50      .75      .90      .95
##    328.0    655.0    982.0   1178.2   1243.6
##
## lowest :    1    2    3    4    5, highest: 1305 1306 1307 1308 1309
## ---------------------------------------------------------------------------
## Survived
##        n  missing distinct     Info      Sum     Mean      Gmd
##      891      418        2     0.71      342   0.3838   0.4735
##
## ---------------------------------------------------------------------------
## Pclass
##        n  missing distinct     Info     Mean      Gmd
##     1309        0        3    0.817    2.295   0.8689
##
## Value          1     2     3
## Frequency    323   277   709
```

```
## Proportion 0.247 0.212 0.542
## ------------------------------------------------------------------------
## Name
##        n  missing distinct
##     1309        0     1307
##
## lowest : Abbing, Mr. Anthony              Abbott, Master. Eugene Joseph    Abbott, Mr. Rossmore Edwa
## highest: Zabour, Miss. Hileni             Zabour, Miss. Thamine            Zakarian, Mr. Mapriededer
## ------------------------------------------------------------------------
## Sex
##        n  missing distinct
##     1309        0        2
##
## Value      female   male
## Frequency     466    843
## Proportion  0.356  0.644
## ------------------------------------------------------------------------
## Age
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1046      263       98    0.999    29.88    16.06        5       14
##      .25      .50      .75      .90      .95
##       21       28       39       50       57
##
## lowest :  0.17  0.33  0.42  0.67  0.75, highest: 70.50 71.00 74.00 76.00 80.00
## ------------------------------------------------------------------------
## SibSp
##        n  missing distinct     Info     Mean      Gmd
##     1309        0        7     0.67   0.4989    0.777
##
## Value          0     1     2     3     4     5     8
## Frequency    891   319    42    20    22     6     9
## Proportion 0.681 0.244 0.032 0.015 0.017 0.005 0.007
## ------------------------------------------------------------------------
## Parch
##        n  missing distinct     Info     Mean      Gmd
##     1309        0        8    0.549    0.385   0.6375
##
## Value          0     1     2     3     4     5     6     9
## Frequency   1002   170   113     8     6     6     2     2
## Proportion 0.765 0.130 0.086 0.006 0.005 0.005 0.002 0.002
## ------------------------------------------------------------------------
## Ticket
##        n  missing distinct
##     1309        0      929
##
## lowest : 110152      110413      110465      110469      110489
## highest: W./C. 6608  W./C. 6609  W.E.P. 5734 W/C 14208   WE/P 5735
## ------------------------------------------------------------------------
## Fare
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1308        1      281        1     33.3    38.61    7.225    7.568
##      .25      .50      .75      .90      .95
##    7.896   14.454   31.275   78.051  133.650
##
```

```
## lowest :   0.0000   3.1708   4.0125   5.0000   6.2375
## highest: 227.5250 247.5208 262.3750 263.0000 512.3292
## ---------------------------------------------------------------------------
## Cabin
##        n  missing distinct
##      295     1014      186
##
## lowest : A10 A11 A14 A16 A18, highest: F33 F38 F4  G6  T
## ---------------------------------------------------------------------------
## Embarked
##        n  missing distinct
##     1307        2        3
##
## Value          C     Q     S
## Frequency    270   123   914
## Proportion 0.207 0.094 0.699
## ---------------------------------------------------------------------------
```

Llegados a este punto debemos decidir cómo manejar estos registros que contienen valores desconocidos para algún campo. Una opción podría ser eliminar los registros que incluyen este tipo de valores, pero ello supondría desaprovechar información. Podemos reemplazar los elementos vacíos con valores extraídos a partir de la distribución de los datos (por ejemplo, la media o la mediana) o podemos utilizar métodos predictivos. Usaremos ambos, apoyándonos en gráficos para decidir qué valor final usar.

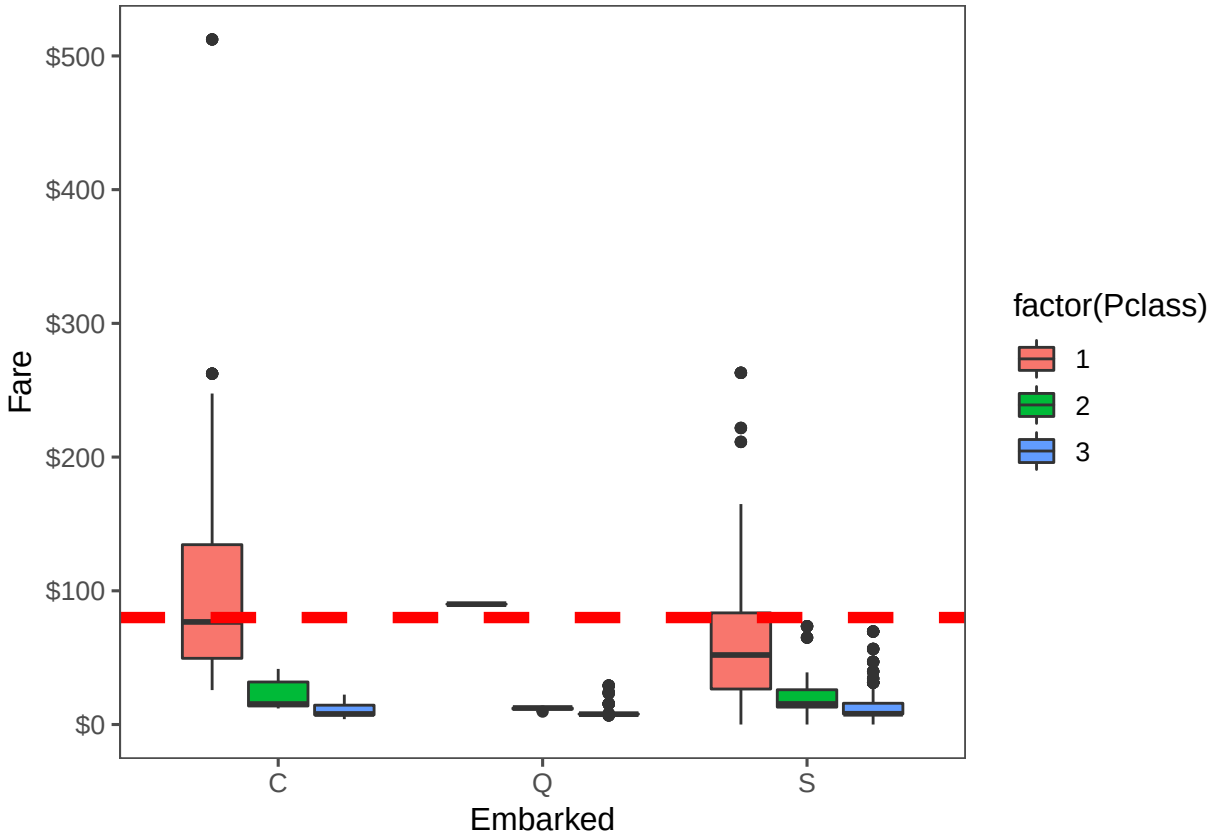
Comenzaremos con los valores perdidos en la variable **Embarked**. Vemos que dichos valores pertenecen a los pasajeros 62 y 830 y que el precio de su pasaje fue de 80$. Estudiaremos las variables **Pclass** y **Fare** para intentar averiguar dónde embarcaron estos pasajeros:

```
if( !require( ggplot2 ) ) {
    install.packages( 'ggplot2', repos = 'http://cran.us.r-project.org' )
    library( ggplot2 )
}

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

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

full %>%
    filter( Embarked == "" ) %>%
    select( PassengerId, Fare )
```

PassengerId	Fare
62	80
830	80

```
embark_fare <- full %>%
  filter( PassengerId != 62 & PassengerId != 830 )
```

```
ggplot( embark_fare,
        aes( x = Embarked, y = Fare, fill = factor( Pclass ) ) ) +
  geom_boxplot() +
  geom_hline( aes( yintercept = 80 ), colour = 'red', linetype = 'dashed', lwd = 2 ) +
  scale_y_continuous( labels = dollar_format() ) +
  theme_few()
```



Se puede apreciar en el gráfico anterior como los pasajeros de primera que embarcaron en Charbourg ('C') pagaron de media 80$, por tanto, podemos afirmar con bastante seguridad que el valor perdido que buscamos es 'C'.

```
full$Embarked[c( 62, 830 )] <- 'C'
```

La información sobre las cabinas no es importante para el estudio que vamos a realizar, por lo que ignoraremos los valores perdidos para este atributo.

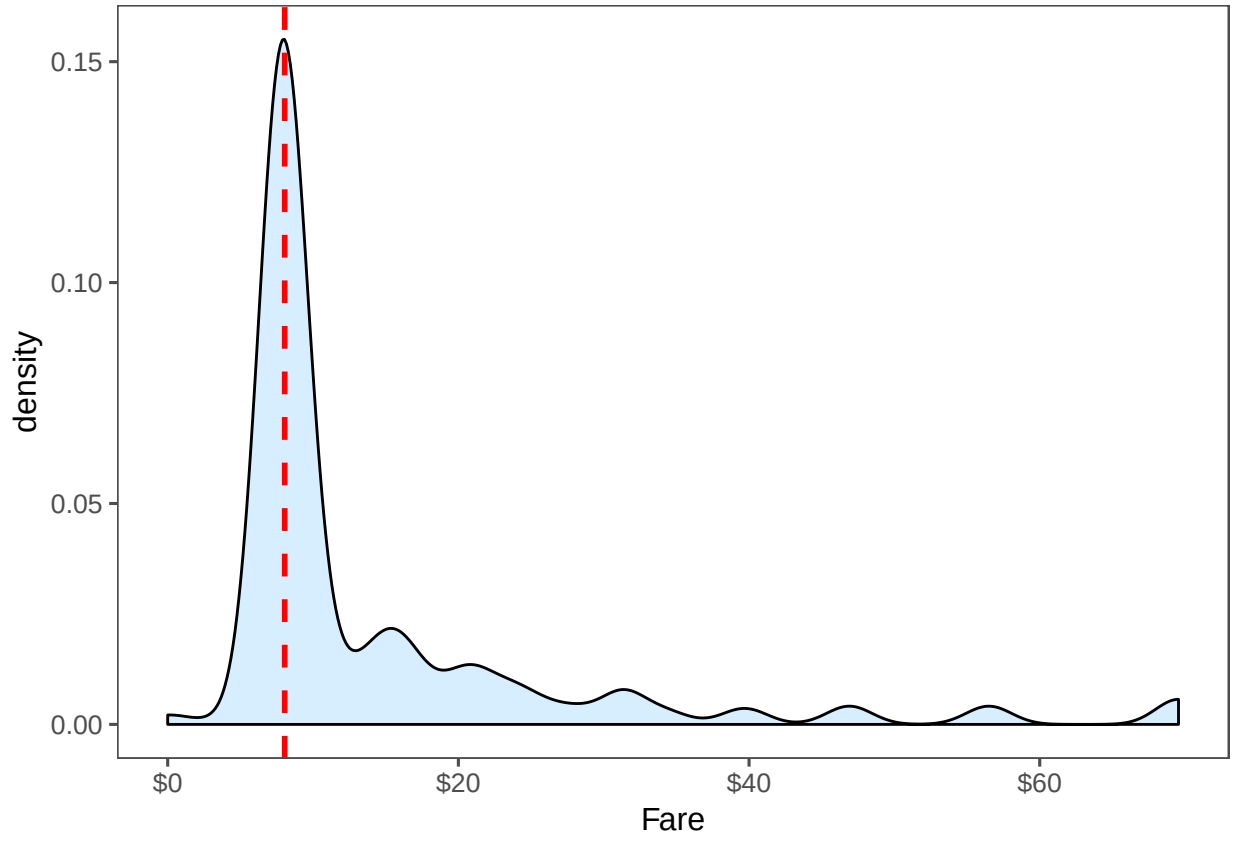
Pasemos ahora a los valores 'NA'. Comenzaremos por el único pasajero sin tarifa asociada:

```
full %>%
    filter( is.na( Fare ) )
```

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

Este pasajero de tercera clase, partió de Southampton ('S'). Vamos a graficar las tarifas de todos aquellos pasajeros que viajaban en la misma clase y partieron del mismo puerto:

```
ggplot( full[full$Pclass == '3' & full$Embarked == 'S', ],
        aes( x = Fare ) ) +
    geom_density( fill = '#99d6ff', alpha = 0.4 ) +
    geom_vline( aes( xintercept = median( Fare, na.rm = T ) ),
                colour = 'red', linetype = 'dashed', lwd = 1 ) +
    scale_x_continuous( labels = dollar_format() ) +
    theme_few()
```



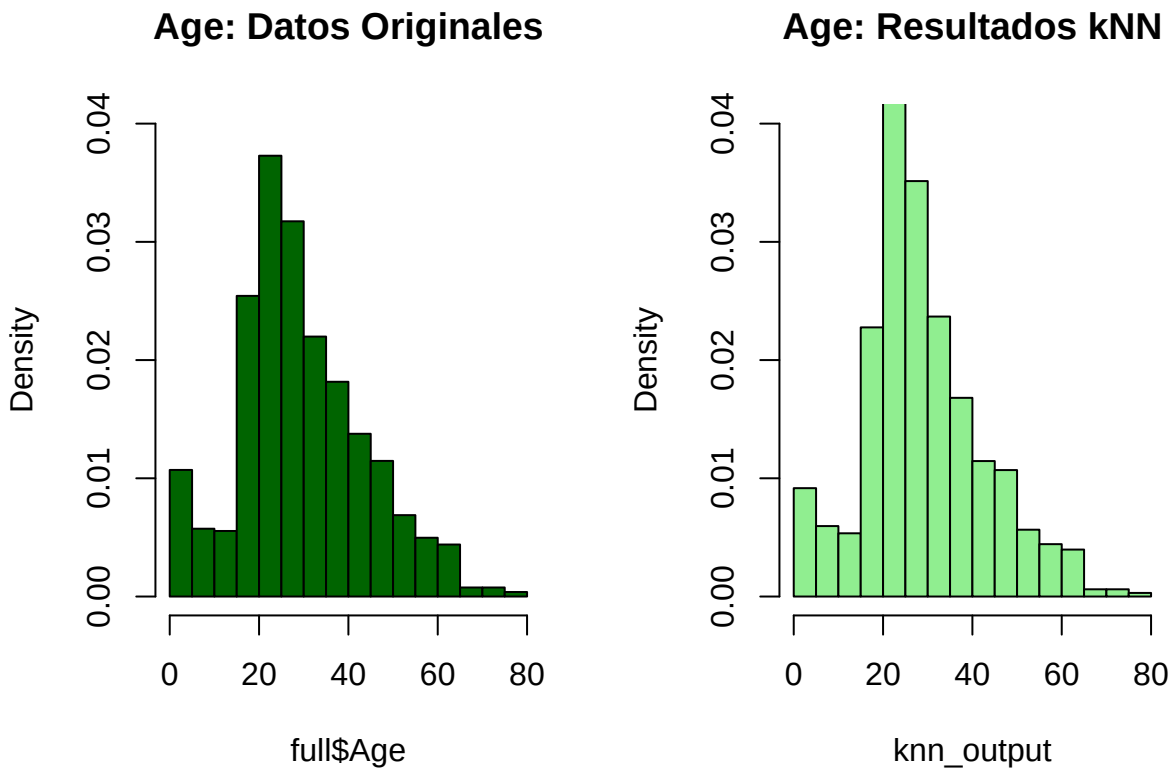
Por lo visto en el gráfico, podemos reemplazar el valor perdido por la media de su clase y puerto de embarque.

```
full$Fare[1044] <- median( full[full$Pclass == '3' & full$Embarked == 'S', ]$Fare,
                          na.rm = TRUE )
```

Finalizaremos el tratamiento de los valores perdidos con el campo **Age**. Emplearemos un método de imputación de valores basado en la similitud o diferencia entre los registros: la imputación basada en k vecinos más próximos (en inglés, *kNN-imputation*). La elección de esta alternativa se realiza bajo la hipótesis de que nuestros registros guardan cierta relación.

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

knn_output <- kNN(full)$Age

par( mfrow = c( 1, 2 ) )
hist( full$Age, freq = F, main = 'Age: Datos Originales', col = 'darkgreen',
```

```
      ylim = c( 0,0.04 ) )
hist( knn_output, freq = F, main = 'Age: Resultados kNN', col = 'lightgreen',
      ylim = c( 0, 0.04 ) )
```



Comparando los histogramas de los datos originales y del resultado del algoritmo *kNN*, vemos que se puede considerar una aproximación bastante fiable, por lo que reemplazamos los valores originales:

```
full$Age <- knn_output
```

Nota: los 418 valores 'NA' presentes en la variable **Survived**, se deben a que provienen del conjunto original de test y, por tanto, dicha variable no se proporcionaba. Al utilizar la función **bind_rows**, *R* ha completado los valores perdidos con valores 'NA'. Dado que esta es la variable a clasificar, no corregiremos los valores 'NA'.

2.3.2. Identificación y tratamiento de valores extremos

Los valores extremos u *outliers* son aquellos que parecen no ser congruentes si los comparamos con el resto de los datos. Para identificarlos, podemos hacer uso de dos vías:

1. Representar un diagrama de caja por cada variable y ver qué valores distan mucho del rango intercuartílico (la caja)
2. Utilizar la función boxplots.stats() de R.

Optaremos por el segundo método, así, se mostrarán sólo los valores atípicos para aquellas variables que los contienen:

```
lapply( full, function( x ) { if( is.numeric( x ) ) boxplot.stats( x )$out } )
```

```
## $PassengerId
```

```
## integer(0)
##
## $Survived
## integer(0)
##
## $Pclass
## integer(0)
##
## $Name
## NULL
##
## $Sex
## NULL
##
## $Age
##  [1] 66.0 65.0 59.0 71.0 70.5 61.0 61.0 59.0 62.0 63.0 65.0 61.0 60.0 64.0
## [15] 65.0 63.0 71.0 64.0 62.0 62.0 62.0 60.0 61.0 61.0 80.0 60.0 70.0 60.0
## [29] 60.0 60.0 70.0 62.0 74.0 62.0 63.0 60.0 60.0 67.0 76.0 63.0 61.0 60.5
## [43] 64.0 61.0 60.0 64.0 64.0 59.0
##
## $SibSp
##  [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
##
## $Parch
##   [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
##
## $Ticket
## NULL
##
## $Fare
##   [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
##
## $Cabin
## NULL
##
## $Embarked
## NULL
```

Los valores extremos obtenidos para la variable **Age** son perfectamente plausibles para personas de tercera edad. El precio (variable **Fare**) es una entidad variable en función de la demanda y oferta y, por lo que respecta a las variables **SibSp** y **Parch**, si bien algunos valores pueden parecer excesivos, siempre se han dado casos de familias muy numerosas. Es por ello que el manejo de estos valores extremos consistirá en simplemente dejarlos como actualmente están recogidos.

2.4. Análisis de los datos

2.4.1. Selección de los grupos de datos que se quieren analizar/comparar

A continuación, seleccionamos los grupos dentro de nuestro conjunto de datos que pueden resultar interesantes para analizar y/o comparar. Optaremos por aquellas variables de tipo númerico más la variable **Sex**, ya que intuimos que puede ser importante de cara al resultado final.

```
train <- full[1:891,]
test <- full[892:1309,]

str( 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 21 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" ...
```

```
train.numeric <- dplyr::select_if( train, is.numeric )
train.numeric$Sex <- as.integer( as.factor( train$Sex ) )

test.numeric <- dplyr::select_if( test, is.numeric )
test.numeric$Sex <- as.integer( as.factor( test$Sex ) )
```

```r
str( train.numeric )
```

```
## 'data.frame':    891 obs. of  8 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 ...
##  $ Age        : num  22 38 26 35 35 21 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 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Sex        : int  2 1 1 1 2 2 2 2 1 1 ...
```

2.4.2. Comprobación de la normalidad y homogeneidad de la varianza

Para comprobar que los valores que toman nuestras variables cuantitativas provienen de una población distribuida normalmente, utilizaremos la prueba de normalidad de **Shapiro-Wilk**. Asumiendo como hipótesis nula que la población está distribuida normalmente, si el p-valor es menor al nivel de significancia, generalmente $\alpha = 0.05$, entonces la hipótesis nula es rechazada y se concluye que los datos no cuentan con una distribución normal. Si, por el contrario, el p-valor es mayor a α, se concluye que no se puede rechazar dicha hipótesis y se asume que los datos siguen una distribución normal.

```r
lapply( train.numeric, function( x ) { shapiro.test( x ) } )
```

```
## $PassengerId
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.9548, p-value = 6.308e-16
##
##
## $Survived
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.61666, p-value < 2.2e-16
##
##
## $Pclass
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.71833, p-value < 2.2e-16
##
##
## $Age
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.97771, p-value = 1.977e-10
##
##
```

```
## $SibSp
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.51297, p-value < 2.2e-16
##
##
## $Parch
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.53281, p-value < 2.2e-16
##
##
## $Fare
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.52189, p-value < 2.2e-16
##
##
## $Sex
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.6041, p-value < 2.2e-16
```

Dados los resultados del bloque anterior, comprobamos que nuestras variables no siguen una distribución normal.

Seguidamente, pasamos a estudiar la homogeneidad de varianzas mediante la aplicación de un test de **Fligner-Killeen**. Esta prueba es utilizada cuando los datos no cumplen con la condición de normalidad, extremo este comprobado en el punto anterior. La hipótesis nula asume igualdad de varianzas en los diferentes grupos de datos, por lo que p-valores inferiores al nivel de significancia indicarán heterocedasticidad.

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

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

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

Dado que la prueba resulta en un p-valor inferior al nivel de significancia (< 0.05), se rechaza la hipótesis nula de homocedasticidad y se concluye que la variable **Survived** presenta varianzas estadísticamente diferentes para los dos sexos. En otras palabras, hemos comprobado mediante pruebas estadísticas nuestra intuición sobre la influencia de la variable **Sex** en la probabilidad de sobrevivir al hundimiento.

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

2.4.3.1. ¿Qué variables cuantitativas influyen más en la supervivencia?

Siguiendo el hilo del apartado anterior, procedemos a realizar un análisis de correlación entre las distintas variables para determinar cuáles de ellas ejercen una mayor influencia en la supervivencia al naugragio. Para ello, se utilizará el coeficiente de correlación de **Spearman**, puesto que hemos visto que tenemos datos que no siguen una distribución normal.

```
( correlation.matrix <- cor( train.numeric, method = "spearman" ) )
```

```
##              PassengerId     Survived      Pclass         Age       SibSp
## PassengerId  1.000000000 -0.005006661 -0.03409135  0.04771502 -0.06116077
## Survived    -0.005006661  1.000000000 -0.33966794 -0.04179613  0.08887948
## Pclass      -0.034091350 -0.339667937  1.00000000 -0.40054066 -0.04301877
## Age          0.047715020 -0.041796135 -0.40054066  1.00000000 -0.20813984
## SibSp       -0.061160766  0.088879485 -0.04301877 -0.20813984  1.00000000
## Parch        0.001235178  0.138265633 -0.02280134 -0.26610131  0.45001397
## Fare        -0.013975134  0.323736139 -0.68803167  0.12605278  0.44711299
## Sex          0.042938880 -0.543351381  0.13577453  0.09885762 -0.19520430
##                    Parch        Fare         Sex
## PassengerId  0.001235178 -0.01397513  0.04293888
## Survived     0.138265633  0.32373614 -0.54335138
## Pclass      -0.022801342 -0.68803167  0.13577453
## Age         -0.266101310  0.12605278  0.09885762
## SibSp        0.450013971  0.44711299 -0.19520430
## Parch        1.000000000  0.41007381 -0.25451198
## Fare         0.410073808  1.00000000 -0.25959350
## Sex         -0.254511982 -0.25959350  1.00000000
```

Así, identificamos cuáles son las variables más correlacionadas con la probabilidad de sobrevivir en función de su proximidad con los valores -1 y +1. Teniendo esto en cuenta, queda patente cómo la variable más relevante es **Sex**, seguida de **Pclass** y **Fare**.

2.4.3.2. ¿Hay diferencias en la probabilidad de sobrevivir entre hombres y mujeres?

La segunda prueba estadística que se aplicará consistirá en un contraste de hipótesis sobre dos muestras para determinar si la probabilidad de sobrevivir es superior dependiendo del sexo del pasajero. Para ello, tendremos dos muestras: la primera de ellas se corresponderá con los valores para hombres y, la segunda, con aquellos de las mujeres.

Se debe destacar que un test paramétrico como el que a continuación se utiliza necesita que los datos sean normales, si la muestra es de tamaño inferior a 30. Como en nuestro caso, n $>$ 30, el contraste de hipótesis siguiente es válido.

```
men.survived <- train.numeric[train.numeric$Sex == 2,]$Survived
women.survived <- train.numeric[train.numeric$Sex == 1,]$Survived
```

Planteamos el siguiente contraste de hipótesis de dos muestras sobre la diferencia de medias, el cual es unilateral atendiendo a la formulación de la hipótesis alternativa:

$$H_0 : \mu_1 - \mu_2 = 0$$
$$H_1 : \mu_1 - \mu_2 < 0$$

donde μ_1 es la media de la población de la que se extrae la primera muestra y μ_2 es la media de la población de la que extrae la segunda. Tomaremos $\alpha = 0,05$.

```r
t.test( men.survived, women.survived, alternative = "less" )
```

```
##
##  Welch Two Sample t-test
##
## data:  men.survived and women.survived
## t = -18.672, df = 584.43, p-value < 2.2e-16
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##        -Inf -0.5043259
## sample estimates:
## mean of x mean of y
## 0.1889081 0.7420382
```

Puesto que obtenemos un p-valor menor que el valor de significación fijado, rechazamos la hipótesis nula. Por tanto, podemos concluir que, efectivamente, la probabilidad de sobrevivir al naufragio era mayor siendo mujer.

2.4.3.3. Modelo de regresión lineal

Tal y como se planteó en los objetivos de la actividad, buscamos predecir qué tipo de pasajeros sobrevivieron a la catástrofe. Así, se calculará un modelo de regresión lineal utilizando regresores cuantitativos con el que poder realizar las predicciones de supervivencia.

Para obtener un modelo de regresión lineal considerablemente eficiente, lo que haremos será calcular varios modelos de regresión utilizando las variables que estén más correladas con respecto a la variable **Survived**, según la tabla obtenida en el apartado 2.4.3.1. Entre todos los modelos que obtengamos, escogeremos el mejor utilizando como criterio aquel que presente un mayor coeficiente de determinación (R^2).

```r
model.1 <- lm( Survived ~ Sex, data = train.numeric )
summary( model.1 )
```

```
##
## Call:
## lm(formula = Survived ~ Sex, data = train.numeric)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7420 -0.1889 -0.1889  0.2580  0.8111
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.29517    0.04917   26.34   <2e-16 ***
## Sex         -0.55313    0.02866  -19.30   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4087 on 889 degrees of freedom
## Multiple R-squared:  0.2952, Adjusted R-squared:  0.2944
## F-statistic: 372.4 on 1 and 889 DF,  p-value: < 2.2e-16
```

```r
model.2 <- lm( Survived ~ Pclass, data = train.numeric )
summary( model.2 )
```

```
##
## Call:
## lm(formula = Survived ~ Pclass, data = train.numeric)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.6416 -0.2476 -0.2476  0.3584  0.7524
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.83863    0.04510   18.60   <2e-16 ***
## Pclass      -0.19700    0.01837  -10.72   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4581 on 889 degrees of freedom
## Multiple R-squared:  0.1146, Adjusted R-squared:  0.1136
## F-statistic:   115 on 1 and 889 DF,  p-value: < 2.2e-16
```

```
model.3 <- lm( Survived ~ Fare, data = train.numeric )
summary( model.3 )
```

```
##
## Call:
## lm(formula = Survived ~ Fare, data = train.numeric)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9653 -0.3391 -0.3222  0.6044  0.6973
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.3026994  0.0187849  16.114  < 2e-16 ***
## Fare        0.0025195  0.0003174   7.939 6.12e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4705 on 889 degrees of freedom
## Multiple R-squared:  0.06621,    Adjusted R-squared:  0.06516
## F-statistic: 63.03 on 1 and 889 DF,  p-value: 6.12e-15
```

En este caso, tenemos que el primer modelo es el más conveniente dado que tiene un mayor coeficiente de determinación. No obstante, dado que el coeficiente de determinación no es muy elevado, vamos a calcular un nuevo modelo utilizando un algoritmo supervisado.

2.4.3.4. Métodos de clasificación

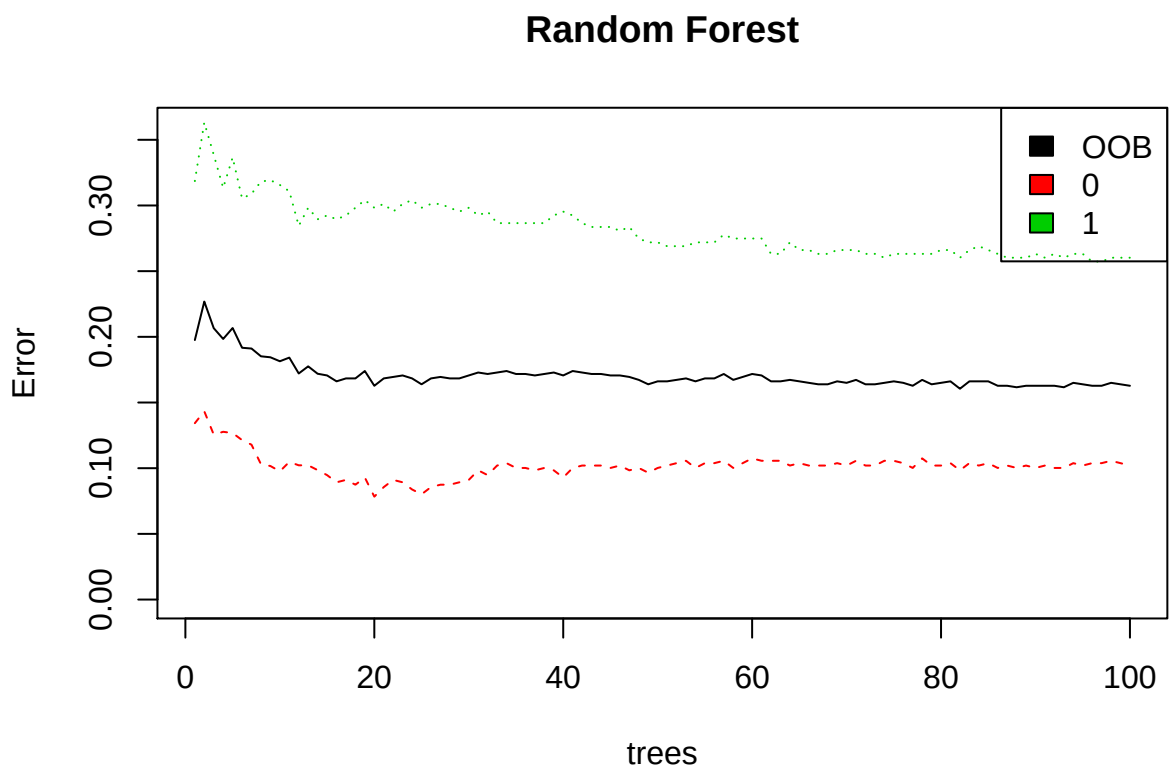
En este apartado, vamos a utilizar uno de los métodos de clasificación más sofisticados, el *Random Forest*. Calcularemos primero el modelo y, posteriormente, graficaremos el error del mismo:

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

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

```
model.rf <- randomForest( factor( Survived ) ~ Pclass + Sex + Age +
                            SibSp + Parch + Fare,
                          data = train.numeric, ntree = 100, importance = TRUE )
model.rf
```

```
##
## Call:
##  randomForest(formula = factor(Survived) ~ Pclass + Sex + Age +      SibSp + Parch + Fare, data = tra
##                Type of random forest: classification
##                      Number of trees: 100
## No. of variables tried at each split: 2
##
##         OOB estimate of  error rate: 16.27%
## Confusion matrix:
##     0   1 class.error
## 0 493  56   0.1020036
## 1  89 253   0.2602339
```

```
plot( model.rf, ylim = c( 0, 0.36 ), main = "Random Forest" )
legend( 'topright', colnames( model.rf$err.rate ), col = 1:3, fill = 1:3 )
```




```
importance <- importance( model.rf )
varImportance <- data.frame( Variables = row.names( importance ),
                             Importance = round( importance[ ,'MeanDecreaseGini'], 2 ) )

rankImportance <- varImportance %>%
```

```
  mutate( Rank = paste0( '#', dense_rank( desc( Importance ) ) ) )

prediction <- predict( model.rf, newdata = test.numeric )
```

En el gráfico, se puede apreciar como el error general se sitúa por debajo del 20%. Es llamativo como, comparando las tasas de error para cada valor de la variable **Survived**, vemos que el modelo es más preciso clasificando las muertes.

Adicionalmente, en el bloque anterior calculamos la importancia relativa de cada variable en el modelo calculado. Usaremos dichos cálculos en la sección 2.5.3.

Por último, utilizamos el modelo resultante para predecir la variable **Survived** en el conjunto de *test*. Una vez obtenidas las predicciones, procedemos a completar el conjunto de *test* y exportamos los datos a un nuevo fichero (**titanic-clean.csv**):

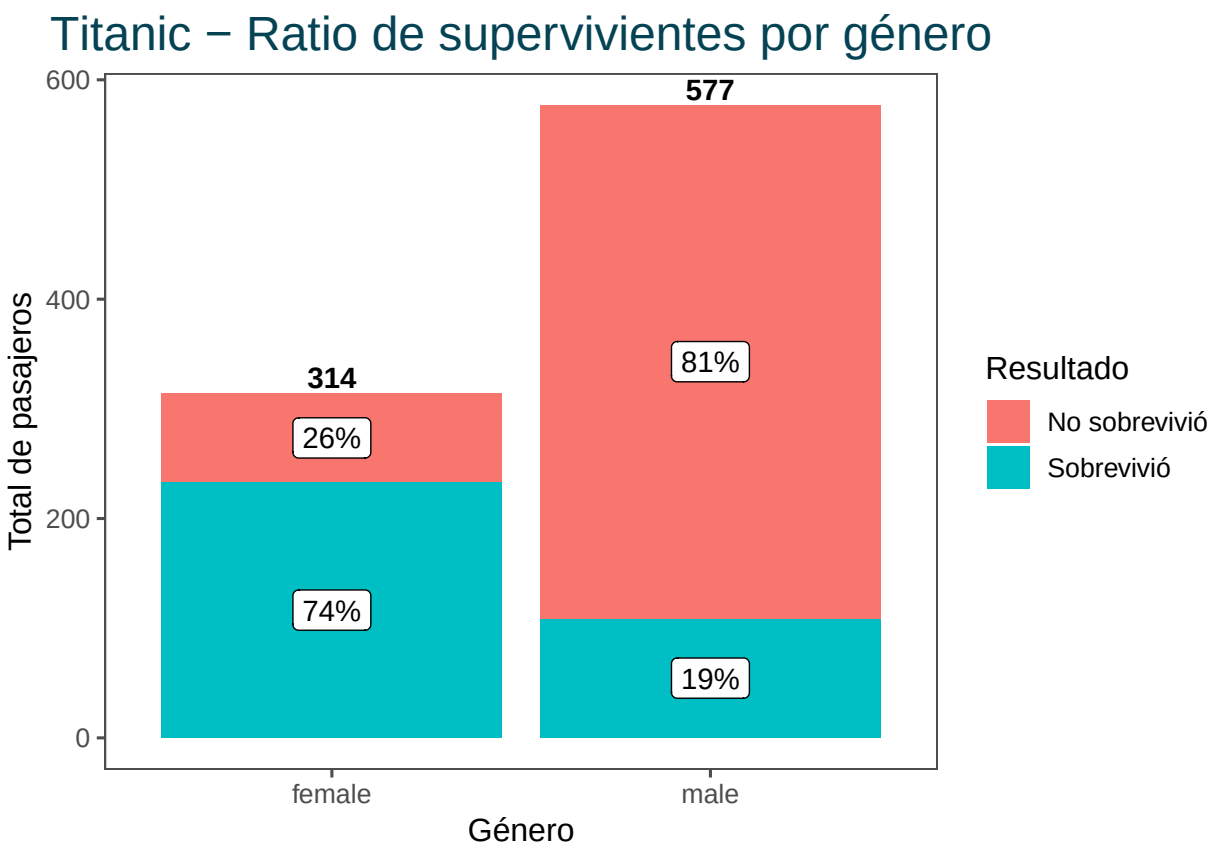
```
test$Survived <- as.integer( as.character( prediction ) )
full.clean <- bind_rows( train, test )
write.csv( full.clean, "../data/titanic-clean.csv", row.names = FALSE )
```

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

2.5.1. Ratio de supervivencia por género y categoría

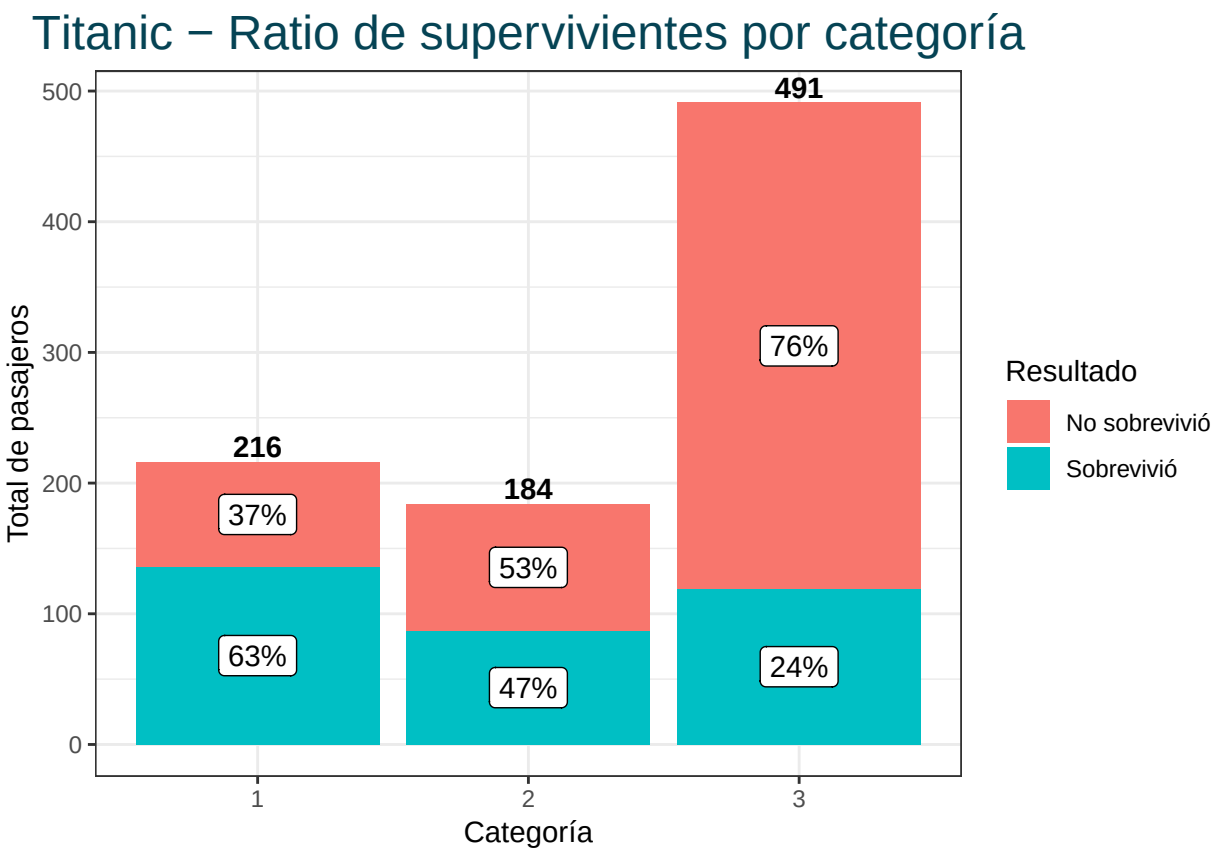
En el apartado 2.4.3.1. vimos cómo las variables **Sex** y **Pclass** presentaban la mayor correlación con la variable **Survived**. También vimos en el apartado 2.4.2. cómo el hecho de ser mujer significaba una mayor probabilidad de supervivencia. Vamos a visualizar estas afirmaciones a partir de los dados del conjunto de *training*.

```
gender <- train %>%
  group_by( Sex ) %>%
  summarise( Count = n() )

gender_ratio <- train %>%
  group_by( Sex, Survived ) %>%
  summarise( Count = n() ) %>%
  mutate( Percentage = round( Count / sum( Count ) * 100 ) )

train %>%
  ggplot() +
  geom_bar( aes( x = Sex, fill = factor( Survived ) ) ) +
  geom_text( data = gender,
             aes( x = Sex, y = Count, label = Count ),
             position = position_dodge( width = 0.9 ),
             vjust = -0.25,
             fontface = "bold" ) +
  geom_label( data = gender_ratio,
              aes( x = Sex, y = Count, label = paste0( Percentage, "%" ),
                   group = Survived ),
              position = position_stack( vjust = 0.5 ) ) +
  theme_few() +
  theme( plot.title = element_text( hjust = 0.5, size = 18, color = "#054354" ) ) +
  ggtitle( "Titanic - Ratio de supervivientes por género" ) +
  scale_x_discrete( name = "Género" ) +
  scale_y_continuous( name = "Total de pasajeros" ) +
  scale_fill_discrete( name = "Resultado", labels = c( "No sobrevivió", "Sobrevivió" ) )
```



Vemos como el 81% de la población masculina pereció durante el hundimiento del Titanic. En la población femenina, ese porcentaje se reduce al 26%.

```r
pclass <- train %>%
  group_by( Pclass ) %>%
  summarise( Count = n() )

pclass_ratio <- train %>%
  group_by( Pclass, Survived ) %>%
  summarise( Count = n() ) %>%
  mutate( Percentage = round( Count / sum( Count ) * 100 ) )

train %>%
  ggplot() +
  geom_bar( aes( x = factor( Pclass ), fill = factor( Survived ) ) ) +
  geom_text( data = pclass,
             aes( x = factor( Pclass ), y = Count, label = Count ),
             position = position_dodge( width = 0.9 ),
             vjust = -0.25,
             fontface = "bold" ) +
  geom_label( data = pclass_ratio,
             aes( x = factor( Pclass ), y = Count, label = paste0( Percentage, "%" ),
                  group = Survived ),
             position = position_stack( vjust = 0.5 ) ) +
  theme_bw() +
  theme( plot.title = element_text( hjust = 0.5, size = 18, color = "#054354" ) ) +
  ggtitle( "Titanic - Ratio de supervivientes por categoría" ) +
  scale_x_discrete( name = "Categoría" ) +
  scale_y_continuous( name = "Total de pasajeros" ) +
  scale_fill_discrete( name = "Resultado", labels = c( "No sobrevivió", "Sobrevivió" ) )
```

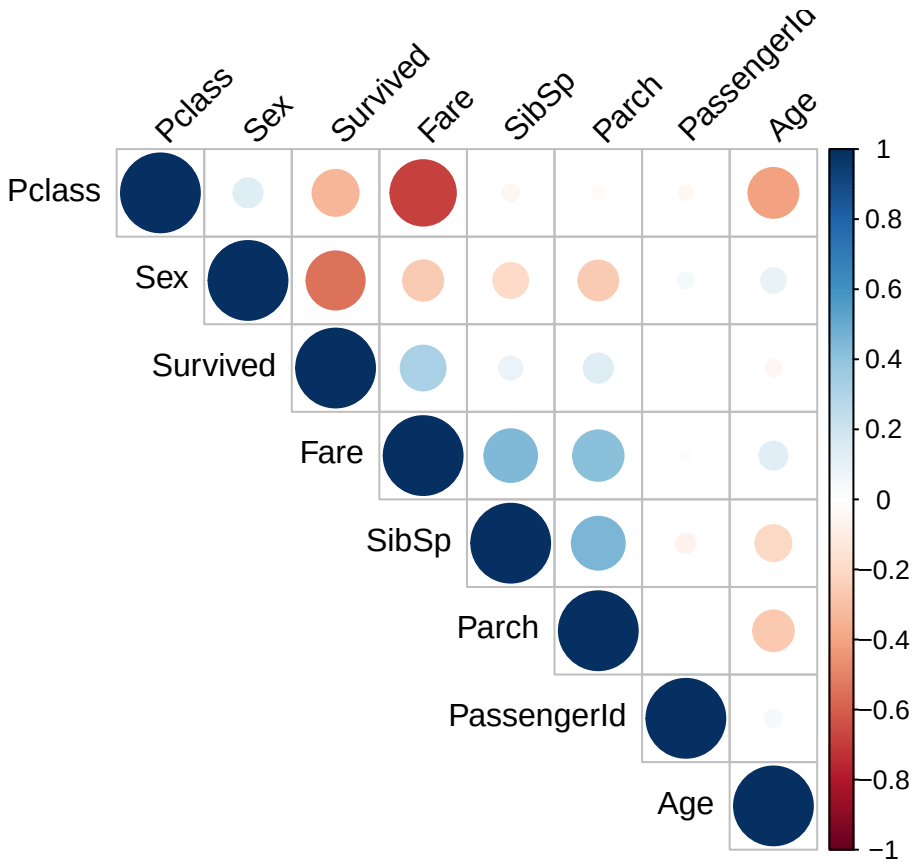


Vemos como el 76% de los pasajeros de tercera categoría no sobrevivieron al naufragio. Ese porcentaje se reduce al 53% para los pasajeros de segunda categoría y al 37% para los de primera.

2.5.2. ¿Qué variables cuantitativas influyen más en la supervivencia?

En el apartado 2.4.3.1. vimos la matriz de correlaciones entre nuestra variables cuantitativas. Vamos a visualizar la misma información mediante un gráfico:

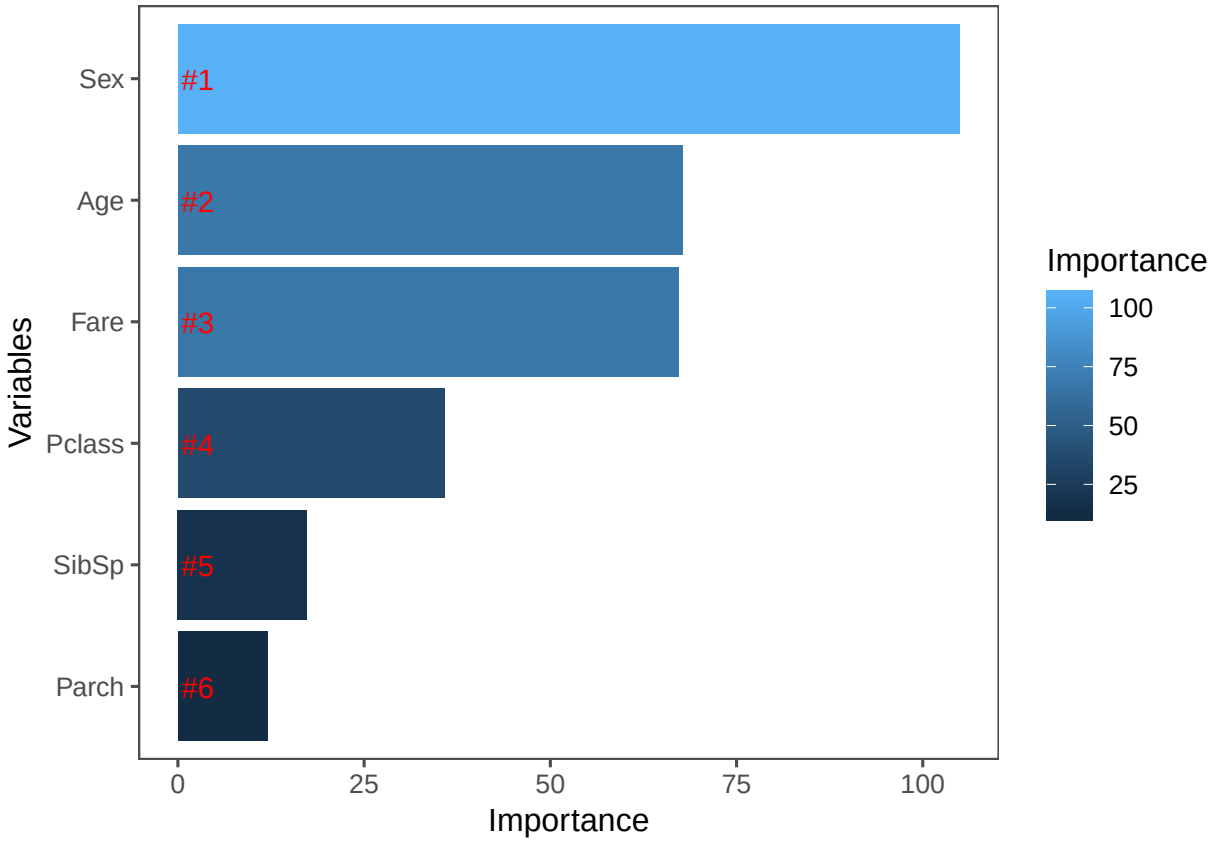
```
if( !require( corrplot ) ) {
    install.packages( 'corrplot', repos = 'http://cran.us.r-project.org' )
    library( corrplot )
}

corrplot( correlation.matrix, type = "upper", order = "hclust", tl.col = "black",
          tl.srt = 45 )
```



2.5.3. Importancia de las variables

Por último, recuperamos la clasificación sobre la importancia de las variables en el cálculo del modelo usando *Random Forest* para mostrar dicha información en el siguiente gráfico de barras:

```
ggplot( rankImportance,
        aes( x = reorder( Variables, Importance ), y = Importance, fill = Importance ) ) +
  geom_bar( stat = 'identity' ) +
  geom_text( aes( x = Variables, y = 0.5, label = Rank ),
             hjust = 0, vjust = 0.55, size = 4, colour = 'red' ) +
  labs( x = 'Variables' ) +
  coord_flip() +
  theme_few()
```



El género del pasajero (**Sex**) se sigue manteniendo como la variable más importante en la predicción de la supervivencia, pero la categoría (**Pclass**) cae el cuarto lugar. La tarifa (**Fare**) escala hasta el segundo.

2.6. Resolución del problema.

Se han realizado cuatro tipos de pruebas estadísticas sobre un conjunto de datos que se correspondía con datos relativos a los pasajeros del viaje inaugural del Titanic, con el motivo de cumplir en la medida de lo posible con el objetivo que se planteaba al comienzo. Para cada una de ellas, hemos podido ver cuáles son los resultados que arrojan, mediante tablas y gráficos, y qué conocimientos pueden extraerse a partir de ellas.

Así, el análisis de correlación y el contraste de hipótesis nos ha permitido conocer cuáles de estas variables ejercen una mayor influencia sobre la posibilidad de sobrevivir al naufragio, mientras que el modelo de clasificación obtenido mediante la aplicación de un *Random Forest* ha resultado de utilidad a la hora de realizar predicciones para esta variable dadas unas características concretas.

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 del segundo, se ha optado por incluir los valores extremos en los análisis dado que parecen no resultar del todo atípicos.

3. Recursos

- Calvo M., Subirats L., Pérez D. (2019). Introducción a la limpieza y análisis de los datos. Editorial UOC
- Megan Squire (2015). Clean Data. Packt Publishing Ltd
- Jiawei Han, Micheine Kamber, Jian Pei (2012). Data mining: concepts and techniques. Morgan Kaufmann
- Jason W. Osborne (2010). Data Cleaning Basics: Best Practices in Dealing with Extreme Scores. Newborn and Infant Nursing Reviews; 10 (1): pp. 1527-3369
- Peter Dalgaard (2008). Introductory statistics with R. Springer Science & Business Media
- Megan L. Risdal (2016). Exploring survival on the Titanic (https://www.kaggle.com/mrisdal/exploring-survival-on-the-titanic).

4. Contribuciones

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