

# Práctica 2: Limpieza y Validación de los Datos

*Beatriz Elena Jaramillo Gallego*

*08 de Mayo de 2018*

## Contents

<b>Detalles de la actividad</b>	<b>2</b>
Descripción . . . . .	2
Competencias . . . . .	2
Objetivos . . . . .	2
<b>Resolución Práctica</b>	<b>3</b>
1. Descripción del dataset . . . . .	3
2. Integración y selección . . . . .	4
3. Limpieza de los datos . . . . .	6
3.1. Ceros y elementos vacíos . . . . .	6
3.2. Valores extremos . . . . .	7
4. Análisis de los datos. . . . .	14
4.1. Selección de los grupos de datos a analizar . . . . .	14
4.2. Normalidad . . . . .	14
4.3. Pruebas Estadísticas . . . . .	17
5. Visualización . . . . .	26
6. Resolución del problema. . . . .	29
7. Código . . . . .	29
<b>Referencias</b>	<b>30</b>

# Detalles de la actividad

## 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.

## 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 continuarestudiando 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 Práctica

## 1. Descripción del dataset

Los datos para el análisis se ha obtenido a partir de este enlace en Kaggle Titanic: Machine Learning from Disaster (<https://www.kaggle.com/c/titanic>) y está constituido por 12 (variables) que presentan 891 pasajeros(filas o registros) en el archivo de train y 418 pasajeros(filas o registros) en el archivo de test.

Los datos se han dividido en dos grupos:

- Conjunto de entrenamiento (train.csv): El conjunto de entrenamiento se debe usar para construir sus modelos de aprendizaje automático
- Conjunto de prueba (test.csv): El conjunto de prueba se debe usar para ver qué tan bien se desempeña su modelo en datos no vistos.

### Variables

- PassengerId: Un identificador numerico del pasajero. Es una variable numérica.
- Survived: Varibale binaria donde se indica si el pasajero sobrevivio o no. (0 = No, 1 = Yes)
- Pclass: La clase en la que viajaba el pasajero. Es una variable numérica. (1 = 1st, 2 = 2nd, 3 = 3rd)
- Name: El nombre del pasajero. Es una variable nominal.
- Sex: El sexo del pasajero. Es una varuable nominal.
- Age: La edad del pasajero. Es una variable numérica.
- SibSp: Numero de familiares cosanguineos de la persona abordo del Titanic. Es una variable numérica
- Parch: Numero de familaires de diferente grado que acompañaban a la persona abordo del Titanic. Es una variable numérica
- Ticket: El ticket correspondiente al pasajero al momento del abordaje. Es una variable nominal.
- Fare: La tarifa del ticket segun la clase en la que abordo el pasajero. Es una variable numérica
- Cabin: El identificador de la cabina que utilizo la persona durante el viaje. Es una variable nominal
- Embarked: Indica el lugar de embarque de la persona. Es una variable nominal. (C = Cherbourg, Q = Queenstown, S = Southampton)

**Pregunta/problema pretende responder** Problema: Podria llegarse a determinar/predecir la supervivencia de los pasajeros del Titanic? Para reponder el anterior problema se dede tener en cuenta las características como edad, sexo y la tarifa del ticket según la clase en la que abordo el pasajero. Después de realizar un análisis de supervivencia podré responder esta pregunta.

**Notas Variables** Pclass: un proxy para el estado socio-económico (SES) 1er = superior 2do = Medio Tercero = Más bajo

Sibsp: el conjunto de datos define las relaciones familiares de esta manera ... Hermano = hermano, hermana, hermanastro, hermanastra Cónyuge = esposo, esposa (las amantes y los novios fueron ignorados)

Parch: El conjunto de datos define las relaciones familiares de esta manera ... Padre = madre, padre Niño = hija, hijo, hijastra, hijastro Algunos niños viajaban solo con una niñera, por lo tanto parch = 0 para ellos.

## 2. Integración y selección

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

Hare un vistazo general de como llegan los datos en el dataset train, el cual tiene las 12 variables anteriormente descritas.

```
#setwd("C:/Users/Admin/Dropbox/Master/Tipologia de Datos/PRACTICA 2")
titanic <- read.csv2("train.csv",header = TRUE,sep = ";",dec = ",",stringsAsFactors=FALSE)
cat(paste0("Carga fichero train.csv OK.", "\n\n"))
```

```
## Carga fichero train.csv OK.
```

```
str(titanic)
```

```
## '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 : chr "22" "38" "26" "35" ...
## $ 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 7.13 7.92 53.1 8.05 ...
## $ Cabin : chr "" "C85" "" "C123" ...
## $ Embarked : chr "S" "C" "S" "S" ...
```

Variables Categoricals: Survived, Sex, and Embarked. Ordinal: Pclass.

Variables Continuas: Age, Fare. Discrete: SibSp, Parch

Las Variables Cabin, Ticket y PassengerId son informativas y serán usadas en el análisis.

Haciendo algún tratamiento de datos. La variable **Fare** debe ser numerica y se encuentra como factor, la he convertido en numerica.

```
titanic$Name <- as.factor(titanic$Name)
titanic$Sex <- as.factor(titanic$Sex)
titanic$Ticket <- as.factor(titanic$Ticket)
titanic$Embarked <- as.factor(titanic$Embarked)
titanic$Age <- as.numeric(titanic$Age)
titanic$Fare <- as.numeric(titanic$Fare)

summary(titanic)
```

```
## PassengerId Survived Pclass
## Min. : 1.0 Min. :0.0000 Min. :1.000
## 1st Qu.:223.5 1st Qu.:0.0000 1st Qu.:2.000
## Median :446.0 Median :0.0000 Median :3.000
## 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
##
```

```

##                                     Name      Sex      Age
## Abbing, Mr. Anthony                : 1   female:314   Min.    : 0.42
## Abbott, Mr. Rossmore Edward        : 1   male  :577   1st Qu.:20.12
## Abbott, Mrs. Stanton (Rosa Hunt)   : 1                                     Median :28.00
## Abelson, Mr. Samuel                : 1                                     Mean   :29.70
## Abelson, Mrs. Samuel (Hannah Wzosky): 1                                     3rd Qu.:38.00
## Adahl, Mr. Mauritz Nils Martin     : 1                                     Max.   :80.00
## (Other)                            :885                                     NA's   :177
##      SibSp      Parch      Ticket      Fare
## Min.    :0.000   Min.    :0.0000   1601    : 7   Min.    : 6.438
## 1st Qu.:0.000   1st Qu.:0.0000   347082  : 7   1st Qu.: 7.925
## Median :0.000   Median :0.0000   CA. 2343: 7   Median :14.458
## Mean    :0.523   Mean    :0.3816   3101295 : 6   Mean    :32.865
## 3rd Qu.:1.000   3rd Qu.:0.0000   347088  : 6   3rd Qu.:31.000
## Max.    :8.000   Max.    :6.0000   CA 2144 : 6   Max.    :512.329
##                                     (Other) :852   NA's    :15
##      Cabin      Embarked
## Length:891      : 2
## Class :character C:168
## Mode  :character Q: 77
##                                     S:644
##
##
##
##

```

### 3. Limpieza de los datos

#### 3.1. Ceros y elementos vacíos

¿Los datos contienen ceros o elementos vacíos? ¿Cómo gestionarías cada uno de estos casos?

Hay 263 datos **NA** en la variable **Age**, hay 418 NA en la variable **Survived**, que son los valores que anteriormente adicione del dataset test.

```
# Números de valores desconocidos por campo
sapply(titanic, function(x) sum(is.na(x)))
```

```
## PassengerId    Survived    Pclass      Name      Sex      Age
##           0           0           0           0           0      177
##      SibSp      Parch      Ticket      Fare      Cabin    Embarked
##           0           0           0          15           0           0
```

```
cat("\n\n")
```

```
cat(paste0("Valores vacios Age ",round((177/891)*100,2)), "%")
```

```
## Valores vacios Age 19.87 %
```

```
cat("\n\n")
```

```
cat(paste0("Valores vacios Fare ",round((15/891)*100,2)), "%")
```

```
## Valores vacios Fare 1.68 %
```

```
cat("\n\n")
```

```
# Para aquellos pasajeros que se desconoce la cabina, se le cambia su vacio por un string que indica d
titanic$Cabin[titanic$Cabin==""] <- "Unknown"
titanic$Cabin <- as.factor(titanic$Cabin)
```

```
# Hay 2 pasajeros en la columna Embarked que estan vacios y la mayoría embarco por S entonces lo he agr
errores = which(titanic$Embarked=="")
titanic$Embarked[errores] = "S"
titanic$Embarked <- factor(titanic$Embarked)
rm(errores)
```

Para manejar los registros que contienen valores desconocidos para algún campo, una opción podría ser eliminar los registros que incluyen este tipo de valores de la variable **Age**, pero ello supondría desaprovechar el 19.87% de esta información y para la variable **Fare** quitar los NA upondría desaprovechar el 1.68% de esta información

Se empleará el método para imputa de una manera sofisticada para que no toda la matriz de distancia tenga que calcularse: la imputación basada en k vecinos más próximos (en inglés, kNN-imputation). Por lo tanto, la implementación del paquete **VIM** también es aplicable para conjuntos de datos razonablemente grandes

```
# Imputación de valores mediante la función kNN() del paquete VIM
suppressWarnings(suppressMessages(library(VIM)))
titanic$Age <- kNN(titanic)$Age
titanic$Fare <- kNN(titanic)$Fare
sapply(titanic, function(x) sum(is.na(x)))
```

```
## PassengerId    Survived    Pclass      Name      Sex      Age
##           0           0           0           0           0           0
```

```
##      SibSp      Parch      Ticket      Fare      Cabin      Embarked
##          0          0          0          0          0          0
```

```
summary(titanic)
```

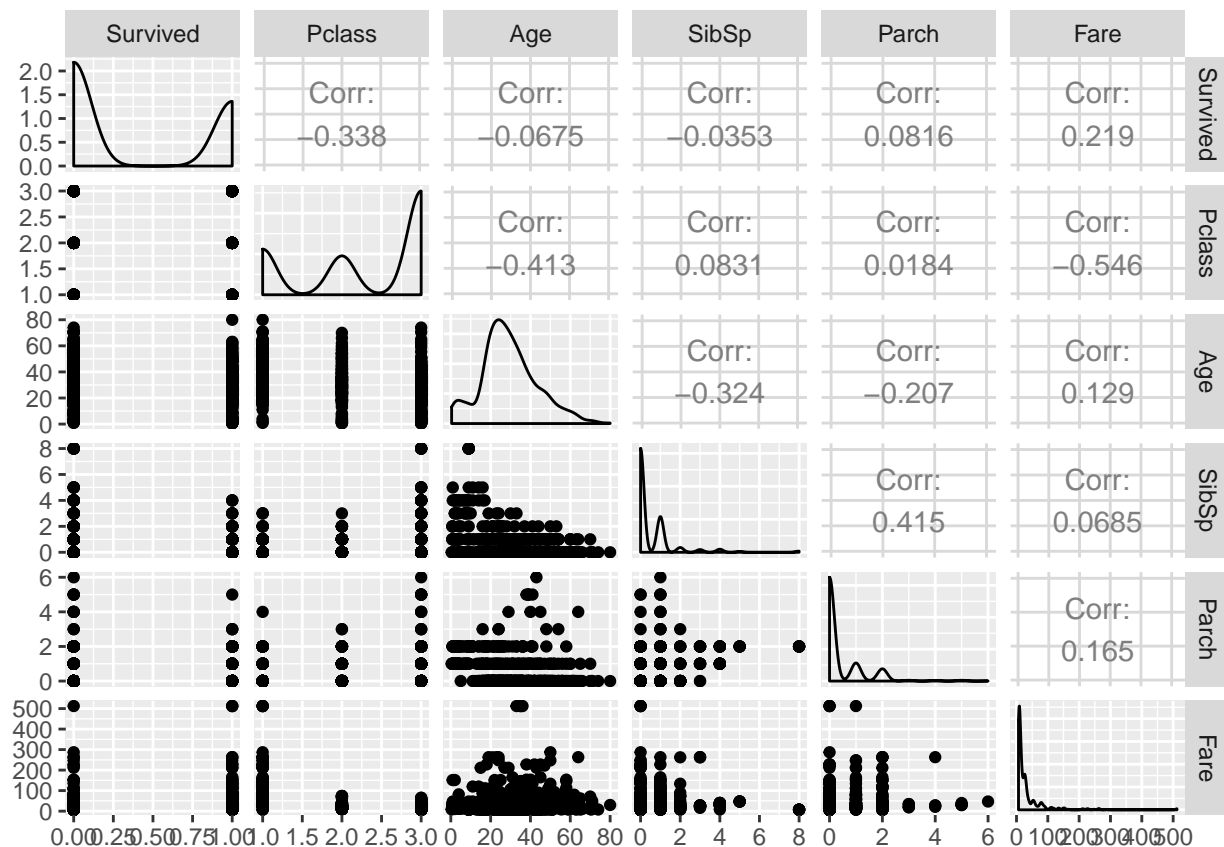
```
##      PassengerId      Survived      Pclass
##      Min.       : 1.0      Min.       :0.0000      Min.       :1.000
##      1st Qu.:223.5      1st Qu.:0.0000      1st Qu.:2.000
##      Median :446.0      Median :0.0000      Median :3.000
##      Mean   :446.0      Mean   :0.3838      Mean    :2.309
##      3rd Qu.:668.5      3rd Qu.:1.0000      3rd Qu.:3.000
##      Max.   :891.0      Max.   :1.0000      Max.    :3.000
##
##
##              Name      Sex      Age
## Abbing, Mr. Anthony      : 1  female:314      Min.    : 0.42
## Abbott, Mr. Rossmore Edward      : 1  male  :577      1st Qu.:21.00
## Abbott, Mrs. Stanton (Rosa Hunt)      : 1                      Median :28.00
## Abelson, Mr. Samuel      : 1                      Mean   :29.39
## Abelson, Mrs. Samuel (Hannah Wizesky): 1                      3rd Qu.:37.00
## Adahl, Mr. Mauritz Nils Martin      : 1                      Max.   :80.00
## (Other)                          :885
##      SibSp      Parch      Ticket      Fare
##      Min.       :0.000      Min.       :0.0000      1601      : 7      Min.       : 6.438
##      1st Qu.:0.000      1st Qu.:0.0000      347082     : 7      1st Qu.: 7.925
##      Median :0.000      Median :0.0000      CA. 2343: 7      Median : 14.454
##      Mean   :0.523      Mean   :0.3816      3101295 : 6      Mean   : 32.694
##      3rd Qu.:1.000      3rd Qu.:0.0000      347088     : 6      3rd Qu.: 31.000
##      Max.   :8.000      Max.   :6.0000      CA 2144 : 6      Max.   :512.329
##
##              (Other) :852
##
##      Cabin      Embarked
##      Unknown      :687      C:168
##      B96 B98      : 4      Q: 77
##      C23 C25 C27: 4      S:646
##      G6          : 4
##      C22 C26      : 3
##      D           : 3
##      (Other)      :186
```

### 3.2. Valores extremos

#### Identificación y tratamiento de valores extremos

Los valores extremos o outliers son aquellos que parecen no ser congruentes sin los comparamos con el resto de los datos.

```
ggpairs(titanic[c(2,3,6,7,8,10)])
```



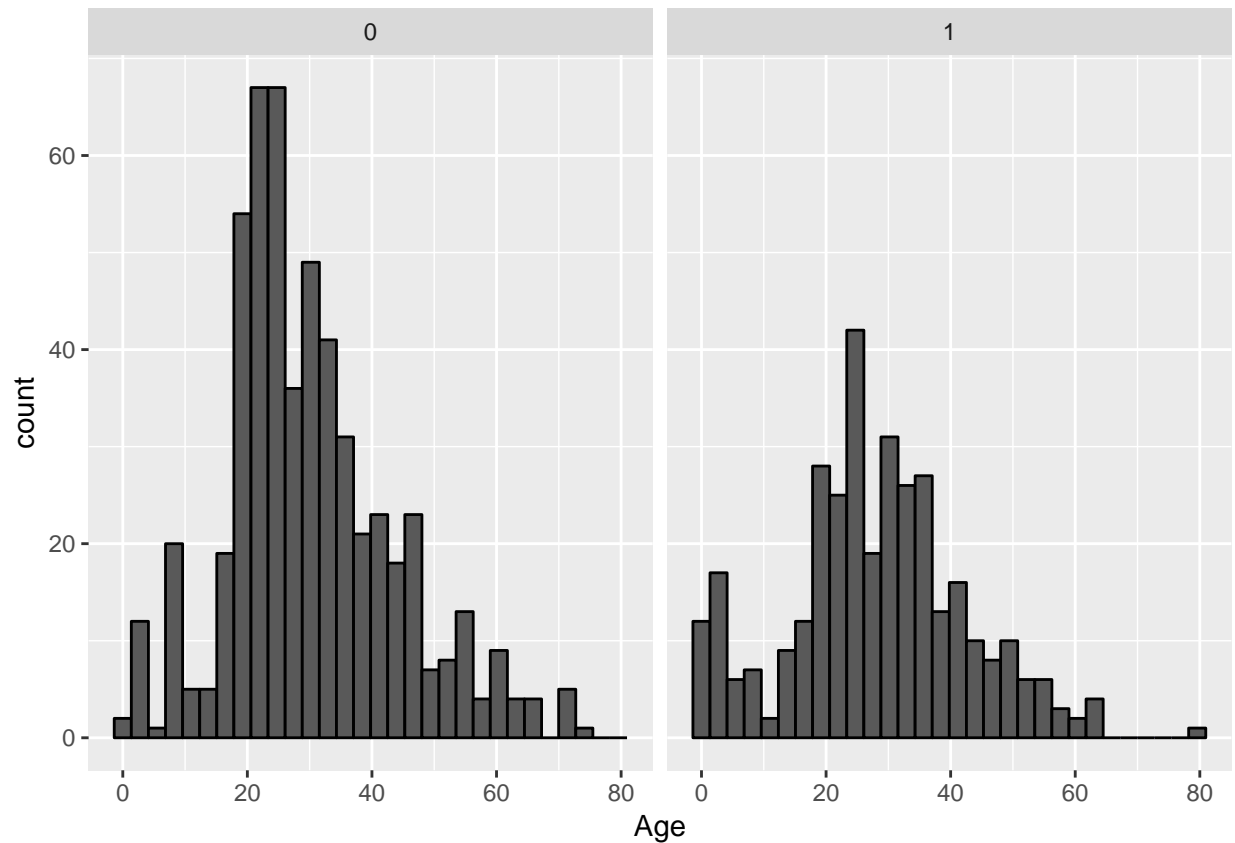
*#gráfico la variable Age, sin NA*

```
ggplot(titanic, aes(x = Age)) +
  geom_histogram(fill = "darkblue", alpha = .5) +
  geom_histogram(colour = "black")+
  facet_wrap(~ Survived)
```

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.





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

```
## [1] 66.0 65.0 71.0 70.5 62.0 63.0 65.0 64.0 65.0 63.0 71.0 64.0 62.0 62.0
```

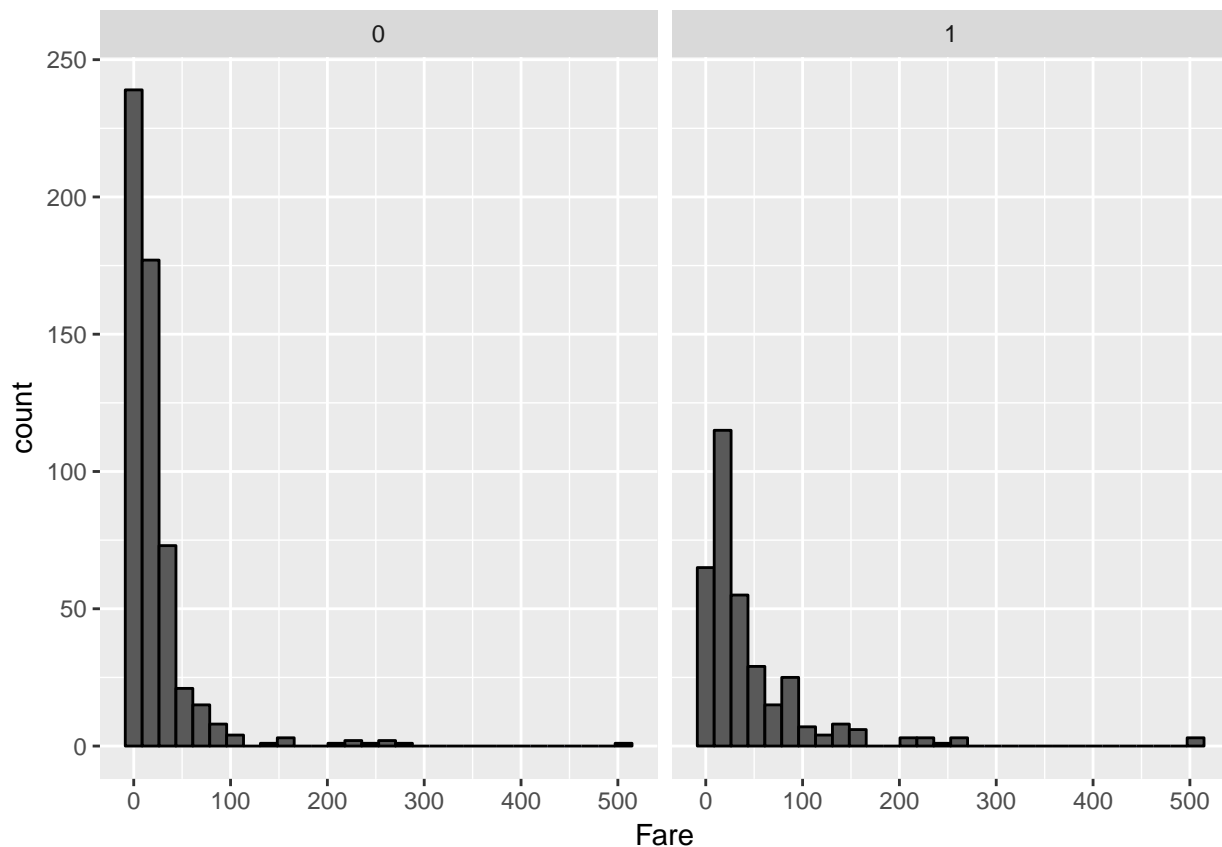
```
## [15] 80.0 70.0 70.0 62.0 74.0
```

```
#gráfico la variable Fare
```

```
ggplot(titanic, aes(x = Fare)) +  
  geom_histogram(fill = "darkblue", alpha = .5) +  
  geom_histogram(colour = "black")+  
  facet_wrap(~ Survived)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



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

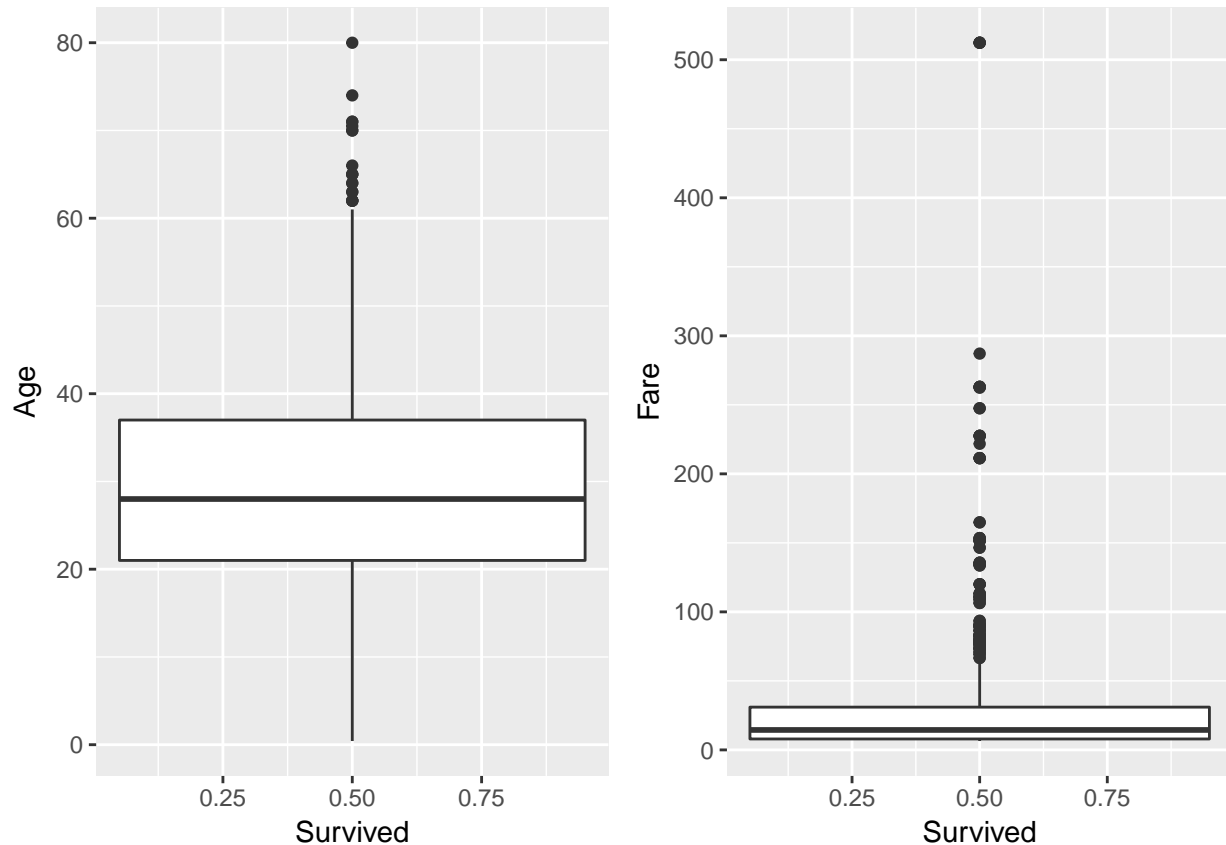
```
## [1] 263.0000 146.5200 82.1700 76.7290 80.0000 83.4800 73.5000
## [8] 263.0000 77.2900 247.5200 73.5000 77.2900 79.2000 66.6000
## [15] 287.1250 146.5200 113.2750 76.2900 90.0000 83.4800 90.0000
## [22] 79.2000 86.5000 512.3290 79.6500 153.4630 135.6300 78.8500
## [29] 91.0800 151.5500 247.5200 151.5500 110.8830 108.9000 83.1600
## [36] 164.8700 134.5000 135.6300 153.4630 133.6500 66.6000 134.5000
## [43] 263.0000 75.2500 69.3000 135.6300 82.1700 211.5000 227.5250
## [50] 73.5000 120.0000 113.2750 90.0000 120.0000 263.0000 81.8600
## [57] 89.1000 91.0800 90.0000 78.2670 151.5500 86.5000 108.9000
## [64] 93.5000 221.7800 106.4250 71.0000 106.4250 110.8830 227.5250
## [71] 79.6500 110.8830 79.6500 79.2000 78.2670 153.4630 69.3000
## [78] 76.7290 73.5000 113.2750 133.6500 73.5000 512.3290 76.7290
## [85] 211.3380 110.8830 227.5250 151.5500 227.5250 211.3380 512.3290
## [92] 78.8500 262.3750 71.0000 86.5000 120.0000 77.9580 211.3380
## [99] 79.2000 120.0000 93.5000 79.2000 80.0000 83.1580 89.1040
## [106] 164.8670 512.3292 83.1580
```

```
p <- ggplot(titanic, aes(x=Survived, y=Age, fill=Survived)) +
  geom_boxplot()
q <- ggplot(titanic, aes(x=Survived, y=Fare, fill=Survived)) +
  geom_boxplot()

grid.arrange(p,q,ncol = 2)
```

```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
```

```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
```



En este caso los datos atipicos surgen de un error de procedimiento, tales como la entrada de datos o un error de codificación. Estos casos atipicos deberan subsanarse en el filtrado de los datos, y si no se puede, deberan eliminarse del analisis o recodificarse como datos ausentes.

Estos datos “atipicos” los dejare como están ya que los he filtrado y estos pasajeros son los que viajan en Primera Clase y 5 casos que viajan en 2 clase y el valor de su pasaje concuerda.

Cuando termine de realizar el análisis sin quitar estos datos, realizare un análisis similar y quitare estos datos para posteriormente hacer una comparación y concluir si estos 108 casos tienen un valor significativo en este análisis, pero a priori podría decirse que tiene un valor significativo ya que se sabe que las personas de 1 clase son las que tenían mayor posibilidad de sobrevivir con respecto a las demás clases.

```
length(boxplot.stats(titanic$Fare)$out)
```

```
## [1] 108
```

```
ati_class1 <- filter(titanic, titanic$Fare >=66)
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
head(ati_class1)
```

```
## PassengerId Survived Pclass
## 1          28         0      1
```

```
## 2      32      1      1
## 3      35      0      1
## 4      53      1      1
## 5      62      1      1
## 6      63      0      1
##
##              Name      Sex Age SibSp Parch
## 1      Fortune, Mr. Charles Alexander   male  19      3      2
## 2 Spencer, Mrs. William Augustus (Marie Eugenie) female  35      1      0
## 3              Meyer, Mr. Edgar Joseph   male  28      1      0
## 4      Harper, Mrs. Henry Sleeper (Myna Haxtun) female  49      1      0
## 5              Icard, Miss. Amelie female  38      0      0
## 6      Harris, Mr. Henry Birkhardt   male  45      1      0
##      Ticket      Fare      Cabin Embarked
## 1      19950 263.000 C23 C25 C27      S
## 2 PC 17569 146.520      B78      C
## 3 PC 17604 82.170      Unknown      C
## 4 PC 17572 76.729      D33      C
## 5      113572 80.000      B28      S
## 6      36973 83.480      C83      S
```

Otras variables numéricas que se utilizan en el problema.

```
# La cantidad de personas en cada clase
p <- ggplot(titanic, aes(x=Pclass)) +
  geom_histogram()

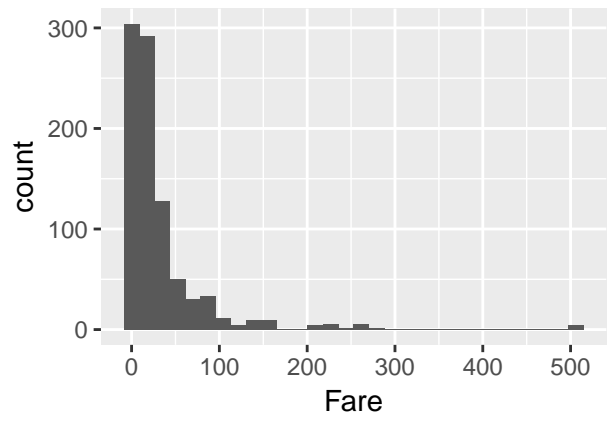
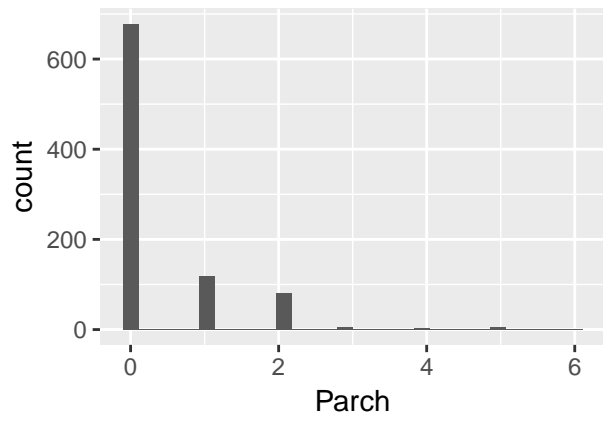
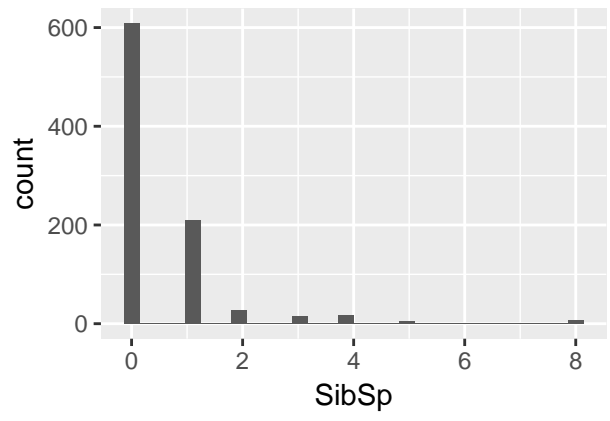
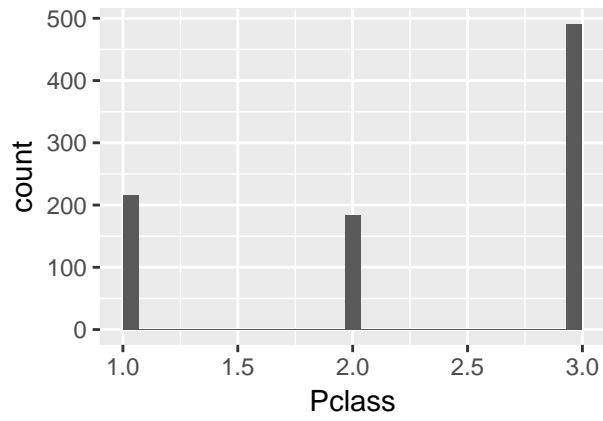
# La cantidad hermanos/conyuges de la personas
r <- ggplot(titanic, aes(x=SibSp)) +
  geom_histogram()

# La cantidad de parientes y niños abordo
s <- ggplot(titanic, aes(x=Parch)) +
  geom_histogram()

# La tarifas de los tickets por personas
t <- ggplot(titanic, aes(x=Fare)) +
  geom_histogram()

grid.arrange(p,r,s,t,ncol = 2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



## 4. Análisis de los datos.

### 4.1. Selección de los grupos de datos a analizar

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

Los métodos estadísticos utilizados en el análisis de supervivencia frecuentemente presentan cierta complejidad en los cálculos, esto debido a la naturaleza de las observaciones las cuales suelen presentar censura y/o truncamiento. La utilización de software estadístico para realizar dicho análisis se ha vuelto indispensable en la práctica.

#### Estimador de Kaplan-Meier y Fleming-Harrington

Los estimadores de Kaplan-Meier y Fleming-Harrington para la función de supervivencia es obtenido a través paquete estadístico survival mediante la función survfit(). Esta función en su forma más sencilla, solo requiere un objeto de supervivencia creado por la función Surv(). Los argumentos de la función survfit() son los siguientes:

1. formula. Un objeto fórmula y x, que debe tener un objeto Surv como variable respuesta a la izquierda del " ~ " y, si se desea, el nombre de las covariables por la derecha. Uno de los términos puede ser un objeto estrato. Para una sola curva de supervivencia del lado derecho se coloca 1.
2. data. objeto data frame donde están los datos.
3. type. Tipo de estimador: "kaplan-meier" o "fleming-harrington".

**Árboles de Decisión** Un Árbol de Decisión es un modelo de predicción utilizado para modelar construcciones lógicas sobre el contenido de bases de datos, para la toma de decisiones en base a esas entradas, es decir, es una forma gráfica y analítica de representar todos los eventos que pueden surgir a partir de una decisión asumida en cierto momento. Estoy comenzando a comprender esta herramienta estadística, toda vez que un heterogéneo conjunto de variables condicionan los resultados clínicos de nuestros pacientes y nos pueden ayudar a entender mejor el proceso de salud enfermedad, sobre todo en el área de la patología y medicina oral.

### 4.2. Normalidad

#### Comprobación de la normalidad y homogeneidad de la varianza.

Para la comprobación de que los valores que toman nuestras variables cuantitativas provienen de una población con distribución normal, utilizaremos la prueba de normalidad de Anderson-Darling.

Con esto, comprobaremos que para que cada prueba se obtiene un p-valor superior al nivel de significación prefijado  $\alpha = 0,05$ . Si esto se cumple, entonces se considera que la variable se distribuye normalmente.

(He incluido Age y Fare, ya que después de haber hecho las correcciones pertinentes porque estas variables son también numéricas)

```
library(nortest)
alpha = 0.05
col.names = colnames(titanic)
for (i in 1:ncol(titanic)) {
  if (i == 1) cat("Variables que no se distribuyen normalmente:\n")
  if (is.integer(titanic[,i]) | is.numeric(titanic[,i])) {
    p_val = ad.test(titanic[,i])$p.value
    if (p_val < alpha) {
      cat(col.names[i])
    }
  }
}
```

```

        # Format output
    if (i < ncol(titanic) - 1) cat(", ")
    }
}
}

```

```

## Variables que no se distribuyen normalmente:
## PassengerId, Survived, Pclass, Age, SibSp, Parch, Fare,

```

Luego se estudia la homogeneidad de varianzas mediante la aplicación del test de Fligner-Killeen. En este caso, estudiaremos esta homogeneidad en cuanto a los grupos conformados por las edades de los pasajeros que han o no sobrevivido. En el siguiente test, la hipótesis nula consiste en que ambas varianzas son iguales.

```

fligner.test(Age ~ Survived, data = titanic)

```

```

##
## Fligner-Killeen test of homogeneity of variances
##
## data: Age by Survived
## Fligner-Killeen:med chi-squared = 0.98606, df = 1, p-value =
## 0.3207

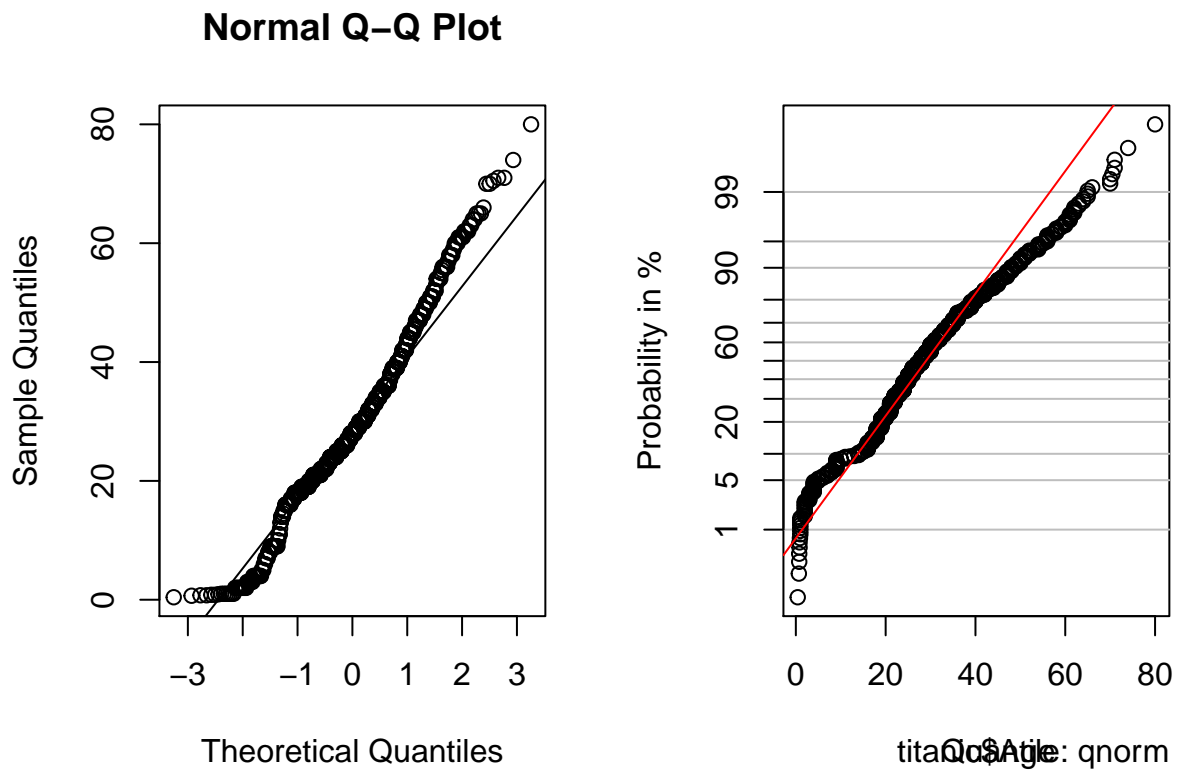
```

Puesto que obtenemos un p-valor superior a 0,05, aceptamos la hipótesis de que las varianzas de ambas muestras son homogéneas.

```

par(mfrow=c(1,2))
qqnorm(titanic$Age); qqline(titanic$Age)
# p-plot: you should observe a good fit of the straight line
probplot(titanic$Age, qdist=qnorm)

```

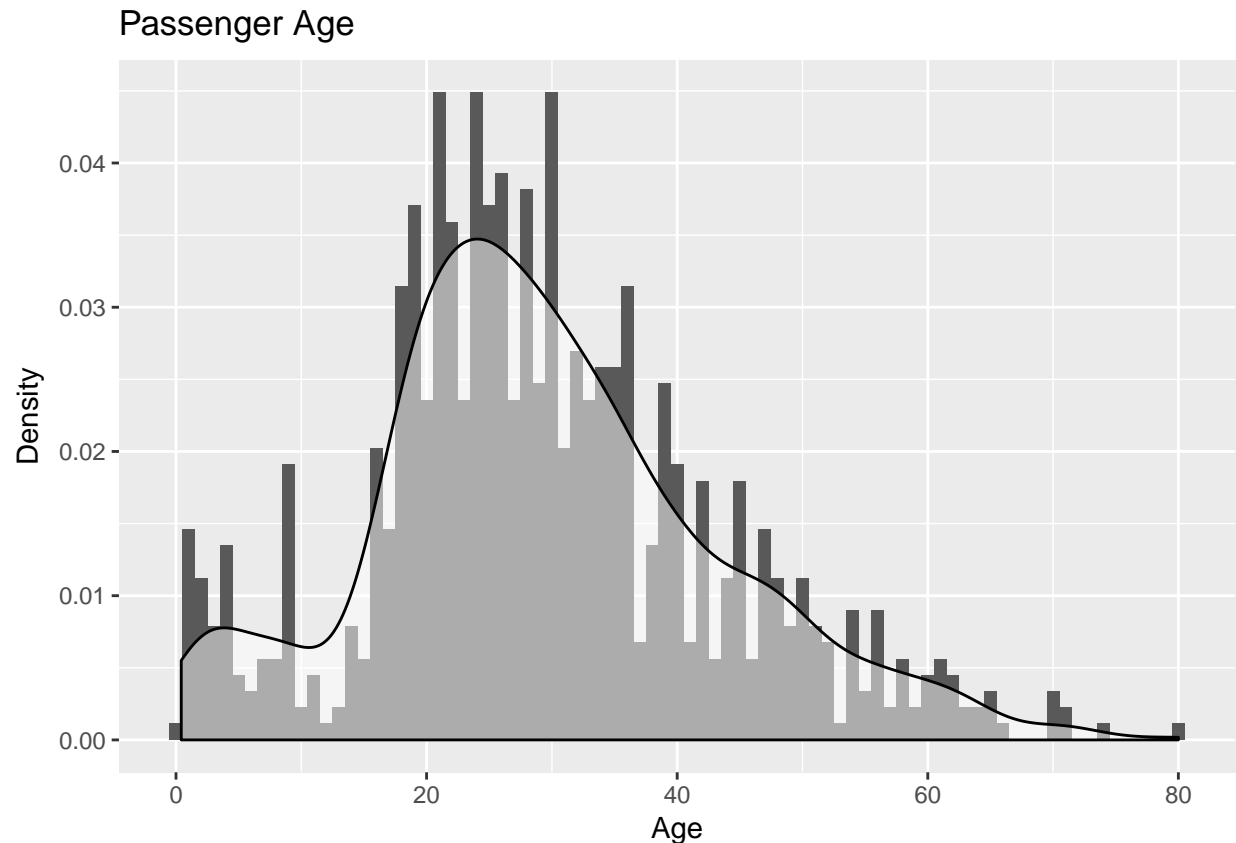


Las desviaciones de la línea recta son leves. Esto indica una distribución normal.

El histograma:

```
# Let's overlay a PDG on a histogram of age
ggplot(titanic, aes(x=Age)) +
  ggtitle("Passenger Age") +
  xlab("Age") +
  ylab("Density") +
  geom_histogram(aes(y=..density..), binwidth=1)+
  geom_density(alpha=.5, fill="#FFFFFF")
```





#### 4.3. Pruebas Estadísticas

##### Aplicación de pruebas estadísticas para comparar los grupos de datos

En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc.

Ahora poseemos un mejor conocimiento de los elementos y atributos de las variables, pero esta no nos dicen la manera en que se relacionan. Para ello aplicaremos un análisis de componentes principales al conjunto de datos.

**Correlación entre variables** En primer lugar, procedemos a realizar un análisis de correlación entre las distintas variables para determinar cuáles de ellas ejercen una mayor influencia sobre la supervivencia. Para ello, se utilizará el coeficiente de correlación de Spearman, puesto que hemos visto que tenemos datos que no siguen una distribución normal.

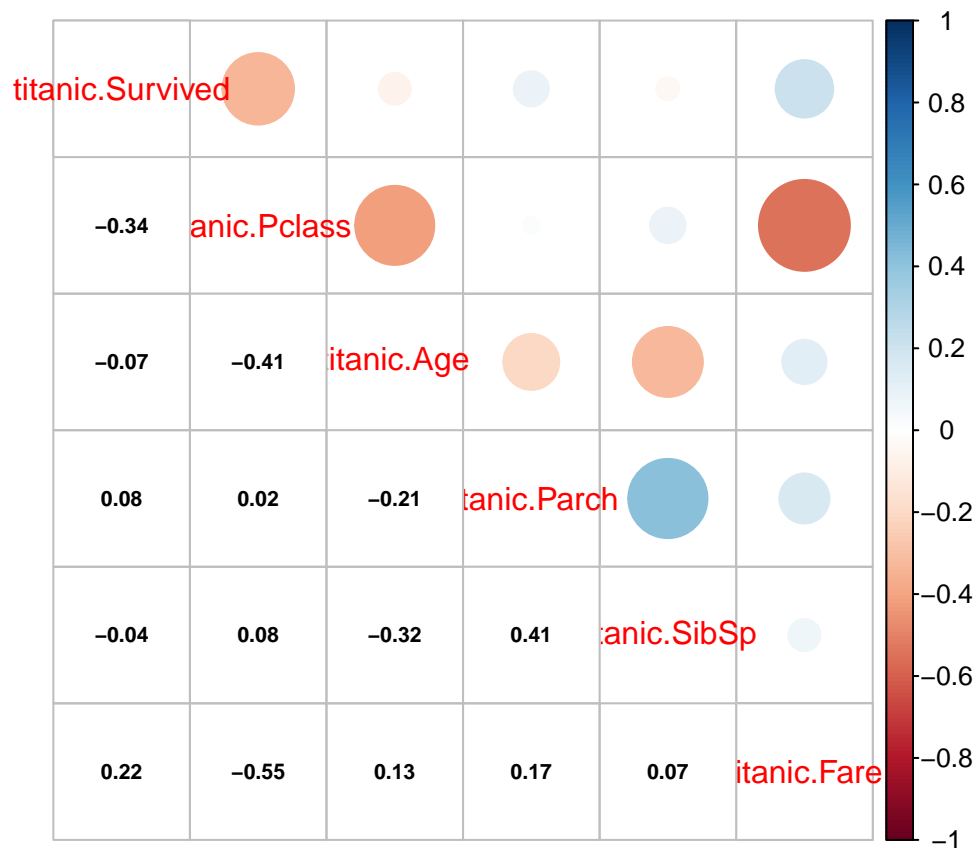
```
#variables cuantitativas
cuant <- data.frame(
  titanic$Survived,titanic$Pclass,titanic$Age,titanic$Parch,titanic$SibSp,titanic$Fare)
# Cuales variables parecieran tener alguna relacion entre si
M <- cor(cuant)
round(M,2)
```

```
##               titanic.Survived titanic.Pclass titanic.Age titanic.Parch
## titanic.Survived             1.00         -0.34      -0.07         0.08
## titanic.Pclass               -0.34             1.00      -0.41         0.02
```

```
## titanic.Age          -0.07          -0.41          1.00          -0.21
## titanic.Parch        0.08           0.02         -0.21           1.00
## titanic.SibSp        -0.04           0.08         -0.32           0.41
## titanic.Fare         0.22          -0.55          0.13           0.17
##          titanic.SibSp titanic.Fare
## titanic.Survived     -0.04          0.22
## titanic.Pclass        0.08         -0.55
## titanic.Age          -0.32          0.13
## titanic.Parch         0.41          0.17
## titanic.SibSp         1.00          0.07
## titanic.Fare          0.07          1.00
```

*#Correlación*

```
corrplot.mixed(M, lower.col = "black", number.cex = .7)
```



Las variables Survived con Pclass y Fare están altamente correlacionadas.

## Regresión Lineal

Podría resultar útil realizar predicciones sobre la supervivencia de los pasajeros del Titanic dadas las variables que tenemos a nuestra disposición. Calcularé un modelo de regresión lineal utilizando regresores tanto cuantitativos como cualitativos con el que poder realizar las predicciones para predecir la supervivencia y ver el peso y sentido de cada una de las variables.

```
(gender_model <- lm(Survived ~ Sex, data=titanic))
```

```
##
## Call:
## lm(formula = Survived ~ Sex, data = titanic)
##
## Coefficients:
## (Intercept)      Sexmale
##      0.7420      -0.5531
```

```
summary(gender_model)$r.squared
```

```
## [1] 0.2952307
```

Este modelo predice que las mujeres tienen un 74.2% de supervivencia y los hombres tienen un  $74.2 - 55.3 = 18.9\%$  de probabilidad de supervivencia.

Haremos una regresión lineal con todas las variables.

```
(gender_model2 <- lm(Survived~Pclass+Sex+Age+SibSp+Parch+Fare+Embarked, data=titanic))
```

```
##
## Call:
## lm(formula = Survived ~ Pclass + Sex + Age + SibSp + Parch +
##      Fare + Embarked, data = titanic)
##
## Coefficients:
## (Intercept)      Pclass      Sexmale          Age      SibSp
##  1.4675856  -0.2045654  -0.4978270  -0.0069178  -0.0497387
##      Parch      Fare      EmbarkedQ      EmbarkedS
## -0.0116240  -0.0001875  -0.0066888  -0.0670940
```

```
summary(gender_model2)$r.squared
```

```
## [1] 0.405258
```

Luego utilizando las variables que estén más correlacionadas con respecto a Survived. Así, de entre todos los modelos que tengamos, escogeremos el mejor utilizando como criterio aquel que presente un mayor coeficiente de determinación ( $R^2$ ).

```
(gender_model3 <- lm(Survived~Pclass+Fare, data=titanic))
```

```
##
## Call:
## lm(formula = Survived ~ Pclass + Fare, data = titanic)
##
## Coefficients:
## (Intercept)      Pclass      Fare
##  0.788525  -0.1817624  0.0004469
```

```
summary(gender_model3)$r.squared
```

```
## [1] 0.1161798
```

Vemos que habian 313 mujeres y 578 hombres lo que representa un 35,13% de mujeres y un 64,87% de hombres en el conjunto de entrenamiento. Ahora relacionemos esta información con la sobrevivencia de los pasajeros y tambien la tasa de sobrevivencia de los niños.

```

# Con respecto al Sexo de las personas podemos ver la siguiente distribucion
table(titanic$Sex)

##
## female    male
##      314    577

# Siguiendo el contexto del problema examinemos el sexo de los sobrevivientes
table(titanic$Sex,titanic$Survived)

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

Surv_age <- ifelse(titanic$Age<=18,"Niño","Adulto")
table(Surv_age,titanic$Survived)

##
## Surv_age    0    1
## Adulto  468 266
## Niño    81  76

table(Surv_age,titanic$Survived, by=titanic$Sex) # Sobrevivieron mas niñas que niños

## , , by = female
##
##
## Surv_age    0    1
## Adulto   52 183
## Niño     29  50
##
## , , by = male
##
##
## Surv_age    0    1
## Adulto  416  83
## Niño     52  26

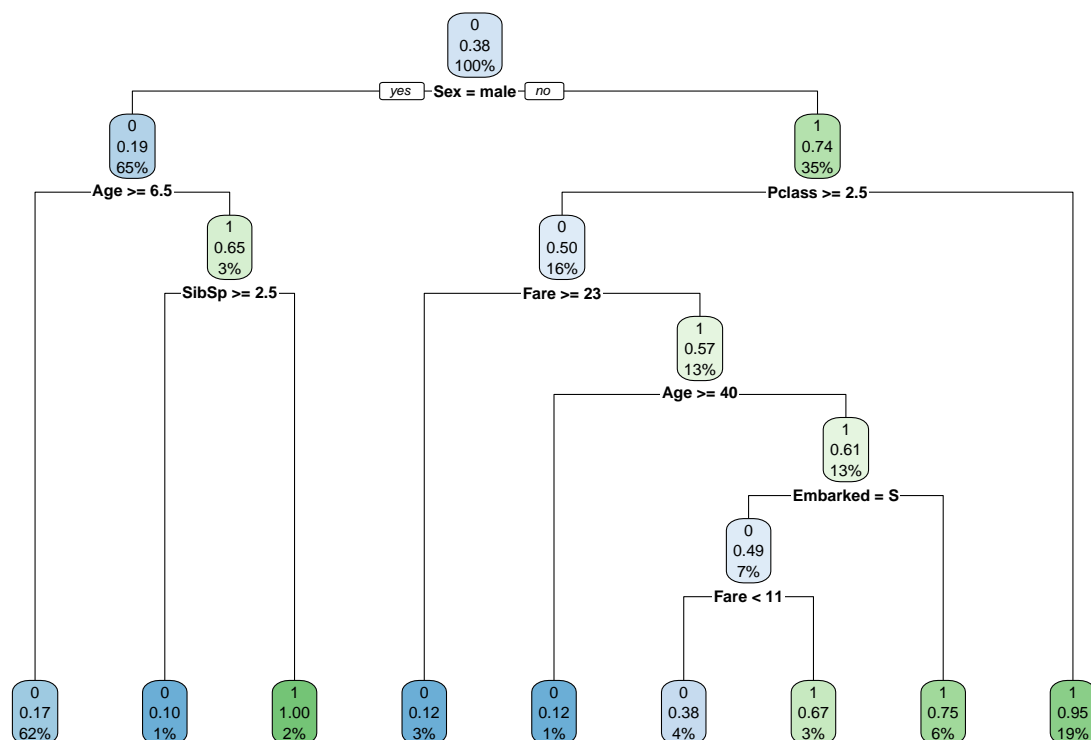
```

Ahora comparemos usando arboles decision como metodo para generar un modelo predictivo.

```

# Tomamos un subconjunto de los mismos
titanic2 <- subset(titanic,select=-c(PassengerId,Name,Ticket,Cabin))
modelo_arbol <- rpart(Survived ~ ., data=titanic2,method="class")
#arbol de decisión
rpart.plot(modelo_arbol)

```



```

prediction <- predict(modelo_arbol,titanic2[,1])
prediction <- ifelse(prediction<0.5,0,1)
matriz_conf <- table(prediction = prediction[,2], true = titanic2[,1])
matriz_conf

```

```

##           true
## prediction  0   1
##           0 519 111
##           1  30 231

```

De acuerdo a la matriz de confusión tenemos el mismo índice de acierto y de desacierto que teníamos inicialmente, con una pequeña variación entre falsos positivos y falsos negativos.

### Análisis de supervivencia

```

flex <- flexsurvreg(Surv(Age,Survived) ~ 1, data = titanic, dist = "exp") #Ajuste exponencial
flex

```

```

## Call:
## flexsurvreg(formula = Surv(Age, Survived) ~ 1, data = titanic,
##             dist = "exp")
##
## Estimates:
##      est      L95%      U95%      se
## rate 0.013062 0.011748 0.014522 0.000706
##
## N = 891, Events: 342, Censored: 549

```

```

## Total time at risk: 26183.67
## Log-likelihood = -1825.624, df = 1
## AIC = 3653.247
# Guardando el Objeto Surv
titanic.surv <- Surv(titanic$Age,titanic$Survived) #Creando objeto tipo Surv
titanic.km <- survfit(titanic.surv ~ 1, data = titanic, type = "kaplan-meier") #Estimación Kaplan Meier
summary(titanic.km)

## Call: survfit(formula = titanic.surv ~ 1, data = titanic, type = "kaplan-meier")
##
##      time n.risk n.event survival std.err lower 95% CI upper 95% CI
##      0.42   891      1   0.999 0.00112   0.997   1.000
##      0.67   890      1   0.998 0.00159   0.995   1.000
##      0.75   889      2   0.996 0.00224   0.991   1.000
##      0.83   887      2   0.993 0.00274   0.988   0.999
##      0.92   885      1   0.992 0.00296   0.986   0.998
##      1.00   884      5   0.987 0.00386   0.979   0.994
##      2.00   877      3   0.983 0.00431   0.975   0.992
##      3.00   867      5   0.977 0.00498   0.968   0.987
##      4.00   860      9   0.967 0.00598   0.956   0.979
##      5.00   848      4   0.963 0.00637   0.950   0.975
##      6.00   844      2   0.960 0.00656   0.948   0.973
##      7.00   841      1   0.959 0.00665   0.946   0.972
##      8.00   836      2   0.957 0.00683   0.944   0.970
##      9.00   831      4   0.952 0.00717   0.938   0.967
##     11.00   812      1   0.951 0.00726   0.937   0.966
##     12.00   808      1   0.950 0.00735   0.936   0.965
##     13.00   807      2   0.948 0.00751   0.933   0.963
##     14.00   805      3   0.944 0.00776   0.929   0.959
##     15.00   798      4   0.939 0.00807   0.924   0.955
##     16.00   793      6   0.932 0.00852   0.916   0.949
##     17.00   775      6   0.925 0.00895   0.908   0.943
##     18.00   762     11   0.912 0.00968   0.893   0.931
##     19.00   734     12   0.897 0.01043   0.877   0.918
##     20.00   701      5   0.890 0.01074   0.870   0.912
##     21.00   680      5   0.884 0.01106   0.862   0.906
##     22.00   640     12   0.867 0.01184   0.844   0.891
##     23.00   608      8   0.856 0.01235   0.832   0.880
##     24.00   587     19   0.828 0.01349   0.802   0.855
##     25.00   547      8   0.816 0.01396   0.789   0.844
##     26.00   514     15   0.792 0.01484   0.764   0.822
##     27.00   479     12   0.772 0.01554   0.743   0.803
##     28.00   458      7   0.761 0.01593   0.730   0.792
##     29.00   424     10   0.743 0.01653   0.711   0.776
##     30.00   402     12   0.720 0.01723   0.688   0.755
##     31.00   362      9   0.703 0.01781   0.669   0.738
##     32.00   344      9   0.684 0.01837   0.649   0.721
##     32.50   322      1   0.682 0.01843   0.647   0.719
##     33.00   320      8   0.665 0.01893   0.629   0.703

```

##	34.00	299	8	0.647	0.01944	0.610	0.686
##	35.00	276	13	0.617	0.02028	0.578	0.658
##	36.00	253	13	0.585	0.02106	0.545	0.628
##	37.00	225	1	0.582	0.02112	0.542	0.625
##	38.00	219	5	0.569	0.02146	0.529	0.613
##	39.00	207	8	0.547	0.02200	0.506	0.592
##	40.00	185	6	0.529	0.02244	0.487	0.575
##	41.00	168	2	0.523	0.02262	0.481	0.569
##	42.00	162	8	0.497	0.02327	0.454	0.545
##	43.00	146	1	0.494	0.02336	0.450	0.542
##	44.00	141	4	0.480	0.02372	0.436	0.529
##	45.00	131	5	0.462	0.02419	0.416	0.511
##	47.00	110	1	0.457	0.02433	0.412	0.508
##	48.00	97	7	0.424	0.02557	0.377	0.478
##	49.00	87	5	0.400	0.02633	0.352	0.455
##	50.00	80	5	0.375	0.02695	0.326	0.432
##	51.00	70	2	0.364	0.02723	0.315	0.422
##	52.00	63	3	0.347	0.02771	0.297	0.406
##	53.00	57	1	0.341	0.02788	0.290	0.400
##	54.00	56	3	0.323	0.02831	0.272	0.383
##	55.00	48	1	0.316	0.02851	0.265	0.377
##	56.00	45	2	0.302	0.02892	0.250	0.364
##	58.00	35	3	0.276	0.03005	0.223	0.342
##	60.00	28	2	0.256	0.03097	0.202	0.325
##	62.00	19	2	0.229	0.03306	0.173	0.304
##	63.00	15	2	0.199	0.03501	0.141	0.281
##	80.00	1	1	0.000	NaN	NA	NA

La estimación devuelve los siguientes valores:

- time : Tiempo de la observación
- n.risk : El número de sujetos en riesgo.
- n.evento : El número de sujetos que presentaron el evento.
- survival : La estimación de la función de supervivencia.
- std.err : La desviación estándar de la estimación.
- lower y upper CI\* : Los intervalos de confianza para la estimación.

La función `survfit()` devuelve un resumen de la estimación, la información se puede acceder agregando el símbolo “\$” seguido del nombre del elemento de la lista.

Una mejor manera de extraer la información es utilizando la función `fortify()` sobre el objeto `survfit`, esta función devuelve un `data.frame` con la información. Al tener presencia de covariables se anexa la columna llamada `strata`.

```
# Ahora veamos la tasa de supervivencia de las personas segun la clase con la
# que abordaron
table(titanic$Survived,titanic$Pclass)
```

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

Ciertamente, los pasajeros de primera clase tienen un índice de sobrevivencia claramente mas alto que las

otras dos clases. Los pasajeros de segunda clase tenian casi la misma proporcion de pasajeros vivos y muertos y la tercera clase esta claramente en desventaja a la hora de poder alcanzar un bote salvavidas.

### Desviación estándar e Intervalos de confianza de la estimación de la función de supervivencia

Tanto como la desviación estándar y los Intervalos de confianza de la curva de supervivencia es estimada mediante la función `survfit()` con los siguientes argumentos:

- `error`: Tipo de estimación para las desviaciones, los posibles valores son “greenwood”(defecto) para la fórmula de Greenwood o “tsiatis” para la fórmula de Tsiatis/Aalen.
- `conf.type`: Tipo de transformación para calcular los intervalos de confianza, “plain”, “log”(defecto), \* “log-log”.
- `conf.int`: El nivel de confianza para el intervalo de confianza (.95 por defecto).

```
titanic.km <- survfit(Surv(Age, Survived) ~ 1, data = titanic, type = "kaplan-meier",
  error = "tsiatis", conf.type = "log-log", conf.int = 0.99)
```

```
summary(titanic.km)
```

```
## Call: survfit(formula = Surv(Age, Survived) ~ 1, data = titanic, type = "kaplan-meier",
##      error = "tsiatis", conf.type = "log-log", conf.int = 0.99)
```

```
##
```

##	time	n.risk	n.event	survival	std.err	lower	99% CI	upper	99% CI
##	0.42	891	1	0.999	0.00112		0.985		1.000
##	0.67	890	1	0.998	0.00158		0.986		1.000
##	0.75	889	2	0.996	0.00224		0.984		0.999
##	0.83	887	2	0.993	0.00274		0.981		0.998
##	0.92	885	1	0.992	0.00296		0.979		0.997
##	1.00	884	5	0.987	0.00386		0.972		0.994
##	2.00	877	3	0.983	0.00430		0.968		0.991
##	3.00	867	5	0.977	0.00497		0.960		0.987
##	4.00	860	9	0.967	0.00596		0.948		0.980
##	5.00	848	4	0.963	0.00635		0.942		0.976
##	6.00	844	2	0.960	0.00654		0.940		0.974
##	7.00	841	1	0.959	0.00663		0.938		0.973
##	8.00	836	2	0.957	0.00681		0.935		0.971
##	9.00	831	4	0.952	0.00715		0.930		0.968
##	11.00	812	1	0.951	0.00724		0.929		0.967
##	12.00	808	1	0.950	0.00733		0.927		0.966
##	13.00	807	2	0.948	0.00749		0.924		0.964
##	14.00	805	3	0.944	0.00774		0.920		0.961
##	15.00	798	4	0.939	0.00805		0.915		0.957
##	16.00	793	6	0.932	0.00849		0.907		0.951
##	17.00	775	6	0.925	0.00892		0.898		0.945
##	18.00	762	11	0.912	0.00965		0.883		0.933
##	19.00	734	12	0.897	0.01039		0.867		0.921
##	20.00	701	5	0.890	0.01070		0.859		0.915
##	21.00	680	5	0.884	0.01101		0.852		0.909
##	22.00	640	12	0.867	0.01178		0.834		0.895
##	23.00	608	8	0.856	0.01229		0.821		0.885
##	24.00	587	19	0.828	0.01339		0.790		0.860
##	25.00	547	8	0.816	0.01385		0.777		0.849



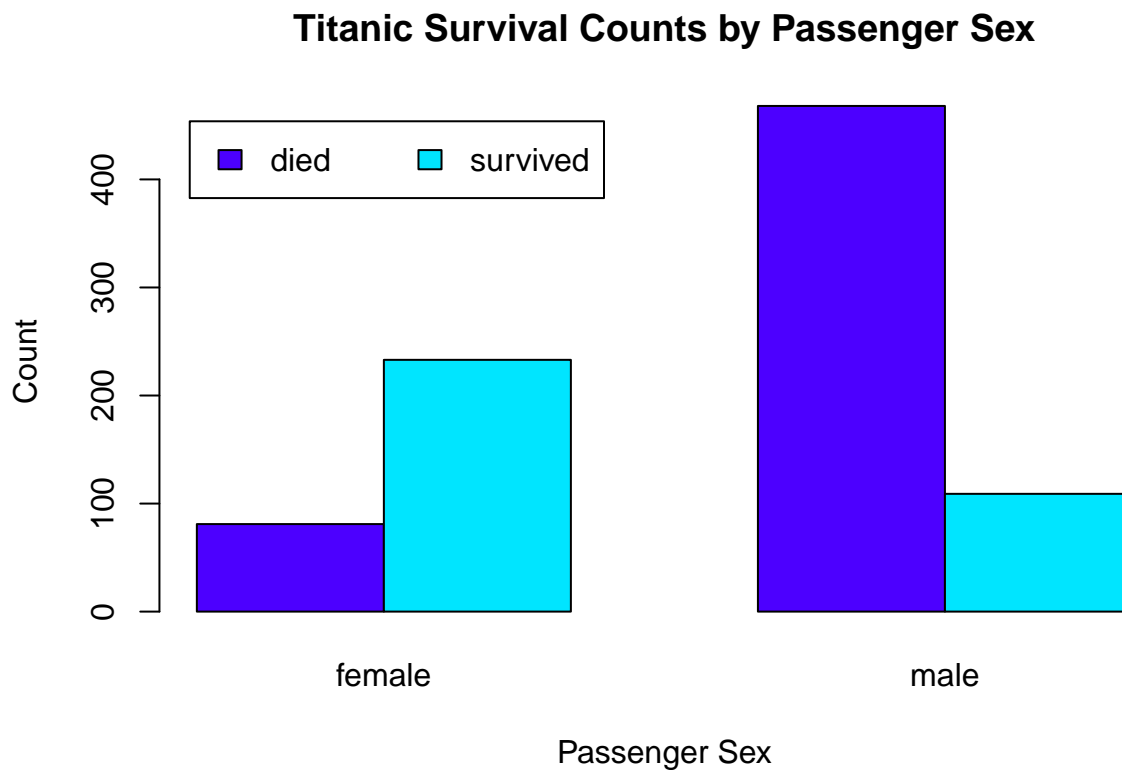
##	26.00	514	15	0.792 0.01471	0.751	0.827
##	27.00	479	12	0.772 0.01539	0.730	0.809
##	28.00	458	7	0.761 0.01578	0.717	0.798
##	29.00	424	10	0.743 0.01637	0.698	0.782
##	30.00	402	12	0.720 0.01706	0.674	0.762
##	31.00	362	9	0.703 0.01762	0.655	0.745
##	32.00	344	9	0.684 0.01817	0.635	0.728
##	32.50	322	1	0.682 0.01823	0.633	0.726
##	33.00	320	8	0.665 0.01873	0.614	0.711
##	34.00	299	8	0.647 0.01923	0.595	0.694
##	35.00	276	13	0.617 0.02001	0.563	0.666
##	36.00	253	13	0.585 0.02074	0.530	0.636
##	37.00	225	1	0.582 0.02080	0.527	0.634
##	38.00	219	5	0.569 0.02114	0.513	0.622
##	39.00	207	8	0.547 0.02166	0.490	0.601
##	40.00	185	6	0.529 0.02210	0.471	0.584
##	41.00	168	2	0.523 0.02227	0.464	0.579
##	42.00	162	8	0.497 0.02288	0.437	0.555
##	43.00	146	1	0.494 0.02298	0.433	0.551
##	44.00	141	4	0.480 0.02334	0.419	0.538
##	45.00	131	5	0.462 0.02379	0.399	0.521
##	47.00	110	1	0.457 0.02394	0.395	0.518
##	48.00	97	7	0.424 0.02505	0.359	0.488
##	49.00	87	5	0.400 0.02575	0.333	0.465
##	50.00	80	5	0.375 0.02632	0.307	0.442
##	51.00	70	2	0.364 0.02660	0.296	0.432
##	52.00	63	3	0.347 0.02707	0.278	0.417
##	53.00	57	1	0.341 0.02726	0.272	0.411
##	54.00	56	3	0.323 0.02766	0.253	0.394
##	55.00	48	1	0.316 0.02787	0.246	0.388
##	56.00	45	2	0.302 0.02827	0.231	0.376
##	58.00	35	3	0.276 0.02923	0.204	0.353
##	60.00	28	2	0.256 0.03007	0.183	0.336
##	62.00	19	2	0.229 0.03186	0.153	0.315
##	63.00	15	2	0.199 0.03337	0.121	0.290
##	80.00	1	1	0.000 0.00000	NA	NA

## 5. Visualización

Representación de los resultados a partir de tablas y gráficas

```
counts = table(titanic$Survived, titanic$Sex)
barplot(counts,
        main = "Titanic Survival Counts by Passenger Sex",
        xlab = "Passenger Sex",
        ylab = "Count",
        col = topo.colors(2),
        beside = TRUE)

legend("topleft",
      inset = .03,
      legend = c("died", "survived"),
      fill = topo.colors(2),
      horiz = TRUE)
```

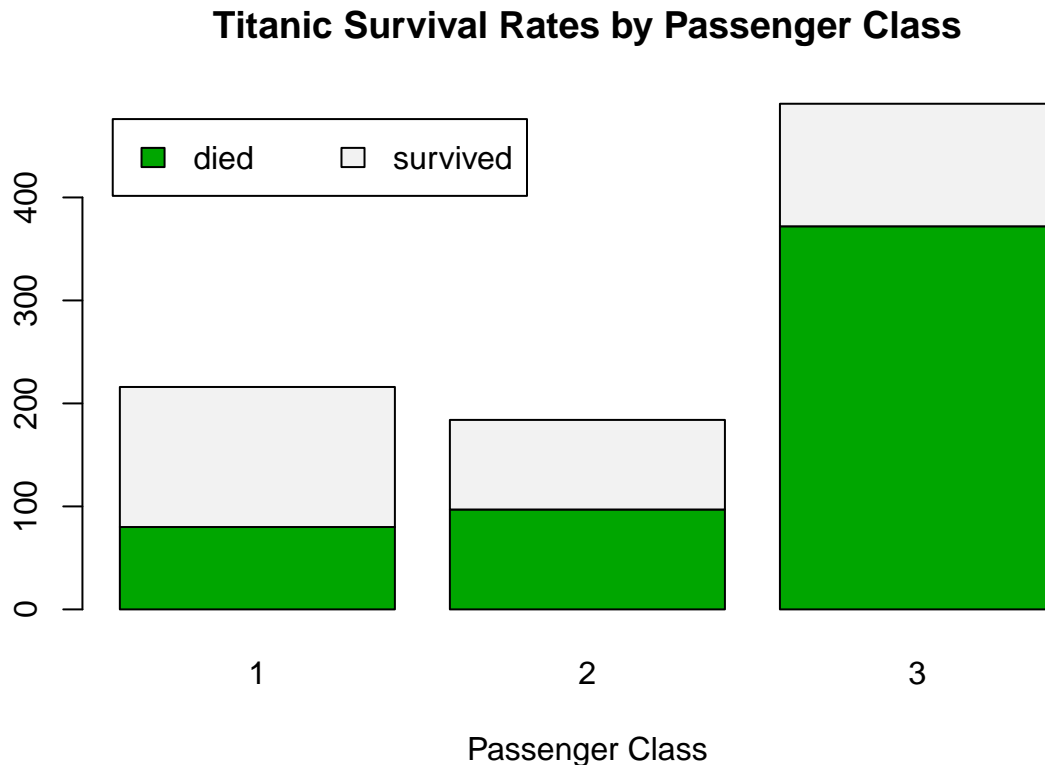


```
# STACKED BAR CHART WITH COLORS AND LEGEND
counts = table(titanic$Survived, titanic$Pclass)

# CONSTRUCT BARCHART
barplot(counts,
        main = "Titanic Survival Rates by Passenger Class",
        xlab = "Passenger Class",
```

```
col = terrain.colors(2))

# ADD LEGEND
legend("topleft",
      inset = .03,
      legend = c("died", "survived"),
      fill = terrain.colors(2),
      horiz = TRUE)
```



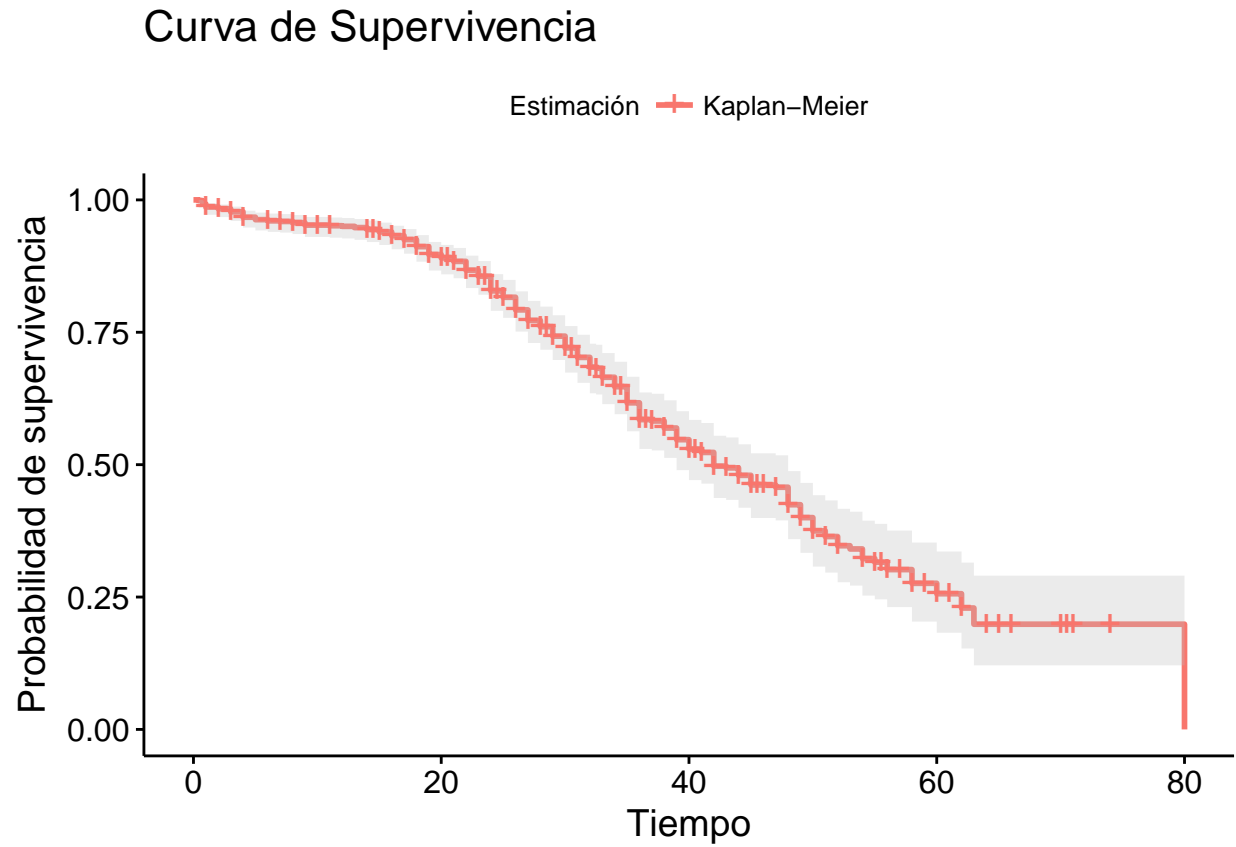
### Graficación de la curva de supervivencia

La curva de supervivencia estimada se gráfica con la función `ggsurvplot()` de la paquetería `survminer`, está gráfica esta hecha utilizando la librería `ggplot2` y contiene un numero grande de parámetros, por lo que solamente ilustraremos los más importantes y se recomienda revisar los demás utilizando el comando `help("ggsurvplot")`.

- `fit`: Objeto tipo `survfit`.
- `data`: Un conjunto de datos utilizado para ajustar curvas de supervivencia.
- `fun`: Transformación de la curva de supervivencia(Opcional), las posibles opciones son: “event” para los eventos acumulados, “cumhaz” para el riesgo acumulado y “pct” para la curva de supervivencia en porcentaje.
- `conf.int`: Indicador para graficar los intervalos de confianza.
- `title`: Titulo
- `xlab`: Eje x
- `ylab`: Eje Y
- `legends.lab`: Vector de nombres para identificar las curvas.

- legend.title : Título de la leyenda.

```
ggsurvplot(fit = titanic.km, data = titanic, conf.int = T, title = "Curva de Supervivencia",
  xlab = "Tiempo", ylab = "Probabilidad de supervivencia", legend.title = "Estimación",
  legend.labs = "Kaplan-Meier")
```



## 6. Resolución del problema.

- Con el análisis de correlación pude conocer cuáles de las variables iniciales variables ejercen una mayor influencia para analizar/predecir la supervivencia de los pasajeros del titanic.
- Se ha hecho un preprocesamiento para manejar los casos de ceros o elementos vacíos y valores extremos (outliers). Hice una imputación de datos faltantes en las variables Age y Fare y así conservamos la totalidad de los datos y no eliminamos registros del conjunto de datos inicial.
- Los datos outliers los he conservado porque he comprobado que la variable Fare pertenece a personas que pagaron un ticket de primera clase.
- Podríamos concluir que en efecto, mas pasajeros mujeres que hombres sobrevivieron, por lo que podemos establecer una prioridad a la hora del abordaje de los botes salvavidas. Con los niños no fue tanto el caso. La proporción esta mas equilibrada casi al punto de que la mitad de los niños y las niñas sobrevivieron, con una leve inclinacion a la proporción en los niños.
- Con los árboles de decisión y usando los pasajeros el Titánic, obtendremos que aquellos individuos que murieron y que no murieron presentaron diferencias, y la regresión logística lo que hará será detectar estas diferencias entre el desenlace de muerte/sobrevivida y entre ellas mismas. Al final la máquina nos dirá cuáles fueron las más poderosas, y el investigador podrá decir y darle un número a cada una de ellas para crear un algoritmo o regla con puntos o valores para cada variable. Por ejemplo, darle 2 puntos a ser de primera clase, 1 punto al sexo, 1 a la edad, etc.
- Como pueden notar, no existe UNO solo que sea un factor infalible a la hora de predecir sobrevivida. Habrá algunos más fuertes que otros; por ejemplo, ser de primera clase es el factor más fuerte y predictor de sobrevivida, aunque no es garantía (ya que murieron 4 personas, incluyendo una niña de 4 años de primera clase). Pero si eres menor de 10 años, niña, y en primera clase, tus probabilidades se irán al casi 100% de supervivencia y de que llegues a Nueva York sano y salvo.

## 7. Código

El código de la práctica esta implicito en cada uno de los chunk del reporte.

## Referencias

Los siguientes recursos son de utilidad para la realización de la práctica:

- Megan Squire (2015). Clean Data. Packt Publishing Ltd.
- Jiawei Han, Micheline 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
- Wes McKinney (2012). Python for Data Analysis. O'Reilley Media, Inc
- Tutorial de Github "<https://guides.github.com/activities/hello-world>"
- Ejemplos de contrastes de hipótesis con R: "[https://rstudio-pubs-static.s3.amazonaws.com/65042\\_a1784120e81a430f9de400ed9b899b0b.html](https://rstudio-pubs-static.s3.amazonaws.com/65042_a1784120e81a430f9de400ed9b899b0b.html)"
- Tutorial dplyr: "<https://github.com/fdelaunay/tutorial-dplyr-es/blob/master/R/tutorial-dplyr.md>"
- Test de Shapiro-Wilk: "<https://rpro.wikispaces.com/Test+de+Shapiro-Wilk>"
- Estadística descriptiva: "Introducción al análisis de datos", Àngel J. Gil Estallo
- Intervalos de confianza, Àngel J. Gil Estallo
- Contrastes de hipótesis, Carles Rovira Escofet
- Contraste de dos muestras, Josep Gibergans Bàguena