M2.851 - Tipología y ciclo de vida de los datos Practica 2

Francisco Javier Melchor y Enrique Otero

04/01/2021

Contents

Paquetes

Los paquetes que se van a utilizar para el desarrollo de esta actividad, son los siguientes:

```
if(!require(ggplot2)){
  install.packages("ggplot2")
  library(ggplot2)
}
if(!require(arc)){
  install.packages("arc")
  library(arc)
}

if(!require(ggcorrplot)){
  install.packages("ggcorrplot")
  library(ggcorrplot)
}
```

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:

A. 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.

B.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:

1. **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.

2. **Saber** identificar los datos relevantes y los tratamientos necesarios (integración, limpieza y validación) para llevar a cabo un proyecto analítico.

3. **Aprender** a analizar los datos adecuadamente para abordar la información contenida en los datos.

4. **Identificar** la mejor representación de los resultados para aportar conclusiones sobre el problema planteado en el proceso analítico.

5. **Actuar** con los principios éticos y legales relacionados con la manipulación de datos en Tipología y ciclo de vida de los datos Práctica 2 pág 2 función del ámbito de aplicación.

6. **Desarrollar** las habilidades de aprendizaje que les permitan continuar estudiando de un modo que tendrá que ser en gran medida autodirigido o autónomo.

7. **Desarrollar** la capacidad de búsqueda, gestión y uso de información y recursos en el ámbito de la ciencia de datos.

Descripción de la Práctica a realizar

El objetivo de esta actividad será el tratamiento de un dataset, que puede ser el creado en la práctica 1 o bien cualquier dataset libre disponible en Kaggle (https://www.kaggle.com). Algunos ejemplos de dataset con los que podéis trabajar son:

Red Wine Quality (https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009)
Titanic: Machine Learning from Disaster (https://www.kaggle.com/c/titanic)

El último ejemplo corresponde a una competición activa de Kaggle de manera que, opcionalmente, podéis aprovechar el trabajo realizado durante la práctica para entrar en esta competición.

Preguntas y desarrollo de respuestas

Siguiendo las principales etapas de un proyecto analítico, las diferentes tareas a realizar (y justificar) son las siguientes:

1. Descripción del dataset. ¿Por qué es importante y qué pregunta/problema pretende responder?

Para nuestra practica especifica hemos elegido el dataset asociado al ejemplo de Kaggle:
Titanic: Machine Learnin from Disaster

El hundimiento del Titanic es uno de los naufragios más trágicos de la historia.

El 15 de abril de 1912, durante su viaje inaugural, el RMS Titanic, ampliamente considerado "insumergible", se hundió tras chocar con un iceberg. Desafortunadamente, no había suficientes botes salvavidas para todos a bordo, lo que resultó en la muerte de 1502 de los 2224 pasajeros y la tripulación.

Si bien hubo algún elemento de suerte involucrado en sobrevivir, parece que algunos grupos de personas tenían más probabilidades de sobrevivir que otros.

En este desafío, se pide crear un modelo predictivo que responda a la pregunta: **"¿Qué tipo de personas tenían más probabilidades de sobrevivir?"** utilizando datos de pasajeros (es decir, nombre, edad, sexo, clase socioeconómica, etc.). En términos de análitica se trata de un problema de **clasificación**, esto es, usar esas variables independientes para predecir la categoría a la que pertenece cada registro, o, dicho de otra manera, predecir si un pasajero dado va a sobrevivir o no.

El enlace de descarga de este ejemplo contiene tres ficheros:

train.csv. Se trata del dataset *test* sobre el que entrenamos a nuestros modelos de analítica.
test.csv. Es el dataset donde probamos, con nuevos datos, nuestros modelos de analítica. Para este caso no se incluye el resultado (variable dependiente *Survived*) ya que es el objetivo del concurso.
gender_submission.csv. Contiene un ejemplo de como debe presentarse el formato de salida con el resultado de nuestros modelos. Se trata de un conjunto de predicciones que asumen que todas y solo mujeres sobreviven.

Según los datos proporcionados en la web de Kaggle, las variables de los datasets son:

Variable	Definition	Key
Survived	If passenger survived	0 = No, 1 = Yes
Pclass	Ticket Class	1 = 1st, 2 = 2nd, 3 = 3rd
Sex	Passenger Sex	
Age	Passenger Age in years	
SibSp	Number of sibings/spouses of the passenger aboard the Titanic	
Parch	Number of parents/children of the passenger aboard the Titanic	
Ticket	Ticket number	
Fare	Passenger fare	
Cabin	Cabin Number	
Embarked	Port of Embarkation	

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

En este caso al estar realizando el análisis de un único dataset, no es necesario realizar ninguna integración de distintas fuentes, pues sólo existe una.

Por otro lado, con respecto a la selección, en este caso al no ser un dataset excesivamente grande y al no tener fijado un objetivo diferente que analizar todo el conjunto de sus datos y no una parte de ellos, no se

realizará ninguna selección del dataset de origen ni se acotará el mismo.

3. Limpieza de datos

En esta sección realizaremos una limpieza del dataset incluido en el fichero **train.csv**.

Para ello, lo primero que realizaremos es la lectura del fichero **train.csv** y comprobar como han sido interpretadas por R las variables que forman el mismo.

```
ttc <- read.csv("./Data/train.csv",na.strings=c("","  ","NA"))
head(ttc)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                 Name    Sex Age SibSp Parch
## 1                            Braund, Mr. Owen Harris   male  22     1     0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
## 3                             Heikkinen, Miss. Laina female  26     0     0
## 4       Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
## 5                           Allen, Mr. William Henry   male  35     0     0
## 6                                   Moran, Mr. James   male  NA     0     0
##             Ticket    Fare Cabin Embarked
## 1        A/5 21171  7.2500  <NA>        S
## 2         PC 17599 71.2833   C85        C
## 3 STON/O2. 3101282  7.9250  <NA>        S
## 4           113803 53.1000  C123        S
## 5           373450  8.0500  <NA>        S
## 6           330877  8.4583  <NA>        Q
```

```
str(ttc)
```

```
## 'data.frame':    891 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
##  $ Sex        : chr  "male" "female" "female" "female" ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  NA "C85" NA "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
```

```
# Contamos con 891 observaciones de las 12 variables decritas al inicio de esta seccion.
```

Como se puede observar, la mayoría de las variables han sido interpretadas correctamente por R, pero tanto la variable **Sex** como la variable **Embarked**, han sido formateadas como variables de tipo carácter (chr) y realmente son variables categóricas. Por otro lado, la variable **Survived** y **Pclass** han sido interpretadas como variables numéricas cuando realmente son variables categóricas. Procedemos a continuación a convertir las variables nombradas.

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

str(ttc)
```

```
## 'data.frame':    891 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 Bradley (Florence Briggs Thayer)"
##  $ 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  NA "C85" NA "C123" ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 3 1 3 3 3 2 3 3 3 1 ...
```

Una vez que todas las variables han sido interpretadas correctamente, podemos proceder a realizar la limpieza y el procesamiento de los datos que contiene este dataframe.

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

A continuación se procede a comprobar si existen valores nulos o elementos vacíos en el dataframe a analizar:

```
colSums(is.na(ttc))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0         177
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           0         687           2
```

Como se puede observar, existen valores faltantes en las columnas **Age, Cabin y Embarked**.

Para responder a la pregunta de *¿Cómo gestionarías cada uno de estos casos?*, primero hay que analizar el número total de filas del dataset que se está analizando:

```
nrow = nrow(ttc)
```

El número total de filas con las que cuenta el dataset son: 891

Teniendo en cuenta la dimensión del dataframe y el número de valores faltantes, procedemos a realizar las siguientes consideraciones:

- En el caso de la variable **Age**, al contar con una proporción de 19% de valores faltantes, al ser una proporción baja, se realizará una imputación de dichos valores. Dicha imputación será a través del estimador de la mediana, para evitar sesgos causadps por valores atípicos, y dicha imputación se dividirá por clases, es decir, se calcularán la mediana de edad resultante en cada una de las clases y dependiendo de si el valor faltante pertenece a una clase u otra se le imputará la mediana resultante de la edad en dicha clase.
- En el caso de la variable **Cabin**, al tratarse de más de un 70% de valores nulos o no válidos, dicha variable será eliminada del conjunto de datos a tratar, ya que no tenemos suficientte información en la que basarnos (un 30% de los casos) para realizar una imputación.

- Por último, para la variable **Embarked**, al tratarse únicamente de 2 casos con respecto al total que son ‘r nrow, se eliminarán aquellas filas con dicho valor a nulo, pues al ser una cantidad tan pequeña, no merece la pena realizar una imputación.

A continuación procedemos a realizar los cambios comentados:

Primero comenzaremos con la eliminación de la variable **Cabin**

```
ttc$Cabin <- NULL
head(ttc)
```

```
##   PassengerId Survived Pclass
## 1           1        0      3
## 2           2        1      1
## 3           3        1      3
## 4           4        1      1
## 5           5        0      3
## 6           6        0      3
##                                                  Name    Sex Age SibSp Parch
## 1                             Braund, Mr. Owen Harris   male  22     1     0
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
## 3                              Heikkinen, Miss. Laina female  26     0     0
## 4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
## 5                            Allen, Mr. William Henry   male  35     0     0
## 6                                    Moran, Mr. James   male  NA     0     0
##             Ticket    Fare Embarked
## 1        A/5 21171  7.2500        S
## 2         PC 17599 71.2833        C
## 3 STON/O2. 3101282  7.9250        S
## 4           113803 53.1000        S
## 5           373450  8.0500        S
## 6           330877  8.4583        Q
```

Ahora procederemos a eliminar aquellas filas donde la variable **Embarked** tiene un valor no válido:

```
ttc <- ttc[!is.na(ttc$Embarked),]
colSums(is.na(ttc))
```

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

Por último, procedemos a realizar la imputación de la variable **Age**:

```
imputationFunct <- function(x){
    if (is.na(x["Age"])){
      x["Age"]<- median(ttc$Age[ttc$Pclass==x["Pclass"] & !is.na (ttc$Age)])
    } else{
      x<-x
    }
  return (x["Age"])
}

ttc$Age <- apply(ttc,1,imputationFunct)
ttc$Age <- as.numeric(ttc$Age)
sapply(ttc,function(x) sum(is.na(x)))
```

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

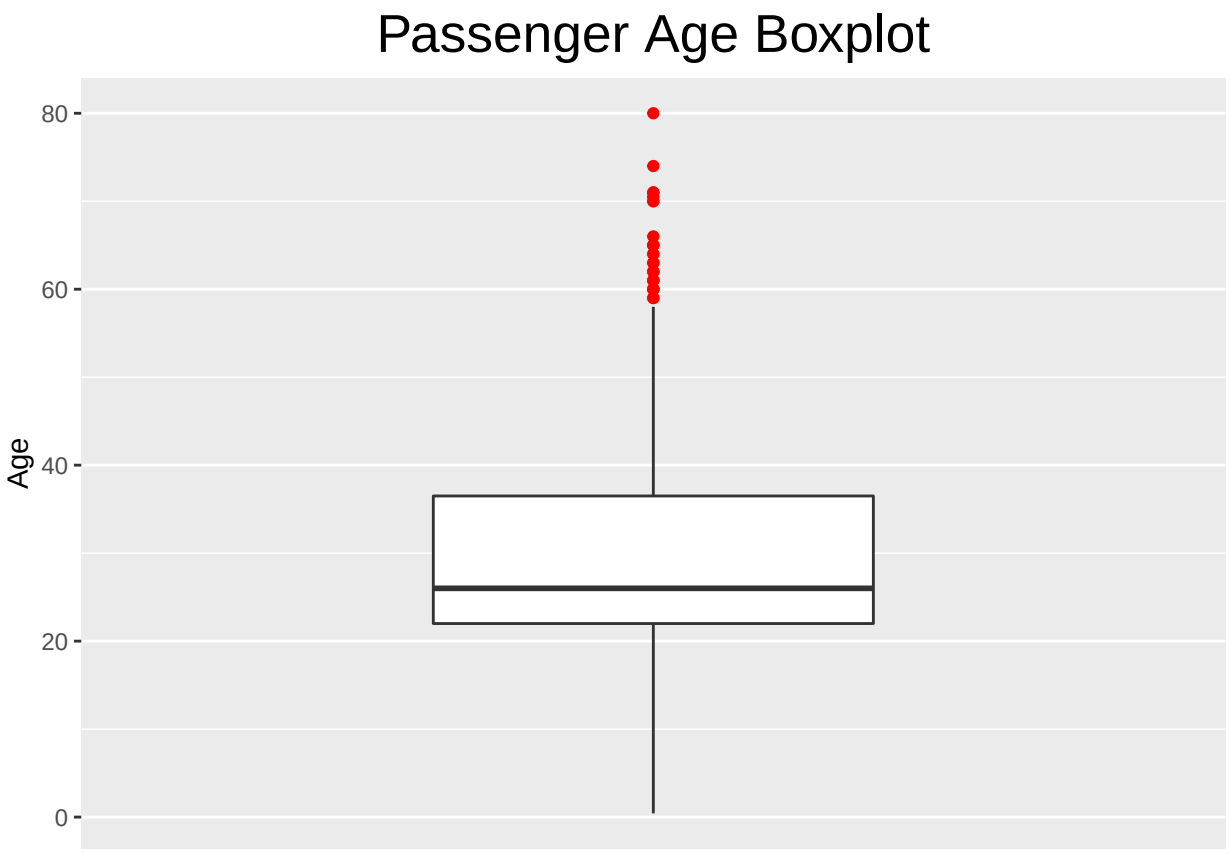
3.2 Identificación y tratamiento de valores extremos

Una vez analizados y resueltos los valores faltantes del dataset a analizar, se procede a comprobar si existen valores atípicos en el mismo. Para ello, primero representaremos las variables numéricas con un diagrama de cajas y bigotes, lo cual nos permitirá visualizar gráficamente a simple vista si existen valores atípicos.

```
# Boxplot para la variable Age
summary(ttc$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.42   22.00   26.00   29.02   36.50   80.00
```

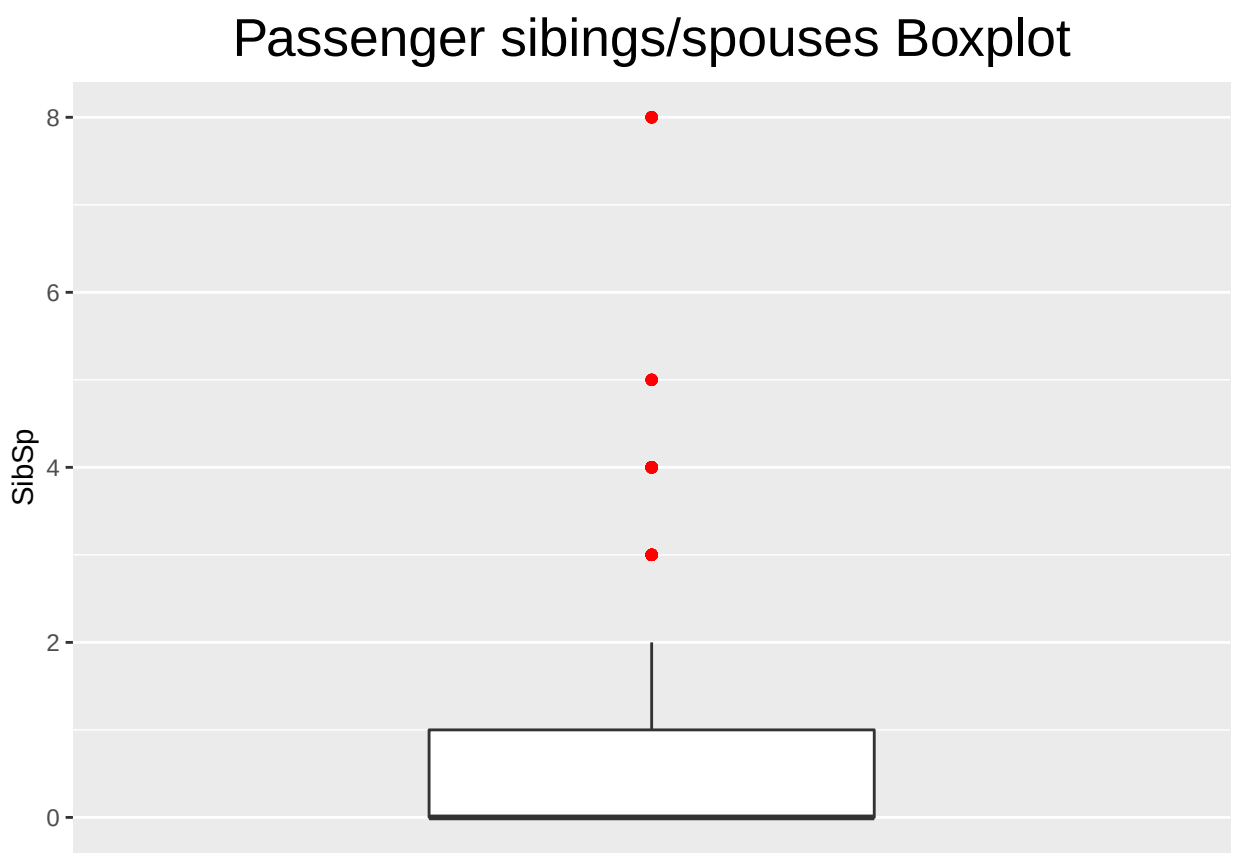
```
ggplot(data = ttc, aes(y = Age)) +
  geom_boxplot(outlier.colour = "red") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  ggtitle("Passenger Age Boxplot") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```




```
# Boxplot para la variable SibSp
summary(ttc$SibSp)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.5242  1.0000  8.0000
```

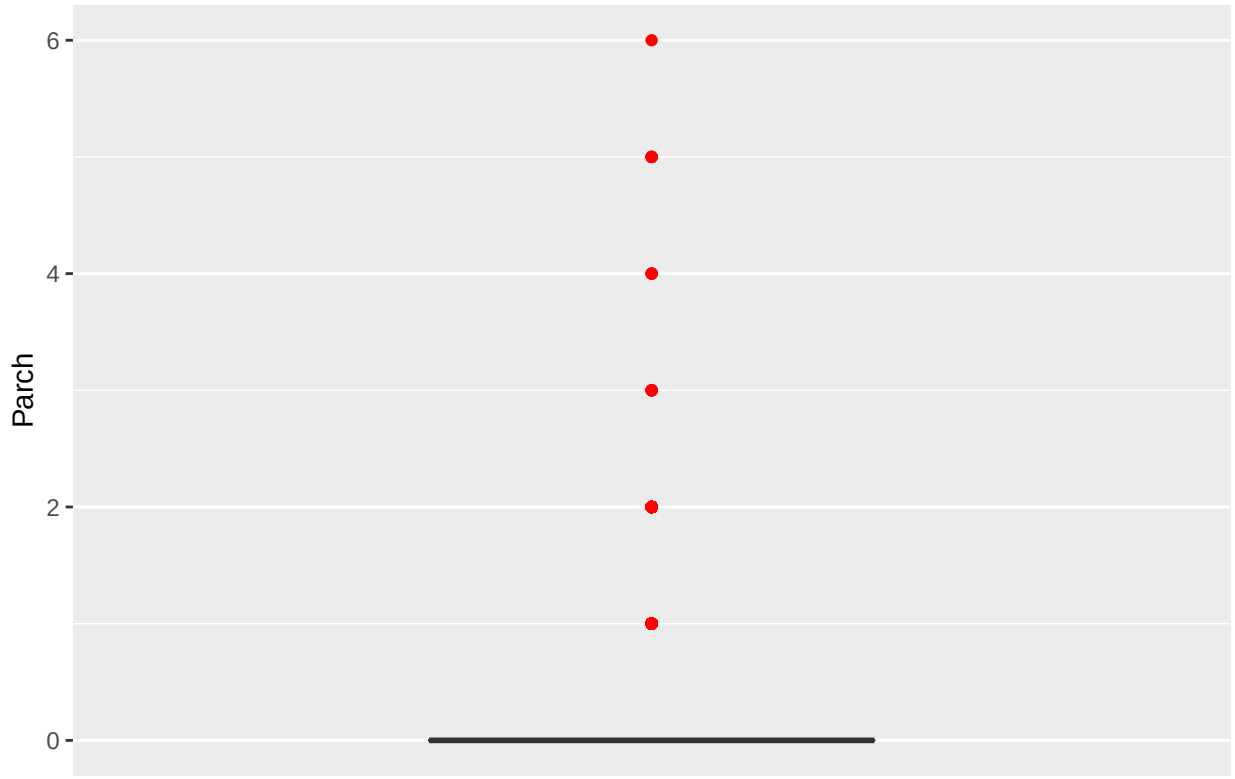
```r
ggplot(data = ttc, aes(y = SibSp)) +
  geom_boxplot(outlier.colour = "red") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  ggtitle("Passenger sibings/spouses Boxplot") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```




```r
# Boxplot para la variable Parch
summary(ttc$Parch)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.3825  0.0000  6.0000
```

```r
ggplot(data = ttc, aes(y = Parch)) +
  geom_boxplot(outlier.colour = "red") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  ggtitle("Passenger parents/children Boxplot") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```




```r
# Boxplot para la variable Fare
summary(ttc$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   7.896  14.454  32.097  31.000 512.329
```

```r
ggplot(data = ttc, aes(y = Fare)) +
  geom_boxplot(outlier.colour = "red") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  ggtitle("Passenger paid Fare Boxplot") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```

Una vez representadas las diferentes variables numéricas, procedemos a extraer las conclusiones oportunas de cada una de ellas. Cuando existen valores atípicos en los datos, pueden darse debido a varias opciones, puede ser porque son valores tomados con una unidad diferente que haga que algunos casos difieran de manera atípica de otros, puede que esos valores representen un valor faltante o nulo o puede que dichos valores formen parte de la muestra y por lo tanto sean valores reales y que hay que tener en cuenta a la hora de realizar los diferentes análisis. También puede darse el caso que sean valores que por el contexto se denote que no han sido tomados correctamente (como por ejemplo una edad de 150 años).

En este caso, nos encontramos con que tenemos valores atípicos en todas las variables numéricas, aunque sí que es cierto que en algunas de ellas dichos valores se encuentran más aislados, como es el caso de la variable **Fare**, que indica la tarifa del pasajero/a.

- En el caso de la variable **Age**, vemos que la mediana se encuentra apróximadamente entre los 25 años, pero que existen valores atípicos a partir de 60 y que hay casos de pasajeros hasta con 80 años. Sí que es verdad que se trata de un valor atípico con respecto a la mayoría de personas que se encuentran en el barco, pero no es un valor imposible de encontrar, por lo que se considera que forma parte del conjunto de datos y que se tiene que tener en cuenta a la hora de realizar el análisis.
- En el caso de la variable **Sibing/spouses** el valor más destacado es el caso de 8. Sí que es verdad que se trata de un valor poco casual, pero puede darse el caso de que una persona tenga 7 hermanos/as y una esposa, o múltiples combinaciones, es decir, a simple vista, no parece ser un valor irreal, por lo que se considera que también se debe tener en cuenta para el análisis.
- En el caso de la variable **Parents/children** pasa un poco como con la variable anterior, el valor máximo es 6 pero no se trata de un valor imposible o improbable, y más por la época en la que se basan los datos, en la que tener 5 o 6 hijos era algo común, por lo que no se considera oportuno realizar ningún cambio en dichos valores.
- Por último, la variable **Fare** resulta ser la variable que contiene los valores atípicos que más se alejan de

la desviación estándar de la misma, pues ya de por sí un valor por encima de los 250 resulta ser bastante atípico (según los datos), con lo que en el caso de estar por encima de 500, sitúa dicho valor demasiado alejado de los demás. Dado el significado de la variable y el contexto, puede tratarse perfectamente de un valor real, ya que en los cruceros existen pasajes muy lujosos que tienen un precio muy por encima de un pasaje estándar. No obstante, si que es cierto, que aunque pueda tratarse de valores reales, al estar tan extremadamete alejado de la desviación estándar de la población, puede hacer que los diferentes análisis que se apliquen estén sesgados por dichos valores.

A continuación se procede a estudiar cuantos casos de la variable **Fare** se encuentran por encima de 500 y a realizar una comparación del resultado obtenido por una medida de dispersión robusta a la presencia de valores atípicos, la mediana, con una no robusta a ellos, la media.

```
outlier_cases = nrow(ttc[ttc$Fare > 500,])
mean_Fare = mean(ttc$Fare)
median_Fare = median(ttc$Fare)

outlier_cases
```

```
## [1] 3
```

```
mean_Fare
```

```
## [1] 32.09668
```

```
median_Fare
```

```
## [1] 14.4542
```

Como se puede observar, el número total de casos atípicos son 3 en todo el dataset.

Por otro lado, el valor obtenido por la media es de 32.0966809, mientras que por la mediana es de 14.4542. Si comparamos ambos resultado, podemos ver que el valor obtenido por la media es apróximadamente el doble que el obtenido por la mediana, lo que indica que los valores atípicos están sesgando dicha medida de dispersión, pero dicho sesgo no se debe a los 3 casos que se encuentran por encima de 500, si que es cierto que influirán, pero el grueso del sesgo se debe que existen muchos casos por encima de la mediana.

Al tratarse de tan pocos casos de los que se encuentran exageradamente desviados (3) y al parecer por el contexto que pueden tratarse de valores reales, se considera que deben ser utilizados para los distintos análisis o métodos estadísticos que se realicen, pero que se deberá tener en cuenta su presencia para aplicar análisis que sean robustos a la presencia de valores atípicos. Pues realizar una imputación de todos los valores que son realmente atípicos o eliminarlos, conllevaría una gran pérdida de información que no es necesaria.

4. Análisis de los datos.

A continuación, procederemos a realizar una visualización de las diferentes columnas o variables que forman el dataset, para ver como se distribuyen las mismas.

Comenzaremos por aquellas variables categóricas o cualitativas, estas son:

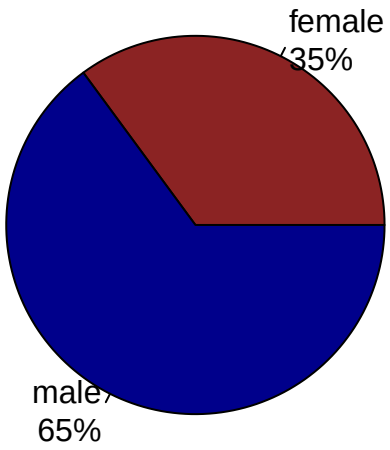
```
factors = unlist(lapply(ttc, is.factor))
which(factors, arr.ind = TRUE)
```

```
## Survived   Pclass      Sex Embarked
##        2        3        5       11
```

Procedemos a continuación a representar cada una de ellas:

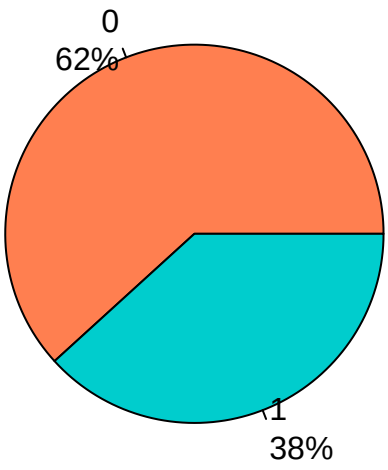
```
mytableSex <- table(ttc$Sex)
pctSex <- round(mytableSex/sum(mytableSex)*100)
lblsSex <- paste(names(mytableSex), "\n", pctSex, sep="")
lblsSex <- paste (lblsSex, '%', sep="")
```

```
pie(mytableSex, labels = lblsSex,
    main="Distribución de la variable Sex\n", col=c("brown4","darkblue"))
```

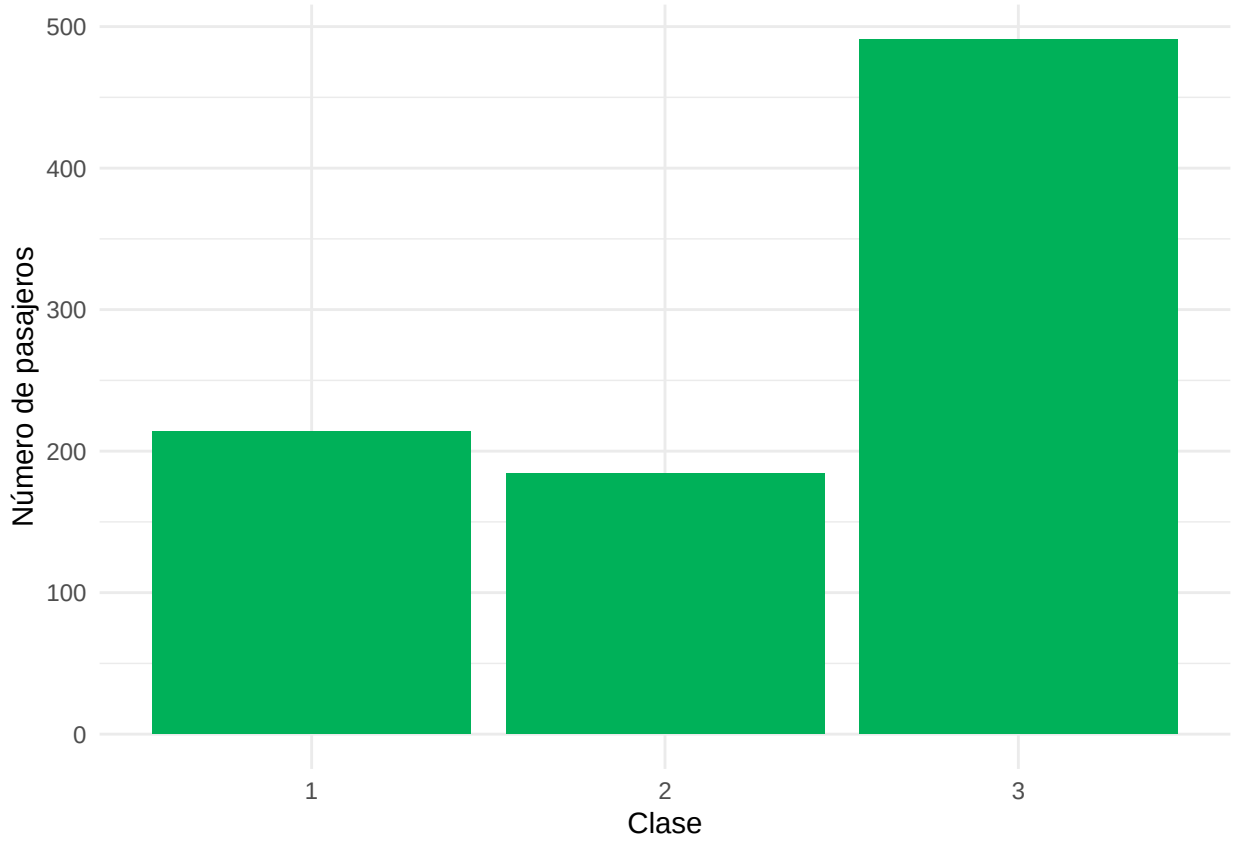



```
mytableSurvived <- table(ttc$Survived)
pctSurvived <- round(mytableSurvived/sum(mytableSurvived)*100)
lblsSurvived<- paste(names(mytableSurvived), "\n", pctSurvived, sep="")
lblsSurvived <- paste (lblsSurvived, '%', sep="")
pie(mytableSurvived, labels = lblsSurvived,
    main="Distribución de la variable Survived\n", col = c("coral","cyan3"))
```

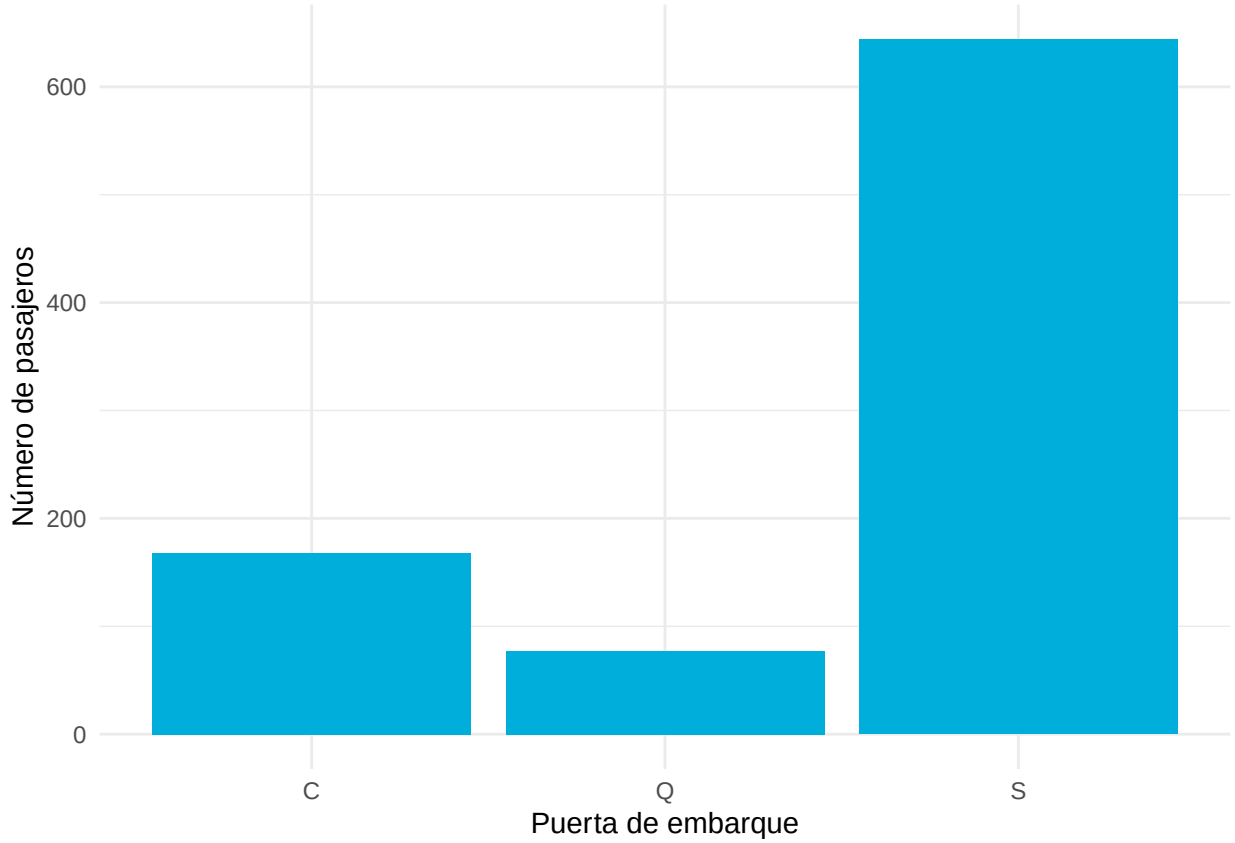
Distribución de la variable Survived




```
tablePclass<-table(ttc$Pclass)
dfPclass<-data.frame(tablePclass)

p<-ggplot(data=dfPclass, aes(x=Var1, y=Freq)) +
  geom_bar(stat="identity", fill="#00b159")+
  theme_minimal()+
  xlab("Clase")+
  ylab("Número de pasajeros")
p
```




```
tableEmb<-table(ttc$Embarked)
dfEmb<-data.frame(tableEmb)

p<-ggplot(data=dfEmb, aes(x=Var1, y=Freq)) +
  geom_bar(stat="identity", fill="#00aedb")+
  theme_minimal()+
  xlab("Puerta de embarque")+
  ylab("Número de pasajeros")
p
```



De las gráficas anteriores podemos obtener las siguientes conclusiones:

- Hay una mayor proporción de hombres que de mujeres a bordo.
- La mayoría de los personas que iban a bordo no sobrevivieron.
- La mayoría de las personas viajaron en tercera clase, y el número de personas que viajaban en segunda y en primera clase era muy similar.
- La gran mayoría de pasajeros entraron por la puerta de embarque "S"

Una vez representadas las variables categóricas, procedemos a representar las variables continuas del dataset, las cuales son:

```
numerics = unlist(lapply(ttc, is.numeric))
which(numerics, arr.ind = TRUE)
```

```
## PassengerId         Age       SibSp       Parch        Fare
##           1           6           7           8          10
```

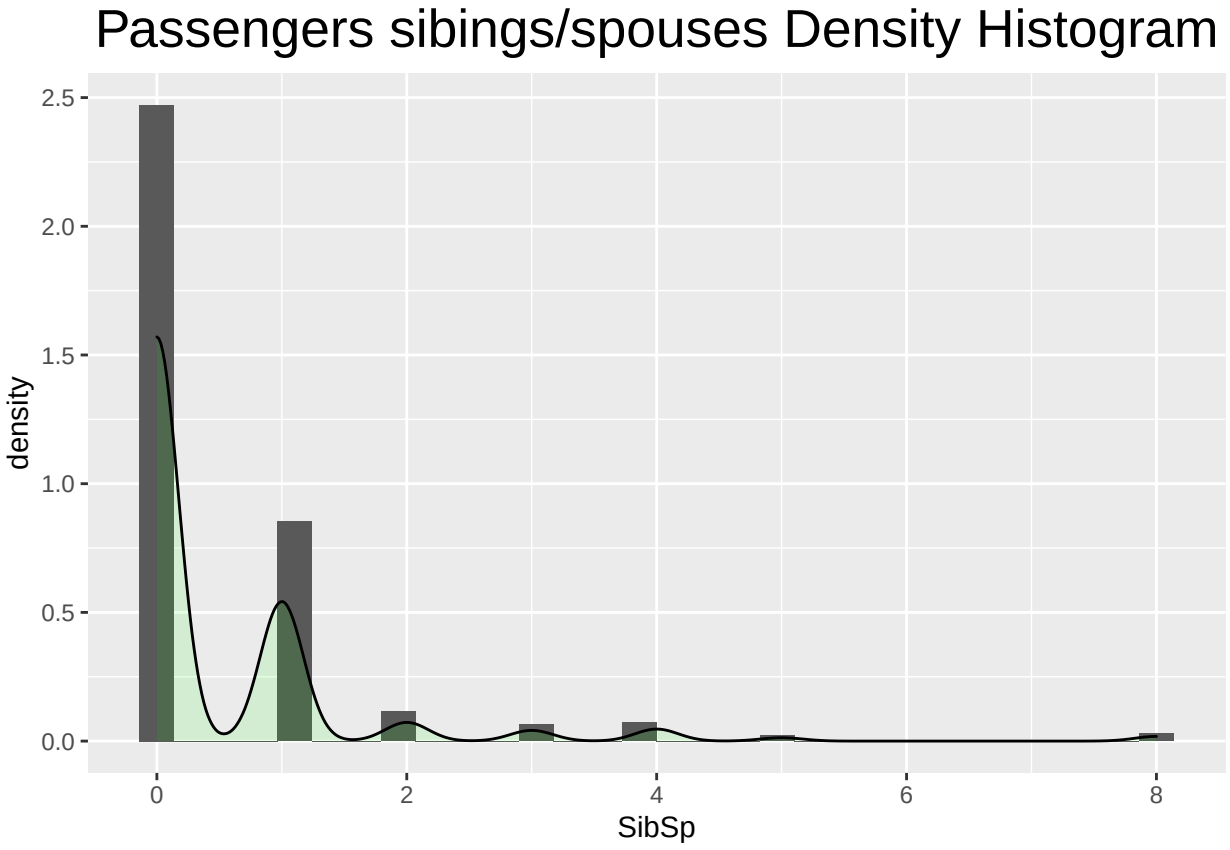
De todas las variables que han resultado ser numéricas, representaremos todas menos la variable **PassengerId** que indica únicamente el identificador de cada pasajero o pasajera.

```
# Histograma para la variable Age
ggplot(ttc, aes(x = Age)) +
  geom_histogram(aes(y = ..density..)) +
  geom_density(alpha = 0.1, fill = "blue") +
  ggtitle("Passengers Age Density Histogram") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```

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

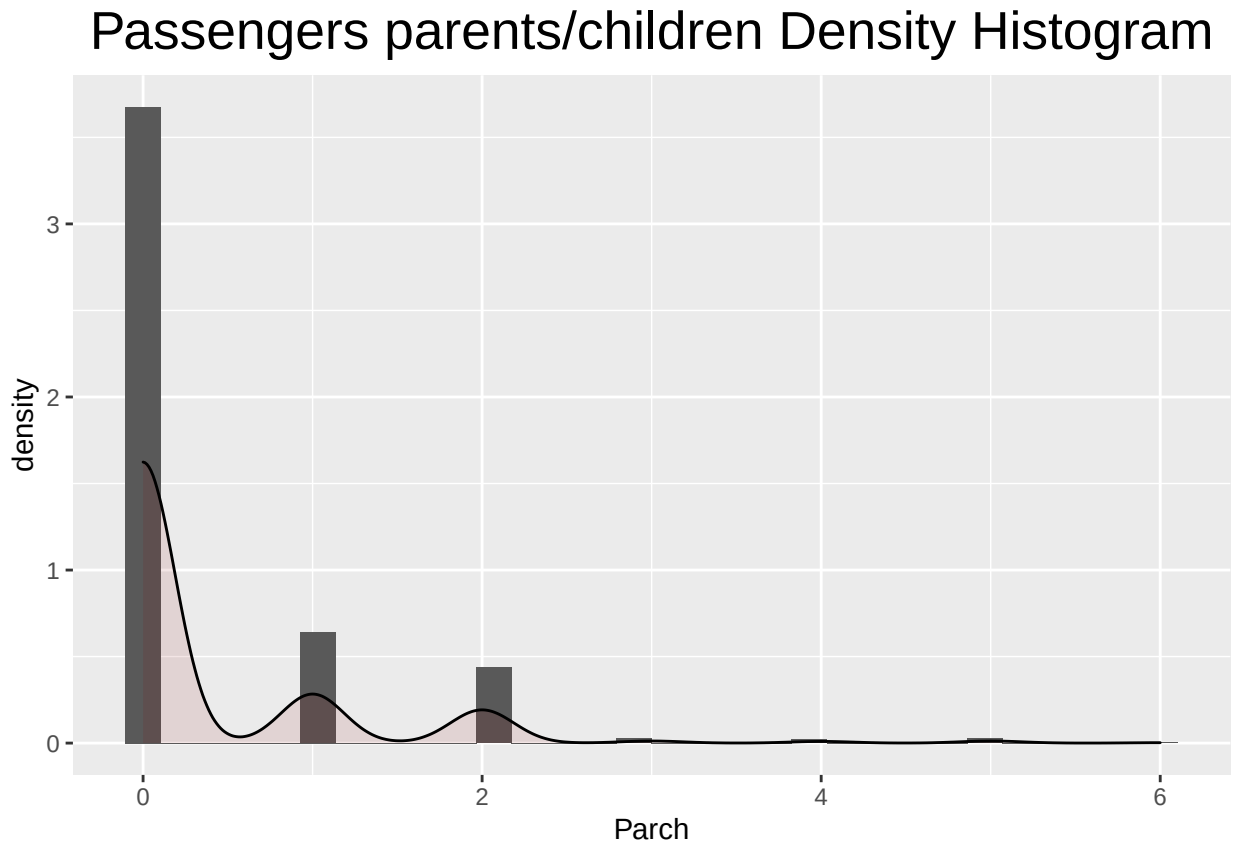

```
# Histograma para la variable SibSp
ggplot(ttc, aes(x = SibSp)) +
  geom_histogram(aes(y = ..density..)) +
  geom_density(alpha = 0.1, fill = "green") +
  ggtitle("Passengers sibings/spouses Density Histogram") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```

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



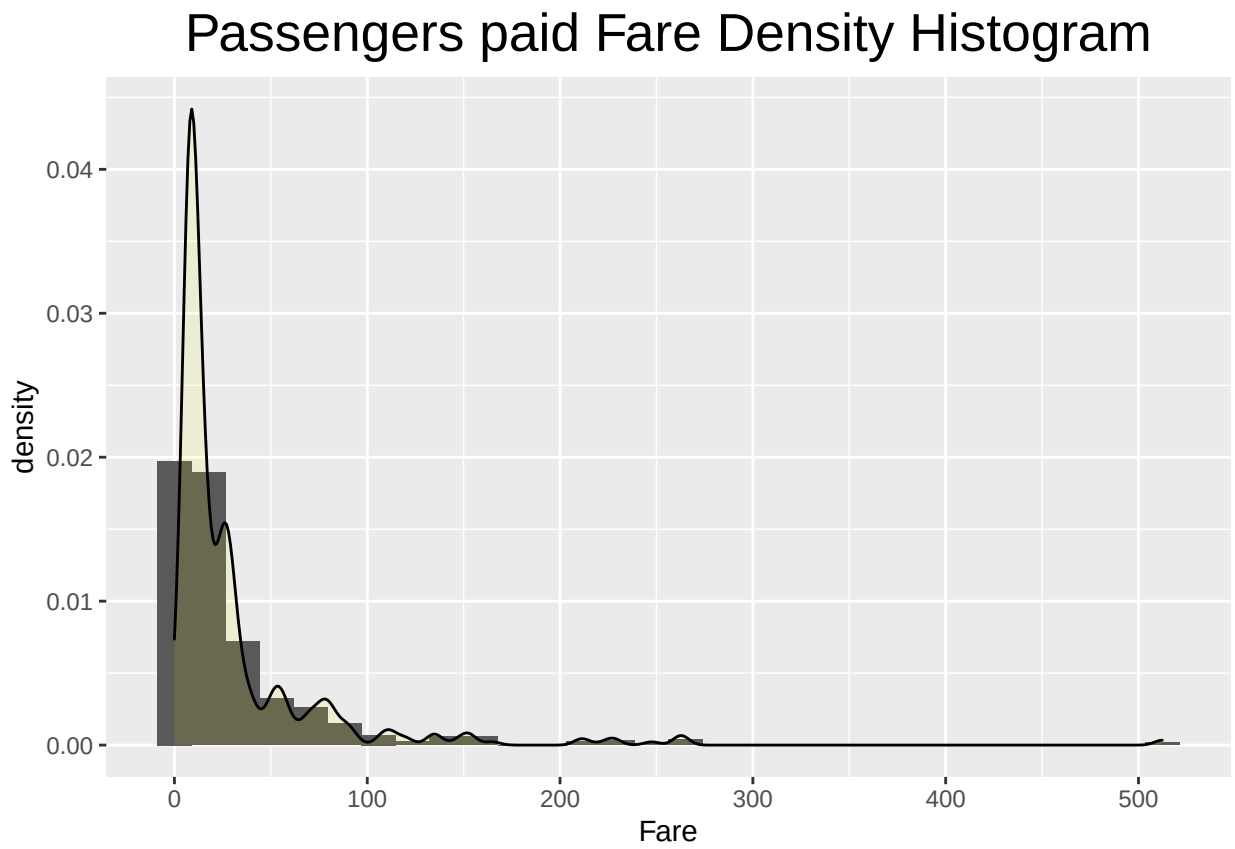

```
# Histograma para la variable Parch
ggplot(ttc, aes(x = Parch)) +
  geom_histogram(aes(y = ..density..)) +
  geom_density(alpha = 0.1, fill = "darkred") +
  ggtitle("Passengers parents/children Density Histogram") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```

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




```
# Histograma para la variable Fare
ggplot(ttc, aes(x = Fare)) +
  geom_histogram(aes(y = ..density..)) +
  geom_density(alpha = 0.1, fill = "yellow") +
  ggtitle("Passengers paid Fare Density Histogram") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```

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

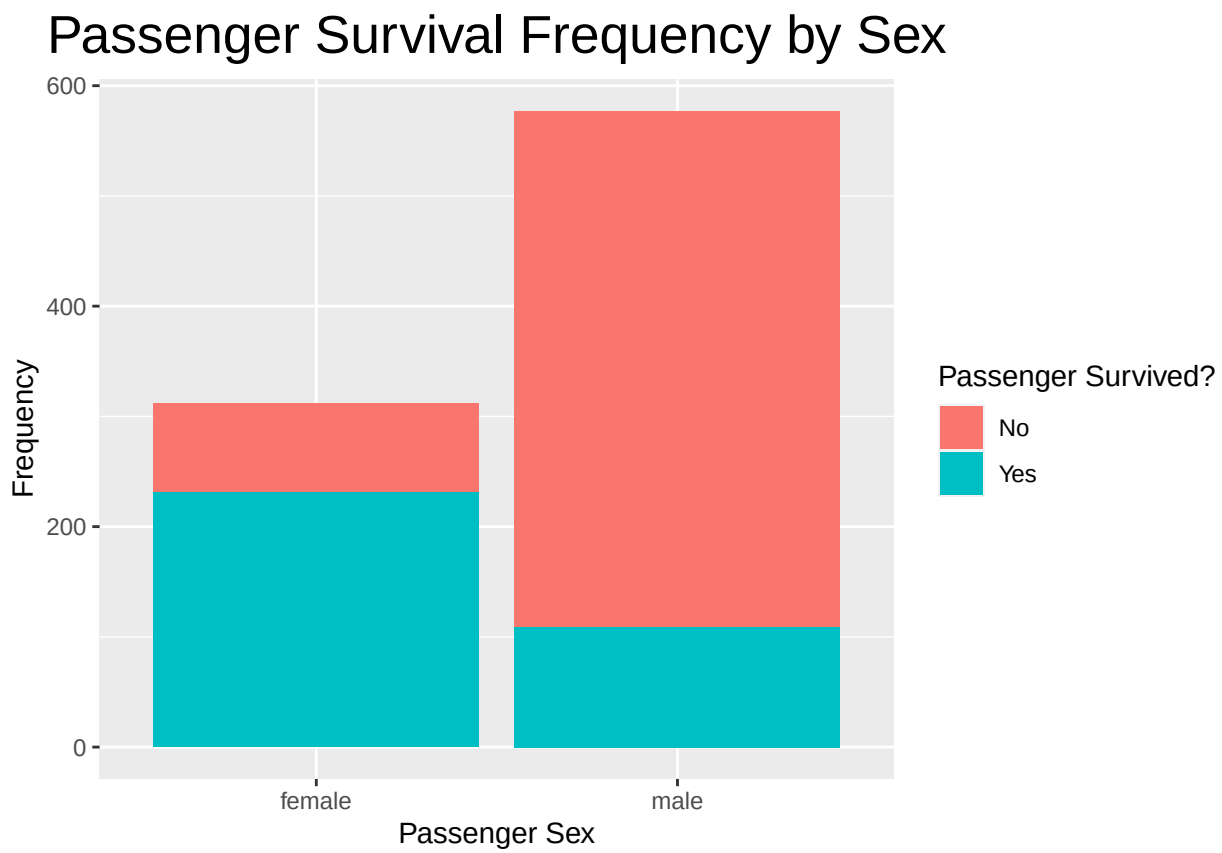


De las gráficas anteriores podemos obtener las siguientes conclusiones:

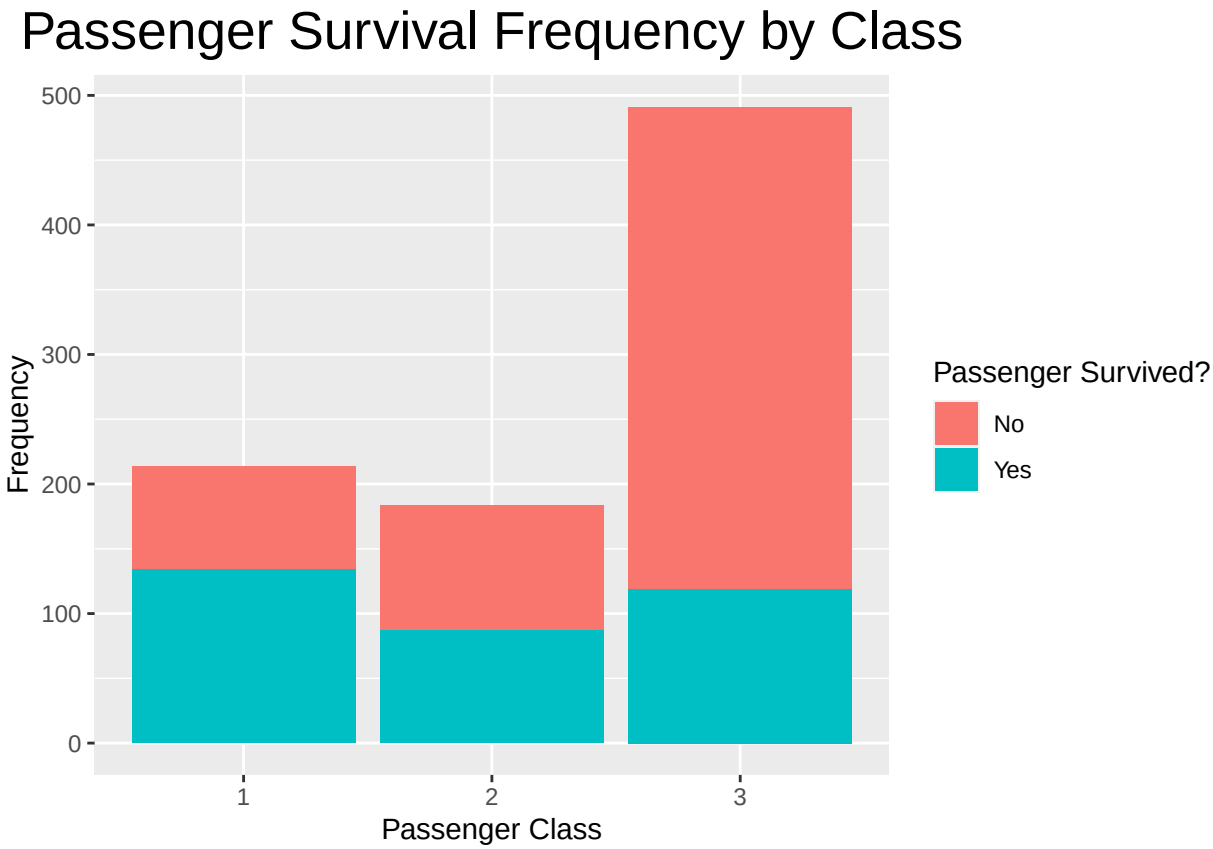
- La variable **Age** se distribuye apróximadamente de una forma normal.
- Las demás variables numéricas presentan una distribución unimodal sesgada hacia la izquierda.

Para una visualizacion general de los datos, podemos representar graficamente los supervivientes agrupados por diversas variables.

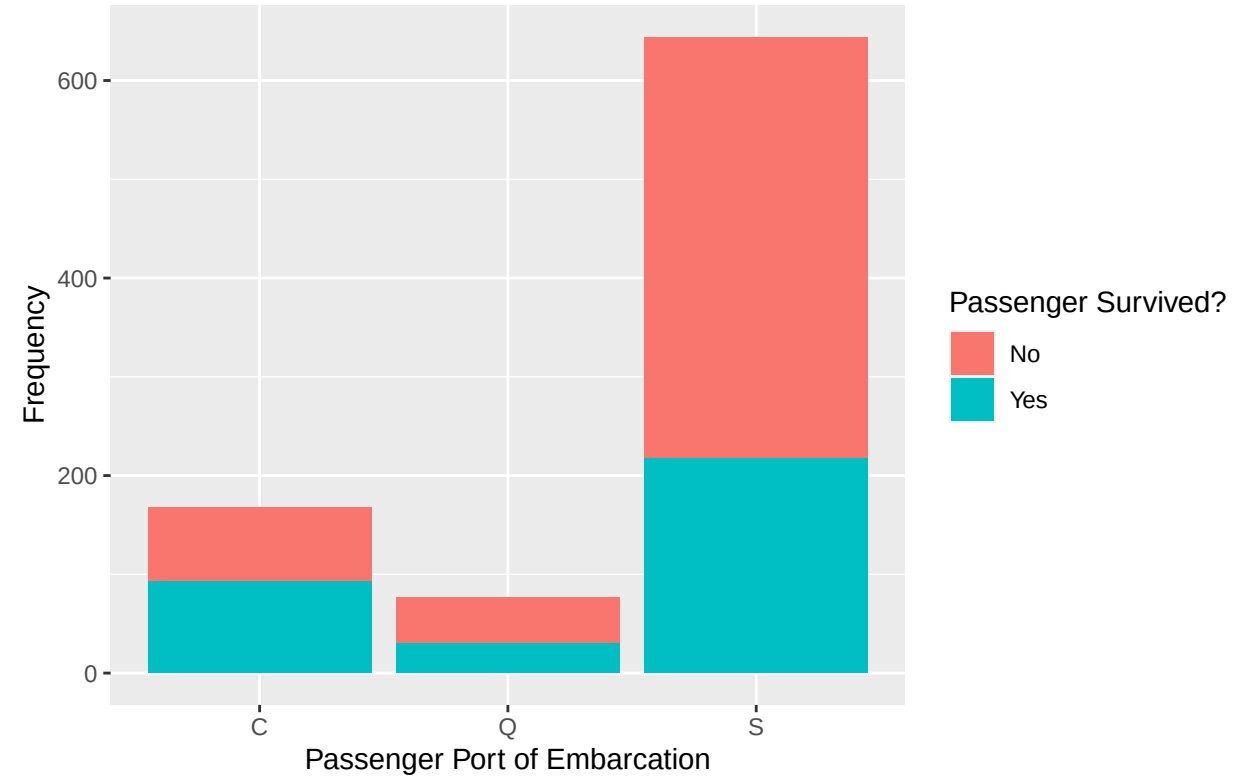
```
# Por ejemplo, la frecuencia de supervivientes por Sexo
ggplot(as.data.frame(table(ttc$Survived, ttc$Sex)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival Frequency by Sex") +
  theme(plot.title = element_text(size = 20, hjust = 0.5)) +
  xlab("Passenger Sex") + ylab("Frequency")
```



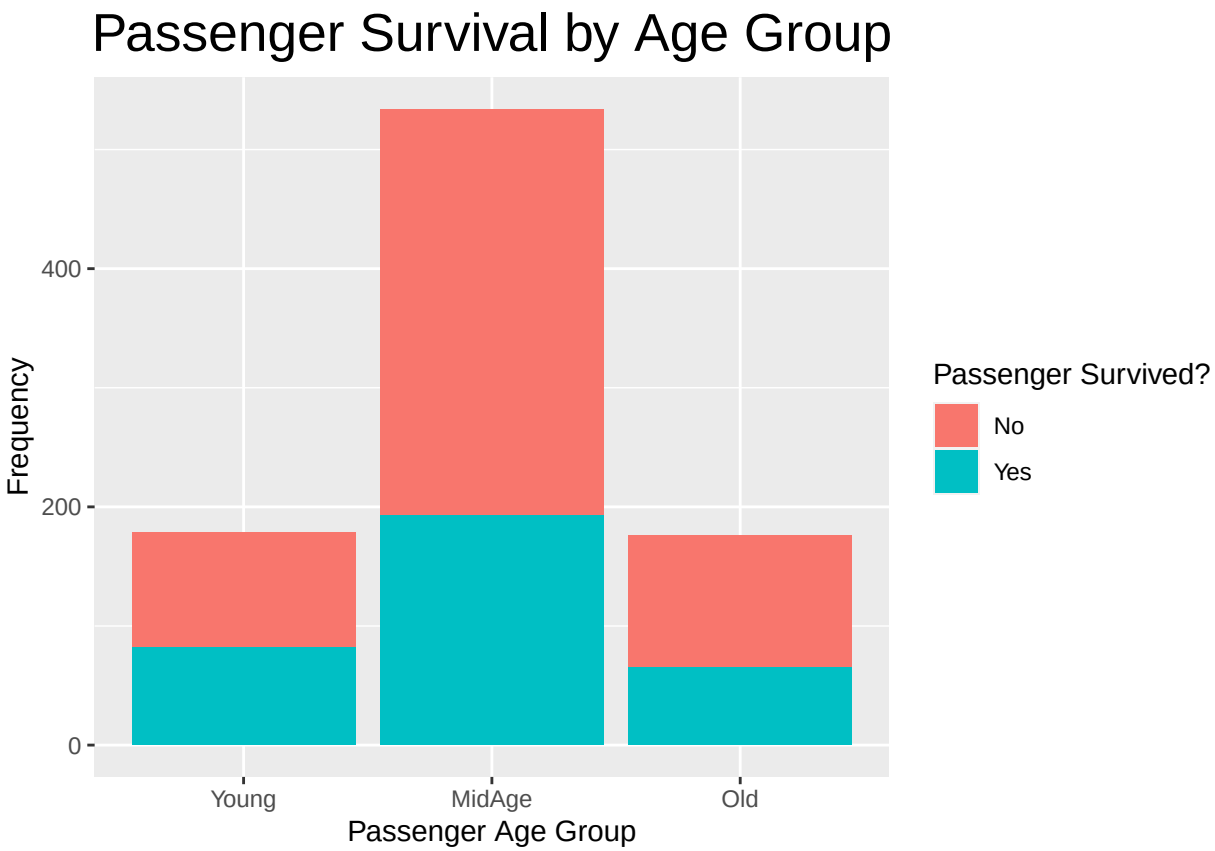

```
# O la frecuencia de supervivientes por Clase
ggplot(as.data.frame(table(ttc$Survived, ttc$Pclass)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival Frequency by Class") +
  theme(plot.title = element_text(size = 20, hjust = 0.5)) +
  xlab("Passenger Class") + ylab("Frequency")
```




```
# O incluso la frecuencia de supervivientes por puerto de embarque
ggplot(as.data.frame(table(ttc$Survived, ttc$Embarked)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival by Port of Embarcation") +
  theme(plot.title = element_text(size = 20, hjust = 0.5)) +
  xlab("Passenger Port of Embarcation") + ylab("Frequency")
```

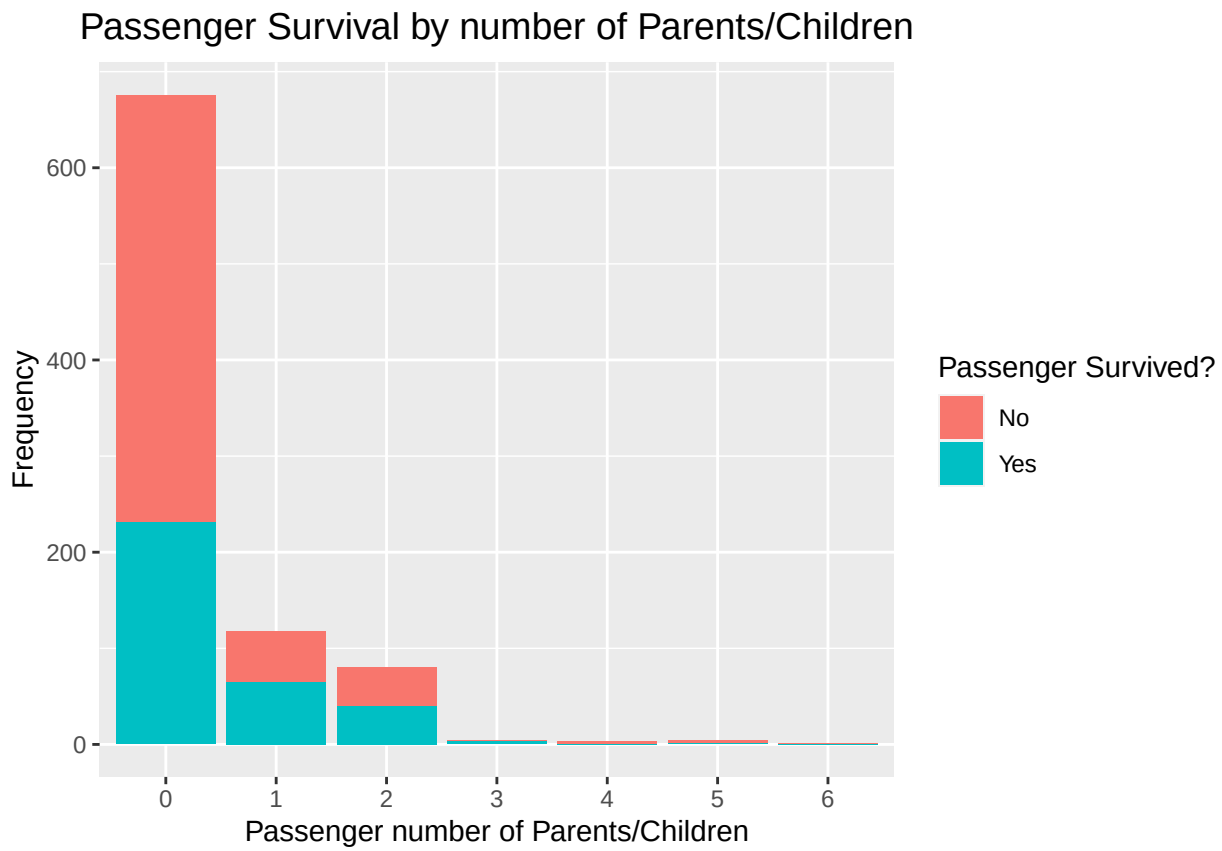



```
# Tambien por el grupo de Edad, aunque previamente debemos discretizar la variable Age
# ya que es numerica continua.

ttc$AgeD <- discretize(ttc$Age,
                       method = "cluster", breaks = 3, labels=c("Young", "MidAge", "Old"))

ggplot(as.data.frame(table(ttc$Survived, ttc$AgeD)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival by Age Group") +
  theme(plot.title = element_text(size = 20, hjust = 0.5)) +
  xlab("Passenger Age Group") + ylab("Frequency")
```




```
# Otra grafica interesante puede ser aquella que muestre la frecuencia de supervivencia
# dependiendo de si el pasajero tenia familiares con el en el barco o viajaban solos

ggplot(as.data.frame(table(ttc$Survived, ttc$SibSp)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival by number of Siblings/Spouses") +
  theme(plot.title = element_text(size = 14, hjust = 0.5)) +
  xlab("Passenger number of siblings/Spouses") + ylab("Frequency")
```

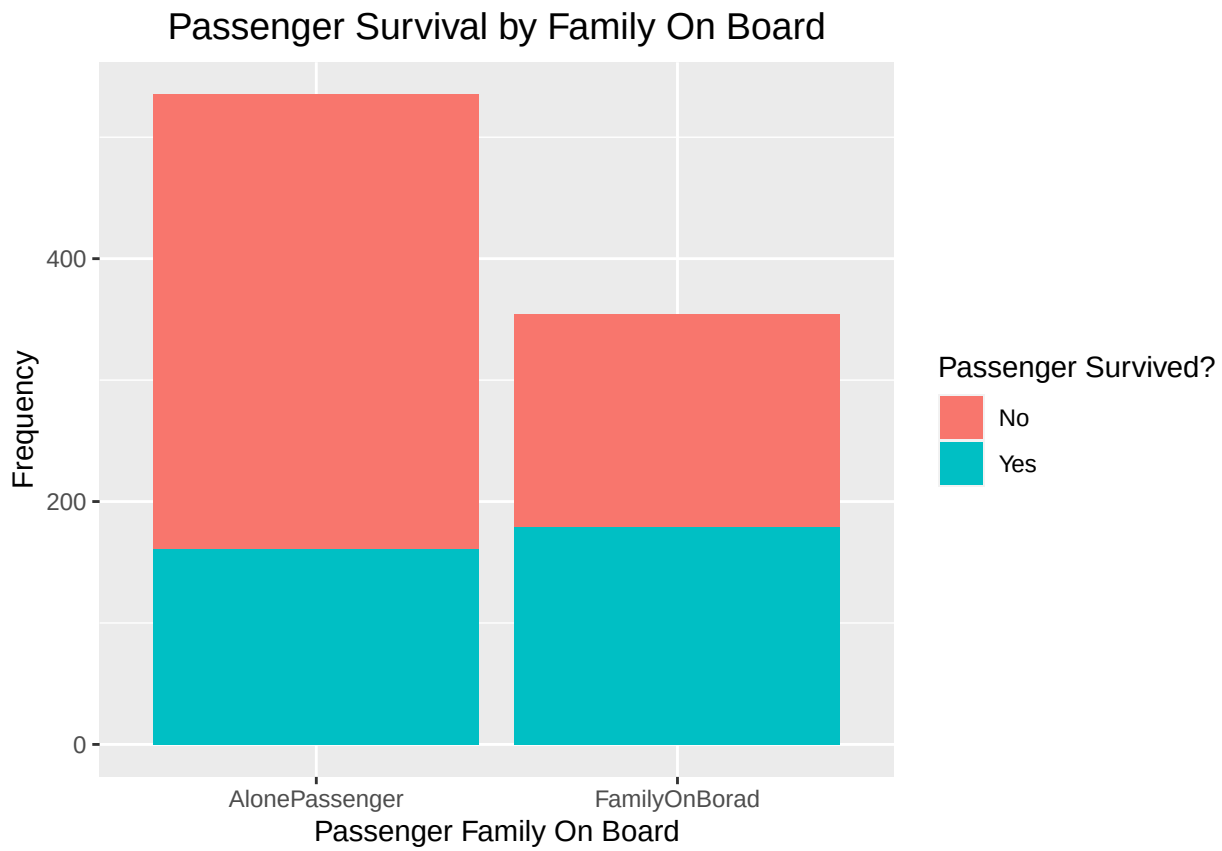

```
ggplot(as.data.frame(table(ttc$Survived, ttc$Parch)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival by number of Parents/Children") +
  theme(plot.title = element_text(size = 14, hjust = 0.5)) +
  xlab("Passenger number of Parents/Children") + ylab("Frequency")
```



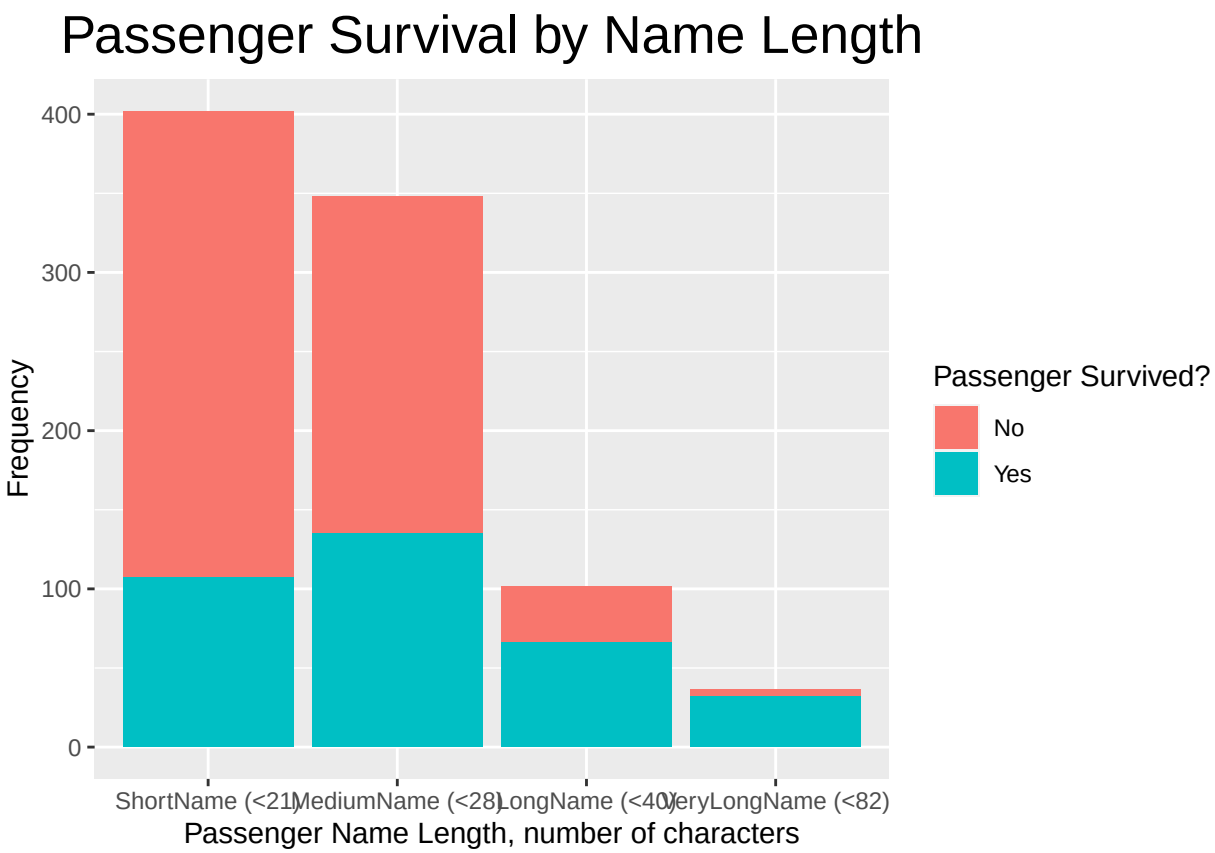

```
# O en general, si el pasajero tenia familia a bordo
ttc$PassengerFamily <- ifelse(ttc$SibSp != 0 | ttc$Parch != 0, 'FamilyOnBorad', "AlonePassenger")
table(ttc$Survived, ttc$PassengerFamily)
```

```
##
##     AlonePassenger FamilyOnBorad
##   0            374           175
##   1            161           179
```

```
ggplot(as.data.frame(table(ttc$Survived, ttc$PassengerFamily)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival by Family On Board") +
  theme(plot.title = element_text(size = 14, hjust = 0.5)) +
  xlab("Passenger Family On Board") + ylab("Frequency")
```

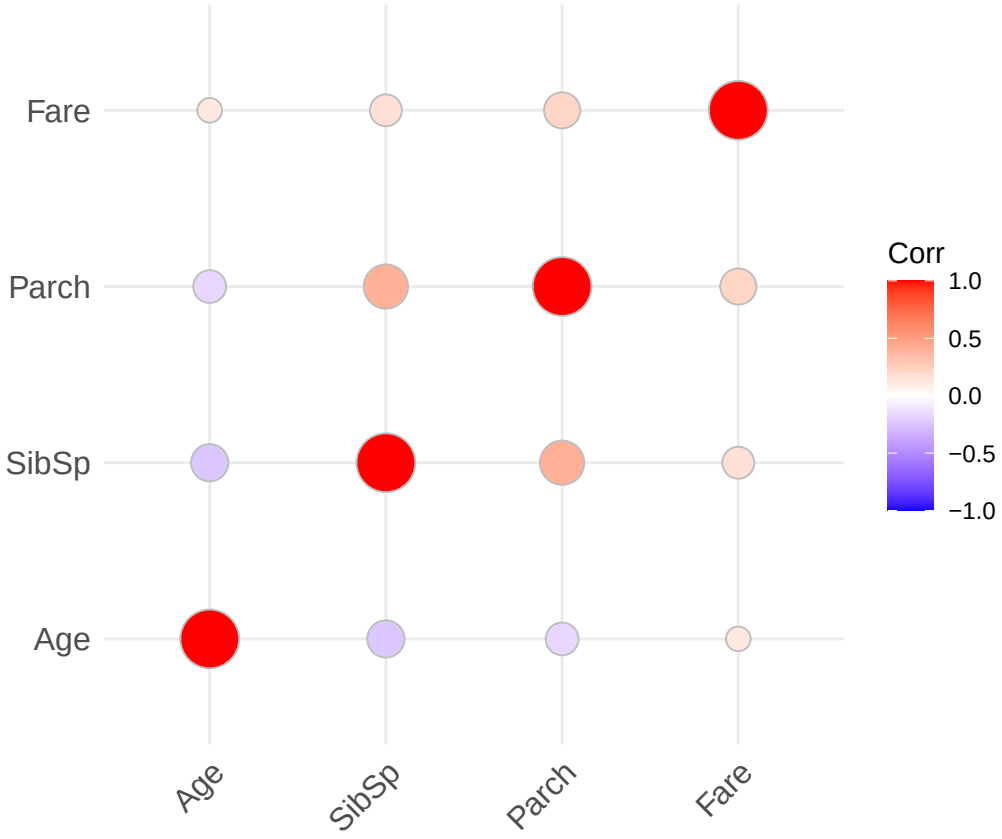



```
# Por ultimo una relacion interesante, es la frecuencia de supervicencia asociada
# a la longitud del nombre del pasajero, bajo una premisa inicial de que, cuanto
# mas largo fuera el nombre, el pasajero podria tener una clase social mas elevada
ttc$NameLength <- vector("numeric", nrow(ttc))
for (i in 1:nrow(ttc)) {
  ttc$NameLength[i] <- nchar(as.character(ttc$Name)[i])
}
ttc$NameLengthD <- discretize(ttc$NameLength,
           method = "cluster", breaks = 4, labels=c("ShortName (<21)",
                                                    "MediumName (<28)",
                                                    "LongName (<40)",
                                                    "VeryLongName (<82)"))
ggplot(as.data.frame(table(ttc$Survived, ttc$NameLengthD)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival by Name Length") +
  theme(plot.title = element_text(size = 20, hjust = 0.5)) +
  xlab("Passenger Name Length, number of characters") + ylab("Frequency")
```



Por ultimo en el proceso de exploracion de los datos, se puede obtener una matriz de correlacion sobre las variables numericas del dataset:

```
ttc_num <- subset(ttc, select=c(Age, SibSp, Parch, Fare))
ttccorr <- cor(ttc_num)
ggcorrplot(ttccorr, method = "circle")
```


Fare
Parch
SibSp
Age
Age
SibSp
Parch
Fare
Corr
1.0
0.5
0.0
−0.5
−1.0