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

Francisco Javier Melchor y Enrique Otero

04/01/2021

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:

Pregunta 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 analítica 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	

Variable	Definition	Key
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	

Análisis exploratorio del dataset de entrenamiento.

```
# Carga de Librerias
library(ggplot2)
library(arc)
library(ggcorrplot)
```

En esta sección realizamos una primera exploración del dataset incluido en el fichero **train.csv**

```
# Cargamos el dataset train.csv en la variable ttc
ttc <- read.csv("/home/dsninja/DataScience/WorkAreas/RStudio/datasets/titanic_train.csv")
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              S
## 2         PC 17599 71.2833   C85        C
## 3 STON/O2. 3101282  7.9250              S
## 4           113803 53.1000  C123        S
## 5           373450  8.0500              S
## 6           330877  8.4583              Q
```

```
str(ttc)
```

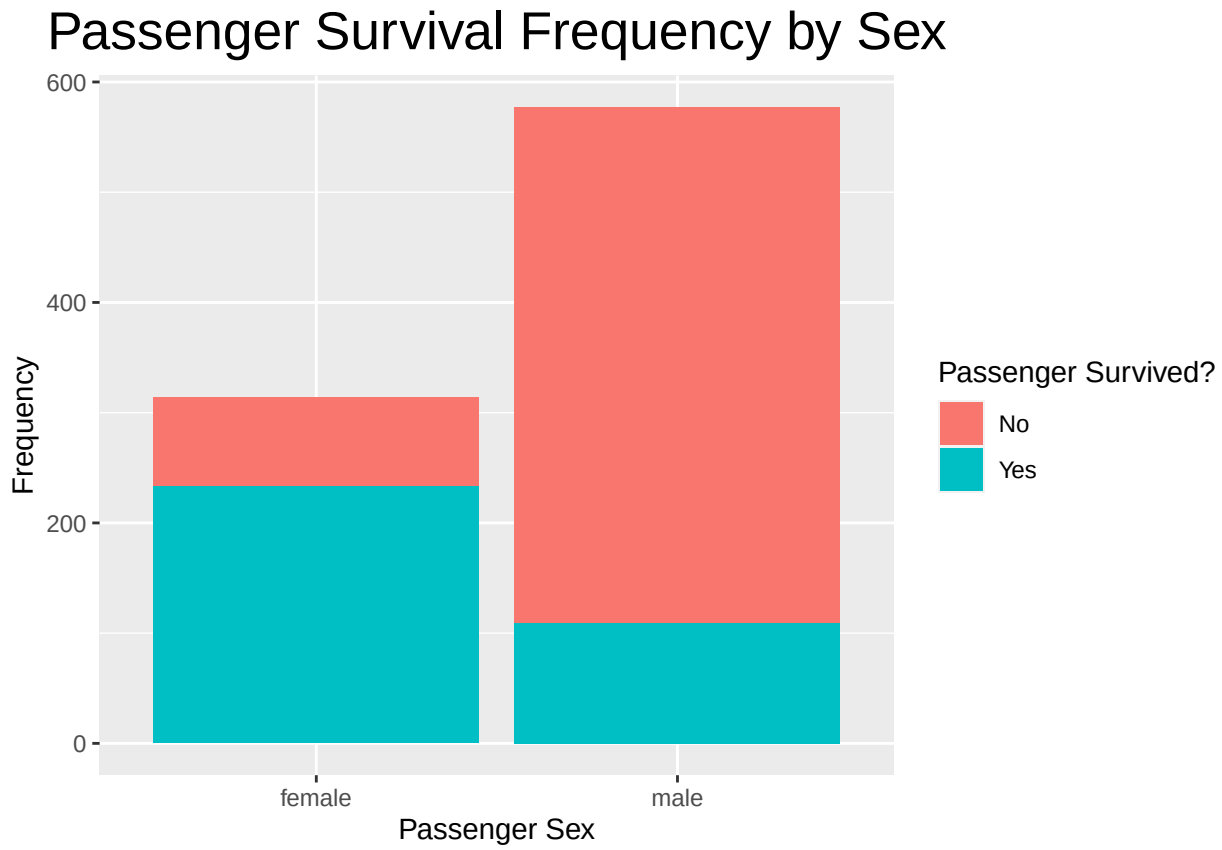
```
## 'data.frame':    891 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 417 581
##  $ 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     : Factor w/ 681 levels "110152","110413",..: 524 597 670 50 473 276 86 396 345 133 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : Factor w/ 148 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
```

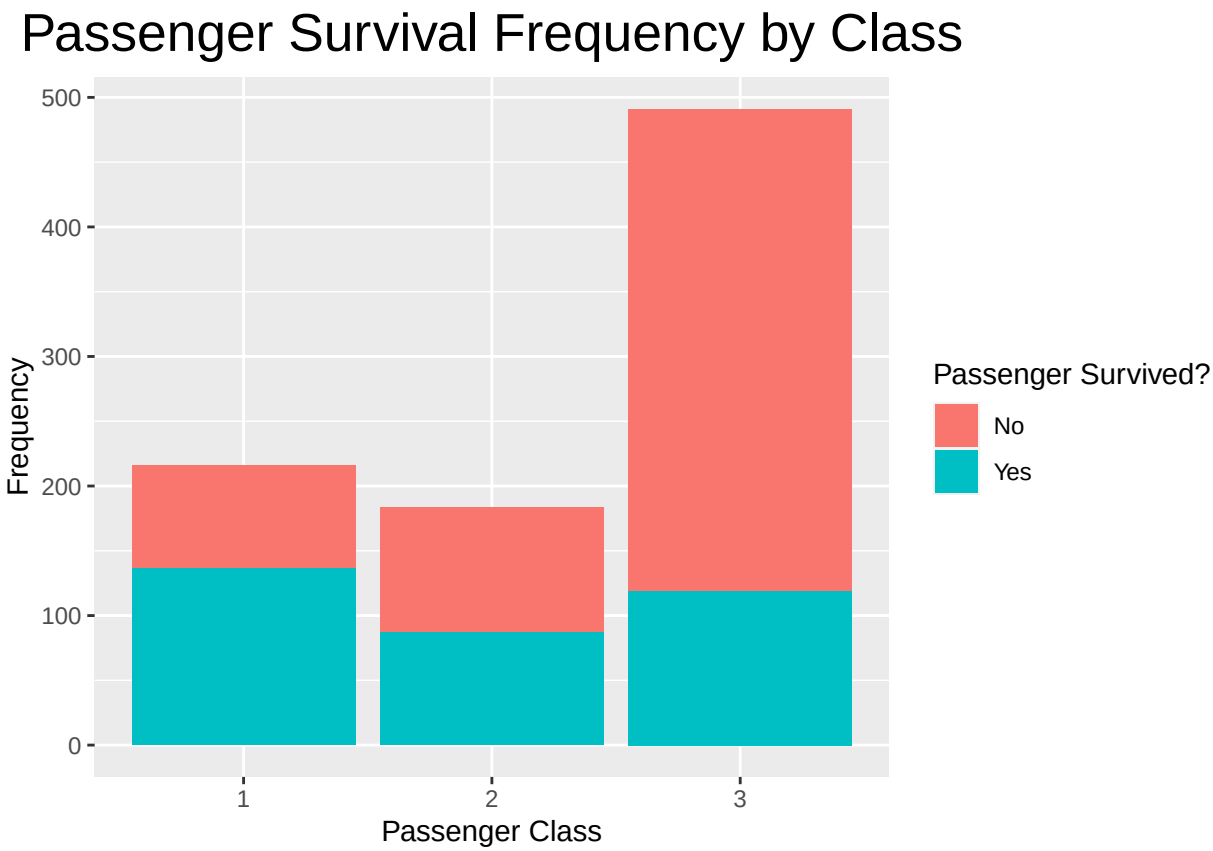
```
# Contamos con 891 observaciones de las 12 variables decritas al inicio de esta seccion.
```

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

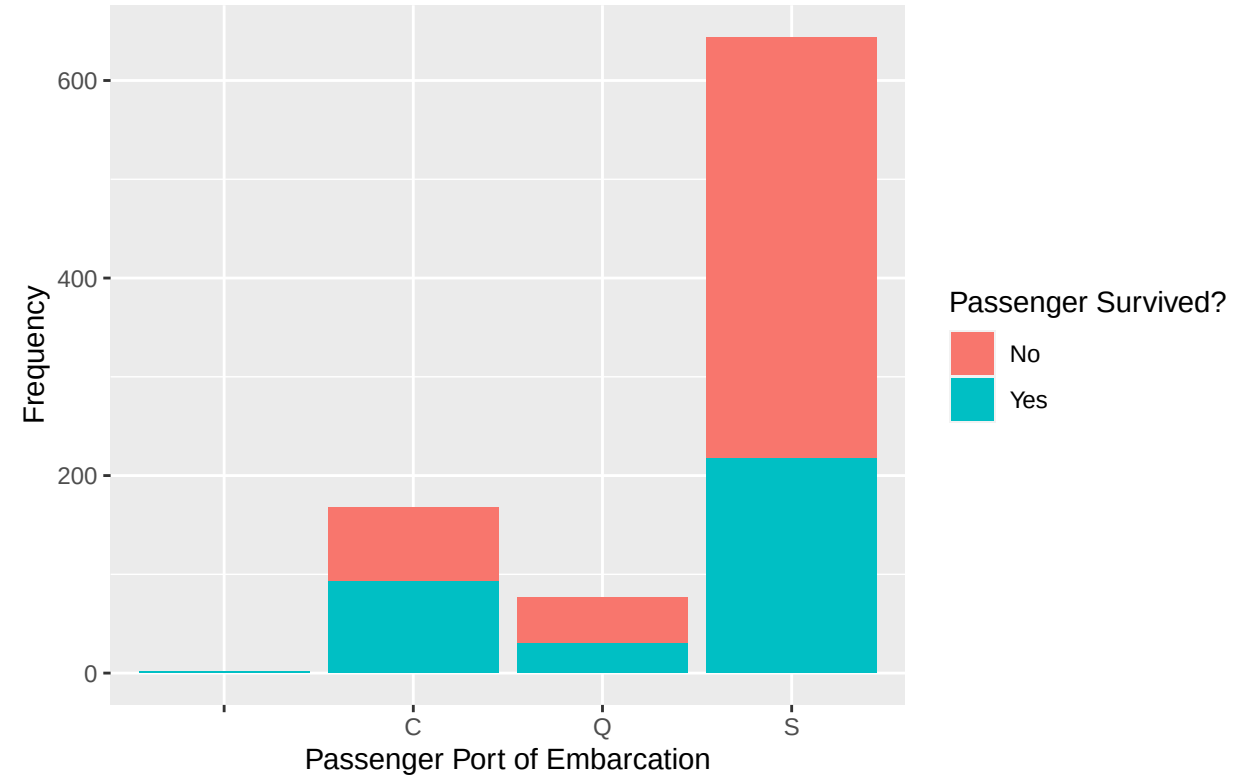
```
# Por ejemplo, la frecuencia de supervivientes por Sexo
ggplot(as.data.frame(table(ttc$Survived, ttc$Sex)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival Frequency by Sex") +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



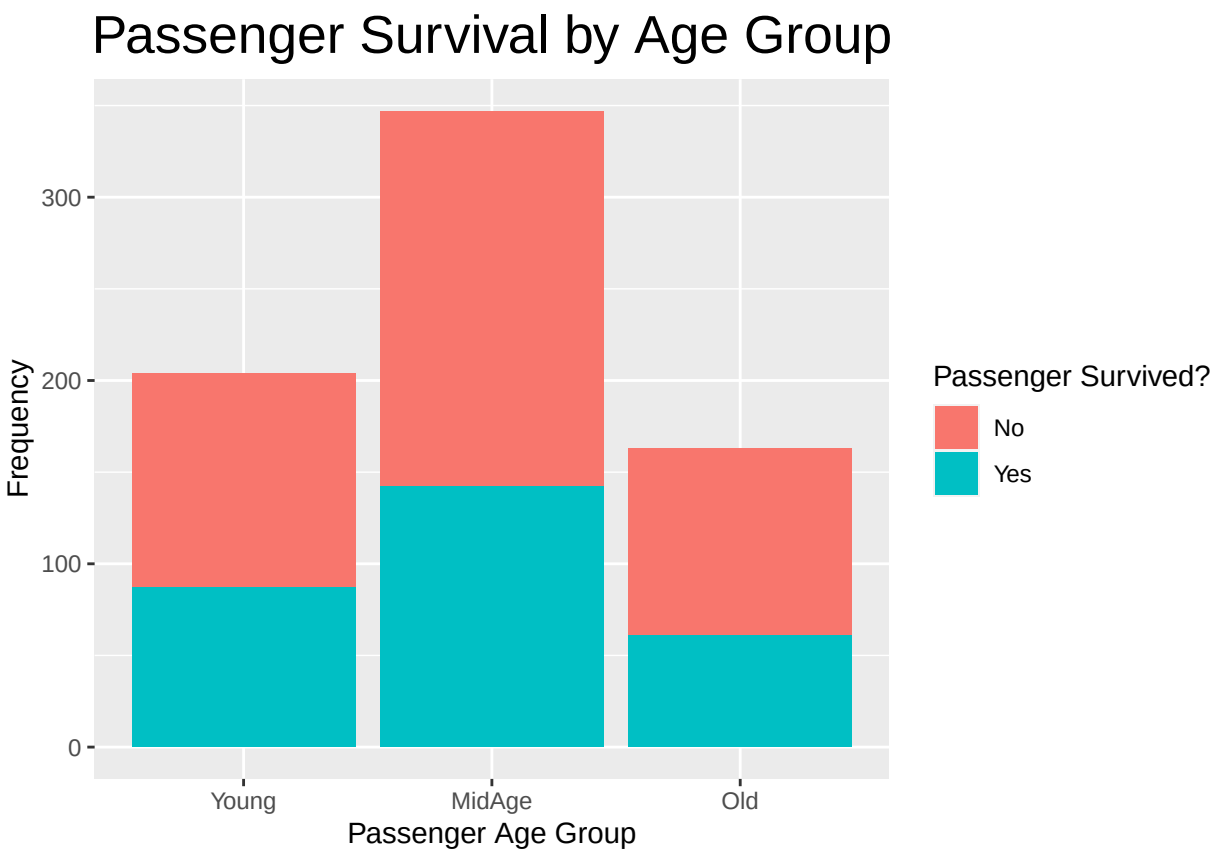

```
# O la frecuencia de supervivientes por Clase
ggplot(as.data.frame(table(ttc$Survived, ttc$Pclass)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival Frequency by Class") +
  theme(plot.title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

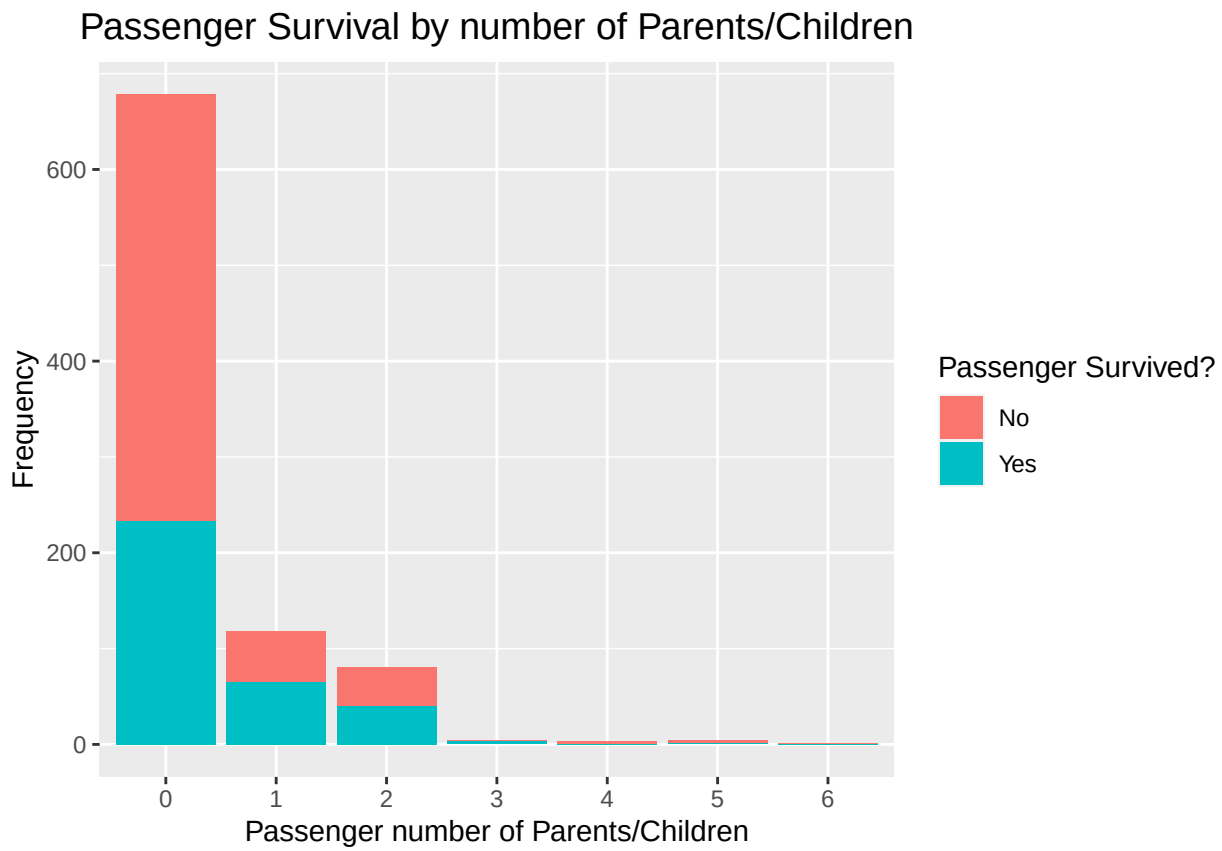



```
# Tambien por el grupo de Edad, aunque previamente debemos discretizar la variable Age
# ya que es numerica continua.

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")
```

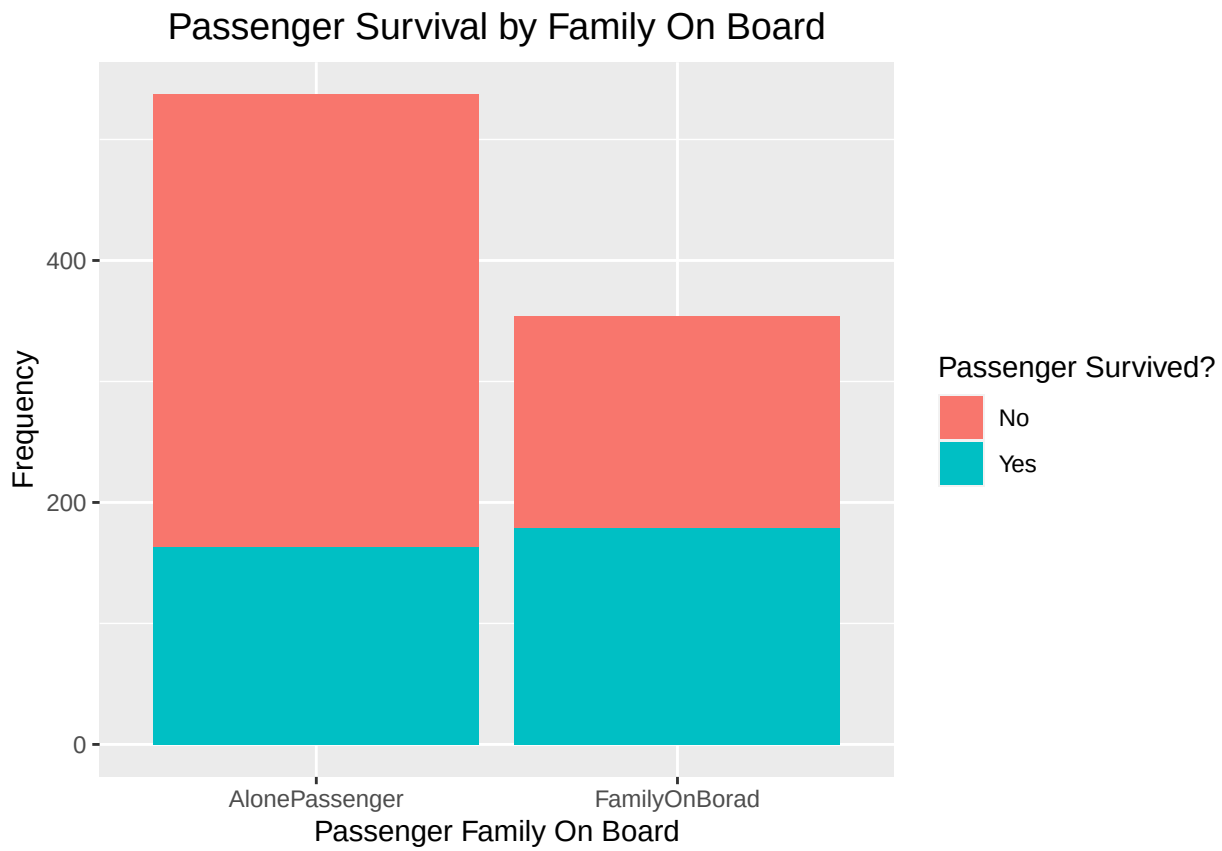

```
ggplot(as.data.frame(table(ttc$Survived, ttc$Parch)), aes(Var2, Freq, fill=Var1)) +
  geom_bar(stat="identity") +
  scale_fill_discrete(name = "Passenger Survived?", labels = c("No", "Yes")) +
  ggtitle("Passenger Survival by number of Parents/Children") +
  theme(plot.title = element_text(size =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



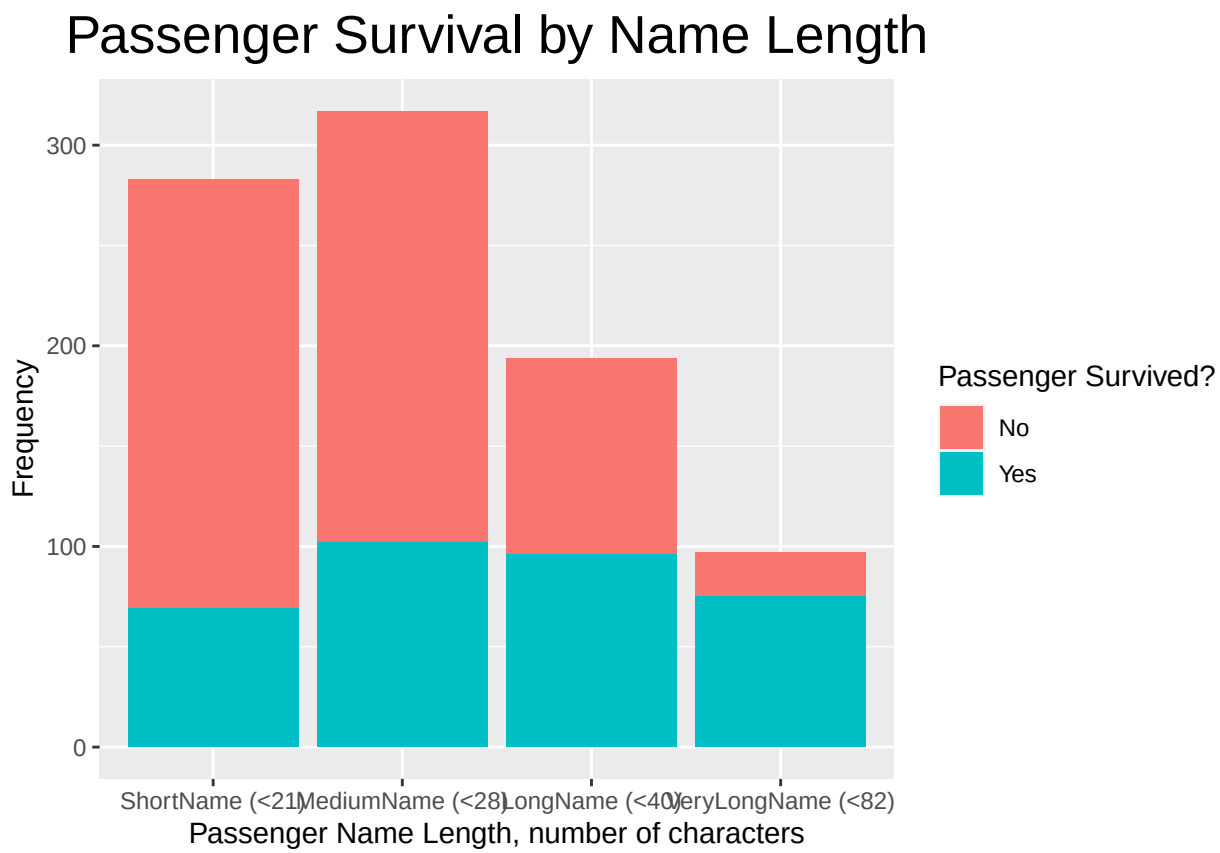

```
# O en general, si el pasajero tenia familia a bordo
ttc$PassengerFamily <- ifelse(ttc$SibSp != 0 | ttc$Parch != 0, 'FamilyOnBorad', "AlonePassenger")
table(ttc$Survived, ttc$PassengerFamily)
```

```
##
##     AlonePassenger FamilyOnBorad
##   0            374           175
##   1            163           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")
```



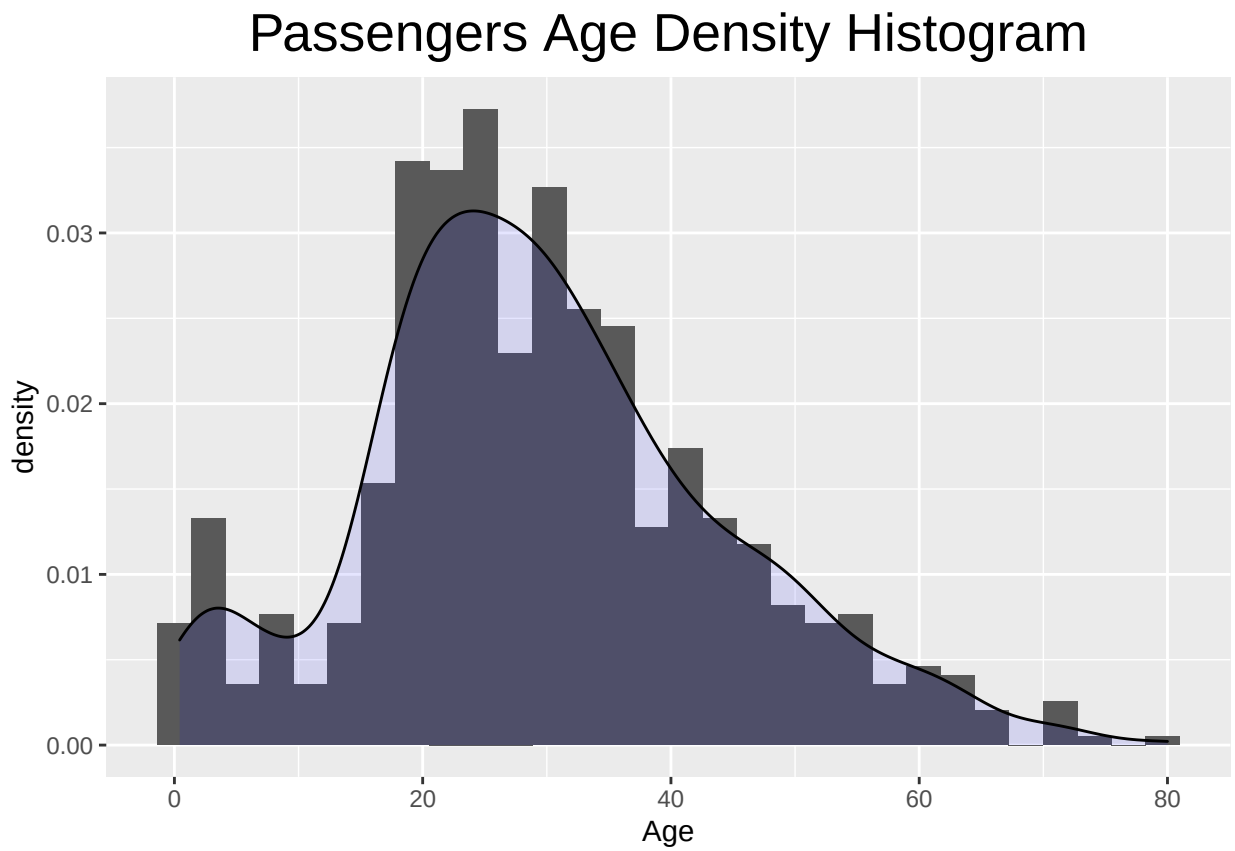

```
# Por ultimo una relacion interesante, es la frecuencia de supervicencia asociada
# a la longitud del nombre del pasajero, bajo una premisa inicial de que, cuanto
# mas largo fuera el nombre, el pasajero podria tener una clase social mas elevada
ttc$NameLength <- vector("numeric", nrow(ttc))
for (i in 1:nrow(ttc)) {
  ttc$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")
```



Tambien podemos representar los Histrogramas de las variables numericas continuas del dataset.

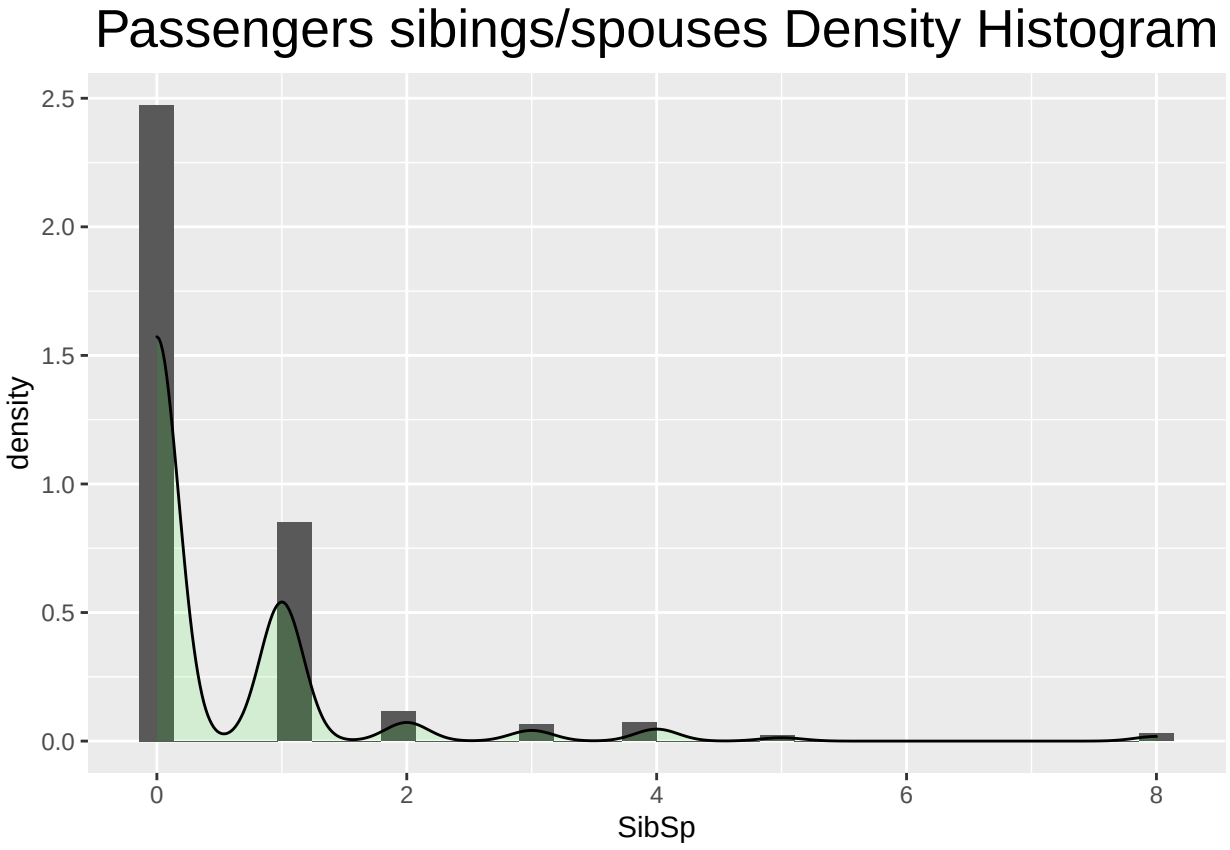
```
# Histograma para la variable Age
ggplot(ttc, aes(x = Age)) +
  geom_histogram(aes(y = ..density..)) +
  geom_density(alpha = 0.1, fill = "blue") +
  ggtitle("Passengers Age Density Histogram") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 177 rows containing non-finite values (stat_bin).
## Warning: Removed 177 rows containing non-finite values (stat_density).
```



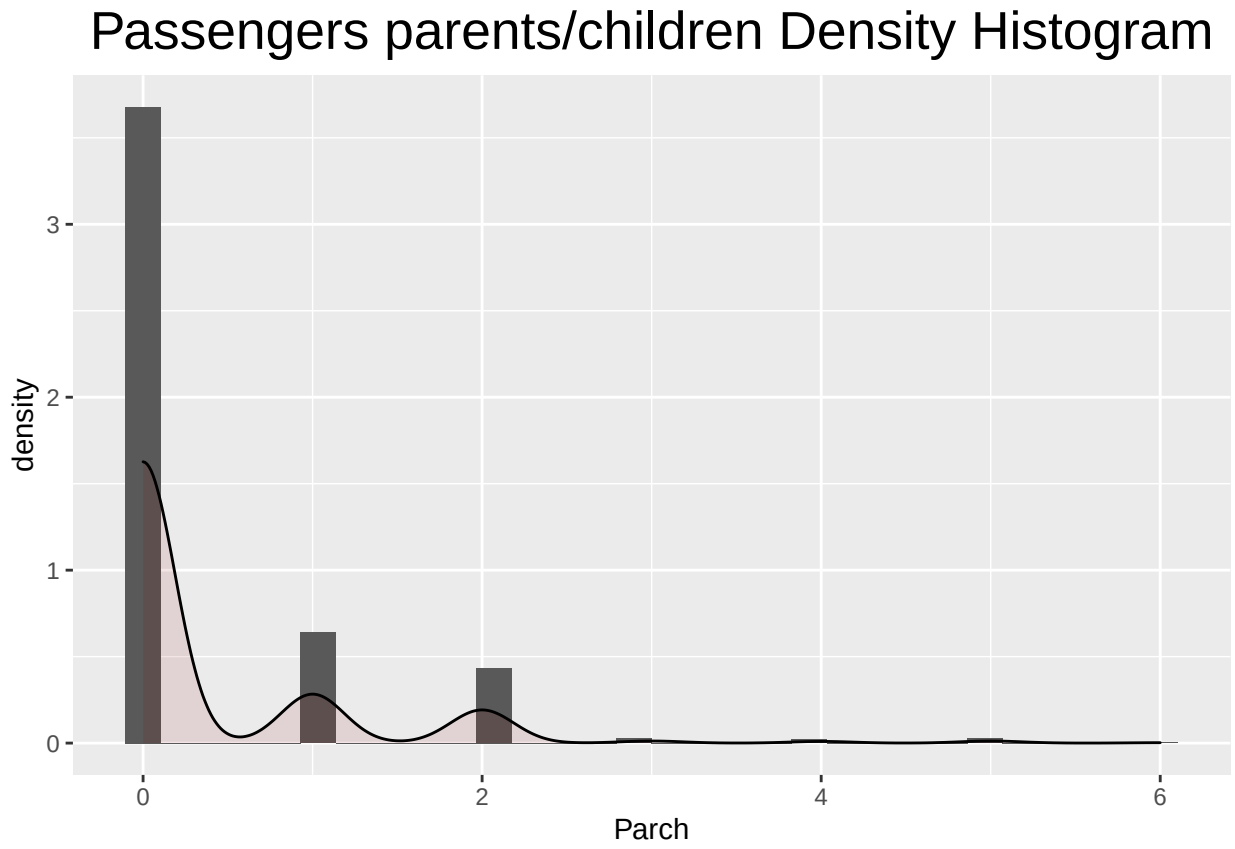

```
# Histograma para la variable SibSp
ggplot(ttc, aes(x = SibSp)) +
  geom_histogram(aes(y = ..density..)) +
  geom_density(alpha = 0.1, fill = "green") +
  ggtitle("Passengers sibings/spouses Density Histogram") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```

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



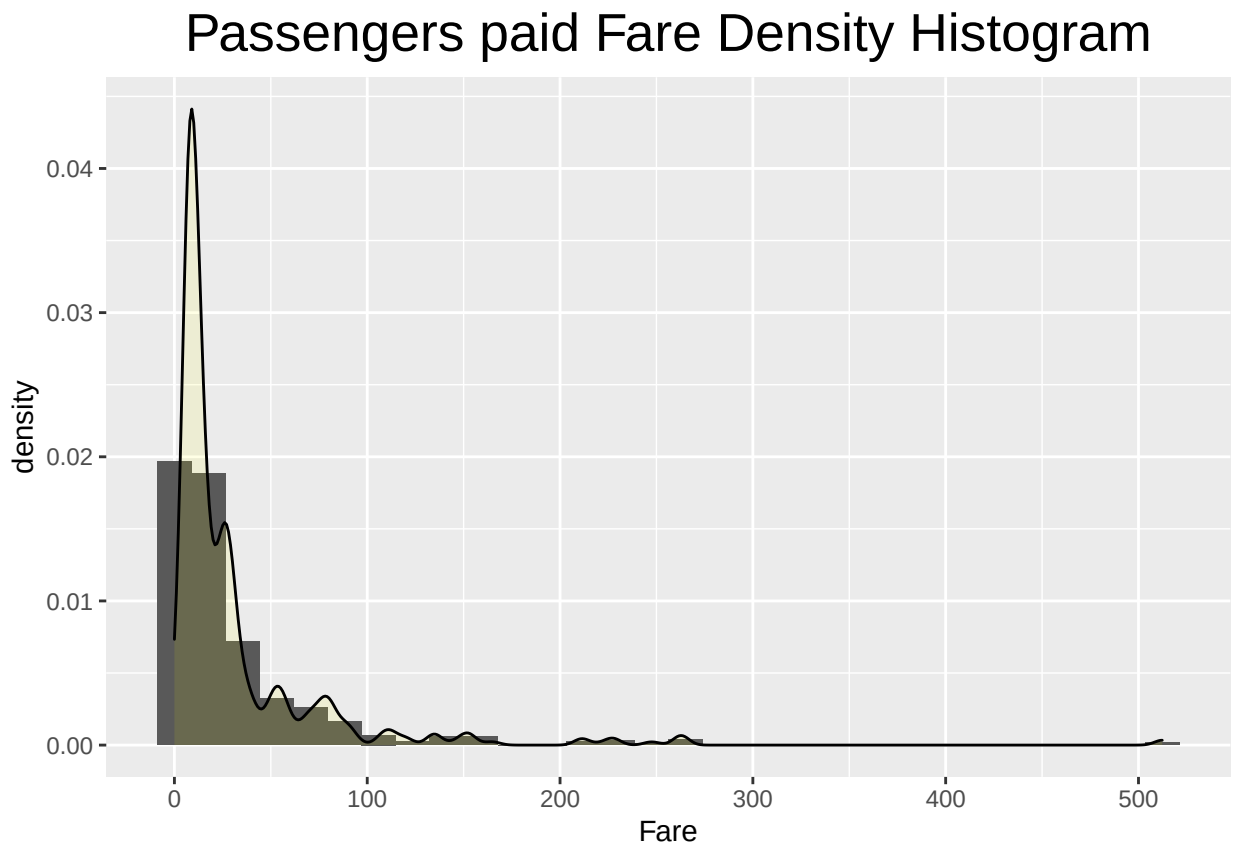

```
# Histograma para la variable Parch
ggplot(ttc, aes(x = Parch)) +
  geom_histogram(aes(y = ..density..)) +
  geom_density(alpha = 0.1, fill = "darkred") +
  ggtitle("Passengers parents/children Density Histogram") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```

```
## `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`.
```



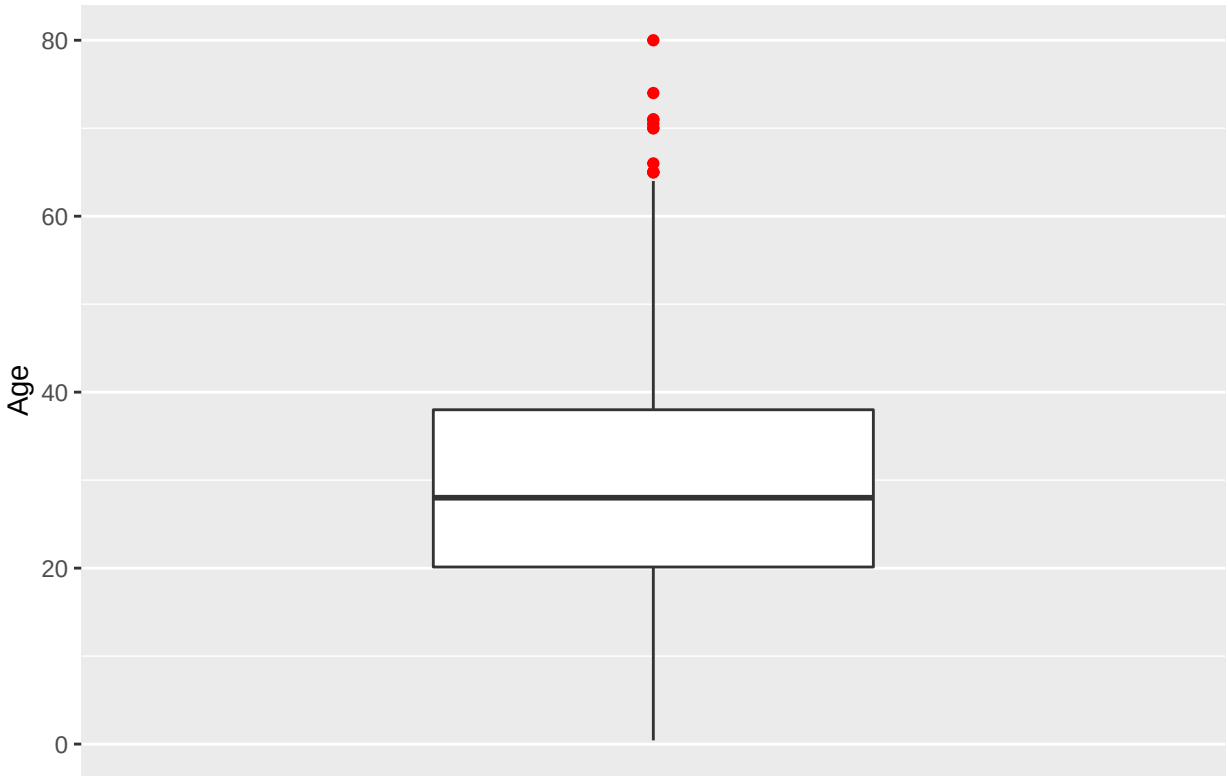
Los diagramas de boxplots de las variables numericas tambien son significativos, sobre todo para la deteccion de outliers:

```
# Boxplot para la variable Age
# Esta variable, ademas de outliers, posee 177 registros NAs
summary(ttc$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.42   20.12   28.00   29.70   38.00   80.00     177
```

```
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))
```

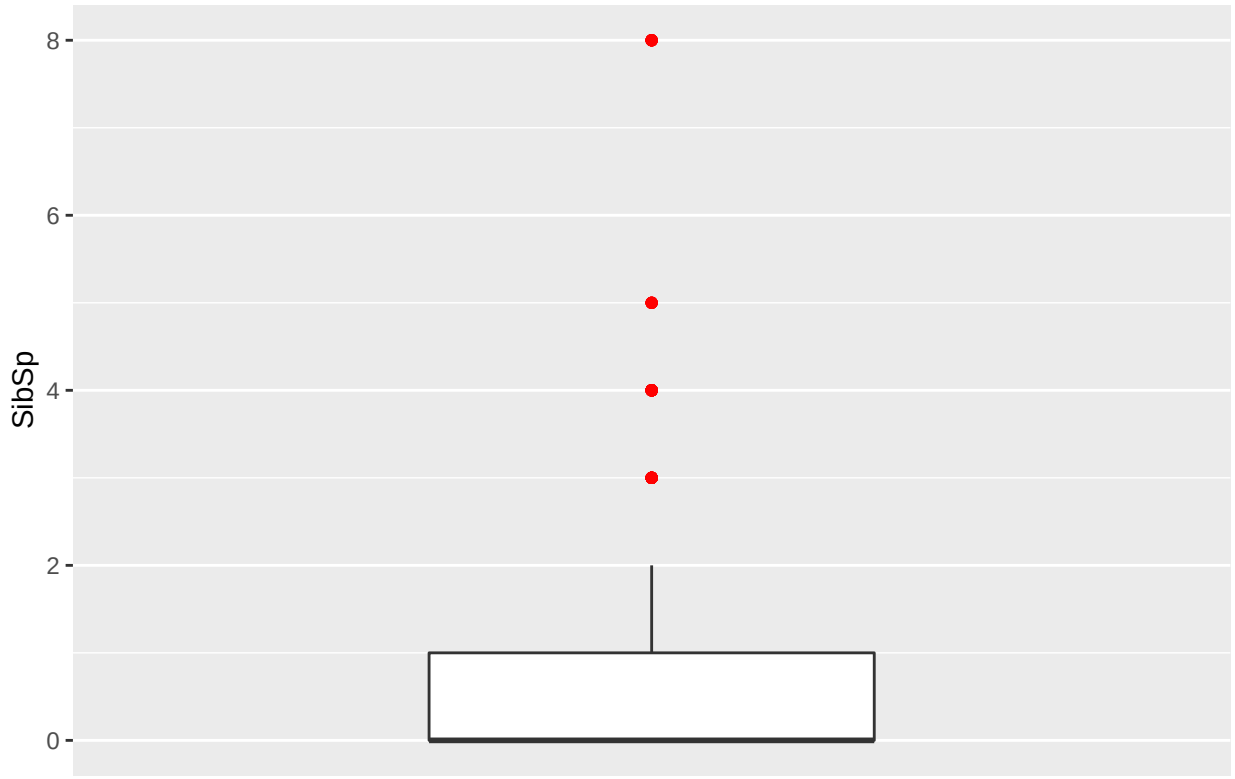
```
## Warning: Removed 177 rows containing non-finite values (stat_boxplot).
```




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

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.523   1.000   8.000
```

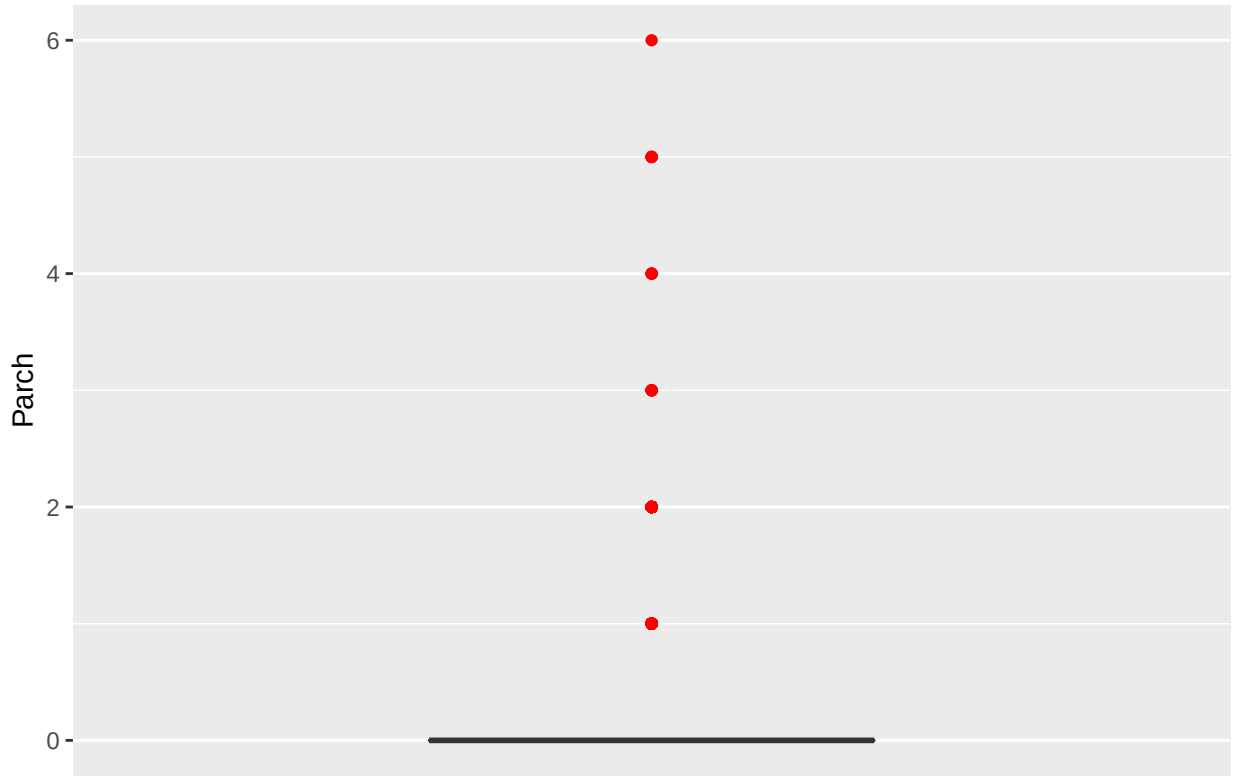
```
ggplot(data = ttc, aes(y = SibSp)) +
  geom_boxplot(outlier.colour = "red") +
  scale_x_discrete(guide = guide_axis(n.dodge = 2)) +
  ggtitle("Passenger sibings/spouses Boxplot") +
  theme(plot.title = element_text(size = 20, hjust = 0.5))
```




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

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.3816  0.0000  6.0000
```

```
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))
```



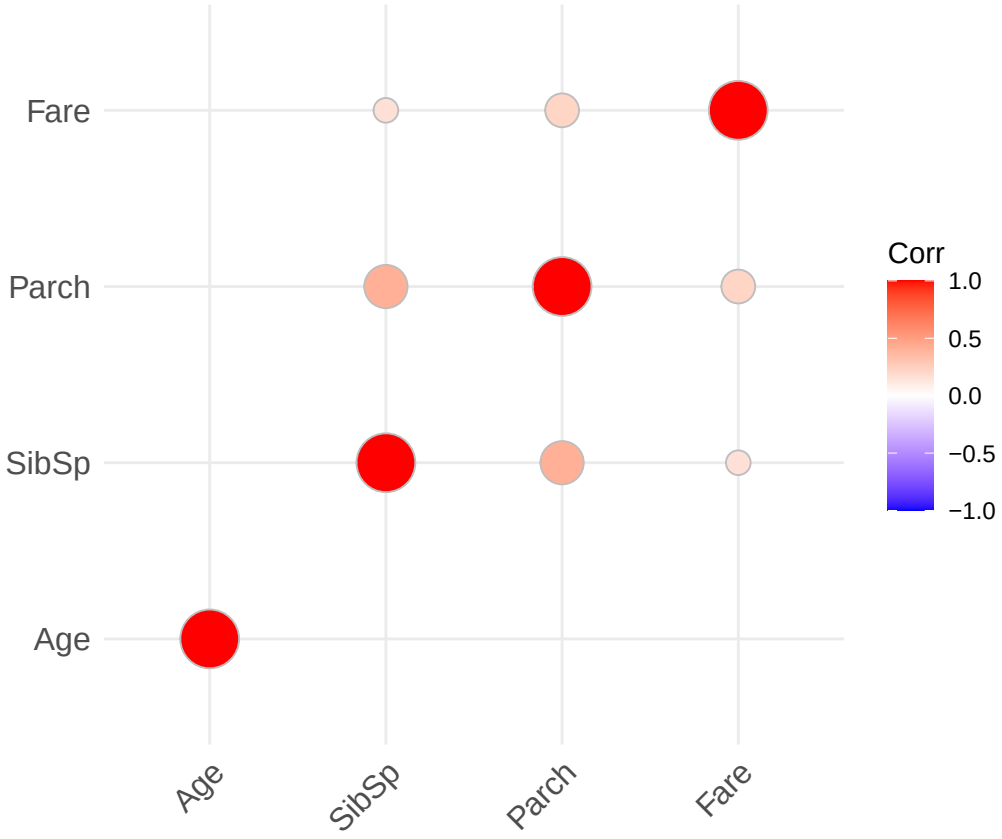

```
# Boxplot para la variable Fare
summary(ttc$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    7.91   14.45   32.20   31.00  512.33
```

```
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))
```

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