Universitat Oberta de Catalunya

Estudios de Informática, Multimedia y Telecomunicaciones

Práctica 2: Limpieza y análisis de datos

Cristina Merino García de la Reina, Isabel González Valle

31 de mayo, 2020

Descripción

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 (https://www.kaggle.com)).

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 esta práctica hemos buscado un dataset de los vuelos que fueron cancelados o que sufrieron retrasos en durante el año 2015 comunicados a través del Departamento de Estadísticas de Transportes de los Estados Unidos. Este conjunto de datos contiene la información correspondiente

a los vuelos operados por las grandes compañías aéreas.

Poder conocer y tener una referencia del motivo de los vuelos cancelados o retrasados es interesante para las personas que deben usar este medio de transporte. Se pueden analizar diferentes problemáticas, como la relación entre los vuelos retrasados y los días de la semana, así como identificar cuál puede ser el mejor mes para viajar, que aeropuerto debemos evitar y por último, buscaremos identificar cuál es la mejor compañía para viajar. Intentaremos dar respuesta a algunas de estas preguntas durante esta práctica.

https://www.kaggle.com/usdot/flight-delays (https://www.kaggle.com/usdot/flight-delays)

El tipo de licencia de este dataset es: *CC0 1.0 Universal (CC0 1.0) Public Domain Dedication* . Por lo tanto es público y puede ser utilizado libremente para el trabajo que vamos a realizar.

El conjunto de datos elegido contiene 31 variables y casi 6 millones de observaciones, por lo que de cara a la práctica reduciremos la cantidad de datos, intentando que la muestra a utilizar represente el conjunto de datos original lo más fielmente posible, para ello utilizaremos las técnicas de muestreo que se estudiaron en el módulo anterior.

Del mismo modo, eliminaremos aquellas variables que no aporten valor al estudio que vamos a realizar, reduciendo así la dimensionalidad del conjunto de datos.

En cuanto a las variables del dataset, se tienen las siguientes:

YEAR: Año del vuelo (2015)

MONTH: Mes del vuelo

DAY: Día del vuelo

DAY_OF_WEEK: Día de la semana, donde el día 1=lunes y el 7=Domingo

AIRLINE: Código de la aerolínea

FLIGHT_NUMBER: Número de vuelo

TAIL_NUMBER: Número de identificación de la aeronave

ORIGIN_AIRPORT: Aeropuerto Origen

DESTINATION_AIRPORT: Aeropuerto Destino

SCHEDULED_DEPARTURE: Hora programada de salida en formato hhmm (55 -> 00:55)

DEPARTURE_TIME: Hora de salida del vuelo en formato hhmm

DEPARTURE_DELAY: Diferencia en minutos entre la salida programada y la real (valores negativos identifican salidas del vuelo con antelaión)

TAXI_OUT: Tiempo de rodaje del avión desde que deja la puerta de embarque hasta despegue.

WHEELS_OFF: Hora en la que el avión despega, momento en el que las ruedas del avión dejan de tocar el suelo

SCHEDULED_TIME: Tiempo programado de vuelo.

ELAPSED_TIME: Tiempo total de vuelo contado desde el momento que el avión se pone en marcha hasta que para completamente en destino, es decir contando el rodaje en el aeropuerto.

AIR_TIME: Tiempo desde despegue hasta aterrizaje

DISTANCE: Distancia en millas

WHEELS_ON: Hora en la que el avión toca tierra.

TAXI_IN: Tiempo de rodaje en el aeropuerto destino hasta que el avión para completamente.

SCHEDULED_ARRIVAL: Hora programada de llegada en formato hhmm

ARRIVAL_TIME: Hora de llegada real en formato hhmm

ARRIVAL_DELAY: Diferencia en minutos entre la salida programada y la real

DIVERTED: Vuelo desviado (0-No, 1-Sí) *CANCELLED: Vuelo Cancelado (0-No, 1-Sí)*

CANCELLATION_REASON: Motivo de cancelación (A-Carrier, B-Weather, C-National Air System, D-Security) *AIR_SYSTEM_DELAY: Tiempo de retraso por el motivo indicado*

SECURITY_DELAY: Tiempo de retraso por el motivo indicado

AIRLINE_DELAY: Tiempo de retraso por el motivo indicado

LATE_AIRCRAFT_DELAY: Tiempo de retraso por el motivo indicado

*WEATHER_DELAY: Tiempo de retraso por el motivo indicado

```
#Cargamos el dataset
vuelos <- read.csv("flights.csv", sep=c(","), header = TRUE)

head(vuelos)
```

```
##   YEAR MONTH DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER TAIL_NUMBER ORIGIN_AIRPOR
T
## 1 2015     1   1           4      AS            98      N407AS            AN
C
## 2 2015     1   1           4      AA          2336      N3KUAA            LA
X
## 3 2015     1   1           4      US           840      N171US            SF
O
## 4 2015     1   1           4      AA           258      N3HYAA            LA
X
## 5 2015     1   1           4      AS           135      N527AS            SE
A
## 6 2015     1   1           4      DL           806      N3730B            SF
O
##   DESTINATION_AIRPORT SCHEDULED_DEPARTURE DEPARTURE_TIME DEPARTURE_DELAY
## 1                 SEA                   5           2354             -11
## 2                 PBI                  10              2              -8
## 3                 CLT                  20             18              -2
## 4                 MIA                  20             15              -5
## 5                 ANC                  25             24              -1
## 6                 MSP                  25             20              -5
##   TAXI_OUT WHEELS_OFF SCHEDULED_TIME ELAPSED_TIME AIR_TIME DISTANCE WHEELS_O
N
## 1       21         15            205          194      169     1448       40
4
## 2       12         14            280          279      263     2330       73
7
## 3       16         34            286          293      266     2296       80
0
## 4       15         30            285          281      258     2342       74
8
## 5       11         35            235          215      199     1448       25
4
## 6       18         38            217          230      206     1589       60
4
##   TAXI_IN SCHEDULED_ARRIVAL ARRIVAL_TIME ARRIVAL_DELAY DIVERTED CANCELLED
## 1       4              430          408           -22        0         0
## 2       4              750          741            -9        0         0
## 3      11              806          811             5        0         0
## 4       8              805          756            -9        0         0
## 5       5              320          259           -21        0         0
## 6       6              602          610             8        0         0
##   CANCELLATION_REASON AIR_SYSTEM_DELAY SECURITY_DELAY AIRLINE_DELAY
## 1                                  NA             NA            NA
## 2                                  NA             NA            NA
## 3                                  NA             NA            NA
## 4                                  NA             NA            NA
## 5                                  NA             NA            NA
## 6                                  NA             NA            NA
##   LATE_AIRCRAFT_DELAY WEATHER_DELAY
## 1                  NA            NA
## 2                  NA            NA
```

```
## 3                 NA              NA
## 4                 NA              NA
## 5                 NA              NA
## 6                 NA              NA
```

Este conjunto de datos tiene un tamaño demasiado grande para algunas de las operaciones que necesitamos hacer y por este motivo hemos decidido realizar la práctica con un subconjunto del mismo. En el caso necesario, todos los cálculos se podrían repetir con el conjunto completo.

```
#Reducción de la cantidad
set.seed(222)
index <- sample(1:nrow(vuelos), size=0.05*nrow(vuelos))
vuelos_reduc <- vuelos[index,]
str(vuelos_reduc)
```

```
## 'data.frame':    290953 obs. of  31 variables:
##  $ YEAR               : int  2015 2015 2015 2015 2015 2015 2015 2015 2015 20
15 ...
##  $ MONTH              : int  11 4 4 7 2 7 12 2 3 2 ...
##  $ DAY                : int  16 7 1 28 9 23 24 22 8 8 ...
##  $ DAY_OF_WEEK        : int  1 2 3 2 1 4 4 7 7 7 ...
##  $ AIRLINE            : Factor w/ 14 levels "AA","AS","B6",..: 5 14 4 5 14 1
14 10 8 4 ...
##  $ FLIGHT_NUMBER      : int  5084 1023 2182 4330 1963 2148 1915 4636 2950 21
04 ...
##  $ TAIL_NUMBER        : Factor w/ 4898 levels "","7819A","7820L",..: 3370 35
05 4730 149 3706 4748 1306 1795 4640 4580 ...
##  $ ORIGIN_AIRPORT     : Factor w/ 628 levels "10135","10136",..: 327 358 439
504 483 346 344 593 535 523 ...
##  $ DESTINATION_AIRPORT: Factor w/ 629 levels "10135","10136",..: 614 577 328
459 500 490 368 432 573 393 ...
##  $ SCHEDULED_DEPARTURE: int  825 930 540 1545 1055 600 1125 1956 1145 1935
...
##  $ DEPARTURE_TIME     : int  819 943 538 1559 1105 553 1124 2002 1242 1932
...
##  $ DEPARTURE_DELAY    : int  -6 13 -2 14 10 -7 -1 6 57 -3 ...
##  $ TAXI_OUT           : int  14 10 12 21 7 15 18 18 15 37 ...
##  $ WHEELS_OFF         : int  833 953 550 1620 1112 608 1142 2020 1257 2009
...
##  $ SCHEDULED_TIME     : int  128 240 57 108 190 74 80 63 68 137 ...
##  $ ELAPSED_TIME       : int  129 214 47 105 166 68 86 57 68 140 ...
##  $ AIR_TIME           : int  111 201 28 75 153 47 64 35 51 95 ...
##  $ DISTANCE           : int  674 1407 153 468 1363 184 439 216 268 680 ...
##  $ WHEELS_ON          : int  924 1214 618 1735 1545 655 1346 2055 1348 2044
...
##  $ TAXI_IN            : int  4 3 7 9 6 6 4 4 2 8 ...
##  $ SCHEDULED_ARRIVAL  : int  933 1230 637 1733 1605 714 1345 2059 1253 2052
...
##  $ ARRIVAL_TIME       : int  928 1217 625 1744 1551 701 1350 2059 1350 2052
...
##  $ ARRIVAL_DELAY      : int  -5 -13 -12 11 -14 -13 5 0 57 0 ...
##  $ DIVERTED           : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ CANCELLED          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ CANCELLATION_REASON: Factor w/ 5 levels "","A","B","C",..: 1 1 1 1 1 1 1
1 1 1 ...
##  $ AIR_SYSTEM_DELAY   : int  NA NA NA NA NA NA NA NA 13 NA ...
##  $ SECURITY_DELAY     : int  NA NA NA NA NA NA NA NA 0 NA ...
##  $ AIRLINE_DELAY      : int  NA NA NA NA NA NA NA NA 0 NA ...
##  $ LATE_AIRCRAFT_DELAY: int  NA NA NA NA NA NA NA NA 44 NA ...
##  $ WEATHER_DELAY      : int  NA NA NA NA NA NA NA NA 0 NA ...
```

```
summary(vuelos_reduc)
```

```
##       YEAR          MONTH            DAY         DAY_OF_WEEK       AIRLINE
##  Min.   :2015   Min.   : 1.00   Min.   : 1.00   Min.   :1.000   WN     :6316
9
##  1st Qu.:2015   1st Qu.: 4.00   1st Qu.: 8.00   1st Qu.:2.000   DL     :4365
0
##  Median :2015   Median : 7.00   Median :16.00   Median :4.000   AA     :3634
7
##  Mean   :2015   Mean   : 6.53   Mean   :15.72   Mean   :3.925   OO     :2937
3
##  3rd Qu.:2015   3rd Qu.: 9.00   3rd Qu.:23.00   3rd Qu.:6.000   EV     :2861
7
##  Max.   :2015   Max.   :12.00   Max.   :31.00   Max.   :7.000   UA     :2589
3
##                                                                 (Other):6390
4
##  FLIGHT_NUMBER   TAIL_NUMBER     ORIGIN_AIRPORT   DESTINATION_AIRPORT
##  Min.   :   1          :   709   ATL    : 17374   ATL    : 17274
##  1st Qu.: 732   N480HA :   193   ORD    : 14214   ORD    : 13989
##  Median :1688   N486HA :   193   DFW    : 11971   DFW    : 12169
##  Mean   :2170   N478HA :   185   DEN    :  9663   LAX    :  9823
##  3rd Qu.:3228   N483HA :   185   LAX    :  9498   DEN    :  9581
##  Max.   :7438   N488HA :   184   IAH    :  7407   PHX    :  7424
##                 (Other):289304   (Other):220826   (Other):220693
##  SCHEDULED_DEPARTURE DEPARTURE_TIME DEPARTURE_DELAY      TAXI_OUT
##  Min.   :   3        Min.   :   1   Min.   : -68.00   Min.   :  1.0
##  1st Qu.: 916        1st Qu.: 920   1st Qu.:  -5.00   1st Qu.: 11.0
##  Median :1325        Median :1329   Median :  -2.00   Median : 14.0
##  Mean   :1328        Mean   :1334   Mean   :   9.41   Mean   : 16.1
##  3rd Qu.:1730        3rd Qu.:1739   3rd Qu.:   7.00   3rd Qu.: 19.0
##  Max.   :2359        Max.   :2400   Max.   :1380.00   Max.   :167.0
##                      NA's   :4309   NA's   :4309      NA's   :4465
##    WHEELS_OFF   SCHEDULED_TIME   ELAPSED_TIME      AIR_TIME
##  Min.   :   1   Min.   : 20.0   Min.   : 17.0   Min.   :  8.0
##  1st Qu.: 935   1st Qu.: 85.0   1st Qu.: 82.0   1st Qu.: 60.0
##  Median :1342   Median :123.0   Median :118.0   Median : 94.0
##  Mean   :1356   Mean   :141.6   Mean   :136.9   Mean   :113.4
##  3rd Qu.:1753   3rd Qu.:173.0   3rd Qu.:169.0   3rd Qu.:144.0
##  Max.   :2400   Max.   :705.0   Max.   :685.0   Max.   :661.0
##  NA's   :4465   NA's   :1       NA's   :5303    NA's   :5303
##     DISTANCE        WHEELS_ON       TAXI_IN        SCHEDULED_ARRIVAL
##  Min.   :  31.0   Min.   :   1   Min.   :  1.000   Min.   :   1
##  1st Qu.: 372.0   1st Qu.:1054   1st Qu.:  4.000   1st Qu.:1110
##  Median : 647.0   Median :1507   Median :  6.000   Median :1519
##  Mean   : 821.5   Mean   :1470   Mean   :  7.424   Mean   :1493
##  3rd Qu.:1062.0   3rd Qu.:1911   3rd Qu.:  9.000   3rd Qu.:1917
##  Max.   :4983.0   Max.   :2400   Max.   :183.000   Max.   :2359
##                   NA's   :4650   NA's   :4650
##   ARRIVAL_TIME  ARRIVAL_DELAY         DIVERTED          CANCELLED
##  Min.   :   1   Min.   : -87.000   Min.   :0.000000   Min.   :0.00000
##  1st Qu.:1058   1st Qu.: -13.000   1st Qu.:0.000000   1st Qu.:0.00000
##  Median :1511   Median :  -5.000   Median :0.000000   Median :0.00000
##  Mean   :1475   Mean   :   4.482   Mean   :0.002719   Mean   :0.01551
```

```
##  3rd Qu.:1916   3rd Qu.:   8.000   3rd Qu.:0.000000   3rd Qu.:0.00000
##  Max.   :2400   Max.   :1384.000   Max.   :1.000000   Max.   :1.00000
##  NA's   :4650   NA's   :5303
##  CANCELLATION_REASON AIR_SYSTEM_DELAY SECURITY_DELAY   AIRLINE_DELAY
##   :286441            Min.   :  0.00   Min.   : 0.00    Min.   :   0.00
##  A:  1313            1st Qu.:  0.00   1st Qu.: 0.00    1st Qu.:   0.00
##  B:  2485            Median :  2.00   Median : 0.00    Median :   2.00
##  C:   713            Mean   : 13.76   Mean   : 0.07    Mean   :  18.86
##  D:     1            3rd Qu.: 18.00   3rd Qu.: 0.00    3rd Qu.:  19.00
##                      Max.   :628.00   Max.   :99.00    Max.   :1380.00
##                      NA's   :237819   NA's   :237819   NA's   :237819
##  LATE_AIRCRAFT_DELAY WEATHER_DELAY
##  Min.   :   0.00     Min.   :  0.00
##  1st Qu.:   0.00     1st Qu.:  0.00
##  Median :   3.00     Median :  0.00
##  Mean   :  23.73     Mean   :  2.89
##  3rd Qu.:  29.00     3rd Qu.:  0.00
##  Max.   :1294.00     Max.   :896.00
##  NA's   :237819      NA's   :237819
```

```
length(vuelos_reduc$YEAR)
```

```
## [1] 290953
```

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

Vamos a cargar los datos de localización de los aeropuertos y haremos un merge de los datos de los aeropuertos con el dataset que tenemos para, entre otras cosas, hacer una visualización en un mapa. Renombramos las columnas para dejar la misma nomenclatura en aquellas que queremos unir

```
airports <- read.csv("datasets_810_1496_airports.csv", header=TRUE)
head(airports)
```

```
##   IATA_CODE                              AIRPORT        CITY STATE COUNTRY
## 1       ABE Lehigh Valley International Airport   Allentown    PA     USA
## 2       ABI            Abilene Regional Airport     Abilene    TX     USA
## 3       ABQ   Albuquerque International Sunport Albuquerque    NM     USA
## 4       ABR           Aberdeen Regional Airport    Aberdeen    SD     USA
## 5       ABY  Southwest Georgia Regional Airport      Albany    GA     USA
## 6       ACK          Nantucket Memorial Airport   Nantucket    MA     USA
##   LATITUDE  LONGITUDE
## 1 40.65236  -75.44040
## 2 32.41132  -99.68190
## 3 35.04022 -106.60919
## 4 45.44906  -98.42183
## 5 31.53552  -84.19447
## 6 41.25305  -70.06018
```

```
colnames(vuelos_reduc)[8]  <- "ORIGIN_CODE"
colnames(airports) <- c("ORIGIN_CODE","ORIGIN_AIRPORT", "ORIGIN_CITY", "ORIGIN_
STATE", "ORIGIN_COUNTRY", "ORIGIN_LATITUDE", "ORIGIN_LONGITUDE" )

flight_airports <- left_join(vuelos_reduc,airports,by="ORIGIN_CODE")
```

```
## Warning: Column `ORIGIN_CODE` joining factors with different levels, coercin
g to
## character vector
```

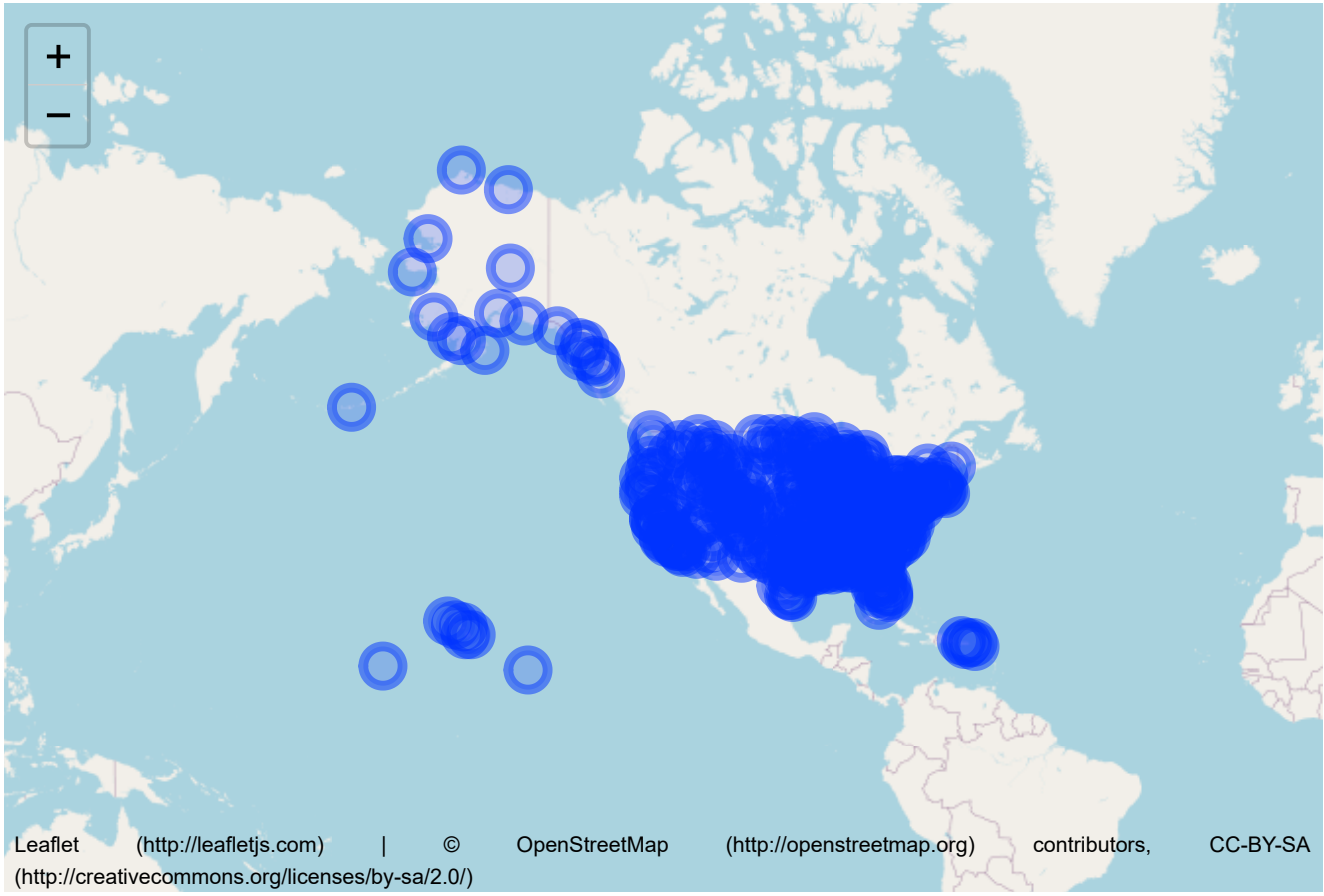
Nos da error, porque como nuestro dataset es un subconjunto de los datos iniciales, puede que no estén todos los aeropuertos, por lo que tenemos que igualar los niveles de los dos campos tipo factor.

```
combined <- sort(union(levels(vuelos_reduc$ORIGIN_CODE), levels(airports$ORIGIN
_CODE)))
flight_airports <- left_join(mutate(vuelos_reduc, ORIGIN_CODE=factor(ORIGIN_COD
E, levels=combined)),
                             mutate(airports, ORIGIN_CODE=factor(ORIGIN_CODE, l
evels=combined)))
```

```
## Joining, by = "ORIGIN_CODE"
```

```
#visualización del volumen de vuelos de cada aeropuerto
longitude <- unique(flight_airports$ORIGIN_LONGITUDE)
latitude <- unique(flight_airports$ORIGIN_LATITUDE)
df = data.frame(Lat = latitude, Long = longitude)
leaflet(df) %>% addTiles() %>% addCircleMarkers() #map visualization
```

```
## Warning in validateCoords(lng, lat, funcName): Data contains 1 rows with eit
her
## missing or invalid lat/lon values and will be ignored
```



A partir de aquí, vamos a identificar los tipos de variables y a quedarnos con los datos que nos interesarán para realizar nuestro estudio.

```
res <- sapply(vuelos_reduc,class)
kable(data.frame(variables=names(res),clase=as.vector(res)),
      caption = "asignación de clase de objeto R a cada variable")
```

asignación de clase de objeto R a cada variable

variables	clase
YEAR	integer
MONTH	integer
DAY	integer
DAY_OF_WEEK	integer
AIRLINE	factor
FLIGHT_NUMBER	integer
TAIL_NUMBER	factor
ORIGIN_CODE	factor
DESTINATION_AIRPORT	factor
SCHEDULED_DEPARTURE	integer
DEPARTURE_TIME	integer
DEPARTURE_DELAY	integer
TAXI_OUT	integer
WHEELS_OFF	integer
SCHEDULED_TIME	integer
ELAPSED_TIME	integer
AIR_TIME	integer

variables	clase
DISTANCE	integer
WHEELS_ON	integer
TAXI_IN	integer
SCHEDULED_ARRIVAL	integer
ARRIVAL_TIME	integer
ARRIVAL_DELAY	integer
DIVERTED	integer
CANCELLED	integer
CANCELLATION_REASON	factor
AIR_SYSTEM_DELAY	integer
SECURITY_DELAY	integer
AIRLINE_DELAY	integer
LATE_AIRCRAFT_DELAY	integer
WEATHER_DELAY	integer

```
vuelos[1:4] <- lapply(vuelos[1:4], as.numeric)
vuelos[6] <- lapply(vuelos[6], as.numeric)
vuelos[10:25] <- lapply(vuelos[10:25], as.numeric)
vuelos[27:31] <- lapply(vuelos[27:31], as.numeric)
res <- sapply(vuelos,class)
tabla_datos <- data.frame(variables=names(res),clase=as.vector(res))
tabla_datos %>% knitr::kable("html")  %>% kable_styling(position='center', font
_size=12, fixed_thead=list(enabled=T))
```

variables	clase
YEAR	numeric
MONTH	numeric
DAY	numeric
DAY_OF_WEEK	numeric
AIRLINE	factor
FLIGHT_NUMBER	numeric
TAIL_NUMBER	factor
ORIGIN_AIRPORT	factor
DESTINATION_AIRPORT	factor
SCHEDULED_DEPARTURE	numeric
DEPARTURE_TIME	numeric
DEPARTURE_DELAY	numeric
TAXI_OUT	numeric
WHEELS_OFF	numeric
SCHEDULED_TIME	numeric

variables	clase
ELAPSED_TIME	numeric
AIR_TIME	numeric
DISTANCE	numeric
WHEELS_ON	numeric
TAXI_IN	numeric
SCHEDULED_ARRIVAL	numeric
ARRIVAL_TIME	numeric
ARRIVAL_DELAY	numeric
DIVERTED	numeric
CANCELLED	numeric
CANCELLATION_REASON	factor
AIR_SYSTEM_DELAY	numeric
SECURITY_DELAY	numeric
AIRLINE_DELAY	numeric
LATE_AIRCRAFT_DELAY	numeric
WEATHER_DELAY	numeric

```
str(vuelos)
```

```
## 'data.frame':    5819079 obs. of  31 variables:
##  $ YEAR               : num  2015 2015 2015 2015 2015 ...
##  $ MONTH              : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ DAY                : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ DAY_OF_WEEK        : num  4 4 4 4 4 4 4 4 4 4 ...
##  $ AIRLINE             : Factor w/ 14 levels "AA","AS","B6",..: 2 1 12 1 2 4
9 12 1 4 ...
##  $ FLIGHT_NUMBER      : num  98 2336 840 258 135 ...
##  $ TAIL_NUMBER        : Factor w/ 4898 levels "","7819A","7820L",..: 1624 15
58 423 1518 2133 1143 2767 2412 1563 3936 ...
##  $ ORIGIN_AIRPORT     : Factor w/ 628 levels "10135","10136",..: 324 483 585
483 584 585 481 483 585 481 ...
##  $ DESTINATION_AIRPORT: Factor w/ 629 levels "10135","10136",..: 585 543 374
511 325 524 524 374 394 328 ...
##  $ SCHEDULED_DEPARTURE: num  5 10 20 20 25 25 25 30 30 30 ...
##  $ DEPARTURE_TIME     : num  2354 2 18 15 24 ...
##  $ DEPARTURE_DELAY    : num  -11 -8 -2 -5 -1 -5 -6 14 -11 3 ...
##  $ TAXI_OUT           : num  21 12 16 15 11 18 11 13 17 12 ...
##  $ WHEELS_OFF         : num  15 14 34 30 35 38 30 57 36 45 ...
##  $ SCHEDULED_TIME     : num  205 280 286 285 235 217 181 273 195 221 ...
##  $ ELAPSED_TIME       : num  194 279 293 281 215 230 170 249 193 203 ...
##  $ AIR_TIME           : num  169 263 266 258 199 206 154 228 173 186 ...
##  $ DISTANCE           : num  1448 2330 2296 2342 1448 ...
##  $ WHEELS_ON          : num  404 737 800 748 254 604 504 745 529 651 ...
##  $ TAXI_IN            : num  4 4 11 8 5 6 5 8 3 5 ...
##  $ SCHEDULED_ARRIVAL  : num  430 750 806 805 320 602 526 803 545 711 ...
##  $ ARRIVAL_TIME       : num  408 741 811 756 259 610 509 753 532 656 ...
##  $ ARRIVAL_DELAY      : num  -22 -9 5 -9 -21 8 -17 -10 -13 -15 ...
##  $ DIVERTED           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ CANCELLED          : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ CANCELLATION_REASON: Factor w/ 5 levels "","A","B","C",..: 1 1 1 1 1 1 1
1 1 1 ...
##  $ AIR_SYSTEM_DELAY   : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ SECURITY_DELAY     : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ AIRLINE_DELAY      : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ LATE_AIRCRAFT_DELAY: num  NA NA NA NA NA NA NA NA NA NA ...
##  $ WEATHER_DELAY      : num  NA NA NA NA NA NA NA NA NA NA ...
```

De todas las variables cargadas, de momento nos vamos a quedar con las siguientes:

MONTH
DAY
DAY_OF_WEEK
AIRLINE
ORIGIN_CODE
DESTINATION_AIRPORT
SCHEDULED_DEPARTURE
DEPARTURE_TIME
DEPARTURE_DELAY
SCHEDULED_TIME
ELAPSED_TIME
AIR_TIME
DISTANCE

SCHEDULED_ARRIVAL
ARRIVAL_TIME
ARRIVAL_DELAY

```
vuelos_reduc <- dplyr::select(vuelos_reduc, -"YEAR",-"TAIL_NUMBER",-"AIR_SYSTEM
_DELAY",-"SECURITY_DELAY",
                              -"AIRLINE_DELAY",-"LATE_AIRCRAFT_DELAY",-"WEATHER
_DELAY",
                               -"TAXI_OUT",-"TAXI_IN",-"WHEELS_OFF", -"WHEELS_O
N",-"DIVERTED",
                              -"CANCELLED",-"CANCELLATION_REASON")

str(vuelos_reduc)
```

```
## 'data.frame':    290953 obs. of  17 variables:
##  $ MONTH              : int  11 4 4 7 2 7 12 2 3 2 ...
##  $ DAY                : int  16 7 1 28 9 23 24 22 8 8 ...
##  $ DAY_OF_WEEK        : int  1 2 3 2 1 4 4 7 7 7 ...
##  $ AIRLINE            : Factor w/ 14 levels "AA","AS","B6",..: 5 14 4 5 14 1
14 10 8 4 ...
##  $ FLIGHT_NUMBER      : int  5084 1023 2182 4330 1963 2148 1915 4636 2950 21
04 ...
##  $ ORIGIN_CODE        : Factor w/ 628 levels "10135","10136",..: 327 358 439
504 483 346 344 593 535 523 ...
##  $ DESTINATION_AIRPORT: Factor w/ 629 levels "10135","10136",..: 614 577 328
459 500 490 368 432 573 393 ...
##  $ SCHEDULED_DEPARTURE: int  825 930 540 1545 1055 600 1125 1956 1145 1935
...
##  $ DEPARTURE_TIME     : int  819 943 538 1559 1105 553 1124 2002 1242 1932
...
##  $ DEPARTURE_DELAY    : int  -6 13 -2 14 10 -7 -1 6 57 -3 ...
##  $ SCHEDULED_TIME     : int  128 240 57 108 190 74 80 63 68 137 ...
##  $ ELAPSED_TIME       : int  129 214 47 105 166 68 86 57 68 140 ...
##  $ AIR_TIME           : int  111 201 28 75 153 47 64 35 51 95 ...
##  $ DISTANCE           : int  674 1407 153 468 1363 184 439 216 268 680 ...
##  $ SCHEDULED_ARRIVAL  : int  933 1230 637 1733 1605 714 1345 2059 1253 2052
...
##  $ ARRIVAL_TIME       : int  928 1217 625 1744 1551 701 1350 2059 1350 2052
...
##  $ ARRIVAL_DELAY      : int  -5 -13 -12 11 -14 -13 5 0 57 0 ...
```

3. Limpieza de los datos

3.1 Elementos vacíos

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

En este apartado, vamos a comprobar los valores que contienen nuestras variables para detectar si hay errores en los mismo, si tenemos elementos vacios o ceros o si hay datos fuera de los valores esperados, por ejemplo en las horas, días o meses.

```
#Comprobamos valore nulos o valores perdidos
sapply(vuelos_reduc, function(x) sum(is.na(x)))
```

```
##               MONTH                 DAY         DAY_OF_WEEK             AIRL
INE
##                   0                   0                   0
0
##       FLIGHT_NUMBER         ORIGIN_CODE DESTINATION_AIRPORT SCHEDULED_DEPART
URE
##                   0                   0                   0
0
##      DEPARTURE_TIME     DEPARTURE_DELAY      SCHEDULED_TIME        ELAPSED_T
IME
##                4309                4309                   1                5
303
##            AIR_TIME            DISTANCE   SCHEDULED_ARRIVAL        ARRIVAL_T
IME
##                5303                   0                   0                4
650
##       ARRIVAL_DELAY
##                5303
```

```
#Otra forma de sacar los valores nulos
colSums(is.na(vuelos_reduc))
```

```
##               MONTH                 DAY         DAY_OF_WEEK             AIRL
INE
##                   0                   0                   0
0
##       FLIGHT_NUMBER         ORIGIN_CODE DESTINATION_AIRPORT SCHEDULED_DEPART
URE
##                   0                   0                   0
0
##      DEPARTURE_TIME     DEPARTURE_DELAY      SCHEDULED_TIME        ELAPSED_T
IME
##                4309                4309                   1                5
303
##            AIR_TIME            DISTANCE   SCHEDULED_ARRIVAL        ARRIVAL_T
IME
##                5303                   0                   0                4
650
##       ARRIVAL_DELAY
##                5303
```

De los valores nulos que hemos identificado, hacemos una revisión para conocer el motivo de esos valores. Comprobamos que en el caso de los valores nulos en las variables DEPARTURE_DELAY y ARRIVAL_DELAY, se trata de aquellos vuelos que han sido cancelados o desviados, por lo que, como

nuestro estudio va a estar basado en los vuelos completados, eliminaremos todos estos valores nulos.

```
#Comprobamos los valores nulos de la columna DEPARTURE_DELAY y ARRIVAL_DELAY
head(vuelos %>% filter(is.na(vuelos$DEPARTURE_DELAY)))
```

```
##   YEAR MONTH DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER TAIL_NUMBER ORIGIN_AIRPOR
T
## 1 2015     1   1           4      AS           136      N431AS            AN
C
## 2 2015     1   1           4      AA          2459      N3BDAA            PH
X
## 3 2015     1   1           4      OO          5254      N746SK            MA
F
## 4 2015     1   1           4      MQ          2859      N660MQ            SG
F
## 5 2015     1   1           4      OO          5460      N583SW            RD
D
## 6 2015     1   1           4      MQ          2926      N932MQ            CH
S
##   DESTINATION_AIRPORT SCHEDULED_DEPARTURE DEPARTURE_TIME DEPARTURE_DELAY
## 1                 SEA                 135             NA              NA
## 2                 DFW                 200             NA              NA
## 3                 IAH                 510             NA              NA
## 4                 DFW                 525             NA              NA
## 5                 SFO                 530             NA              NA
## 6                 DFW                 545             NA              NA
##   TAXI_OUT WHEELS_OFF SCHEDULED_TIME ELAPSED_TIME AIR_TIME DISTANCE WHEELS_O
N
## 1       NA         NA            205           NA       NA     1448        N
A
## 2       NA         NA            120           NA       NA      868        N
A
## 3       NA         NA             87           NA       NA      429        N
A
## 4       NA         NA             95           NA       NA      364        N
A
## 5       NA         NA             90           NA       NA      199        N
A
## 6       NA         NA            190           NA       NA      987        N
A
##   TAXI_IN SCHEDULED_ARRIVAL ARRIVAL_TIME ARRIVAL_DELAY DIVERTED CANCELLED
## 1      NA              600           NA            NA        0         1
## 2      NA              500           NA            NA        0         1
## 3      NA              637           NA            NA        0         1
## 4      NA              700           NA            NA        0         1
## 5      NA              700           NA            NA        0         1
## 6      NA              755           NA            NA        0         1
##   CANCELLATION_REASON AIR_SYSTEM_DELAY SECURITY_DELAY AIRLINE_DELAY
## 1                   A               NA             NA            NA
## 2                   B               NA             NA            NA
## 3                   B               NA             NA            NA
## 4                   B               NA             NA            NA
## 5                   A               NA             NA            NA
## 6                   B               NA             NA            NA
##   LATE_AIRCRAFT_DELAY WEATHER_DELAY
## 1                  NA            NA
## 2                  NA            NA
```

```
## 3                NA              NA
## 4                NA              NA
## 5                NA              NA
## 6                NA              NA
```

```
head(vuelos %>% filter(is.na(vuelos$ARRIVAL_DELAY)))
```

```
##   YEAR MONTH DAY DAY_OF_WEEK AIRLINE FLIGHT_NUMBER TAIL_NUMBER ORIGIN_AIRPOR
T
## 1 2015     1   1           4      AS           136      N431AS            AN
C
## 2 2015     1   1           4      AA          2459      N3BDAA            PH
X
## 3 2015     1   1           4      OO          5254      N746SK            MA
F
## 4 2015     1   1           4      MQ          2859      N660MQ            SG
F
## 5 2015     1   1           4      OO          5460      N583SW            RD
D
## 6 2015     1   1           4      MQ          2926      N932MQ            CH
S
##   DESTINATION_AIRPORT SCHEDULED_DEPARTURE DEPARTURE_TIME DEPARTURE_DELAY
## 1                 SEA                 135             NA              NA
## 2                 DFW                 200             NA              NA
## 3                 IAH                 510             NA              NA
## 4                 DFW                 525             NA              NA
## 5                 SFO                 530             NA              NA
## 6                 DFW                 545             NA              NA
##   TAXI_OUT WHEELS_OFF SCHEDULED_TIME ELAPSED_TIME AIR_TIME DISTANCE WHEELS_O
N
## 1       NA         NA            205           NA       NA     1448        N
A
## 2       NA         NA            120           NA       NA      868        N
A
## 3       NA         NA             87           NA       NA      429        N
A
## 4       NA         NA             95           NA       NA      364        N
A
## 5       NA         NA             90           NA       NA      199        N
A
## 6       NA         NA            190           NA       NA      987        N
A
##   TAXI_IN SCHEDULED_ARRIVAL ARRIVAL_TIME ARRIVAL_DELAY DIVERTED CANCELLED
## 1      NA              600           NA            NA        0         1
## 2      NA              500           NA            NA        0         1
## 3      NA              637           NA            NA        0         1
## 4      NA              700           NA            NA        0         1
## 5      NA              700           NA            NA        0         1
## 6      NA              755           NA            NA        0         1
##   CANCELLATION_REASON AIR_SYSTEM_DELAY SECURITY_DELAY AIRLINE_DELAY
## 1                   A               NA             NA            NA
## 2                   B               NA             NA            NA
## 3                   B               NA             NA            NA
## 4                   B               NA             NA            NA
## 5                   A               NA             NA            NA
## 6                   B               NA             NA            NA
##   LATE_AIRCRAFT_DELAY WEATHER_DELAY
## 1                  NA            NA
## 2                  NA            NA
```

```
## 3                   NA             NA
## 4                   NA             NA
## 5                   NA             NA
## 6                   NA             NA
```

```
#De momento como lo que queremos es trabajar con los vuelos retrasados vamos a
 eliminar los valores nulos de
#estas variables
vuelos_reduc <- vuelos_reduc[!is.na(vuelos_reduc$DEPARTURE_DELAY),]
vuelos_reduc <- vuelos_reduc[!is.na(vuelos_reduc$ARRIVAL_DELAY),]


#Comprobamos que ya no quedan valores nulos
colSums(is.na(vuelos_reduc))
```

```
##               MONTH                 DAY         DAY_OF_WEEK               AIRL
INE
##                   0                   0                   0
0
##       FLIGHT_NUMBER         ORIGIN_CODE DESTINATION_AIRPORT SCHEDULED_DEPART
URE
##                   0                   0                   0
0
##      DEPARTURE_TIME     DEPARTURE_DELAY      SCHEDULED_TIME        ELAPSED_T
IME
##                   0                   0                   0
0
##            AIR_TIME            DISTANCE   SCHEDULED_ARRIVAL        ARRIVAL_T
IME
##                   0                   0                   0
0
##       ARRIVAL_DELAY
##                   0
```

```
summary(vuelos_reduc)
```

```
##      MONTH              DAY          DAY_OF_WEEK      AIRLINE      FLIGHT_NUMB
ER
##  Min.   : 1.000   Min.   : 1.00   Min.   :1.00   WN     :62175   Min.   :
1
##  1st Qu.: 4.000   1st Qu.: 8.00   1st Qu.:2.00   DL     :43369   1st Qu.: 73
0
##  Median : 7.000   Median :16.00   Median :4.00   AA     :35655   Median :168
0
##  Mean   : 6.554   Mean   :15.73   Mean   :3.93   OO     :28784   Mean   :216
2
##  3rd Qu.: 9.000   3rd Qu.:23.00   3rd Qu.:6.00   EV     :27742   3rd Qu.:320
9
##  Max.   :12.000   Max.   :31.00   Max.   :7.00   UA     :25499   Max.   :743
8
##                                                  (Other):62426
##   ORIGIN_CODE     DESTINATION_AIRPORT SCHEDULED_DEPARTURE DEPARTURE_TIME
##  ATL    : 17189   ATL    : 17090      Min.   :   3        Min.   :   1
##  ORD    : 13742   ORD    : 13482      1st Qu.: 916        1st Qu.: 920
##  DFW    : 11625   DFW    : 11753      Median :1323        Median :1329
##  DEN    :  9532   LAX    :  9710      Mean   :1327        Mean   :1334
##  LAX    :  9354   DEN    :  9433      3rd Qu.:1730        3rd Qu.:1739
##  IAH    :  7284   PHX    :  7341      Max.   :2359        Max.   :2400
##  (Other):216924   (Other):216841
##  DEPARTURE_DELAY    SCHEDULED_TIME   ELAPSED_TIME      AIR_TIME
##  Min.   : -68.000   Min.   : 20.0   Min.   : 17.0   Min.   :  8.0
##  1st Qu.:  -5.000   1st Qu.: 85.0   1st Qu.: 82.0   1st Qu.: 60.0
##  Median :  -2.000   Median :123.0   Median :118.0   Median : 94.0
##  Mean   :   9.329   Mean   :141.8   Mean   :136.9   Mean   :113.4
##  3rd Qu.:   7.000   3rd Qu.:174.0   3rd Qu.:169.0   3rd Qu.:144.0
##  Max.   :1380.000   Max.   :705.0   Max.   :685.0   Max.   :661.0
##
##     DISTANCE      SCHEDULED_ARRIVAL  ARRIVAL_TIME  ARRIVAL_DELAY
##  Min.   :  31.0   Min.   :   1      Min.   :   1   Min.   : -87.000
##  1st Qu.: 373.0   1st Qu.:1110      1st Qu.:1058   1st Qu.: -13.000
##  Median : 649.0   Median :1518      Median :1511   Median :  -5.000
##  Mean   : 823.6   Mean   :1492      Mean   :1475   Mean   :   4.482
##  3rd Qu.:1065.0   3rd Qu.:1916      3rd Qu.:1916   3rd Qu.:   8.000
##  Max.   :4983.0   Max.   :2359      Max.   :2400   Max.   :1384.000
##
```

Comprobamos los datos de meses y días, por ver que no hay valores extraños. También revisaremos que no hay distancias ni tiempos horarios negativos.

```
#Comprobamos si hay valores extraños en las variables día, día de la semana y m
es
month_wrong <- which(vuelos_reduc$MONTH > 12 | vuelos_reduc$MONTH < 1)
month_wrong
```

```
## integer(0)
```

```
day_wrong <- which(vuelos_reduc$DAY > 31 | vuelos_reduc$DAY < 1)
day_wrong
```

```
## integer(0)
```

```
day_week_wrong <- which(vuelos_reduc$DAY_OF_WEEK > 7 | vuelos_reduc$DAY_OF_WEEK
<1)
day_week_wrong
```

```
## integer(0)
```

```
str(vuelos_reduc)
```

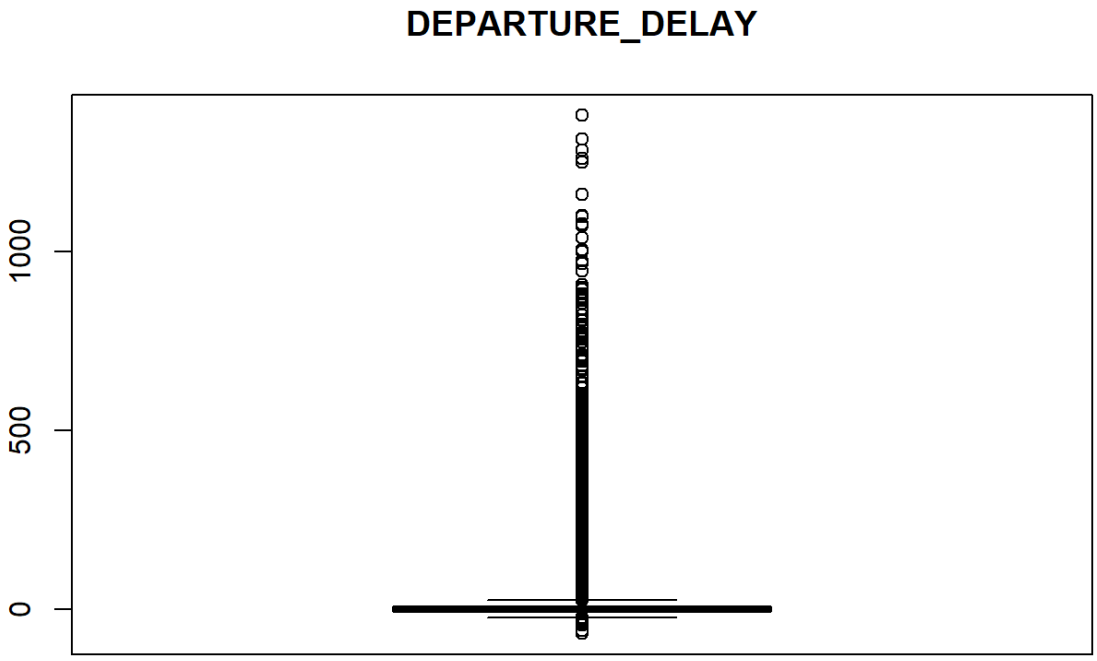
```
## 'data.frame':    285650 obs. of  17 variables:
##  $ MONTH              : int  11 4 4 7 2 7 12 2 3 2 ...
##  $ DAY                : int  16 7 1 28 9 23 24 22 8 8 ...
##  $ DAY_OF_WEEK        : int  1 2 3 2 1 4 4 7 7 7 ...
##  $ AIRLINE            : Factor w/ 14 levels "AA","AS","B6",..: 5 14 4 5 14 1
14 10 8 4 ...
##  $ FLIGHT_NUMBER      : int  5084 1023 2182 4330 1963 2148 1915 4636 2950 21
04 ...
##  $ ORIGIN_CODE        : Factor w/ 628 levels "10135","10136",..: 327 358 439
504 483 346 344 593 535 523 ...
##  $ DESTINATION_AIRPORT: Factor w/ 629 levels "10135","10136",..: 614 577 328
459 500 490 368 432 573 393 ...
##  $ SCHEDULED_DEPARTURE: int  825 930 540 1545 1055 600 1125 1956 1145 1935
...
##  $ DEPARTURE_TIME     : int  819 943 538 1559 1105 553 1124 2002 1242 1932
...
##  $ DEPARTURE_DELAY    : int  -6 13 -2 14 10 -7 -1 6 57 -3 ...
##  $ SCHEDULED_TIME     : int  128 240 57 108 190 74 80 63 68 137 ...
##  $ ELAPSED_TIME       : int  129 214 47 105 166 68 86 57 68 140 ...
##  $ AIR_TIME           : int  111 201 28 75 153 47 64 35 51 95 ...
##  $ DISTANCE           : int  674 1407 153 468 1363 184 439 216 268 680 ...
##  $ SCHEDULED_ARRIVAL  : int  933 1230 637 1733 1605 714 1345 2059 1253 2052
...
##  $ ARRIVAL_TIME       : int  928 1217 625 1744 1551 701 1350 2059 1350 2052
...
##  $ ARRIVAL_DELAY      : int  -5 -13 -12 11 -14 -13 5 0 57 0 ...
```

Comprobamos que los valores de distancias y horas en base a los valores mínimos y máximos son correctos. En el caso de los aeropuertos, conocemos que hay aeropuertos que tienen diferente nomenclatura para un mismo aeropuerto, de momento no vamos a realizar ninguna modificación sobre este dato, supondremos que son aeropuertos diferentes.

3.2 Identificación y tratamiento de valores extremos

Vamos a identificar los posibles valores extermos que tenemos en las variables de tiempo y distancia. En base a los resultados decidiremos que acciones tomar con estas variables.

```
boxplot(vuelos_reduc$DEPARTURE_DELAY, main="DEPARTURE_DELAY")
```




```
table(vuelos_reduc$DEPARTURE_DELAY)
```

```
##
##   -68   -61   -44   -39   -38   -37   -35   -34   -33   -31   -30   -29   -2
8
##     1     1     1     1     1     1     1     3     1     3     2     3
8
##   -27   -26   -25   -24   -23   -22   -21   -20   -19   -18   -17   -16   -1
5
##     3     9     9    18    29    40    39    62    92   164   187   315   52
5
##   -14   -13   -12   -11   -10    -9    -8    -7    -6    -5    -4    -3    -
2
##   746  1100  1658  2464  5224  5971  8700 12221 16301 22017 22123 22739 2163
2
##    -1     0     1     2     3     4     5     6     7     8     9    10    1
1
## 19241 16350  7970  6060  5175  4630  4245  3796  3392  3170  2947  2794  250
2
##    12    13    14    15    16    17    18    19    20    21    22    23    2
4
##  2310  2230  2000  1991  1709  1784  1587  1529  1474  1344  1280  1226  109
1
##    25    26    27    28    29    30    31    32    33    34    35    36    3
7
##  1206  1012  1004   930   909   912   809   799   783   719   789   679   66
4
##    38    39    40    41    42    43    44    45    46    47    48    49    5
0
##   602   572   546   543   544   507   493   493   512   452   413   424   46
8
##    51    52    53    54    55    56    57    58    59    60    61    62    6
3
##   403   413   357   373   352   327   339   325   337   328   319   303   29
3
##    64    65    66    67    68    69    70    71    72    73    74    75    7
6
##   276   298   266   273   283   256   246   219   216   234   223   193   20
0
##    77    78    79    80    81    82    83    84    85    86    87    88    8
9
##   199   197   207   183   186   183   174   192   160   164   155   173   15
7
##    90    91    92    93    94    95    96    97    98    99   100   101   10
2
##   155   148   134   149   131   149   146   138   126   116   156   130   12
3
##   103   104   105   106   107   108   109   110   111   112   113   114   11
5
##   147   128   124   123   125   130   100   121    93    97   118   100    8
8
##   116   117   118   119   120   121   122   123   124   125   126   127   12
8
##    91    99    87    98    92    89    82    95    85    85    82    87    8
```

```
1
##   129   130   131   132   133   134   135   136   137   138   139   140   14
1
##    79    74    76    66    87    81    68    78    64    66    57    74    5
9
##   142   143   144   145   146   147   148   149   150   151   152   153   15
4
##    66    54    54    58    69    64    69    62    53    53    51    47    4
8
##   155   156   157   158   159   160   161   162   163   164   165   166   16
7
##    50    37    41    52    51    50    39    46    50    30    37    41    5
5
##   168   169   170   171   172   173   174   175   176   177   178   179   18
0
##    36    40    46    50    47    45    37    40    42    41    37    36    3
9
##   181   182   183   184   185   186   187   188   189   190   191   192   19
3
##    35    35    32    38    29    33    30    27    24    31    19    17    2
4
##   194   195   196   197   198   199   200   201   202   203   204   205   20
6
##    34    37    21    25    20    21    27    24    20    18    20    21    2
4
##   207   208   209   210   211   212   213   214   215   216   217   218   21
9
##    16    24    20    19    18    27    23    16    32    15    21    19    2
1
##   220   221   222   223   224   225   226   227   228   229   230   231   23
2
##    15    27    14    15    12    23    18    18    11    20    21    16
6
##   233   234   235   236   237   238   239   240   241   242   243   244   24
5
##    24    17    14    14    11    18    11    15    10     7     7    10    1
1
##   246   247   248   249   250   251   252   253   254   255   256   257   25
8
##    13    14    12    15    14     8    10    11    16     6    11     9
9
##   259   260   261   262   263   264   265   266   267   268   269   270   27
1
##    14     8     8     6    11    12    10    10    12     8     6    10
8
##   272   273   274   275   276   277   278   279   280   281   282   283   28
4
##    15    13     6     7     6     8     6     6    10    10     5     6
9
##   285   286   287   288   289   290   291   292   293   294   295   296   29
7
##     3     6     5     6     4     4     7     6     4     9     7     2
9
```

```
##   298   299   300   301   302   303   304   305   306   307   308   309   31
0
##     5    10    11    10     9     6    10     4     5     3     7     5     1
2
##   311   312   313   314   315   316   317   318   319   320   321   322   32
3
##     5     4     4     5     4     3     3     7     3     4     2     6
5
##   324   325   326   327   328   329   330   331   332   333   334   335   33
6
##     5     4     3     5     8     7     3     6     7     4     2     3
8
##   337   338   339   340   341   342   343   344   345   346   347   348   34
9
##     3     5     2     6     2     5     3     1     6     5     4     1
6
##   350   351   352   353   354   355   356   357   358   359   360   361   36
2
##     3     1     4     2     3     2     5     1     4     3     3     1
1
##   363   364   365   366   367   368   369   370   371   372   373   374   37
5
##     2     2     1     5     2     3     3     4     2     4     4     3
2
##   376   377   378   379   380   381   382   383   385   386   387   388   39
0
##     2     5     4     4     2     3     1     2     3     1     3     2
3
##   391   392   393   394   395   396   397   398   401   402   403   404   40
5
##     4     1     3     5     2     2     3     2     1     2     1     3
5
##   406   408   409   410   411   412   413   416   417   418   419   420   42
1
##     4     1     1     2     1     2     2     2     1     1     1     1
2
##   422   423   424   425   426   428   429   431   433   437   438   439   44
0
##     1     1     2     2     1     2     1     3     1     1     2     2
1
##   442   444   446   449   450   451   453   454   457   458   459   460   46
1
##     2     3     3     3     1     1     1     1     2     1     1     1
3
##   462   465   466   468   471   473   474   475   479   480   481   482   48
6
##     3     1     1     1     2     1     1     1     1     1     1     1
1
##   487   488   489   492   494   496   498   499   500   501   502   503   50
4
##     2     1     1     1     1     1     1     1     1     1     1     1
1
##   505   506   507   508   510   516   522   525   533   535   537   540   54
```

```
2
##     2     3     1     1     1     2     1     1     1     1     1     1
1
##   543   547   550   553   554   555   559   561   565   577   579   580   58
2
##     2     1     1     1     1     3     1     1     1     1     1     1
1
##   585   588   591   594   597   603   618   619   630   634   649   651   65
2
##     1     3     1     1     2     2     1     1     1     1     1     1
1
##   668   671   679   680   681   693   695   702   706   707   711   725   72
7
##     1     1     1     1     1     1     1     1     2     1     2     2
1
##   728   731   746   748   754   758   762   768   770   774   786   790   79
5
##     1     1     1     1     1     1     1     1     1     1     1     1
1
##   806   808   821   836   840   854   862   866   872   880   895   899   90
7
##     1     1     1     2     1     1     1     1     1     1     1     1
1
##   908   946   965   973  1000  1006  1037  1071  1075  1076  1098  1099  116
0
##     1     1     1     1     1     1     1     1     1     1     1     1
1
##  1250  1259  1284  1314  1380
##     1     1     1     1     1
```

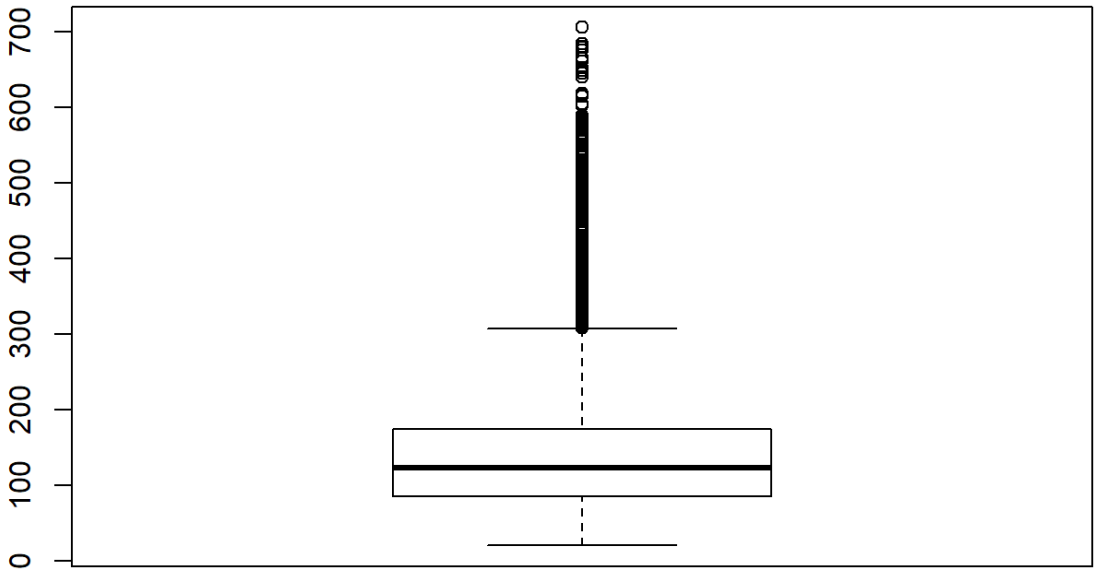
```
table(boxplot.stats(vuelos_reduc$DEPARTURE_DELAY)$out)
```

```
##
##  -68  -61  -44  -39  -38  -37  -35  -34  -33  -31  -30  -29  -28  -27  -26
-25
##    1    1    1    1    1    1    1    3    1    3    2    3    8    3    9
9
##  -24   26   27   28   29   30   31   32   33   34   35   36   37   38   39
40
##   18 1012 1004  930  909  912  809  799  783  719  789  679  664  602  572
546
##   41   42   43   44   45   46   47   48   49   50   51   52   53   54   55
56
##  543  544  507  493  493  512  452  413  424  468  403  413  357  373  352
327
##   57   58   59   60   61   62   63   64   65   66   67   68   69   70   71
72
##  339  325  337  328  319  303  293  276  298  266  273  283  256  246  219
216
##   73   74   75   76   77   78   79   80   81   82   83   84   85   86   87
88
##  234  223  193  200  199  197  207  183  186  183  174  192  160  164  155
173
##   89   90   91   92   93   94   95   96   97   98   99  100  101  102  103
104
##  157  155  148  134  149  131  149  146  138  126  116  156  130  123  147
128
##  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119
120
##  124  123  125  130  100  121   93   97  118  100   88   91   99   87   98
92
##  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135
136
##   89   82   95   85   85   82   87   81   79   74   76   66   87   81   68
78
##  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151
152
##   64   66   57   74   59   66   54   54   58   69   64   69   62   53   53
51
##  153  154  155  156  157  158  159  160  161  162  163  164  165  166  167
168
##   47   48   50   37   41   52   51   50   39   46   50   30   37   41   55
36
##  169  170  171  172  173  174  175  176  177  178  179  180  181  182  183
184
##   40   46   50   47   45   37   40   42   41   37   36   39   35   35   32
38
##  185  186  187  188  189  190  191  192  193  194  195  196  197  198  199
200
##   29   33   30   27   24   31   19   17   24   34   37   21   25   20   21
27
##  201  202  203  204  205  206  207  208  209  210  211  212  213  214  215
216
##   24   20   18   20   21   24   16   24   20   19   18   27   23   16   32
```

```
15
##  217  218  219  220  221  222  223  224  225  226  227  228  229  230  231
232
##   21   19   21   15   27   14   15   12   23   18   18   11   20   21   16
6
##  233  234  235  236  237  238  239  240  241  242  243  244  245  246  247
248
##   24   17   14   14   11   18   11   15   10    7    7   10   11   13   14
12
##  249  250  251  252  253  254  255  256  257  258  259  260  261  262  263
264
##   15   14    8   10   11   16    6   11    9    9   14    8    8    6   11
12
##  265  266  267  268  269  270  271  272  273  274  275  276  277  278  279
280
##   10   10   12    8    6   10    8   15   13    6    7    6    8    6    6
10
##  281  282  283  284  285  286  287  288  289  290  291  292  293  294  295
296
##   10    5    6    9    3    6    5    6    4    4    7    6    4    9    7
2
##  297  298  299  300  301  302  303  304  305  306  307  308  309  310  311
312
##    9    5   10   11   10    9    6   10    4    5    3    7    5   12    5
4
##  313  314  315  316  317  318  319  320  321  322  323  324  325  326  327
328
##    4    5    4    3    3    7    3    4    2    6    5    5    4    3    5
8
##  329  330  331  332  333  334  335  336  337  338  339  340  341  342  343
344
##    7    3    6    7    4    2    3    8    3    5    2    6    2    5    3
1
##  345  346  347  348  349  350  351  352  353  354  355  356  357  358  359
360
##    6    5    4    1    6    3    1    4    2    3    2    5    1    4    3
3
##  361  362  363  364  365  366  367  368  369  370  371  372  373  374  375
376
##    1    1    2    2    1    5    2    3    3    4    2    4    4    3    2
2
##  377  378  379  380  381  382  383  385  386  387  388  390  391  392  393
394
##    5    4    4    2    3    1    2    3    1    3    2    3    4    1    3
5
##  395  396  397  398  401  402  403  404  405  406  408  409  410  411  412
413
##    2    2    3    2    1    2    1    3    5    4    1    1    2    1    2
2
##  416  417  418  419  420  421  422  423  424  425  426  428  429  431  433
437
##    2    1    1    1    1    2    1    1    2    2    1    2    1    3    1
1
```

```
##  438  439  440  442  444  446  449  450  451  453  454  457  458  459  460
461
##    2    2    1    2    3    3    3    1    1    1    1    2    1    1    1
3
##  462  465  466  468  471  473  474  475  479  480  481  482  486  487  488
489
##    3    1    1    1    2    1    1    1    1    1    1    1    1    2    1
1
##  492  494  496  498  499  500  501  502  503  504  505  506  507  508  510
516
##    1    1    1    1    1    1    1    1    1    1    2    3    1    1    1
2
##  522  525  533  535  537  540  542  543  547  550  553  554  555  559  561
565
##    1    1    1    1    1    1    1    2    1    1    1    1    3    1    1
1
##  577  579  580  582  585  588  591  594  597  603  618  619  630  634  649
651
##    1    1    1    1    1    3    1    1    2    2    1    1    1    1    1
1
##  652  668  671  679  680  681  693  695  702  706  707  711  725  727  728
731
##    1    1    1    1    1    1    1    1    1    2    1    2    2    1    1
1
##  746  748  754  758  762  768  770  774  786  790  795  806  808  821  836
840
##    1    1    1    1    1    1    1    1    1    1    1    1    1    1    2
1
##  854  862  866  872  880  895  899  907  908  946  965  973 1000 1006 1037 1
071
##    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
1
## 1075 1076 1098 1099 1160 1250 1259 1284 1314 1380
##    1    1    1    1    1    1    1    1    1    1
```

```
boxplot(vuelos_reduc$SCHEDULED_TIME, main="SCHEDULED_TIME")
```



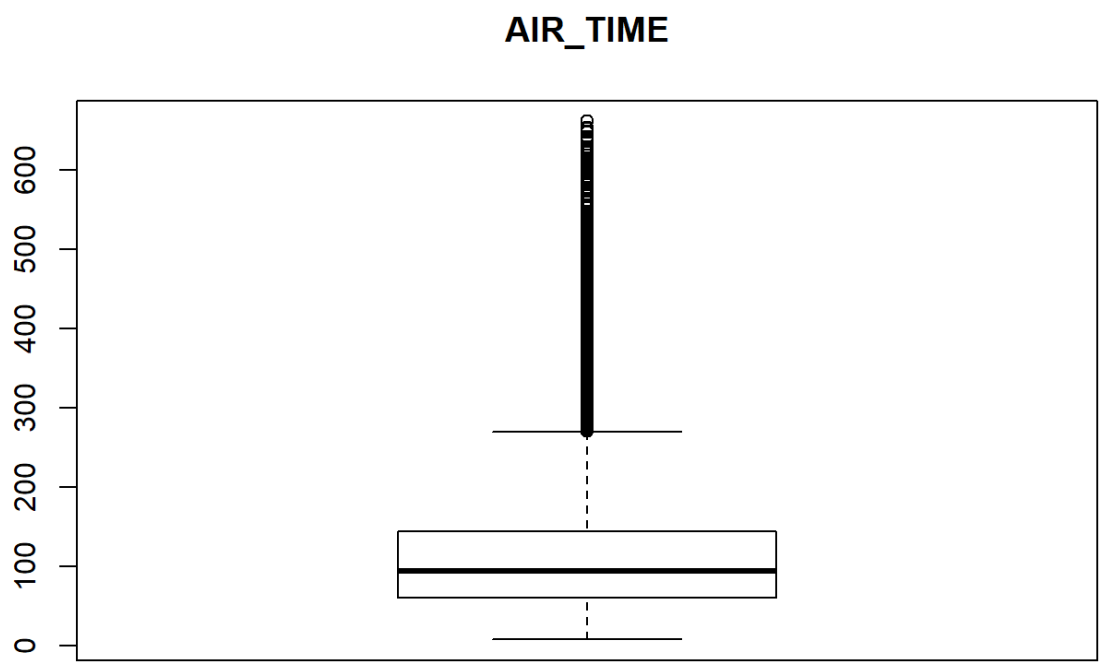

```
table(vuelos_reduc$SCHEDULED_TIME)
```

```
##
##   20   21   22   23   24   25   26   29   30   31   32   33   34   35   36
37
##    3    5    3    9    8    6    1    5   24    5  141  232  109  213  182
481
##   38   39   40   41   42   43   44   45   46   47   48   49   50   51   52
53
##  141  290  147  144  221  374  443  448  454  501  411  576 1060  732  766
812
##   54   55   56   57   58   59   60   61   62   63   64   65   66   67   68
69
##  810 1649  790  839  924  929 3247  894 1019 1182 1184 4481 1211 1350 1362 1
346
##   70   71   72   73   74   75   76   77   78   79   80   81   82   83   84
85
## 4699 1345 1439 1436 1477 5250 1741 1545 1518 1606 5535 1523 1587 1572 1496 5
735
##   86   87   88   89   90   91   92   93   94   95   96   97   98   99  100
101
## 1494 1525 1698 1516 5096 1447 1574 1556 1628 4202 1593 1513 1514 1510 3645 1
375
##  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116
117
## 1274 1410 1430 3633 1192 1435 1412 1554 3860 1338 1441 1464 1481 3667 1461 1
285
##  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132
133
## 1379 1531 3382 1170 1146 1338 1408 3143 1159 1183 1159 1235 2932 1099  995 1
073
##  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148
149
## 1090 3062  958 1046  987 1044 3057  893 1015  854  991 2869  907  868  988 1
074
##  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164
165
## 3049  969  957 1156 1142 3172  976  955 1015 1050 3073 1009  982  974 1065 2
764
##  166  167  168  169  170  171  172  173  174  175  176  177  178  179  180
181
##  977  928  969  962 2472  823  779  919  835 2136  821  831  715  793 1708
658
##  182  183  184  185  186  187  188  189  190  191  192  193  194  195  196
197
##  708  666  653 1481  552  614  660  580 1152  467  488  463  470 1167  387
424
##  198  199  200  201  202  203  204  205  206  207  208  209  210  211  212
213
##  443  454 1233  363  433  470  435 1153  392  393  419  411 1160  358  381
417
##  214  215  216  217  218  219  220  221  222  223  224  225  226  227  228
229
##  364  875  349  256  336  311  836  239  281  304  270  837  239  350  245
```

```
309
##  230  231  232  233  234  235  236  237  238  239  240  241  242  243  244
245
##  738  356  324  349  356  992  311  316  326  335 1058  281  286  291  259
733
##  246  247  248  249  250  251  252  253  254  255  256  257  258  259  260
261
##  234  307  292  193  748  209  232  224  224  632  204  230  220  244  642
142
##  262  263  264  265  266  267  268  269  270  271  272  273  274  275  276
277
##  198  195  213  563  201  217  212  219  502  152  187  205  189  519  180
219
##  278  279  280  281  282  283  284  285  286  287  288  289  290  291  292
293
##  206  183  483  183  186  167  163  402  104  134  114  197  450  100  134
158
##  294  295  296  297  298  299  300  301  302  303  304  305  306  307  308
309
##  152  289  154  103  118  145  300  100  127  165  203  350  147  172  174
133
##  310  311  312  313  314  315  316  317  318  319  320  321  322  323  324
325
##  390  141  142  167  275  391  152  176  163  178  425  152  203  211  218
465
##  326  327  328  329  330  331  332  333  334  335  336  337  338  339  340
341
##  145  178  226  200  483  166  177  129  230  312  123  170  116  183  419
128
##  342  343  344  345  346  347  348  349  350  351  352  353  354  355  356
357
##  133  155  150  266   70   82   99  106  315   95   97  142  122  235   83
84
##  358  359  360  361  362  363  364  365  366  367  368  369  370  371  372
373
##   91  109  187   52   60   81  101  174   63   52  104  107  179   76   83
56
##  374  375  376  377  378  379  380  381  382  383  384  385  386  387  388
389
##  100  145   88   79   53   84  176   94   69  101   79  143   42   70   77
61
##  390  391  392  393  394  395  396  397  398  399  400  401  402  403  404
405
##  196   48   39   82   48  130   43   49   47   58  103   58   22   42   36
66
##  406  407  408  409  410  411  412  413  414  415  416  417  418  419  420
421
##   21   25   29   17   22    9   12   12    8    4    4    8    9    8    3
3
##  423  424  426  427  428  430  433  434  435  436  437  438  439  440  441
443
##    2    7    2    5    3    5    1    3    3    5    2    1    3   11    9
12
```

```
##   444   448   449   450   451   453   454   455   457   459   460   461   462   465   466
467
##     4     6     2     1     1     3     2     1     8     9     8     5     3     7     6
4
##   468   469   470   471   473   475   477   479   480   482   485   487   491   492   493
494
##     2     1    16     2     2     3     2     2     2     3     9     1     2     2     4
2
##   496   497   498   500   502   503   505   507   508   509   512   513   514   515   516
517
##     3    10    11     6     4     2     4     2     4     4     4     1     3     2     8
1
##   518   519   520   522   523   526   533   535   537   540   541   543   544   545   548
550
##     3     8     6     5     2     2     1     1     3     1     3     1     3     5     1
1
##   554   555   556   557   558   559   560   561   562   565   570   575   578   579   580
581
##     1     2     2     2     3     6     3     2     3     1     3    11     3     3     1
5
##   583   584   586   587   588   602   604   614   616   618   640   645   648   652   660
665
##     6     1     1     2     1     1     3     8     1     2    10     1     2     5     2
3
##   675   679   680   683   705
##     3     1     1     1     2
```

```
boxplot(vuelos_reduc$AIR_TIME, main="AIR_TIME")
```



```
table(vuelos_reduc$AIR_TIME)
```

```
##
##    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22
23
##    3    7    5    4    5   10   32   56  100  148  196  277  456  649  766
723
##   24   25   26   27   28   29   30   31   32   33   34   35   36   37   38
39
##  683  743  780  879 1079 1151 1170 1197 1263 1307 1335 1420 1505 1568 1661 1
833
##   40   41   42   43   44   45   46   47   48   49   50   51   52   53   54
55
## 2145 2274 2406 2397 2449 2426 2368 2200 2191 2238 2248 2249 2301 2394 2335 2
303
##   56   57   58   59   60   61   62   63   64   65   66   67   68   69   70
71
## 2465 2308 2321 2383 2384 2447 2495 2612 2484 2527 2494 2338 2264 2152 2129 1
983
##   72   73   74   75   76   77   78   79   80   81   82   83   84   85   86
87
## 1942 1878 1864 1961 1938 1960 1935 1972 2079 2105 2047 2099 2124 2095 2080 2
145
##   88   89   90   91   92   93   94   95   96   97   98   99  100  101  102
103
## 2112 2030 1994 1909 1825 1847 1770 1669 1633 1695 1572 1559 1600 1583 1458 1
467
##  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118
119
## 1434 1532 1445 1472 1507 1473 1486 1427 1382 1360 1401 1452 1464 1487 1457 1
453
##  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134
135
## 1421 1440 1435 1461 1510 1460 1516 1410 1431 1367 1394 1469 1384 1345 1386 1
336
##  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150
151
## 1359 1301 1329 1253 1253 1199 1171 1140 1190 1150 1064 1074 1040 1099 1027 1
017
##  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166
167
##  972  962  847  870  817  793  801  750  757  656  641  615  643  605  618
556
##  168  169  170  171  172  173  174  175  176  177  178  179  180  181  182
183
##  589  563  586  530  546  560  537  570  542  541  530  526  508  506  486
489
##  184  185  186  187  188  189  190  191  192  193  194  195  196  197  198
199
##  507  454  470  422  462  442  462  437  450  484  482  443  480  493  456
488
##  200  201  202  203  204  205  206  207  208  209  210  211  212  213  214
215
##  443  450  487  448  460  410  422  411  438  390  386  424  382  344  390
```

```
364
##  216  217  218  219  220  221  222  223  224  225  226  227  228  229  230
231
##  341  323  343  320  330  323  330  312  322  324  329  319  306  289  303
299
##  232  233  234  235  236  237  238  239  240  241  242  243  244  245  246
247
##  280  279  266  258  273  274  267  271  254  242  228  226  259  243  214
224
##  248  249  250  251  252  253  254  255  256  257  258  259  260  261  262
263
##  212  222  202  216  219  215  209  191  210  212  185  196  210  204  201
191
##  264  265  266  267  268  269  270  271  272  273  274  275  276  277  278
279
##  194  183  221  199  195  212  228  205  192  231  242  243  211  214  204
249
##  280  281  282  283  284  285  286  287  288  289  290  291  292  293  294
295
##  222  250  225  214  210  215  223  226  246  234  221  245  223  257  250
228
##  296  297  298  299  300  301  302  303  304  305  306  307  308  309  310
311
##  186  181  213  198  206  192  193  172  154  206  195  193  168  148  167
158
##  312  313  314  315  316  317  318  319  320  321  322  323  324  325  326
327
##  160  159  147  154  140  174  160  149  141  159  148  145  126  149  118
108
##  328  329  330  331  332  333  334  335  336  337  338  339  340  341  342
343
##  145  138  116  111  116  109  106  135  113   93   90   93   80   90   86
71
##  344  345  346  347  348  349  350  351  352  353  354  355  356  357  358
359
##   94   89   73   75   75   68   63   60   68   57   58   62   58   47   43
46
##  360  361  362  363  364  365  366  367  368  369  370  371  372  373  374
375
##   32   40   45   41   40   52   33   33   37   35   24   26   22   17   18
21
##  376  377  378  379  380  381  382  383  384  385  386  387  388  389  390
391
##   23   13   11   14   20   15    7   12   10    9    5   11    5    8    5
4
##  392  393  394  395  396  397  398  399  400  401  402  403  404  405  406
407
##   11   11    3    5    5    3    9   10    2    2    3    8    3    3    6
5
##  408  409  410  411  412  413  414  415  416  417  418  419  420  421  422
423
##    4    3    2    6   10    5    6    4    5    4    5    2    4    4    5
3
```

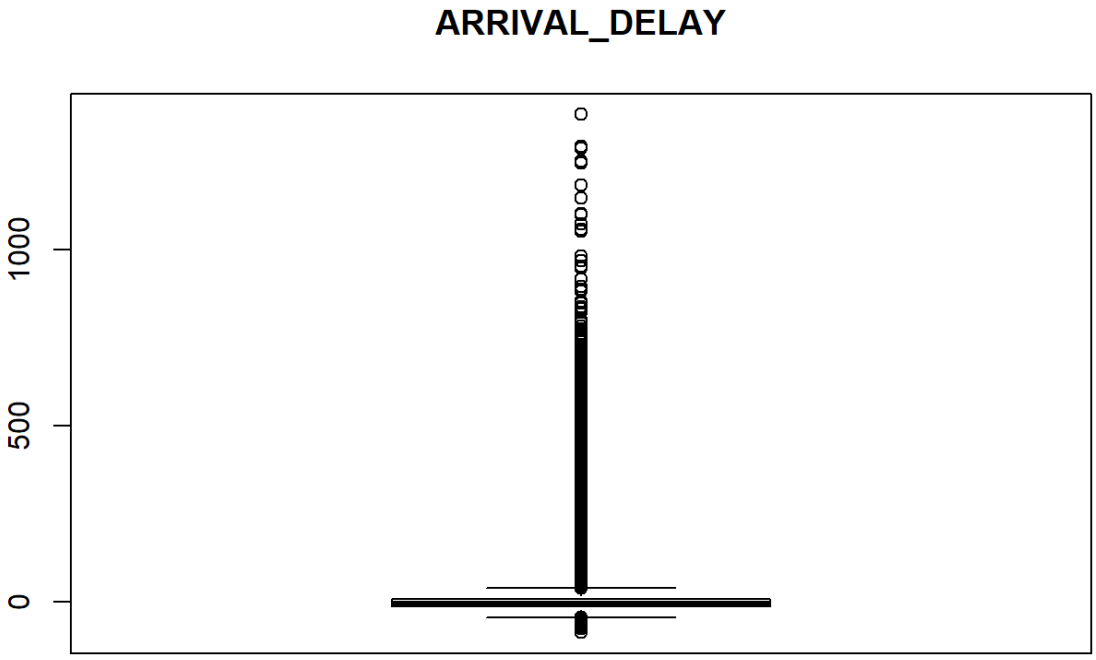
```
##  424  425  426  427  428  429  430  431  432  433  434  435  438  439  440
441
##    4    5    1    3    1    1    5    3    3    3    2    3    3    1    4
2
##  443  444  445  446  447  448  449  450  451  452  453  455  456  457  458
459
##    2    4    1    2    2    1    3    3    1    4    2    1    2    2    2
8
##  460  461  462  463  464  465  466  467  468  469  470  471  472  473  474
475
##    3    5    1    3    2    5    2    2    1    1    1    1    3    6    6
2
##  476  477  478  480  481  482  483  484  485  486  487  488  489  490  491
492
##    1    2    1    1    2    4    2    1    3    4    1    7    1    4    2
1
##  493  494  495  497  498  499  500  502  503  504  505  506  507  508  509
510
##    2    2    2    1    1    2    2    3    1    1    4    2    1    3    1
3
##  511  513  514  515  516  517  518  519  521  522  523  524  527  529  530
533
##    1    1    2    1    1    2    3    3    2    1    5    1    2    1    3
3
##  536  537  538  539  540  541  542  544  546  547  548  552  553  556  559
560
##    1    2    1    1    1    2    2    6    3    1    1    2    3    2    1
3
##  561  562  567  569  575  576  578  581  585  587  588  590  595  597  602
603
##    1    1    1    1    2    1    1    1    1    1    1    1    1    1    2
1
##  606  608  610  612  613  614  619  622  626  627  628  629  636  638  641
642
##    2    3    1    1    1    1    1    2    1    2    1    1    1    1    1
1
##  647  649  652  653  661
##    1    1    1    1    1
```

```
boxplot(vuelos_reduc$DISTANCE, main="DISTANCE")
```


```
table(boxplot.stats(vuelos_reduc$DISTANCE)$out)
```

```
##
## 2105 2106 2110 2116 2125 2130 2133 2136 2139 2149 2153 2158 2161 2165 2172 2
173
##   69  179   12   28  212    6  227   24  340   48  261   50   41   68  109
134
## 2174 2176 2182 2218 2227 2237 2239 2244 2248 2253 2254 2267 2279 2282 2288 2
295
##  164  271  292  242  177   34   93   17  382  133   41   30  103   31  376
75
## 2296 2297 2300 2304 2306 2311 2327 2329 2330 2335 2338 2342 2343 2345 2349 2
350
##  218   34  138    9  114  112   29  202    4   86  139  393  145   35   86
35
## 2354 2355 2358 2360 2367 2370 2378 2381 2384 2393 2398 2400 2402 2404 2405 2
406
##    8   90   30    9   84   82  143  108   16    1  234   47  409   46   42
16
## 2409 2411 2415 2417 2419 2422 2425 2434 2442 2446 2447 2449 2454 2457 2458 2
462
##   63   22    6   46  366  225  105  133   61  296   80   26  607   48    8
28
## 2465 2466 2475 2486 2496 2504 2519 2520 2521 2537 2541 2552 2554 2556 2562 2
565
##   99   28 1225  306  165  149   63   30  290   45   27   40   56  596   72
541
## 2569 2576 2583 2584 2585 2586 2588 2599 2602 2603 2607 2611 2614 2615 2631 2
636
##   40   22   30  108  109  913   95   12   29   77   13  420   68  128   12
5
## 2640 2676 2677 2681 2688 2689 2694 2701 2704 2715 2717 2724 2762 2777 2785 2
797
##  120   21  163    9   58   10   10   59  446   14   29   38   88   26   10
5
## 2845 2846 2860 2874 2917 2979 2994 3043 3266 3302 3329 3365 3414 3417 3711 3
784
##   54   69   40    3  120   37   37    9    8   18   15   23    7   12   52
77
## 3801 3904 3972 4184 4243 4502 4817 4962 4983
##   33   36    4   16   27   42   12   31   36
```

```
boxplot(vuelos_reduc$ARRIVAL_DELAY, main="ARRIVAL_DELAY")
```




```
table(vuelos_reduc$ARRIVAL_DELAY)
```

```
##
##  -87  -77  -75  -72  -71  -69  -67  -66  -65  -63  -62  -61  -60  -59  -58
-57
##    1    1    1    1    1    1    1    1    1    3    5    4    6    1    9
9
##  -56  -55  -54  -53  -52  -51  -50  -49  -48  -47  -46  -45  -44  -43  -42
-41
##   10   11   17   19   24   24   27   48   53   40   66   92   92  109  137
143
##  -40  -39  -38  -37  -36  -35  -34  -33  -32  -31  -30  -29  -28  -27  -26
-25
##  180  216  236  278  344  460  530  574  757  804 1018 1219 1366 1653 1922 2
235
##  -24  -23  -22  -21  -20  -19  -18  -17  -16  -15  -14  -13  -12  -11  -10
-9
## 2596 2934 3357 3963 4403 5004 5541 6005 6464 7024 7511 7803 8174 8471 8789 8
797
##   -8   -7   -6   -5   -4   -3   -2   -1    0    1    2    3    4    5    6
7
## 8966 8711 8610 8127 7887 7416 7038 6688 6359 5627 5288 4948 4596 4313 3837 3
646
##    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22
23
## 3241 3063 2906 2609 2474 2397 2183 1981 1927 1718 1641 1577 1469 1323 1433 1
286
##   24   25   26   27   28   29   30   31   32   33   34   35   36   37   38
39
## 1201 1157 1049 1030  965  887  923  842  769  819  750  732  627  690  687
594
##   40   41   42   43   44   45   46   47   48   49   50   51   52   53   54
55
##  604  587  543  528  511  481  486  469  415  424  434  356  395  375  355
369
##   56   57   58   59   60   61   62   63   64   65   66   67   68   69   70
71
##  336  364  328  321  308  325  290  258  239  301  285  261  256  251  251
242
##   72   73   74   75   76   77   78   79   80   81   82   83   84   85   86
87
##  220  227  229  226  228  165  197  200  205  195  181  203  192  179  164
175
##   88   89   90   91   92   93   94   95   96   97   98   99  100  101  102
103
##  160  160  155  157  170  146  149  141  161  117  128  123  147  140  120
117
##  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118
119
##  102  112  113  110  112  106  111  111   96  115   99   96  103  108   87
82
##  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134
135
##  102   86   83   73   87   88   80   79   86   81   73   72   79   94   70
```

```
95
##  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150
151
##   71   67   63   60   58   65   71   59   79   49   66   46   48   50   59
58
##  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166
167
##   51   61   49   54   54   59   46   67   44   42   43   56   42   52   39
48
##  168  169  170  171  172  173  174  175  176  177  178  179  180  181  182
183
##   36   41   45   41   40   37   31   38   35   30   30   39   27   27   29
36
##  184  185  186  187  188  189  190  191  192  193  194  195  196  197  198
199
##   26   29   26   30   30   26   24   27   24   26   18   31   30   19   26
16
##  200  201  202  203  204  205  206  207  208  209  210  211  212  213  214
215
##   24   21   23   27   19   19   33   20   36   17   23   22   26   21   24
23
##  216  217  218  219  220  221  222  223  224  225  226  227  228  229  230
231
##   16   22   16   16   22   20   23   16   21   15   18   21   15   15   21
13
##  232  233  234  235  236  237  238  239  240  241  242  243  244  245  246
247
##    9   18   15   16   18   13   14   12    8   17   14   12    9   12   12
8
##  248  249  250  251  252  253  254  255  256  257  258  259  260  261  262
263
##   16   10    8   10   12   13    6   10    8   11   13    7   13   12    9
10
##  264  265  266  267  268  269  270  271  272  273  274  275  276  277  278
279
##    9    7   14   10    7   10    5    8   10   10   10   10    9    2    6
10
##  280  281  282  283  284  285  286  287  288  289  290  291  292  293  294
295
##    6    8    7   10    8    7    6    7    3    5    4    9    9    3    9
7
##  296  297  298  299  300  301  302  303  304  305  306  307  308  309  310
311
##    3    6   11    9    5    2    3    8    5    3    8   10    4    9    7
2
##  312  313  314  315  316  317  318  319  320  321  322  323  324  325  326
327
##    5    8    5    2    7    3    3   10    2    5    4   11    1    4    6
3
##  328  329  330  331  332  333  334  335  336  337  338  339  340  341  342
343
##    2    4    4    8    5    5    6    2    6    4    6    4    6    2    4
4
```

```
##  344  345  346  347  348  349  350  351  352  353  354  355  356  357  358
359
##    4    1    3    3    3    3    1    4    2    6    3    7    2    4    2
1
##  360  361  362  363  364  365  367  368  369  370  371  372  373  374  375
376
##    3    3    1    3    3    3    1    2    2    2    3    2    3    4    1
2
##  377  378  379  380  381  382  383  384  385  386  387  388  389  391  392
393
##    2    2    4    4    1    4    1    2    1    2    2    2    6    3    3
2
##  395  396  398  399  400  401  402  403  405  406  407  410  411  412  413
415
##    1    2    1    2    2    5    3    4    2    2    2    1    1    3    2
1
##  417  418  419  420  421  422  423  424  425  426  427  428  429  432  433
435
##    3    2    2    2    1    1    3    2    1    1    3    2    2    1    1
2
##  436  437  438  439  440  442  445  446  447  450  451  452  453  456  457
459
##    1    2    1    1    1    1    1    1    3    2    6    2    1    4    1
1
##  462  463  464  465  467  468  472  474  475  477  479  487  489  490  493
494
##    1    2    1    2    3    1    1    3    3    2    1    1    2    1    1
1
##  498  500  501  502  503  505  506  509  513  515  521  526  529  530  531
532
##    1    2    1    2    1    1    1    1    1    1    2    1    1    1    1
1
##  533  535  546  547  550  551  552  553  556  561  562  563  564  571  573
574
##    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
1
##  577  580  583  586  590  595  596  600  612  624  627  628  636  639  649
662
##    1    2    1    1    1    1    1    2    1    1    1    1    1    1    1
1
##  663  670  674  679  684  685  691  692  697  701  703  706  715  719  721
725
##    1    1    2    1    1    1    1    1    1    1    1    1    1    1    1
1
##  729  739  744  745  747  754  757  763  774  775  783  784  794  799  801
802
##    1    2    1    1    1    1    1    1    1    1    1    1    1    1    1
1
##  821  833  837  847  853  854  883  888  896  917  947  954  968  981 1053 1
057
##    1    1    1    1    1    1    2    1    1    1    1    2    1    1    1
1
```

```
## 1073 1099 1102 1147 1183 1247 1251 1289 1294 1384
##    1    1    1    1    1    1    1    1    1    1
```

El conjunto de datos que estamos tratando tiene muchos valores muy dispersos, de momento no vamos a hacer nada con ellos, los mantendremos y en algunos casos concretos, si fuese necesario, realizaremos los filtrados correspondientes.

4. Análisis de los datos

4.1 Selección de los grupos de datos >

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

```
# Para poder trabajar mejor con las variables de tiempo, y dado que no nos van
 a interesar en principio los minutos
# vamos a dejar solamente las horas para poder analizar mejor este dato.
vuelos_reduc$SCHEDULED_DEPARTURE_HOUR=format(round(trunc(vuelos_reduc$SCHEDULED
_DEPARTURE/100),digits=0), nsmall=0)
vuelos_reduc$SCHEDULED_DEPARTURE_HOUR <- as.numeric(vuelos_reduc$SCHEDULED_DEPA
RTURE_HOUR)
head(vuelos_reduc$SCHEDULED_DEPARTURE_HOUR)
```

```
## [1]  8  9  5 15 10  6
```

```
table(vuelos_reduc$SCHEDULED_DEPARTURE_HOUR)
```

```
##
##     0     1     2     3     4     5     6     7     8     9    10    11    1
2
##   736   297    60    34    33  5756 20063 19297 18745 17405 18383 17633 1757
1
##    13    14    15    16    17    18    19    20    21    22    23
## 17852 16233 18195 16228 19053 16117 16315 12733  9072  5695  2144
```

```
vuelos_reduc$DEPARTURE_HOUR=format(round(trunc(vuelos_reduc$DEPARTURE_TIME/100
),digits=0), nsmall=0)
vuelos_reduc$DEPARTURE_HOUR <- as.factor(vuelos_reduc$DEPARTURE_HOUR)
head(vuelos_reduc$DEPARTURE_HOUR)
```

```
## [1]  8  9  5 15 11  5
## 25 Levels:  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 ... 2
4
```

```
table(vuelos_reduc$DEPARTURE_HOUR)
```

```
##
##     0     1     2     3     4     5     6     7     8     9    10    11    1
2
##  1258   489   121    48   493  9758 18295 17385 17678 16948 17972 17332 1734
3
##    13    14    15    16    17    18    19    20    21    22    23    24
## 17092 16410 17628 16413 18053 15904 16375 13381  9887  6556  2811    20
```

```
vuelos_reduc$ARRIVAL_HOUR=format(round(trunc(vuelos_reduc$ARRIVAL_TIME/100),dig
its=0), nsmall=0)
vuelos_reduc$ARRIVAL_HOUR <- as.factor(vuelos_reduc$ARRIVAL_HOUR)
head(vuelos_reduc$ARRIVAL_HOUR)
```

```
## [1]  9 12  6 17 15  7
## 25 Levels:  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 ... 2
4
```

```
table(vuelos_reduc$ARRIVAL_HOUR)%>% knitr::kable("html")  %>% kable_styling(pos
ition='center', font_size=12, fixed_thead=list(enabled=T))
```

Var1	Freq
0	4963
1	1524
2	471
3	181
4	606
5	2053
6	4850
7	10154
8	13810
9	16512
10	16853
11	17020
12	16798
13	16815
14	17228
15	16374
16	18418

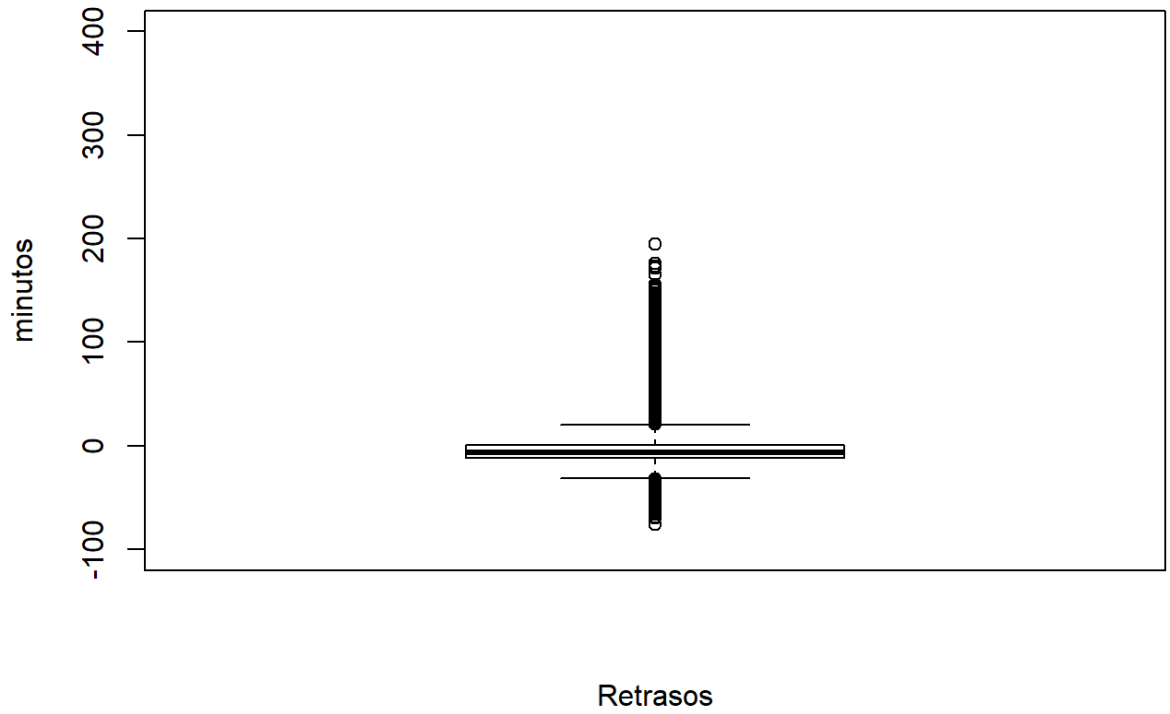
Var1	Freq
17	17300
18	17478
19	17006
20	17359
21	16372
22	14323
23	11033
24	149

Creamos una nueva variables que identifique el retraso total, en base a la perdida o ganancia en el tiempo de llegada frente al tiempo de adelanto o retraso en la salida

```
vuelos_reduc <- mutate(vuelos_reduc, RETRASO_TOTAL=ARRIVAL_DELAY - DEPARTURE_DE
LAY)

str(vuelos_reduc)
```

```
## 'data.frame':    285650 obs. of  21 variables:
##  $ MONTH                   : int  11 4 4 7 2 7 12 2 3 2 ...
##  $ DAY                     : int  16 7 1 28 9 23 24 22 8 8 ...
##  $ DAY_OF_WEEK             : int  1 2 3 2 1 4 4 7 7 7 ...
##  $ AIRLINE                 : Factor w/ 14 levels "AA","AS","B6",..: 5 14 4 5
14 1 14 10 8 4 ...
##  $ FLIGHT_NUMBER           : int  5084 1023 2182 4330 1963 2148 1915 4636 29
50 2104 ...
##  $ ORIGIN_CODE             : Factor w/ 628 levels "10135","10136",..: 327 35
8 439 504 483 346 344 593 535 523 ...
##  $ DESTINATION_AIRPORT     : Factor w/ 629 levels "10135","10136",..: 614 57
7 328 459 500 490 368 432 573 393 ...
##  $ SCHEDULED_DEPARTURE     : int  825 930 540 1545 1055 600 1125 1956 1145 1
935 ...
##  $ DEPARTURE_TIME          : int  819 943 538 1559 1105 553 1124 2002 1242 1
932 ...
##  $ DEPARTURE_DELAY         : int  -6 13 -2 14 10 -7 -1 6 57 -3 ...
##  $ SCHEDULED_TIME          : int  128 240 57 108 190 74 80 63 68 137 ...
##  $ ELAPSED_TIME            : int  129 214 47 105 166 68 86 57 68 140 ...
##  $ AIR_TIME                : int  111 201 28 75 153 47 64 35 51 95 ...
##  $ DISTANCE                 : int  674 1407 153 468 1363 184 439 216 268 680
...
##  $ SCHEDULED_ARRIVAL       : int  933 1230 637 1733 1605 714 1345 2059 1253
2052 ...
##  $ ARRIVAL_TIME            : int  928 1217 625 1744 1551 701 1350 2059 1350
2052 ...
##  $ ARRIVAL_DELAY           : int  -5 -13 -12 11 -14 -13 5 0 57 0 ...
##  $ SCHEDULED_DEPARTURE_HOUR: num  8 9 5 15 10 6 11 19 11 19 ...
##  $ DEPARTURE_HOUR          : Factor w/ 25 levels " 0"," 1"," 2",..: 9 10 6 1
6 12 6 12 21 13 20 ...
##  $ ARRIVAL_HOUR             : Factor w/ 25 levels " 0"," 1"," 2",..: 10 13 7
18 16 8 14 21 14 21 ...
##  $ RETRASO_TOTAL           : int  1 -26 -10 -3 -24 -6 6 -6 0 3 ...
```

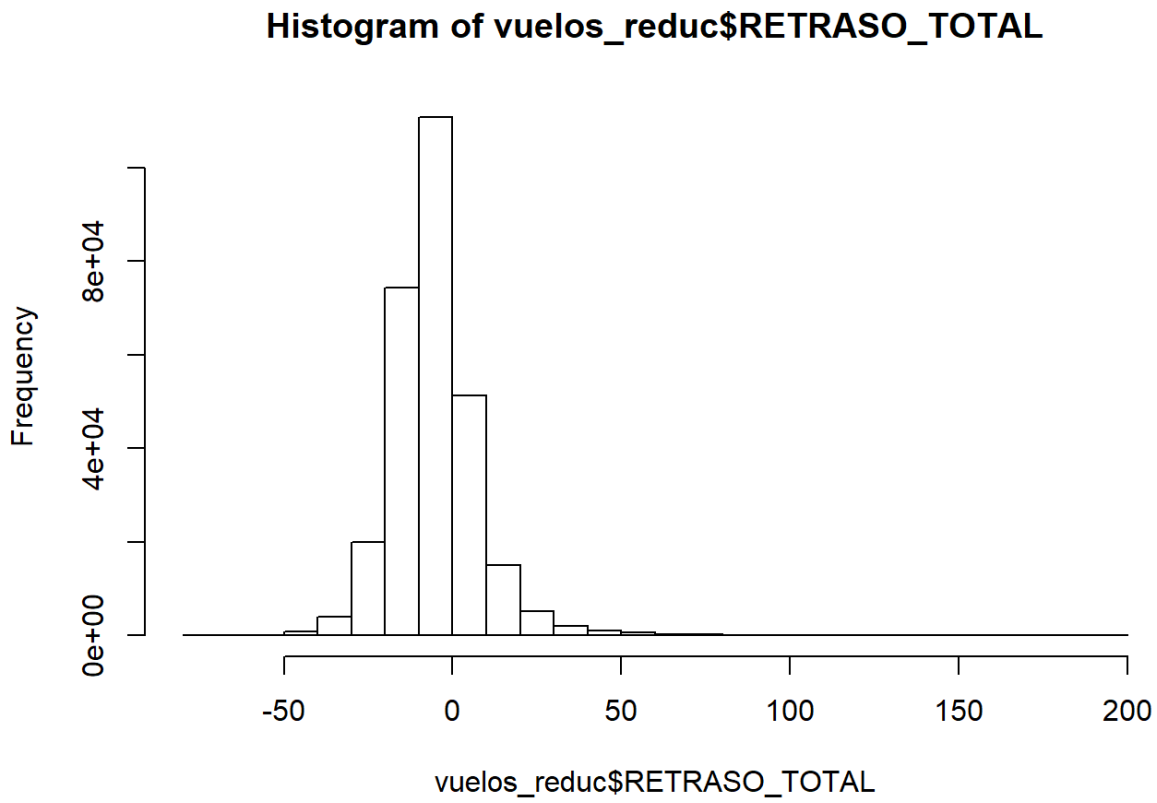
```
boxplot(vuelos_reduc$RETRASO_TOTAL , xlab="Retrasos", ylab="minutos", ylim=c(-1
00, 400) )
```



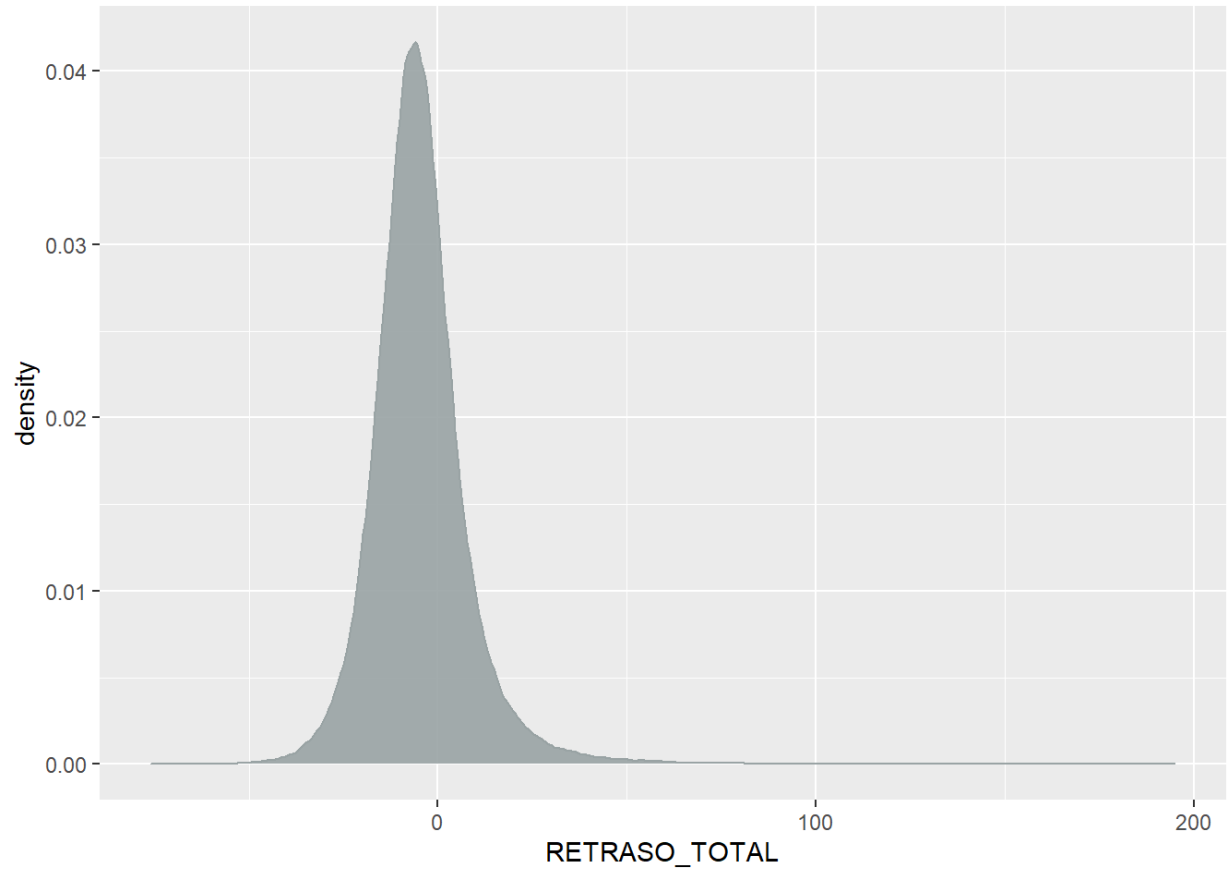

```
table(boxplot.stats(vuelos_reduc$RETRASO_TOTAL)$out)
```

```
##
## -76 -70 -69 -66 -64 -63 -62 -61 -60 -59 -58 -57 -56 -55 -54 -53 -52 -51 -50
-49
##   1   1   1   1   1   1   4   2   2   2   1   5   5  11   9  13  29  15  27
36
## -48 -47 -46 -45 -44 -43 -42 -41 -40 -39 -38 -37 -36 -35 -34 -33 -32  21  22
23
##  34  39  61  60  68  70 106 116 139 173 167 230 286 347 391 449 575 746 725
613
##  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42
43
## 561 490 494 435 382 351 302 250 269 260 218 222 203 202 150 162 131 142  98
111
##  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62
63
## 104 107  94  60  84  85  95  62  53  65  56  65  46  52  46  48  42  35  31
26
##  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82
83
##  30  25  14  25  21  17  24  18  18  19  24  19  23  14  18  17  18  14  10
9
##  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 100 101 102
103
##  10  15  10  13  14   2   4   5   3   9   5   6   3   6   8   4   3   5   3
3
## 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122
123
##   9   6   5   5   4   5   5   3   6   2   4   3   4   3   5   3   3   2   2
1
## 125 126 127 128 129 130 131 132 134 136 137 139 140 141 142 143 144 145 146
147
##   1   1   3   3   3   3   5   2   3   1   1   1   1   1   2   1   2   1   2
1
## 148 150 152 153 156 165 172 173 176 195
##   1   1   1   1   1   1   1   1   1   1
```

```
hist(vuelos_reduc$RETRASO_TOTAL)
```




```
vuelos_reduc %>%
  ggplot( aes(x=RETRASO_TOTAL)) +
  geom_density(fill="#99A3A4", color="#99A3A4", alpha=0.9)
```



4.2 Comprobación de la normalidad y homogeneidad de la varianza

```
#Vamos a comprobar la normalidad de la muestra para los valores DEPARTURE_DELA
Y, ARRIVAL_DELAY y DISTANCE
# utilizando la prueba shapiro-Wilk test.

shapiro.test(vuelos_reduc$DEPARTURE_DELAY[1:5000])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  vuelos_reduc$DEPARTURE_DELAY[1:5000]
## W = 0.45789, p-value < 2.2e-16
```

```
shapiro.test(vuelos_reduc$ARRIVAL_DELAY[1:5000])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  vuelos_reduc$ARRIVAL_DELAY[1:5000]
## W = 0.58009, p-value < 2.2e-16
```

```
shapiro.test(vuelos_reduc$DISTANCE[1:5000])
```

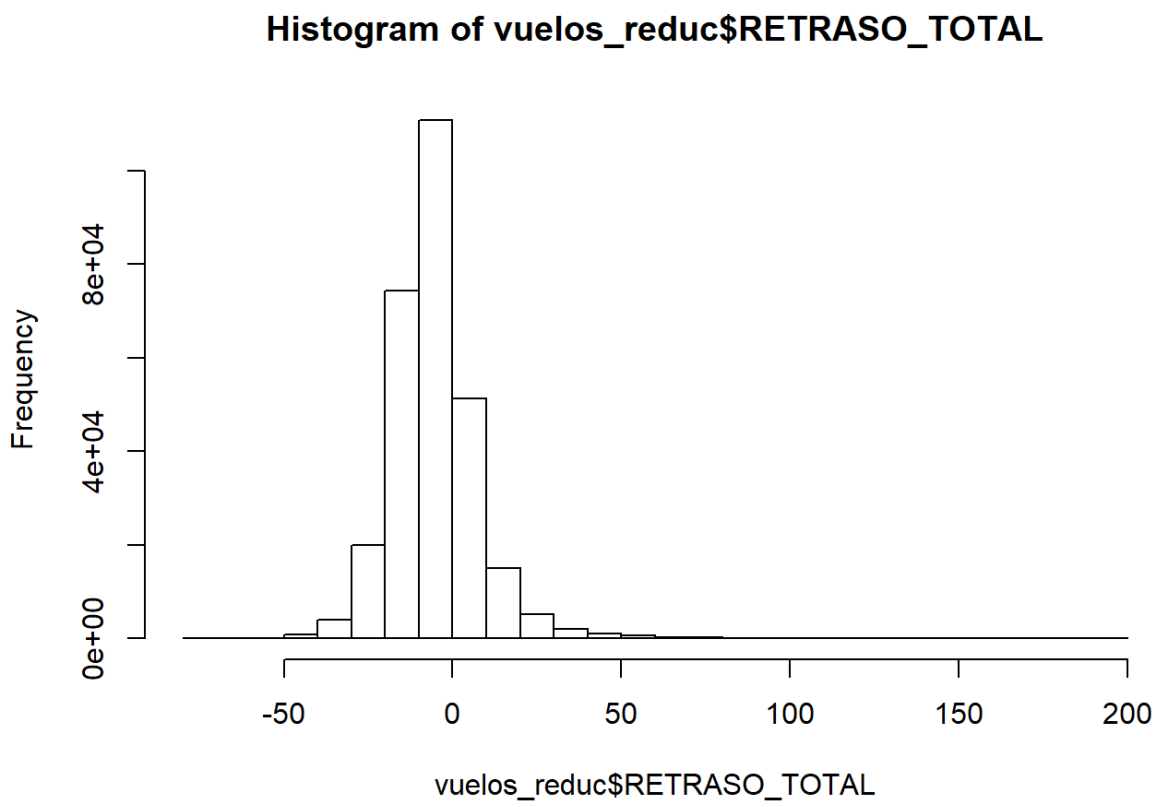
```
##
##  Shapiro-Wilk normality test
##
## data:  vuelos_reduc$DISTANCE[1:5000]
## W = 0.88101, p-value < 2.2e-16
```

```
shapiro.test(vuelos_reduc$RETRASO_TOTAL[1:5000])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  vuelos_reduc$RETRASO_TOTAL[1:5000]
## W = 0.91736, p-value < 2.2e-16
```

La prueba de Saphiro-Wilk, solo es posible para un máximo de 5000 registros, por lo que hemos realizado la prueba con un subconjunto con esa cantidad, y nos da como resultado que debemos rechazar la hipótesis nula, es decir, nos indicaría que las variables no siguen una distribución normal. Sin embargo, por el teorema central del límite, la distribución de la media de cualquier muestra de datos se considera cada vez más normal según aumenta el tamaño de la misma y para muestras superiores con N>30 se puede suponer normalidad, dado que podría aproximarse a una disribución normal.

```
hist(vuelos_reduc$RETRASO_TOTAL)
```



Aparentemenete, para esta variable, según el gráfico podríamos decir que la variable sigue una distribución normal.

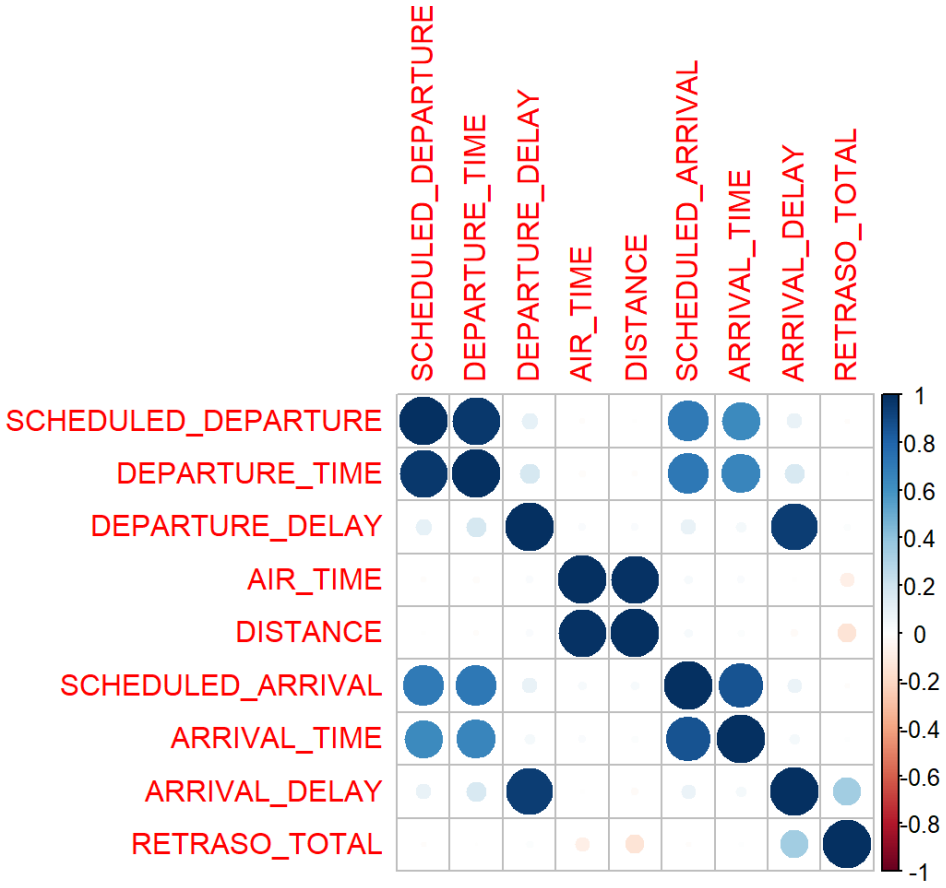
Falta comprobar homogeneidad de la varianza, utilizar test de Filgner-Killeen

4.3 Aplicación de 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. Aplicar al menos tres métodos de análisis diferentes.

Lo primero que vamos a hacer es comprobar la correlación entre algunas de las variables:

```
correlacion <- dplyr::select(vuelos_reduc, "SCHEDULED_DEPARTURE", "DEPARTURE_TI
ME", "DEPARTURE_DELAY",
                      "AIR_TIME", "DISTANCE", "SCHEDULED_ARRIVAL", "ARRIVAL_TIM
E","ARRIVAL_DELAY",
                      "RETRASO_TOTAL")

corr.res<-cor(correlacion)
corrplot(corr.res,method="circle")
```



Comprobamos una fuerte relación entre el retraso en la salida y el retraso en la llegada, que puee deberse a los vuelos no retrasados o quizá pueda ser, que al contrario de lo que podríamos pensar, el retraso en la salida no es recuperado en la llegada.
También vemos una fuerte relación entre la distancia y el tiempo de vuelo, algo que era de esperar. Se tiene también una relación, aunque no tan fuerte entre el tiempo estimado de llegada y el tiempo real de llegada.

Nos preguntamos ahora por la relación entre el retraso/adelanto de un vuelo con la hora del día en la que se realiza Para ello usaremos un modelo de regresión, dado que la variable día de la semana es de tipo factor y además crearemos una nueva variable que nos indique si el vuelo se ha retrasado o no.
Primero vamos a crear la tabla de contingencia y calcularemos la estimación Odds Ratio para ver la relación entre las variables Veremos si existe relación entre la variable dependiente, en nuestro caso si hay retraso o no y las variables explicativas.
Dividimos la muestra:

```
retraso <- data.frame (RETRASO=vuelos_reduc$RETRASO_TOTAL)
retraso$RETRASO <- ifelse (retraso$RETRASO>0, "SI", "NO")
retraso <- data.frame (retraso, WEEKEND=vuelos_reduc$DAY_OF_WEEK)
#Contamos como fin de semana los viernes, sábados y domingos
retraso$WEEKEND <- ifelse ((retraso$WEEKEND=="5" | retraso$WEEKEND=="6" | retra
so$WEEKEND=="7"), "WEEKEND", "WEEKDAY")
str(retraso)
```

```
## 'data.frame':    285650 obs. of  2 variables:
##  $ RETRASO: chr  "SI" "NO" "NO" "NO" ...
##  $ WEEKEND: chr  "WEEKDAY" "WEEKDAY" "WEEKDAY" "WEEKDAY" ...
```

```
table (retraso$RETRASO)
```

```
##
##     NO     SI
## 209810  75840
```

```
tabla_retraso_dias <- with(retraso, table(retraso$RETRASO,retraso$WEEKEND))
tabla_retraso_dias %>% knitr::kable("html")  %>% kable_styling(position='cente
r', font_size=12, fixed_thead=list(enabled=T))
```

	WEEKDAY	WEEKEND
NO	122977	86833
SI	45656	30184

Aplicamos la función chi-cuadrado de Pearson a las variables para conocer si podemos aceptar la hipótesis nula y por lo tanto las variables no están relacionadas.

```
chisq.test(tabla_retraso_dias, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla_retraso_dias
## X-squared = 58.006, df = 1, p-value = 2.613e-14
```

El p-value encontrado con el test chi-cuadrado es p-value < 2.2e-16 que se encuentra muy por debajo del nivel de significación marcado de 0.05, por lo que rechazamos la hipótesis nula y por lo tanto podemos concluir en este caso que existe relación entre retraso y el día de la semana.

```
library(epitools)
```

```
##
## Attaching package: 'epitools'
```

```
## The following object is masked from 'package:survival':
##
##     ratetable
```

```
oddsratio(tabla_retraso_dias, verbose = TRUE)
```

```
## $x
##
##      WEEKDAY WEEKEND
##   NO  122977   86833
##   SI   45656   30184
##
## $data
##
##         WEEKDAY WEEKEND  Total
##   NO     122977   86833 209810
##   SI      45656   30184  75840
##   Total  168633  117017 285650
##
## $p.exposed
##
##           WEEKDAY   WEEKEND     Total
##   NO    0.7292582 0.7420546 0.7345003
##   SI    0.2707418 0.2579454 0.2654997
##   Total 1.0000000 1.0000000 1.0000000
##
## $p.outcome
##
##           WEEKDAY   WEEKEND Total
##   NO    0.5861351 0.4138649     1
##   SI    0.6020042 0.3979958     1
##   Total 0.5903483 0.4096517     1
##
## $measure
##     odds ratio with 95% C.I.
##       estimate     lower     upper
##   NO 1.0000000        NA        NA
##   SI 0.9362943 0.9205703 0.9522808
##
## $conf.level
## [1] 0.95
##
## $p.value
##     two-sided
##        midp.exact fisher.exact   chi.square
##   NO           NA           NA           NA
##   SI 2.475797e-14 2.518942e-14 2.612958e-14
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "median-unbiased estimate & mid-p exact CI"
```

```
oddsratio(tabla_retraso_dias, rev="columns", verbose = TRUE)
```

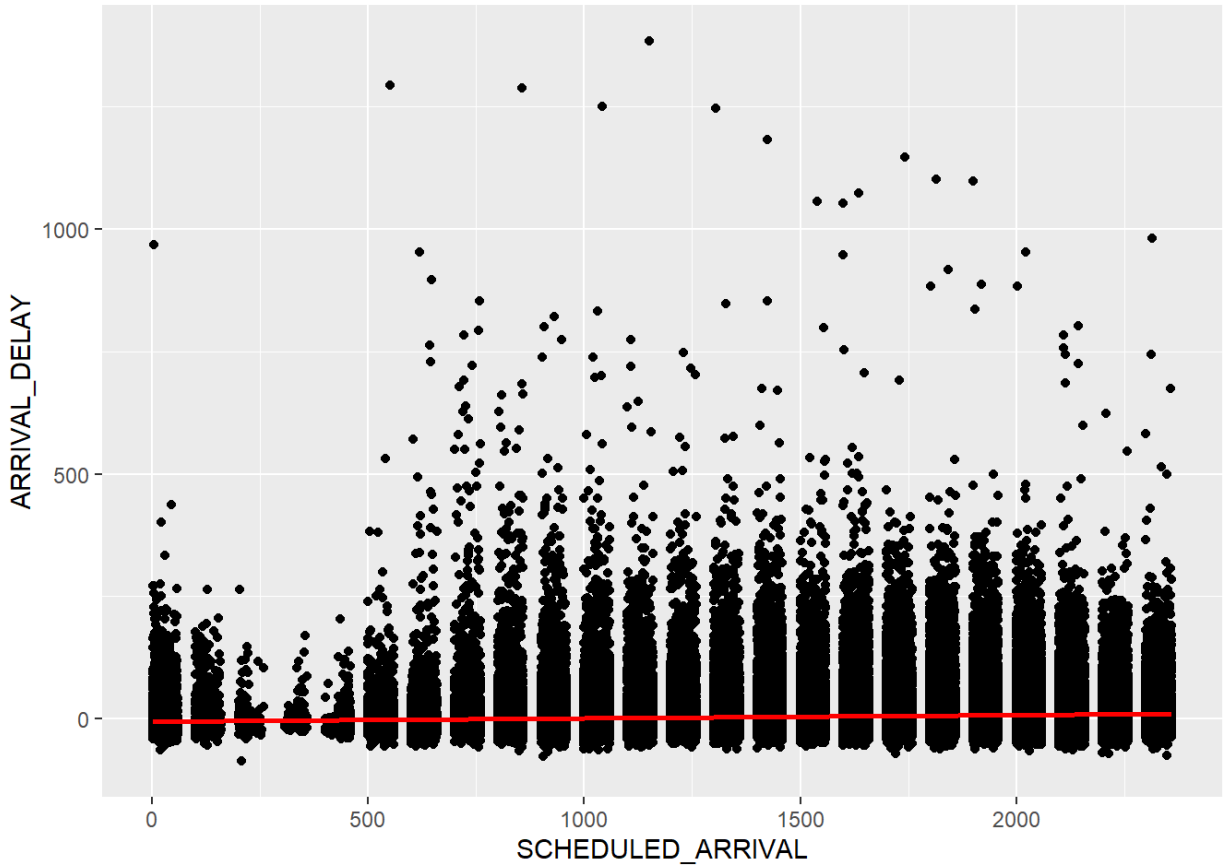
```
## $x
##
##      WEEKEND WEEKDAY
##   NO   86833  122977
##   SI   30184   45656
##
## $data
##
##         WEEKEND WEEKDAY  Total
##   NO      86833  122977 209810
##   SI      30184   45656  75840
##   Total  117017  168633 285650
##
## $p.exposed
##
##           WEEKEND   WEEKDAY     Total
##   NO    0.7420546 0.7292582 0.7345003
##   SI    0.2579454 0.2707418 0.2654997
##   Total 1.0000000 1.0000000 1.0000000
##
## $p.outcome
##
##           WEEKEND   WEEKDAY Total
##   NO    0.4138649 0.5861351     1
##   SI    0.3979958 0.6020042     1
##   Total 0.4096517 0.5903483     1
##
## $measure
##     odds ratio with 95% C.I.
##      estimate   lower    upper
##   NO 1.000000      NA       NA
##   SI 1.068025 1.05011 1.086283
##
## $conf.level
## [1] 0.95
##
## $p.value
##     two-sided
##      midp.exact fisher.exact   chi.square
##   NO         NA           NA           NA
##   SI 2.4869e-14 2.518942e-14 2.612958e-14
##
## $correction
## [1] FALSE
##
## attr(,"method")
## [1] "median-unbiased estimate & mid-p exact CI"
```

El Odds Ratio nos indica que la razón entre la ocurrencia de retraso del vuelo frente a no retraso es 0,93 veces mayor en día laborable y de 1.068 veces superior en fin de semana. Identificamos que no es una diferencia muy pronunciada.

Realizamos el modelo de regresión lineal simple en el que estudiaremos la relación entre el retraso del vuelo en la llegada junto con la hora establecida

```
ggplot(vuelos_reduc, aes(x=SCHEDULED_ARRIVAL, y=ARRIVAL_DELAY)) +
  geom_point() +
  geom_smooth(method=lm , color="red", se=FALSE)
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
retraso_modelo1 <- lm(ARRIVAL_DELAY~SCHEDULED_ARRIVAL, vuelos_reduc)
retraso_modelo1
```

```
##
## Call:
## lm(formula = ARRIVAL_DELAY ~ SCHEDULED_ARRIVAL, data = vuelos_reduc)
##
## Coefficients:
##       (Intercept)  SCHEDULED_ARRIVAL
##         -5.396513           0.006619
```

```
summary(retraso_modelo1)
```

```
##
## Call:
## lm(formula = ARRIVAL_DELAY ~ SCHEDULED_ARRIVAL, data = vuelos_reduc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##  -85.15  -17.65   -8.89    3.46 1381.78
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -5.3965125  0.2291401  -23.55   <2e-16 ***
## SCHEDULED_ARRIVAL  0.0066194  0.0001454   45.53   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.38 on 285648 degrees of freedom
## Multiple R-squared:  0.007204,   Adjusted R-squared:  0.007201
## F-statistic:  2073 on 1 and 285648 DF,  p-value: < 2.2e-16
```

No conseguimos un buen modelo, seguramente porque no tenemos una dependencia lineal entre los valores.

Probamos a incluir en el modelo la variable distancia

```
retraso_modelo2 <- lm(ARRIVAL_DELAY~SCHEDULED_ARRIVAL + DISTANCE, vuelos_reduc)
retraso_modelo2
```

```
##
## Call:
## lm(formula = ARRIVAL_DELAY ~ SCHEDULED_ARRIVAL + DISTANCE, data = vuelos_red
uc)
##
## Coefficients:
##       (Intercept)  SCHEDULED_ARRIVAL           DISTANCE
##         -4.067214           0.006686          -0.001734
```

```
summary(retraso_modelo2)
```

```
##
## Call:
## lm(formula = ARRIVAL_DELAY ~ SCHEDULED_ARRIVAL + DISTANCE, data = vuelos_red
uc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##  -82.81  -17.74   -9.00    3.50 1381.13
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -4.0672143  0.2471748  -16.45   <2e-16 ***
## SCHEDULED_ARRIVAL  0.0066857  0.0001454   45.98   <2e-16 ***
## DISTANCE          -0.0017341  0.0001212  -14.31   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.37 on 285647 degrees of freedom
## Multiple R-squared:  0.007915,   Adjusted R-squared:  0.007909
## F-statistic:  1140 on 2 and 285647 DF,  p-value: < 2.2e-16
```

```
retraso_modelo3 <- lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE + DISTANCE + SCHEDULED_
ARRIVAL, vuelos_reduc)
retraso_modelo3
```

```
##
## Call:
## lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DISTANCE +
##     SCHEDULED_ARRIVAL, data = vuelos_reduc)
##
## Coefficients:
##         (Intercept)  SCHEDULED_DEPARTURE             DISTANCE
##           -6.124261             0.006122            -0.001598
##   SCHEDULED_ARRIVAL
##            0.002543
```

```
summary(retraso_modelo3)
```

```
##
## Call:
## lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DISTANCE +
##     SCHEDULED_ARRIVAL, data = vuelos_reduc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##  -93.18  -17.75   -8.79    3.64 1381.71
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -6.1242606  0.2572649  -23.80   <2e-16 ***
## SCHEDULED_DEPARTURE  0.0061219  0.0002159   28.36   <2e-16 ***
## DISTANCE            -0.0015978  0.0001211  -13.19   <2e-16 ***
## SCHEDULED_ARRIVAL    0.0025431  0.0002060   12.35   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.31 on 285646 degrees of freedom
## Multiple R-squared:  0.0107, Adjusted R-squared:  0.01069
## F-statistic:  1030 on 3 and 285646 DF,  p-value: < 2.2e-16
```

```
#y=-4.216+0.006x1+0.0013x2+0.003x3
```

Revisar Estos factores no tienen impacto o no están relacionados con el retraso.

Modelos de regresión logística:
Utilizaremos el dataframe creada anteriormente para incluir las variables en el formato correcto para la regresión. ¿Está el retraso asociado a los vuelos de fin de semana y a la hora de salida?

```
retraso <- data.frame(retraso, DEPARTURE_HOUR=vuelos_reduc$DEPARTURE_HOUR)
retraso$WEEKEND <- ifelse(retraso$RETRASO=="NO", 0, 1)
retraso[1:2] <- lapply(retraso[1:2], as.factor)
str(retraso)
```

```
## 'data.frame':    285650 obs. of  3 variables:
##  $ RETRASO       : Factor w/ 2 levels "NO","SI": 2 1 1 1 1 1 2 1 1 2 ...
##  $ WEEKEND       : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 2 1 1 2 ...
##  $ DEPARTURE_HOUR: Factor w/ 25 levels " 0"," 1"," 2",..: 9 10 6 16 12 6 12
21 13 20 ...
```

```
retraso_glm1 <- glm(RETRASO~WEEKEND , data=retraso, family=binomial)
```

```
## Warning: glm.fit: algorithm did not converge
```

```
retraso_glm1
```

```
##
## Call:  glm(formula = RETRASO ~ WEEKEND, family = binomial, data = retraso)
##
## Coefficients:
## (Intercept)     WEEKEND1
##      -26.57        53.13
##
## Degrees of Freedom: 285649 Total (i.e. Null);  285648 Residual
## Null Deviance:       330600
## Residual Deviance: 1.657e-06     AIC: 4
```

```
summary(retraso_glm1)
```

```
##
## Call:
## glm(formula = RETRASO ~ WEEKEND, family = binomial, data = retraso)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -2.409e-06  -2.409e-06  -2.409e-06   2.409e-06   2.409e-06
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -26.57     777.48  -0.034    0.973
## WEEKEND1       53.13    1508.88   0.035    0.972
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.3063e+05  on 285649  degrees of freedom
## Residual deviance: 1.6572e-06  on 285648  degrees of freedom
## AIC: 4
##
## Number of Fisher Scoring iterations: 25
```

```
retraso <- data.frame(retraso, DISTANCE=vuelos_reduc$DISTANCE)
str(retraso)
```

```
## 'data.frame':    285650 obs. of  4 variables:
##  $ RETRASO       : Factor w/ 2 levels "NO","SI": 2 1 1 1 1 1 2 1 1 2 ...
##  $ WEEKEND       : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 2 1 1 2 ...
##  $ DEPARTURE_HOUR: Factor w/ 25 levels " 0"," 1"," 2",..: 9 10 6 16 12 6 12
21 13 20 ...
##  $ DISTANCE      : int  674 1407 153 468 1363 184 439 216 268 680 ...
```

```
retraso_glm2 <- glm(RETRASO~DEPARTURE_HOUR + DISTANCE, data=retraso, family=bin
omial)
retraso_glm2
```

```
##
## Call:  glm(formula = RETRASO ~ DEPARTURE_HOUR + DISTANCE, family = binomial,
##     data = retraso)
##
## Coefficients:
##      (Intercept)  DEPARTURE_HOUR 1  DEPARTURE_HOUR 2  DEPARTURE_HOUR 3
##       -1.1378960         0.2646880         0.0804630        -0.0235203
## DEPARTURE_HOUR 4  DEPARTURE_HOUR 5  DEPARTURE_HOUR 6  DEPARTURE_HOUR 7
##        0.1638994         0.1328819         0.2918231         0.3304082
## DEPARTURE_HOUR 8  DEPARTURE_HOUR 9  DEPARTURE_HOUR10  DEPARTURE_HOUR11
##        0.2719672         0.2674248         0.2656829         0.2291342
## DEPARTURE_HOUR12  DEPARTURE_HOUR13  DEPARTURE_HOUR14  DEPARTURE_HOUR15
##        0.1948057         0.2294003         0.2067539         0.2055543
## DEPARTURE_HOUR16  DEPARTURE_HOUR17  DEPARTURE_HOUR18  DEPARTURE_HOUR19
##        0.2370170         0.2739408         0.2889416         0.2173022
## DEPARTURE_HOUR20  DEPARTURE_HOUR21  DEPARTURE_HOUR22  DEPARTURE_HOUR23
##        0.1849256         0.1172167         0.1637671         0.0899838
## DEPARTURE_HOUR24          DISTANCE
##        0.2060898        -0.0001415
##
## Degrees of Freedom: 285649 Total (i.e. Null);  285624 Residual

## Null Deviance:       330600
## Residual Deviance: 330100    AIC: 330100
```

```
summary(retraso_glm2)
```

```
##
## Call:
## glm(formula = RETRASO ~ DEPARTURE_HOUR + DISTANCE, family = binomial,
##     data = retraso)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.8554  -0.8013  -0.7740   1.5698   1.9131
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -1.138e+00  7.046e-02 -16.148  < 2e-16 ***
## DEPARTURE_HOUR 1  2.647e-01  1.252e-01   2.114 0.034503 *
## DEPARTURE_HOUR 2  8.046e-02  2.293e-01   0.351 0.725642
## DEPARTURE_HOUR 3 -2.352e-02  3.623e-01  -0.065 0.948239
## DEPARTURE_HOUR 4  1.639e-01  1.249e-01   1.312 0.189576
## DEPARTURE_HOUR 5  1.329e-01  7.371e-02   1.803 0.071430 .
## DEPARTURE_HOUR 6  2.918e-01  7.177e-02   4.066 4.78e-05 ***
## DEPARTURE_HOUR 7  3.304e-01  7.181e-02   4.601 4.20e-06 ***
## DEPARTURE_HOUR 8  2.720e-01  7.183e-02   3.786 0.000153 ***
## DEPARTURE_HOUR 9  2.674e-01  7.193e-02   3.718 0.000201 ***
## DEPARTURE_HOUR10  2.657e-01  7.183e-02   3.699 0.000217 ***
## DEPARTURE_HOUR11  2.291e-01  7.193e-02   3.186 0.001445 **
## DEPARTURE_HOUR12  1.948e-01  7.198e-02   2.707 0.006799 **
## DEPARTURE_HOUR13  2.294e-01  7.197e-02   3.187 0.001435 **
## DEPARTURE_HOUR14  2.068e-01  7.208e-02   2.869 0.004124 **
## DEPARTURE_HOUR15  2.056e-01  7.194e-02   2.857 0.004270 **
## DEPARTURE_HOUR16  2.370e-01  7.205e-02   3.290 0.001003 **
## DEPARTURE_HOUR17  2.739e-01  7.181e-02   3.815 0.000136 ***
## DEPARTURE_HOUR18  2.889e-01  7.206e-02   4.010 6.07e-05 ***
## DEPARTURE_HOUR19  2.173e-01  7.208e-02   3.015 0.002570 **
## DEPARTURE_HOUR20  1.849e-01  7.261e-02   2.547 0.010867 *
## DEPARTURE_HOUR21  1.172e-01  7.368e-02   1.591 0.111627
## DEPARTURE_HOUR22  1.638e-01  7.542e-02   2.171 0.029902 *
## DEPARTURE_HOUR23  8.998e-02  8.307e-02   1.083 0.278685
## DEPARTURE_HOUR24  2.061e-01  5.215e-01   0.395 0.692714
## DISTANCE         -1.415e-04  7.266e-06 -19.476  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 330629  on 285649  degrees of freedom
## Residual deviance: 330070  on 285624  degrees of freedom
## AIC: 330122
##
## Number of Fisher Scoring iterations: 4
```

```
retraso <- data.frame(retraso, MONTH=vuelos_reduc$MONTH)
str(retraso)
```

```
## 'data.frame':    285650 obs. of  5 variables:
##  $ RETRASO       : Factor w/ 2 levels "NO","SI": 2 1 1 1 1 1 2 1 1 2 ...
##  $ WEEKEND       : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 2 1 1 2 ...
##  $ DEPARTURE_HOUR: Factor w/ 25 levels " 0"," 1"," 2",..: 9 10 6 16 12 6 12
21 13 20 ...
##  $ DISTANCE      : int  674 1407 153 468 1363 184 439 216 268 680 ...
##  $ MONTH         : int  11 4 4 7 2 7 12 2 3 2 ...
```

```
retraso_glm3 <- glm(RETRASO~DEPARTURE_HOUR + DISTANCE + MONTH, data=retraso, fa
mily=binomial)
retraso_glm3
```

```
##
## Call:  glm(formula = RETRASO ~ DEPARTURE_HOUR + DISTANCE + MONTH, family = b
inomial,
##     data = retraso)
##
## Coefficients:
##      (Intercept)  DEPARTURE_HOUR 1  DEPARTURE_HOUR 2  DEPARTURE_HOUR 3
##       -0.9317093         0.2536010         0.0959716        -0.0475269
## DEPARTURE_HOUR 4  DEPARTURE_HOUR 5  DEPARTURE_HOUR 6  DEPARTURE_HOUR 7
##        0.1803717         0.1291169         0.2842065         0.3237933
## DEPARTURE_HOUR 8  DEPARTURE_HOUR 9  DEPARTURE_HOUR10  DEPARTURE_HOUR11
##        0.2644184         0.2612018         0.2583670         0.2202682
## DEPARTURE_HOUR12  DEPARTURE_HOUR13  DEPARTURE_HOUR14  DEPARTURE_HOUR15
##        0.1872997         0.2200206         0.1989120         0.1985692
## DEPARTURE_HOUR16  DEPARTURE_HOUR17  DEPARTURE_HOUR18  DEPARTURE_HOUR19
##        0.2280018         0.2662613         0.2811967         0.2104773
## DEPARTURE_HOUR20  DEPARTURE_HOUR21  DEPARTURE_HOUR22  DEPARTURE_HOUR23
##        0.1774520         0.1101825         0.1603159         0.0904431
## DEPARTURE_HOUR24          DISTANCE             MONTH
##        0.2186166        -0.0001403        -0.0308997
##
## Degrees of Freedom: 285649 Total (i.e. Null);  285623 Residual
## Null Deviance:       330600
## Residual Deviance: 329500    AIC: 329500
```

```
summary(retraso_glm3)
```

```
##
## Call:
## glm(formula = RETRASO ~ DEPARTURE_HOUR + DISTANCE + MONTH, family = binomia
l,
##     data = retraso)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9172  -0.8051  -0.7600   1.5259   1.9588
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -9.317e-01  7.100e-02 -13.122  < 2e-16 ***
## DEPARTURE_HOUR 1  2.536e-01  1.253e-01   2.024 0.043000 *
## DEPARTURE_HOUR 2  9.597e-02  2.295e-01   0.418 0.675801
## DEPARTURE_HOUR 3 -4.753e-02  3.626e-01  -0.131 0.895722
## DEPARTURE_HOUR 4  1.804e-01  1.251e-01   1.442 0.149206
## DEPARTURE_HOUR 5  1.291e-01  7.377e-02   1.750 0.080090 .
## DEPARTURE_HOUR 6  2.842e-01  7.183e-02   3.957 7.60e-05 ***
## DEPARTURE_HOUR 7  3.238e-01  7.187e-02   4.505 6.63e-06 ***
## DEPARTURE_HOUR 8  2.644e-01  7.189e-02   3.678 0.000235 ***
## DEPARTURE_HOUR 9  2.612e-01  7.199e-02   3.628 0.000285 ***
## DEPARTURE_HOUR10  2.584e-01  7.189e-02   3.594 0.000326 ***
## DEPARTURE_HOUR11  2.203e-01  7.199e-02   3.060 0.002216 **
## DEPARTURE_HOUR12  1.873e-01  7.204e-02   2.600 0.009322 **
## DEPARTURE_HOUR13  2.200e-01  7.203e-02   3.055 0.002254 **
## DEPARTURE_HOUR14  1.989e-01  7.214e-02   2.757 0.005826 **
## DEPARTURE_HOUR15  1.986e-01  7.200e-02   2.758 0.005815 **
## DEPARTURE_HOUR16  2.280e-01  7.211e-02   3.162 0.001567 **
## DEPARTURE_HOUR17  2.663e-01  7.187e-02   3.705 0.000212 ***
## DEPARTURE_HOUR18  2.812e-01  7.212e-02   3.899 9.65e-05 ***
## DEPARTURE_HOUR19  2.105e-01  7.214e-02   2.918 0.003526 **
## DEPARTURE_HOUR20  1.775e-01  7.267e-02   2.442 0.014609 *
## DEPARTURE_HOUR21  1.102e-01  7.374e-02   1.494 0.135131
## DEPARTURE_HOUR22  1.603e-01  7.548e-02   2.124 0.033681 *
## DEPARTURE_HOUR23  9.044e-02  8.313e-02   1.088 0.276626
## DEPARTURE_HOUR24  2.186e-01  5.223e-01   0.419 0.675528
## DISTANCE         -1.403e-04  7.274e-06 -19.293  < 2e-16 ***
## MONTH            -3.090e-02  1.250e-03 -24.714  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 330629  on 285649  degrees of freedom
## Residual deviance: 329457  on 285623  degrees of freedom
## AIC: 329511
##
## Number of Fisher Scoring iterations: 4
```

```
#Realizamos una predicción de retraso con nuestro modelo, en el caso de un vuel
o que salga a las 16horas y
#con una distancia de 1000 milas en el mes de junio
newdata = data.frame(DEPARTURE_HOUR = "16" , DISTANCE=1000, MONTH=6)
predict(retraso_glm3, newdata , type="response")
```

```
##         1
## 0.2631941
```

```
newdata2 = data.frame(DEPARTURE_HOUR ="18", DISTANCE=1000, MONTH=6)
predict(retraso_glm3, newdata2 , type="response")
```

```
##         1
## 0.2736389
```

```
table(retraso$DEPARTURE_HOUR)
```

```
##
##     0     1     2     3     4     5     6     7     8     9    10    11    1
2
##  1258   489   121    48   493  9758 18295 17385 17678 16948 17972 17332 1734
3
##    13    14    15    16    17    18    19    20    21    22    23    24
## 17092 16410 17628 16413 18053 15904 16375 13381  9887  6556  2811    20
```

```
#Incluimos la variables aeropuerto.
retraso <- data.frame(retraso, ORIGIN_AIRPORT=vuelos_reduc$ORIGIN_CODE)

# Para que los resultados estén acotados, elegimos los 10 aeropuertos con más t
ráfico:
retraso$ORIGIN_AIRPORT <- ifelse (retraso$ORIGIN_AIRPORT %in%
                                  c("ATL","ORD","DFW","DEN","LAX","SFO","PHX","IAH",
"LAS","MSP"),
                                retraso$ORIGIN_AIRPORT,
                                "OTROS")
str(retraso)
```

```
## 'data.frame':    285650 obs. of  6 variables:
##  $ RETRASO       : Factor w/ 2 levels "NO","SI": 2 1 1 1 1 1 2 1 1 2 ...
##  $ WEEKEND       : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 2 1 1 2 ...
##  $ DEPARTURE_HOUR: Factor w/ 25 levels " 0"," 1"," 2",..: 9 10 6 16 12 6 12
21 13 20 ...
##  $ DISTANCE      : int  674 1407 153 468 1363 184 439 216 268 680 ...
##  $ MONTH         : int  11 4 4 7 2 7 12 2 3 2 ...
##  $ ORIGIN_AIRPORT: chr  "327" "OTROS" "OTROS" "OTROS" ...
```

```
retraso_glm4 <- glm(RETRASO~DEPARTURE_HOUR + DISTANCE + MONTH + ORIGIN_AIRPORT,
data=retraso,
                    family=binomial)
retraso_glm4
```

```
##
## Call:  glm(formula = RETRASO ~ DEPARTURE_HOUR + DISTANCE + MONTH + ORIGIN_AI
RPORT,
##     family = binomial, data = retraso)
##
## Coefficients:
##         (Intercept)     DEPARTURE_HOUR 1     DEPARTURE_HOUR 2
##           -1.299165             0.248077             0.099677
##    DEPARTURE_HOUR 3     DEPARTURE_HOUR 4     DEPARTURE_HOUR 5
##           -0.062661             0.151950             0.097454
##    DEPARTURE_HOUR 6     DEPARTURE_HOUR 7     DEPARTURE_HOUR 8
##            0.256344             0.307172             0.264754
##    DEPARTURE_HOUR 9     DEPARTURE_HOUR10     DEPARTURE_HOUR11
##            0.266330             0.259755             0.212098
##    DEPARTURE_HOUR12     DEPARTURE_HOUR13     DEPARTURE_HOUR14
##            0.183300             0.218004             0.192995
##    DEPARTURE_HOUR15     DEPARTURE_HOUR16     DEPARTURE_HOUR17
##            0.197356             0.227755             0.258516
##    DEPARTURE_HOUR18     DEPARTURE_HOUR19     DEPARTURE_HOUR20
##            0.270313             0.217082             0.186652
##    DEPARTURE_HOUR21     DEPARTURE_HOUR22     DEPARTURE_HOUR23
##            0.127291             0.183636             0.103306
##    DEPARTURE_HOUR24              DISTANCE                MONTH
##            0.192060            -0.000158            -0.031650
##   ORIGIN_AIRPORT392    ORIGIN_AIRPORT393    ORIGIN_AIRPORT458
##            0.353516             0.357102             0.502981
##   ORIGIN_AIRPORT481    ORIGIN_AIRPORT483    ORIGIN_AIRPORT523
##            0.274928             0.551343             0.353967
##   ORIGIN_AIRPORT535    ORIGIN_AIRPORT546    ORIGIN_AIRPORT585
##            0.271894             0.433100             0.437769
## ORIGIN_AIRPORTOTROS
##            0.424004
##
## Degrees of Freedom: 285649 Total (i.e. Null);  285613 Residual
## Null Deviance:       330600
## Residual Deviance: 328800    AIC: 328900
```

```
summary(retraso_glm4)
```

```
##
## Call:
## glm(formula = RETRASO ~ DEPARTURE_HOUR + DISTANCE + MONTH + ORIGIN_AIRPORT,
##     family = binomial, data = retraso)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9792  -0.8091  -0.7578   1.5068   2.1045
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -1.299e+00  7.342e-02 -17.695  < 2e-16 ***
## DEPARTURE_HOUR 1     2.481e-01  1.254e-01   1.978 0.047931 *
## DEPARTURE_HOUR 2     9.968e-02  2.296e-01   0.434 0.664171
## DEPARTURE_HOUR 3    -6.266e-02  3.627e-01  -0.173 0.862857
## DEPARTURE_HOUR 4     1.520e-01  1.252e-01   1.213 0.224967
## DEPARTURE_HOUR 5     9.745e-02  7.399e-02   1.317 0.187782
## DEPARTURE_HOUR 6     2.563e-01  7.205e-02   3.558 0.000374 ***
## DEPARTURE_HOUR 7     3.072e-01  7.207e-02   4.262 2.03e-05 ***
## DEPARTURE_HOUR 8     2.648e-01  7.206e-02   3.674 0.000239 ***
## DEPARTURE_HOUR 9     2.663e-01  7.216e-02   3.691 0.000224 ***
## DEPARTURE_HOUR10     2.598e-01  7.206e-02   3.605 0.000312 ***
## DEPARTURE_HOUR11     2.121e-01  7.217e-02   2.939 0.003295 **
## DEPARTURE_HOUR12     1.833e-01  7.221e-02   2.538 0.011138 *
## DEPARTURE_HOUR13     2.180e-01  7.220e-02   3.019 0.002533 **
## DEPARTURE_HOUR14     1.930e-01  7.232e-02   2.669 0.007614 **
## DEPARTURE_HOUR15     1.974e-01  7.218e-02   2.734 0.006252 **
## DEPARTURE_HOUR16     2.278e-01  7.228e-02   3.151 0.001628 **
## DEPARTURE_HOUR17     2.585e-01  7.207e-02   3.587 0.000334 ***
## DEPARTURE_HOUR18     2.703e-01  7.230e-02   3.739 0.000185 ***
## DEPARTURE_HOUR19     2.171e-01  7.233e-02   3.001 0.002690 **
## DEPARTURE_HOUR20     1.867e-01  7.285e-02   2.562 0.010407 *
## DEPARTURE_HOUR21     1.273e-01  7.392e-02   1.722 0.085080 .
## DEPARTURE_HOUR22     1.836e-01  7.562e-02   2.428 0.015164 *
## DEPARTURE_HOUR23     1.033e-01  8.323e-02   1.241 0.214513
## DEPARTURE_HOUR24     1.921e-01  5.225e-01   0.368 0.713179
## DISTANCE            -1.580e-04  7.392e-06 -21.373  < 2e-16 ***
## MONTH               -3.165e-02  1.254e-03 -25.235  < 2e-16 ***
## ORIGIN_AIRPORT392    3.535e-01  3.032e-02  11.660  < 2e-16 ***
## ORIGIN_AIRPORT393    3.571e-01  2.849e-02  12.534  < 2e-16 ***
## ORIGIN_AIRPORT458    5.030e-01  3.221e-02  15.617  < 2e-16 ***
## ORIGIN_AIRPORT481    2.749e-01  3.443e-02   7.984 1.42e-15 ***
## ORIGIN_AIRPORT483    5.513e-01  3.018e-02  18.267  < 2e-16 ***
## ORIGIN_AIRPORT523    3.540e-01  3.606e-02   9.817  < 2e-16 ***
## ORIGIN_AIRPORT535    2.719e-01  2.751e-02   9.883  < 2e-16 ***
## ORIGIN_AIRPORT546    4.331e-01  3.280e-02  13.205  < 2e-16 ***
## ORIGIN_AIRPORT585    4.378e-01  3.314e-02  13.209  < 2e-16 ***
## ORIGIN_AIRPORTOTROS  4.240e-01  1.990e-02  21.305  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 330629  on 285649  degrees of freedom
## Residual deviance: 328844  on 285613  degrees of freedom
## AIC: 328918
##
## Number of Fisher Scoring iterations: 4
```

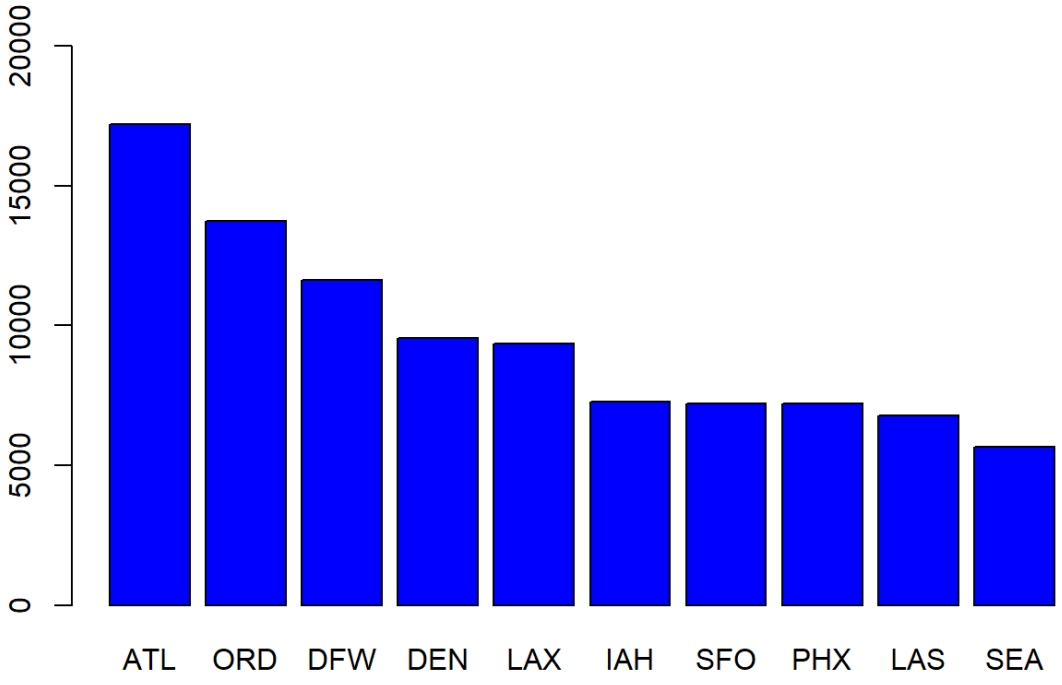
PDTE: *Contraste de hipótesis.La proporción de vuelos retrasados es inferior a la de vuelos en los tiempos establecidos*
Para ello realizaremos un contraste sobre la proporción para muestras grandes

Aplicamos ahora el algoritmo random forest para conocer el aeropuerto que debemos evitar

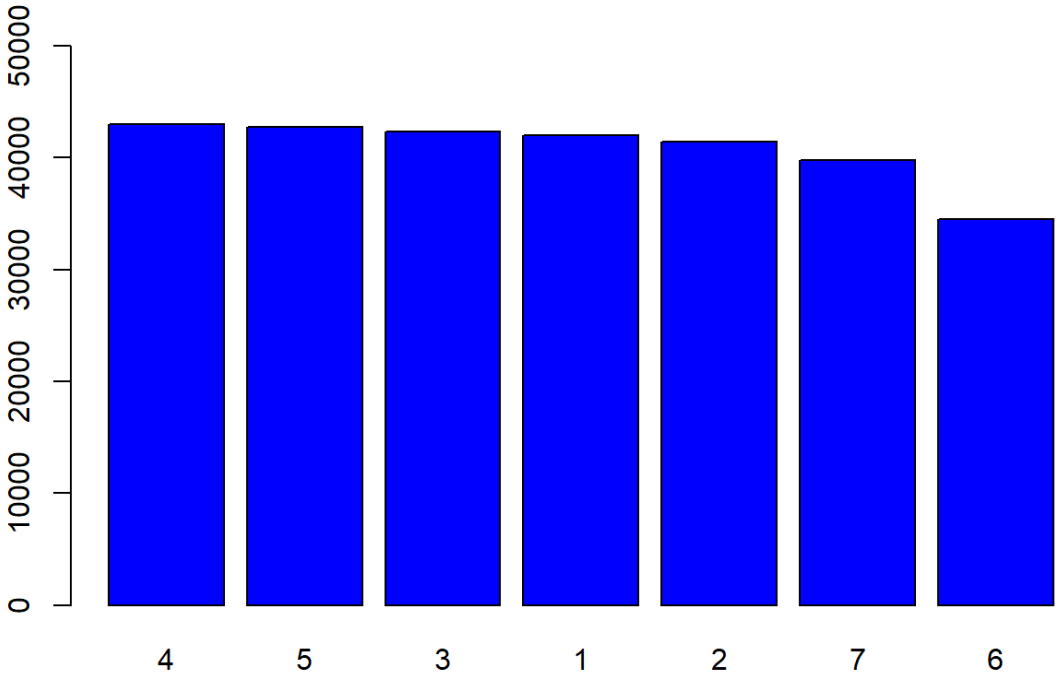
5. Representación de los resultados

Representación de los resultados a partir de tablas y gráficas.

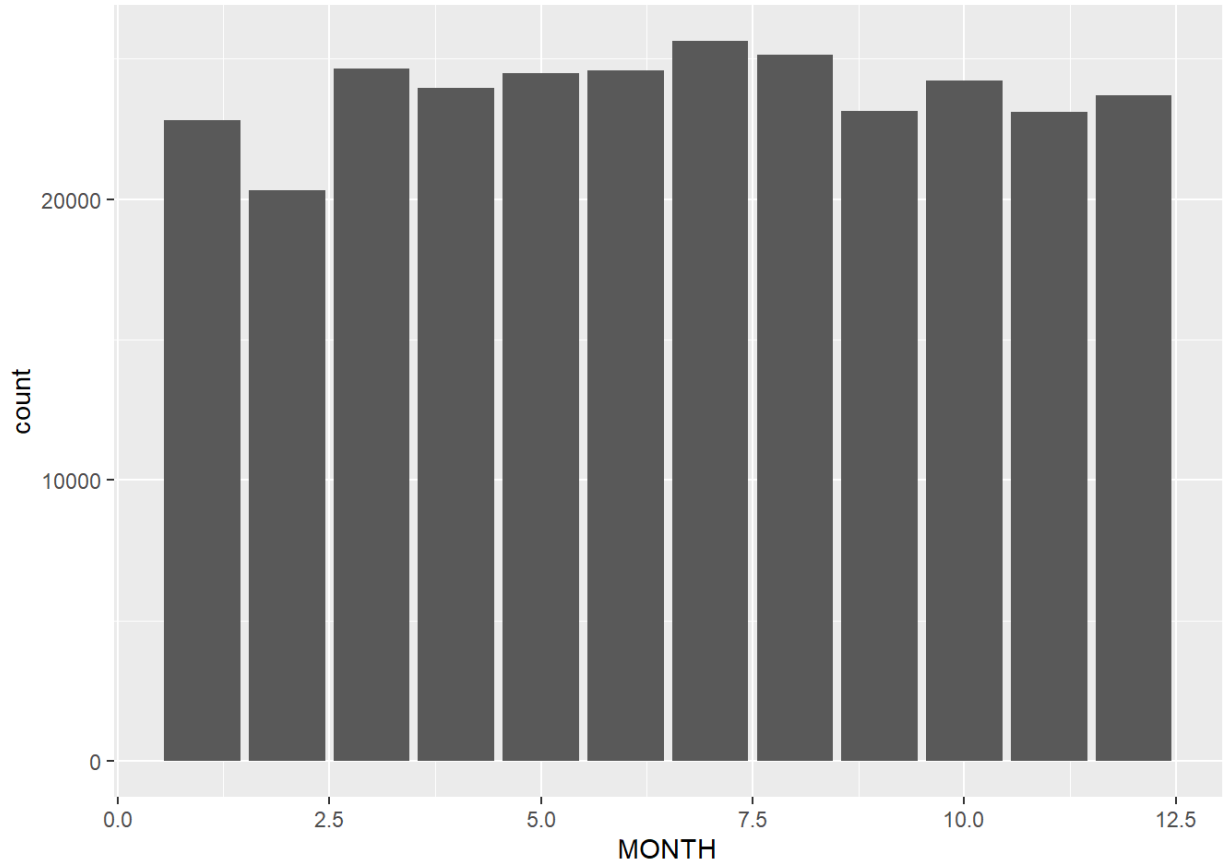
```
#Comprobamos en un gráfico de barras los aeropuertos más populares.
popular_airports <- sort(table(vuelos_reduc$ORIGIN_CODE), decreasing = TRUE )
barplot(popular_airports[1:10], col = "blue", ylim = c(0,20000))
```



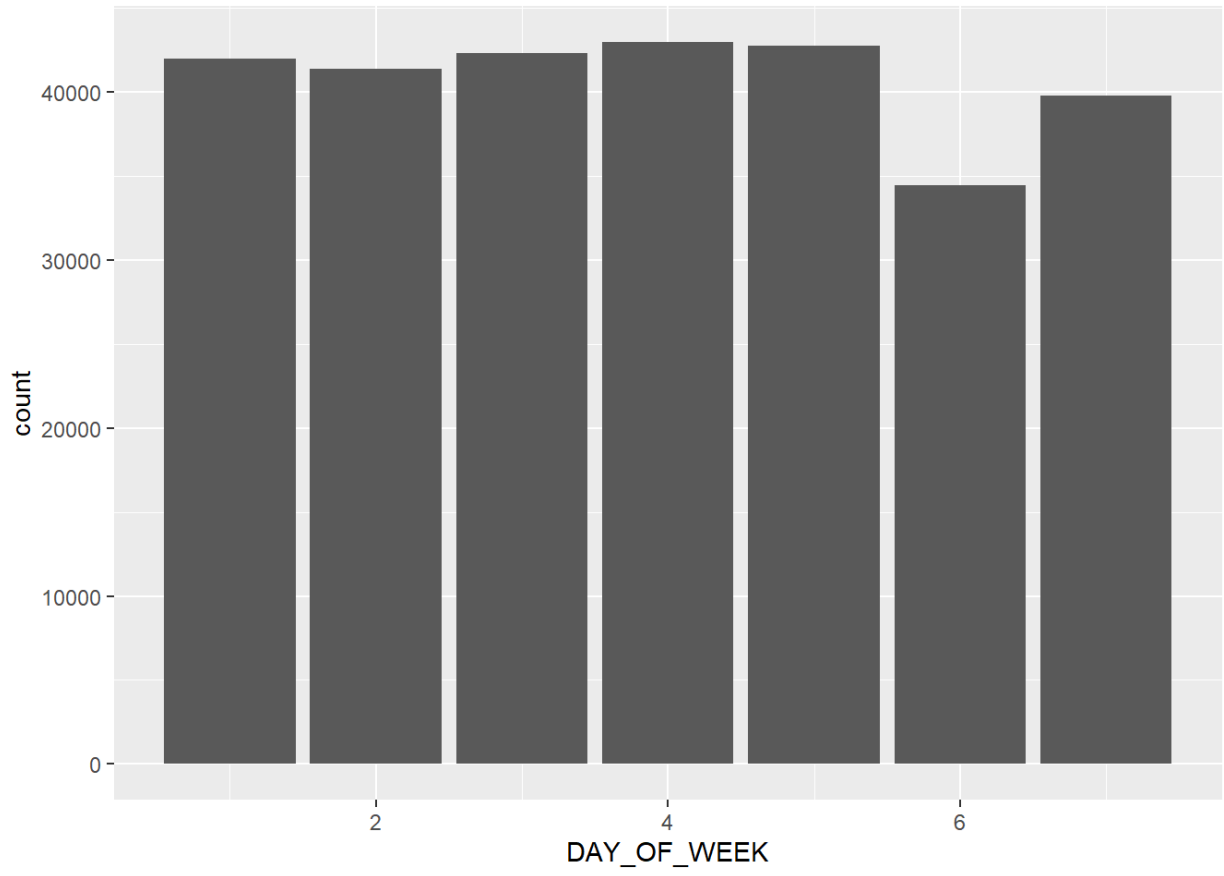
```
#visualización del volumen de vuelos de cada día de la semana

dias_semana <- sort(table(vuelos_reduc$DAY_OF_WEEK), decreasing=TRUE)


barplot(dias_semana, col = "blue", ylim = c(0,50000))
```



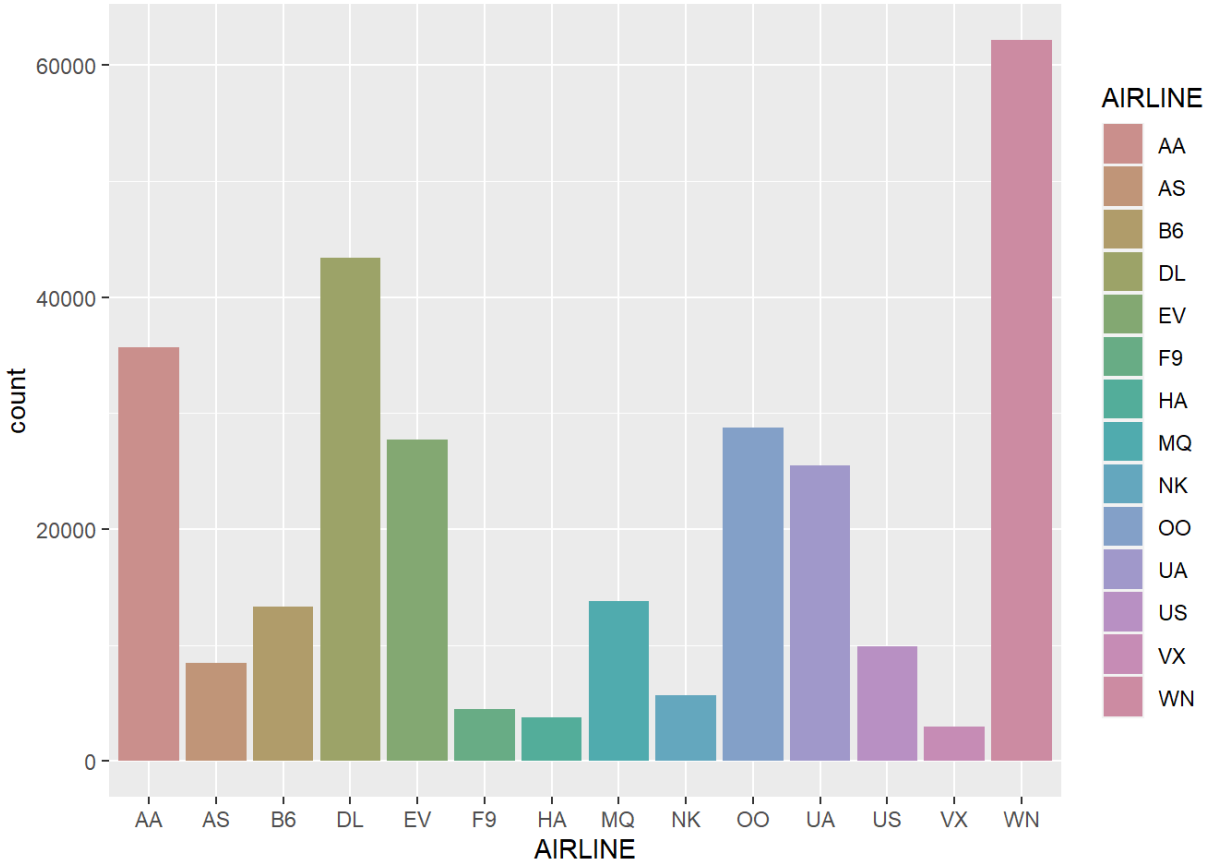

```
#Se comprueba que el tráfico diario en los días laborables es muy similar, vien
dose un cambio de tendencia en el fin de semana, en concreto en el sábado.


ggplot(vuelos_reduc, aes(x=MONTH, fill=MONTH )) +
  geom_bar( ) +
  scale_fill_hue(c = 20)
```



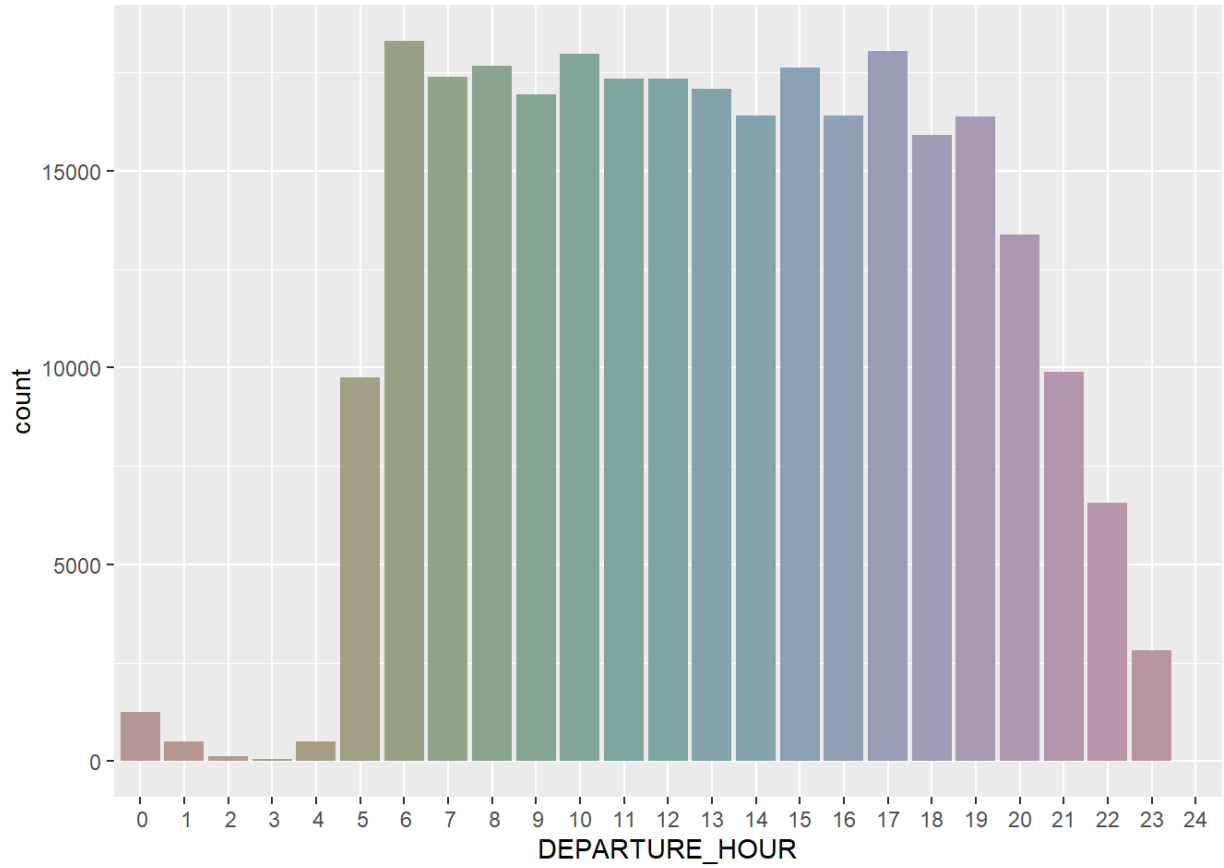

```
ggplot(vuelos_reduc, aes(x=DAY_OF_WEEK, fill=DAY_OF_WEEK )) +
  geom_bar( ) +
  scale_fill_hue(c = 40)
```



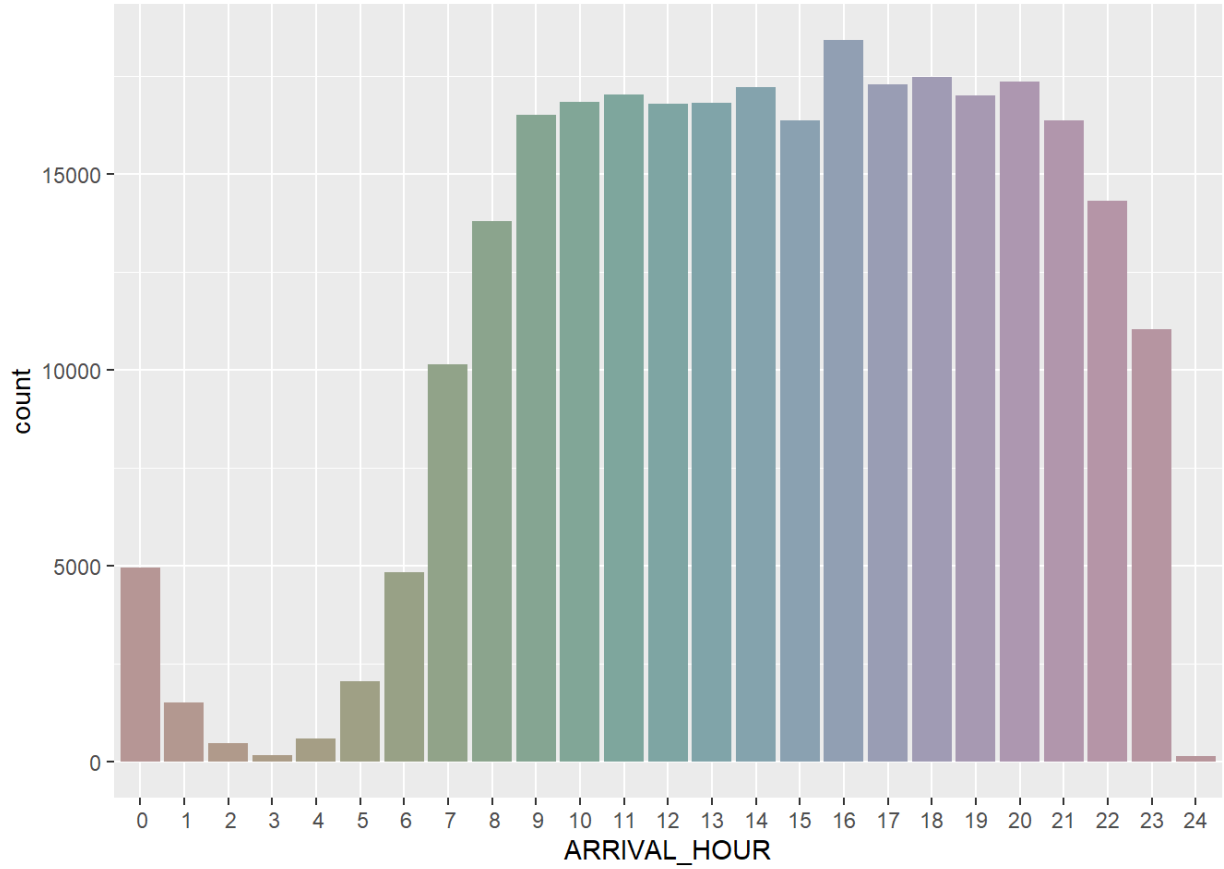
```
#Aerolíneas
ggplot(vuelos_reduc, aes(x=AIRLINE, fill=AIRLINE )) +
  geom_bar( ) +
  scale_fill_hue(c = 40)
```



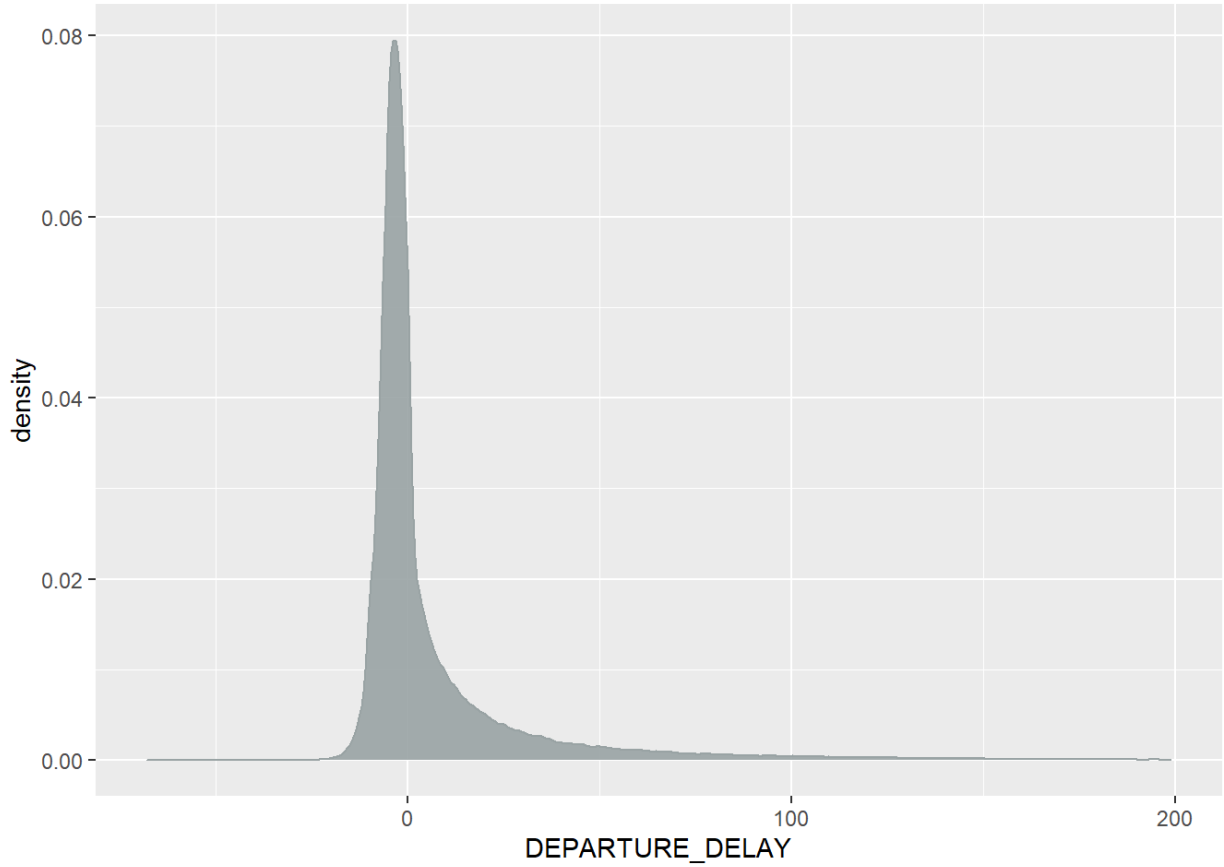

```
#Horas de salida de los vuelos
ggplot(vuelos_reduc, aes(x=DEPARTURE_HOUR, fill=DEPARTURE_HOUR )) +
  geom_bar( ) +
  scale_fill_hue(c = 20) +
  theme(legend.position="none")
```



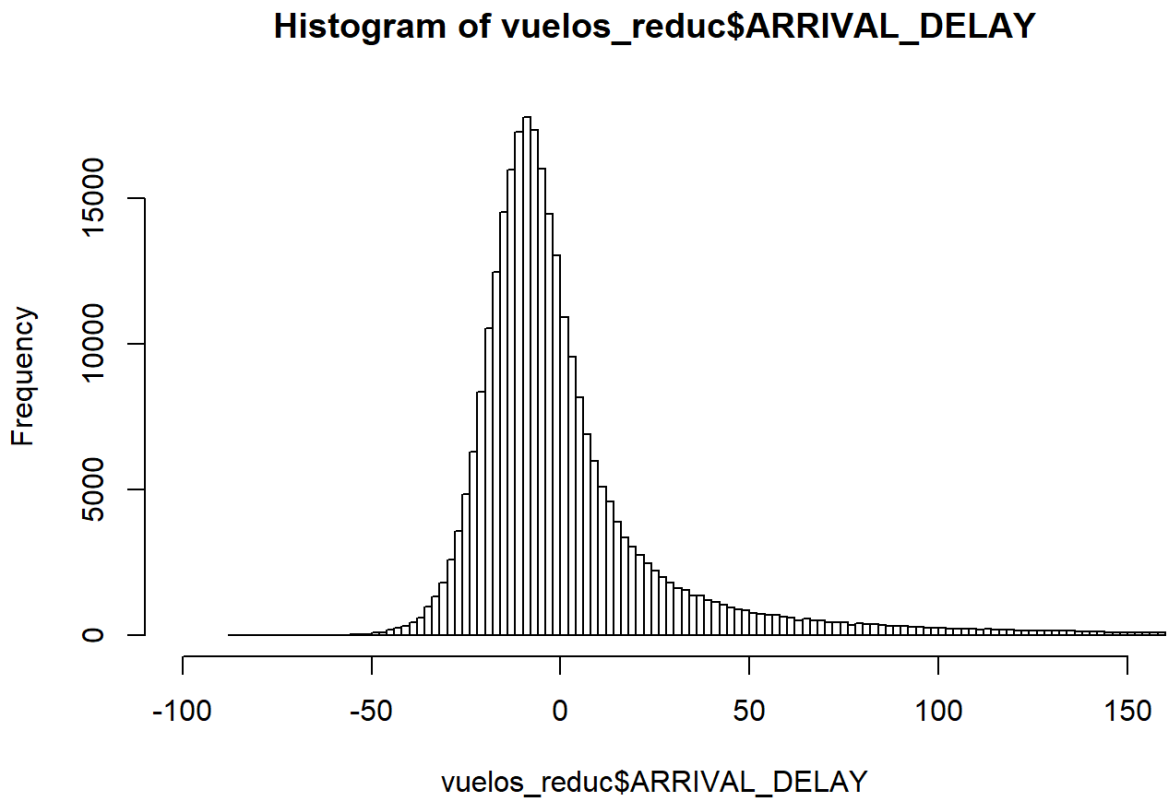

```
#Horas de llegada
ggplot(vuelos_reduc, aes(x=ARRIVAL_HOUR, fill=ARRIVAL_HOUR )) +
  geom_bar( ) +
  scale_fill_hue(c = 20) +
  theme(legend.position="none")
```



```
#Analizamos los vuelos retrasados cuando el retraso es menor de 200 minutos

vuelos_reduc %>%
  filter( DEPARTURE_DELAY<200) %>%
  ggplot( aes(x=DEPARTURE_DELAY)) +
  geom_density(fill="#99A3A4", color="#99A3A4", alpha=0.9)
```




```
hist(vuelos_reduc$ARRIVAL_DELAY,breaks = 1000, xlim = c(-100,150))
```



...

6. Resolución del problema

A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

...

7. Código

Adjuntar el código con el que se ha realizado la limpieza, análisis y representación de los datos.

...

8. Contribuciones al trabajo

Contribuciones	Firma
Investigación previa	IGV, CMGR
Redacción de las respuestas	IGV, CMGR
Desarrollo código	IGZ, CMGR

Referencias:

https://rpubs.com (https://rpubs.com)
https://www.kaggle.com/usdot/flight-delays (https://www.kaggle.com/usdot/flight-delays)