

Practica2

Maite Piedra

3/6/2020

```
#install.packages("knitr")
#install.packages("VIM")
library(knitr)
library(VIM)

## Loading required package: colorspace
## Loading required package: grid
## VIM is ready to use.

## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues

##
## Attaching package: 'VIM'

## The following object is masked from 'package:datasets':
##
##       sleep

setwd('/Users/maitepiedrayera/Developer/uoc/Tipologia_ciclo_vida_datos/data/')
ruta = paste(getwd(),"hotel_bookings.csv", sep = "/")
dh <- read.csv(ruta, header = TRUE, encoding = "latin1" ,sep = ",",
stringsAsFactors = TRUE)
```

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

Como podemos ver nuestro dataset es bastante grande, esta formado por 119390 filas y 32 columnas

```
str(dh)

## 'data.frame': 119390 obs. of 32 variables:
## $ hotel : Factor w/ 2 levels "City Hotel","Resort Hotel": 2 2 2 2 2 2 2 2 ...
## $ is_canceled : int 0 0 0 0 0 0 0 1 1 ...
## $ lead_time : int 342 737 7 13 14 14 0 9 85 75 ...
## $ arrival_date_year : int 2015 2015 2015 2015 2015 2015 2015 2015 2015 ...
## $ arrival_date_month : Factor w/ 12 levels "April","August",...: 6 6 6 6 6 6 6 6 ...
## $ arrival_date_week_number : int 27 27 27 27 27 27 27 27 27 27 ...
## $ arrival_date_day_of_month : int 1 1 1 1 1 1 1 1 1 1 ...
## $ stays_in_weekend_nights : int 0 0 0 0 0 0 0 0 0 ...
## $ stays_in_week_nights : int 0 0 1 1 2 2 2 2 3 3 ...
## $ adults : int 2 2 1 1 2 2 2 2 2 2 ...
## $ children : int 0 0 0 0 0 0 0 0 0 ...
## $ babies : int 0 0 0 0 0 0 0 0 0 ...
## $ meal : Factor w/ 5 levels "BB","FB","HB",...: 1 1 1 1 1 1 1 2 1 3 ...
## $ country : Factor w/ 178 levels "ABW","AGO","AIA",...: 137 137 60 60 60 60 60 137 ...
```

```

## $ market_segment : Factor w/ 8 levels "Aviation","Complementary",...: 4 4 4 3 7 7 4 4
## $ distribution_channel : Factor w/ 5 levels "Corporate","Direct",...: 2 2 2 1 4 4 2 2 4 4 ...
## $ is_repeated_guest : int 0 0 0 0 0 0 0 0 0 0 ...
## $ previous_cancellations : int 0 0 0 0 0 0 0 0 0 0 ...
## $ previous_bookings_not_canceled: int 0 0 0 0 0 0 0 0 0 0 ...
## $ reserved_room_type : Factor w/ 10 levels "A","B","C","D",...: 3 3 1 1 1 1 3 3 1 4 ...
## $ assigned_room_type : Factor w/ 12 levels "A","B","C","D",...: 3 3 3 1 1 1 3 3 1 4 ...
## $ booking_changes : int 3 4 0 0 0 0 0 0 0 0 ...
## $ deposit_type : Factor w/ 3 levels "No Deposit","Non Refund",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ agent : Factor w/ 334 levels "1","10","103",...: 334 334 334 157 103 103 334 ...
## $ company : Factor w/ 353 levels "10","100","101",...: 353 353 353 353 353 353 353 ...
## $ days_in_waiting_list : int 0 0 0 0 0 0 0 0 0 0 ...
## $ customer_type : Factor w/ 4 levels "Contract","Group",...: 3 3 3 3 3 3 3 3 3 3 ...
## $ adr : num 0 0 75 75 98 ...
## $ required_car_parking_spaces : int 0 0 0 0 0 0 0 0 0 0 ...
## $ total_of_special_requests : int 0 0 0 0 1 1 0 1 1 0 ...
## $ reservation_status : Factor w/ 3 levels "Canceled","Check-Out",...: 2 2 2 2 2 2 2 2 2 1 ...
## $ reservation_status_date : Factor w/ 926 levels "2014-10-17","2014-11-18",...: 122 122 123 122 ...

```

A continuación describimos cada una de las columnas que conforman nuestro dataset

- 1.- Hotel :Hotel (H1 = Resort Hotel or H2 = City Hotel)
- 2.- Is_canceled: Indica si la reserva fue cancelada (1) o no (0)
- 3.- Lead_time: Número de días transcurridos entre la fecha de entrada de la reserva en sistema de gestión de reservas (PMS) y la fecha de llegada
- 4.- Arrival_date_year: Año
- 5.- Arrival_date_month: Mes
- 6.- Arrival_date_week_number: Día de la semana
- 7.- Arrival_date_day_of_month: Día del mes en el que entró en el hotel
- 8.- Stays_in_weekend_nights: Número de noches de fin de semana (sábado o domingo) que el huésped se hospedó o reservó para quedarse en el hotel
- 9.- Stays_in_week_nights: Número de noches entre semana (Lunes a viernes) que el huésped se hospedó o reservó para quedarse en el hotel
- 10.- Adults: Número de adultos.
- 11.- Children: Número de niños.
- 12.- Babies: Número de bebés.
- 13.- Meal: Type of meal booked: Las categorías se presentan en paquetes estándar de comidas de hospitalidad: Indefinido / SC - sin paquete de comidas; BB - Alojamiento y desayuno; HB - Media pensión (desayuno y otra comida, generalmente cena); FB - Pensión completa (desayuno, almuerzo y cena)
- 14.- Country: País de origen.
- 15.- Market_segment: Designación del segmento de mercado. En categorías, el término “TA” significa “Agentes de viajes” y “TO” significa “Operadores turísticos”.
- 16.- Distribution_channel: Canal de distribución de reservas. El término “TA” significa “Agentes de viajes” y “TO” significa “Operadores turísticos”.
- 17.-Is_repeated_guest: Valor que indica si el nombre de la reserva era de un huésped repetido (1) o no (0).
- 18.- Previous_cancellations: Número de reservas anteriores que el cliente canceló antes de la reserva actual.

- 19.- Previous_bookings_not_canceled: Número de reservas anteriores no canceladas por el cliente antes de la reserva actual.
- 20.- Reserved_room_type: Código de tipo de habitación reservado. El código se presenta en lugar de la designación por razones de anonimato.
- 21.- Assigned_room_type: Código para el tipo de habitación asignada a la reserva. A veces, el tipo de habitación asignada difiere del tipo de habitación reservada debido a razones de operación del hotel (por ejemplo, sobreventa) o por solicitud del cliente. El código se presenta en lugar de la designación por razones de anonimato.
- 22.- Booking_changes:Número de cambios / modificaciones realizados en la reserva desde el momento en que se ingresó en el PMS (sistema de gestión de reservas) hasta el momento del check-in o cancelación.
- 23.- Deposit_type: Indicación de si el cliente realizó un depósito para garantizar la reserva. Esta variable puede asumir tres categorías: Sin depósito: no se realizó ningún depósito; Sin reembolso: se realizó un depósito por el valor del costo total de la estadía; Reembolsable: se realizó un depósito con un valor por debajo del costo total de la estadía.
- 24.- Agent: Identificación de la agencia de viajes que realizó la reserva.
- 25.-Company: Identificación de la empresa / entidad que realizó la reserva o responsable de pagar la reserva. Se presenta la identificación en lugar de la designación por razones de anonimato.
- 26.- Days_in_waiting_list: Número de días que la reserva estuvo en la lista de espera antes de ser confirmada al cliente.
- 27.- Customer_type: Tipo de reserva, asumiendo una de cuatro categorías: Contrato: cuando la reserva tiene una asignación u otro tipo de contrato asociado; Grupo: cuando la reserva está asociada a un grupo; Transitoria: cuando la reserva no forma parte de un grupo o contrato, y no está asociada a otra reserva transitoria; Parte transitoria: cuando la reserva es transitoria, pero está asociada a al menos otra reserva transitoria
- 28.- Adr: Tarifa diaria promedio según se define dividiendo la suma de todas las transacciones de alojamiento por el número total de noches de estadía.
- 29.- Required_car_parking_spaces: Número de plazas de aparcamiento requeridas por el cliente.
- 30.-Total_of_special_requests: Número de solicitudes especiales realizadas por el cliente (por ejemplo, cama doble o piso alto).
- 31.- Reservation_status: Último estado de la reserva, asumiendo una de tres categorías: Cancelada: la reserva fue cancelada por el cliente; check-out: el cliente se ha registrado pero ya se ha ido; No-Show: el cliente no hizo el check-in e informó al hotel del motivo.
- 32.- Reservation_status_date: Fecha en la que se estableció el último estado. Esta variable se puede usar junto con el Estado de reserva para comprender cuándo se canceló la reserva o cuándo el cliente realizó el check-out del hotel
- Con este dataset, responder a preguntas como: ¿Cual es el mejor época del año para hacer una reserva? ¿Cual es el número de días óptimo para que la estadía en un hotel me salga rentable?

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

A partir de todas las variables que tenemos se nos plantea cuales son las mas indicadas para resolver nuestras preguntas, como tenemos muchas variables, vamos a eliminar aquellas que creemos que no tienen especial relevancia para las preguntas que nos ocupan, y luego procederemos a limpiar el dataset.

En nuestro caso nos quedamos con 14 variables que consideramos importantes para nuestro estudio y eliminamos del dataset las siguientes: Lead_time, Arrival_date_year, meal, Distribution_channel, Previous_cancellations, Previous_bookings_not_canceled, Reserved_room_type, Assigned_room_type, Book-

```
ing_changes, Agent, Company, Days_in_waiting_list, Customer_type, Required_car_parking_spaces, Total_of_special_requests, Reservation_status, Reservation_status_date, country y market_segment.
```

```
# eliminamos las siguientes columnas
dh_reducido = dh[, -c(3,4,13:16,18:22,24:26,29:32 )]

# venos de que tipo es cada variable en nuestro dataset
tipo_dato = sapply(dh_reducido, function(x) class(x))
kable(data.frame(variables = names(dh_reducido), tipo_variable = as.vector(tipo_dato)))
```

variables	tipo_variable
hotel	factor
is_canceled	integer
arrival_date_month	factor
arrival_date_week_number	integer
arrival_date_day_of_month	integer
stays_in_weekend_nights	integer
stays_in_week_nights	integer
adults	integer
children	integer
babies	integer
is_repeated_guest	integer
deposit_type	factor
customer_type	factor
adr	numeric

```
# convertimos a numeric aquellos valores que son enteros, para poder trabajar mas facilmente con ellos.
dh_reducido[, c(2,4:11)] <- sapply(dh_reducido[, c(2,4:11)], as.numeric)
tipo_dato_reducido = sapply(dh_reducido, function(x) class(x))
kable(data.frame(variables = names(dh_reducido), tipo_variable = as.vector(tipo_dato_reducido)))
```

variables	tipo_variable
hotel	factor
is_canceled	numeric
arrival_date_month	factor
arrival_date_week_number	numeric
arrival_date_day_of_month	numeric
stays_in_weekend_nights	numeric
stays_in_week_nights	numeric
adults	numeric
children	numeric
babies	numeric
is_repeated_guest	numeric
deposit_type	factor
customer_type	factor
adr	numeric

```
# el resumende nuestro dataset seria el siguiente
summary(dh_reducido)
```

```
##          hotel      is_canceled   arrival_date_month
##  City Hotel :79330    Min.   :0.0000   August :13877
```

```

##  Resort Hotel:40060    1st Qu.:0.0000    July     :12661
##                           Median :0.0000    May      :11791
##                           Mean    :0.3704    October:11160
##                           3rd Qu.:1.0000    April   :11089
##                           Max.    :1.0000    June    :10939
##                                         (Other):47873
##  arrival_date_week_number arrival_date_day_of_month stays_in_weekend_nights
##  Min.    : 1.00          Min.    : 1.0          Min.    : 0.0000
##  1st Qu.:16.00          1st Qu.: 8.0          1st Qu.: 0.0000
##  Median :28.00          Median :16.0          Median : 1.0000
##  Mean   :27.17          Mean   :15.8          Mean   : 0.9276
##  3rd Qu.:38.00          3rd Qu.:23.0          3rd Qu.: 2.0000
##  Max.   :53.00          Max.   :31.0          Max.   :19.0000
##
##  stays_in_week_nights    adults       children       babies
##  Min.    : 0.0          Min.    : 0.000    Min.    : 0.000000
##  1st Qu.: 1.0          1st Qu.: 2.000    1st Qu.: 0.0000  1st Qu.: 0.000000
##  Median : 2.0          Median : 2.000    Median : 0.0000  Median : 0.000000
##  Mean   : 2.5          Mean   : 1.856    Mean   : 0.1039  Mean   : 0.007949
##  3rd Qu.: 3.0          3rd Qu.: 2.000    3rd Qu.: 0.0000  3rd Qu.: 0.000000
##  Max.   :50.0          Max.   :55.000    Max.   :10.0000  Max.   :10.000000
##                                         NA's    :4
##  is_repeated_guest    deposit_type        customer_type
##  Min.    :0.000000  No Deposit:104641  Contract      : 4076
##  1st Qu.:0.000000  Non Refund: 14587  Group        :  577
##  Median :0.000000  Refundable:   162  Transient    :89613
##  Mean   :0.03191   Customer: 25124  Transient-Party:25124
##  3rd Qu.:0.000000
##  Max.   :1.000000
##
##  adr
##  Min.    : -6.38
##  1st Qu.: 69.29
##  Median : 94.58
##  Mean   : 101.83
##  3rd Qu.: 126.00
##  Max.   :5400.00
##

```

3.- Limpieza de los datos.

Analizaremos uno por uno los datos devueltos por nuestro summary - Hotel: los valores que toma la variable con City Hotel y Resort Hotel, osea son los valores esperados. - is_canceled: vemos que el min valor es 0 y el max 1 y esa variable se mueve entre esos valores, odea esta correcto. - arrival_date_month: este valor prodriamos normalizarlo, asignando un numero a cada mes (1 => Enero.. 12=> Diciembre), asi nos aseguramos que los valores se mueven entre 1 y 12 y con eso eliminamos la posibilidad de que haya un mes mal escrito o algo por el estilo. - arrival_date_week_number : tenemos que el menor valor es 1 (llegar en la primera semana) y el mayor valor 53, sabiendo que un año tiene 53 semanas lo podemos considerar correcto. - arrival_date_day_of_month: En principio estaria bien , ya que se mueve entre 1 y 31, osea los dias de los meses. - stays_in_weekend_nights: numero de noches que caen en fin de semana, tenemos un valor sospechoso, que puede ser un posible outlier, pasar 19 fines de semana en un hotel es bastante raro. - stays_in_week_nights: pasa lo mismo que en el caso anterior, seria una variable a examinar un poco mas a fondo, ya que tenemos un valor bastante inusual (50). - adults: tenemos un valor un poco extraño igual que en los casos anteriores que es un posible outlier, 55 adultos en una reserva, en este caso cabe la posibilidad

que se auna reserva de grupo, pero igualmente es algo que hay que mirar. - children: Lo mismo con esta variable, es raro una reserva donde hayan 10 niños, aunque puede ser de una reserva de grupo, y ademas vemos que tenemos valores perdidos

- babies: pasa lo mismo que con children.

- market_segment - is_repeated_guest : En principio estaria correcto, toma valores entre 0 y 1. - deposit_type: estaria correcto, pues solo hay tres tipos de deposito. - customer_type: Estaría correcto, pues tenemos 4 tipos de cliente - adr: este valor equivale a la tarifa diaria promedio, y es una variable que tenemos que mirar pues toma valores negativos y valores 0 cosa que no deberia ser ademas su valor mas alto es de 5400, que puede ser un posible outlier.

Una vez analizadas las variables una por una, vamos a realizar los cambios que dejimos anteriormente.

```
dh_reducido$arrival_date_month <- factor(dh_reducido$arrival_date_month,
                                             levels = c("January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"),
                                             labels = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12))
```

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

Es normal que al trabajar con un gran volumen de datos tengamos datos que esten desaparecidos o incluso que tengan valores erroneos.

Los valores 0 o vacios (NA) pueden aparecer porque se desconoce el dato y se dejo vacio en su momento, o porque la persona o algoritmos que tenia que introducir los datos en ese momento tuvo algun problema, en el caso de una persona fisica, se puede deber a un olvido, y en el caso de un programa/algoritmo un fallo en el método de recoleccion de datos o de escritura.

En cualquier caso tenemos valores missing en nuestro dataset. La forma mas común de solventar este problema es:

- Eliminar dichas filas de datos, con la consecuente perdida de información. En nuestro caso son solo 4 filas dentro de 119390 filas que tenemos, osea la perdida de información es casi inexistente, pero en datasets donde el número de datos faltantes es bastante mayor no sería una buena solución.

- Rellenar los valores faltantes con el valor de la media, de todos los valores obtenidos, no es una técnica muy optima pero no eliminamos las filas.
- Rellenar los valores faltantes usando el algoritmo Knn, osea usa a los vecinos mas cercanos para predecir que valor debería ser ese valor faltante.

En nuestro caso, aunque son solo 4 valores faltantes vamos a utilizar la técnica de los vecinos mas cercanos, ya que siempre es mejor trabajar con datos simulados, que con datos vacios y por supuesto mejor que eliminar las filas, aunque sean 4.

```
# rellenamos los valores faltantes.
dh_reducido$children = kNN(dh_reducido)$children

summary(dh_reducido$children)

##      Min. 1st Qu. Median   Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.1039  0.0000 10.0000
```

3.2. Identificación y tratamiento de valores extremos.

Vamos a explicar como podemos detectar los valores extremos a parte e graficamente con el boxplot como lo podemos saber numericamente. Para ello usamos la funcion summary que nos muestra los quartiles de los variables.

Sabemos que entre Q1 y Q3 se encuentra el 50% de los valores obtenidos en el estudio y esta distancia de llama distancia Intercuartilica (IQR).

Un valor atípico leve se define como aquel que esta 1,5 veces el rango intercuartilico por debajo de Q1 o por encima de Q3 Y un valor atípico extremo se define como aquel que esta 3 veces el rango intercuartilico por debajo de Q1 o por encima de Q3

Cuando la mediana esta muy distante de la media (casi el doble) podemos decir que pasa algo raro, osea que hay valores tan altos que estan tragiversando el estudio.

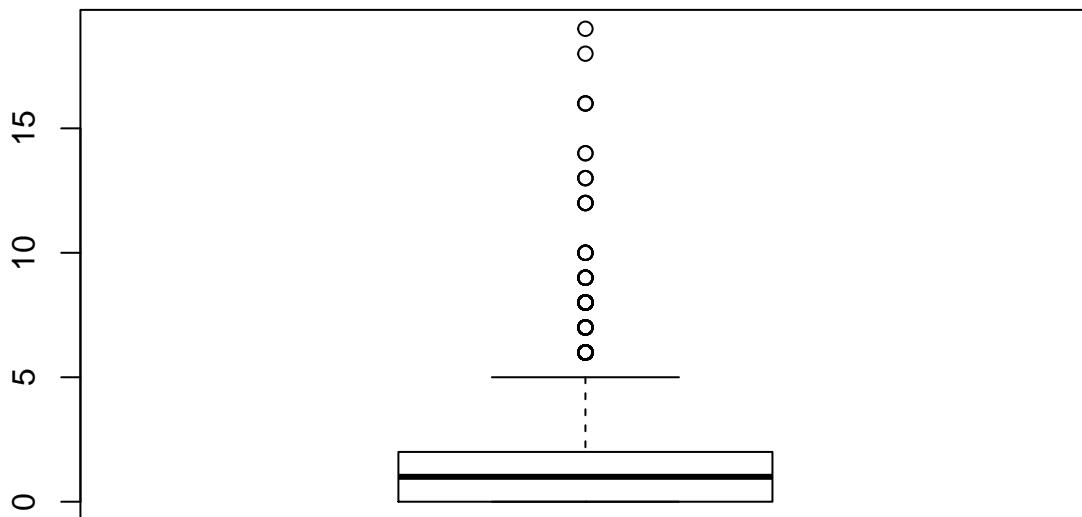
Ahora aplicamos esto para cada caso.

```
attach(dh_reducido)

#Creamos una funcion que nos haga los calculos para no repetirlos con cada caso.
outlierReplace = function(entrada){
  iqr.valor = IQR(entrada)
  cuantiles = quantile(entrada, c(0.25, 0.50, 0.75))
  # Todo valor inferior a este se considera outlier
  outlier_min = as.numeric(cuantiles[1])-1.5*iqr.valor
  # Todo valor superior a este se considera outlier
  outlier_max = as.numeric(cuantiles[3])+1.5*iqr.valor
  #ahora reemplazamos el dato por la media
  entrada[entrada < outlier_min] = round(mean(entrada))
  entrada[entrada > outlier_max] = round(median(entrada))
  return(entrada)
}

boxplot(dh_reducido$stays_in_weekend_nights, main="Estadía de dias los fines de semana")
```

Estadía de días los fines de semana

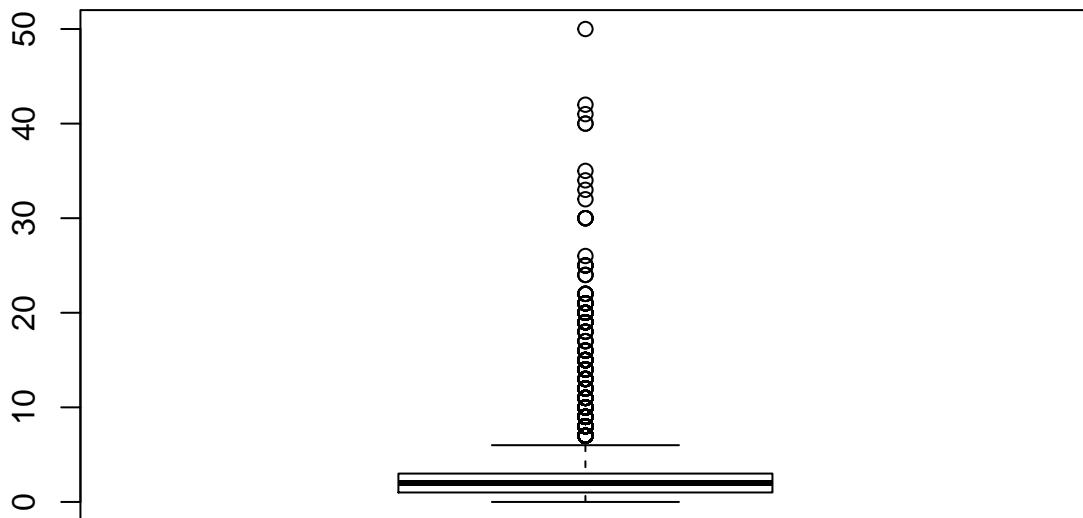


```
summary(dh_reducido$stays_in_weekend_nights)

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
## 0.0000 0.0000 1.0000 0.9276 2.0000 19.0000

boxplot(dh_reducido$stays_in_weekend_nights, main="Estadía de días los fines de semana")
```

Estadía de días los fines de semana

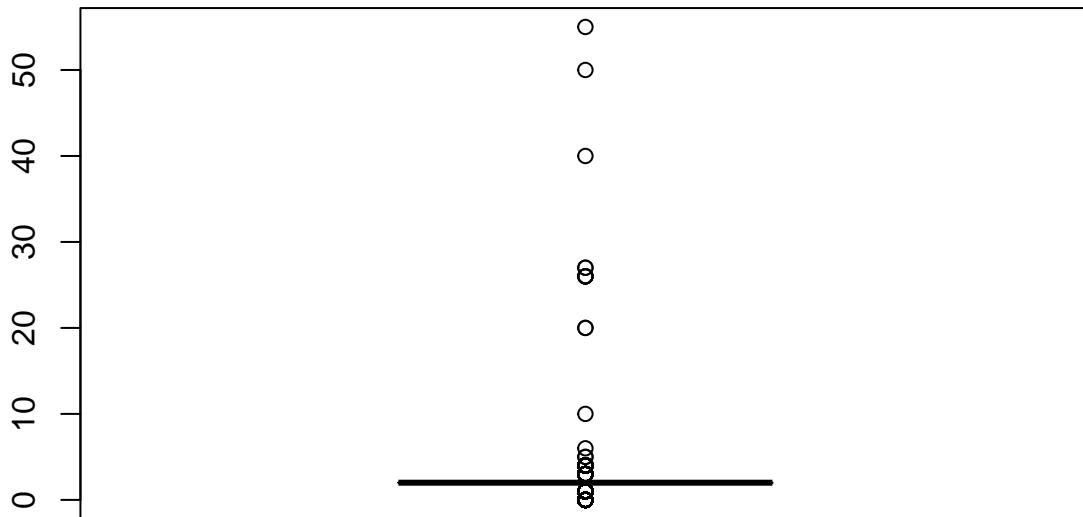


```
summary(dh_reducido$stays_in_week_nights)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##      0.0    1.0    2.0    2.5    3.0    50.0
```

```
boxplot(dh_reducido$adults, main="Adultos")
```

Adultos

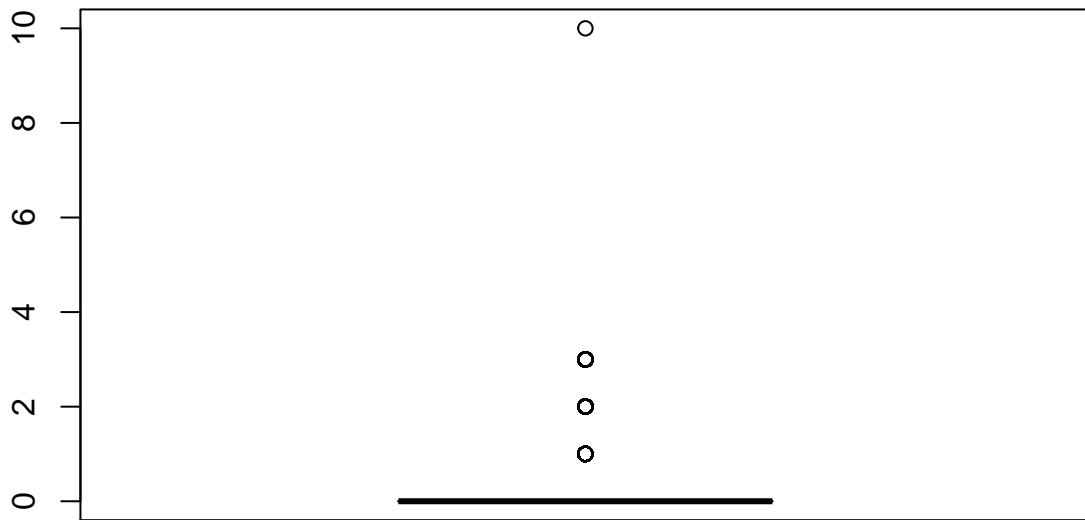


```
summary(dh_reducido$adults)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##      0.000  2.000  2.000   1.856  2.000  55.000
```

```
boxplot(dh_reducido$children, main="Niños")
```

Niños

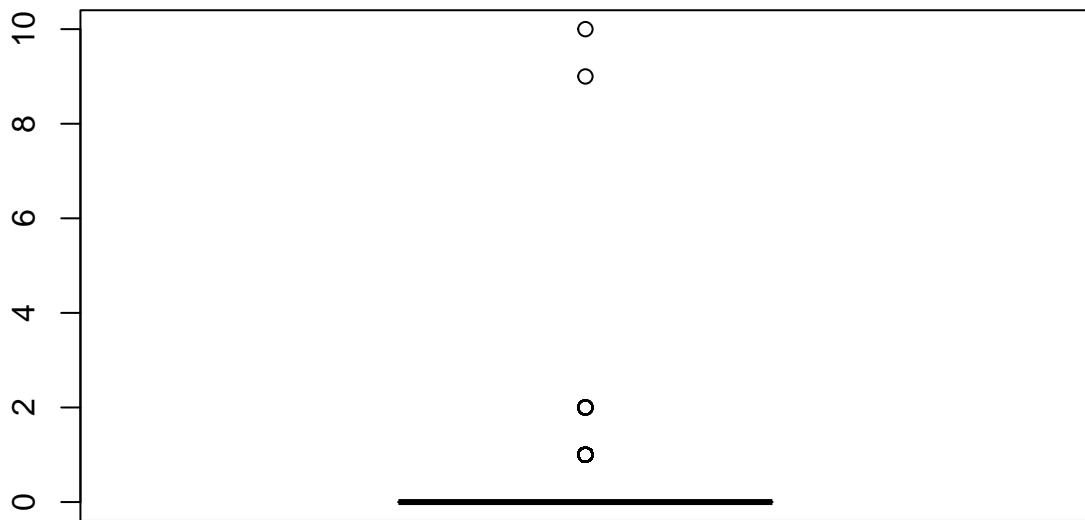


```
summary(dh_reducido$children)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
## 0.0000 0.0000 0.0000 0.1039 0.0000 10.0000
```

```
boxplot(dh_reducido$babies, main="Bebes")
```

Bebes

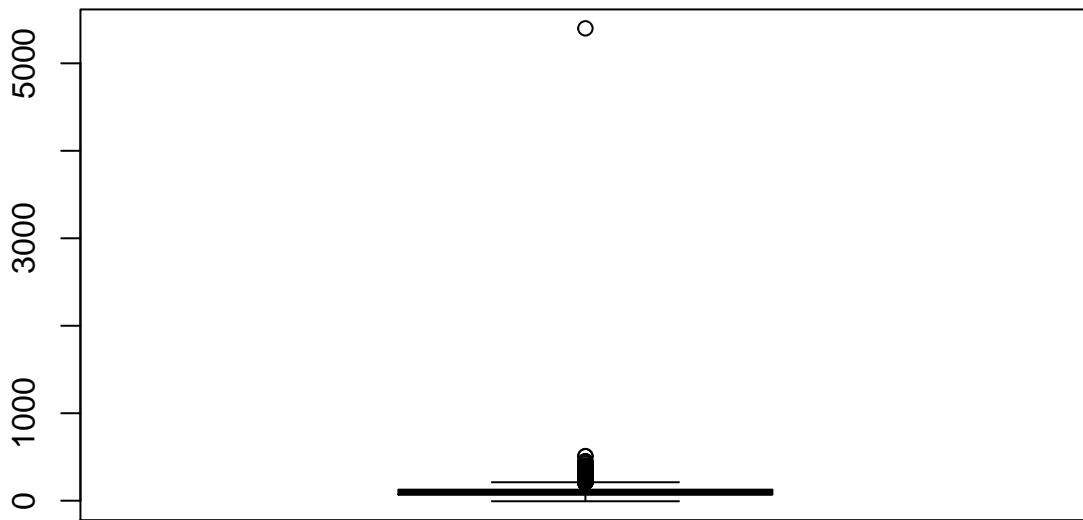


```
summary(dh_reducido$babies)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
## 0.000000 0.000000 0.000000 0.007949 0.000000 10.000000
```

```
boxplot(dh_reducido$adr, main="tarifa diaria promedio")
```

tarifa diaria promedio



```
summary(dh_reducido$adr)

##      Min. 1st Qu. Median   Mean 3rd Qu.   Max.
##    -6.38    69.29   94.58 101.83 126.00 5400.00
```

En nuestro caso a pesar de haber creado una función para eliminar los valores outlier, no la hemos necesitado, puesto que tenemos valores outlier pero como bien dice la teoría la media y la mediana no distan mucho entre si, como vemos en el summary de cada variable, por lo tanto no consideramos los valores outlier peligroso o dañinos para nuestro estudio, es decir si los corregimos el resultado final con los valores outlier o sin ellos no será muy diferente. Por lo tanto los dejamos.

```
# Extraemos en un CSV los datos finales que usaremos.
write.csv(dh_reducido, "datos_hoteles.csv")
```

4.- Análisis de los datos.

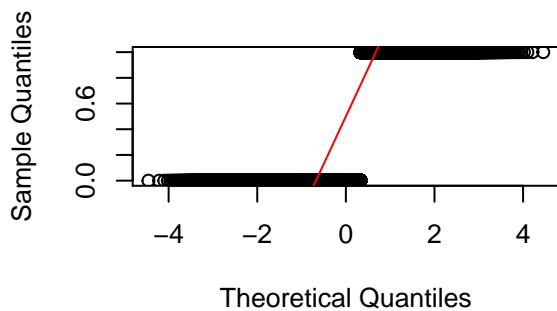
4.1. Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).

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

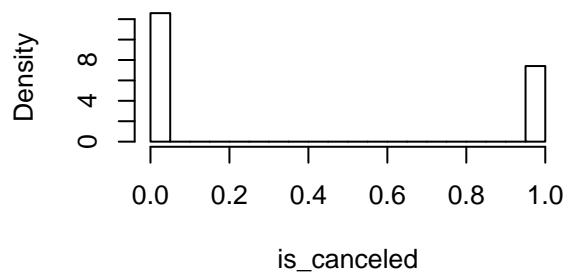
```
comprobar_normalidad = function(datos){
  for(i in 1:ncol(datos)){
    if(is.numeric(datos[,i])){
      qqnorm(datos[,i], main = paste("Normal Q-Q plot para ", colnames(datos)[i]))
      qqline(datos[,i], col="red")
      hist(datos[,i],
            main = paste("Histograma para ", colnames(datos)[i]),
            xlab = colnames(datos)[i], freq = FALSE)
    }
  }
}

par(mfrow=c(2,2))
comprobar_normalidad(dh_reducido)
```

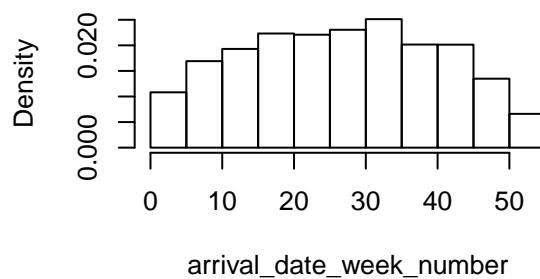
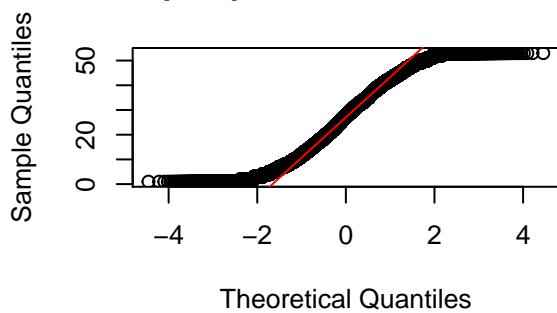
Normal Q-Q plot para is_canceled



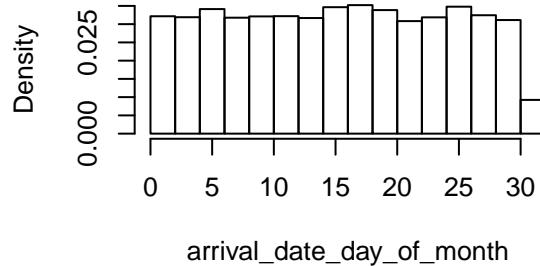
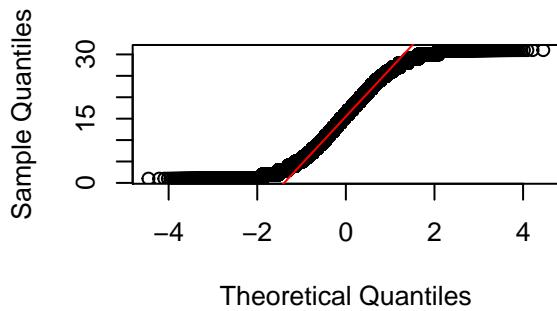
Histograma para is_canceled



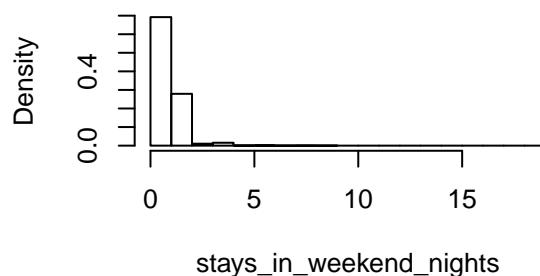
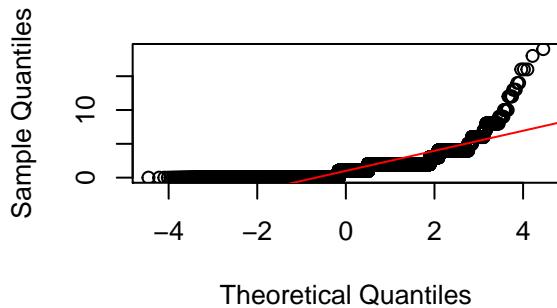
Normal Q-Q plot para arrival_date_week_n



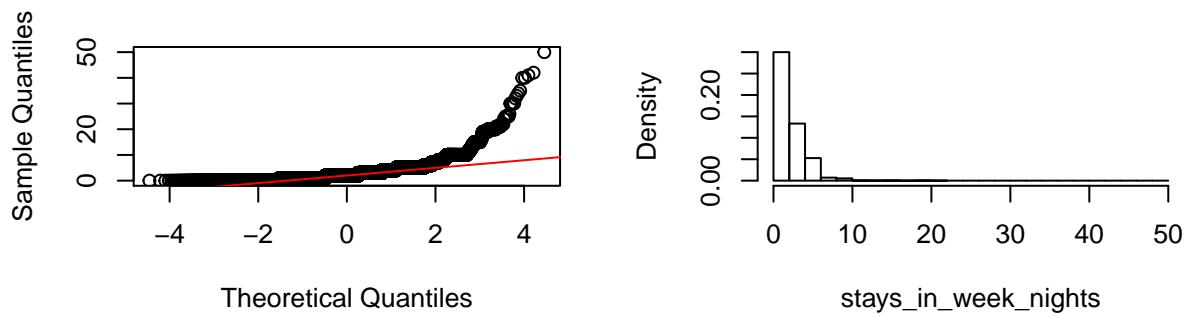
Normal Q-Q plot para arrival_date_day_of_Histograma para arrival_date_day_of_m



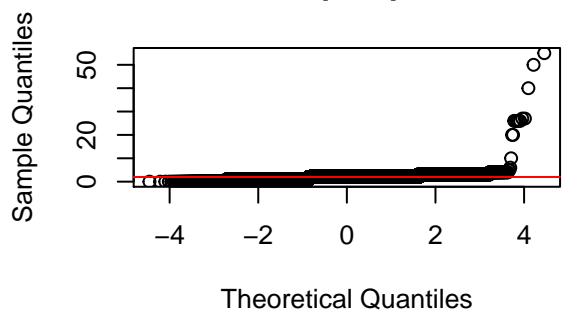
Normal Q-Q plot para stays_in_weekend_n



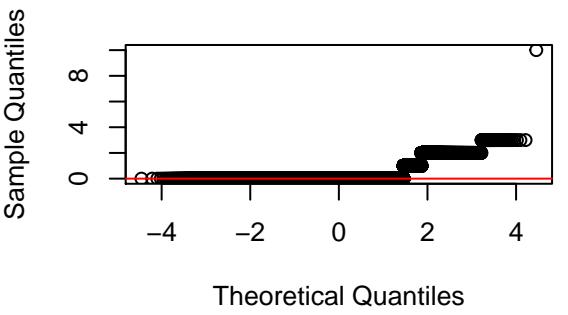
Normal Q-Q plot para stays_in_week_nights **Histograma para stays_in_week_nights**



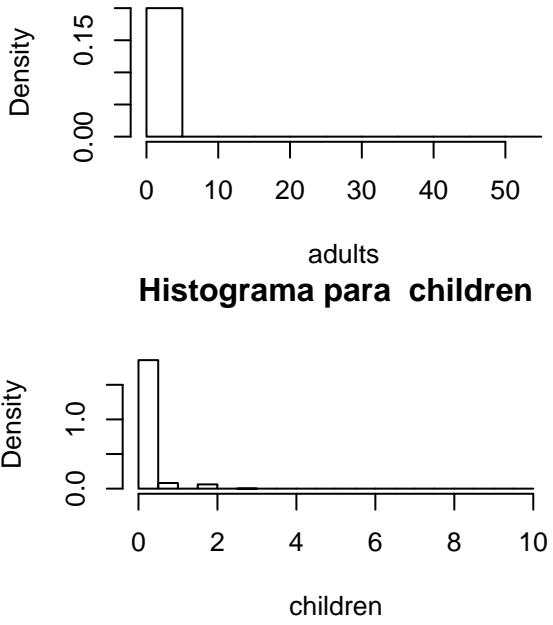
Normal Q-Q plot para adults



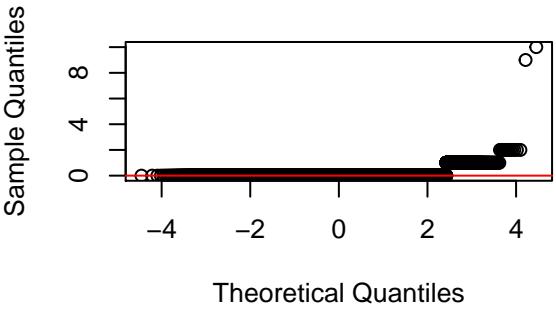
Normal Q-Q plot para children



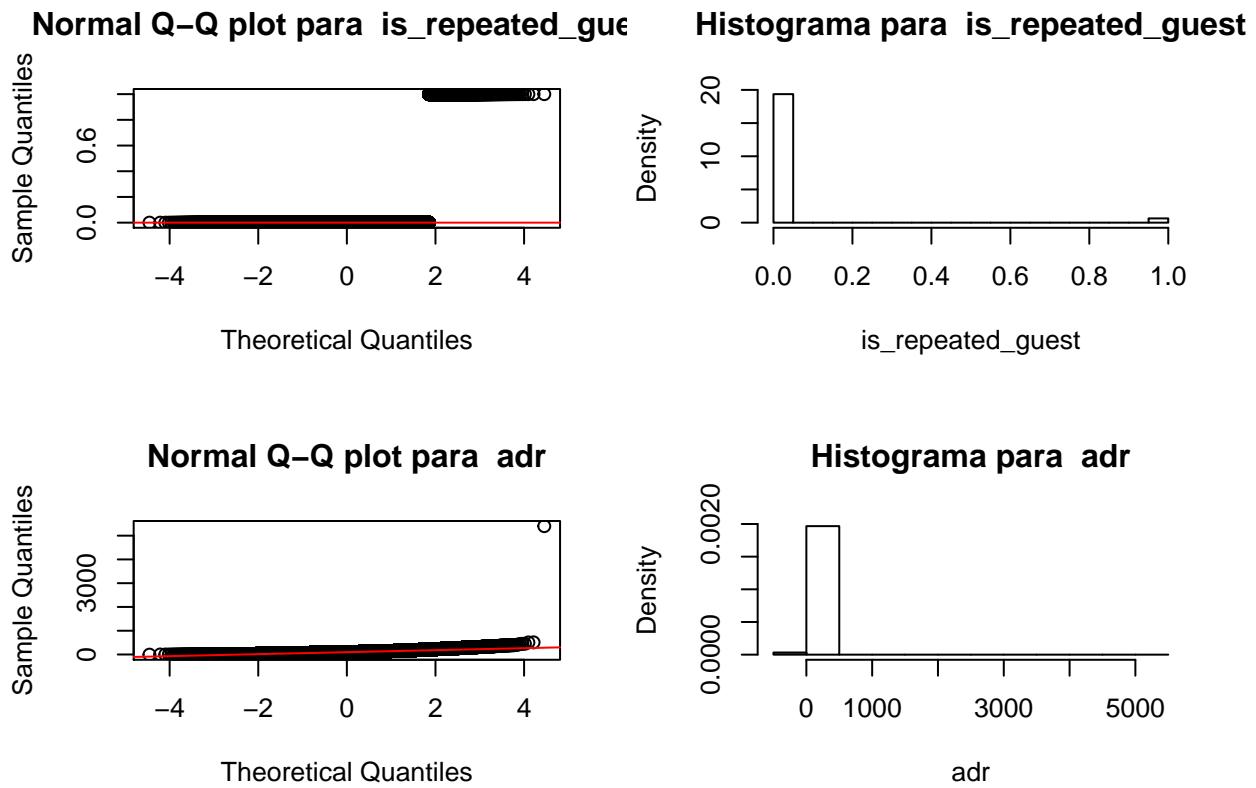
Histograma para adults



Normal Q-Q plot para babies



Histograma para babies



```
# ahora para ver si las varibales estan normalizadas aplico el test de
# Shapiro Wilk a cada variable numérica.
```

4.3. 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.