R Notebook

1. **Tema**

   Análisis del servicio de alojamiento de Airbnb en nueve ciudades en Europa del 2023.

2. **Objetivo general**

   Analizar del servicio de alojamiento de Airbnb en nueve ciudades en Europa del 2023.

3. **Objetivos**

- Analizar las ciudades con mejor ranking de experiencia de Airbnb en Europa del 2023.
- Describir las características de los alojamientos más caros y más económicos en ciudades de Europa del 2023.
- Analizar la atracción de un Airbnb y su relación con su ubicación en ciudades de Europa del 2023.

3. **Instrumento de recolección**

- Acceso a data de fuente secundaria (41 mil observaciones: Datos sobre los AIRBNB de 9 ciudades famosas de Europa, Amsterdam, Athens, Barcelona, Berlin, Budapest, Lisbon, Paris, Rome and Vienna.)

  https://www.kaggle.com/datasets/dipeshkhemani/airbnb-cleaned-europe-dataset?group=bookmarked

  https://drive.google.com/file/d/1I1u893jIu2HbdGp-ljNbc6_ANqmrJ2Xq/view?usp=share_link

- Clasificación de variables

N°	Variable	Tipo
1	City	(Categórica nominal)
2	Cuándo lo reservan	(Categórica nominal)
3	Room Type	(Categórica nominal)
4	Shared Room	(Categórica nominal)
5	Private Room	(Categórica nominal)
6	Superhost	(Categórica nominal)
7	Multiple Rooms	(Categórica nominal)
8	Cleanliness Rating	(Categórica ordinal)
9	Guest Satisfaction	(Categórica ordinal)
10	Business (ofertas)	(Categórica nominal)
11	Person Capacity	(Numérica discreta)
12	Price	(Numérica continua)
13	Bedrooms	(Numérica discreta)
14	City Center (km)	(Numérica continua)
15	Metro Distance (km)	(Numérica continua)
16	Normalised attraction Index	(Numérica continua)
17	Normalised restaurant index	(Numérica continua)

3. **Planificación : realizar un diagrama de Gantt hasta la semana 15. (FALTA)**

4. **Lectura de base de datos**

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
D <- read_csv("Aemf1.csv") #EL ARCHIVO Aemf1 y datos.AIRBNB es lo mismo.
```

```
## Rows: 41714 Columns: 19

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): City, Day, Room Type
## dbl (13): Price, Person Capacity, Multiple Rooms, Business, Cleanliness Rati...
## lgl  (3): Shared Room, Private Room, Superhost
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

- Eliminamos la columna 18 y 16, ya que de esas variables usamos sólo sus valores normalizados.

```
View(D)

D$`Attraction Index` <- NULL
D$`Restraunt Index` <- NULL
```

- Re asignación de nombres :

```
names(D)
```

```
##  [1] "City"                       "Price"
##  [3] "Day"                        "Room Type"
##  [5] "Shared Room"                "Private Room"
##  [7] "Person Capacity"            "Superhost"
##  [9] "Multiple Rooms"             "Business"
## [11] "Cleanliness Rating"         "Guest Satisfaction"
## [13] "Bedrooms"                   "City Center (km)"
## [15] "Metro Distance (km)"        "Normalised Attraction Index"
## [17] "Normalised Restraunt Index"
```

```
D %>% rename( Tipo = `Room Type` , Capacidad = `Person Capacity`, Oferta = Business, Habitaciones = Bedr
```

```
names(D2)
```

```
##  [1] "Ciudad"              "Precio"              "Day"
##  [4] "Tipo"                "Share_Room"          "Private_Room"
##  [7] "Capacidad"           "Superhost"           "Multiple Rooms"
## [10] "Oferta"              "Limpieza"            "Satisfaccion"
## [13] "Habitaciones"        "City Center (km)"    "Metro Distance (km)"
## [16] "Ind.Atraccion"       "Ind.Restaurantes"
```

- Búsqueda de NA's y comprobación de casos completos

```
Total_de_NAs <- sum(is.na(D))
print(paste("El total de NAs es :", Total_de_NAs))
```

```
## [1] "El total de NAs es : 0"
```

```
Observaciones_completas <- sum(complete.cases(D2))
print(paste("Observaciones completas es : ",Observaciones_completas ))
```

```
## [1] "Observaciones completas es :  41714"
```

```
Observaciones_incompletas <- sum(!complete.cases(D2))
print(paste("Observaciones incompletas es : ", Observaciones_incompletas))
```

```
## [1] "Observaciones incompletas es :  0"
```

Limpieza de Datos

El primer paso para la limpieza de datos es hacer un resumen de los datos que se tienen.

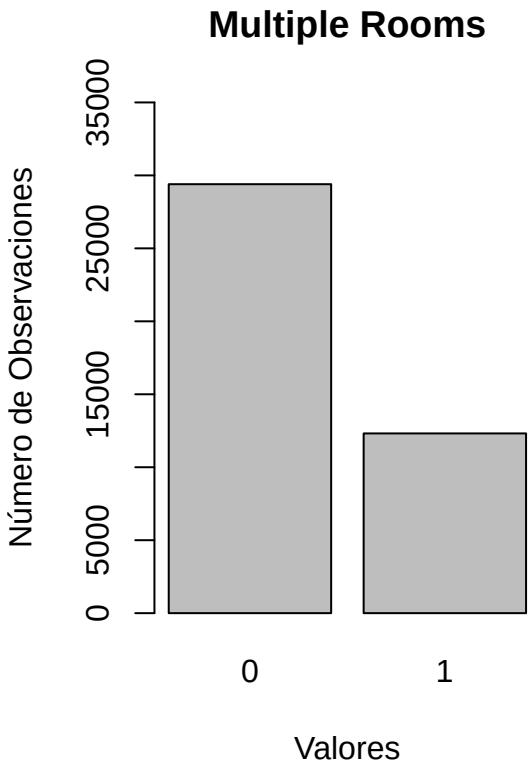
```
summary(D2)
```

```
##     Ciudad              Precio             Day                Tipo
##  Length:41714       Min.   :   34.78   Length:41714       Length:41714
##  Class :character   1st Qu.:  144.02   Class :character   Class :character
##  Mode  :character   Median :  203.82   Mode  :character   Mode  :character
##                     Mean   :  260.09
##                     3rd Qu.:  297.37
##                     Max.   :18545.45
##  Share_Room      Private_Room      Capacidad     Superhost
##  Mode :logical   Mode :logical   Min.   :2.000   Mode :logical
##  FALSE:41398     FALSE:28580     1st Qu.:2.000   FALSE:30055
##  TRUE :316       TRUE :13134     Median :3.000   TRUE :11659
##                                  Mean   :3.237
##                                  3rd Qu.:4.000
##                                  Max.   :6.000
##  Multiple Rooms       Oferta          Limpieza       Satisfaccion
```

```
##  Min.   :0.0000   Min.   :0.0000   Min.   : 2.000   Min.   : 20.0
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.: 9.000   1st Qu.: 90.0
##  Median :0.0000   Median :0.0000   Median :10.000   Median : 95.0
##  Mean   :0.2953   Mean   :0.3412   Mean   : 9.442   Mean   : 93.1
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:10.000   3rd Qu.: 98.0
##  Max.   :1.0000   Max.   :1.0000   Max.   :10.000   Max.   :100.0
##   Habitaciones    City Center (km)   Metro Distance (km) Ind.Atraccion
##  Min.   : 0.000   Min.   : 0.01504   Min.   : 0.002301   Min.   :  0.9263
##  1st Qu.: 1.000   1st Qu.: 1.27591   1st Qu.: 0.236693   1st Qu.:  5.5107
##  Median : 1.000   Median : 2.25324   Median : 0.391220   Median :  9.9511
##  Mean   : 1.166   Mean   : 2.67979   Mean   : 0.603921   Mean   : 11.7197
##  3rd Qu.: 1.000   3rd Qu.: 3.58449   3rd Qu.: 0.678702   3rd Qu.: 15.4670
##  Max.   :10.000   Max.   :25.28456   Max.   :14.273577   Max.   :100.0000
##  Ind.Restaurantes
##  Min.   :  0.5928
##  1st Qu.: 11.1320
##  Median : 21.8144
##  Mean   : 25.5536
##  3rd Qu.: 36.8214
##  Max.   :100.0000
```

En primer lugar, se observa que Multiple rooms tiene como valor mínimo 0 y como valor máximo 1; esto mismo ocurre en el caso de la variable Oferta. Procedemos a analizar qué valores se encuentran dentro de estas dos variables.

```
par(mfrow=c(1,2))

counts <- table(D2$`Multiple Rooms`)
barplot(counts, main = "Multiple Rooms", ylab = "Número de Observaciones", xlab = "Valores" , ylim = c(0
```




```
counts <- table(D2$Oferta)
barplot(counts, main = "Oferta", ylab = "Número de Observaciones", xlab = "Valores" , ylim = c(0,30000))
```

Como se observa, ambas variables cuentan solamente con "0" y "1" como valores, por lo que se trata de variables booleanas. Se observa que, además de estas dos variables, se tiene que Share_Room, Private_Room y Superhost también son variables de tipo bool. Para facilitar las operaciones futuras, se procederá a cambiar los "0" y "False" a la cadena "No", y los "1" y "True" a la cadena "Sí".

```
vec <- factor(D2$Oferta, labels = c('No','Sí'))
D2$Oferta <- vec
vec <- factor(D2$`Multiple Rooms`, labels = c('No','Sí'))
D2$`Multiple Rooms` <- vec
vec <- factor(D2$Share_Room, labels = c('No','Sí'))
D2$Share_Room <- vec
vec <- factor(D2$Private_Room, labels = c('No','Sí'))
D2$Private_Room <- vec
vec <- factor(D2$Superhost, labels = c('No','Sí'))
D2$Superhost <- vec
```

A continuación, se hace nuevamente un resumen para verificar los cambios realizados a las variables.

```
summary(D2)
```

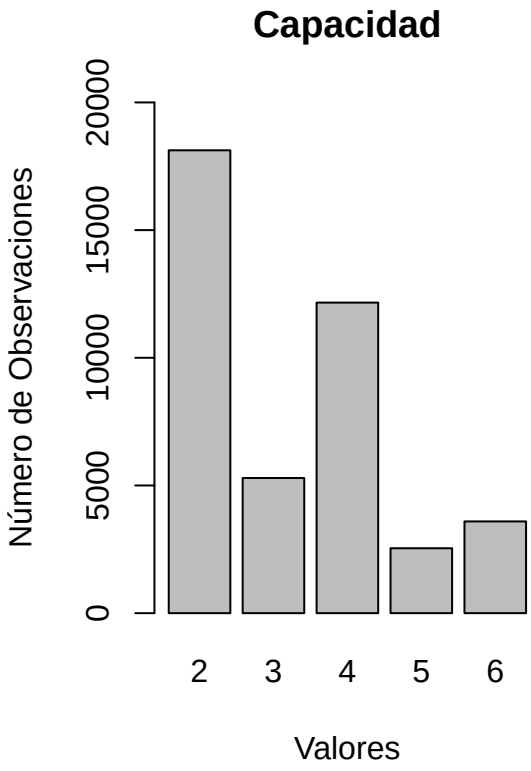
```
##     Ciudad              Precio            Day                Tipo
##  Length:41714       Min.   :   34.78   Length:41714       Length:41714
##  Class :character   1st Qu.:  144.02   Class :character   Class :character
##  Mode  :character   Median :  203.82   Mode  :character   Mode  :character
##                     Mean   :  260.09
```

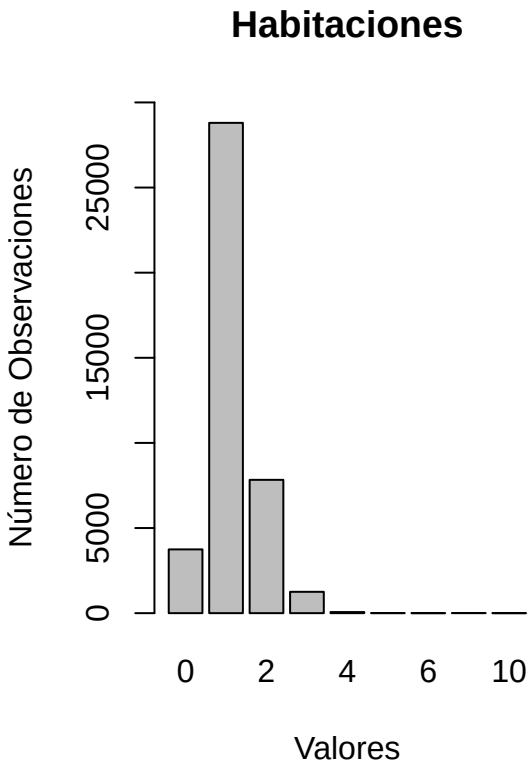
```
##                     3rd Qu.:  297.37
##                     Max.   :18545.45
##  Share_Room Private_Room   Capacidad     Superhost  Multiple Rooms Oferta
##  No:41398   No:28580     Min.   :2.000   No:30055   No:29397       No:27482
##  Sí:  316   Sí:13134     1st Qu.:2.000   Sí:11659   Sí:12317       Sí:14232
##                          Median :3.000
##                          Mean   :3.237
##                          3rd Qu.:4.000
##                          Max.   :6.000
##     Limpieza       Satisfaccion    Habitaciones    City Center (km)
##  Min.   : 2.000   Min.   : 20.0   Min.   : 0.000   Min.   : 0.01504
##  1st Qu.: 9.000   1st Qu.: 90.0   1st Qu.: 1.000   1st Qu.: 1.27591
##  Median :10.000   Median : 95.0   Median : 1.000   Median : 2.25324
##  Mean   : 9.442   Mean   : 93.1   Mean   : 1.166   Mean   : 2.67979
##  3rd Qu.:10.000   3rd Qu.: 98.0   3rd Qu.: 1.000   3rd Qu.: 3.58449
##  Max.   :10.000   Max.   :100.0   Max.   :10.000   Max.   :25.28456
##  Metro Distance (km) Ind.Atraccion      Ind.Restaurantes
##  Min.   : 0.002301   Min.   :  0.9263   Min.   :  0.5928
##  1st Qu.: 0.236693   1st Qu.:  5.5107   1st Qu.: 11.1320
##  Median : 0.391220   Median :  9.9511   Median : 21.8144
##  Mean   : 0.603921   Mean   : 11.7197   Mean   : 25.5536
##  3rd Qu.: 0.678702   3rd Qu.: 15.4670   3rd Qu.: 36.8214
##  Max.   :14.273577   Max.   :100.0000   Max.   :100.0000
```

Verificar valores enteros : De este resumen, también se puede observar que el Indice de Atracción normalizado y el Índice de Restaurantes normalizado se encuentran dentro del rango establecido (0-100).

A continuación se verificará que la variable capacidad contenga solamente números enteros, ya que un airbnb no puede tener capacidad para un número no entero de personas.El mismo análisis se realizará para la variable Habitaciones, ya que sucede lo mismo con esta variable.

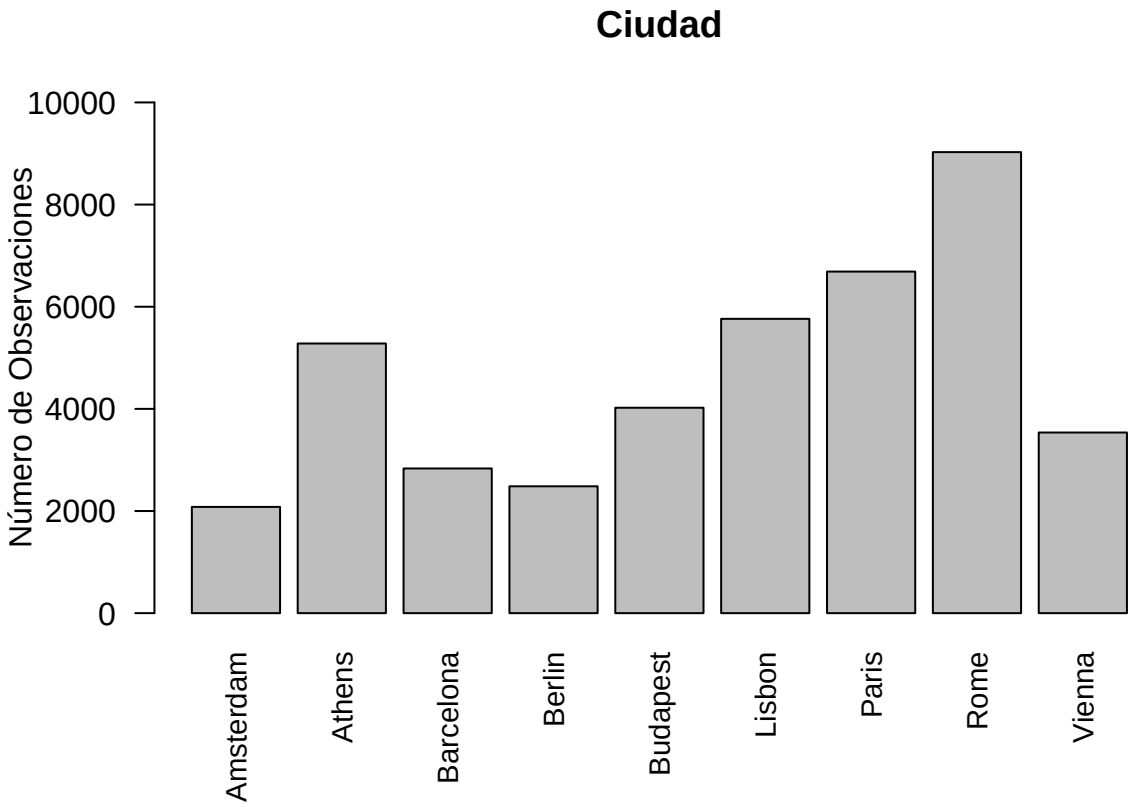
```
par(mfrow=c(1,2))
counts <- table(D2$Capacidad)
barplot(counts, main = "Capacidad", ylab = "Número de Observaciones", xlab = "Valores" , ylim = c(0,200


counts <- table(D2$Habitaciones)
barplot(counts, main = "Habitaciones", ylab = "Número de Observaciones", xlab = "Valores" , ylim = c(0,
```





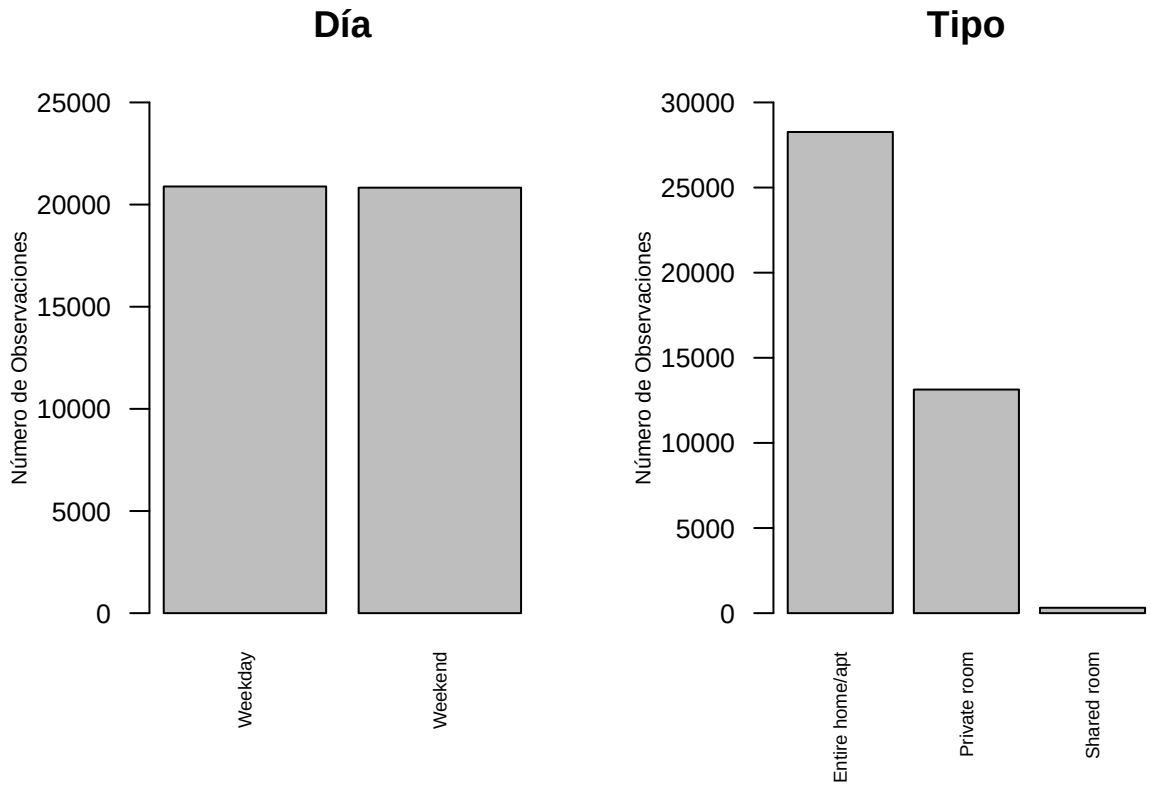
Verificando variable Ciudad, Día y Tipo Se observa que ambas variables cumplen con los criterios previamente mencionados. Por último se procederá a analizar la variable Ciudad, Día y Tipo; estas son catalogadas por R como variables de tipo char. Para el caso, de la variable Ciudad, deberían haber 9 valores distintos, que representen las 9 ciudades a las que pertenece la data.

```
counts <- table(D2$Ciudad)
barplot(counts, main = "Ciudad", ylab = "Número de Observaciones" , las=2, ylim = c(0,10000), cex.names
```



Una vez comprobado que solamente existen 9 ciudades en el Dataframe, se procede a revisar el Día y Tipo de airbnb.

```
par(mfrow=c(1,2)) #QUE HACE mfrow ?
counts <- table(D2$Day)
barplot(counts, main = "Día", ylab = "Número de Observaciones", las = 2, ylim = c(0,25000), cex.axis = 
counts <- table(D2$Tipo)
barplot(counts, main = "Tipo", ylab = "Número de Observaciones", las = 2, ylim = c(0,30000), cex.axis =
```



Se observa que tampoco existen anomalías en estas variables, por lo que podemos concluir que nuestra data se encuentra en condiciones de ser utilizada para los análisis posteriores.

«««< HEAD

```
library(funModeling)
```

```
## Loading required package: Hmisc

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':
##
##     src, summarize

## The following objects are masked from 'package:base':
##
##     format.pval, units

## funModeling v.1.9.4 :)
## Examples and tutorials at livebook.datascienceheroes.com
##  / Now in Spanish: librovivodecienciadedatos.ai
```

```
status(D2)
```

```
##                                variable q_zeros    p_zeros q_na p_na q_inf
## Ciudad                           Ciudad       0 0.00000000    0    0     0
## Precio                           Precio       0 0.00000000    0    0     0
## Day                                 Day       0 0.00000000    0    0     0
## Tipo                               Tipo       0 0.00000000    0    0     0
## Share_Room                   Share_Room       0 0.00000000    0    0     0
## Private_Room               Private_Room       0 0.00000000    0    0     0
## Capacidad                     Capacidad       0 0.00000000    0    0     0
## Superhost                     Superhost       0 0.00000000    0    0     0
## Multiple Rooms           Multiple Rooms       0 0.00000000    0    0     0
## Oferta                           Oferta       0 0.00000000    0    0     0
## Limpieza                       Limpieza       0 0.00000000    0    0     0
## Satisfaccion                Satisfaccion       0 0.00000000    0    0     0
## Habitaciones                Habitaciones    3745 0.08977801    0    0     0
## City Center (km)       City Center (km)       0 0.00000000    0    0     0
## Metro Distance (km) Metro Distance (km)       0 0.00000000    0    0     0
## Ind.Atraccion             Ind.Atraccion       0 0.00000000    0    0     0
## Ind.Restaurantes       Ind.Restaurantes       0 0.00000000    0    0     0
##                      p_inf      type unique
## Ciudad                   0 character      9
## Precio                   0   numeric   8087
## Day                      0 character      2
## Tipo                     0 character      3
## Share_Room               0    factor      2
## Private_Room             0    factor      2
## Capacidad                0   numeric      5
## Superhost                0    factor      2
## Multiple Rooms           0    factor      2
## Oferta                   0    factor      2
## Limpieza                 0   numeric      9
## Satisfaccion             0   numeric     51
## Habitaciones             0   numeric      9
## City Center (km)         0   numeric  41714
## Metro Distance (km)      0   numeric  41714
## Ind.Atraccion            0   numeric  41697
## Ind.Restaurantes         0   numeric  41697
```

======= »»»> 34956cedce1c136ea59c35bed128582071d0140d