



# **Universidad Peruana de Ciencias Aplicadas**

**FACULTAD DE INGENIERÍA**

**“Año del Fortalecimiento de la Soberanía Nacional”**

## **CC50-Administración de la Información**

**Profesora:** Reyes Silva, Patricia Daniela

### **Integrantes:**

Anto Chávez, Carolain Marisol - u201319550

López Takahashi, Rodrigo Andrés - u201615003

**2022 – 1**

# Índice

<b>CASO DE ANÁLISIS</b>	<b>2</b>
<b>CONJUNTO DE DATOS (DATA SET)</b>	<b>2</b>
<b>ANÁLISIS EXPLORATORIO DE DATOS</b>	<b>5</b>
CARGAR DATOS	5
INSPECCIONAR DATOS	6
PRE-PROCESAR DATOS	6
Completar los valores NA	7
Completar los outliers	10
VISUALIZAR DATOS	11
¿Cuántas reservas se realizan por tipo de hotel? o ¿Qué tipo de hotel prefiere la gente?	11
¿Está aumentando la demanda con el tiempo?	13
¿Cuándo se producen las temporadas de reservas: alta, media y baja?	14
¿Cuándo es menor la demanda de reservas?	14
¿Cuántas reservas incluyen niños y/o bebés?	15
¿Es importante contar con espacios de estacionamiento?	16
¿En qué meses del año se producen más cancelaciones de reservas?	17
<b>CONCLUSIONES PRELIMINARES</b>	<b>18</b>
<b>BIBLIOGRAFÍA</b>	<b>19</b>

## 1. CASO DE ANÁLISIS

El dataset escogido es llamado “Hotel booking demand” el cual es un conjunto de datos modificado. Su versión original se puede obtener de la página web sciencedirect (<https://www.sciencedirect.com/science/article/pii/S2352340918315191>). Este fue publicado en Febrero del 2019 por Nuno Antonio, Ana de Almeida y Luis Nunes de diferentes centros educativos de Portugal.

El análisis de este dataset sería importante para los hoteles, porque les permitiría predecir el comportamiento de sus clientes, y así se poder mejorar la atención a estos. También el análisis del dataset sería importante para estudiantes que estén o deseen aprender ciencia de datos.

## 2. CONJUNTO DE DATOS (DATA SET)

Descripción de la estructura de los datos

Variable	Tipo	Descripción
ADR	Numérico	Tarifa diaria promedio
Adults	Entero	Número de adultos
Agent	Categoría	ID de la agencia de viajes que realizó la reserva
ArrivalDateDayOfMonth	Entero	Día del mes de la fecha de llegada
ArrivalDateMonth	Categoría	Mes de fecha de llegada con 12 categorías (de enero a diciembre)
ArrivalDateWeekNumber	Entero	Número de semana de la fecha de llegada
ArrivalDateYear	Entero	Año de la fecha de llegada
AssignedRoomType	Categoría	Código del tipo de habitación asignado a la reserva. En ocasiones, el tipo de habitación asignado difiere del tipo de habitación reservado debido a razones de funcionamiento del hotel (por ejemplo, overbooking) o por solicitud del cliente. Se presenta el código en lugar de la designación por razones de anonimato
Babies	Entero	Número de bebés
BookingChanges	Entero	Número de cambios/modificaciones realizados en la reserva desde el momento en que se ingresa la reserva en el PMS hasta el momento del check-in o la cancelación
Children	Entero	Numero de niños

Company	Categoría	ID de la empresa/entidad que realizó la reserva o responsable del pago de la reserva. Se presenta ID en lugar de designación por razones de anonimato
Country	Categoría	País de origen. Las categorías se representan en el formato ISO 3155–3:2013
CustomerType	Categoría	<p>Tipo de reserva, asumiendo una de cuatro categorías:</p> <p>Contract - cuando la reserva tiene asociada una asignación u otro tipo de contrato;</p> <p>Group – cuando la reserva está asociada a un grupo;</p> <p>Transient – cuando la reserva no es parte de un grupo o contrato, y no está asociada a otra reserva transitoria;</p> <p>Transient-party: cuando la reserva es transitoria, pero está asociada al menos a otra reserva transitoria</p>
DaysInWaitingList	Entero	Número de días que la reserva estuvo en lista de espera antes de ser confirmada al cliente
DepositType	Categoría	<p>Indicación de si el cliente realizó un depósito para garantizar la reserva. Esta variable puede asumir tres categorías:</p> <p>No Deposit: no se realizó ningún depósito;</p> <p>Non Refund: se realizó un depósito por el valor del costo total de la estadía;</p> <p>Refundable: se realizó un depósito con un valor inferior al costo total de la estadía.</p>
DistributionChannel	Categoría	Canal de distribución de reservas. El término "TA" significa "Agentes de viajes" y "TO" significa "Operadores turísticos"
IsCanceled	Categoría	Valor que indica si la reserva fue cancelada (1) o no (0)
IsRepeatedGuest	Categoría	Valor que indica si el nombre de la reserva era de un huésped repetido (1) o no (0)
LeadTime	Entero	Número de días transcurridos entre la fecha de entrada de la reserva en el PMS y la fecha de llegada

MarketSegment	Categoría	Designación del segmento de mercado. En las categorías, el término "TA" significa "Agentes de viajes" y "TO" significa "Operadores turísticos"
Meal	Categoría	Tipo de comida reservada. Las categorías se presentan en paquetes estándar de comidas de hospitalidad:  Indefinido / SC - sin paquete de comida;  BB – Alojamiento y Desayuno;  HB – Media pensión (desayuno y otra comida, generalmente cena);  FB – Pensión completa (desayuno, comida y cena)
PreviousBookingsNotCanceled	Entero	Número de reservas anteriores no canceladas por el cliente antes de la reserva actual
PreviousCancellations	Entero	Número de reservas anteriores que fueron canceladas por el cliente antes de la reserva actual
RequiredCardParkingSpaces	Entero	Número de plazas de aparcamiento requeridas por el cliente
ReservationStatus	Categoría	Último estado de la reserva, asumiendo una de las tres categorías:  Canceled – La reserva fue cancelada por el cliente;  Check-Out – El cliente se ha registrado pero ya se ha ido;  No-Show – el cliente no se registró e informó al hotel del motivo
ReservationStatusDate	Date	Fecha en la que se estableció el último estado. Esta variable se puede usar junto con ReservationStatus para comprender cuándo se canceló la reserva o cuándo se retiró el cliente del hotel.
ReservedRoomType	Categoría	Código del tipo de habitación reservado. Se presenta el código en lugar de la designación por razones de anonimato
StaysInWeekendNights	Entero	Número de noches de fin de semana (sábado o domingo) que el huésped se alojó o reservó para quedarse en el hotel

StaysInWeekNights	Entero	Número de noches de la semana (de lunes a viernes) que el huésped se hospedó o reservó para quedarse en el hotel
TotalOfSpecialRequests	Entero	Número de solicitudes especiales realizadas por el cliente (por ejemplo, cama doble o piso alto)

Nota: Traducido y adaptado de Hotel booking demand datasets

### 3. ANÁLISIS EXPLORATORIO DE DATOS

El presente análisis fue realizado con el programa R studio (versión 3.6).

#### 3.1. CARGAR DATOS

Realizamos la carga de datos correspondiente:

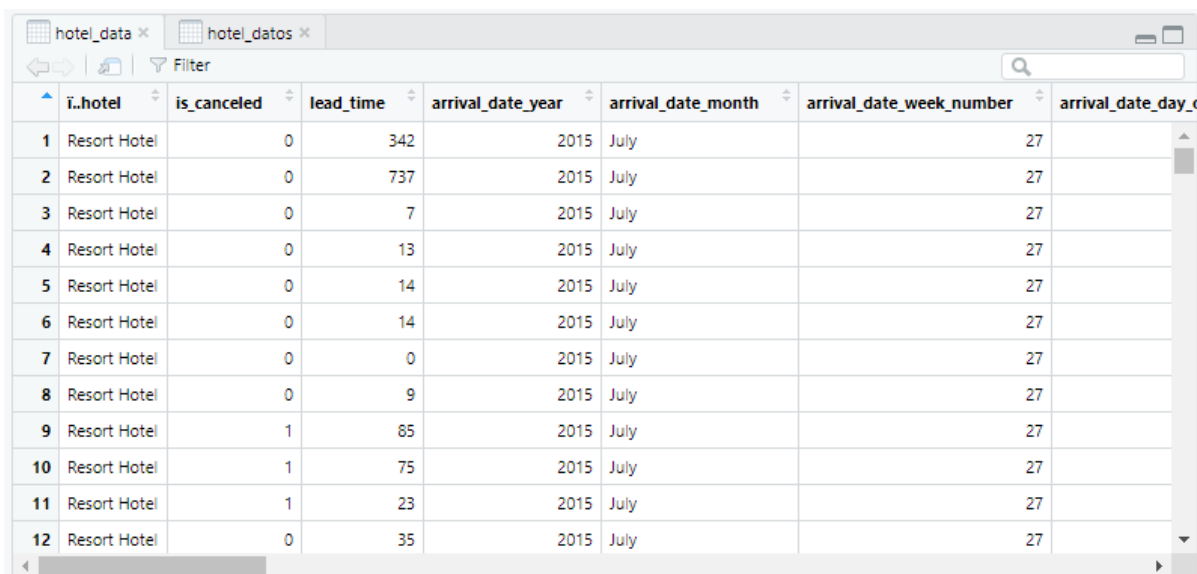
```
#CARGAR DATOS
hotel_data <- read.csv ("data/hotel_bookings_miss.csv", header = TRUE, stringsAsFactors = FALSE, sep = ",")
```

Y omitimos todos aquellos datos duplicados, ya que incluso si se determina su tipo de datos y otros, siguen siendo duplicados:

```
# Duplicados
#solo usaremos columnas no duplicadas
hotel_datos<-unique(hotel_data)
```

Observamos el resumen de sus variables y la tabla cargada:

▶ hotel_data	119390 obs. of 32 variables
▶ hotel_datos	87443 obs. of 33 variables



	i.hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month
1	Resort Hotel	0	342	2015	July	27	
2	Resort Hotel	0	737	2015	July	27	
3	Resort Hotel	0	7	2015	July	27	
4	Resort Hotel	0	13	2015	July	27	
5	Resort Hotel	0	14	2015	July	27	
6	Resort Hotel	0	14	2015	July	27	
7	Resort Hotel	0	0	2015	July	27	
8	Resort Hotel	0	9	2015	July	27	
9	Resort Hotel	1	85	2015	July	27	
10	Resort Hotel	1	75	2015	July	27	
11	Resort Hotel	1	23	2015	July	27	
12	Resort Hotel	0	35	2015	July	27	

	i..hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month
1	Resort Hotel	0	342	2015	July	27	
2	Resort Hotel	0	737	2015	July	27	
3	Resort Hotel	0	7	2015	July	27	
4	Resort Hotel	0	13	2015	July	27	
5	Resort Hotel	0	14	2015	July	27	
7	Resort Hotel	0	0	2015	July	27	
8	Resort Hotel	0	9	2015	July	27	
9	Resort Hotel	1	85	2015	July	27	
10	Resort Hotel	1	75	2015	July	27	
11	Resort Hotel	1	23	2015	July	27	
12	Resort Hotel	0	35	2015	July	27	
13	Resort Hotel	0	68	2015	July	27	

### 3.2. INSPECCIONAR DATOS

Posteriormente, convertimos todos los valores al tipo de dato más conveniente:

```

1..hotel      is_canceled  lead_time      arrival_date_year arrival_date_month arrival_date_week_number
City Hotel :53428      0:63379      Min. : 0.00      Min. :2015      August :11258      Min. : 1.00
Resort Hotel:34015    1:24064      1st Qu.: 11.00    1st Qu.:2016      July :10058        1st Qu.:16.00
                                          Median : 49.00    Median :2016      May : 8359         Median :27.00
                                          Mean : 79.92     Mean :2016       April : 7913        Mean :26.84
                                          3rd Qu.:125.00   3rd Qu.:2017     June : 7767         3rd Qu.:37.00
                                          Max. :737.00     Max. :2017       March : 7520        Max. :53.00
                                          NA's :21         NA's :6          (Other):34568      NA's :25

arrival_date_day_of_month stays_in_weekend_nights stays_in_week_nights adults children
Min. : 1.00              Min. : 0.000      Min. : 0.000      Min. : 0.000      Min. : 0.0000
1st Qu.: 8.00            1st Qu.: 0.000    1st Qu.: 1.000    1st Qu.: 2.000    1st Qu.: 0.0000
Median :16.00            Median : 1.000    Median : 2.000    Median : 2.000    Median : 0.0000
Mean :15.82              Mean : 1.005     Mean : 2.626     Mean : 1.876     Mean : 0.1386
3rd Qu.:23.00            3rd Qu.: 2.000    3rd Qu.: 4.000    3rd Qu.: 2.000    3rd Qu.: 0.0000
Max. :31.00              Max. :19.000     Max. :50.000     Max. :55.000     Max. :10.0000
NA's :7                  NA's :25         NA's :12         NA's :12         NA's :4

babies      meal      country      market_segment distribution_channel is_repeated_guest
Min. : 0.00000 BB      :68011      PRT :27484      Online TA :51631      Corporate: 5086      0:84028
1st Qu.: 0.00000 FB      : 365       GBR :10436      Offline TA/TO:13894 Direct :12994      1: 3415
Median : 0.00000 HB      : 9092      FRA : 8839      Direct :11804      GDS : 181
Mean : 0.01082 SC      : 9481      ESP : 7258      Groups : 4966      TA/TO :69177
3rd Qu.: 0.00000 Undefined: 494 DEU : 5388      Corporate : 4217      Undefined: 5
Max. :10.00000      ITA : 3066      Complementary: 702
NA's :31          (Other):24972 (Other) : 229

previous_cancellations previous_bookings_not_canceled reserved_room_type assigned_room_type booking_changes
Min. : 0.0000      Min. : 0.0000      A :56593      A :46349      Min. : 0.0000
1st Qu.: 0.0000    1st Qu.: 0.0000    D :17403      D :22439      1st Qu.: 0.0000
Median : 0.0000    Median : 0.0000    E : 6050      E : 7196      Median : 0.0000
Mean : 0.0304      Mean : 0.1839      F : 2823      F : 3627      Mean : 0.2716
3rd Qu.: 0.0000    3rd Qu.: 0.0000    G : 2052      G : 2498      3rd Qu.: 0.0000
Max. :26.0000      Max. :72.0000      B : 999       C : 2168      Max. :21.0000
                                          (Other): 1523    (Other): 3166

deposit_type      agent      company      days_in_waiting_list customer_type adr
No Deposit:86282  9 :28759    NULL :82179    Min. : 0.0000      Contract : 3139      Min. : -6.38
Non Refund: 1054 240 :13041   40 : 851      1st Qu.: 0.0000    Group : 544         1st Qu.: 72.00
Refundable: 107 NULL :12205  223 : 503      Median : 0.0000    Transient :72017     Median : 98.10
14 : 3349 45 : 238      Mean : 0.7506      Transient-Party:11743 Mean : 106.32
7 : 3300 153 : 206      3rd Qu.: 0.0000    Max. :391.0000      3rd Qu.: 134.00
250 : 2779 154 : 133      Max. :391.0000
(Other):24010 (Other): 3333 NA's :7

required_car_parking_spaces total_of_special_requests reservation_status reservation_status_date
Min. :0.00000      Min. :0.0000      Canceled :23049      Min. :2020-01-01
1st Qu.:0.00000    1st Qu.:0.0000    Check-out:63379      1st Qu.:2020-04-01
Median :0.00000    Median :0.0000    No-Show : 1015      Median :2020-06-24
Mean :0.08418      Mean :0.6984      Mean :2020-06-23
3rd Qu.:0.00000    3rd Qu.:1.0000    3rd Qu.:2020-09-10
Max. :8.00000      Max. :5.0000      Max. :2020-12-31

```

La carga e inspección se encuentran en el archivo `_datos_carga_preproc.R`.

### 3.3. PRE-PROCESAR DATOS

## Completar los valores NA

Realizado en el archivo `_preprocs_NAs.R`.

Ingresamos datos aleatorios a los valores NA de las siguientes variables:

```
# COLUMNAS: lead_time, arrival_date_year, arrival_date_day_of_month y days_in_waiting_list :
#Ingresar datos aleatorios
rand.valor <- function(x){
  faltantes <- is.na(x)
  tot.faltantes <- sum(faltantes)
  x.obs <- x[!faltantes]
  valorado <- x
  valorado[faltantes] <- sample(x.obs, tot.faltantes, replace = TRUE)
  return (valorado)
}
random.df <- function(df, cols){
  nombres <- names(df)
  for (col in cols) {
    df[nombres[col]] <- rand.valor(df[,col])
  }
  df
}
# usaremos datos aleatorios para estas columnas
hotel_datos<-random.df(hotel_datos, c(3, 4, 7, 26))
```

Dado que han sido actualizados los datos de arribo para fechas, agregamos la variable **arrival\_date**, que nos será útil posteriormente por utilizar un tipo de dato fecha.

```
# ARRIVAL_DATE
# volvemos a completar los datos de esta columna calculada
hotel_datos$arrival_date <- paste(hotel_datos$arrival_date_year,
                                match(substr(hotel_datos$arrival_date_month, 1, 3), month.abb),
                                hotel_datos$arrival_date_day_of_month, sep="-")
hotel_datos$arrival_date <- as.Date(hotel_datos$arrival_date)
summary(hotel_datos$arrival_date)
```

Utilizamos los datos del tipo de reserved\_room\_type para definir los datos faltantes de la variable children:

```
# CHILDREN:
# Dado que las familias con niños requieren un poco más de espacio en el cuarto,
# usaremos la variable tipo de cuarto para comparar
boxplot(hotel_datos$children ~ hotel_datos$reserved_room_type, ylim = c(0,3),
        main = "Tipos de cuartos reservados vs Cantidad de niños",
        xlab = "Tipos de cuarto", ylab="Cantidad de niños")
# en las categorías de cuartos a, d, e, l y p no suelen tener niños
# en la categoría H suelen haber 1 niño, y en la categoría F,
# aunque no la mayoría, la distribución se concentra en 1 niño
# en las categorías C y G, aunque no todos, la mediana se halla en 2 niños
# en la categoría B, la mediana se encuentra en 0 niños
hotel_datos[is.na(hotel_datos$children),][c('reserved_room_type','children')]
empty_children_rows <- rownames(hotel_datos[is.na(hotel_datos$children),])
# son 4 filas: 40601 40668 40680 41161
# se observa que todos tienen reserved_room_type B, en los cuales la cantidad de niños
# varía entre 0, 1 y 2
hotel_datos[is.na(hotel_datos$children),'children'] <- sample(c(0,1), replace=TRUE, size=4)
hotel_datos[empty_children_rows,c('reserved_room_type','children')]
# Respecto a la cantidad de bebés, no se aprecia una categoría
# que muestre cierta relación con la cantidad de bebés
```



En el caso de la variable babies, dado que no tiene una correlación mínima con alguna otra variable, y considerando que la moda es 0, se usará este dato para los valores restantes:

```
# BABIES:
barplot(table(hotel_datos$babies))
empty_babies_rows <- rownames(hotel_datos[is.na(hotel_datos$babies),])
# Sin embargo, se aprecia que una mayoría considerable de personas
# no se hospeda con bebés, por lo cual se completarán los datos con 0
hotel_datos[is.na(hotel_datos$babies), 'babies'] <- 0
hotel_datos[empty_babies_rows, c('reserved_room_type', 'babies')]
```

En el caso de la variable “adults”, modificamos los datos de los adultos que sean igual a 0 y no hayan cancelado. Le insertamos datos aleatorios a los valores NA:

```
# ADULTS:
#verificamos datos atipicos de los adultos
boxplot(x = hotel_datos$adults)
table(hotel_datos$adults)
#Datos con 0 adultos volverlos NA
hotel_datos$adults[hotel_datos$adults == 0 & hotel_datos$is_canceled == 0] <- NA
#Reemplazamos datos NA por valores
hotel_datos[is.na(hotel_datos$adults),][, c('reserved_room_type', 'adults')]
empty_adults_rows <- rownames(hotel_datos[is.na(hotel_datos$adults),])
hotel_datos[is.na(hotel_datos$adults), 'adults'] <- sample(c(1,2), replace=TRUE, size=300)
hotel_datos[empty_adults_rows, c('reserved_room_type', 'adults')]
```

Utilizando la variable generada arrival\_date, completamos el número de semana utilizando su atributo yday (que nos indica el número de día en el año) y su división entera entre 7 (dado que son 7 los días de la semana), adicionando 1 al resultado de esta debido a que el índice empieza en 0<sup>1</sup>:

```
# arrival_date_week_number (25)
empty_weeks_nums <- rownames(hotel_datos[is.na(hotel_datos$arrival_date_week_number), ])
hotel_datos[empty_weeks_nums, c('arrival_date', 'arrival_date_week_number')]
empty_weeks_days <- as.POSIXlt(hotel_datos[empty_weeks_nums, 'arrival_date'])
empty_weeks_weeks <- (empty_weeks_days$yday) %/% 7 + 1
hotel_datos$arrival_date_week_number[is.na(hotel_datos$arrival_date_week_number)] = empty_weeks_weeks
hotel_datos[empty_weeks_nums, c('arrival_date', 'arrival_date_week_number')]
```

Para el campo stays\_in\_weekend\_nights utilizamos el arrival\_date para identificar qué día de la semana se hospedó, luego, de ser fin de semana, permanecerá entre 0 a 2 días en la noche. Caso contrario, se calculará cuántos días laborales permaneció, dividiéndolo enteramente entre 5 (dado que son 5 días laborales) y se multiplicará por 2 (dado que son 2

---

<sup>1</sup> Adaptado de la respuesta 8 de:

<https://stackoverflow.com/questions/22439540/how-to-get-week-numbers-from-dates>

los días fin de semana):

```
# stays_in_weekend_nights (25)
empty_stays.weekends.nums <- row.names(hotel_datos[is.na(hotel_datos$stays_in_weekend_nights), ])
hotel_datos[empty_stays.weekends.nums, c('arrival_date', 'stays_in_weekend_nights', 'stays_in_week_nights')]
#verificamos que el arrival date no sea un fin de semana
empty_stays.weekends.arrday <- weekdays(hotel_datos[empty_stays.weekends.nums, 'arrival_date'])
#https://www.rdocumentation.org/packages/lubridate/versions/0.1/topics/wday
empty_stays.weekends.fill <- function () {
  new_weekends = c()
  for (i in 1:length(empty_stays.weekends.nums)) {
    ind = as.integer(empty_stays.weekends.nums[i])
    left_days = switch (
      empty_stays.weekends.arrday[i],
      'Monday' = 5,
      'Tuesday' = 4,
      'Wednesday' = 3,
      'Thursday' = 2,
      'Friday' = 1,
      0
    )
    num_w_days = 0
    #si arribo un lunes y permanecio 5 dias, no llegó a weekend
    #si arribo un martes y permanecio 4 dias, no llegó a weekend, ...
    if (hotel_datos$stays_in_week_nights[ind] == 0) {
      if (empty_stays.weekends.arrday[i] == 'Saturday') {
        num_w_days = sample(c(0,1,2), replace = TRUE, size=1)
      }
      else if (empty_stays.weekends.arrday[i] == 'Sunday') {
        num_w_days = 1
      }
      else {
        num_w_days = 0
      }
    } else if (hotel_datos$stays_in_week_nights[ind] < left_days) {
      num_w_days = 0
    } else {
      #si arribo un lunes (5) y permanecio 8 dias, llegó a 1 weekend
      #si arribo un lunes (5) y permanecio 10 dias, llegó a 2 weekend
      num_w_days = (hotel_datos$stays_in_week_nights[ind] % 5) * 2
    }
    new_weekends <- append(new_weekends, num_w_days)
  }
  return(new_weekends)
}
hotel_datos$stays_in_weekend_nights[is.na(hotel_datos$stays_in_weekend_nights)] <- empty_stays.weekends.fill()
hotel_datos[empty_stays.weekends.nums, c('arrival_date', 'stays_in_weekend_nights', 'stays_in_week_nights')]
```

Para stays\_in\_week\_nights se utilizó un proceso parecido pero inverso. Se utilizó, además, la variable stays\_in\_weekend\_nights para definir cuántas semanas había permanecido (si permaneció más de 2 días de fin de semana, se consideró cada 2 días como una nueva semana, por tanto se multiplicaría la cantidad por 5 (días laborales). Adicionalmente, incluso si fuera una o varias semanas, se sumaría una cantidad aleatoria acorde a los días restantes para el fin de semana.

```

# stays_in_week_nights (12)
empty_stays.weeks.nums <- row.names(hotel_datos[is.na(hotel_datos$stays_in_week_nights), ])
hotel_datos[empty_stays.weeks.nums,c('arrival_date', 'stays_in_weekend_nights', 'stays_in_week_nights')]
#verificamos que el arrival date no sea un fin de semana
empty_stays.weeks.weekends <- hotel_datos[empty_stays.weeks.nums, 'stays_in_weekend_nights']
empty_stays.weeks.arrday <- weekdays(hotel_datos[empty_stays.weeks.nums, 'arrival_date'])
empty_stays.weeks.fill <- function () {
  new_weeks = c()
  for (i in 1:length(empty_stays.weeks.nums)) {
    ind = as.integer(empty_stays.weeks.nums[i])
    cat(i,"a",ind,'\n')
    num_w_days = 0
    left_days = switch (
      empty_stays.weeks.arrday[i],
      'Monday' = 5,
      'Tuesday' = 4,
      'Wednesday' = 3,
      'Thursday' = 2,
      'Friday' = 1,
      1
    )
    #si arribo un lunes y weekend=0, puede ser random de 0-5
    #si arribo un martes y weekend=0, puede ser random de 0-4
    num_w_days = sample(0:left_days,replace = TRUE, size=1)
    if (empty_stays.weeks.weekends[i] > 0){
      cat('-',empty_stays.weeks.weekends[i],'\n')
      #si llegó lunes y weekend=1 o 2 => 5 + rand
      #si llegó martes y weekend=1 o 2 => 4 + rand
      #si llegó lunes y weekend=3 o 4 => 5*2 + rand
      if (empty_stays.weeks.weekends[i] %% 2) {
        num_ws = (empty_stays.weeks.weekends[i]%%2)
      } else {
        num_ws = ((empty_stays.weeks.weekends[i]+1)%2)
      }
      num_w_days = num_w_days + (num_ws*5)
    }
    new_weeks <- append(new_weeks,num_w_days)
  }
  return(new_weeks)
}
empty_stays.weeks.fill()
hotel_datos$stays_in_week_nights[is.na(hotel_datos$stays_in_week_nights)] <- empty_stays.weeks.fill()
hotel_datos[empty_stays.weeks.nums,c('arrival_date', 'stays_in_weekend_nights', 'stays_in_week_nights')]

```

## Completar los outliers

Se realizó la verificación de los outliers para las variables de interés y se guardó la tabla modificada en el archivo `_preprocs_outliers.R`. A continuación analizaremos las variables:

Los datos atípicos de la variable children's será reemplazado si hay más de 4 niños, por el valor 0:

```

# IDENTIFICACIÓN DE DATOS ATÍPICOS (outliers) Y GUARDADO DE BD.

# CHILDREN:
summary(hotel_datos$children)
boxplot(hotel_datos$children)
# observamos como dato atípico quien indica más de 4 hijos:
outline_children_rows <- as.integer(rownames(hotel_datos[hotel_datos$children > 4,]))
hotel_datos[outline_children_rows,c('reserved_room_type','children')]
# Como en la D no suele haber niños, se colocará 0
hotel_datos[hotel_datos$children > 4,'children'] <- 0
hotel_datos[outline_children_rows,c('reserved_room_type','children')]

```

Como existen 2 datos atípicos y no se aprecia una relación entre la categoría de habitación y la cantidad de bebés entonces se reemplazarán estos por 0:

```

# BABIES: solo hay 2 datos atípicos: 10 y 9
# No se aprecia una categoría que muestre cierta relación con la cantidad de bebés
outlier_babies_rows <- rownames(hotel_datos[hotel_datos$babies >= 9,])
hotel_datos$babies[hotel_datos$babies >= 9] <- 0
hotel_datos[outlier_babies_rows,c('reserved_room_type','babies')]

```

En la variable “adults” hay 14 valores atípicos. Si estos valores son mayor a 10 adultos entonces se reemplaza por la moda, en este caso 2:

```
# ADULTS: hay 14 valores atípicos: 10, 20, 26, 27, 50 , 50 ,55
outlier_adults_rows <- rownames(hotel_datos[hotel_datos$adults >= 10,])
hotel_datos$adults[hotel_datos$adults >= 10] <- 2
hotel_datos[outlier_adults_rows,c('reserved_room_type','adults')]
```

Respecto a required\_car\_parking\_spaces, observamos que tiene como valor atípico principal, 8 parqueos. Utilizaremos la columna adults para eliminar estos datos, considerando que un adulto puede tener un parqueo como máximo, todos aquellos parqueos que tengan un requerimiento superior a la cantidad de adultos, serán actualizados al máximo de adultos:

```
# required_car_parking_spaces
boxplot(x = hotel_datos$required_car_parking_spaces)
table(hotel_datos$required_car_parking_spaces)
# 0      1      2      3      8
#80130 7280    28     3     2
# se observa que como datos atípicos aquellos que son superiores a 1 (parking spaces)
# compararemos los aparcamientos requeridos con el la cantidad de adultos
boxplot(required_car_parking_spaces ~ adults, data=hotel_datos)
# observamos que es posible encontrar hasta 3 reservas
# sin embargo, se considerará como dato no válido aquellos que
# tengan una cantidad de aparcamientos superior a la cantidad de adultos
# y se colocarán con el máximo de adultos
hotel_datos$required_car_parking_spaces[
  hotel_datos$adults < hotel_datos$required_car_parking_spaces
] <- hotel_datos$adults[hotel_datos$adults < hotel_datos$required_car_parking_spaces]
boxplot(required_car_parking_spaces ~ adults, data=hotel_datos)
```

En los campos reservation\_status e is\_canceled, no observamos outliers:

```
#reservation_status e is_canceled
boxplot(reservation_status_date ~ reservation_status * is_canceled, data = hotel_datos,
  ylab = "Meses y Si fue cancelado", xlab = "Estado de reserva", drop=TRUE, horizontal = TRUE)
# No se encontraron datos atípicos. Si han sido cancelados están marcados como tal
```

Luego de procesar estos datos, los almacenamos en el archivo **hotel\_data\_proc.csv** en la carpeta data:

```
# GUARDAR DATOS
write.csv(hotel_datos,"data/hotel_data_proc.csv", row.names = TRUE)
```

### 3.4. VISUALIZAR DATOS

A partir de las preguntas planteadas se realizó el análisis de los datos:

- a. ¿Cuántas reservas se realizan por tipo de hotel? o ¿Qué tipo de hotel prefiere la gente?

Gráfico 1:

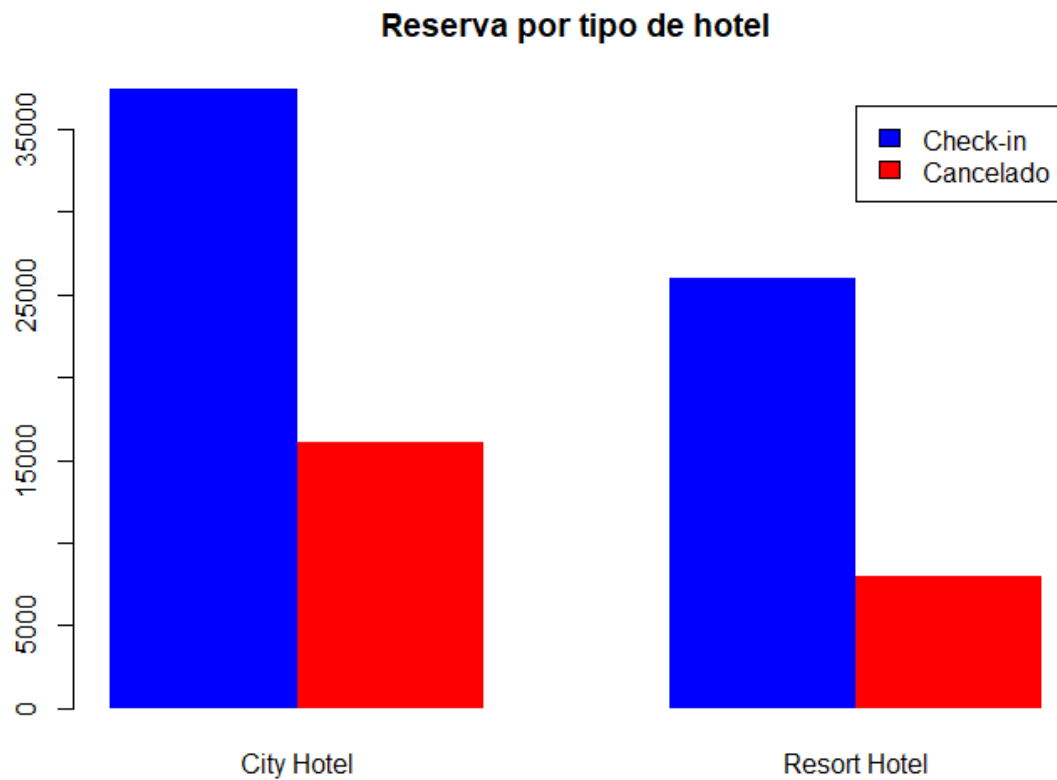


Tabla 1:

```

      0      1
City Hotel 37379 16049
Resort Hotel 26000 8015
table(hotel_datos$hotel)

City Hotel Resort Hotel
53428      34015

```

El gráfico de barras y las tablas, nos permiten entender mejor cómo está distribuido las reservas por el tipo de hotel, así como observar las reservas que fueron completadas como las que fueron canceladas por tipo de hotel.

**b. ¿Está aumentando la demanda con el tiempo?**

Gráfico 2:

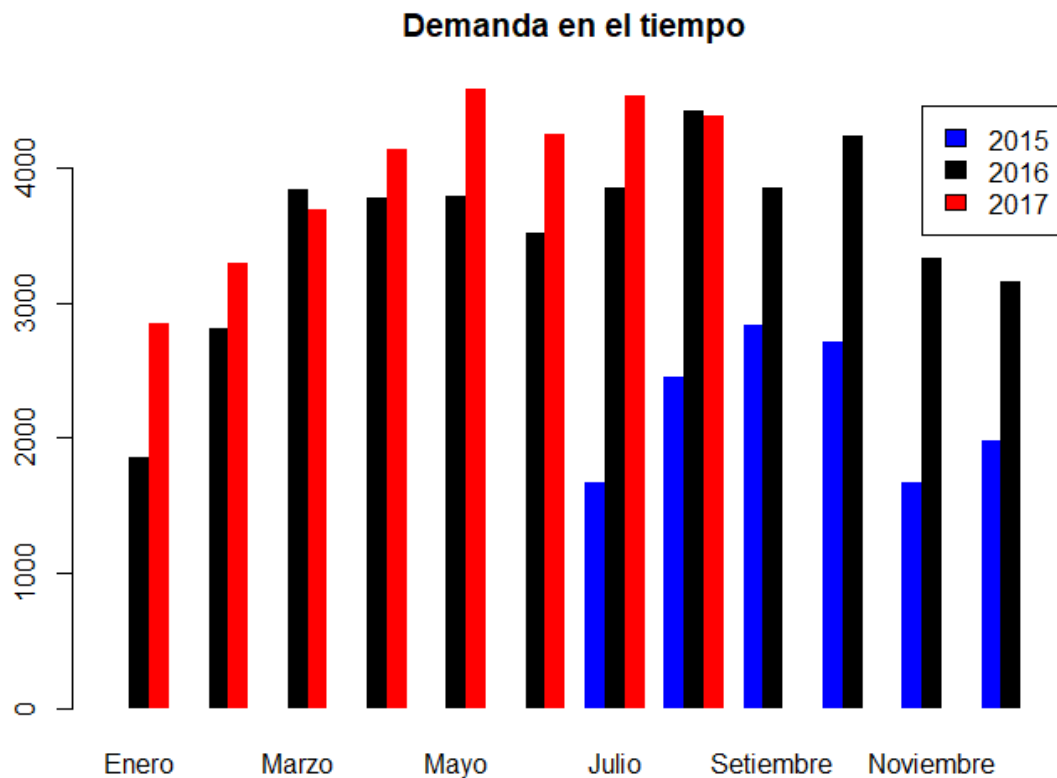


Tabla 2:

```

month      2015 2016 2017
Enero      0 1850 2845
Febrero    0 2808 3295
Marzo      0 3834 3686
Abril      1 3772 4140
Mayo       0 3782 4577
Junio      0 3518 4249
Julio     1675 3851 4532
Agosto    2453 4424 4381
Setiembre 2840 3855   0
Octubre   2705 4237   0
Noviembre 1668 3330   0
Diciembre 1985 3150   0
> table(hotel_datos$arrival

  2015  2016  2017
13327 42411 31705

```

Este gráfico de barras nos permite observar cómo están distribuidas las reservas realizadas en los diferentes meses de los años. Esto nos permitirá saber si la demanda aumenta a través de los años.

**c. ¿Cuándo se producen las temporadas de reservas: alta, media y baja?**

Gráfico 3:

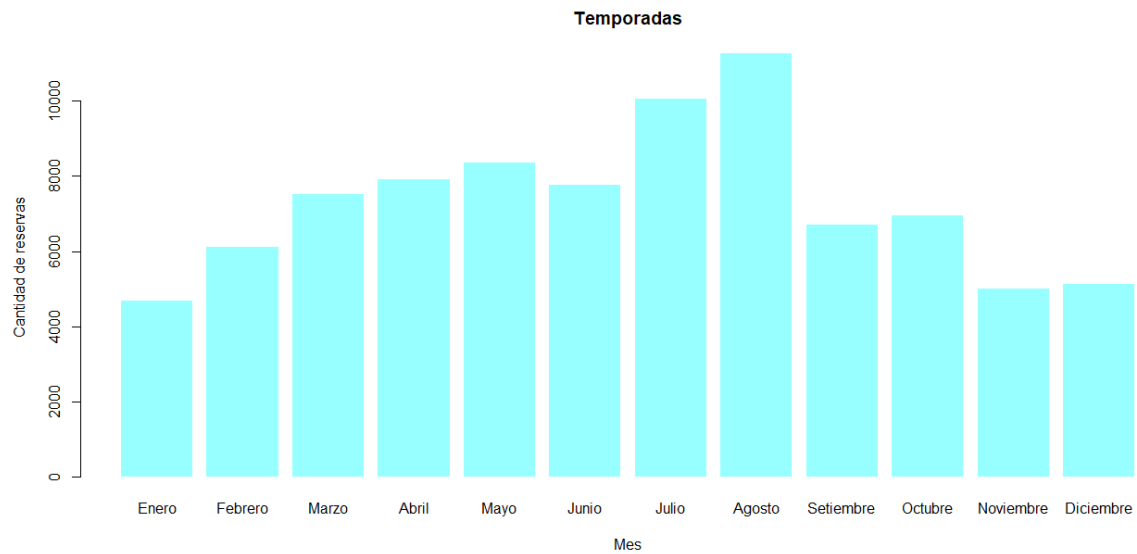


Tabla 3:

month	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Setiembre	Octubre	Noviembre	Diciembre
	4695	6103	7520	7913	8359	7767	10058	11258	6695	6942	4998	5135

Este gráfico nos permite darnos una idea de cuándo son las temporadas altas, medias y bajas si es que agrupamos por 4 meses seguidos. En este caso sumaremos Setiembre, Octubre, Noviembre y Diciembre obteniendo 23770 reservas. Con Enero, Febrero, Marzo y Abril obtenemos 26231 reservas y con los últimos 4 meses restantes 37442.

**d. ¿Cuándo es menor la demanda de reservas?**

Gráfico 4:

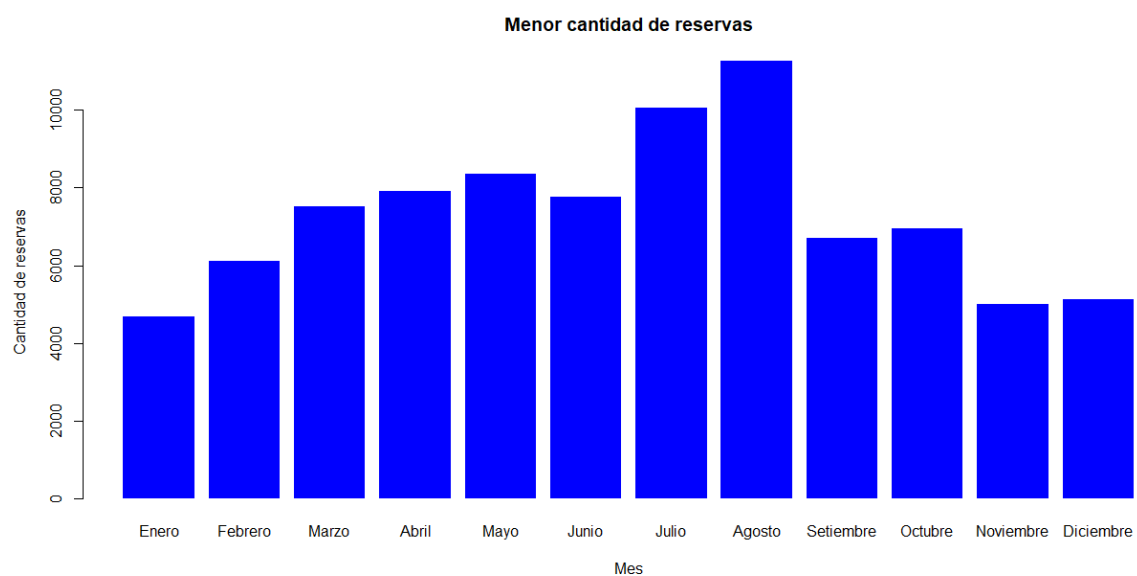


Tabla 4 :

month	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Setiembre	Octubre	Noviembre
	4695	6103	7520	7913	8359	7767	10058	11258	6695	6942	4998
Diciembre	5135										

Este gráfico nos permite saber cual es el mes que tiene menos demanda de reservas, en este caso es el mes de Enero.

#### e. ¿Cuántas reservas incluyen niños y/o bebés?

Gráfico 5:

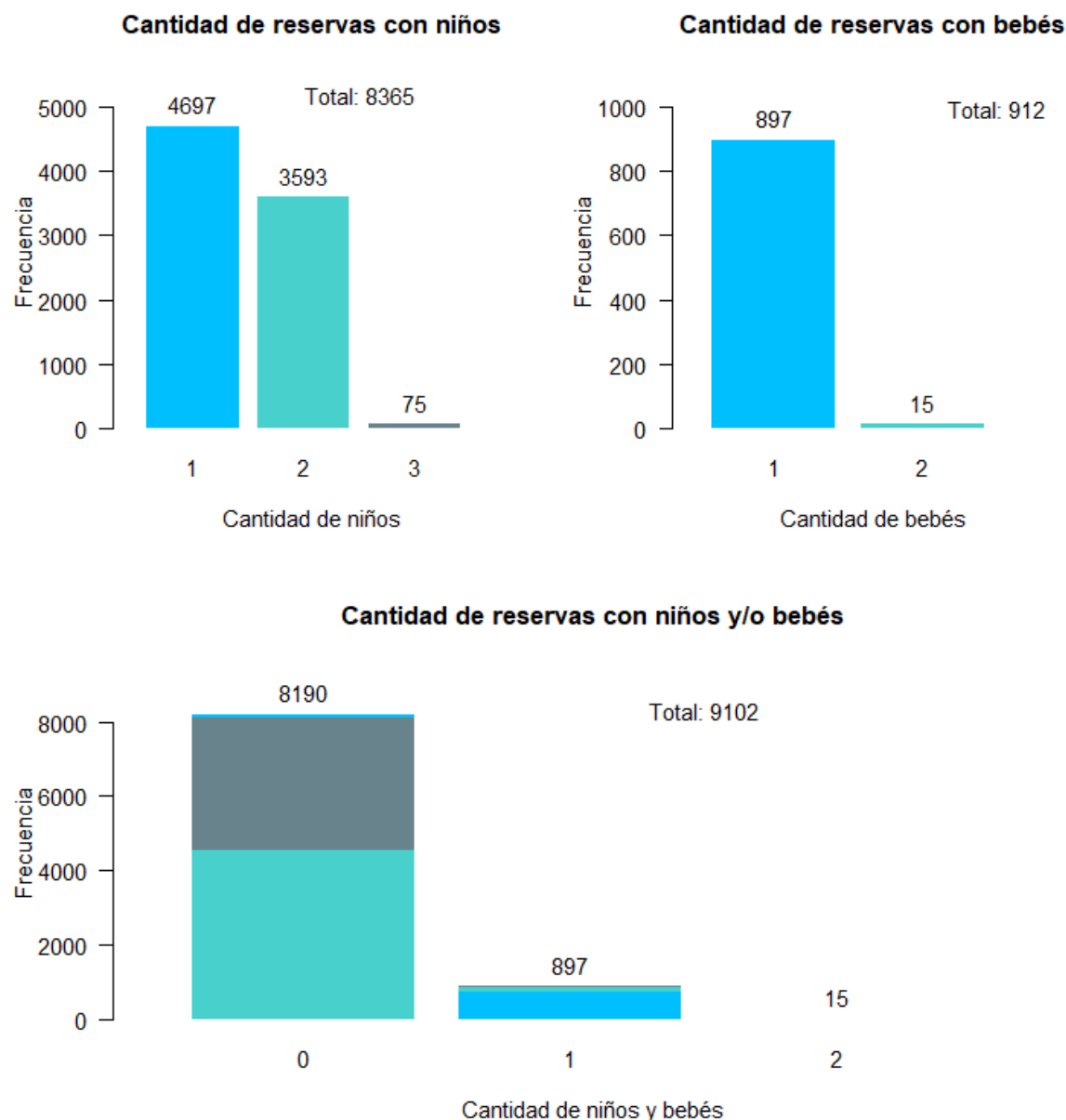




Tabla 5:

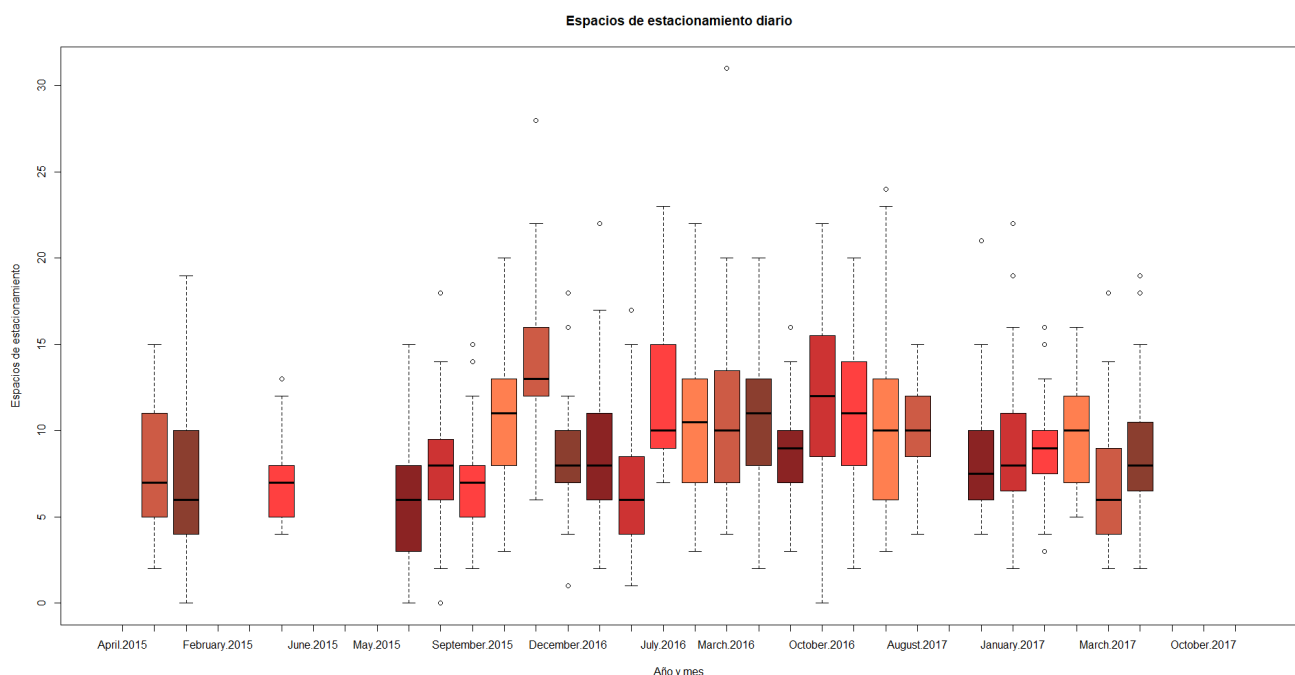
	babies		
children	0	1	2
0	0	725	12
1	4554	140	3
2	3561	32	0
3	75	0	0

Se observa que priman las reservas con solo 1 niño, seguida de las de 2 niños, siendo mucho menor la de 3 niños. Además, es de considerar que son 8 365 reservas del total de 87 443 reservas quienes reservan con algún niño. Adicionalmente, solo 912 personas reservaron considerando a bebés (y, en su gran mayoría, solo 1 bebé).

Además, del total de personas que reservaron con 0 bebés, 4 554 incluyen 1 niño; 3 561, 2 niños y 75, 3 niños. Así mismo, de quienes reservaron con 1 bebé, 725 reservaron sin niños; 140 con 1 bebé y 32, 2 bebés. Entre quienes reservaron con 2 bebés, 12 no incluyen niños y solo 3, un niño.

#### f. ¿Es importante contar con espacios de estacionamiento?

Gráfico 6:<sup>2</sup>



<sup>2</sup> Se utilizó la librería dplyr para agrupar los datos de cantidad de parqueos diarios  
<https://www.r-project.org/nosvn/pandoc/dplyr.html>

Tabla 6:

	n_parkings																															
arrival_date_month	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	28	31					
April	0	0	0	5	0	4	5	5	4	6	0	7	6	4	3	2	3	2	1	0	1	0	0	1	1	0	0					
August	0	0	1	1	4	6	7	4	4	11	7	8	16	5	6	5	0	2	2	0	1	1	1	0	0	1	0					
December	1	3	1	2	4	5	8	6	7	7	3	5	3	2	1	0	2	0	1	1	0	0	0	0	0	0	0					
February	0	0	2	1	3	4	7	5	10	5	6	3	3	2	1	1	1	1	0	0	0	1	1	0	0	0	0					
January	0	1	3	3	7	6	6	7	6	6	3	4	4	1	0	1	1	1	0	1	0	0	1	0	0	0	0					
July	0	0	0	1	8	5	4	11	18	10	11	6	4	2	2	4	2	0	2	1	0	1	0	1	0	0	0					
June	0	0	0	1	3	4	4	6	7	2	4	8	3	7	2	3	2	3	0	0	0	0	1	0	0	0	0					
March	0	0	1	4	5	7	4	5	4	8	3	4	4	3	2	1	1	1	2	0	2	0	0	0	0	0	1					
May	0	0	3	0	5	1	5	7	5	8	4	3	6	5	1	1	1	3	2	1	1	0	0	0	0	0	0					
November	1	1	0	7	6	2	8	8	7	3	6	1	6	1	1	1	1	0	0	0	0	0	0	0	0	0	0					
October	2	0	1	1	1	2	5	8	7	5	8	3	3	3	3	1	3	1	2	1	0	0	2	0	0	0	0					
September	0	0	2	1	4	4	3	10	7	7	3	1	5	3	6	1	1	0	1	0	1	0	0	0	0	0	0					

Los datos fueron calculados acumulando la cantidad diaria de parqueos requeridos por cada una de las reservas. A partir de esta acumulación diaria se realizó la primera gráfica, en la que observamos cómo ha variado este requerimiento en el tiempo. Así también, en la tabla se puede observar la cantidad requerida para parquear diariamente por meses.

Además, se ha de aclarar que la última fecha de llegada registrada es “2017-08-31”, por lo que se observa que en los meses siguientes a mayo del 2017 no se registraron requerimientos por parqueo.

#### g. ¿En qué meses del año se producen más cancelaciones de reservas?

Gráfico 7:

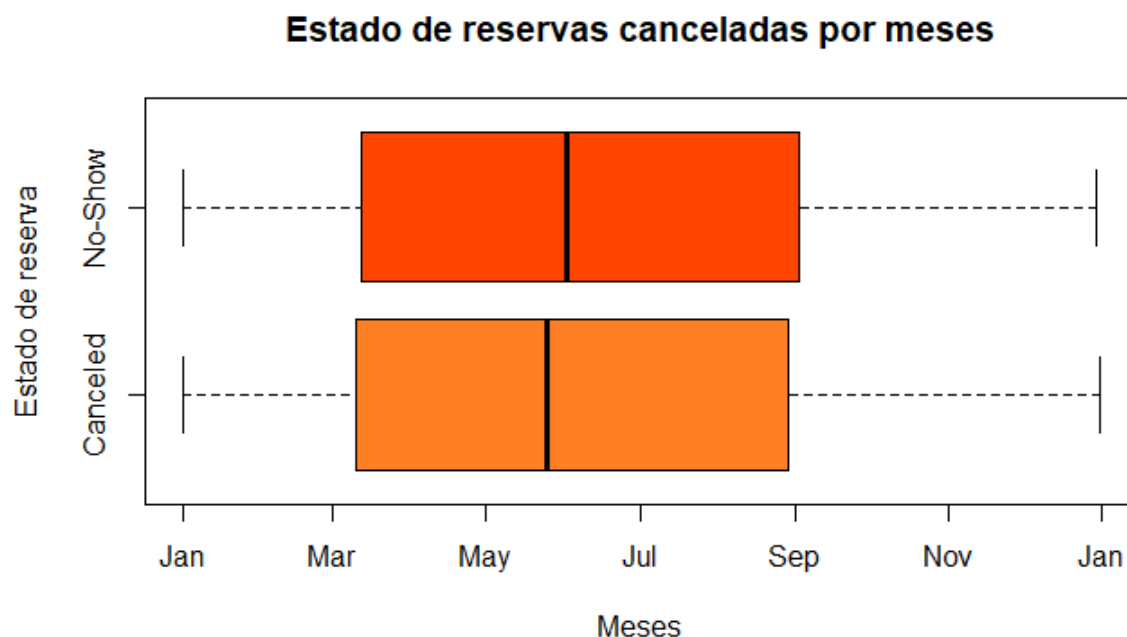


Tabla 7: Estados de reservas canceladas por meses

	months											
reservation_status	01:January	02:February	03:March	04:April	05:May	06:June	07:July	08:August	09:September	10:October	11:November	12:December
Canceled	2437	2426	2543	2265	2209	1919	2012	1623	1437	1527	1284	1367
Check-out	0	0	0	0	0	0	0	0	0	0	0	0
No-Show	69	140	116	75	102	81	91	83	62	64	69	63

Como se puede observar, entre los meses de marzo a septiembre se dieron las principales cancelaciones; sin embargo, son los meses entre mayo y julio, los más frecuentes. Así también, la mayoría de las cancelaciones son realizadas por las mismas personas.

#### 4. CONCLUSIONES PRELIMINARES

A partir de los gráficos observados anteriormente, hemos de concluir a partir de las preguntas planteadas lo siguiente:

¿Cuántas reservas se realizan por tipo de hotel? o ¿Qué tipo de hotel prefiere la gente?

Tras analizar los datos, se observa que hay 53428 reservas en los hoteles tipo City de las cuales 37379 son check-in y 16049 cancelados. En el caso de los Resort Hotel presenta 34015 reservaciones de las cuales 26000 son check-in y 8015 son cancelaciones. A partir de esta data se puede llegar a inferir que el tipo de hotel que las personas prefieren es el City Hotel. Como los hoteles tipo City son preferidos por las personas, se recomendaría aumentar la cantidad de hoteles de este tipo.

¿Está aumentando la demanda con el tiempo?

Tras analizar los datos, se puede observar que la demanda a través del tiempo está en aumento. En el año 2017 se presenta menor cantidad de alojamientos porque solo se está contando hasta el mes de julio. Si comparamos solo los meses que el 2017 presenta, podemos observar que el 2017 tiene mayor cantidad de alojamientos en comparación al 2016 en las mismas fechas. Entonces se puede afirmar que hay un aumento de demanda con el tiempo. Como la demanda está aumentando en el tiempo se recomienda crear paquetes personalizados y promociones para sus clientes para seguir aumentando la demanda.

¿Cuándo se producen las temporadas de reservas: alta, media y baja?

Se puede observar que la temporada baja se da en los meses de Setiembre, Octubre, Noviembre y Diciembre. La temporada media se da en enero, febrero, marzo y abril. Por último, la temporada alta se da en mayo, junio, julio y agosto. Se recomendaría premiar la fidelidad de sus huéspedes en temporada baja para poder aumentar las reservas durante estos meses.

¿Cuándo es menor la demanda de reservas?

Se puede observar que la demanda de reservas es menor entre los meses de noviembre y enero. La demanda de reservas es menor a finales e inicios de años según la data presentada. El mes con menos demanda es Enero. Se recomendaría ofrecer paquetes especiales o organizar eventos en el mes de Enero para poder aumentar la demanda.

¿Cuántas reservas incluyen niños y/o bebés?

Se destaca que al menos el 10% reserva con algún niño o bebé ( $(9\ 102 \cdot 100\%) / 87\ 443 = 10.4\%$ ). Así también, que de las 9 102 reservas que incluyeron niños o bebés, solo 175 reservaron incluyendo ambos. Además que, principalmente, reservan considerando 1 a 2 niños y no bebés. A partir de ello se puede sugerir colocar pequeños espacios recreativos para menores de edad.

¿Es importante contar con espacios de estacionamiento?

A partir de los datos analizados, no se considera importante contar con espacios recreativos, puesto que estos no han sido requeridos en los últimos meses del 2017. Sin embargo, también se observa que los que fueron considerados antes de esa fecha, requirieron de principalmente entre 10 a 15 parqueos. Por ello, aunque no es de importancia, tal vez sería conveniente realizar alguna asociación con algún local cercano.

¿En qué meses del año se producen más cancelaciones de reservas?

Entre los meses de mayo y julio se producen más cancelaciones de reservas, las mismas que son, casi en su totalidad, realizadas por las mismas personas. A partir de ello habría que analizar la causa. Por ejemplo, si una empresa de hoteles externa a las consideradas ofrece algún beneficio adicional, podría realizarse alguna campaña publicitaria por esas fechas. Si el motivo fuera que las personas tienen más dificultades laborales en esos meses, podrían plantearse ofertas económicas por 1 o 2 días de reserva o considerar algunos espacios como oficina, entre otros.

Consideramos que todos estos datos son de gran importancia para los hoteles y, después de un mayor análisis, podrían considerar oportunas algunas de las sugerencias mencionadas anteriormente.

## 5. BIBLIOGRAFÍA

- Antonio N., Almeida A., Nunes L. (2019) Hotel booking demand datasets. Data in Brief. Vol. 22. Pages 41-49. ISSN 2352-3409. Recuperado de:  
<https://doi.org/10.1016/j.dib.2018.11.126> .  
(<https://www.sciencedirect.com/science/article/pii/S2352340918315191> )
- Coder, R. (2021, 18 noviembre). Plot en R. R CODER. Recuperado 4 de mayo de 2022, de <https://r-coder.com/plot-en-r/>
- Hernández, F. et. Correa, J. (2020). Gráficos con R. Recuperado el 2 de mayo del 2022, de: <https://fhernanb.github.io/Graficos-con-R/doscuanti.html>