

Tipología y ciclo de vida de los datos - Práctica 2

Javier Guimerans Alonso, Gerson Villalba Arana

29/05/2022

Contents

Descripción del dataset	2
Carga de datos	2
Descripción de variables	3
Integración y selección	4
Limpieza de datos	5
Valores nulos	5
Cambio de tipo	6
Detección de outliers	7
Análisis de datos	9
Comprobación de normalidad	9
Análisis 1: Influencia del precio medio en la cancelación	11
Análisis 2: Influencia de procedencia de reserva sobre la cancelación	13
Análisis 3: Predicción de cancelación de reserva y factores que más influyen	14
Análisis 4: Influencia de régimen de alojamiento sobre el precio en el resort	17
Representación de resultados	23
Resolución del problema	23
Exportación de datos	23
Código	24
Contribución	24
Bibliografía	24

Descripción del dataset

En la presente práctica se va a realizar los procesos de selección, limpieza, preparación y visualización de datos para posteriormente, realizar análisis sobre éstos.

El tema elegido para la realización de la práctica es el de optimización de procesos de reserva de noches de hotel en web especializadas. Como base para el trabajo, se parte de un dataset con información histórica sobre reservas de hoteles en dos tipos de hoteles distintos en **Portugal**. El dataset se encuentra disponible públicamente en el repositorio de Kaggle:

<https://www.kaggle.com/insatanic/hotel-booking-data-analysis>.

En base a la información proporcionada por dicho dataset, en primer lugar limpiaremos los datos para a continuación poder realizar distintos análisis sobre la influencia que distintas variables que tenemos disponibles tienen sobre la cancelación de reservas. Esta información puede ser valiosísima tanto para hoteles como para webs de reserva, ya que con esta información:

- Se tiene una visión más realista de la ocupación real que va a tener un hotel en base a las reservas actuales, lo cual puede ayudar a optimizar recursos y gastos: ¿es necesario tener productos perecederos para 500 comensales en el buffet, o es suficiente contar para 400? ¿Cuántas habitaciones van a tener que ser limpiadas?
- Se puede sobre-reservar el hotel (overbooking) en cierta medida, teniendo en cuenta que algunas de las reservas actuales se van a acabar cancelando. De esta manera se optimiza la ocupación del hotel y por lo tanto los ingresos.
- Se puede conocer el perfil de cliente que es más propenso a cancelar. De esta forma, podemos dirigir nuestro marketing al público más interesante desde el punto de vista de negocio: aquellos con menor probabilidad de cancelar una reserva.
- Se pueden buscar estrategias para evitar la cancelación de reservas sobre aquellas que detectamos con más probabilidades de ser canceladas. Podemos ofrecer incentivos para los huéspedes, como cambiarle gratuitamente a una habitación superior, ofrecerles descuentos en actividades del lugar, etc.
- Toda esta información puede repercutir en miles de euros en aumento de ingresos y optimización de costes en un sólo hotel, en millones en una web de reserva con miles de hoteles que pierde una comisión cada vez que un cliente cancela una reserva.

Por supuesto, el alcance de este trabajo es limitado y trataremos sólo con este dataset, pero podría ser ampliado en gran medida, añadiendo nueva información, tanto en forma de nuevas muestras como de nuevas características que nos aportasen más información. En cualquier caso, el dataset es ya de por sí muy completo, y ni siquiera realizaremos un análisis sobre todos los atributos que tenemos disponibles.

Carga de datos

En primer lugar cargaremos los datos originales de los que disponemos y mostramos su estructura.

```
library(tidyverse)
library(reshape)
library(psych)
library(MASS)
library(agricolae)
```

```
df <- read.csv('../data/hotel_bookings_raw.csv', sep=",", na.strings=c("", "NULL"))
str(df)
```

```
## 'data.frame': 119390 obs. of 32 variables:
```

```

## $ hotel : chr "Resort Hotel" "Resort Hotel" "Resort Hotel" "Resort Hotel"
## $ 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 : chr "July" "July" "July" "July" ...
## $ arrival_date_week_number : int 27 27 27 27 27 27 27 27 27 ...
## $ arrival_date_day_of_month : int 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 : chr "0" "0" "0" "0" ...
## $ babies : int 0 0 0 0 0 0 0 0 0 ...
## $ meal : chr "BB" "BB" "BB" "BB" ...
## $ country : chr "PRT" "PRT" "GBR" "GBR" ...
## $ market_segment : chr "Direct" "Direct" "Direct" "Corporate" ...
## $ distribution_channel : chr "Direct" "Direct" "Direct" "Corporate" ...
## $ is_repeated_guest : int 0 0 0 0 0 0 0 0 0 ...
## $ previous_cancellations : int 0 0 0 0 0 0 0 0 0 ...
## $ previous_bookings_not_canceled: int 0 0 0 0 0 0 0 0 0 ...
## $ reserved_room_type : chr "C" "C" "A" "A" ...
## $ assigned_room_type : chr "C" "C" "C" "A" ...
## $ booking_changes : int 3 4 0 0 0 0 0 0 0 ...
## $ deposit_type : chr "No Deposit" "No Deposit" "No Deposit" "No Deposit" ...
## $ agent : int NA NA NA 304 240 240 NA 303 240 15 ...
## $ company : int NA NA NA NA NA NA NA NA NA ...
## $ days_in_waiting_list : int 0 0 0 0 0 0 0 0 0 ...
## $ customer_type : chr "Transient" "Transient" "Transient" "Transient" ...
## $ adr : num 0 0 75 75 98 ...
## $ required_car_parking_spaces : int 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 : chr "Check-Out" "Check-Out" "Check-Out" "Check-Out" ...
## $ reservation_status_date : chr "2015-07-01" "2015-07-01" "2015-07-02" "2015-07-02" ...

```

Descripción de variables

Como podemos ver, tenemos las siguientes variables:

- **hotel.** Hotel tratado. Variable categórica.
- **is_canceled.** Especifica si la reserva a sido (1) o no (0) finalmente cancelada.
- **lead_time.** Tiempo de antelación con la que se reservó el hotel, en días. Variable numérica.
- **arrival_date_year.** Año de comienzo de reserva. Variable numérica.
- **arrival_date_month.** Mes de comienzo de reserva. Variable numérica.
- **arrival_date_week_number.** Semana del año en el que comienza de reserva. Variable numérica.
- **arrival_date_day_of_month.** Día del mes en el que comienza de reserva. Variable numérica.
- **stays_in_weekend_nights.** Noches totales de reserva en fin de semana. Variable numérica.
- **stays_in_week_nights.** Noches totales de reserva entre semana. Variable numérica.
- **adults.** Número de adultos en la reserva. Variable numérica.
- **children.** Número de niños en la reserva. Variable numérica.
- **babies.** Número de bebés en la reserva. Variable numérica.
- **meal.** Régimen de comidas contratado. Variable categórica.
- **country.** País de procedencia de los clientes en formato ISO3. Variable categórica.
- **market_segment.** Designación de segmento de mercado. Variable categórica.
- **distribution_channel.** Canal de reserva utilizado para la reserva. Variable categórica.

- **is_repeated_guest**. Indica si el huésped ya ha reservado en el hotel previamente. Variable categórica booleana.
 - **previous_cancellations**. Indica el número de veces que el cliente ha cancelado una reserva en el hotel. Variable numérica.
 - **previous_bookings_not_canceled**. Indica el número de veces que el cliente ha reservado y no cancelado en el hotel. Variable numérica.
 - **reserved_room_type**. Tipo de habitación reservada. Variable categórica.
 - **assigned_room_type**. Tipo de habitación asignada, que puede o no coincidir con la reservada. Variable categórica.
 - **booking_changes**. Cambios que se han realizado en la reserva desde que se realiza. Variable numérica.
 - **deposit_type**. Indica si el cliente ha realizado un depósito o no para asegurarse la reserva. Variable categórica.
 - **agent**. Identificador del agente que ha realizado la reserva. Variable categórica.
 - **company**. Identificador de la compañía que ha realizado la reserva. Variable categórica.
 - **days_in_waiting_list**. Número de días que la reserva estuvo pendiente de confirmación hacia el cliente. Variable numérica.
 - **customer_type**. Tipo de cliente. Variable categórica.
 - **adr**. Average Daily Rate. Coste promedio de la habitación por noche en la reserva. Variable numérica
 - **required_car_parking_spaces**. Número de plazas de parking reservadas. Variable numérica
 - **total_of_special_requests**. Cantidad de solicitudes especiales que ha realizado el huésped. Variable numérica
 - **reservation_status**. Estado de la reserva. Variable categórica.
 - **reservation_status_date**. Fecha de la última actualización de estado. Variable de fecha.
-

Integración y selección

Comprobamos en primer lugar el número de elementos y atributos que tenemos en el dataset.

```
nrow(df)
## [1] 119390

ncol(df)
## [1] 32
```

Como disponemos de un dataset muy grande, vamos a quedarnos con un subconjunto de éste que nos permita realizar los análisis que queremos. Nos quedaremos con los siguientes atributos, ya que el resto no los utilizaremos en la presente práctica:

- **hotel**
- **is_canceled**
- **arrival_date_month**
- **lead_time**
- **meal**
- **adr**
- **lead_time**

Además, crearemos tres variables binarias nuevas para simplificar algunas de las existentes:

- **total_nights**, que será la suma de las variables **stays_in_weekend_nights** y **stays_in_week_nights**, y que indicará por tanto el número total de noches de reserva.
- **is_international**, que crearemos a partir de **country**. Indicará si la reserva es internacional o no (de Portugal).
- **more_than_two** a partir de **adults**, **children** y **babies**. Variable binaria que indicará si la reserva es para más de dos personas en total. En esta categoría se incluirán por lo tanto familias como grupos que reserven más de una habitación y familias que reserven una habitación familiar.
- **with_children**, a partir de **children** y **babies**. Indicará si se viaja con algún niño o bebé.

```
df$children <- as.integer(df$children)
df$more_than_two <- (df$adults + df$children + df$babies) > 2
df$with_children <- (df$children + df$babies) != 0
df$is_international <- df$country != 'PRT'
df$total_nights <- df$stays_in_weekend_nights + df$stays_in_week_nights
df <- df %>% dplyr::select(hotel, is_canceled, arrival_date_month, lead_time, meal, adr, lead_time, total_nights, is_international, more_than_two, with_children)
str(df)
```

```
## 'data.frame': 119390 obs. of 10 variables:
##   $ hotel           : chr "Resort Hotel" "Resort Hotel" "Resort Hotel" "Resort Hotel" ...
##   $ is_canceled     : int 0 0 0 0 0 0 0 1 1 ...
##   $ arrival_date_month: chr "July" "July" "July" "July" ...
##   $ lead_time       : int 342 737 7 13 14 14 0 9 85 75 ...
##   $ meal            : chr "BB" "BB" "BB" "BB" ...
##   $ adr             : num 0 0 75 75 98 ...
##   $ total_nights    : int 0 0 1 1 2 2 2 2 3 3 ...
##   $ is_international: logi FALSE FALSE TRUE TRUE TRUE TRUE ...
##   $ more_than_two   : logi FALSE FALSE FALSE FALSE FALSE FALSE ...
##   $ with_children   : logi FALSE FALSE FALSE FALSE FALSE FALSE ...
```

Limpieza de datos

Valores nulos

Comprobamos las variables que tienen valores nulos.

```
colSums(is.na(df))
```

	hotel	is_canceled	arrival_date_month	lead_time
##	0	0	0	0
##	meal	adr	total_nights	is_international
##	0	0	0	488
##	more_than_two	with_children		
##	4	4		

Además, vamos a ver los valores únicos disponibles en las variables categóricas.

```

df %>% select_if(negate(is.numeric)) %>% sapply(function(x) unique(x))

## $hotel
## [1] "Resort Hotel" "City Hotel"
##
## $arrival_date_month
## [1] "July"      "August"     "September" "October"    "November"   "December"
## [7] "January"   "February"   "March"      "April"      "May"        "June"
##
## $meal
## [1] "BB"        "FB"        "HB"        "SC"        "Undefined"
##
## $is_international
## [1] FALSE  TRUE   NA
##
## $more_than_two
## [1] FALSE  TRUE   NA
##
## $with_children
## [1] FALSE  TRUE   NA

sum(df$meal == 'Undefined')

## [1] 1169

```

Vemos que tenemos valores nulos en las variables *is_international*, *more_than_two* y *with_children*. Además, podemos ver que en la variables *meal* también tenemos valores nulos, esta vez bajo la categoría “*Undefined*”, que no viene a ser más que un valor no disponible, y lo cambiamos para que se vea reflejado como *NA*. Teniendo en cuenta que el número de registros que tienen valores nulos es muy bajo en comparación con el total, optamos por eliminar dichos registros del dataset.

```

df$meal[df$meal == 'Undefined'] <- NA
df <- df %>% drop_na(meal, is_international, more_than_two, with_children)
colSums(is.na(df))

```

	hotel	is_canceled	arrival_date_month	lead_time
##	0	0	0	0
##	meal	adr	total_nights	is_international
##	0	0	0	0
##	more_than_two	with_children		
##	0	0		

Cambio de tipo

Cambiamos todas las variables categóricas a tipo “factor”, ya que de esta forma se trabaja de forma más eficiente con ellas, y además será requerido para utilizar ciertos modelos. Además, nos aseguramos de que “January” sea el primer factor del mes, para así asegurarnos que lo tenemos de referencia cuando se realice *one hot encoding* sobre esta variable. Además, cambiaremos la variable *is_canceled* a tipo booleano.

```

df$is_canceled <- as.logical(df$is_canceled)
df <- df %>% mutate_if(is.character, as.factor)
df$arrival_date_month <- relevel(df$arrival_date_month , "January")
str(df)

## 'data.frame': 117733 obs. of 10 variables:
## $ hotel : Factor w/ 2 levels "City Hotel","Resort Hotel": 2 2 2 2 2 2 2 2 2 ...
## $ is_canceled : logi FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ arrival_date_month: Factor w/ 12 levels "January","April",...: 6 6 6 6 6 6 6 6 6 ...
## $ lead_time : int 342 737 7 13 14 14 0 9 85 75 ...
## $ meal : Factor w/ 4 levels "BB","FB","HB",...: 1 1 1 1 1 1 1 2 1 3 ...
## $ adr : num 0 0 75 75 98 ...
## $ total_nights : int 0 0 1 1 2 2 2 2 3 3 ...
## $ is_international : logi FALSE FALSE TRUE TRUE TRUE ...
## $ more_than_two : logi FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ with_children : logi FALSE FALSE FALSE FALSE FALSE FALSE ...

```

Detección de outliers

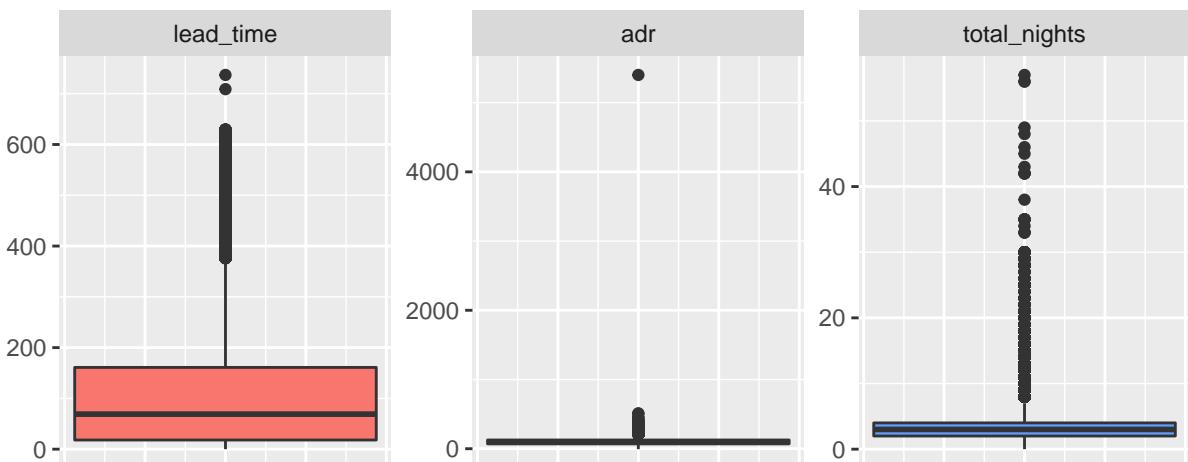
Realizamos una representación visual de los datos numéricos con boxplot para ver visualmente los outliers que podamos tener, y obtenemos las estadísticas descriptivas de éstas.

```

df %>%
  select_if(is.numeric) %>%
  melt() %>%
  ggplot(aes(y=value, fill=variable))+
  geom_boxplot()+
  facet_wrap(~variable, scales="free") +
  labs(x="", y="")+
  ggtitle("Distribución de variables numéricas")+
  theme(legend.position="none", axis.text.x=element_blank(), axis.ticks.x = element_blank())

```

Distribución de variables numéricas



```

df %>% select_if(is.numeric) %>% describe()

##          vars     n   mean      sd median trimmed    mad    min   max range
## lead_time      1 117733 104.48 107.17      69    87.70 88.96  0.00 737 737.00
## adr           2 117733 102.10  50.44      95    98.03 41.51 -6.38 5400 5406.38
## total_nights   3 117733   3.42   2.54      3     3.05  1.48  0.00   57  57.00
##                skew kurtosis   se
## lead_time      1.34      1.68 0.31
## adr            10.71   1035.35 0.15
## total_nights   3.13     23.94 0.01

```

Para *total_nights*, como una reserva no puede ser de ninguna noche, tomamos este valor como erróneo. Además, tomaremos como valores outliers aquellas reservas por más de un mes (30 días).

Para la variable *adr* evidente que existen valores erróneos en los datos:

- No es posible un coste medio de la habitación menor que cero.
- Parece poco probable que una habitación de hotel en Portugal pueda ser reservada por menos de 20€ la noche, poniendo un límite bastante bajo. Tomaremos las muestras con un valor inferior a este como erróneas y las eliminaremos.
- Parece poco probable que habitaciones se hayan reservado con un precio promedio de más de 5000€ la noche cuando el valor medio de todas las reservas es de 102€. Claramente son valores outliers que eliminaremos. Nos quedaremos sólo con los registros con un coste promedio menor o igual a 400€ la noche, que parece un límite superior razonable.

En la variable *lead_time* vemos valores muy altos, pero no parece que sean erróneos, por lo que la dejamos como está.

```

df <- df %>% mutate(total_nights = replace(total_nights, total_nights<1 |
                                              total_nights>30, NA))
df <- df %>% mutate(adr = replace(adr, adr<20 | adr>400, NA))
colSums(is.na(df))

```

```

##          hotel      is_canceled arrival_date_month      lead_time
##             0                  0                  0                  0
##          meal        adr      total_nights is_international
##             0       2313                 714                  0
## more_than_two with_children
##             0                      0

```

Con los valores que hemos eliminado por ser erróneos y que tenemos ahora como valor perdido, vamos a realizar una imputación por la media. En el caso de la variable *total_nights*, al ser de tipo entero, la rendondeamos.

```

df$total_nights[is.na(df$total_nights)] <- as.integer(mean(df$total_nights, na.rm=TRUE))
df$adr[is.na(df$adr)] <- mean(df$adr, na.rm=TRUE)
colSums(is.na(df))

```

```

##          hotel      is_canceled arrival_date_month      lead_time
##             0                  0                  0                  0

```

```

##               meal                  adr      total_nights  is_international
##               0                   0                   0                   0
## more_than_two with_children
##               0                   0

```

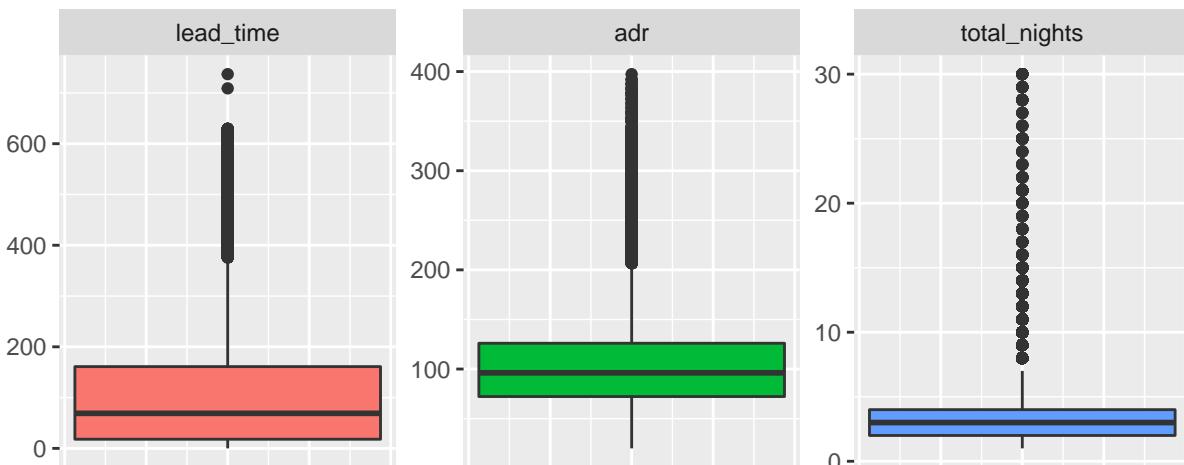
Comprobamos la distribución de las variables numéricas tras la limpieza de outliers.

```

df %>%
  select_if(is.numeric) %>%
  melt() %>%
  ggplot(aes(y=value, fill=variable))+
  geom_boxplot()+
  facet_wrap(~variable, scales="free") +
  labs(x="", y "")+
  ggtitle("Distribución de variables numéricas tras eliminar outliers")+
  theme(legend.position="none", axis.text.x=element_blank(),
        axis.ticks.x = element_blank())

```

Distribución de variables numéricas tras eliminar outliers



Análisis de datos

Comprobación de normalidad

Para comprobar la normalidad de las variables numéricas, en primer lugar realizamos una representación en histograma de éstas.

```

df %>%
  select_if(is.numeric) %>%
  melt() %>%
  ggplot(aes(value))+
  geom_histogram(fill='skyblue4', bins=25, alpha=0.8)+

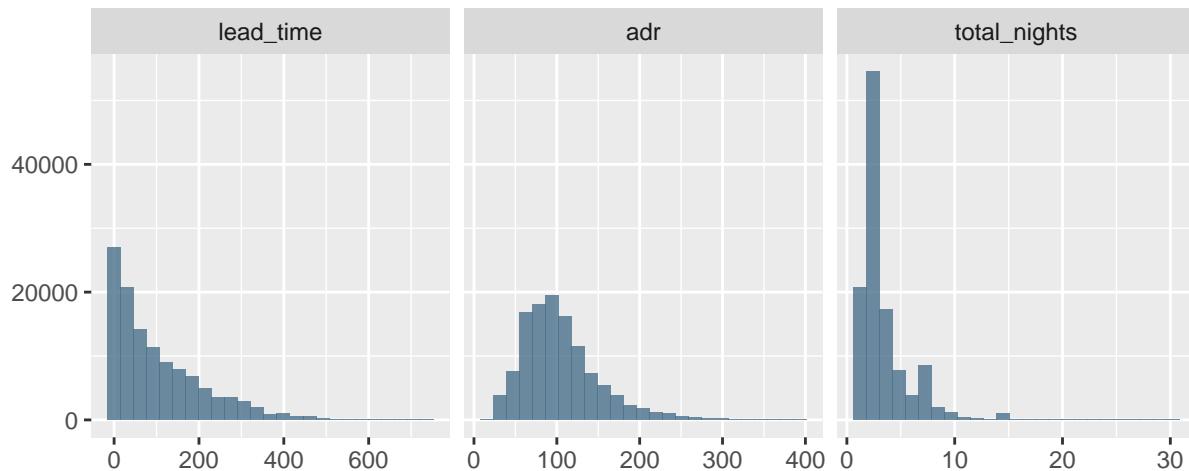
```

```

facet_wrap(~variable, scale="free_x") +
labs(x="", y "")+
ggtitle("Histograma de variables numéricas")+
theme(legend.position="none")

```

Histograma de variables numéricas



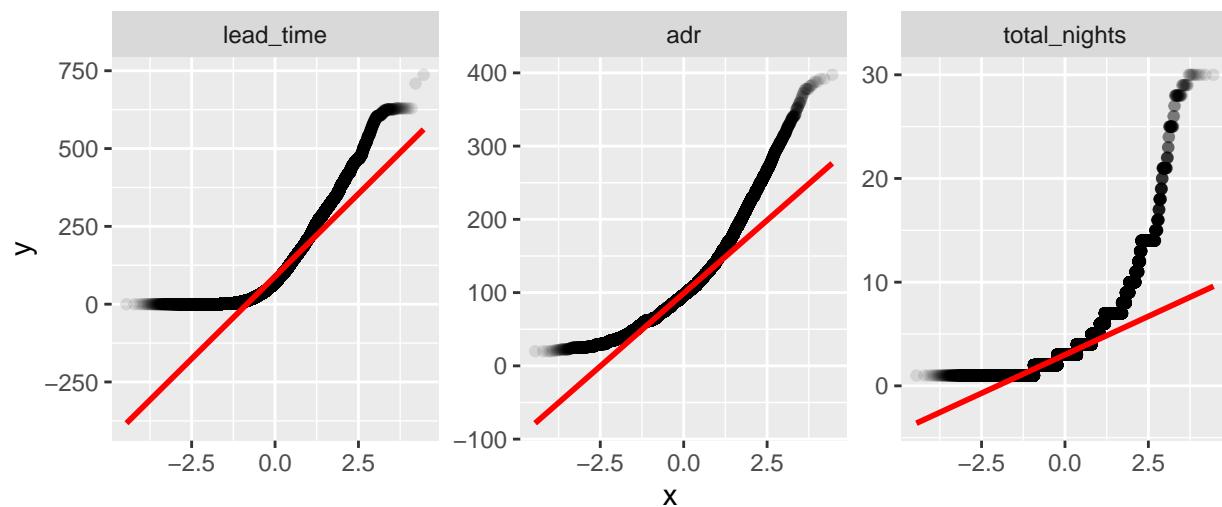
A continuación realizamos la representación del gráfico QQ para evaluar normalidad.

```

df %>%
  select_if(is.numeric) %>%
  melt() %>%
  ggplot(aes(sample = value))+
  stat_qq(alpha=0.1)+
  stat_qq_line(color='red', size=1)+
  facet_wrap(~variable, scales="free") +
  ggtitle("QQ Plot Variables numéricas")

```

QQ Plot Variables numéricas



Podemos comprobar visualmente tanto por el histograma como por el gráfico QQ que ninguna de las tres variables numéricas que tenemos siguen una distribución normal. Podríamos realizar un test Shapiro-Wilk para testear normalidad, pero viendo los resultados gráficos no es necesario. En todo caso, hay que tener en cuenta que tenemos un número muy grande de muestras (»30), por lo que por el teorema central del límite podremos decir que la media de cualquiera de estas tres variables seguirán una distribución normal.

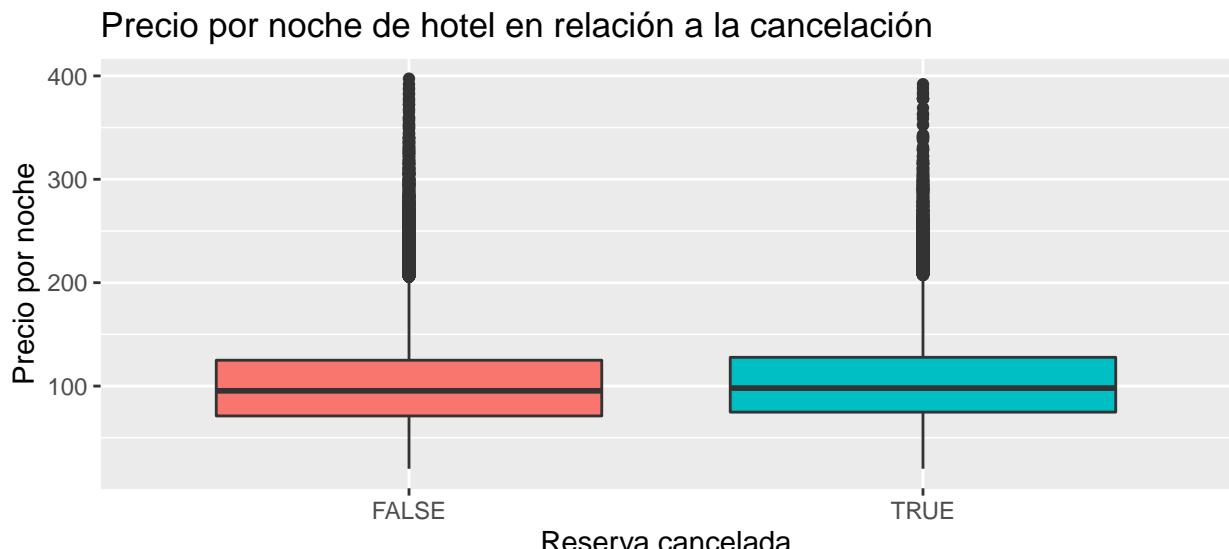
Además, si para algún análisis necesitáramos que las distribución de la variable fuese normal, podríamos realizar transformaciones sobre estas variables para normalizarlas. En principio y mientras no sea necesario, no lo haremos.

Análisis 1: Influencia del precio medio en la cancelación

Nos planteamos la siguiente pregunta: ¿Es el precio medio de las reservas canceladas distinto al de las no canceladas?

Realizamos en primer lugar con un boxplot la representación de la variable *adr* en función de si la reserva es cancelada o no.

```
df %>%
  ggplot(aes(x=is_canceled, y=adr, fill=is_canceled))+
  geom_boxplot()+
  labs(x="Reserva cancelada", y="Precio por noche")+
  ggtitle("Precio por noche de hotel en relación a la cancelación")+
  theme(legend.position="none")
```



Podemos ver por la gráfica que no queda del todo clara la respuesta a nuestra pregunta. Por otro lado, podemos mirar la media de coste de habitación por hotel.

```
df %>%
  group_by(is_canceled) %>%
  summarize(adr.mean = mean(adr, na.rm = TRUE),
            adr.min = min(adr, na.rm = TRUE),
            adr.max=max(adr, na.rm = TRUE))
```

```
## # A tibble: 2 x 4
```

```

##   is_canceled adr.mean adr.min adr.max
##   <lg1>          <dbl>   <dbl>   <dbl>
## 1 FALSE           103.     20     397.
## 2 TRUE            106.     20     392

```

Vemos que en la muestra que tenemos el coste medio de la noche en las reservas canceladas es algo mayor que en las no canceladas, si bien esta diferencia no es muy grande.

Planteamos el contraste de hipótesis sobre las medias de la siguiente forma:

$$H_0 : \mu_1 = \mu_2$$

$$H_1 : \mu_1 \neq \mu_2$$

Donde μ_1 será la media de coste de habitación en las reservas canceladas y μ_2 será la media de coste de habitación en las reservas no canceladas. Tal y como lo hemos planteado, se trata de un contraste bilateral sobre la media de dos muestras

Antes de realizar el contraste planteado, tenemos que comprobar la igualdad de varianzas de las dos poblaciones, que estimaremos a partir de ambas.

```
var.test(df$adr[df$is_canceled], df$adr[!df$is_canceled])
```

```

##
## F test to compare two variances
##
## data: df$adr[df$is_canceled] and df$adr[!df$is_canceled]
## F = 0.95017, num df = 43866, denom df = 73865, p-value = 2.209e-09
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.9344463 0.9661998
## sample estimates:
## ratio of variances
##                 0.9501733

```

Como el p-valor nos sale muy bajo, tenemos que rechazar la hipótesis nula que implicaría una igualdad de varianzas. Por lo tanto, asumiremos como cierta la hipótesis alternativa de que ambas son distintas. Con esta información ya podemos realizar el contraste planteado.

```
t.test(df$adr[df$is_canceled], df$adr[!df$is_canceled],
       alternative='two.sided', var.equal = FALSE, conf.level = 0.95)
```

```

##
## Welch Two Sample t-test
##
## data: df$adr[df$is_canceled] and df$adr[!df$is_canceled]
## t = 10.692, df = 94023, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  2.393197 3.467597
## sample estimates:
## mean of x mean of y
## 105.8862 102.9558

```

Obtenemos un p-valor muy bajo, por lo que tenemos que rechazar la hipótesis nula que asumía las medias de precio de ambos hoteles iguales. Podemos concluir, por lo tanto, que efectivamente existe una diferencia estadísticamente significativa en el precio medio de la noche entre reservas que acaban siendo canceladas y las que no, y que las canceladas tienen una media de coste mayor, que sería lo esperado.

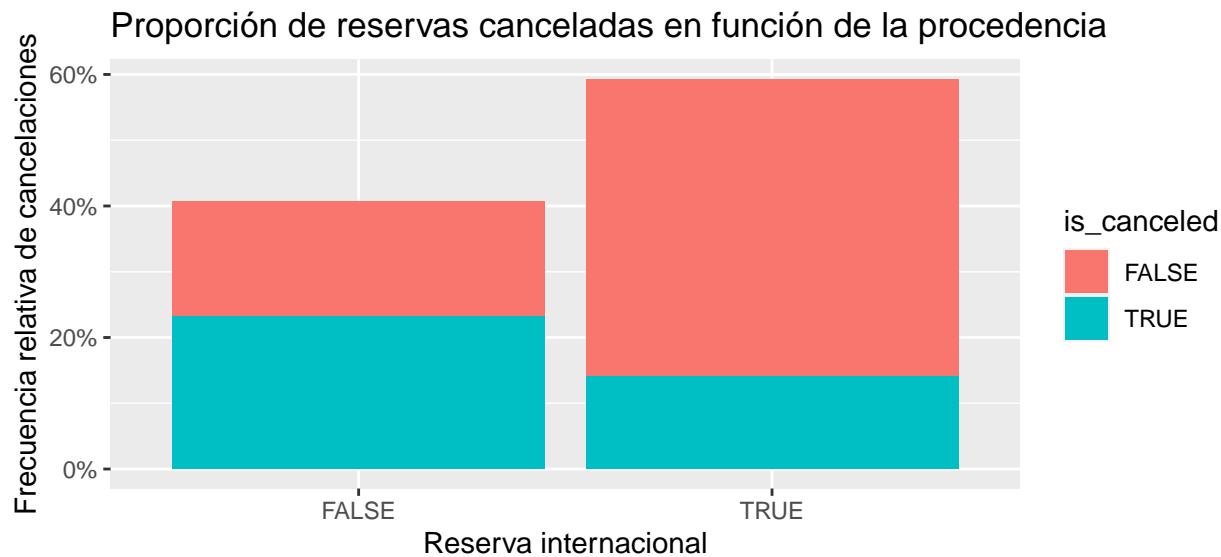
Análisis 2: Influencia de procedencia de reserva sobre la cancelación

Teniendo en cuenta la mayor facilidad que puede tener un viajero nacional para cancelar una reserva realizada en un hotel, nos planteamos la siguiente pregunta de investigación:

¿El la proporción de reservas canceladas menor en el caso de viajeros internacionales que en el de nacionales?

Realizamos en primer lugar una representación visual de la proporción de reservas canceladas en reservas nacionales e internacionales.

```
df %>%
  ggplot(aes(x=is_international, fill=is_canceled))+
  geom_bar(aes(y = (.count..)/sum(..count..))) +
  scale_y_continuous(labels=scales::percent) +
  labs(x="Reserva internacional", y="Frecuencia relativa de cancelaciones") +
  ggtitle("Proporción de reservas canceladas en función de la procedencia")
```



En la gráfica podemos ver una clara diferencia en la proporción de cancelaciones entre viajeros internacionales y los que no lo son. Plantemos formalmente el contraste de hipótesis de la siguiente manera:

$$H_0 : p_1 = p_2$$

$$H_1 : p_1 < p_2$$

Donde p_1 es la proporción de cancelaciones entre reservas internacionales y p_2 la proporción de cancelaciones en reservas nacionales.

Dividimos la población entre reservas nacionales e internacionales nos quedamos con la variable que indica si la reserva ha sido cancelada.

```
international <- df$is_canceled[df$is_international]
national <- df$is_canceled[!df$is_international]
```

Realizamos el contraste de hipótesis sobre proporciones planteado.

```
prop.test(x=c(sum(international), sum(national)),
           n=c(length(international), length(national)),
           alternative = 'less', correct = FALSE, conf.level = 0.95)
```

```
##
## 2-sample test for equality of proportions without continuity
## correction
##
## data: c(sum(international), sum(national)) out of c(length(international), length(national))
## X-squared = 13298, df = 1, p-value < 2.2e-16
## alternative hypothesis: less
## 95 percent confidence interval:
## -1.0000000 -0.3262171
## sample estimates:
## prop 1   prop 2
## 0.2380523 0.5688391
```

El p-valor obtenido es muy bajo, por lo que tenemos que descartar la hipótesis nula. Por lo tanto, podemos decir que, efectivamente, la proporción de cancelaciones entre reservas internacionales es menor que entre reservas nacionales.

Análisis 3: Predicción de cancelación de reserva y factores que más influyen

Si queremos predecir la variable *is_canceled*, tomaremos esta variable como variable dependiente para nuestro modelo. Hay que tener en cuenta que, al tratarse de una variable categórica, utilizaremos un modelo de regresión logística. También se podrían utilizar otros modelos de clasificación como árboles de decisión o máquinas de vector de soporte (SVM), entre otras.

Tomaremos como variables explicativas las siguientes:

- hotel
- arrival_date_month
- is_international
- total_nights
- lead_time
- adr
- more_than_two
- with_children

Creamos el modelo de regresión y inspeccionamos los resultados.

```
model.canceled = glm(formula = is_canceled ~ hotel + arrival_date_month +
                      is_international + total_nights + lead_time + adr +
                      more_than_two + with_children,
                      family=binomial(link = "logit"), data=df)
summary(model.canceled)
```

```

## Call:
## glm(formula = is_canceled ~ hotel + arrival_date_month + is_international +
##      total_nights + lead_time + adr + more_than_two + with_children,
##      family = binomial(link = "logit"), data = df)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max 
## -2.8468 -0.8194 -0.5603  0.9183  2.5267 
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)    
## (Intercept)             -8.128e-01  3.610e-02 -22.512 < 2e-16 ***
## hotelResort Hotel      -7.699e-01  1.607e-02 -47.912 < 2e-16 ***
## arrival_date_monthApril 7.568e-02  3.905e-02   1.938  0.0526 .  
## arrival_date_monthAugust -6.073e-01  4.013e-02 -15.134 < 2e-16 *** 
## arrival_date_monthDecember -3.372e-02  4.314e-02  -0.781  0.4345  
## arrival_date_monthFebruary 8.473e-02  4.130e-02   2.052  0.0402 *  
## arrival_date_monthJuly    -6.519e-01  4.005e-02 -16.278 < 2e-16 *** 
## arrival_date_monthJune    -3.859e-01  4.020e-02  -9.601 < 2e-16 *** 
## arrival_date_monthMarch   -8.185e-02  3.998e-02  -2.047  0.0407 *  
## arrival_date_monthMay     -2.516e-01  3.951e-02  -6.366 1.94e-10 *** 
## arrival_date_monthNovember -2.650e-01  4.415e-02  -6.003 1.94e-09 *** 
## arrival_date_monthOctober -3.504e-01  3.968e-02  -8.830 < 2e-16 *** 
## arrival_date_monthSeptember -5.730e-01  4.070e-02 -14.080 < 2e-16 *** 
## is_internationalTRUE      -1.710e+00  1.475e-02 -115.979 < 2e-16 *** 
## total_nights              7.108e-02  2.985e-03  23.817 < 2e-16 *** 
## lead_time                 6.642e-03  7.643e-05  86.903 < 2e-16 *** 
## adr                       7.993e-03  1.964e-04  40.698 < 2e-16 *** 
## more_than_twoTRUE          -1.477e-01  3.123e-02  -4.728 2.27e-06 *** 
## with_childrenTRUE          -1.480e-03  3.625e-02  -0.041  0.9674 
## ---                        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 155484  on 117732  degrees of freedom
## Residual deviance: 127513  on 117714  degrees of freedom
## AIC: 127551
##
## Number of Fisher Scoring iterations: 4

```

Podemos ver como la variable categórica que indica el mes se nos ha codificado con *one hot encoding*, tomando como referencia el mes de Enero. Vemos como tenemos casi todas las variables explicativas como estadísticamente significativas con un p-valor muy bajo, a excepción de la variable “April” y “December”, que al tener un p-valor >0.05 no serán meses estadísticamente distintos al mes de Enero de referencia, y *with_children*, que tampoco es significativa.

Obtenemos los odds-ratio como la exponencial de los coeficientes de las variables explicativas obtenidos.

```
round(100*exp(coefficients(model.canceled)), 2)
```

	(Intercept)	hotelResort Hotel
##	44.36	46.30

```

##      arrival_date_monthApril      arrival_date_monthAugust
##                      107.86                  54.48
##  arrival_date_monthDecember arrival_date_monthFebruary
##                      96.68                  108.84
##      arrival_date_monthJuly      arrival_date_monthJune
##                      52.11                  67.98
##      arrival_date_monthMarch     arrival_date_monthMay
##                      92.14                  77.76
##  arrival_date_monthNovember arrival_date_monthOctober
##                      76.72                  70.44
## arrival_date_monthSeptember   is_internationalTRUE
##                      56.38                  18.08
##      total_nights              lead_time
##                      107.37                 100.67
##                      adr            more_than_twoTRUE
##                      100.80                  86.27
## with_childrenTRUE
##                      99.85

```

Los valores mayor de 1 indican un factor de riesgo, indicando un aumento de la probabilidad de ocurrencia de *is_canceled* (reserva cancelada) cuando este atributo aumenta (si es numérico) o aparece (si es binario). De esta forma, de los resultados podemos sacar algunos datos interesantes:

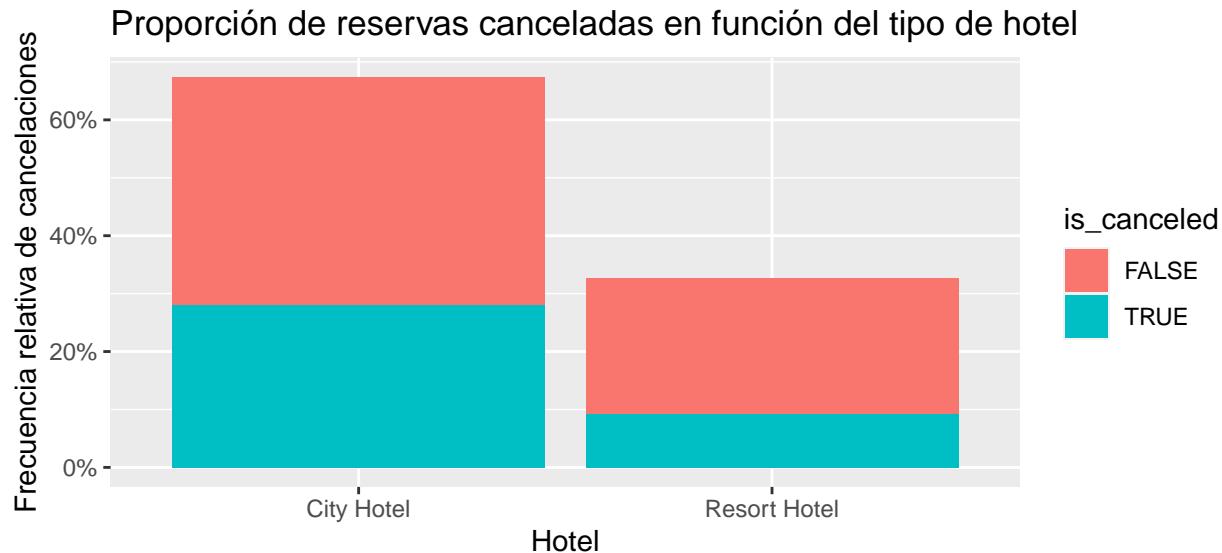
- El hotel resort tiene un 56% menos de probabilidad de ser cancelado que en hotel de ciudad, lo que nos indica que seguramente los viajes de vacaciones sean menos propensos a ser cancelados que viajes de negocios (más habituales en un hotel de ciudad).
- Los meses de julio, agosto y septiembre tienen un 48%, 46% y 44% menos de probabilidades de ser cancelados que el mes de referencia que es Enero.
- Febrero es el mes que tiene una mayor probabilidad de cancelación de reserva, un 9% más que en Enero.
- Las reservas internacionales reducen la probabilidad de cancelación de la reserva en un 86%, un valor extraordinariamente alto, y que ya hemos comprobado antes.
- Las reservas de más de dos personas reducen la probabilidad de cancelarse en un 14%.
- Por cada día adicional en la antelación de la reserva, la probabilidad de cancelación crece un 0.67%.
- Por cada Euro adicional de coste promedio de la noche de habitación, la probabilidad de cancelación crece un 0.8%.

Realizamos dos visualizaciones para comprobar la relación de las variables *hotel* y *more_than_two* con la probabilidad de cancelación.

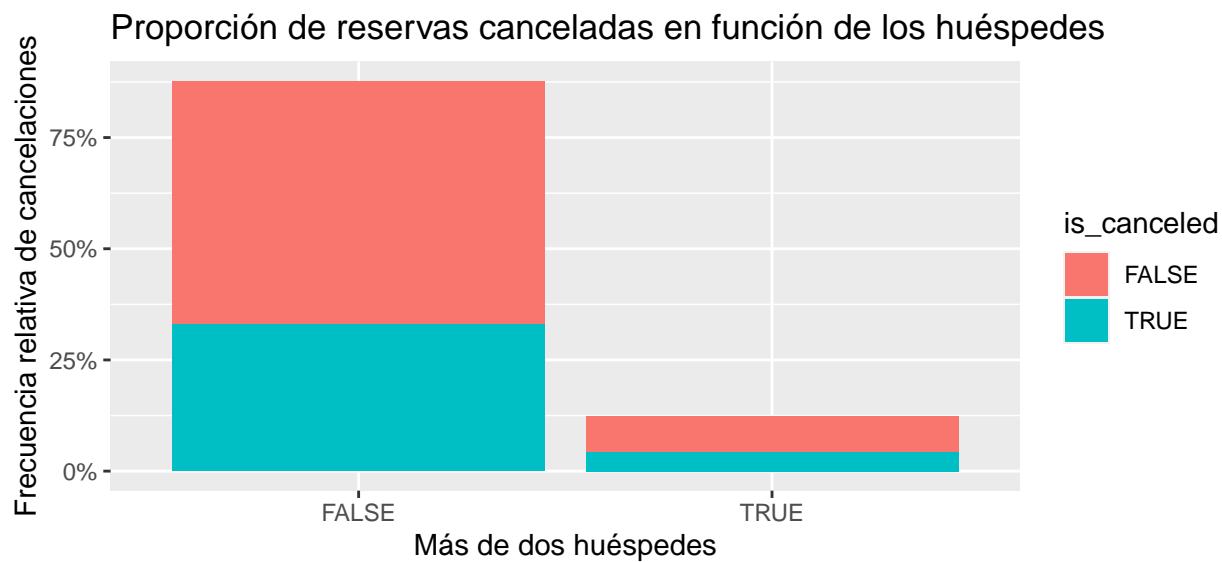
```

df %>%
  ggplot(aes(x=hotel, fill=is_canceled))+
  geom_bar(aes(y = (.count..)/sum(..count..)))+
  scale_y_continuous(labels=scales::percent) +
  labs(x="Hotel", y="Frecuencia relativa de cancelaciones")+
  ggtitle("Proporción de reservas canceladas en función del tipo de hotel")

```



```
df %>%
  ggplot(aes(x=more_than_two, fill=is_canceled))+
  geom_bar(aes(y = (.count..)/sum(..count..)))+
  scale_y_continuous(labels=scales::percent) +
  labs(x="Más de dos huéspedes", y="Frecuencia relativa de cancelaciones")+
  ggtitle("Proporción de reservas canceladas en función de los huéspedes")
```



Análisis 4: Influencia de régimen de alojamiento sobre el precio en el resort

Nos planteamos si existe influencia entre el tipo de régimen de alojamiento contratado y el precio medio de la habitación.

Realizamos en primer lugar un gráfico boxplot con el valor del coste de habitación por noche en función del régimen de alojamiento contratado. Nos centraremos en el caso del resort, que tiene disponibles los siguientes:

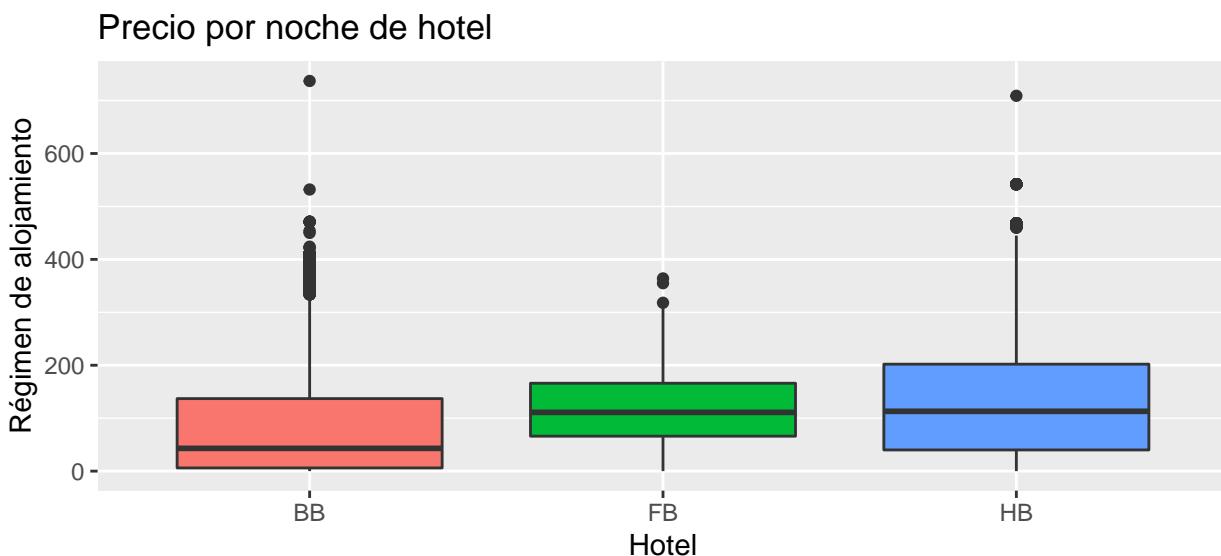
- * **BB:** Alojamiento y desayuno
- * **HB:** Media pensión
- * **FB:** Pensión completa

```

df %>%
  filter(hotel=='Resort Hotel') %>%
  filter(meal!='SC') -> df.resort

df.resort %>%
  ggplot(aes(x=meal, y=lead_time, fill=meal))+
  geom_boxplot()+
  labs(x="Hotel", y="Régimen de alojamiento")+
  ggtitle("Precio por noche de hotel")+
  theme(legend.position="none")

```



El gráfico nos indica que hay evidencias de un valor de coste de habitación distinto en función del régimen contratado. Como en este caso queremos comparar la media de *adr* en función de más de dos grupos, en concreto los tres régimenes de alojamiento, realizaremos un análisis ANOVA.

```

anova.adr <- aov(adr ~ meal, data=df.resort)
summary.aov(anova.adr)

```

```

##          Df Sum Sq Mean Sq F value Pr(>F)
## meal      2 6111973 3055987   889.8 <2e-16 ***
## Residuals 38346 131704304     3435
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Los resultados nos dan un p-valor cercano a cero, lo que indica que, efectivamente, existen diferencias significativas entre al menos dos de los tres grupos.

Hay que tener en cuenta que para dar por válidos los resultados de ANOVA se tienen que cumplir dos condiciones: * Normalidad de los residuos del modelo. * Igualdad de varianzas de los residuos.

Comparamos lo primero con un gráfico QQ.

```

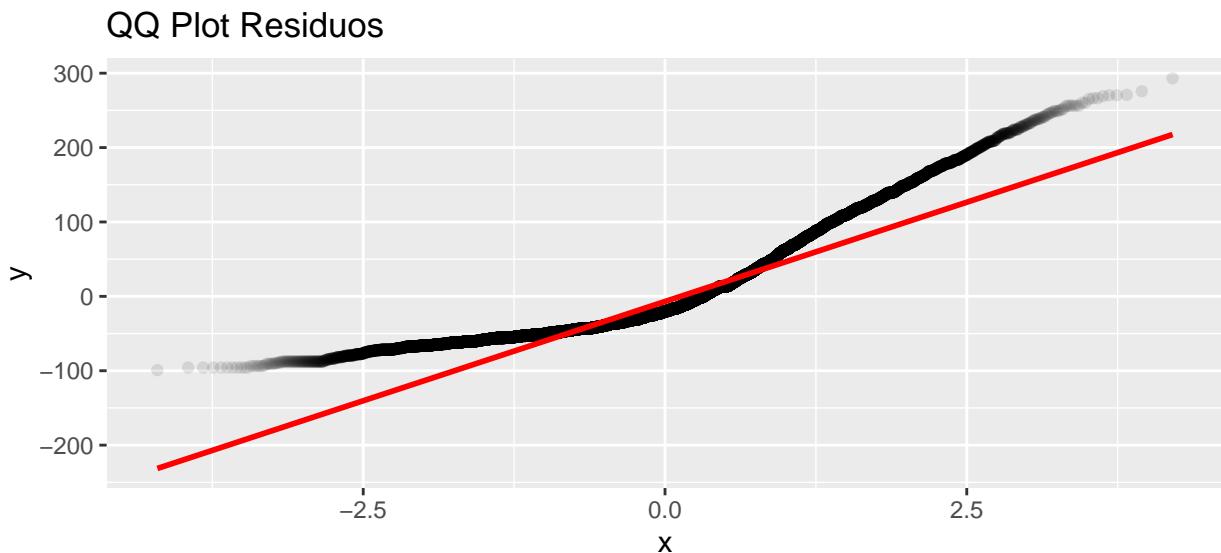
anova.fit.res = data.frame(fit.val=fitted.values(anova.adr), res=resid(anova.adr))

```

```

anova.fit.res %>%
  ggplot(aes(sample = res))+
  stat_qq(alpha=0.1)+
  stat_qq_line(color='red', size=1)+
  ggtitle("QQ Plot Residuos")

```

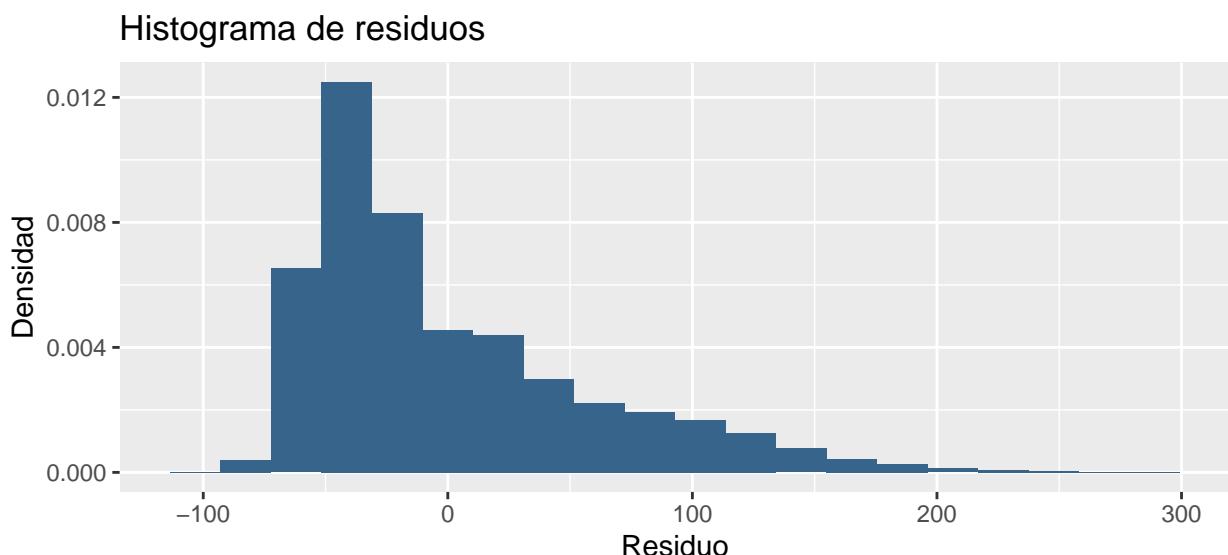


Podemos ver que los residuos no se ajustan bien a la normal, por lo que esta condición no se cumpliría. Es cierto que ANOVA es bastante resistentes a desviaciones leves de la normalidad, especialmente cuando tenemos muestras grandes, como es nuestro caso. Sin embargo, las desviaciones que vemos son bastante grandes. Realizamos un histograma para ver la distribución de estos residuos.

```

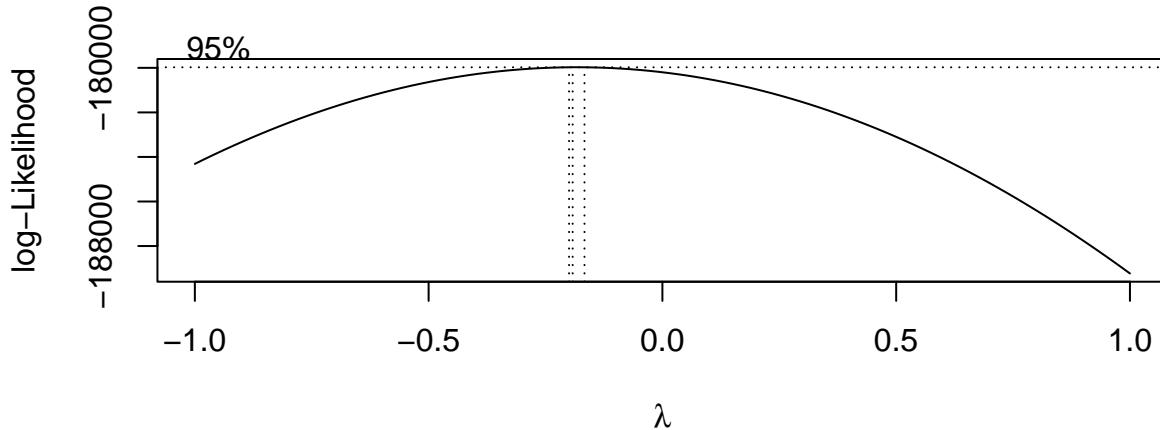
anova.fit.res %>%
  ggplot(aes(x=res))+
  geom_histogram(aes(y=..density..), bins=20, fill="steelblue4")+
  labs(x="Residuo", y="Densidad")+
  ggtitle("Histograma de residuos")

```



Claramente podemos ver que la distribución de los residuos no sigue una normal, ya que en primer lugar no es simétrica, y está claramente muy sesgada a la derecha. Para solucionar esto, podemos hacer una transformación sobre la variable destino, de forma que ésta se parezca más a una normal. Esto lo realizamos con una transformación Box-Cox, donde buscaremos el valor óptimo de λ .

```
bc<-boxcox(anova.adr, lam = seq(-1, 1, 1/10))
```



```
lambda <- bc$x[which.max(bc$y)]
lambda
```

```
## [1] -0.1919192
```

Con la λ óptima calculada, realizamos un nuevo modelo ANOVA aplicando la transformación sobre la variable dependiente.

```
anova.adr2 <- aov((adr^lambda-1)/lambda ~ meal, data=df.resort)
summary.aov(anova.adr2)
```

```
##                               Df Sum Sq Mean Sq F value Pr(>F)
## meal                  2   126.5   63.23   1107 <2e-16 ***
## Residuals      38346 2190.6     0.06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Vemos en el resultado de ANOVA que el p-valor es muy bajo, lo que nos indica que alguna de las tres clases definidas por *meal* (regímenes de alojamiento) son estadísticamente distintas.

Observamos igual que antes el gráfico QQ de los residuos y su histograma.

```
anova.fit.res2 = data.frame(fit.val=fitted.values(anova.adr2), res=resid(anova.adr2))

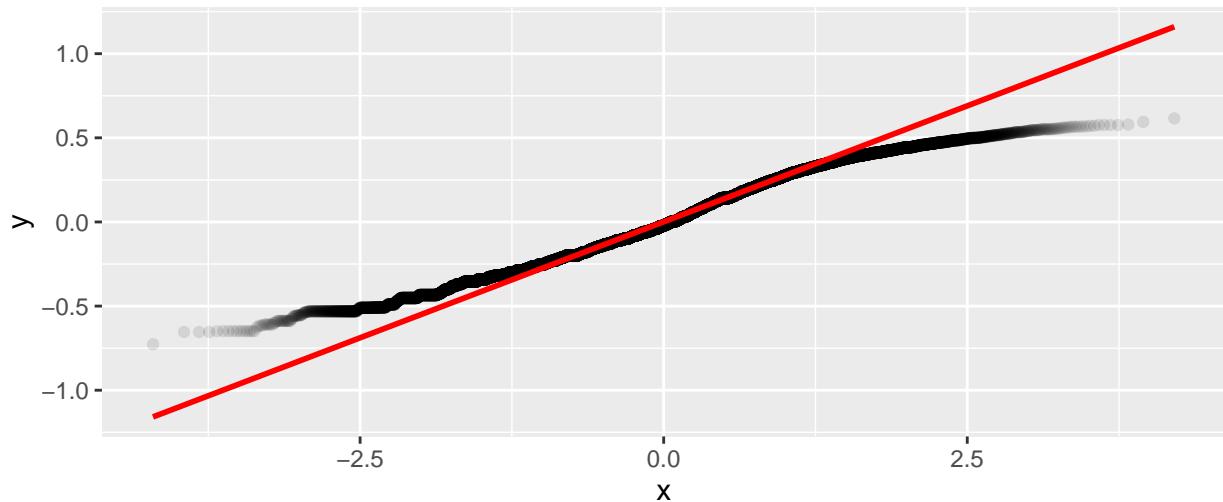
anova.fit.res2 %>%
  ggplot(aes(sample = res))+
```

```

stat_qq(alpha=0.1)+
stat_qq_line(color='red', size=1)+
ggtitle("QQ Plot Residuos")

```

QQ Plot Residuos

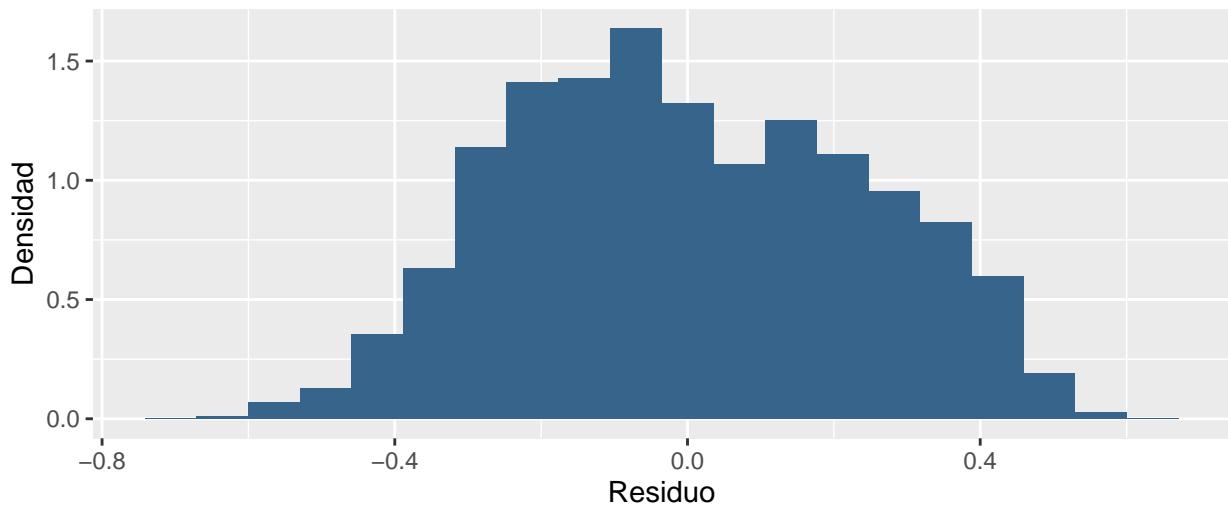


```

anova.fit.res2 %>%
  ggplot(aes(x=res))+
  geom_histogram(aes(y=..density..), bins=20, fill="steelblue4")+
  labs(x="Residuo", y="Densidad")+
  ggtitle("Histograma de residuos")

```

Histograma de residuos



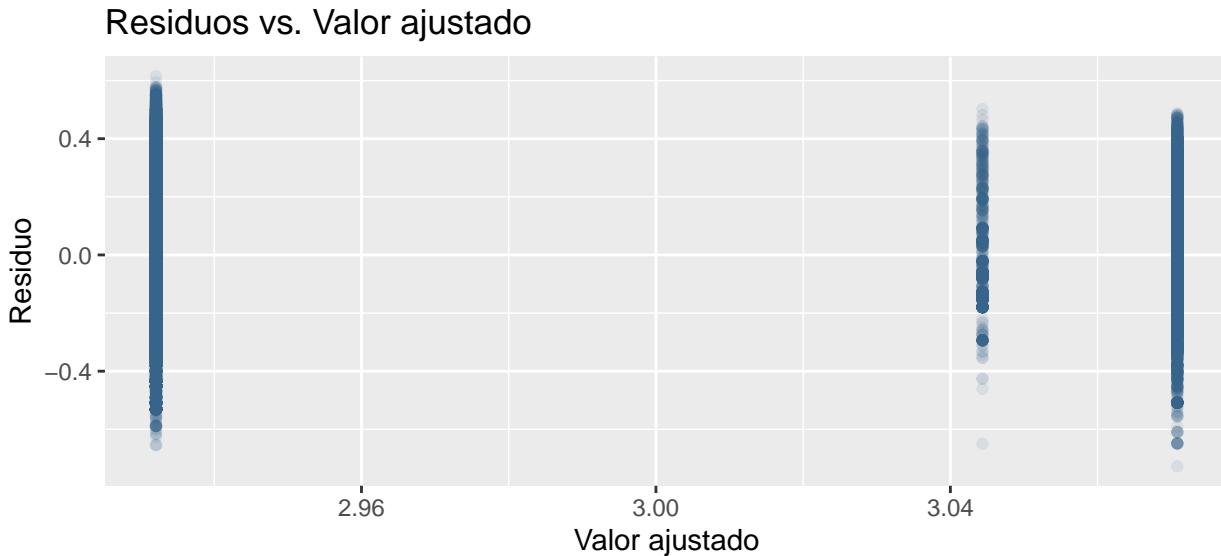
Vemos que ahora efectivamente hemos mejorado mucho la normalidad de los datos, aunque esta no es perfecta. Sin embargo, como hemos dicho antes, ANOVA es muy resistente a pequeñas desviaciones, especialmente con muestras grandes, así que ahora podemos dar por cumplida esta condición de normalidad de residuos.

Evaluamos la segunda condición, que es la homocedasticidad de los residuos, realizando un gráfico de estos.

```

anova.fit.res2 %>%
  ggplot(aes(x=fit.val, y=res))+
  geom_point(alpha=0.1, color="steelblue4")+
  labs(x="Valor ajustado", y="Residuo")+
  ggtitle("Residuos vs. Valor ajustado")

```



Vemos en la gráfica que las extensiones de las líneas de puntos son similares en las tres clases, por lo que podemos dar por buena también la condición de igualdad de varianzas.

Comprobada la idoneidad del modelo, realizamos ahora las comparaciones múltiples entre grupos teniendo en cuenta la corrección de Bonferroni, que se realiza para corregir el nivel de significación a $\frac{\alpha}{n}$, siendo n el número de contrastes de hipótesis que hay en total. Esta corrección, sin embargo, suele considerarse habitualmente como demasiado conservadora.

```
LSD.test(anova.adr2, "meal", group=T, p.adj="bonferroni", console=T)
```

```

##
## Study: anova.adr2 ~ "meal"
##
## LSD t Test for (adr^lambda - 1)/lambda
## P value adjustment method: bonferroni
##
## Mean Square Error:  0.05712722
##
## meal,  means and individual ( 95 %) CI
##
##      X.adr.lambda...1..lambda      std       r      LCL      UCL      Min      Max
## BB          2.932090 0.2470830 29578 2.929366 2.934814 2.278358 3.547501
## FB          3.044365 0.1912720    754 3.027305 3.061426 2.394762 3.546669
## HB          3.070830 0.2111465   8017 3.065598 3.076062 2.343895 3.558397
##
## Alpha: 0.05 ; DF Error: 38346
## Critical Value of t: 2.394085
##
## Groups according to probability of means differences and alpha level( 0.05 )

```

```

## 
## Treatments with the same letter are not significantly different.
## 
##      (adr^lambda - 1)/lambda groups
##   HB       3.070830    a
##   FB       3.044365    b
##   BB       2.932090    c

```

El resultado que obtenemos es que tenemos tres grupos estadísticamente distintos entre sí, correspondientes a los tres regímenes de alojamiento. Por lo tanto, los tres regímenes son estadísticamente distintos.

Representación de resultados

Las representaciones gráficas y tablas se han realizado a lo largo del documento conforme han sido necesarias.

Resolución del problema

En el presente trabajo hemos intentado realizar análisis sobre los factores que influyen sobre la cancelación de reservas de hoteles, una información de alto valor para hoteles y webs de reserva. Partiendo de un dataset con muchas variables, nos hemos centrado en algunas de ellas y simplificado otras, y hemos limpiado los datos eliminando valores vacíos y outliers.

Una vez limpios los datos, hemos comprobado como la media de precio pagado por noche no es la misma entre reservas canceladas y no canceladas. También hemos comprobado cómo la probabilidad de cancelación es mucho mayor entre las reservas nacionales que entre las internacionales. Por último, hemos realizado un modelado de regresión logística y comprobado que las reservas en meses de invierno son las más probables de ser canceladas, y las realizadas en verano las menos. Además, hemos comprobado como las cancelaciones son mucho más probables en el hotel de ciudad que en el resort. También hemos visto nuevamente como las reservas internacionales reducen mucho la probabilidad de ser cancelada, así como las reservas de más de dos personas (familias o grupos). Hemos visto por último que tanto la antelación con la que se realiza la reserva como el precio medio pagado aumentan la probabilidad de que ésta sea cancelada.

Toda esta información obtenida nos puede ayudar enormemente a optimizar campañas de marketing, segmentar clientes, y poder en última instancia predecir la probabilidad de cancelación de una determinada reserva, lo que puede ayudarnos a optimizar costes e ingresos.

Finalmente, hemos realizado un análisis ANOVA sobre el precio medio de habitación en base al régimen contratado, y comprobado como, tal y como cabría esperar, los tres regímenes ofrecidos en el hotel resort forman tres grupos estadísticamente distintos.

Exportación de datos

Exportamos los datos que hemos limpiado y sobre los que hemos realizado los análisis para tenerlos disponibles para futuros usos.

```
write.csv(df, file = "../data/hotel_bookings_clean.csv", sep=",")  
  
## Warning in write.csv(df, file = "../data/hotel_bookings_clean.csv", sep = ","):  
## attempt to set 'sep' ignored
```

Código

El código para el presente análisis se encuentra en el siguiente repositorio de GitHub:

Contribución

La contribución al trabajo de los integrantes del grupo ha sido la siguiente:

- **Investigación previa**: Javier Guimerans Alonso, Gerson Villalba Arana
 - **Redacción de las respuestas**: Javier Guimerans Alonso, Gerson Villalba Arana
 - **Desarrollo código**: Javier Guimerans Alonso, Gerson Villalba Arana
-

Bibliografía

Vegas, E. (2017). Preprocesamiento de datos. Material UOC.

<https://cxl.com/blog/outliers/>

<https://www.statisticshowto.com/box-cox-transformation/>

<https://statsandr.com/blog/anova-in-r/>

https://sites.ualberta.ca/~lkgray/uploads/7/3/6/2/7362679/slides_-_anova_assumptions.pdf

https://www.cienciadedatos.net/documentos/19_anova

<https://arc.lib.montana.edu/book/statistics-with-r-textbook/item/56>

https://raymondltremblay.github.io/ANALITICA/G10_Facet_wrap.html

<http://www.sthda.com/english/articles/40-regression-analysis/166-predict-in-r-model-predictions-and-confidence-intervals/>