Hotel_cancelations_report

Alejandro Muñoz

23/1/2022

Introducción

Proyecto

- Cancelaciones en Hoteles
- Predecir cancelación de reservas en hoteles - AM 2021

Descripción del problema

Con el fin de planear tarifas y actividades de ventas o promoción, los hoteles hacen estimaciones adelantadas de su ocupación en cada día. Una parte de estas estimaciones requiere predecir cuántas de las reservaciones que ya se tienen van a terminar en cancelaciones, lo cual libera inventario que afecta en la planeación.

Objetivo

Predecir cuáles reservaciones son probables que terminen o no en cancelación.

Fuente de datos

Los datos que se utilizaron para este proyecto fueron obtenidos del sitio Kaggle (https://www.kaggle.com/c/cancelaciones-en-hoteles/data).

Los datos originales provienen de Hotel (https://www.sciencedirect.com/science/article/pii/S2352340918315191) booking demand datasets, Antonio, de Almeida, Nunes.

Ambiente

Análisis Exploratorio de Datos

Con el fin de entender los datos realizamos una revisión general de estos (solamente de la base de datos de entrenamiento posterior a haberla dividido en entrenamiento, validación y prueba) y tratamos de identificar aquellas variables que pudieran ser interesantes para nuestro estudio. A continuación se muestra una breve parte de la exploración de datos. Si desea consultar el análisis completo puede encontrarlo en la siguiente liga EDA (https://github.com/marcoyel21/hotel_cancelation_ML21/blob/main/final/EDA_Cancelaciones.Rmd).

El data set está compuesto por las siguientes variables:

Variable	Tipo	Descripción
ADR	Numeric	Tarifa diaria promedio definida por [5]
Adults	Integer	Número de Adultos
Agent	Categorical	DNI de la agencia de viajes que realizó la reservaa
ArrivalDateDayOfMonth	Integer	Día del mes de la fecha de llegada
ArrivalDateMonth	Categorical	Mes de la fecha de llegada con 12 categorías: “enero” a “diciembre”
ArrivalDateWeekNumber	Integer	Número de semana de la fecha de llegada
ArrivalDateYear	Integer	Año de la fecha de llegada

Variable	Tipo	Descripción
AssignedRoomType	Categorical	Código del tipo de habitación asignada a la reserva. A veces, el tipo de habitación asignada difiere del tipo de habitación reservada debido a razones de operación del hotel (por ejemplo, overbooking) o por solicitud del cliente. El código se presenta en lugar de la designación por razones de anonimato
Babies	Integer	Numero de bebes
BookingChanges	Integer	Número de cambios / modificaciones realizadas a la reserva desde el momento en que se ingresó la reserva en el PMS hasta el momento del check-in o la cancelación
Children	Integer	Numero de niños
Company	Categorical	DNI de la empresa / entidad que realizó la reserva o responsable del pago de la reserva. La identificación se presenta en lugar de la designación por razones de anonimato
Country	Categorical	País de origen. Las categorías están representadas en el formato ISO 3155-3: 2013 [6]
CustomerType	Categorical	Tipo de reserva, asumiendo una de cuatro categorías:
DaysInWaitingList	Integer	Número de días que la reserva estuvo en lista de espera antes de que fuera confirmada al cliente
DepositType	Categorical	Indicación sobre si el cliente realizó un depósito para garantizar la reserva. Esta variable puede asumir tres categorías:
DistributionChannel	Categorical	Canal de distribución de reservas. El término “TA” significa “Agentes de viajes” y “TO” significa “Operadores turísticos”
IsCanceled	**Categorical**	**Valor que indica si la reserva fue cancelada (1) o no (0)**
IsRepeatedGuest	Categorical	Valor que indica si el nombre de la reserva fue de un huésped repetido (1) o no (0)
LeadTime	Integer	Número de días transcurridos entre la fecha de entrada de la reserva en el PMS y la fecha de llegada
MarketSegment	Categorical	Designación de segmento de mercado. En las categorías, el término “TA” significa “Agentes de viajes” y “TO” significa “Operadores turísticos”
Meal	Categorical	Tipo de comida reservada. Las categorías se presentan en paquetes de comidas de hospitalidad estándar:
PreviousBookingsNotCanceled	Integer	Número de reservas anteriores no canceladas por el cliente antes de la reserva actual
PreviousCancellations	Integer	Número de reservas anteriores que fueron canceladas por el cliente antes de la reserva actual
RequiredCardParkingSpaces	Integer	Número de plazas de aparcamiento requeridas por el cliente

Variable	Tipo	Descripción
ReservationStatus	Categorical	Último estado de la reserva, asumiendo una de tres categorías:
ReservationStatusDate	Date	Fecha en la que se estableció el último estado. Esta variable se puede utilizar junto con ReservationStatus para comprender cuándo se canceló la reserva o cuándo se registró el cliente en el hotel.
ReservedRoomType	Categorical	Código del tipo de habitación reservado. El código se presenta en lugar de la designación por razones de anonimato
StaysInWeekendNights	Integer	Número de noches de fin de semana (sábado o domingo) que el huésped se hospedó o reservó para alojarse en el hotel
StaysInWeekNights	Integer	Número de noches de la semana (de lunes a viernes) que el huésped se hospedó o reservó para alojarse en el hotel
TotalOfSpecialRequests	Integer	Número de solicitudes especiales realizadas por el cliente (por ejemplo, dos camas individuales o piso alto)

Nuestra variable de interés es **IsCanceled** la cual toma valores de 1 (fue cancelada) y 0 (no fue cancelada). Así que primero veamos la proporción de cancelaciones en los datos.

Cancelado	No cancelado
0.3620854	0.6379146

Usamos la función skim en la base de datos de entrenamiento para conocer las características generales de cada variable.

Data summary

Name	data
Number of rows	91531
Number of columns	30
_____	
Column type frequency:	
character	13
numeric	17
_____	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
hotel	0	1	10	12	0	2	0
is_canceled	0	1	9	12	0	2	0
arrival_date_month	0	1	3	9	0	12	0
meal	0	1	2	9	0	5	0
country	0	1	2	4	0	164	0
market_segment	0	1	6	13	0	8	0
distribution_channel	0	1	3	9	0	5	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
reserved_room_type	0	1	1	1	0	10	0
assigned_room_type	0	1	1	1	0	12	0
deposit_type	0	1	10	10	0	3	0
agent	0	1	1	4	0	302	0
company	0	1	1	4	0	329	0
customer_type	0	1	5	15	0	4	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
lead_time	0	1	96.29	105.45	0.00	15	58.0	145	737	▇▂▁▁▁
arrival_date_year	0	1	2015.90	0.61	2015.00	2016	2016.0	2016	2017	▂▁▇▁▂
arrival_date_week_number	0	1	28.18	15.01	1.00	13	31.0	41	53	▇▆▇▇▇
arrival_date_day_of_month	0	1	15.81	8.76	1.00	8	16.0	23	31	▇▇▇▇▇
stays_in_weekend_nights	0	1	0.90	1.00	0.00	0	1.0	2	19	▇▁▁▁▁
stays_in_week_nights	0	1	2.45	1.94	0.00	1	2.0	3	50	▇▁▁▁▁
adults	0	1	1.84	0.61	0.00	2	2.0	2	55	▇▁▁▁▁
children	4	1	0.09	0.37	0.00	0	0.0	0	10	▇▁▁▁▁
babies	0	1	0.01	0.10	0.00	0	0.0	0	10	▇▁▁▁▁
is_repeated_guest	0	1	0.03	0.18	0.00	0	0.0	0	1	▇▁▁▁▁
previous_cancellations	0	1	0.11	0.96	0.00	0	0.0	0	26	▇▁▁▁▁
previous_bookings_not_canceled	0	1	0.13	1.40	0.00	0	0.0	0	61	▇▁▁▁▁
booking_changes	0	1	0.21	0.64	0.00	0	0.0	0	21	▇▁▁▁▁
days_in_waiting_list	0	1	2.96	19.93	0.00	0	0.0	0	391	▇▁▁▁▁
adr	0	1	92.81	46.72	-6.38	65	86.4	114	5400	▇▁▁▁▁
required_car_parking_spaces	0	1	0.07	0.25	0.00	0	0.0	0	8	▇▁▁▁▁
total_of_special_requests	0	1	0.53	0.77	0.00	0	0.0	1	5	▇▁▁▁▁

Podemos observar que:

- Tenemos 13 variables categorías, de las cuales podemos destacar que 3 tienen un número alto de categorías (country, agent, company).
- Tenemos 17 variables numéricas.
- En este primer acercamiento, podemos identificar que las variables corresponden a:
    - Variables de tiempo: tiempo previo de reservación, fechas de llegada, duración de la reservación.
    - Características de reservación: agencia, país, canal de distribución, segmento de mercado, tipo de depósito, tarifa diaria
    - Características de los clientes y sus preferencias: adultos, bebes, tipo de hotel, tipo de habitación

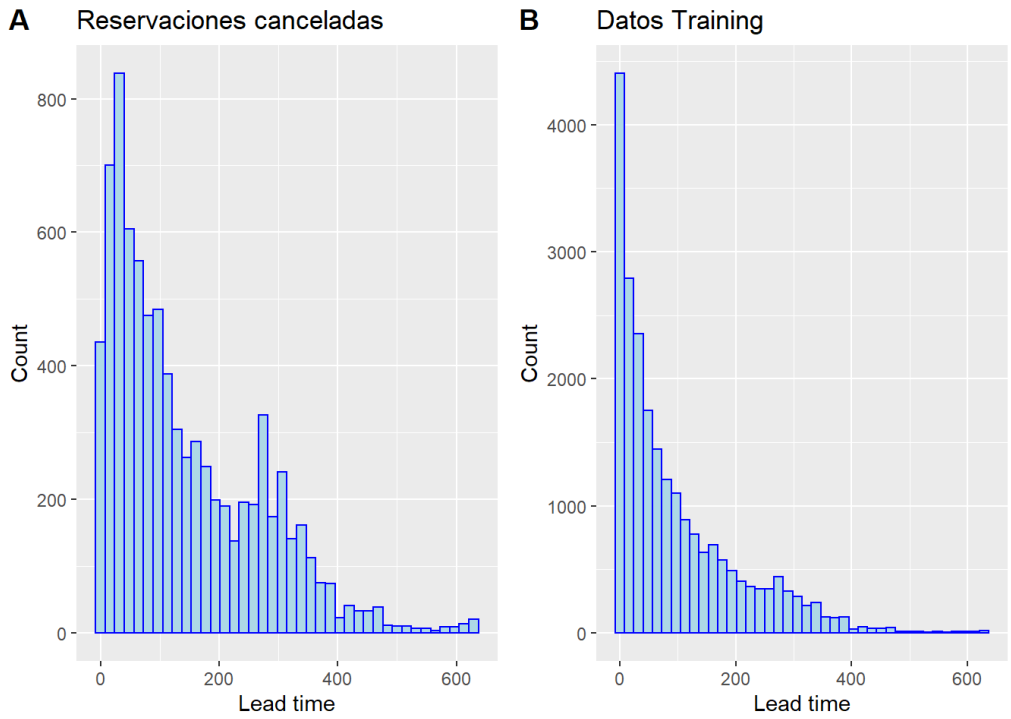
Cancelaciones EDA

Ahora extraemos el subconjunto de cancelados para hacer una revisión de todas las variables con respecto a las reservaciónes canceladas.

```
sub_cancelados <- subset(train, is_canceled == "cancelado")
```

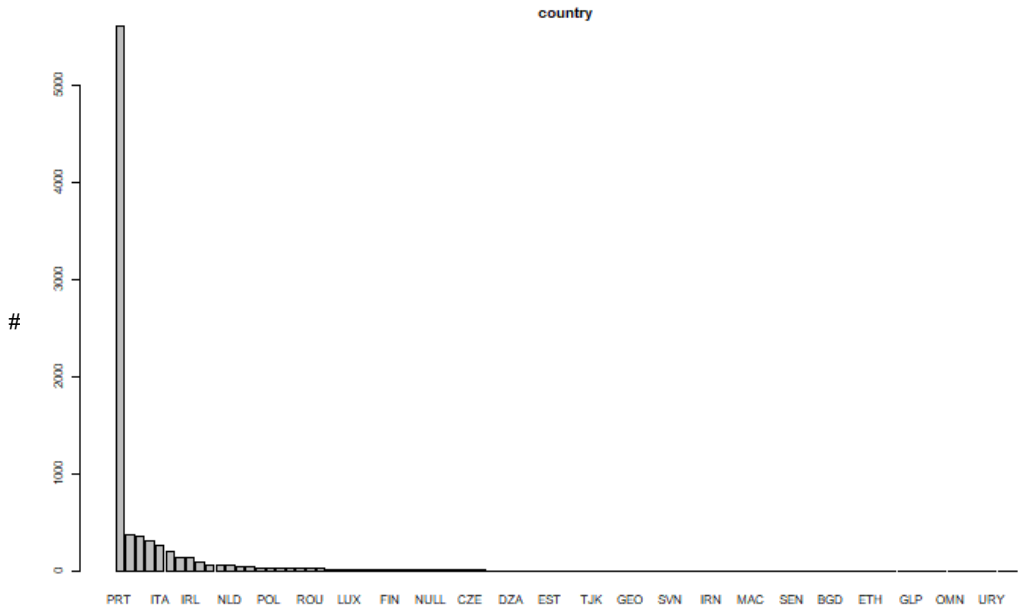
Iniciamos con la revisión de los histogramas de cada variable para ver si podemos identificar algun compartamiento interesante. A continuación se muestran los histogramas de las variables más interesantes a nuestro criterio, nuevamente puede consultar la exploración completa de los datos en EDA (https://github.com/marcoyel21/hotel_cancelation_ML21/blob/main/final/EDA_Cancelaciones.Rmd).

Lead_time: la distribución de sus datos no tiene un comportamiento lógico, porque el mayor número de cancelaciones proviene de o días previos de reservación, pero luego se mueve a valores de 90 días, 40 días y luego regresa a 2 días. será importante ver si existe algún patrón en esta variable.

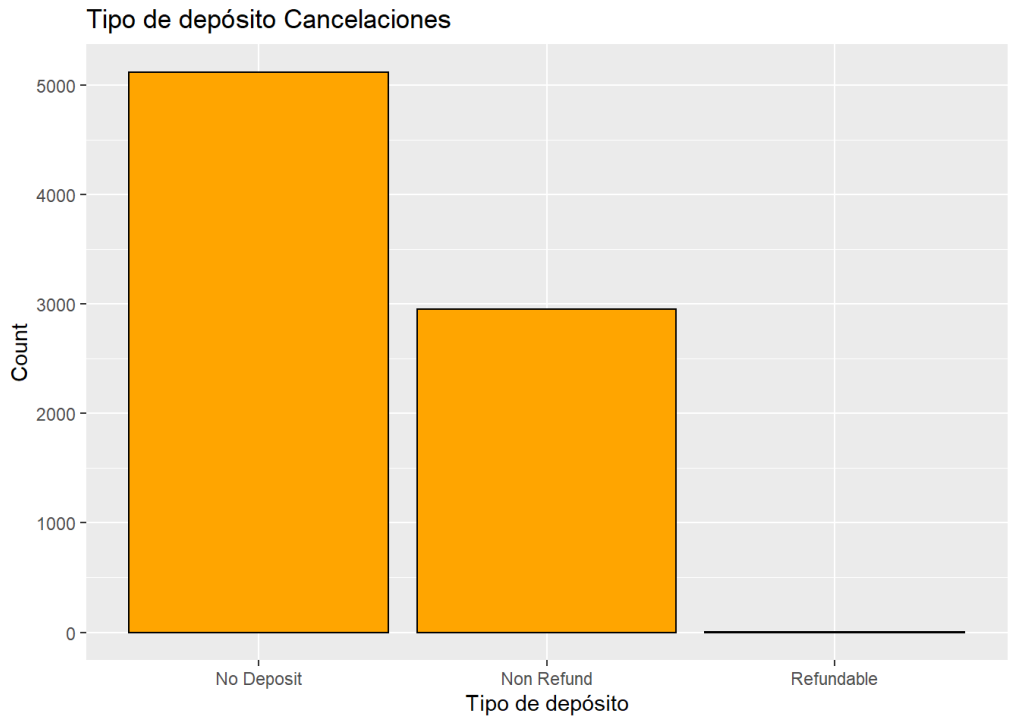


Si observamos la variable Lead_time antes de la extracción de los datos de cancelación vemos solamente que existe un sesgo, sin embargo, al graficar la misma variable seleccionando solo donde se hicieron cancelaciones podemos ver más claramente la distribución y los picos que son interesantes para nuestro análisis

Country: esta variable presenta un dato totalmente atípico en la categoría PRT por lo que es importante considerarla ya que podría explicar una porción importante de las cancelaciones.



Deposit_type: aqui hay otro caso ilógico, ya que la categoría de no rembolsable está muy por arriba de los rembolsable, uno pensaría que debería ser menos frecuenta la cancelación si no te van a devolver tu dinero. por lo que es otra variable importante.



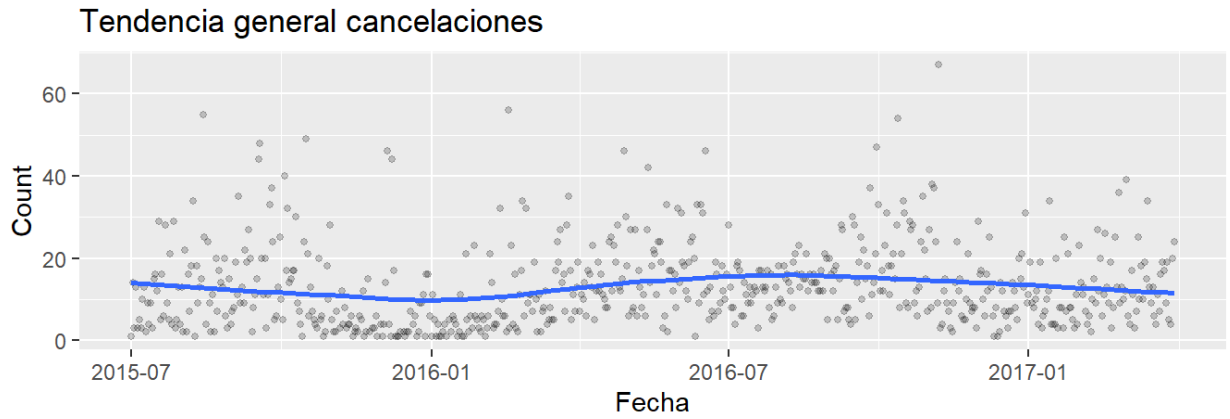
Analisando la variable **deposit_type**, se extrae el subset de deposit_type cancelados. Revisamos los porcentajes de cada categoría en las otras variables y observamos que el 97% de las cancelaciones sin rembolso pertenecen al país PRT.

```
##
##          BEL          CHE           CN          ESP          FRA          GBR
## 0.0006777364 0.0003388682 0.0013554727 0.0081328363 0.0006777364 0.0135547272
##         NULL          POL          PRT
## 0.0010166045 0.0030498136 0.9711962047
```

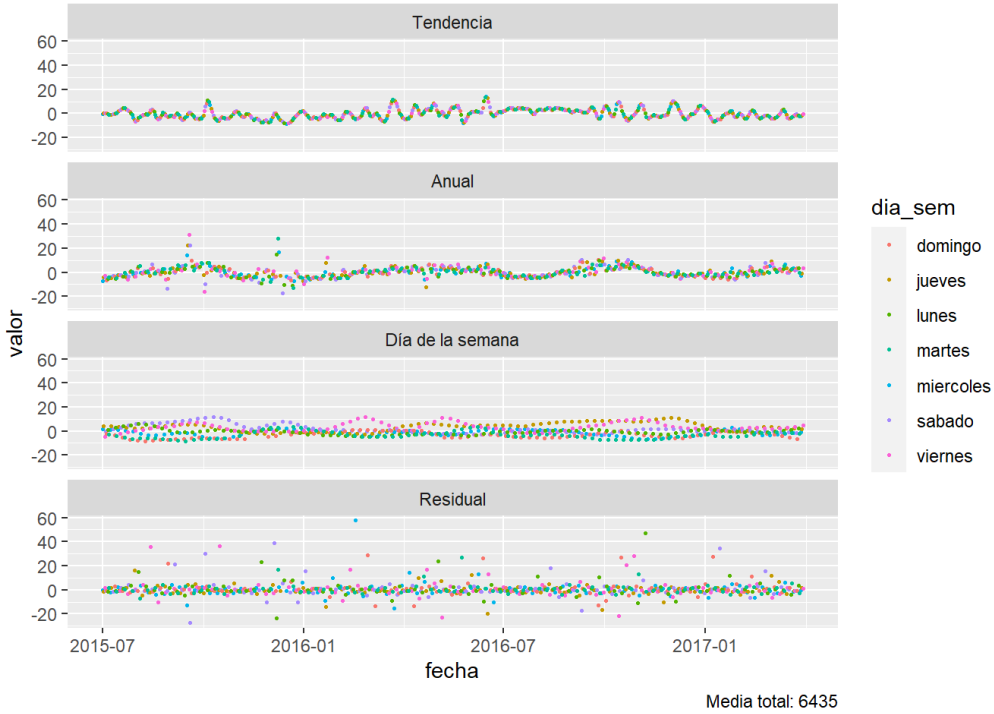
También se analizó la variable de Agente y se observa que ha una relación del agente 1 con las reservaciones del país PRT y las cancelaciones sin reembolso

Análisis de tendenciás en el tiempo EDA

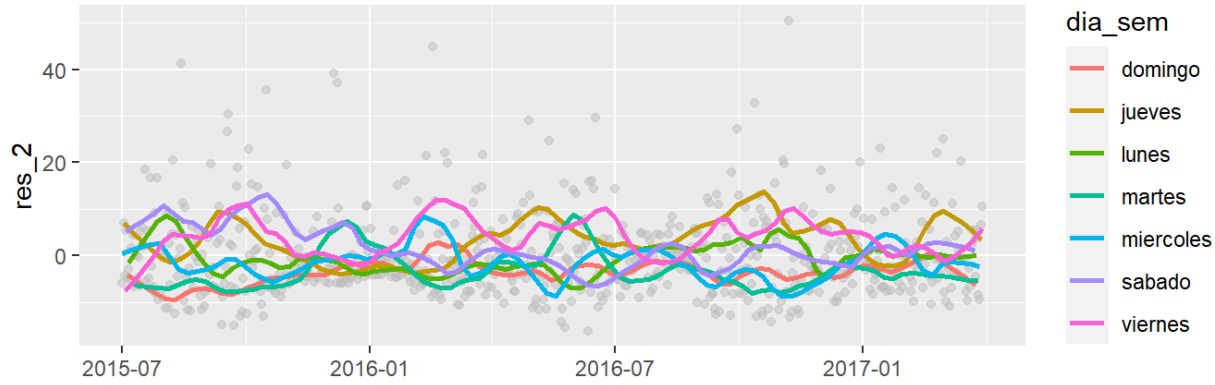
Para analizar tendencias de cancelación en el timepo se agrupan las cancelaciones por fecha.



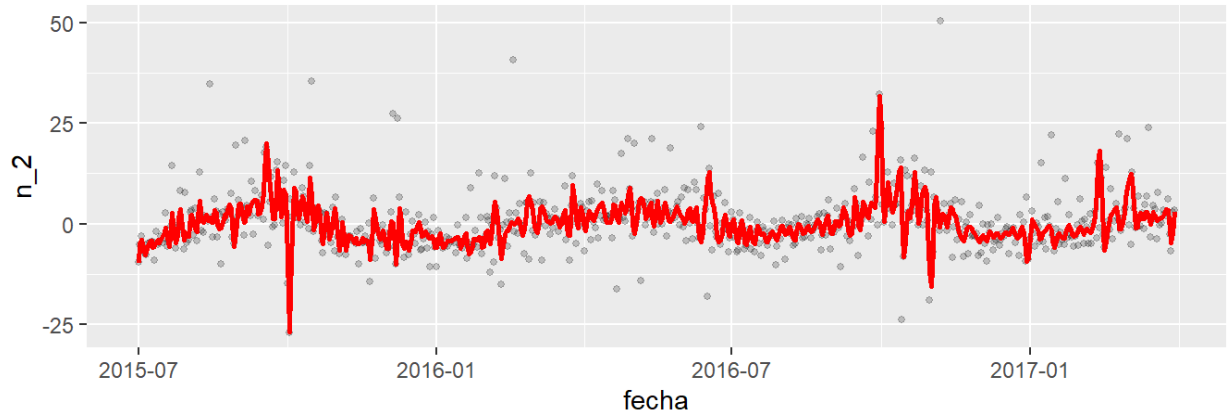
Se procede a hacer un análisis de series de tiempo (https://es.wikipedia.org/wiki/Serie_temporal).



En la gráfica de días de la semana podemos observar picos de cancelaciones los días vieres y el más significativo parece ser en el periodo de semana santa lo cual suena lógico



En la siguiente grafica anual se observar dos picos que puede corresponder a las vacaciones de verano.



Preparación de los Datos

Preprocesamiento

- Muchos datos necesitan preprocesamiento sobretodo porque están codificados como “character” en lugar de “factor”: por ejemplo, las variables:
  arrival_date_year,arrival_date_month,arrival_date_week_number,meal,country,market_segment,distribution_channel,agent,company,customer_ty
  hotel,agent_company.reserved_room_type,assigned_room_type,deposit_type.
- Otros necesitan ser números: children

Ingeniería de caracterísitcas

Para el preprocesamiento de datos se agregaron variables que pensamos serían de utilidad. Entre estas nuevas variables se encuentran:

- **lead_time**: Se cuentan los días de anticipación de la reserva y se divide en 4 grandes grupos del mismo tamaño.
- **dif_room**: Esta variable toma en cuenta si la habitación reservada es la misma que la habitación asignada.
- **singles_adults**: Indica si hay solo adultos (sin niños)
- **pascua, pascua_m1, …., pascua_m6** : indica si tal fecha era Pascua.
- **mag_tasa_can**: Proporciona el ratio entre el total de cancelaciones respecto al total de reservaciones.

** **COMBINACIONES aleatorias**: Incorporamos estas variables de combinaciones al azar buscando interaciones que ayudaran al modelo.* ### Combinaciónes

Asimismo exploramos distintas combinaciones pensando en que los modelos que ibamos a usar tenían la capacidad de seleccionar automáticamente las caracterísitcas más útiles.

- **dias_semana**: Interaccion entre el día de reservación y el número de semana.
- **Agent_company**: La combinación de agent y company.Esta resulto muy util en los casos donde ambas variables tenían valor NULL.
- **dif_room**: Si el cuarto asignado es diferente al cuarto reservado.
- **week_day_sem**: Combinación de día de la semana y número de semana.
- **week_daymonth**: Combinación de día de la semana y número de semana.
- **Tasa de rechazo**: Proporcion de reservaciones canceladas del total de reservaciones registradas.
- **market_dist**: Combinación de market_segment y distribution_chanel.
- **cust_deposti**:Combinación de customer_type y deposit_type.
- **cust_segment**: Combinación de customer_type y market_segment.
- **lead_deposit**: Combinación de lead y del tipo de deposito.
- **lead_week**: Combinación de lead y número de semana de la reserva.
- **meal_reserv**: Combinación de tipo de alimento y tipo de reserva.
- **country_month**: Combinación del mes de la reserva y el país de origen.

CV

Ahora sobre el conjunto de entrenamiento guardaremos un cacho para probar.

```
# proporción que queremos de training
training_size <- 0.8
# filas de training
training_rows <- sample(seq_len(nrow(newdata_train)),
                        size=floor(training_size*nrow(newdata_train)))
#training set
data_training <- newdata_train[training_rows,]
#training cuenta con la y


#validation set
# la variable objetivo por separado
data_validation <- newdata_train[-training_rows,-1] #sin la y
y <- newdata_train[-training_rows,1]
```

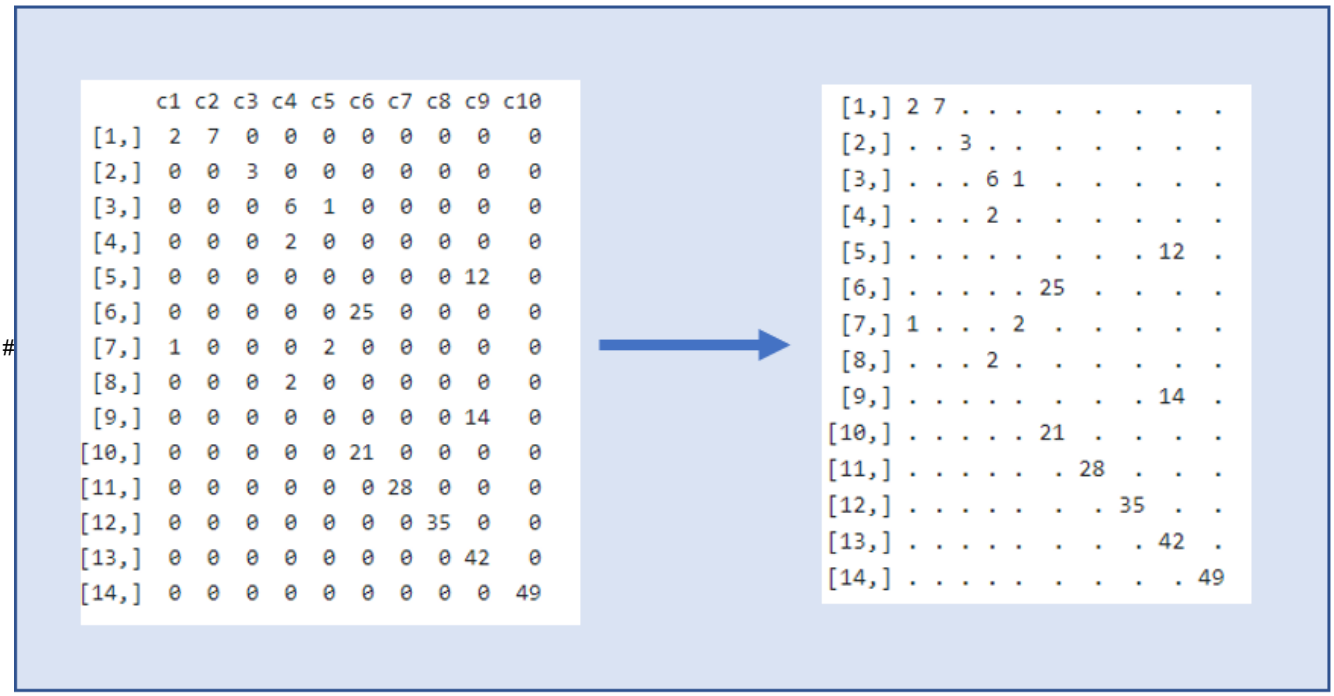
Nivelación de variables

Antes de realizar la conversión a matrices ralas necesitamos indicarle a la computadora que las bases de datos cuentan con los mismas variables y dentro de cada variable categórica, los mismos niveles. Esto debido a que al hacer el CV, es muy probable que no todas las variables conserven la misma cantidad de niveles que la base completa antes del CV. Para ello creamos la siguiente función y la aplicamos a las bases de datos.

```
# creo una funcion para que las bases de datos cuenten con los mismos "levels"
# este paso es crucial para asegurarnos que traning, set y el modelo hablen "el mismo idioma", es decir que tengan las misma
s variables
equallevels <- function(x, y) {
    if (is.data.frame(x) & is.data.frame(y)) {
        com <- intersect(x = names(x), y = names(y))
        for (i in com) {
            if (!is.null(levels(y[[i]]))) {
                x[[i]] <- factor(x[[i]], levels = levels(y[[i]]))
            }
        }
        return(x)
    } else {
        stop("`x` and `y` must be a data.frame.")
    }
}
```

Matrices RALAS

Para el procesamiento de los datos previo al modelaje se hizo one hot encoding (https://www.educative.io/blog/one-hot-encoding), el cuál consiste en transformar las variables categóricas en variables dummy. Cómo ya se mencionó en el EDA, existen variables con muchísimas categorías (country, agent, company). Lo cual nos deja con un data frame lleno de muchos ceros. Para manejar este “data frame” o “matriz” con muchos ceros se hizo uso de las matrices Ralas (http://amunategui.github.io/sparse-matrix-glmnet/) las cuales concervan únicamente las entradas con valores distintos de cero. Para ello se utilizó la función **sparse.model.matrix** de la librería Matrix (https://cran.rproject.org/web/packages/Matrix/index.html). La implementación del código completa la puede ver en la siguiente liga Model (https://github.com/marcoyel21/hotel_cancelation_ML21/blob/main/final/modelo_final%20.Rmd).




```
#Matriz de covariates
#data_training<-sample_train
Xa <-data_training %>% select(-1) #training menos y
Xb <-data_validation
Xc <-equallevels(newdata_test,Xa)

#para manejo de nas, si lo quito, por alguna razon la conversion a matriz rala me quita unas obs

options(na.action='na.pass')
```

Ahora creo 3 matrices ralas para entrenamiento, validación y prueba.

```
#se quita intercepto
#se ponen todas las columnas
Xa <- sparse.model.matrix(~.+0, data = Xa)
Xb <- sparse.model.matrix(~.+0, data = Xb)
Xc <- sparse.model.matrix(~.+0, data = Xc)

#vector de Y´s
Ya<-data_training$y
```

Ahora tengo 3 matrices con una alta cantidad de variables(4,347) (debido al one hot encoding y a la nivelación) para cada dataset del CV. Esto pensando en el feature seleccion que los modelos pueden hacer. Ahora puedo aplicarles cualquier modelo de manera muy ordenada y simple.

Modeling

En esta parte aplicaremos dos modelos: un Lasso-Logit y un XGboosting.

Cross-Validated LASSO-logit

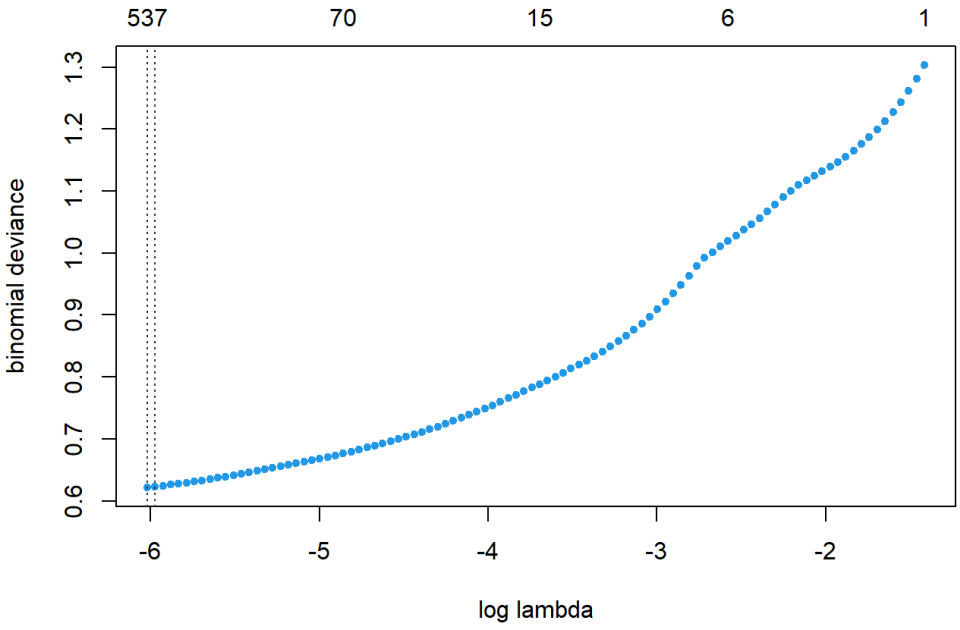
Seestima un cross validated LASSO y se muestra el la gráfica de CV Binomial Deviance vs Complejidad

```
#CV LASSO
# se hacen 5 folds
cvlasso_a<-cv.gamlr(x = Xa, y = Ya, verb = T, family = 'binomial', nfold = 5)
```

```
## Warning in gamlr(x, y, ...): numerically perfect fit for some observations.
```

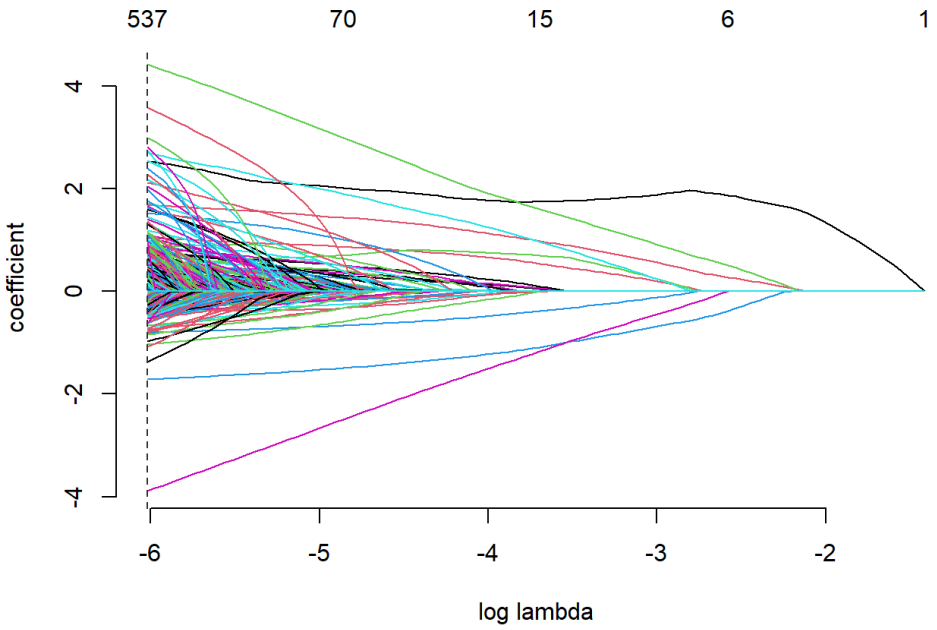
```
## fold 1,2,3,4,5,done.
```

```
#Grafica
plot(cvlasso_a)
```



Grafica Lasso de los coeficientes vs la complejidad del modelo.

```
plot(cvlasso_a$gamlr)
```



Hiper parametro

Automaticamente se elige el lambda que minimiza la devianza OOS.

```
# Identificador para el lambda deseado
# Valor del lambda deseado
#lambda resultante
a_lambda<- colnames(coef(cvlasso_a, select="min"))
cvlasso_a$gamlr$lambda[a_lambda]
```

```
##      seg100
## 0.002430558
```

Variables

A continuacion una tabla con los coeficientes que se selecciona para el CV LASSO. Que sorprendentemente solo fueron 561.

```
coefs<-coef(cvlasso_a, select="min", k=2, corrected=TRUE)
coefs<-as.data.frame(coefs[,1])
names(coefs)<-"valor"
coefs<-coefs %>% filter(valor !=0)
modelvariables<-row.names(coefs)
modelvariables
```

```
##   [1] "intercept"                       "lead_time"
##   [3] "arrival_date_year2015"           "arrival_date_year2017"
##   [5] "arrival_date_monthDecember"      "arrival_date_monthJune"
##   [7] "arrival_date_day_of_month21"     "arrival_date_day_of_month30"
##   [9] "stays_in_weekend_nights"         "stays_in_week_nights"
##  [11] "adults"                          "mealHB"
##  [13] "mealUndefined"                   "countryAGO"
##  [15] "countryARE"                      "countryAUT"
##  [17] "countryBEL"                      "countryBGD"
##  [19] "countryBRA"                      "countryCHE"
##  [21] "countryCHN"                      "countryCPV"
##  [23] "countryCYP"                      "countryDEU"
##  [25] "countryDNK"                      "countryESP"
##  [27] "countryFIN"                      "countryFRA"
##  [29] "countryGBR"                      "countryGEO"
##  [31] "countryGLP"                      "countryHKG"
##  [33] "countryHRV"                      "countryIDN"
##  [35] "countryIRL"                      "countryITA"
##  [37] "countryJEY"                      "countryJPN"
##  [39] "countryKOR"                      "countryLTU"
##  [41] "countryMAC"                      "countryMAR"
##  [43] "countryNGA"                      "countryNLD"
##  [45] "countryPAK"                      "countryPAN"
##  [47] "countryPOL"                      "countryPRT"
##  [49] "countryQAT"                      "countryRUS"
##  [51] "countrySAU"                      "countrySEN"
##  [53] "countrySRB"                      "countryTJK"
##  [55] "countryTUR"                      "countryVEN"
##  [57] "countryZAF"                      "market_segment8"
##  [59] "distribution_channel5"           "is_repeated_guest"
##  [61] "previous_bookings_not_canceled" "reserved_room_typeE"
##  [63] "reserved_room_typeP"             "assigned_room_typeB"
##  [65] "assigned_room_typeC"             "assigned_room_typeI"
##  [67] "assigned_room_typeP"             "booking_changes"
##  [69] "deposit_typeB"                   "agent107"
##  [71] "agent11"                         "agent110"
##  [73] "agent118"                        "agent13"
##  [75] "agent132"                        "agent134"
##  [77] "agent14"                         "agent141"
##  [79] "agent151"                        "agent152"
##  [81] "agent155"                        "agent157"
##  [83] "agent168"                        "agent17"
##  [85] "agent191"                        "agent201"
##  [87] "agent214"                        "agent215"
##  [89] "agent22"                         "agent220"
##  [91] "agent23"                         "agent240"
##  [93] "agent241"                        "agent242"
##  [95] "agent243"                        "agent254"
##  [97] "agent26"                         "agent262"
##  [99] "agent27"                         "agent281"
## [101] "agent288"                        "agent291"
## [103] "agent308"                        "agent314"
## [105] "agent315"                        "agent32"
## [107] "agent332"                        "agent341"
## [109] "agent38"                         "agent390"
## [111] "agent40"                         "agent410"
## [113] "agent440"                        "agent56"
## [115] "agent63"                         "agent69"
## [117] "agent7"                          "agent75"
## [119] "agent8"                          "agent89"
## [121] "agent9"                          "agent94"
## [123] "company102"                      "company110"
## [125] "company153"                      "company204"
## [127] "company218"                      "company242"
## [129] "company270"                      "company277"
## [131] "company280"                      "company307"
## [133] "company309"                      "company31"
## [135] "company321"                      "company373"
## [137] "company38"                       "company39"
```

```
## [139] "company40"                     "company416"
## [141] "company457"                    "company478"
## [143] "company486"                    "company504"
## [145] "company51"                     "company513"
## [147] "company68"                     "company94"
## [149] "companyNULL"                   "customer_typeTransient"
## [151] "required_car_parking_spaces"   "total_of_special_requests"
## [153] "dia_semviernes"                "agent_company107_NULL"
## [155] "agent_company11_NULL"          "agent_company110_NULL"
## [157] "agent_company118_NULL"         "agent_company13_NULL"
## [159] "agent_company134_NULL"         "agent_company14_NULL"
## [161] "agent_company155_NULL"         "agent_company17_NULL"
## [163] "agent_company191_NULL"         "agent_company214_NULL"
## [165] "agent_company240_NULL"         "agent_company242_NULL"
## [167] "agent_company250_NULL"         "agent_company254_NULL"
## [169] "agent_company262_NULL"         "agent_company281_NULL"
## [171] "agent_company291_NULL"         "agent_company315_NULL"
## [173] "agent_company332_NULL"         "agent_company341_NULL"
## [175] "agent_company38_NULL"          "agent_company390_NULL"
## [177] "agent_company410_NULL"         "agent_company440_NULL"
## [179] "agent_company56_NULL"          "agent_company8_NULL"
## [181] "agent_company9_NULL"           "agent_company94_NULL"
## [183] "agent_companyNULL_102"         "agent_companyNULL_110"
## [185] "agent_companyNULL_153"         "agent_companyNULL_204"
## [187] "agent_companyNULL_218"         "agent_companyNULL_270"
## [189] "agent_companyNULL_277"         "agent_companyNULL_280"
## [191] "agent_companyNULL_281"         "agent_companyNULL_309"
## [193] "agent_companyNULL_31"          "agent_companyNULL_321"
## [195] "agent_companyNULL_373"         "agent_companyNULL_38"
## [197] "agent_companyNULL_416"         "agent_companyNULL_457"
## [199] "agent_companyNULL_478"         "agent_companyNULL_486"
## [201] "agent_companyNULL_513"         "agent_companyNULL_68"
## [203] "agent_companyNULL_94"          "singles_adults"
## [205] "dif_room"                      "weekmonthFebruary_9"
## [207] "weekmonthJune_27"              "weekmonthSeptember_40"
## [209] "daymontApril_20"               "daymontApril_29"
## [211] "daymontApril_30"               "daymontApril_4"
## [213] "daymontApril_5"                "daymontApril_6"
## [215] "daymontAugust_17"              "daymontAugust_21"
## [217] "daymontAugust_27"              "daymontDecember_16"
## [219] "daymontDecember_5"             "daymontDecember_6"
## [221] "daymontDecember_7"             "daymontFebruary_27"
## [223] "daymontFebruary_8"             "daymontJuly_1"
## [225] "daymontJuly_10"                "daymontJuly_16"
## [227] "daymontJuly_2"                 "daymontJuly_22"
## [229] "daymontJuly_23"                "daymontJuly_5"
## [231] "daymontJuly_7"                 "daymontJune_10"
## [233] "daymontJune_21"                "daymontJune_26"
## [235] "daymontJune_8"                 "daymontMarch_10"
## [237] "daymontMarch_29"               "daymontMay_1"
## [239] "daymontMay_15"                 "daymontMay_22"
## [241] "daymontMay_26"                 "daymontMay_27"
## [243] "daymontMay_3"                  "daymontNovember_12"
## [245] "daymontNovember_23"            "daymontOctober_12"
## [247] "daymontOctober_13"             "daymontOctober_14"
## [249] "daymontOctober_22"             "daymontOctober_25"
## [251] "daymontOctober_26"             "daymontOctober_27"
## [253] "daymontOctober_8"              "daymontSeptember_1"
## [255] "daymontSeptember_13"           "daymontSeptember_29"
## [257] "weekdaymonthApril_17_20"       "weekdaymonthApril_18_29"
## [259] "weekdaymonthApril_18_30"       "weekdaymonthAugust_33_14"
## [261] "weekdaymonthAugust_34_17"      "weekdaymonthAugust_35_27"
## [263] "weekdaymonthAugust_35_29"      "weekdaymonthDecember_49_5"
## [265] "weekdaymonthDecember_50_4"     "weekdaymonthDecember_50_6"
## [267] "weekdaymonthDecember_50_7"     "weekdaymonthDecember_51_16"
## [269] "weekdaymonthDecember_53_26"    "weekdaymonthFebruary_10_28"
## [271] "weekdaymonthFebruary_8_24"     "weekdaymonthJanuary_1_6"
## [273] "weekdaymonthJuly_27_2"         "weekdaymonthJuly_28_11"
## [275] "weekdaymonthJuly_28_7"         "weekdaymonthJuly_29_16"
```

```
## [277] "weekdaymonthJuly_29_17"         "weekdaymonthJuly_30_22"
## [279] "weekdaymonthJune_24_8"          "weekdaymonthMarch_11_10"
## [281] "weekdaymonthMarch_11_14"        "weekdaymonthMarch_11_8"
## [283] "weekdaymonthMarch_9_1"          "weekdaymonthMay_19_1"
## [285] "weekdaymonthMay_19_3"           "weekdaymonthMay_21_15"
## [287] "weekdaymonthMay_22_26"          "weekdaymonthMay_22_27"
## [289] "weekdaymonthNovember_46_12"     "weekdaymonthNovember_48_20"
## [291] "weekdaymonthNovember_48_23"     "weekdaymonthNovember_48_27"
## [293] "weekdaymonthNovember_49_27"     "weekdaymonthOctober_40_2"
## [295] "weekdaymonthOctober_41_8"       "weekdaymonthOctober_42_12"
## [297] "weekdaymonthOctober_42_13"      "weekdaymonthOctober_42_14"
## [299] "weekdaymonthOctober_42_17"      "weekdaymonthOctober_43_17"
## [301] "weekdaymonthOctober_43_22"      "weekdaymonthOctober_43_23"
## [303] "weekdaymonthOctober_44_25"      "weekdaymonthOctober_44_26"
## [305] "weekdaymonthOctober_44_27"      "weekdaymonthOctober_45_30"
## [307] "weekdaymonthSeptember_36_5"     "weekdaymonthSeptember_37_4"
## [309] "weekdaymonthSeptember_38_13"    "weekdaymonthSeptember_40_29"
## [311] "month_diasemApril_lunes"        "month_diasemAugust_domingo"
## [313] "month_diasemJuly_miercoles"     "month_diasemJune_martes"
## [315] "month_diasemMarch_domingo"      "month_diasemMay_jueves"
## [317] "month_diasemMay_viernes"        "month_diasemNovember_domingo"
## [319] "month_diasemOctober_jueves"     "month_diasemOctober_sabado"
## [321] "week_diasem1_sabado"            "week_diasem10_domingo"
## [323] "week_diasem12_lunes"            "week_diasem15_lunes"
## [325] "week_diasem15_martes"           "week_diasem15_miercoles"
## [327] "week_diasem19_martes"           "week_diasem22_domingo"
## [329] "week_diasem24_viernes"          "week_diasem27_domingo"
## [331] "week_diasem27_miercoles"        "week_diasem28_sabado"
## [333] "week_diasem29_sabado"           "week_diasem29_viernes"
## [335] "week_diasem31_lunes"            "week_diasem33_sabado"
## [337] "week_diasem35_viernes"          "week_diasem36_jueves"
## [339] "week_diasem37_jueves"           "week_diasem38_martes"
## [341] "week_diasem39_sabado"           "week_diasem40_sabado"
## [343] "week_diasem41_martes"           "week_diasem44_domingo"
## [345] "week_diasem45_sabado"           "week_diasem46_jueves"
## [347] "week_diasem47_miercoles"        "tasa_canc"
## [349] "market_dist2_1"                 "market_dist3_TA_TO"
## [351] "market_dist5_2"                 "market_distOfflineTA_TO_TA_TO"
## [353] "cust_depostiTransient_B"        "cust_segmentContract_3"
## [355] "cust_segmentContract_7"         "cust_segmentTransient-Party_1"
## [357] "cust_segmentTransient-Party_7"  "cust_segmentTransient_7"
## [359] "lead_depositA_[ 16, 59)"        "lead_depositA_[ 59,146)"
## [361] "lead_depositA_[146,737]"        "lead_depositB_[ 59,146)"
## [363] "lead_week1_[146,737]"           "lead_week10_[146,737]"
## [365] "lead_week12_[ 59,146)"          "lead_week13_[  0, 16)"
## [367] "lead_week18_[ 16, 59)"          "lead_week18_[ 59,146)"
## [369] "lead_week2_[  0, 16)"           "lead_week2_[ 59,146)"
## [371] "lead_week2_[146,737]"           "lead_week22_[  0, 16)"
## [373] "lead_week22_[146,737]"          "lead_week23_[  0, 16)"
## [375] "lead_week24_[ 16, 59)"          "lead_week29_[  0, 16)"
## [377] "lead_week29_[146,737]"          "lead_week3_[  0, 16)"
## [379] "lead_week3_[146,737]"           "lead_week32_[  0, 16)"
## [381] "lead_week32_[ 16, 59)"          "lead_week32_[ 59,146)"
## [383] "lead_week33_[ 59,146)"          "lead_week35_[ 16, 59)"
## [385] "lead_week36_[ 59,146)"          "lead_week4_[146,737]"
## [387] "lead_week40_[ 59,146)"          "lead_week42_[  0, 16)"
## [389] "lead_week42_[ 16, 59)"          "lead_week42_[ 59,146)"
## [391] "lead_week43_[ 59,146)"          "lead_week44_[  0, 16)"
## [393] "lead_week44_[146,737]"          "lead_week45_[ 59,146)"
## [395] "lead_week47_[  0, 16)"          "lead_week48_[  0, 16)"
## [397] "lead_week48_[ 59,146)"          "lead_week49_[ 59,146)"
## [399] "lead_week5_[  0, 16)"           "lead_week5_[ 59,146)"
## [401] "lead_week50_[  0, 16)"          "lead_week50_[ 59,146)"
## [403] "lead_week51_[146,737]"          "lead_week52_[ 59,146)"
## [405] "lead_week52_[146,737]"          "lead_week53_[146,737]"
## [407] "lead_week6_[  0, 16)"           "lead_week6_[ 59,146)"
## [409] "lead_week6_[146,737]"           "lead_week7_[ 59,146)"
## [411] "lead_week8_[  0, 16)"           "lead_week9_[  0, 16)"
## [413] "meal_reservFB_A"                "meal_reservHB_G"
```

```
## [415] "meal_reservSC_A"               "meal_reservSC_F"
## [417] "meal_reservSC_G"               "meal_reservSC_P"
## [419] "meal_reservUndefined_D"        "country_monthAGO_April"
## [421] "country_monthAGO_December"     "country_monthAGO_February"
## [423] "country_monthAND_January"      "country_monthARM_May"
## [425] "country_monthAUS_April"        "country_monthAUS_February"
## [427] "country_monthAUS_July"         "country_monthAUT_February"
## [429] "country_monthAUT_July"         "country_monthAUT_May"
## [431] "country_monthAUT_October"      "country_monthAZE_March"
## [433] "country_monthBEL_August"       "country_monthBEL_July"
## [435] "country_monthBEL_October"      "country_monthBGR_May"
## [437] "country_monthCHE_July"         "country_monthCHE_March"
## [439] "country_monthCHE_October"      "country_monthCHL_April"
## [441] "country_monthCHL_December"     "country_monthCHN_January"
## [443] "country_monthCHN_July"         "country_monthCHN_May"
## [445] "country_monthCHN_October"      "country_monthCOL_November"
## [447] "country_monthCOL_September"    "country_monthCYP_August"
## [449] "country_monthCYP_May"          "country_monthCZE_August"
## [451] "country_monthDEU_April"        "country_monthDEU_December"
## [453] "country_monthDEU_March"        "country_monthDEU_October"
## [455] "country_monthECU_December"     "country_monthEGY_February"
## [457] "country_monthEGY_November"     "country_monthESP_April"
## [459] "country_monthESP_December"     "country_monthESP_June"
## [461] "country_monthFRA_April"        "country_monthFRA_June"
## [463] "country_monthFRA_March"        "country_monthFRA_May"
## [465] "country_monthFRA_November"     "country_monthGBR_August"
## [467] "country_monthGBR_June"         "country_monthGBR_November"
## [469] "country_monthGBR_October"      "country_monthGEO_March"
## [471] "country_monthGIB_August"       "country_monthGIB_March"
## [473] "country_monthGNB_February"     "country_monthGRC_March"
## [475] "country_monthHUN_April"        "country_monthHUN_November"
## [477] "country_monthIND_June"         "country_monthIRL_July"
## [479] "country_monthIRL_June"         "country_monthIRL_May"
## [481] "country_monthIRL_October"      "country_monthIRN_February"
## [483] "country_monthIRN_March"        "country_monthISR_July"
## [485] "country_monthITA_July"         "country_monthKAZ_July"
## [487] "country_monthKEN_March"        "country_monthLUX_December"
## [489] "country_monthLUX_February"     "country_monthLUX_November"
## [491] "country_monthLVA_March"        "country_monthMAR_August"
## [493] "country_monthMAR_December"     "country_monthMDV_March"
## [495] "country_monthMEX_July"         "country_monthMLT_August"
## [497] "country_monthMOZ_June"         "country_monthMYS_December"
## [499] "country_monthNAM_April"        "country_monthNGA_March"
## [501] "country_monthNOR_July"         "country_monthNULL_September"
## [503] "country_monthOMN_January"      "country_monthPER_March"
## [505] "country_monthPRI_December"     "country_monthPRT_August"
## [507] "country_monthPRT_January"      "country_monthPRT_May"
## [509] "country_monthPRT_November"     "country_monthPRT_October"
## [511] "country_monthPRT_September"    "country_monthQAT_April"
## [513] "country_monthRUS_April"        "country_monthRUS_March"
## [515] "country_monthSAU_February"     "country_monthSAU_March"
## [517] "country_monthSGP_January"      "country_monthSVK_December"
## [519] "country_monthSVN_March"        "country_monthSWE_December"
## [521] "country_monthSWE_February"     "country_monthSWE_March"
## [523] "country_monthTHA_February"     "country_monthTHA_June"
## [525] "country_monthTJK_May"          "country_monthTUN_March"
## [527] "country_monthTUN_October"      "country_monthTUR_February"
## [529] "country_monthTUR_January"      "country_monthTUR_July"
## [531] "country_monthTWN_February"     "country_monthTZA_September"
## [533] "country_monthVEN_January"      "country_monthVEN_September"
## [535] "country_monthZAF_February"     "country_monthZAF_January"
## [537] "country_monthZMB_April"
```

LOG LOSS test OOS

Ahora pruebo el error log loss del lasso

```
#Predicciones
lasso_score <- predict(cvlasso_a,
               newdata = Xb,
               type="response",
               select = "min" )


#dataframe
lasso_validation <- data.frame(y, lasso_score)
colnames(lasso_validation)[2] <- c('lasso_score')

library(MLmetrics)
```

```
##
## Attaching package: 'MLmetrics'
```

```
## The following object is masked from 'package:base':
##
##     Recall
```

```
LogLoss(lasso_validation$lasso_score,lasso_validation$y)
```

```
## [1] 0.3090462
```

Nos dio un error sorprendetemente muy pequeño. Con este modelo logramos realizar un error de 0.41872 y 0.42131 en los datos de test de Kaggle.

XGBOOSTING

Sin embargo, para ganar el concurso optamos por explorar otros modelos que generalmente tienen mayor potencial de ganar este tipo de concursos: XG boosting.

En este caso, se eligieron los hiperparametros mediante un tuning manual explorando el comportamiento del error cuando se fijaban todos los hp excepto uno. De esta manera se fijo la profunidad máxima del arbol en 6 y el learning rate en .06.

Debido a la alta cantidad de variables de las bases de datos (y pues que muchas son poco informativas) el colsample por cada arbol generado es alto: del 70%. De haber tenido solo variables muy informativas pues bajariamos ese porcentaje, sin embargo quicimos explitar la capacidad del modelo de seleccionar por si solo las variables.

```
# Preparar la base de entrenamiento
library(xgboost)
```

```
## Warning: package 'xgboost' was built under R version 4.1.2
```

```
##
## Attaching package: 'xgboost'
```

```
## The following object is masked from 'package:plotly':
##
##     slice
```

```
## The following object is masked from 'package:dplyr':
##
##     slice
```

```
dtrain <- xgb.DMatrix(Xa, label = Ya)
# Label es el target
# Preparar la base de validación

dtest <- xgb.DMatrix(Xb, label = y)
watchlist <- list(train = dtrain, eval = dtest)
# Para evaluar el performance del modelo


# Entrenamiento del modelo

param <- list(max_depth = 6, learning_rate = 0.06,
              objective = "binary:logistic",
              eval_metric = "logloss", subsample = 0.6, colsample_bytree = 0.7)


xgb_model <- xgb.train(params = param, dtrain,
                       early_stopping_rounds =  10,
                       nrounds = 300,
                  watchlist)
```

```
## [1]  train-logloss:0.663428  eval-logloss:0.663375
## Multiple eval metrics are present. Will use eval_logloss for early stopping.
## Will train until eval_logloss hasn't improved in 10 rounds.
##
## [2]  train-logloss:0.634898  eval-logloss:0.634885
## [3]  train-logloss:0.609400  eval-logloss:0.609357
## [4]  train-logloss:0.590304  eval-logloss:0.590249
## [5]  train-logloss:0.569940  eval-logloss:0.570101
## [6]  train-logloss:0.553171  eval-logloss:0.553240
## [7]  train-logloss:0.536228  eval-logloss:0.536058
## [8]  train-logloss:0.521319  eval-logloss:0.521282
## [9]  train-logloss:0.506495  eval-logloss:0.506499
## [10] train-logloss:0.493713  eval-logloss:0.493711
## [11] train-logloss:0.479244  eval-logloss:0.479273
## [12] train-logloss:0.467418  eval-logloss:0.467470
## [13] train-logloss:0.455900  eval-logloss:0.456008
## [14] train-logloss:0.446469  eval-logloss:0.446674
## [15] train-logloss:0.436079  eval-logloss:0.436358
## [16] train-logloss:0.428049  eval-logloss:0.428415
## [17] train-logloss:0.421356  eval-logloss:0.421667
## [18] train-logloss:0.415267  eval-logloss:0.415680
## [19] train-logloss:0.407569  eval-logloss:0.407959
## [20] train-logloss:0.400186  eval-logloss:0.400534
## [21] train-logloss:0.393543  eval-logloss:0.394006
## [22] train-logloss:0.387478  eval-logloss:0.387865
## [23] train-logloss:0.383026  eval-logloss:0.383526
## [24] train-logloss:0.378405  eval-logloss:0.378801
## [25] train-logloss:0.373549  eval-logloss:0.374007
## [26] train-logloss:0.368149  eval-logloss:0.368593
## [27] train-logloss:0.363321  eval-logloss:0.363801
## [28] train-logloss:0.359582  eval-logloss:0.359956
## [29] train-logloss:0.356242  eval-logloss:0.356700
## [30] train-logloss:0.353195  eval-logloss:0.353653
## [31] train-logloss:0.350439  eval-logloss:0.350938
## [32] train-logloss:0.346486  eval-logloss:0.347032
## [33] train-logloss:0.343755  eval-logloss:0.344389
## [34] train-logloss:0.340230  eval-logloss:0.340952
## [35] train-logloss:0.338056  eval-logloss:0.338715
## [36] train-logloss:0.336203  eval-logloss:0.336835
## [37] train-logloss:0.332751  eval-logloss:0.333326
## [38] train-logloss:0.330175  eval-logloss:0.330887
## [39] train-logloss:0.328228  eval-logloss:0.328921
## [40] train-logloss:0.325833  eval-logloss:0.326538
## [41] train-logloss:0.324070  eval-logloss:0.324826
## [42] train-logloss:0.322018  eval-logloss:0.322778
## [43] train-logloss:0.320717  eval-logloss:0.321589
## [44] train-logloss:0.317912  eval-logloss:0.318838
## [45] train-logloss:0.316461  eval-logloss:0.317484
## [46] train-logloss:0.314206  eval-logloss:0.315265
## [47] train-logloss:0.312411  eval-logloss:0.313418
## [48] train-logloss:0.310827  eval-logloss:0.311873
## [49] train-logloss:0.309426  eval-logloss:0.310433
## [50] train-logloss:0.308177  eval-logloss:0.309295
## [51] train-logloss:0.307101  eval-logloss:0.308272
## [52] train-logloss:0.305802  eval-logloss:0.306991
## [53] train-logloss:0.304942  eval-logloss:0.306194
## [54] train-logloss:0.304079  eval-logloss:0.305366
## [55] train-logloss:0.302388  eval-logloss:0.303725
## [56] train-logloss:0.301583  eval-logloss:0.302983
## [57] train-logloss:0.300546  eval-logloss:0.302090
## [58] train-logloss:0.299848  eval-logloss:0.301415
## [59] train-logloss:0.298860  eval-logloss:0.300470
## [60] train-logloss:0.297304  eval-logloss:0.298996
## [61] train-logloss:0.296559  eval-logloss:0.298301
## [62] train-logloss:0.295152  eval-logloss:0.296951
## [63] train-logloss:0.294179  eval-logloss:0.295847
## [64] train-logloss:0.293384  eval-logloss:0.295112
## [65] train-logloss:0.292607  eval-logloss:0.294391
## [66] train-logloss:0.292040  eval-logloss:0.293911
```

```
## [67] train-logloss:0.290009  eval-logloss:0.292114
## [68] train-logloss:0.289500  eval-logloss:0.291646
## [69] train-logloss:0.288502  eval-logloss:0.290698
## [70] train-logloss:0.287767  eval-logloss:0.290050
## [71] train-logloss:0.287358  eval-logloss:0.289692
## [72] train-logloss:0.286838  eval-logloss:0.289252
## [73] train-logloss:0.285719  eval-logloss:0.288376
## [74] train-logloss:0.284919  eval-logloss:0.287606
## [75] train-logloss:0.284356  eval-logloss:0.287089
## [76] train-logloss:0.283192  eval-logloss:0.285997
## [77] train-logloss:0.282787  eval-logloss:0.285648
## [78] train-logloss:0.282346  eval-logloss:0.285208
## [79] train-logloss:0.281575  eval-logloss:0.284459
## [80] train-logloss:0.281134  eval-logloss:0.284083
## [81] train-logloss:0.280562  eval-logloss:0.283609
## [82] train-logloss:0.279631  eval-logloss:0.282838
## [83] train-logloss:0.279101  eval-logloss:0.282317
## [84] train-logloss:0.278513  eval-logloss:0.281749
## [85] train-logloss:0.278245  eval-logloss:0.281538
## [86] train-logloss:0.277773  eval-logloss:0.281202
## [87] train-logloss:0.276541  eval-logloss:0.280169
## [88] train-logloss:0.275834  eval-logloss:0.279492
## [89] train-logloss:0.275555  eval-logloss:0.279313
## [90] train-logloss:0.275313  eval-logloss:0.279119
## [91] train-logloss:0.274784  eval-logloss:0.278727
## [92] train-logloss:0.274345  eval-logloss:0.278394
## [93] train-logloss:0.273914  eval-logloss:0.278044
## [94] train-logloss:0.273518  eval-logloss:0.277686
## [95] train-logloss:0.272579  eval-logloss:0.276887
## [96] train-logloss:0.271922  eval-logloss:0.276274
## [97] train-logloss:0.271439  eval-logloss:0.275832
## [98] train-logloss:0.271148  eval-logloss:0.275572
## [99] train-logloss:0.270606  eval-logloss:0.275077
## [100]    train-logloss:0.270225  eval-logloss:0.274848
## [101]    train-logloss:0.270065  eval-logloss:0.274682
## [102]    train-logloss:0.269884  eval-logloss:0.274533
## [103]    train-logloss:0.269592  eval-logloss:0.274291
## [104]    train-logloss:0.269415  eval-logloss:0.274136
## [105]    train-logloss:0.269149  eval-logloss:0.273923
## [106]    train-logloss:0.268756  eval-logloss:0.273647
## [107]    train-logloss:0.268127  eval-logloss:0.273068
## [108]    train-logloss:0.267490  eval-logloss:0.272477
## [109]    train-logloss:0.267217  eval-logloss:0.272222
## [110]    train-logloss:0.266871  eval-logloss:0.272046
## [111]    train-logloss:0.266481  eval-logloss:0.271803
## [112]    train-logloss:0.266237  eval-logloss:0.271639
## [113]    train-logloss:0.265895  eval-logloss:0.271328
## [114]    train-logloss:0.265398  eval-logloss:0.270900
## [115]    train-logloss:0.265145  eval-logloss:0.270790
## [116]    train-logloss:0.264829  eval-logloss:0.270528
## [117]    train-logloss:0.264677  eval-logloss:0.270420
## [118]    train-logloss:0.264402  eval-logloss:0.270187
## [119]    train-logloss:0.264051  eval-logloss:0.269825
## [120]    train-logloss:0.263620  eval-logloss:0.269537
## [121]    train-logloss:0.263404  eval-logloss:0.269405
## [122]    train-logloss:0.263085  eval-logloss:0.269045
## [123]    train-logloss:0.262618  eval-logloss:0.268627
## [124]    train-logloss:0.262491  eval-logloss:0.268537
## [125]    train-logloss:0.262349  eval-logloss:0.268411
## [126]    train-logloss:0.262211  eval-logloss:0.268348
## [127]    train-logloss:0.262069  eval-logloss:0.268219
## [128]    train-logloss:0.261844  eval-logloss:0.268031
## [129]    train-logloss:0.261348  eval-logloss:0.267593
## [130]    train-logloss:0.261110  eval-logloss:0.267410
## [131]    train-logloss:0.260541  eval-logloss:0.267069
## [132]    train-logloss:0.260402  eval-logloss:0.266951
## [133]    train-logloss:0.260156  eval-logloss:0.266774
## [134]    train-logloss:0.259961  eval-logloss:0.266651
## [135]    train-logloss:0.259787  eval-logloss:0.266532
```

```
## [136]    train-logloss:0.259480  eval-logloss:0.266337
## [137]    train-logloss:0.259244  eval-logloss:0.266168
## [138]    train-logloss:0.259118  eval-logloss:0.266039
## [139]    train-logloss:0.258235  eval-logloss:0.265367
## [140]    train-logloss:0.258026  eval-logloss:0.265213
## [141]    train-logloss:0.257757  eval-logloss:0.264973
## [142]    train-logloss:0.257402  eval-logloss:0.264770
## [143]    train-logloss:0.257073  eval-logloss:0.264535
## [144]    train-logloss:0.256863  eval-logloss:0.264439
## [145]    train-logloss:0.256523  eval-logloss:0.264198
## [146]    train-logloss:0.256324  eval-logloss:0.264070
## [147]    train-logloss:0.255986  eval-logloss:0.263802
## [148]    train-logloss:0.255472  eval-logloss:0.263373
## [149]    train-logloss:0.255330  eval-logloss:0.263276
## [150]    train-logloss:0.255000  eval-logloss:0.263033
## [151]    train-logloss:0.254791  eval-logloss:0.262920
## [152]    train-logloss:0.254632  eval-logloss:0.262790
## [153]    train-logloss:0.254291  eval-logloss:0.262529
## [154]    train-logloss:0.254187  eval-logloss:0.262465
## [155]    train-logloss:0.253996  eval-logloss:0.262328
## [156]    train-logloss:0.253853  eval-logloss:0.262266
## [157]    train-logloss:0.253753  eval-logloss:0.262195
## [158]    train-logloss:0.253592  eval-logloss:0.262073
## [159]    train-logloss:0.253433  eval-logloss:0.261931
## [160]    train-logloss:0.253354  eval-logloss:0.261915
## [161]    train-logloss:0.253100  eval-logloss:0.261706
## [162]    train-logloss:0.252925  eval-logloss:0.261618
## [163]    train-logloss:0.252820  eval-logloss:0.261555
## [164]    train-logloss:0.252714  eval-logloss:0.261509
## [165]    train-logloss:0.252587  eval-logloss:0.261472
## [166]    train-logloss:0.252481  eval-logloss:0.261409
## [167]    train-logloss:0.252339  eval-logloss:0.261353
## [168]    train-logloss:0.252233  eval-logloss:0.261308
## [169]    train-logloss:0.251971  eval-logloss:0.261051
## [170]    train-logloss:0.251845  eval-logloss:0.261012
## [171]    train-logloss:0.251682  eval-logloss:0.260892
## [172]    train-logloss:0.251558  eval-logloss:0.260818
## [173]    train-logloss:0.251242  eval-logloss:0.260566
## [174]    train-logloss:0.251151  eval-logloss:0.260553
## [175]    train-logloss:0.251004  eval-logloss:0.260490
## [176]    train-logloss:0.250758  eval-logloss:0.260299
## [177]    train-logloss:0.250617  eval-logloss:0.260217
## [178]    train-logloss:0.250407  eval-logloss:0.260042
## [179]    train-logloss:0.250308  eval-logloss:0.259987
## [180]    train-logloss:0.249978  eval-logloss:0.259764
## [181]    train-logloss:0.249888  eval-logloss:0.259779
## [182]    train-logloss:0.249751  eval-logloss:0.259682
## [183]    train-logloss:0.249669  eval-logloss:0.259637
## [184]    train-logloss:0.249283  eval-logloss:0.259333
## [185]    train-logloss:0.249186  eval-logloss:0.259269
## [186]    train-logloss:0.249098  eval-logloss:0.259224
## [187]    train-logloss:0.248952  eval-logloss:0.259184
## [188]    train-logloss:0.248724  eval-logloss:0.259073
## [189]    train-logloss:0.248587  eval-logloss:0.258978
## [190]    train-logloss:0.248496  eval-logloss:0.258932
## [191]    train-logloss:0.248310  eval-logloss:0.258797
## [192]    train-logloss:0.248240  eval-logloss:0.258762
## [193]    train-logloss:0.248099  eval-logloss:0.258681
## [194]    train-logloss:0.247889  eval-logloss:0.258557
## [195]    train-logloss:0.247741  eval-logloss:0.258455
## [196]    train-logloss:0.247628  eval-logloss:0.258426
## [197]    train-logloss:0.247522  eval-logloss:0.258383
## [198]    train-logloss:0.247335  eval-logloss:0.258269
## [199]    train-logloss:0.247257  eval-logloss:0.258230
## [200]    train-logloss:0.247080  eval-logloss:0.258144
## [201]    train-logloss:0.246958  eval-logloss:0.258088
## [202]    train-logloss:0.246588  eval-logloss:0.257934
## [203]    train-logloss:0.246447  eval-logloss:0.257860
## [204]    train-logloss:0.246350  eval-logloss:0.257797
```

```
## [205]    train-logloss:0.246233  eval-logloss:0.257748
## [206]    train-logloss:0.246040  eval-logloss:0.257591
## [207]    train-logloss:0.245931  eval-logloss:0.257552
## [208]    train-logloss:0.245696  eval-logloss:0.257367
## [209]    train-logloss:0.245570  eval-logloss:0.257323
## [210]    train-logloss:0.245466  eval-logloss:0.257283
## [211]    train-logloss:0.245350  eval-logloss:0.257214
## [212]    train-logloss:0.245172  eval-logloss:0.257094
## [213]    train-logloss:0.245098  eval-logloss:0.257064
## [214]    train-logloss:0.244993  eval-logloss:0.256973
## [215]    train-logloss:0.244892  eval-logloss:0.256942
## [216]    train-logloss:0.244783  eval-logloss:0.256900
## [217]    train-logloss:0.244711  eval-logloss:0.256858
## [218]    train-logloss:0.244585  eval-logloss:0.256787
## [219]    train-logloss:0.244526  eval-logloss:0.256772
## [220]    train-logloss:0.244327  eval-logloss:0.256717
## [221]    train-logloss:0.244241  eval-logloss:0.256665
## [222]    train-logloss:0.244069  eval-logloss:0.256560
## [223]    train-logloss:0.243944  eval-logloss:0.256533
## [224]    train-logloss:0.243860  eval-logloss:0.256508
## [225]    train-logloss:0.243630  eval-logloss:0.256302
## [226]    train-logloss:0.243513  eval-logloss:0.256230
## [227]    train-logloss:0.243146  eval-logloss:0.255973
## [228]    train-logloss:0.243040  eval-logloss:0.255906
## [229]    train-logloss:0.242977  eval-logloss:0.255865
## [230]    train-logloss:0.242834  eval-logloss:0.255741
## [231]    train-logloss:0.242700  eval-logloss:0.255716
## [232]    train-logloss:0.242293  eval-logloss:0.255433
## [233]    train-logloss:0.242198  eval-logloss:0.255347
## [234]    train-logloss:0.242137  eval-logloss:0.255300
## [235]    train-logloss:0.242067  eval-logloss:0.255256
## [236]    train-logloss:0.241901  eval-logloss:0.255116
## [237]    train-logloss:0.241734  eval-logloss:0.255038
## [238]    train-logloss:0.241652  eval-logloss:0.255018
## [239]    train-logloss:0.241504  eval-logloss:0.254896
## [240]    train-logloss:0.241359  eval-logloss:0.254809
## [241]    train-logloss:0.241316  eval-logloss:0.254788
## [242]    train-logloss:0.241159  eval-logloss:0.254662
## [243]    train-logloss:0.241064  eval-logloss:0.254600
## [244]    train-logloss:0.240952  eval-logloss:0.254525
## [245]    train-logloss:0.240890  eval-logloss:0.254500
## [246]    train-logloss:0.240777  eval-logloss:0.254430
## [247]    train-logloss:0.240714  eval-logloss:0.254371
## [248]    train-logloss:0.240592  eval-logloss:0.254313
## [249]    train-logloss:0.240476  eval-logloss:0.254262
## [250]    train-logloss:0.240392  eval-logloss:0.254220
## [251]    train-logloss:0.240317  eval-logloss:0.254159
## [252]    train-logloss:0.240202  eval-logloss:0.254092
## [253]    train-logloss:0.240077  eval-logloss:0.254022
## [254]    train-logloss:0.240005  eval-logloss:0.253985
## [255]    train-logloss:0.239854  eval-logloss:0.253897
## [256]    train-logloss:0.239695  eval-logloss:0.253723
## [257]    train-logloss:0.239577  eval-logloss:0.253604
## [258]    train-logloss:0.239492  eval-logloss:0.253540
## [259]    train-logloss:0.239438  eval-logloss:0.253523
## [260]    train-logloss:0.239377  eval-logloss:0.253506
## [261]    train-logloss:0.239277  eval-logloss:0.253466
## [262]    train-logloss:0.239081  eval-logloss:0.253295
## [263]    train-logloss:0.238891  eval-logloss:0.253251
## [264]    train-logloss:0.238769  eval-logloss:0.253171
## [265]    train-logloss:0.238646  eval-logloss:0.253138
## [266]    train-logloss:0.238609  eval-logloss:0.253144
## [267]    train-logloss:0.238539  eval-logloss:0.253133
## [268]    train-logloss:0.238478  eval-logloss:0.253084
## [269]    train-logloss:0.238353  eval-logloss:0.252991
## [270]    train-logloss:0.238187  eval-logloss:0.252896
## [271]    train-logloss:0.238102  eval-logloss:0.252836
## [272]    train-logloss:0.238048  eval-logloss:0.252847
## [273]    train-logloss:0.237947  eval-logloss:0.252809
```

```
## [274]    train-logloss:0.237883  eval-logloss:0.252757
## [275]    train-logloss:0.237829  eval-logloss:0.252735
## [276]    train-logloss:0.237682  eval-logloss:0.252695
## [277]    train-logloss:0.237496  eval-logloss:0.252571
## [278]    train-logloss:0.237392  eval-logloss:0.252485
## [279]    train-logloss:0.237331  eval-logloss:0.252480
## [280]    train-logloss:0.237245  eval-logloss:0.252438
## [281]    train-logloss:0.237176  eval-logloss:0.252379
## [282]    train-logloss:0.237104  eval-logloss:0.252349
## [283]    train-logloss:0.237036  eval-logloss:0.252289
## [284]    train-logloss:0.236940  eval-logloss:0.252268
## [285]    train-logloss:0.236861  eval-logloss:0.252251
## [286]    train-logloss:0.236756  eval-logloss:0.252222
## [287]    train-logloss:0.236597  eval-logloss:0.252158
## [288]    train-logloss:0.236523  eval-logloss:0.252126
## [289]    train-logloss:0.236332  eval-logloss:0.252036
## [290]    train-logloss:0.236252  eval-logloss:0.251980
## [291]    train-logloss:0.236163  eval-logloss:0.251921
## [292]    train-logloss:0.236100  eval-logloss:0.251900
## [293]    train-logloss:0.236001  eval-logloss:0.251887
## [294]    train-logloss:0.235941  eval-logloss:0.251855
## [295]    train-logloss:0.235852  eval-logloss:0.251840
## [296]    train-logloss:0.235808  eval-logloss:0.251802
## [297]    train-logloss:0.235727  eval-logloss:0.251746
## [298]    train-logloss:0.235633  eval-logloss:0.251683
## [299]    train-logloss:0.235559  eval-logloss:0.251655
## [300]    train-logloss:0.235468  eval-logloss:0.251574
```

```
# Predicción
xgb_pred <- predict(xgb_model, Xb)
XGpred<-data.frame(y, xgb_pred)
colnames(XGpred)<-c("y","xgb_pred")
```

Se muestran las evaluaciones del modelo, tanto in sample como out of sample, para las primeras y últimas iteraciones.

```
LogLoss(XGpred$xgb_pred,XGpred$y)
```

```
## [1] 0.2515736
```

Este modelo logró ganar el concurso con un error en los datasets de kaggle de 0.37598 y 0.37401.

Conclusiones

- Los modelos lineales nos sirvieron para ir explorando la utilidad de las variables, parámetros y las caracteristicas del modelo sin embargo, una vez descubierto los insights pues podemos optar por modelos más competitivos.
- Como vimos en clase el EDA se debe hacer después de un CV para evitar encontrar hallazgos que generalizen poco.
- El usar matrices ralas nos permitio experimentar muy rapido con los modelos pues reducen el tiempo de entrenamiento. Sin embargo debemos tratar las bases de datos con mucho cuidado. Por ejemplo, se necesitavan nivelar las columnas para que las matrices tuvieran las mismas dimensiones.
- Se puede explotar al máximo la capacidad de cada modelo de ML de seleccionar las variables (y en consecuencia de crear bases de datos de alta dimensión) sin embargo se debe comprender el cómo lo hacen. En nuestro caso, esto implicaba indicarle al modelo que queremos un colsample por cada arbol alto: del 70% y que debemos limitar el tamaño de cada arbol en no más de 6 niveles.

Referencias

- Kaggle (https://www.kaggle.com/c/cancelaciones-en-hoteles/data)
- Hotel (https://www.sciencedirect.com/science/article/pii/S2352340918315191)
- Series de tiempo (https://es.wikipedia.org/wiki/Serie_temporal)
- One hot encoding (https://www.educative.io/blog/one-hot-encoding)
- Matrices Ralas (http://amunategui.github.io/sparse-matrix-glmnet/)
- Matrix (https://cran.rproject.org/web/packages/Matrix/index.html)
- XGBoost Documentation (https://xgboost.readthedocs.io/en/stable/)