Predicting Hotel-booking demand and cancellation

Instructor: Daniel D. Gutierrez

Class:2020 Winter Introduction to data science

Yilin Kuang

Contents

1 Introduction

1.1 Project introduction:

The cancellation rate for booking hotels online is high that creates discomfort for many hotels and create a desire to take precautions. Therefore, predicting reservations that can be cancelled will create a surplus value for hotels and hotels can take action to prevent these cancellations.

In my final project, I will try to explore the dataset and explain how to predict future cancelled reservations in advance by machine learning methods.

Kaggle describes this dataset as follows:

Have you ever wondered when the best time of year to book a hotel room is? Or the optimal length of stay in order to get the best daily rate? What if you wanted to predict whether or not a hotel was likely to receive a disproportionately high number of special requests? This hotel booking dataset can help you explore those questions!

This data set contains booking information for a city hotel and a resort hotel, and includes information such as when the booking was made, length of stay, the number of adults, children, and/or babies, and the number of available parking spaces, among other things. All personally identifying information has been removed from the data.

The data is originally from the article Hotel Booking Demand Datasets, written by Nuno Antonio, Ana Almeida, and Luis Nunes for Data in Brief, Volume 22, February 2019.

The data was downloaded and cleaned by Thomas Mock and Antoine Bichat for TidyTuesday during the week of February 11th, 2020.

1.2 Data set

The dataset could be downloaded from Kaggle with the following link. it's a new data set.https://www.kaggle.com/jessemostipak/hotel-booking-demand

There are 119390 observations and 32 variables in the dataset.

Hotel	Hotel (H1 = Resort Hotel or H2 = City Hotel)
Is_canceled	Value indicating if the booking was canceled (1) or not (0)
lead_time	Number of days that elapsed between the entering date of the booking
arrival_date_year	Year of arrival date;
arrival_date_month	Month of arrival date;
arrival_date_week_number	Week number of year for arrival date;
arrival_date_day_of_month	Day of arrival date
stays_in_weekend_nights	Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel;
stays_in_week_nights	Number of week nights (Monday to Friday) the guest stayed or booked to stay at the hotel;
adults	Number of adults
children	Number of children
babies	Number of babies
meal	Type of meal booked.

Country	Country of origin.
market_segment	Market segment designation
distribution_channel	Booking distribution channel.
is_repeated_guest	Value indicating if the booking name was from a repeated guest (1) or not (0)
previous_cancellations	Num ber of previous bookings that were cancelled by the customer prior to the current booking
previous_bookings_not_canceled	Number of previous bookings not cancelled by the customer prior to the current booking
reserved_room_type	Code of room type reserved. Code is presented instead of designation for anonymity reasons
assigned_room_type	Code for the type of room assigned to the booking.
booking_changes	Number of changes/amendments made to the booking from the moment
deposit_type	Indication on if the customer made a deposit to guarantee the booking.
agent	ID of the travel agency that made the booking
company	ID of the company/entity that made the booking or responsible for paying the booking.
days_in_waiting_list	Number of days the booking was in the waiting list before it was confirmed to the customer
customer_type	Type of booking
adr	Average Daily Rate as defined by dividing the sum of all lodging transactions by the total number of staying nights
required_car_parking_spaces	Number of car parking spaces required by the customer
total_of_special_requests	Number of special requests made by the customer
reservation_status	Reservation last status
reservation_status_date	Date at which the last status was set

2 Loading and exploring data

2.1 Overview

In total, there are more than one hundred thousand observations and 32 variables and of which one is the response variable(reservation_status).I am displaying only a glimpse of the variables. All of them I will discuss in more detail through the project.

Here are some steps to understanding of data sets preliminary，I used head(), str(), summary() to simply look at the data.

```
> dim(df)
[1] 119390     32
```

```
> str(df)
'data.frame':	119390 obs. of  32 variables:
 $ hotel                         : Factor w/ 2 levels "City Hotel","Resort Hotel": 2 2 2 2 2 2 2 2 2 2 ...
 $ is_canceled                   : int  0 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 2015 ...
 $ arrival_date_month            : Factor w/ 12 levels "April","August",..: 6 6 6 6 6 6 6 6 6 6 ...
 $ arrival_date_week_number      : int  27 27 27 27 27 27 27 27 27 27 ...
 $ arrival_date_day_of_month     : int  1 1 1 1 1 1 1 1 1 1 ...
 $ stays_in_weekend_nights       : int  0 0 0 0 0 0 0 0 0 0 ...
 $ stays_in_week_nights          : int  0 0 1 1 2 2 2 2 3 3 ...
 $ adults                        : int  2 2 1 1 2 2 2 2 2 2 ...
 $ children                      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ babies                        : int  0 0 0 0 0 0 0 0 0 0 ...
 $ meal                          : Factor w/ 5 levels "BB","FB","HB",..: 1 1 1 1 1 1 1 2 1 3 ...
 $ country                       : Factor w/ 178 levels "ABW","AGO","AIA",..: 137 137 60 60 60 60 137 137 137 137 ...
 $ market_segment                : Factor w/ 8 levels "Aviation","Complementary",..: 4 4 4 3 7 7 4 4 7 6 ...
 $ distribution_channel          : Factor w/ 5 levels "Corporate","Direct",..: 2 2 2 1 4 4 2 2 4 4 ...
 $ is_repeated_guest             : int  0 0 0 0 0 0 0 0 0 0 ...
 $ previous_cancellations        : int  0 0 0 0 0 0 0 0 0 0 ...
 $ previous_bookings_not_canceled: int  0 0 0 0 0 0 0 0 0 0 ...
 $ reserved_room_type            : Factor w/ 10 levels "A","B","C","D",..: 3 3 1 1 1 1 3 3 1 4 ...
 $ assigned_room_type            : Factor w/ 12 levels "A","B","C","D",..: 3 3 3 1 1 1 3 3 1 4 ...
 $ booking_changes               : int  3 4 0 0 0 0 0 0 0 0 ...
 $ deposit_type                  : Factor w/ 3 levels "No Deposit","Non Refund",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ agent                         : Factor w/ 334 levels "1","10","103",..: 334 334 334 157 103 103 334 156 103 40 ...
 $ company                       : Factor w/ 353 levels "10","100","101",..: 353 353 353 353 353 353 353 353 353 353 ...
 $ days_in_waiting_list          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ customer_type                 : Factor w/ 4 levels "Contract","Group",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ adr                           : num  0 0 75 75 98 ...
 $ required_car_parking_spaces   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ total_of_special_requests     : int  0 0 0 0 1 1 0 1 1 0 ...
 $ reservation_status            : Factor w/ 3 levels "Canceled","Check-Out",..: 2 2 2 2 2 2 2 2 1 1 ...
 $ reservation_status_date       : Factor w/ 926 levels "1/1/2015","1/1/2016",..: 669 669 702 702 735 735 735 735 570 449 ...
```

```
> summary(df)
          hotel        is_canceled       lead_time    arrival_date_year arrival_date_month arrival_date_week_number
 City Hotel  :79330   Min.   :0.0000   Min.   :  0   Min.   :2015      August :13877      Min.   : 1.00           
 Resort Hotel:40060   1st Qu.:0.0000   1st Qu.: 18   1st Qu.:2016      July   :12661      1st Qu.:16.00           
                      Median :0.0000   Median : 69   Median :2016      May    :11791      Median :28.00           
                      Mean   :0.3704   Mean   :104   Mean   :2016      October:11160      Mean   :27.17           
                      3rd Qu.:1.0000   3rd Qu.:160   3rd Qu.:2017      April  :11089      3rd Qu.:38.00           
                      Max.   :1.0000   Max.   :737   Max.   :2017      June   :10939      Max.   :53.00           
                                                                       (Other):47873                              
 arrival_date_day_of_month stays_in_weekend_nights stays_in_week_nights     adults          children      
 Min.   : 1.0              Min.   : 0.0000         Min.   : 0.0          Min.   : 0.000   Min.   : 0.0000  
 1st Qu.: 8.0              1st Qu.: 0.0000         1st Qu.: 1.0          1st Qu.: 2.000   1st Qu.: 0.0000  
 Median :16.0              Median : 1.0000         Median : 2.0          Median : 2.000   Median : 0.0000  
 Mean   :15.8              Mean   : 0.9276         Mean   : 2.5          Mean   : 1.856   Mean   : 0.1039  
 3rd Qu.:23.0              3rd Qu.: 2.0000         3rd Qu.: 3.0          3rd Qu.: 2.000   3rd Qu.: 0.0000  
 Max.   :31.0              Max.   :19.0000         Max.   :50.0          Max.   :55.000   Max.   :10.0000  
                                                                                          NA's   :4        
     babies                  meal          country             market_segment  distribution_channel is_repeated_guest
 Min.   : 0.000000   BB       :92310   PRT    :48590   Online TA    :56477   Corporate: 6677      Min.   :0.00000  
 1st Qu.: 0.000000   FB       :  798   GBR    :12129   Offline TA/TO:24219   Direct   :14645      1st Qu.:0.00000  
 Median : 0.000000   HB       :14463   FRA    :10415   Groups       :19811   GDS      :  193      Median :0.00000  
 Mean   : 0.007949   SC       :10650   ESP    : 8568   Direct       :12606   TA/TO    :97870      Mean   :0.03191  
 3rd Qu.: 0.000000   Undefined: 1169   DEU    : 7287   Corporate    : 5295   Undefined:    5      3rd Qu.:0.00000  
 Max.   :10.000000                     ITA    : 3766   Complementary:  743                        Max.   :1.00000  
                                       (Other):28635   (Other)      :  239                                          
 previous_cancellations previous_bookings_not_canceled reserved_room_type assigned_room_type booking_changes  
 Min.   : 0.00000       Min.   : 0.0000                A      :85994      A      :74053      Min.   : 0.0000  
 1st Qu.: 0.00000       1st Qu.: 0.0000                D      :19201      D      :25322      1st Qu.: 0.0000  
 Median : 0.00000       Median : 0.0000                E      : 6535      E      : 7806      Median : 0.0000  
 Mean   : 0.08712       Mean   : 0.1371                F      : 2897      F      : 3751      Mean   : 0.2211  
 3rd Qu.: 0.00000       3rd Qu.: 0.0000                G      : 2094      G      : 2553      3rd Qu.: 0.0000  
 Max.   :26.00000       Max.   :72.0000                B      : 1118      C      : 2375      Max.   :21.0000  
                                                       (Other): 1551      (Other): 3530                       
     deposit_type        agent          company       days_in_waiting_list         customer_type        adr         
 No Deposit:104641   9      :31961   NULL   :112593   Min.   :  0.000      Contract       : 4076   Min.   :  -6.38  
 Non Refund: 14587   NULL   :16340   40     :   927   1st Qu.:  0.000      Group          :  577   1st Qu.:  69.29  
 Refundable:   162   240    :13922   223    :   784   Median :  0.000      Transient      :89613   Median :  94.58  
                     1      : 7191   67     :   267   Mean   :  2.321      Transient-Party:25124   Mean   : 101.83  
                     14     : 3640   45     :   250   3rd Qu.:  0.000                              3rd Qu.: 126.00  
                     7      : 3539   153    :   215   Max.   :391.000                              Max.   :5400.00  
                     (Other):42797   (Other):  4354                                                                 
 required_car_parking_spaces total_of_special_requests reservation_status reservation_status_date
 Min.   :0.00000             Min.   :0.0000            Canceled :43017     10/21/2015:  1461     
 1st Qu.:0.00000             1st Qu.:0.0000            Check-Out:75166     7/6/2015  :   805     
 Median :0.00000             Median :0.0000            No-Show  : 1207     11/25/2016:   790     
 Mean   :0.06252             Mean   :0.5714                                1/1/2015  :   763     
 3rd Qu.:0.00000             3rd Qu.:1.0000                                1/18/2016 :   625     
 Max.   :8.00000             Max.   :5.0000                                7/2/2015  :   469     
                                                                           (Other)   :114477     
```

```
resort_hotel <- newdf[which(newdf$reservation_status!="Canceled"& newdf$hotel == "Resort Hotel"),]
city_hotel <- newdf[which(newdf$reservation_status!="Canceled"&newdf$hotel == "City Hotel"),]
```

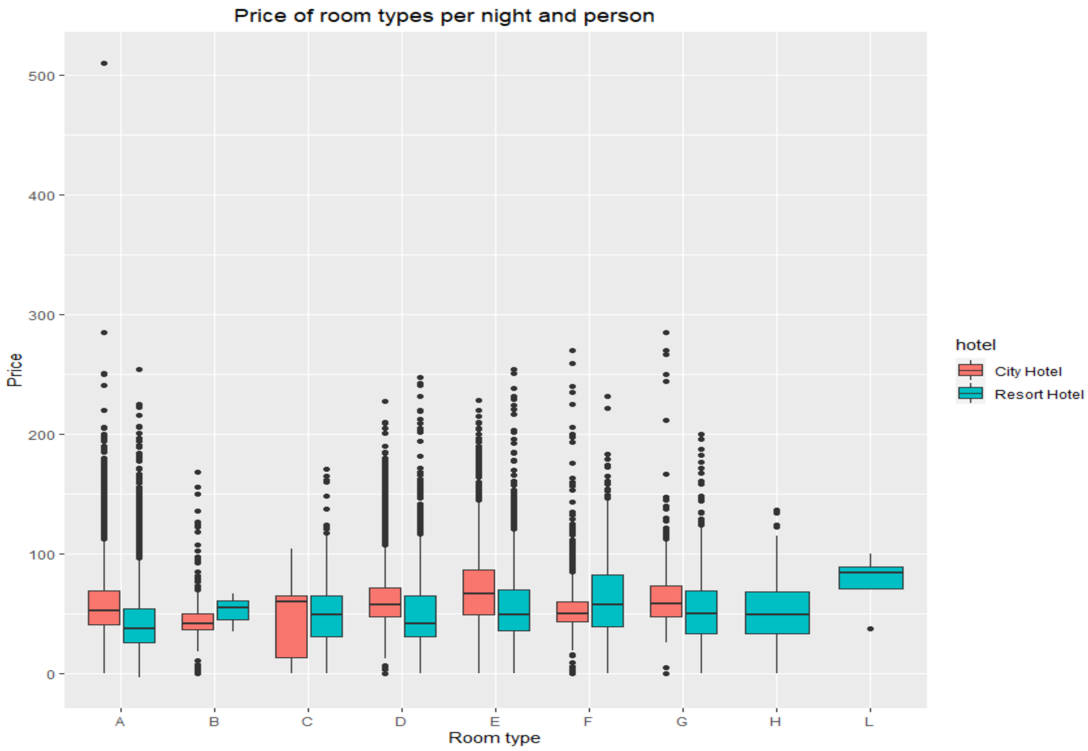
we can easily found we have 2 levels for Hotel, Resort hotel and City hotel, I divided the data set into two sub-data sets according to hotel level try to find some difference differences between this two kind of hotels.

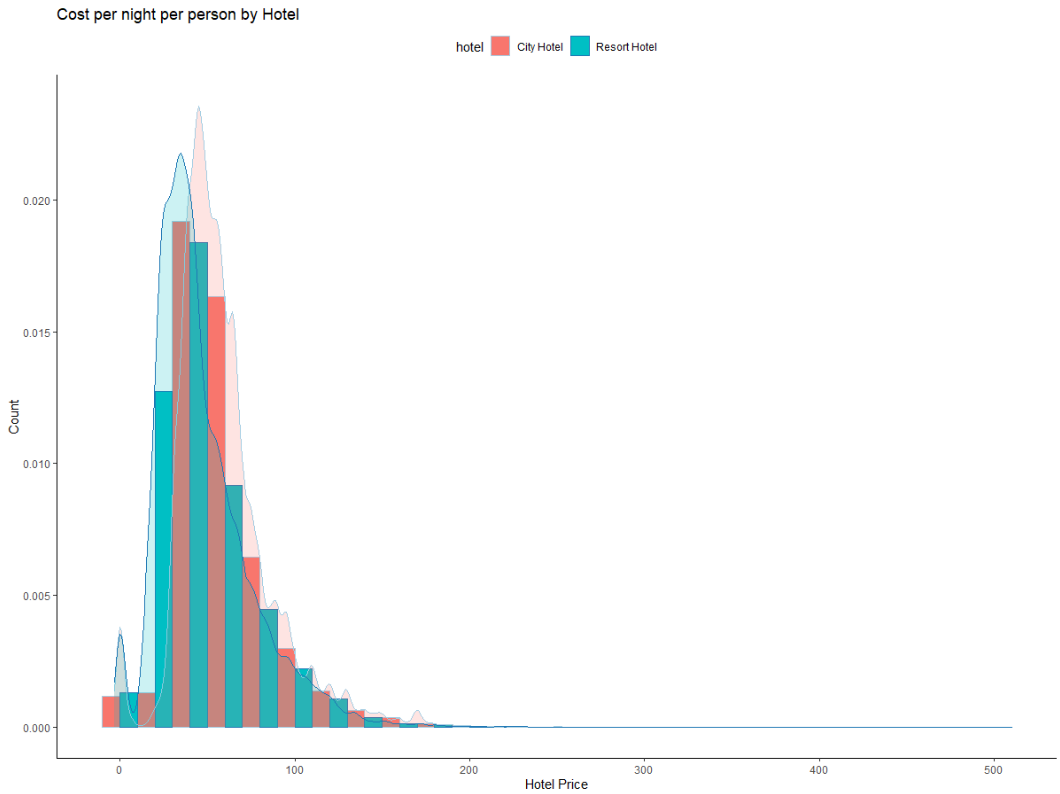
2.2 Correlation between room type and cost of per person per night

Both resort hotel and city hotel have different room types and different meal arrangements. Seasonal factors are also important. Therefore, the prices vary a lot.

I use ggplot to show the correlation between room type and cost of per person per night.

This figure shows the average price per room, depending on its type and the standard deviation. But pay attention, rooms with the same type letter may not necessarily be the same across hotels because of the data anonymization.



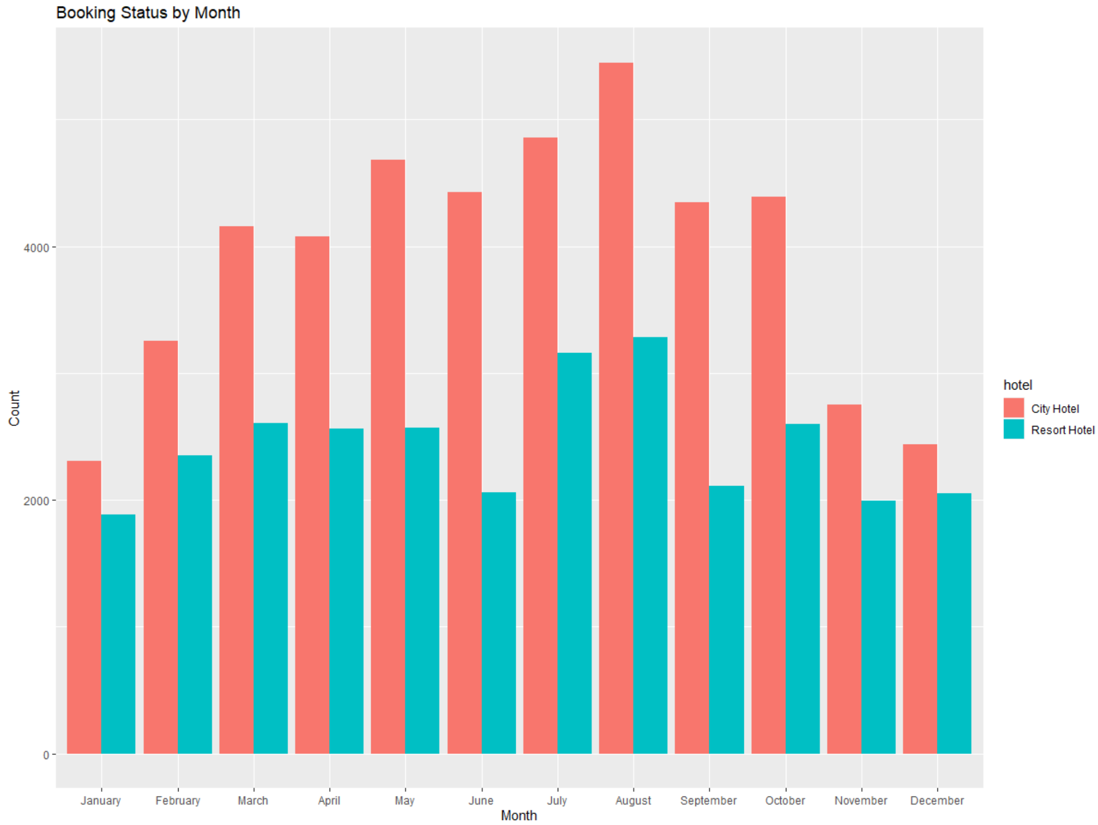


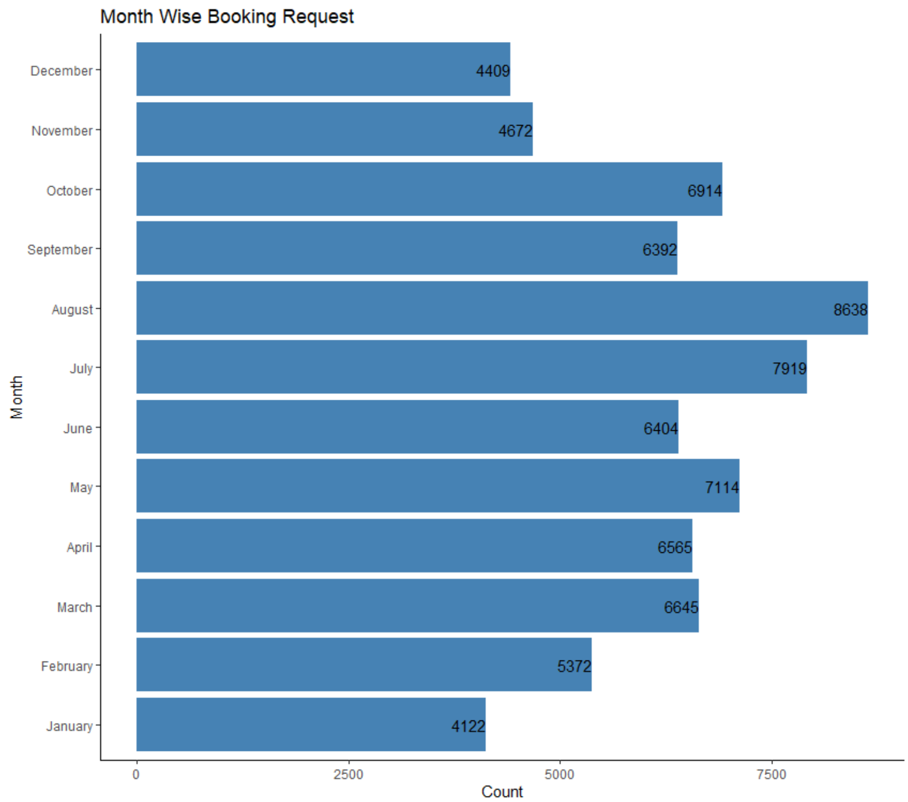
2.3 Hotel traffic on Monthly basis

From variable arrival_date_month, we can know the which month hotel the customer plan to check in,

Let's look at the this 2 plots of monthly trend of booking demand，And just to show you a little bit more clearly, I used 2 tricks, one is flipping the horizontal axis and the vertical axis and order the month chronologically

From the month wise booking analysis, we found out that most of hotel booking request came in the month of July and August followed by May and October. I infer one reason for this may be the weather impact as these are the months of pleasant weather in Europe.





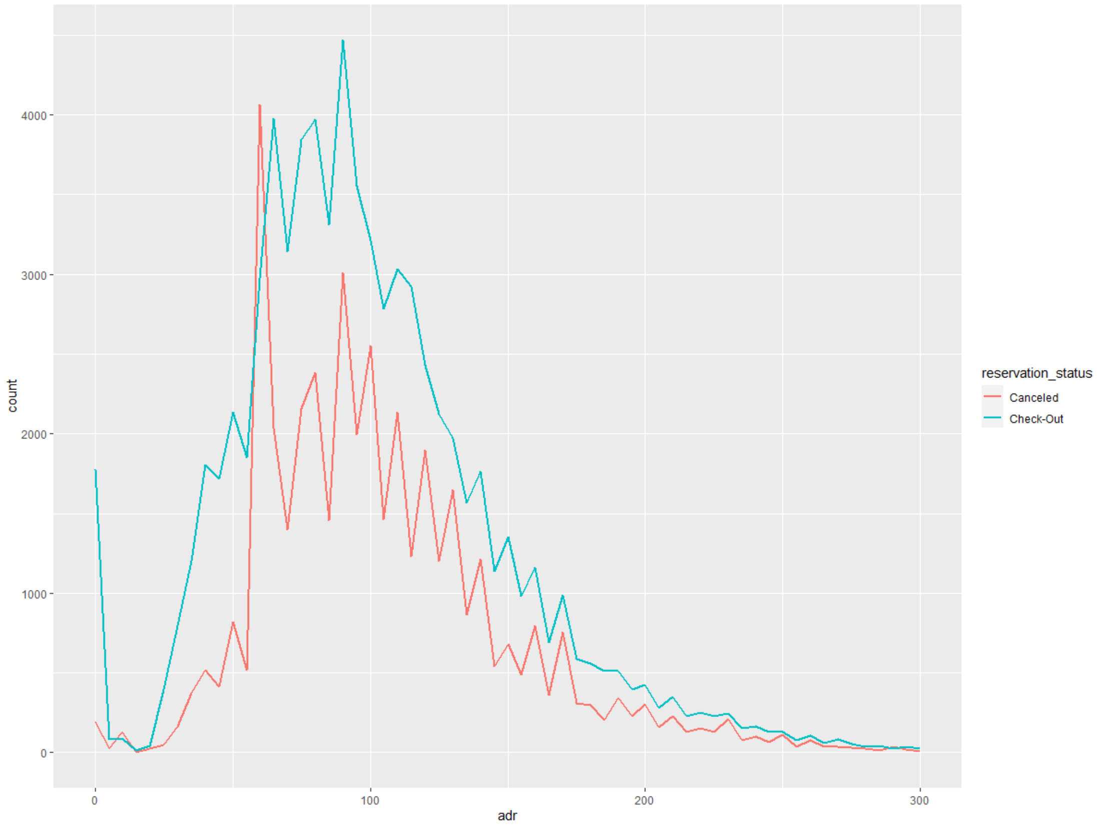
2.4 Distributions of continuous(numerical) variables: (group by reservation_status)

```
# ggplot(data = copy_newdf, aes(lead_time, color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
#
# ggplot(data = copy_newdf, aes(arrival_date_year, color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
# ggplot(data = copy_newdf, aes(arrival_date_day_of_month , color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
# ggplot(data = copy_newdf, aes(stays_in_weekend_nights , color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
# ggplot(data = copy_newdf, aes(stays_in_week_nights  , color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
# ggplot(data = copy_newdf, aes(adults , color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
# ggplot(data = copy_newdf, aes( children  , color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
# ggplot(data = copy_newdf, aes(is_repeated_guest, color = reservation_status))+
#   geom_freqpoly()
# ggplot(data = copy_newdf, aes(previous_cancellations, color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
# ggplot(data = copy_newdf, aes(previous_bookings_not_canceled, color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
# ggplot(data = copy_newdf, aes(booking_changes , color = reservation_status))+
#   geom_freqpoly(binwidth = 5, size = 1)
```

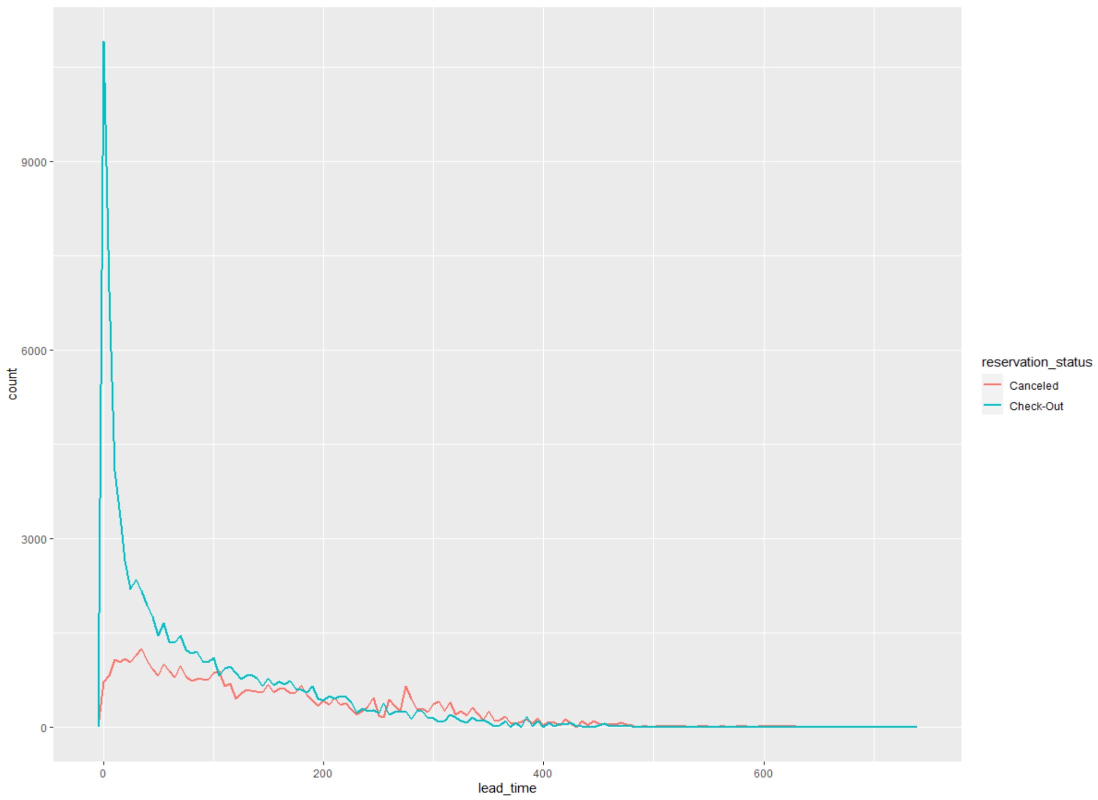
I try to find some differences among the guests in different reservation_status.

For continuous variables, I check for distributions of them. According to the different reservation status, draw the adr distribution plot guest whose reservation status is cancelled or check-out respectively.

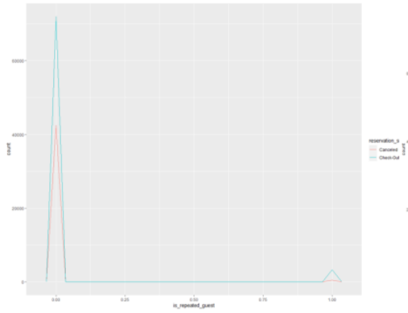
In adr plot, Basically, in all price ranges, the probability of a check-in is higher than the probability of a cancel, but when the price is 60, the cancel rate is very high, and I guess maybe when the cost lower than 60, and the consumer doesn't care about the money, so they don't cancel, and when the price is higher, they choose check-in because cancel may also cause cancel fee. And the rest of the distribution plots, sorry I didn’t find any insights from them. I still select some distribution of the continuous variables to display, although most of them haven't been very useful, I just want to show you some of the experiments that I've done.

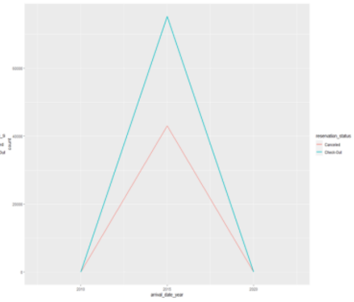


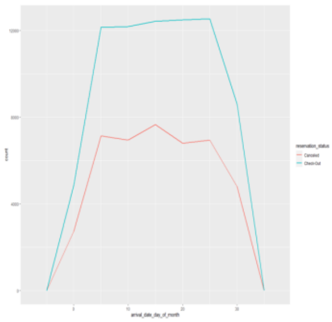
count
4000
3000
2000
1000
0
0
100
200
300
adr
reservation_status
Canceled
Check-Out



count
9000
6000
3000
0
0
200
400
600
lead_time
reservation_status
Canceled
Check-Out



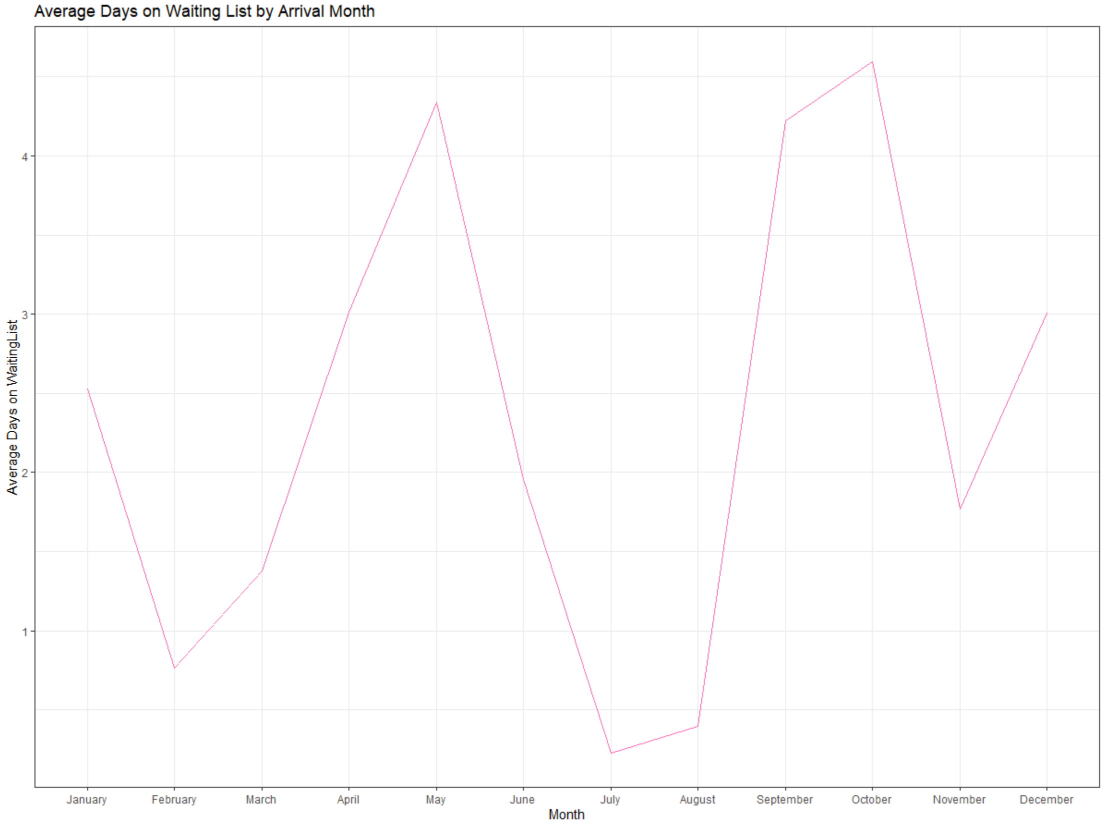




2.5 Average days of waiting list by arrival month:

I then explored the relationship between time and months on the waiting-list, as it is common knowledge that waiting longer on the waiting-list increases the likelihood of cancellation.

From this plot, we can find out that In summer, the waiting time of guests is relatively short, April, may, September and October is the peak, the waiting time is the longest



Then after exploring the date, we will move to model prediction part. I just select I only chose 2% of the data to train my model because this data set have more than ten hundred observations and my computer was slow.

3 Data Transformation (Preprocessing)

3.1 Data Size and Structure

I used read.csv() and set the stringsAsFactors = TRUE. During this process, all of the string variables have already changed to factor variables, therefore I don’t need to transfer to factor variables that reduces my workload. This dataset consist of integer, numeric and factor variables. In total, there are 32 columns/variables(18 numeric variables and 14 categoric variabels), of which one is the response variable (is_canceled).

```r
copy_newdf <- read.csv("hotel_bookings.csv",stringsAsFactors = T)
# numeric_vars
numeric_vars <-which(sapply(copy_newdf,is.numeric))
factor_vars <- which(sapply(copy_newdf, is.factor))
cat('There are', length(numeric_vars), 'numeric variables, and', length(factor_vars), 'categoric variables')

#There are 18 numeric variables, and 14 categoric variables
```

3.2 Deletion of missing observations

First of all, I just deleted missing observations from columns with a small number of NULL values. From the “summary()”, I found there are just 4 missing values in children column and deleted these 4 observations.

```
> summary(df)
          hotel         is_canceled        lead_time     arrival_date_year arrival_date_month arrival_date_week_number
 City Hotel   :79330   Min.   :0.0000   Min.   :  0   Min.   :2015      August  :13877     Min.   : 1.00
 Resort Hotel:40060   1st Qu.:0.0000   1st Qu.: 18   1st Qu.:2016      July    :12661     1st Qu.:16.00
                      Median :0.0000   Median : 69   Median :2016      May     :11791     Median :28.00
                      Mean   :0.3704   Mean   :104   Mean   :2016      October:11160      Mean   :27.17
                      3rd Qu.:1.0000   3rd Qu.:160   3rd Qu.:2017      April   :11089     3rd Qu.:38.00
                      Max.   :1.0000   Max.   :737   Max.   :2017      June    :10939     Max.   :53.00
                                                                       (Other):47873
 arrival_date_day_of_month stays_in_weekend_nights stays_in_week_nights     adults           children
 Min.   : 1.0              Min.   : 0.0000         Min.   : 0.0         Min.   : 0.000   Min.   : 0.0000
 1st Qu.: 8.0              1st Qu.: 0.0000         1st Qu.: 1.0         1st Qu.: 2.000   1st Qu.: 0.0000
 Median :16.0              Median : 1.0000         Median : 2.0         Median : 2.000   Median : 0.0000
 Mean   :15.8              Mean   : 0.9276         Mean   : 2.5         Mean   : 1.856   Mean   : 0.1039
 3rd Qu.:23.0              3rd Qu.: 2.0000         3rd Qu.: 3.0         3rd Qu.: 2.000   3rd Qu.: 0.0000
 Max.   :31.0              Max.   :19.0000         Max.   :50.0         Max.   :55.000   Max.   :10.0000
                                                                                         NA's   :4
     babies                  meal            country              market_segment    distribution_channel is_repeated_guest
 Min.   : 0.000000   BB        :92310   PRT    :48590   Online TA    :56477   Corporate: 6677      Min.   :0.00000
 1st Qu.: 0.000000   FB        :  798   GBR    :12129   Offline TA/TO:24219   Direct   :14645      1st Qu.:0.00000
 Median : 0.000000   HB        :14463   FRA    :10415   Groups       :19811   GDS      :  193      Median :0.00000
 Mean   : 0.007949   SC        :10650   ESP    : 8568   Direct       :12606   TA/TO    :97870      Mean   :0.03191
 3rd Qu.: 0.000000   Undefined: 1169   DEU    : 7287   Corporate    : 5295   Undefined:    5      3rd Qu.:0.00000
 Max.   :10.000000                     ITA    : 3766   Complementary:  743                        Max.   :1.00000
                                       (Other):28635   (Other)      :  239
 previous_cancellations previous_bookings_not_canceled reserved_room_type assigned_room_type booking_changes
 Min.   : 0.00000       Min.   : 0.0000                A      :85994      A      :74053      Min.   : 0.0000
 1st Qu.: 0.00000       1st Qu.: 0.0000                D      :19201      D      :25322      1st Qu.: 0.0000
 Median : 0.00000       Median : 0.0000                E      : 6535      E      : 7806      Median : 0.0000
 Mean   : 0.08712       Mean   : 0.1371                F      : 2897      F      : 3751      Mean   : 0.2211
 3rd Qu.: 0.00000       3rd Qu.: 0.0000                G      : 2094      G      : 2553      3rd Qu.: 0.0000
 Max.   :26.00000       Max.   :72.0000                B      : 1118      C      : 2375      Max.   :21.0000
                                                       (Other): 1551      (Other): 3530
     deposit_type          agent            company        days_in_waiting_list          customer_type        adr
 No Deposit:104641   9      :31961   NULL   :112593   Min.   :  0.000      Contract       : 4076   Min.   :  -6.38
 Non Refund: 14587   NULL   :16340   40     :    927   1st Qu.:  0.000      Group          :  577   1st Qu.:  69.29
 Refundable:   162   240    :13922   223    :    784   Median :  0.000      Transient      :89613   Median :  94.58
                     1      : 7191   67     :    267   Mean   :  2.321      Transient-Party:25124   Mean   : 101.83
                     14     : 3640   45     :    250   3rd Qu.:  0.000                              3rd Qu.: 126.00
                     7      : 3539   153    :    215   Max.   :391.000                              Max.   :5400.00
                     (Other):42797   (Other):  4354
 required_car_parking_spaces total_of_special_requests reservation_status reservation_status_date
 Min.   :0.00000             Min.   :0.0000            Canceled :43017    10/21/2015:  1461
 1st Qu.:0.00000             1st Qu.:0.0000            Check-Out:75166    7/6/2015  :   805
 Median :0.00000             Median :0.0000            No-Show  : 1207    11/25/2016:   790
 Mean   :0.06252             Mean   :0.5714                               1/1/2015  :   763
 3rd Qu.:0.00000             3rd Qu.:1.0000                               1/18/2016 :   625
 Max.   :8.00000             Max.   :5.0000                               7/2/2015  :   469
                                                                          (Other)   :114477
> newdf <- df[complete.cases(df),]
```

3.3 Deletion of some variables

First of all, I am dropping a variable if two variables are highly correlated. For example, the variable **reservation_status** is highly correlated to **is_canceled.**

Secondly, I drop some variables which has more than **53 categories,** since random forest can't handle categorical predictors with more than 53 categories in R. I drop some variables manually, such as company, country, adr_app, reservation_status_date, agent, reservation_status.

```
set.seed(2020)
data_RF = df[sample(1:nrow(copy_newdf),0.02*nrow(copy_newdf),replace = FALSE),]
data_RF = na.omit(data_RF)
data_RF$is_canceled<- as.factor(data_RF$is_canceled)
drops <- c("company","country","adr_pp","reservation_status_date","agent","reservation_status")
data_RF<-data_RF[ , !(names(data_RF) %in% drops)]
smp_size = floor(0.80*nrow(data_RF))
smp_size
train_ind = sample(seq_len(nrow(data_RF)),size=smp_size)
train1 = data_RF[train_ind,]
test1 = data_RF[-train_ind,]
```

3.4 Subtraction of dataset

I use only 2% data of processed dataset which I've been working with before, because the dataset is large and has 119390 observations.

3.5 Split of dataset

I subtract 2% of the dataset and name data_RF. And using data_RF to split train dataset and test dataset for modelling. Taking 80% observations of the data_RF as training dataset(train1) and 20% observations of the data_RF as testing dataset(test1).

4 Feature engineering (Exploring some of the most important variables)

4.1 Responsive variable

Responsive variable is **is_canceled** whose value indicating if the booking was canceled (1) or not (0),

4.2 The most important numeric predictive variables

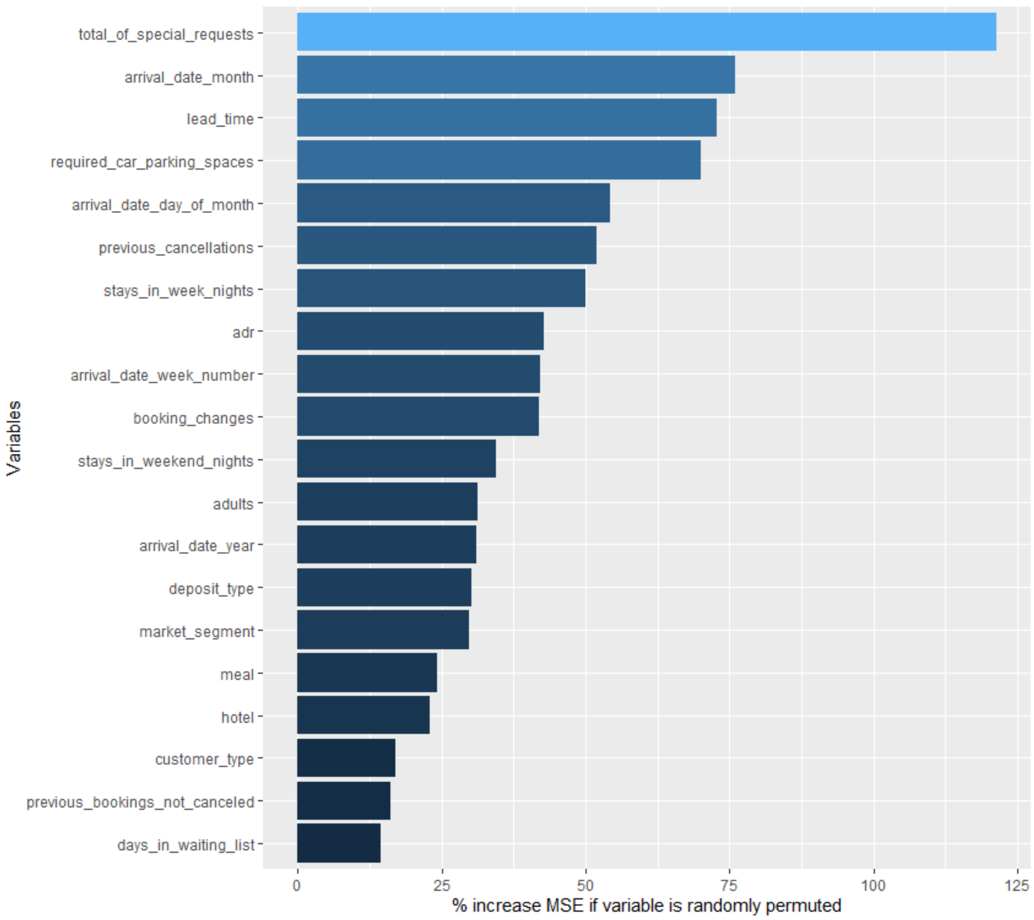
And I'm going to Predict the cancellation to help hotel do some preparation ahead of time. This is a classification problem. I choose is_canceled as the response variable in my model. And I'm going to use logistic regression and random forest to train several models in the next part.

I explore numeric variables at first, I extracted the numeric variable to form a new data frame. I what to know Which numerical features are most relevant to response variable? So I use cor()function to get the correlations. And here is correlation table, from this list it is apparent that lead_time, previous_cancellations total_of_special_requests, required_car_parking_spaces, booking_changes and is_repeated_guest are the 5 most important numerical features.

```
                                    [,1]
is_canceled                  1.000000000
lead_time                    0.302534922
previous_cancellations       0.112416812
adults                       0.065348610
days_in_waiting_list         0.056446949
adr                          0.049693795
stays_in_week_nights         0.024442472
arrival_date_year            0.017459235
arrival_date_week_number     0.011411500
children                     0.004637212
stays_in_weekend_nights     -0.004086414
arrival_date_day_of_month   -0.008004968
babies                      -0.032706968
previous_bookings_not_canceled -0.057929946
is_repeated_guest           -0.086453638
booking_changes             -0.146641583
required_car_parking_spaces -0.193852401
total_of_special_requests   -0.237241383
```

4.3 Variable Importance

Although the correlations are giving a good overview of the most important numeric variables, but we still have categorical variables, I wanted to get an overview of the most important variables including the categorical variables. I use a quick **random forest** to create this Variable importance.




```
                                                      Variables          MSE
reservation_status                           reservation_status  179.2906101
lead_time                                             lead_time    6.5248336
deposit_type                                       deposit_type    6.5148596
required_car_parking_spaces         required_car_parking_spaces    6.4630850
arrival_date_month                           arrival_date_month    6.3976407
previous_cancellations                   previous_cancellations    6.2543801
market_segment                                   market_segment    6.2480218
adr                                                         adr    5.8860746
stays_in_week_nights                       stays_in_week_nights    5.5902527
customer_type                                     customer_type    5.5296239
arrival_date_day_of_month             arrival_date_day_of_month    5.2766241
stays_in_weekend_nights                 stays_in_weekend_nights    5.0935349
meal                                                       meal    5.0371903
adults                                                   adults    4.6255547
distribution_channel                       distribution_channel    4.3690419
previous_bookings_not_canceled previous_bookings_not_canceled    4.1445683
total_of_special_requests             total_of_special_requests    4.0816632
arrival_date_week_number               arrival_date_week_number    4.0076518
booking_changes                                 booking_changes    3.9978371
is_repeated_guest                             is_repeated_guest    3.5984210
arrival_date_year                             arrival_date_year    3.5186872
hotel                                                     hotel    3.2989233
children                                               children    2.7681817
days_in_waiting_list                       days_in_waiting_list    2.1523685
assigned_room_type                           assigned_room_type    0.9300444
reserved_room_type                           reserved_room_type    0.6498728
babies                                                   babies   -1.3576330
```

5.Machine learning algorithm.

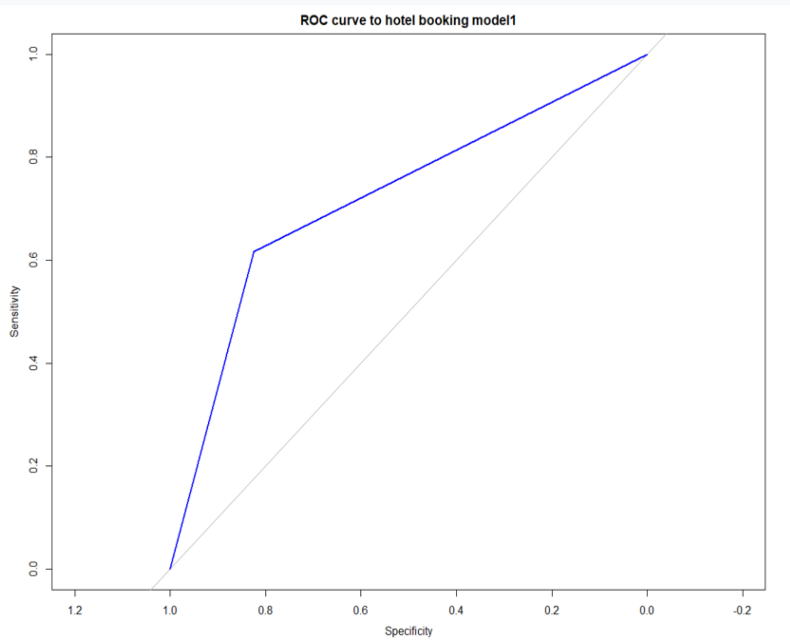
5.1 Logistic regression

I select many variables to train logistic regression model according to **correlation table** and **variable importance chart** in part 5.

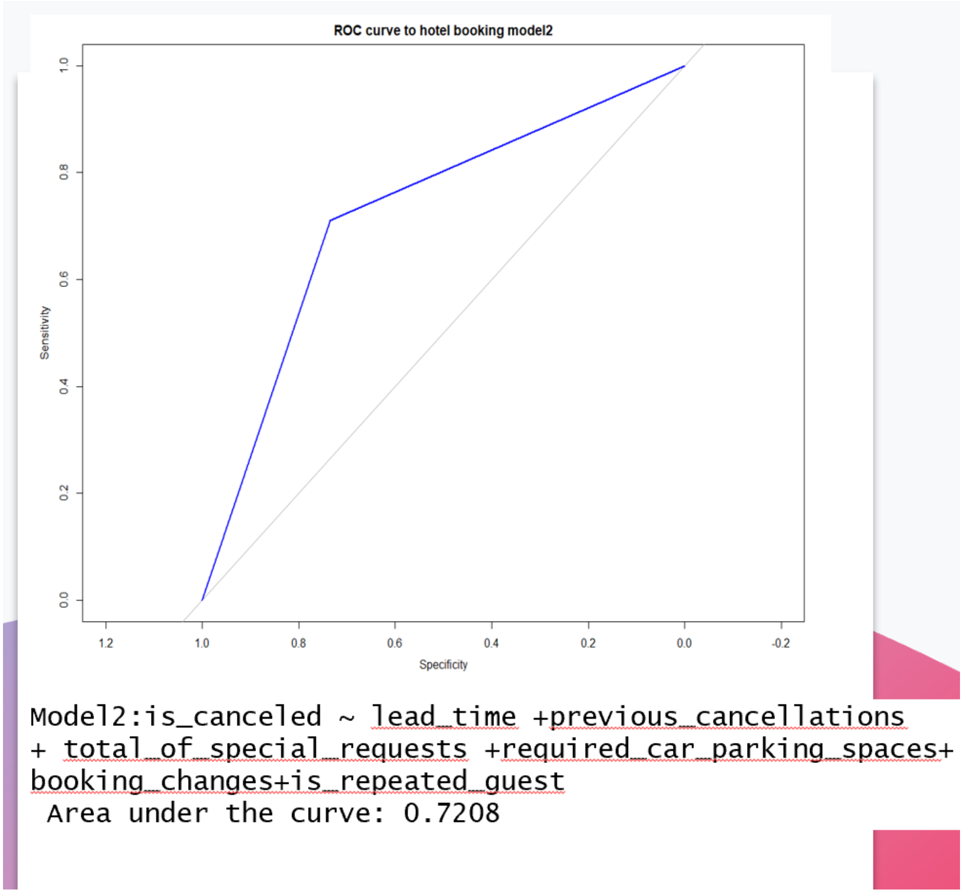
Logistic Regression Model 1: use lead_time +customer_type + hotel +deposit_type + adr +total_of_special_requests as my predict variables,

Logistic Regression Model 2: use lead_time +previous_cancellations + total_of_special_requests +required_car_parking_spaces+ booking_changes+is_repeated_guest as my predict variables, which are top 5 numerical variables in correlation table I just show you.

And I use **Roc** curve to assess the performance of logistic models. In **Logistic Regression Model 1** Area under the curve: 0.7228 and In **Logistic Regression Model 2** Area under the curve: 0.7208. The performance of the two models is similar, with only slight differences. The performance of the two models is similar, with only slight differences.




```
Model1:is_canceled ~lead_time +previous_cancellations
 + total_of_special_requests
+required_car_parking_spaces
+booking_changes+is_repeated_guest
 Area under the curve: 0.7228
```



5.2 Random Forest

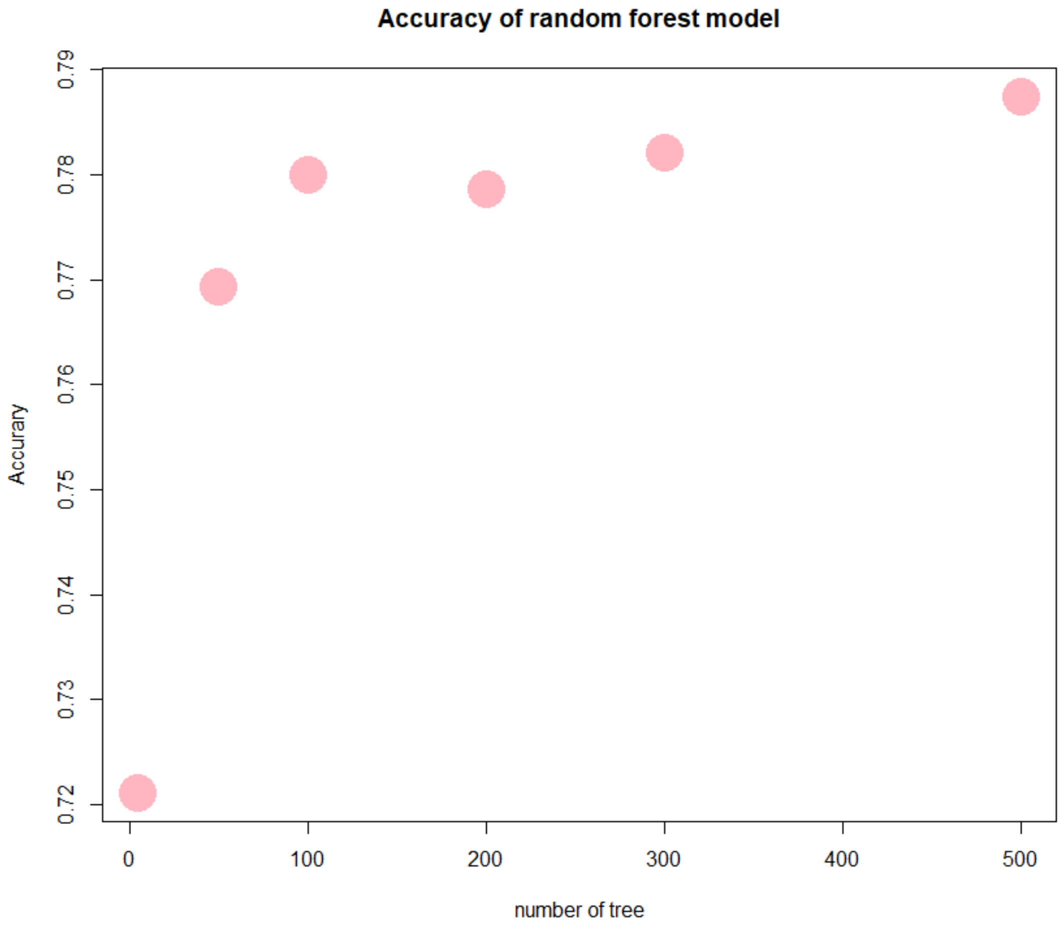
I drop some variables because in the R the random forest can't handle categorical predictors with more than 53 categories. (I mention this in part 4).

I select many variables to train **random forest model** according to **correlation table** and **variable importance chart** in part 5.

Then I started to use random forest to train model, and set the ntree in different value 5, 50, 100, 200, 300, 500 to get different 6 models.

```r
#random forest model1:
set.seed(2020)
RF_model1<- randomForest( is_canceled~.,data = data_RF, na.action=na.omit,ntree=5,importance=TRUE, proximity=TRUE,do.trace=T)
conf1 <- RF_model1$confusion
conf1
RF_model1$confusion[, 'class.error']
accuracy1 <- 1- mean(RF_model1$confusion[, 'class.error'])
accuracy1

#random forest model2:
set.seed(2020)
RF_model2<- randomForest( is_canceled~.,data = data_RF, na.action=na.omit,ntree=50,importance=TRUE, proximity=TRUE,do.trace=T)
conf2 <- RF_model2$confusion
conf2
accuracy2 <- 1- mean(RF_model2$confusion[, 'class.error'])
accuracy2

#random forest model3:
set.seed(2020)
RF_model3<- randomForest( is_canceled~.,data = data_RF, na.action=na.omit,ntree=100,importance=TRUE, proximity=TRUE,do.trace=T)
conf3 <- RF_model3$confusion
conf3
accuracy3 <- 1- mean(RF_model3$confusion[, 'class.error'])
accuracy3

#random forest model4:
set.seed(2020)
RF_model4<- randomForest( is_canceled~.,data = data_RF, na.action=na.omit,ntree=200,importance=TRUE, proximity=TRUE,do.trace=T)
conf4 <- RF_model4$confusion
conf4
accuracy4 <- 1- mean(RF_model4$confusion[, 'class.error'])
accuracy4



#random forest model5:
set.seed(2020)
RF_model5<- randomForest( is_canceled~.,data = data_RF, na.action=na.omit,ntree=300,importance=TRUE, proximity=TRUE,do.trace=T)
conf5 <- RF_model5$confusion
conf5
accuracy5 <- 1- mean(RF_model5$confusion[, 'class.error'])
accuracy5

#random forest model6:
set.seed(2020)
RF_model6<- randomForest( is_canceled~.,data = data_RF, na.action=na.omit,ntree=500,importance=TRUE, proximity=TRUE,do.trace=T)
conf6 <- RF_model6$confusion
conf6
accuracy6 <- 1- mean(RF_model6$confusion[, 'class.error'])
accuracy6
```

And I count the accuracy of these 6 models separately. And I create this scatterplot, form left plot and right table , we can know that when we set ntree equals to 100, the performance is good enough. The accurary of 100 tree model is 0.78, similar to the accuracy of 300 tree model and 500 tree model.



Values	
accuracy1	0.721099222154246
accuracy2	0.769322797373645
accuracy3	0.780038173766987
accuracy4	0.778693693693694
accuracy5	0.782072072072072
accuracy6	0.787484348755535

The increase of the number of Numbers is not significant to the improvement of the accuracy of model prediction, As the number of trees increases, the speed slows down, so 100 is enough So, in a word, the random forest model3 is the best model, which ntree is equal to 100.

6.Further Improvement

(1) This dataset has 32 variables(and using PCA to Reduce Dimension is a good next step.

(2) Combine k-fold cross validation to find out the tree number which can classify the response variable(is_canceled) most accurately.

(3) Country variable need fancy handle, one hot-encoding is a good choice to go further.

7.Conclusion

First, I show preprocessing and feature selection steps in model building processes in this report. The way to create a successful model is to get clean data.

total_of_special_requests +arrival_date_month+lead_time+required_car_parking_spaces are the most useful features to predict status of cancellation. The logistic regression model 2 and random forest model 3 are the best models.

The optimization of the model established afterwards and especially the problem of classification should not be overlooked the importance of recall values. The accuracy by class is one of the most critical points of classification problems.