#CAPSTONE MOVIELENS PROJECT

####by:reena2930
##############################

##Overview/Introduction

Hotel bookings are always in high demand and there are many times we hope we get booking in our favourite hotel or resort. What is surprising to notice after I took up this project to the number of booking cancellations. This is defenitely bad news for hotel business as leads to wastage of food, resources, time and all leading to financial challenges.

In this project I have aimed to develop an algorithm to predict booking cancellations based on the different features involved in the booking process. The data file has been downloaded from an existing available list of Kaggle datasets.

ref: https://www.kaggle.com/jessemostipak/hotel-booking-demand

This data set contains a single file which compares various booking information between two hotels: 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.The csv file has been downloaded and made available as an additional attachment with the report.

The dataset has been divided into 80-30 training and test sets and the final validation will be made on the test set.

##Methods and Analysis

The data set used is already in the clean form. I have attempted to format few columns, and remove few unwanted columns for easy analysis. ***PLEASE ATTACH THE SUPPORTING "hotel_bookingsCSV" file WHEN THE SYSTEM PROMPTS FOR THE CODING TO WORK***

###Dataset loading and creation of edx set and validation set

Note: This process could take a couple of minutes if you do not have the packages updated.

```
if(!require(tidyverse)) install.packages("tidyverse", repos =
"http://cran.us.r-project.org")

## Loading required package: tidyverse

## Warning: package 'tidyverse' was built under R version 3.6.3

## -- Attaching packages ---------------------------------------------------
-- tidyverse 1.3.0 --

## v ggplot2 3.3.0     v purrr   0.3.4
## v tibble  3.0.1     v dplyr   0.8.5
```

```
## v tidyr   1.0.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## Warning: package 'ggplot2' was built under R version 3.6.3

## Warning: package 'tibble' was built under R version 3.6.3

## Warning: package 'tidyr' was built under R version 3.6.3

## Warning: package 'readr' was built under R version 3.6.3

## Warning: package 'purrr' was built under R version 3.6.3

## Warning: package 'dplyr' was built under R version 3.6.3

## Warning: package 'stringr' was built under R version 3.6.3

## Warning: package 'forcats' was built under R version 3.6.3

## -- Conflicts ---------------------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-
project.org")

## Loading required package: caret

## Warning: package 'caret' was built under R version 3.6.3

## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift

if(!require(data.table)) install.packages("data.table", repos =
"http://cran.us.r-project.org")

## Loading required package: data.table

## Warning: package 'data.table' was built under R version 3.6.3

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## The following object is masked from 'package:purrr':
##
##     transpose

if(!require(dslabs)) install.packages("dslabs", repos = "http://cran.us.r-
project.org")

## Loading required package: dslabs

## Warning: package 'dslabs' was built under R version 3.6.3

if(!require(ggplot2)) install.packages("ggplot2", repos = "http://cran.us.r-
project.org")

#PLEASE ATTACH THE SUPPORTING hotel_bookingsCSV file WHEN THE SYSTME PROMPTS
FOR THE CODING TO WORK.


hotel_bookings<-read.csv(file.choose(), header=T)

head(hotel_bookings)

##          hotel is_canceled lead_time arrival_date_year arrival_date_month
## 1 Resort Hotel           0       342              2015               July
## 2 Resort Hotel           0       737              2015               July
## 3 Resort Hotel           0         7              2015               July
## 4 Resort Hotel           0        13              2015               July
## 5 Resort Hotel           0        14              2015               July
## 6 Resort Hotel           0        14              2015               July
##   arrival_date_week_number arrival_date_day_of_month
stays_in_weekend_nights
## 1                       27                         1
0
## 2                       27                         1
0
## 3                       27                         1
0
## 4                       27                         1
0
## 5                       27                         1
0
## 6                       27                         1
0
##   stays_in_week_nights adults children babies meal country market_segment
## 1                    0      2        0      0   BB     PRT         Direct
## 2                    0      2        0      0   BB     PRT         Direct
## 3                    1      1        0      0   BB     GBR         Direct
## 4                    1      1        0      0   BB     GBR      Corporate
## 5                    2      2        0      0   BB     GBR      Online TA
## 6                    2      2        0      0   BB     GBR      Online TA
##   distribution_channel is_repeated_guest previous_cancellations
## 1               Direct                 0                     0
```

```
## 2               Direct                 0                      0
## 3               Direct                 0                      0
## 4            Corporate                 0                      0
## 5                TA/TO                 0                      0
## 6                TA/TO                 0                      0
##   previous_bookings_not_canceled reserved_room_type assigned_room_type
## 1                              0                  C                  C
## 2                              0                  C                  C
## 3                              0                  A                  C
## 4                              0                  A                  A
## 5                              0                  A                  A
## 6                              0                  A                  A
##   booking_changes deposit_type agent company days_in_waiting_list
customer_type
## 1               3   No Deposit  NULL    NULL                    0
Transient
## 2               4   No Deposit  NULL    NULL                    0
Transient
## 3               0   No Deposit  NULL    NULL                    0
Transient
## 4               0   No Deposit   304    NULL                    0
Transient
## 5               0   No Deposit   240    NULL                    0
Transient
## 6               0   No Deposit   240    NULL                    0
Transient
##   adr required_car_parking_spaces total_of_special_requests
reservation_status
## 1   0                           0                         0
Check-Out
## 2   0                           0                         0
Check-Out
## 3  75                           0                         0
Check-Out
## 4  75                           0                         0
Check-Out
## 5  98                           0                         1
Check-Out
## 6  98                           0                         1
Check-Out
##   reservation_status_date
## 1              01-07-2015
## 2              01-07-2015
## 3              02-07-2015
## 4              02-07-2015
## 5              03-07-2015
## 6              03-07-2015
```

```r
#removing unwanted columns from the dataset
drop_cols = c("arrival_date_week_number", "is_repeated_guest",
              "previous_bookings_not_canceled", "adults", "children",
              "babies", "stays_in_weekend_nights", "stays_in_week_nights",
              "agent", "company", "days_in_waiting_list", "deposit_type",
              "total_of_special_requests", "reservation_status",
              "reservation_status_date", "country",
              "required_car_parking_spaces")

hoteldat<-hotel_bookings%>%select(-drop_cols)

## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(drop_cols)` instead of `drop_cols` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.



#dataset with revised number of columns
head(hoteldat)

##          hotel is_canceled lead_time arrival_date_year arrival_date_month
## 1 Resort Hotel           0       342              2015               July
## 2 Resort Hotel           0       737              2015               July
## 3 Resort Hotel           0         7              2015               July
## 4 Resort Hotel           0        13              2015               July
## 5 Resort Hotel           0        14              2015               July
## 6 Resort Hotel           0        14              2015               July
##   arrival_date_day_of_month meal market_segment distribution_channel
## 1                         1   BB         Direct               Direct
## 2                         1   BB         Direct               Direct
## 3                         1   BB         Direct               Direct
## 4                         1   BB      Corporate            Corporate
## 5                         1   BB      Online TA                TA/TO
## 6                         1   BB      Online TA                TA/TO
##   previous_cancellations reserved_room_type assigned_room_type
booking_changes
## 1                      0                  C                  C
3
## 2                      0                  C                  C
4
## 3                      0                  A                  C
0
## 4                      0                  A                  A
0
## 5                      0                  A                  A
0
## 6                      0                  A                  A
0
##   customer_type adr
## 1     Transient   0
```

```
## 2     Transient   0
## 3     Transient  75
## 4     Transient  75
## 5     Transient  98
## 6     Transient  98
```

```
#using as_factor to convert variables into a factor to preserve the value and
variable label attributes
hoteldat <- hoteldat %>%
  mutate(hotel = as_factor(hotel),
         is_canceled = as_factor(is_canceled),
         arrival_date_year = as_factor(arrival_date_year),
         arrival_date_month = as_factor(arrival_date_month),
         meal = as_factor(meal),
        market_segment = as_factor(market_segment),
        distribution_channel = as_factor(distribution_channel),
        previous_cancellations = as_factor(previous_cancellations),
        reserved_room_type = as_factor(reserved_room_type),
       assigned_room_type = as_factor(assigned_room_type))

#changing the 0 to No and 1 to Yes, for visualization purpose
hoteldat<-hoteldat%>%
  mutate(is_canceled = ifelse(str_detect(is_canceled,"0")==TRUE,"No","Yes"))

summary(hoteldat)
```

```
##           hotel       is_canceled          lead_time   arrival_date_year
##  City Hotel  :79330   Length:119390      Min.   :  0   2015:21996
##  Resort Hotel:40060   Class :character   1st Qu.: 18   2016:56707
##                       Mode  :character   Median : 69   2017:40687
##                                          Mean   :104
##                                          3rd Qu.:160
##                                          Max.   :737
##
##  arrival_date_month arrival_date_day_of_month        meal
##  August :13877      Min.   : 1.0              BB       :92310
##  July   :12661      1st Qu.: 8.0              FB       :  798
##  May    :11791      Median :16.0              HB       :14463
##  October:11160      Mean   :15.8              SC       :10650
##  April  :11089      3rd Qu.:23.0              Undefined: 1169
##  June   :10939      Max.   :31.0
##  (Other):47873
##        market_segment  distribution_channel previous_cancellations
##  Online TA    :56477   Corporate: 6677      0      :112906
##  Offline TA/TO:24219   Direct   :14645      1      :  6051
##  Groups       :19811   GDS      :  193      2      :   116
##  Direct       :12606   TA/TO    :97870      3      :    65
##  Corporate    : 5295   Undefined:    5      24     :    48
##  Complementary:  743                        11     :    35
##  (Other)      :  239                        (Other):   169
##  reserved_room_type assigned_room_type booking_changes
```

```
##  A      :85994      A      :74053      Min.   : 0.0000
##  D      :19201      D      :25322      1st Qu.: 0.0000
##  E      : 6535      E      : 7806      Median : 0.0000
##  F      : 2897      F      : 3751      Mean   : 0.2211
##  G      : 2094      G      : 2553      3rd Qu.: 0.0000
##  B      : 1118      C      : 2375      Max.   :21.0000
##  (Other): 1551      (Other): 3530
##          customer_type        adr
##  Contract       : 4076   Min.   :  -6.38
##  Group          :  577   1st Qu.:  69.29
##  Transient      :89613   Median :  94.58
##  Transient-Party:25124   Mean   : 101.83
##                          3rd Qu.: 126.00
##                          Max.   :5400.00
##
```

```
head(hoteldat)
```

```
##          hotel is_canceled lead_time arrival_date_year arrival_date_month
## 1 Resort Hotel          No       342              2015               July
## 2 Resort Hotel          No       737              2015               July
## 3 Resort Hotel          No         7              2015               July
## 4 Resort Hotel          No        13              2015               July
## 5 Resort Hotel          No        14              2015               July
## 6 Resort Hotel          No        14              2015               July
##   arrival_date_day_of_month meal market_segment distribution_channel
## 1                         1   BB         Direct               Direct
## 2                         1   BB         Direct               Direct
## 3                         1   BB         Direct               Direct
## 4                         1   BB      Corporate            Corporate
## 5                         1   BB      Online TA                TA/TO
## 6                         1   BB      Online TA                TA/TO
##   previous_cancellations reserved_room_type assigned_room_type
booking_changes
## 1                      0                  C                  C
3
## 2                      0                  C                  C
4
## 3                      0                  A                  C
0
## 4                      0                  A                  A
0
## 5                      0                  A                  A
0
## 6                      0                  A                  A
0
##   customer_type adr
## 1     Transient   0
```

```
## 2     Transient   0
## 3     Transient  75
## 4     Transient  75
## 5     Transient  98
## 6     Transient  98

dim(hoteldat)

## [1] 119390     15
```

The cleaned data now has 119390 rows and 15 columns.
Both the structure and head options show no missing values and NA's. We can now proceed with splitting the data into training and test sets, in the proportion of 80-20

```
set.seed(1, sample.kind="Rounding")

## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding'
sampler
## used

#we will use a data partition set with 80% training data and 20% test data
test_index <- createDataPartition(y = hoteldat$is_canceled, times = 1, p =
0.2, list = FALSE)
train_set<-hoteldat[-test_index,]
test_set<-hoteldat[test_index,]
```

##Data Exploration

```
summary(train_set)

##           hotel       is_canceled          lead_time     arrival_date_year
##  City Hotel  :63488   Length:95511       Min.   :  0.0   2015:17503
##  Resort Hotel:32023   Class :character   1st Qu.: 18.0   2016:45455
##                       Mode  :character   Median : 69.0   2017:32553
##                                          Mean   :104.1
##                                          3rd Qu.:160.0
##                                          Max.   :737.0
##
##  arrival_date_month arrival_date_day_of_month        meal
##  August :11125      Min.   : 1.00             BB       :73812
##  July   :10169      1st Qu.: 8.00             FB       :  630
##  May    : 9462      Median :16.00             HB       :11575
##  October: 8902      Mean   :15.81             SC       : 8574
##  April  : 8862      3rd Qu.:23.00             Undefined:  920
##  June   : 8717      Max.   :31.00
##  (Other):38274
##        market_segment  distribution_channel previous_cancellations
##  Online TA    :45242   Corporate: 5321      0      :90267
##  Offline TA/TO:19409   Direct   :11660      1      : 4888
##  Groups       :15799   GDS      :  154      2      :   95
##  Direct       :10021   TA/TO    :78371      3      :   51
```

```
##  Corporate    : 4250   Undefined:    5      24     :   39
##  Complementary:  597                        11     :   33
##  (Other)      :  193                        (Other):  138
##  reserved_room_type assigned_room_type booking_changes
##  A      :68790      A      :59313      Min.   : 0.0000
##  D      :15384      D      :20194      1st Qu.: 0.0000
##  E      : 5213      E      : 6253      Median : 0.0000
##  F      : 2302      F      : 2986      Mean   : 0.2223
##  G      : 1665      G      : 2031      3rd Qu.: 0.0000
##  B      :  916      C      : 1906      Max.   :21.0000
##  (Other): 1241      (Other): 2828
##          customer_type        adr
##  Contract       : 3215   Min.   :  -6.38
##  Group          :  460   1st Qu.:  69.29
##  Transient      :71660   Median :  94.50
##  Transient-Party:20176   Mean   : 101.80
##                          3rd Qu.: 126.00
##                          Max.   :5400.00
##
```

The summary of the subset shows that the edx set has 95511 observations with 15 variables and there are no missing values or NA. Each feature represents individual column in the dataset. The test set has the same features except that its 20% of the total dataset.

```
train_set%>%group_by(is_canceled)%>%summarize(n=n())
```

```
## # A tibble: 2 x 2
##   is_canceled     n
##   <chr>       <int>
## 1 No          60132
## 2 Yes         35379
```

```
head(train_set)
```

```
##          hotel is_canceled lead_time arrival_date_year arrival_date_month
## 1 Resort Hotel          No       342              2015               July
## 2 Resort Hotel          No       737              2015               July
## 3 Resort Hotel          No         7              2015               July
## 4 Resort Hotel          No        13              2015               July
## 5 Resort Hotel          No        14              2015               July
## 6 Resort Hotel          No        14              2015               July
##   arrival_date_day_of_month meal market_segment distribution_channel
## 1                         1   BB         Direct               Direct
## 2                         1   BB         Direct               Direct
## 3                         1   BB         Direct               Direct
## 4                         1   BB      Corporate            Corporate
## 5                         1   BB      Online TA                TA/TO
## 6                         1   BB      Online TA                TA/TO
##   previous_cancellations reserved_room_type assigned_room_type
booking_changes
## 1                      0                  C                  C
```

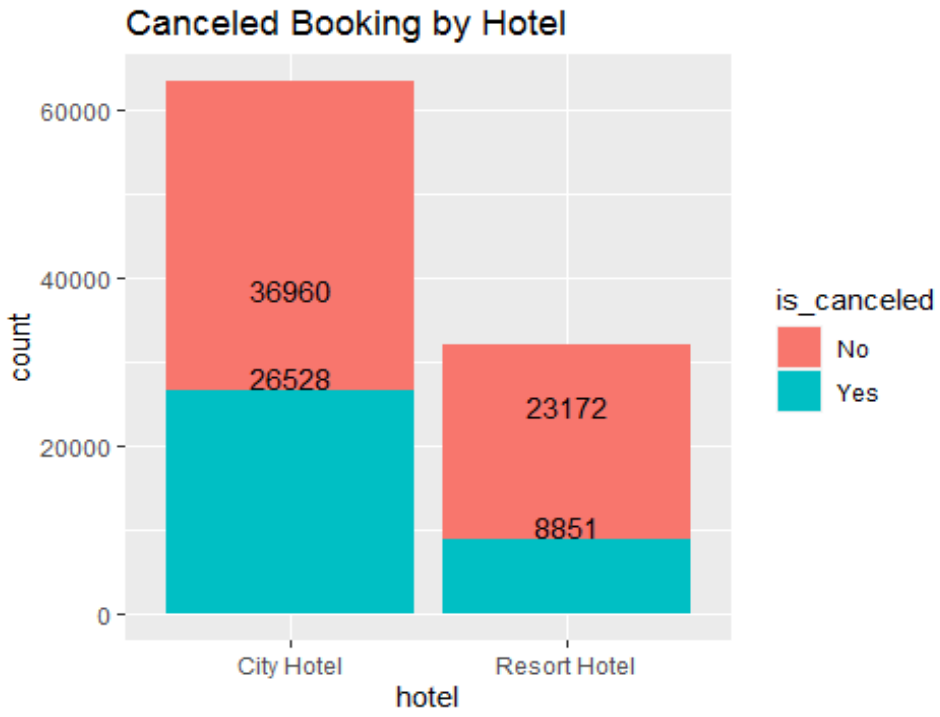
```
3
## 2                     0                 C                 C
4
## 3                     0                 A                 C
0
## 4                     0                 A                 A
0
## 5                     0                 A                 A
0
## 6                     0                 A                 A
0
##   customer_type adr
## 1     Transient   0
## 2     Transient   0
## 3     Transient  75
## 4     Transient  75
## 5     Transient  98
## 6     Transient  98
```

Above is a glimpse of how the data looks. Each row represents data realted to the hotel booking. The “is_canceled” column is the outcome(y) we want to predict. Let us now look at the basic features and characteristics of the datasets

Proportion of bookings that were canceled with both Hotels

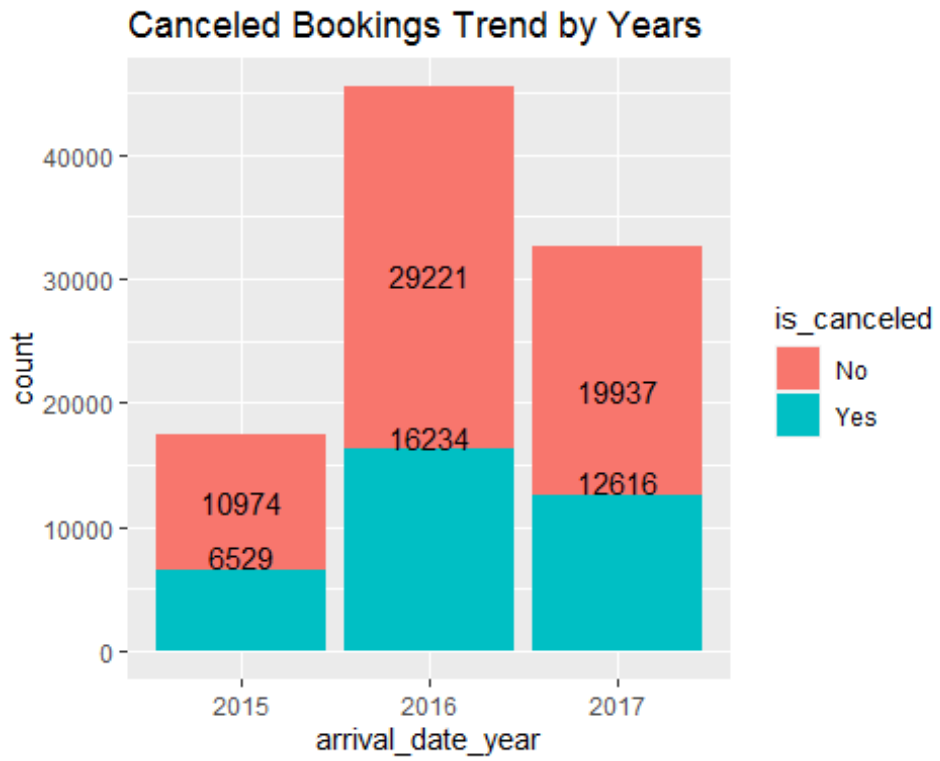
City Hotel seems to have more cancellations than the Resort hotel

```
#Proportion of bookings that were canceled
train_set%>%ggplot(aes(x=hotel,
fill=is_canceled))+geom_bar()+ggtitle("Canceled Booking by
Hotel")+geom_text(stat = "count", aes(label=..count..),vjust=-0.1)
```



Bookings canceled between 2015-2017

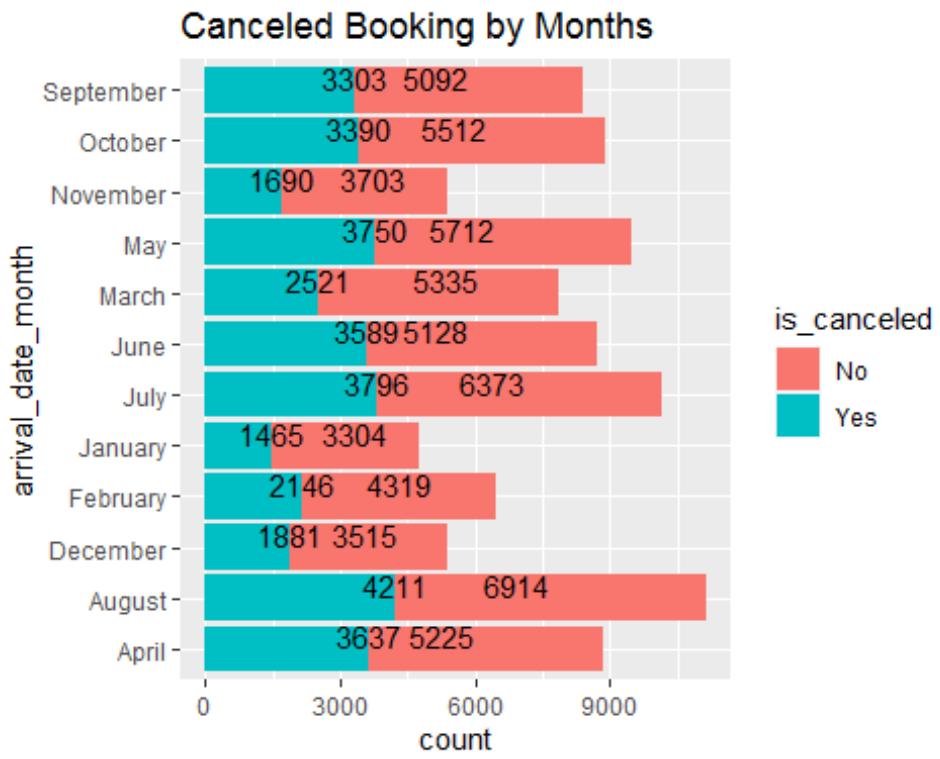
```
train_set%>%ggplot(aes(x=arrival_date_year,
fill=is_canceled))+geom_bar()+ggtitle("Canceled Bookings Trend by
Years")+geom_text(stat = "count", aes(label=..count..),vjust=-0.05)
```



Cancellations were high in 2016 and reduced in 2017, which is still higher than 2015. 2015 shows the lowest cancellations off the 3 years.

Monthly Booking cancellations

```
train_set%>%ggplot(aes(x=arrival_date_month,
fill=is_canceled))+geom_bar()+coord_flip()+ggtitle("Canceled Booking by
Months")+geom_text(stat = "count", aes(label=..count..),vjust=-0.05)
```

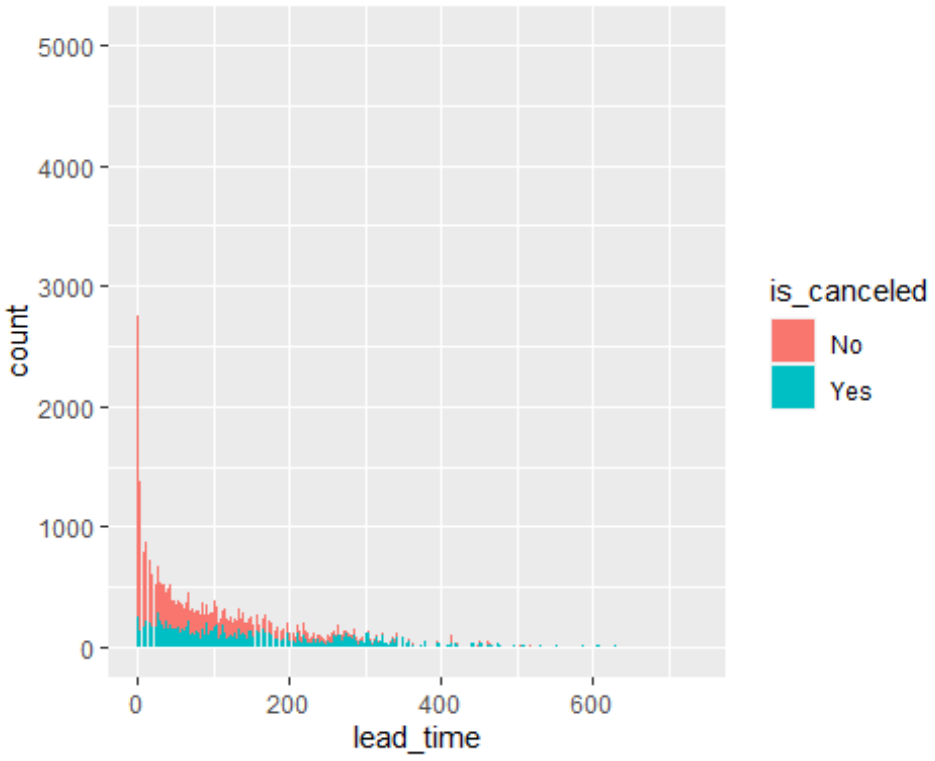


August shows the highest cancellations followed by April, July, June and May. Hotel administration will need to look into the reasons behind this peak in Aug and make necessary changes to boost non cancellations

###Bookings canceled based lead time

Lead Time is the period of time between when a guest makes a reservation, and the actual check-in date.

```
train_set%>%ggplot(aes(x=lead_time,
fill=is_canceled))+geom_bar()+ggtitle("Canceled Booking by Lead Time")
```

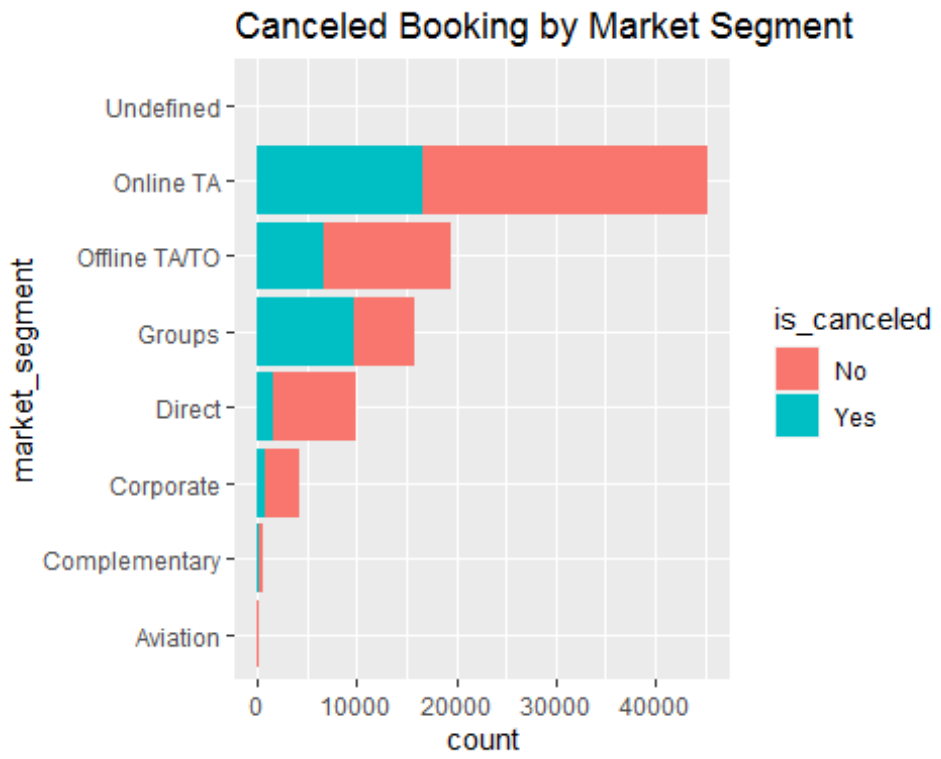


There are not many cancellations after the lead time has elapsed. cancellations soon after booking seem to be a trend.

###Bookings canceled based Market segment

Most of the time bookings via different market segments bring in more business then direct hotel bookings. Let us look at what the trend shows here

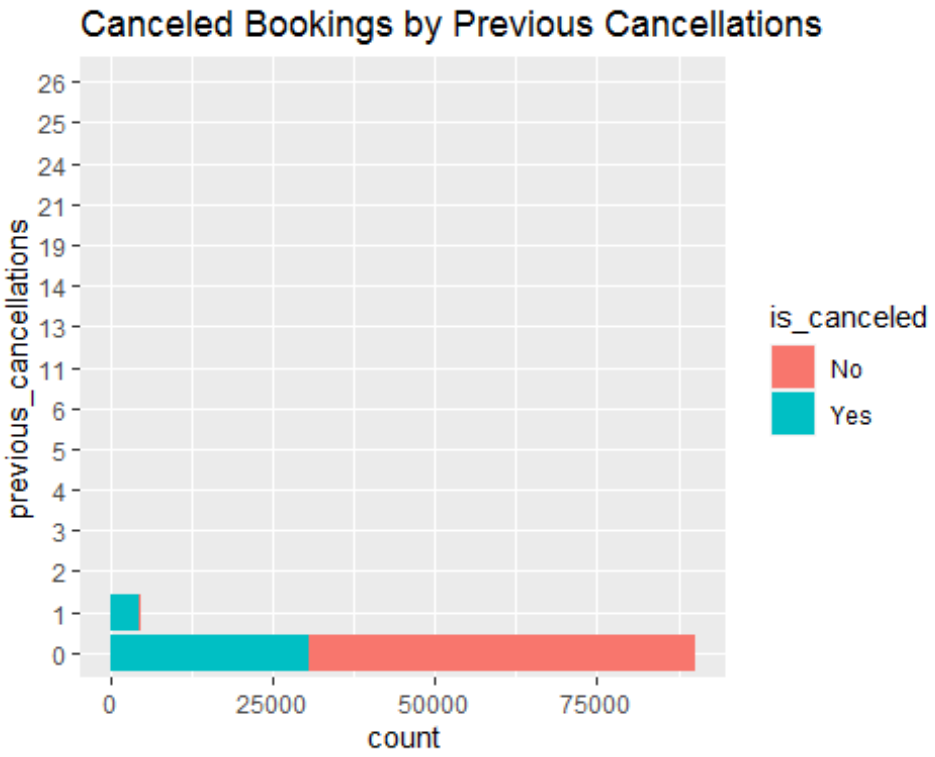
```
train_set%>%ggplot(aes(x=market_segment,
fill=is_canceled))+geom_bar()+coord_flip()+ggtitle("Canceled Booking by
Market Segment")
```



This definitely needs to be looked into by the Hotel administration. Bookings ,made by TA's and TO's via online medium show highest cancellations, though their bookings are good. Group bookings, which bring in more revenue, are second highest in cancellations.

###Bookings canceled based on Previous Cancellations

```
train_set%>%ggplot(aes(x=previous_cancellations,
fill=is_canceled))+geom_bar()+coord_flip()+ggtitle("Canceled Bookings by
Previous Cancellations")
```



Cancellations in rationship to previous cancellations are steep, which is obvious considering the history.

##Model Preparation

###LDA model A relatively simple solution to the problem of having too many parameters is to assume that the correlation structure is the same for all classes, which reduces the number of parameters we need to estimate. We can fit the LDA model using caret. One model is based only using train_set and the second model is used on the test_set

```
#LDA model with train set
set.seed(1, sample.kind = "Rounding")

## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding'
sampler
## used

train_lda <- train(is_canceled ~ arrival_date_year + lead_time + adr +
arrival_date_month + booking_changes, method = "lda", data = train_set)
lda_preds <- predict(train_lda, train_set)
confusionMatrix(data=lda_preds,
reference=factor(train_set$is_canceled))$overall["Accuracy"]
```

```
##  Accuracy
## 0.6815131

#LDA model with test set
test_lda <- train(is_canceled ~ arrival_date_year + lead_time + adr +
arrival_date_month + booking_changes, method = "lda", data = train_set)
lda_preds <- predict(test_lda, test_set)
confusionMatrix(data=lda_preds,
reference=factor(test_set$is_canceled))$overall["Accuracy"]

##  Accuracy
## 0.6809749
```

LDA method	Accuracy
Training Set	0.6815131
Test Set	0.6809749

##Logistic regression model

The simplest prediction method is randomly guessing the outcome without using additional predictors. These methods will help us determine whether our machine learning algorithm performs better than chance.

test_glm<-train(is_canceled~., method="glm", data=test_set)

We will use the glm method with few variations, with both train set and test set

##1st glm model

```
#1st glm model on train set
set.seed(1, sample.kind = "Rounding")

## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding'
sampler
## used

train_glm<- train(is_canceled ~ arrival_date_year + lead_time + adr, method =
"glm", data = train_set)
glm_pred <- predict(train_glm, train_set)
confusionMatrix(data=glm_pred,
reference=factor(train_set$is_canceled))$overall["Accuracy"]

##  Accuracy
## 0.6641643

#1st glm model on test set
test_glm<- train(is_canceled ~ arrival_date_year + lead_time + adr,
method = "glm", data = train_set)
```

```r
glm_pred <- predict(test_glm, test_set)
confusionMatrix(data=glm_pred,
reference=factor(test_set$is_canceled))$overall["Accuracy"]
```

```
##  Accuracy
## 0.6645588
```

GLM Method with 3 predictors

GLM method	Accuracy
Training Set	0.6641643
Test Set	0.6645588

2nd glm model

```r
#2nd glm model on train set
set.seed(1, sample.kind = "Rounding")
```

```
## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding'
sampler
## used
```

```r
train_glm2<- train(is_canceled ~ arrival_date_year + lead_time + adr +
arrival_date_month + booking_changes, method = "glm", data = train_set)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
glm_pred2 <- predict(train_glm2, train_set)
confusionMatrix(data=glm_pred2,
reference=factor(train_set$is_canceled))$overall["Accuracy"]
```

```
##  Accuracy
## 0.6844761
```

```
#2nd glm model on test set
test_glm2<- train(is_canceled ~ arrival_date_year + lead_time + adr +
arrival_date_month + booking_changes, method = "glm", data = train_set)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
glm_pred2<- predict(test_glm2, test_set)
confusionMatrix(data=glm_pred2,
reference=factor(test_set$is_canceled))$overall["Accuracy"]
```

```
##  Accuracy
## 0.6837389
```

GLM Method with 5 predictors

GLM method2	Accuracy
Training Set	0.6645588
Test Set	0.6837389

##RESULT

The model is able to predict cancellations with an accuracy of 68%. The accuracy levels with both LDA and glm were 68% but glm method came with warning notes. LDA model fits our requirement for the requirement of this project.

LDA method	**Accuracy**
Training Set	0.6815131
Test Set	0.6809749

The bookings and cancellations were on higher side on the City Hotel compared to Resort Hotel and Online bookings by TA's and Group cancellations, combined by the tradition of previous cancellation trend and lead time contributed towards cancellation more than other predictors in the data set.

##CONCLUSION

It can be concluded that our recommended LDA model is able to predict the cancellation of hotel booking by 68% considering the few predictors used. However, the data can be formatted further and similar columns combined, like family numbers, booking segments

etc., and more features explored to get a better accuracy level. We also can look at more advanced regression techniques to achieve better levels of accuracy.

******End*******