Hotel Pricing Capstone Project

Abhranil Bhattacharjee

February 22, 2018

## Email: [abhra.tb@gmail.com](mailto:abhra.tb@gmail.com)

## College: Jadavpur University, Kolkata.

## INTRODUCTION

The purpose of this project is to analyze the pricing strategy of hotels in the Indian hotel industry.Many factors drive hotel room prices which are primarily of two types: external and internal. The objective of this project is to identify the factors that matter the most.

## LITERATURE REVIEW

The particular objective of this Study was to analyze the pricing strategy of hotels in the Indian hotel industry.Our aim was to compare the room rents of hotels for 8 days in 42 cities with the description provided in the dataset.Based on them we did some T-tests and correlation test to identify the significance of different variables and that which variable(s) acted as driving force to any changes in the same.We tried to explore the differences in price based on weekends(if the date lies on Saturday or Sunday), holidays(New Year Eve), Metro city, distance from the airport, the presence of swimming pools, Free wifi, free breakfast and so on.The tests helped us determine that which factor(s) related the price of hotel rooms positively and which affect negatively. Based on the tests, we also fitted in a regression model with respect to the variables which may be used to predict the room rent of the hotels or in short, the more significant variables.

## DATA DESCRIPTION

**DEPENDENT VARIABLE**

ROOMRENT- Rent for the cheapest room, double occupancy, in Indian Rupees.

Some hotels have more than one type of double occupancy room. For simplicity, we picked the cheapest room with double occupancy.

**EXTERNAL FACTORS**

DATE- We have hotel room rent data for the following 8 dates for each hotel: {Dec 31, Dec 25, Dec 24, Dec 18, Dec 21, Dec 28, Jan 4, Jan 8} If a hotel is sold out on a given date, assume that the price of the hotel room on the date it is sold out is the maximum price from the sample of dates for which prices are available.

IsWeekend-We use ‘0’ to indicate week days, ‘1’ to indicate weekend dates (Sat / Sun)

IsNewYearEve-‘1’ for Dec 31, ‘0’ otherwise

CityName-Name of the City where the Hotel is located e.g. Mumbai

Population-Population of the City in 2011.

CityRank-Rank order of City by Population (e.g. Mumbai = 0, Delhi = 1, so on.

IsMetroCity-‘1’ if CityName is {Mumbai, Delhi, Kolkatta, Chennai}, ‘0’ otherwise.

IsTouristDestination-We use ‘1’ if the city is primarily a tourist destination, ‘0’ otherwise. For example, Goa and Agra are primarily tourist destinations. We assume that most people who visit Goa and Agra and stay in their hotels are in these cities primarily for tourism.

**INTERNAL FACTORS**

HotelName- e.g. Park Hyatt Goa Resort and Spa

StarRating- e.g. 5

AIRPORT- Distance between Hotel and closest major Airport

HotelAddress- e.g. Arrossim Beach, Cansaulim, Goa

HotelPincode- e.g . 403712

HotelDescription- e.g. 5-star beachfront resort with spa, near Arossim Beach

FreeWifi- ‘1’ if the hotel offers Free Wifi, ‘0’ otherwise

FreeBreakfast- ‘1’ if the hotel offers Free Breakfast, ‘0’ otherwise

HotelCapacity- e.g. 242. (enter ‘0’ if not available)

HasSwimmingPool- ‘1’ if they have a swimming pool, ‘0’ otherwise

## MODEL ANALYSIS

The dataset was read into R. The data was summarized to understand the mean, median, standard deviation of eachvariable. The problem was formulated as Y = F(x1, x2, x3..)The Dependent Variable(s) (i.e. the Y in the Y = F(x)) in the Dataset was identified as RoomRent.

hotelprice=read.csv(paste("Cities42.csv",sep=""))  
summary(lm(RoomRent ~HasSwimmingPool+IsNewYearEve+IsTouristDestination+IsMetroCity+StarRating+Airport+HotelCapacity+FreeWifi+FreeBreakfast, data = hotelprice))

##   
## Call:  
## lm(formula = RoomRent ~ HasSwimmingPool + IsNewYearEve + IsTouristDestination +   
## IsMetroCity + StarRating + Airport + HotelCapacity + FreeWifi +   
## FreeBreakfast, data = hotelprice)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -11696 -2375 -701 1063 309539   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -8906.418 405.396 -21.970 < 2e-16 \*\*\*  
## HasSwimmingPool 2227.069 159.327 13.978 < 2e-16 \*\*\*  
## IsNewYearEve 844.123 174.046 4.850 1.25e-06 \*\*\*  
## IsTouristDestination 2113.725 134.336 15.735 < 2e-16 \*\*\*  
## IsMetroCity -1548.328 138.527 -11.177 < 2e-16 \*\*\*  
## StarRating 3564.570 110.489 32.262 < 2e-16 \*\*\*  
## Airport 11.265 2.710 4.157 3.24e-05 \*\*\*  
## HotelCapacity -10.990 1.026 -10.714 < 2e-16 \*\*\*  
## FreeWifi 485.597 224.134 2.167 0.0303 \*   
## FreeBreakfast 182.992 123.296 1.484 0.1378   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6607 on 13222 degrees of freedom  
## Multiple R-squared: 0.1887, Adjusted R-squared: 0.1881   
## F-statistic: 341.7 on 9 and 13222 DF, p-value: < 2.2e-16

cities.df <- read.csv(paste(file = "Cities42.csv", sep = " "))  
View(cities.df)

## APPENDICES

library(psych)  
describe(cities.df)

## vars n mean sd median trimmed  
## CityName\* 1 13232 18.07 11.72 16 17.29  
## Population 2 13232 4416836.87 4258386.00 3046163 4040816.22  
## CityRank 3 13232 14.83 13.51 9 13.30  
## IsMetroCity 4 13232 0.28 0.45 0 0.23  
## IsTouristDestination 5 13232 0.70 0.46 1 0.75  
## IsWeekend 6 13232 0.62 0.48 1 0.65  
## IsNewYearEve 7 13232 0.12 0.33 0 0.03  
## Date\* 8 13232 14.30 2.69 14 14.39  
## HotelName\* 9 13232 841.19 488.16 827 841.18  
## RoomRent 10 13232 5473.99 7333.12 4000 4383.33  
## StarRating 11 13232 3.46 0.76 3 3.40  
## Airport 12 13232 21.16 22.76 15 16.39  
## HotelAddress\* 13 13232 1202.53 582.17 1261 1233.25  
## HotelPincode 14 13232 397430.26 259837.50 395003 388540.47  
## HotelDescription\* 15 13224 581.34 363.26 567 575.37  
## FreeWifi 16 13232 0.93 0.26 1 1.00  
## FreeBreakfast 17 13232 0.65 0.48 1 0.69  
## HotelCapacity 18 13232 62.51 76.66 34 46.03  
## HasSwimmingPool 19 13232 0.36 0.48 0 0.32  
## mad min max range skew  
## CityName\* 11.86 1.0 42 41.0 0.48  
## Population 3846498.95 8096.0 12442373 12434277.0 0.68  
## CityRank 11.86 0.0 44 44.0 0.69  
## IsMetroCity 0.00 0.0 1 1.0 0.96  
## IsTouristDestination 0.00 0.0 1 1.0 -0.86  
## IsWeekend 0.00 0.0 1 1.0 -0.51  
## IsNewYearEve 0.00 0.0 1 1.0 2.28  
## Date\* 2.97 1.0 20 19.0 -0.77  
## HotelName\* 641.97 1.0 1670 1669.0 0.01  
## RoomRent 2653.85 299.0 322500 322201.0 16.75  
## StarRating 0.74 0.0 5 5.0 0.48  
## Airport 11.12 0.2 124 123.8 2.73  
## HotelAddress\* 668.65 1.0 2108 2107.0 -0.37  
## HotelPincode 257975.37 100025.0 7000157 6900132.0 9.99  
## HotelDescription\* 472.95 1.0 1226 1225.0 0.11  
## FreeWifi 0.00 0.0 1 1.0 -3.25  
## FreeBreakfast 0.00 0.0 1 1.0 -0.62  
## HotelCapacity 28.17 0.0 600 600.0 2.95  
## HasSwimmingPool 0.00 0.0 1 1.0 0.60  
## kurtosis se  
## CityName\* -0.88 0.10  
## Population -1.08 37019.65  
## CityRank -0.76 0.12  
## IsMetroCity -1.08 0.00  
## IsTouristDestination -1.26 0.00  
## IsWeekend -1.74 0.00  
## IsNewYearEve 3.18 0.00  
## Date\* 1.92 0.02  
## HotelName\* -1.25 4.24  
## RoomRent 582.06 63.75  
## StarRating 0.25 0.01  
## Airport 7.89 0.20  
## HotelAddress\* -0.88 5.06  
## HotelPincode 249.76 2258.86  
## HotelDescription\* -1.25 3.16  
## FreeWifi 8.57 0.00  
## FreeBreakfast -1.61 0.00  
## HotelCapacity 11.39 0.67  
## HasSwimmingPool -1.64 0.00

## Dataset

dim(cities.df)

## [1] 13232 19

## Visualization by city name.

table(cities.df$CityName)

##   
## Agra Ahmedabad Amritsar Bangalore   
## 432 424 136 656   
## Bhubaneswar Chandigarh Chennai Darjeeling   
## 120 336 416 136   
## Delhi Gangtok Goa Guwahati   
## 2048 128 624 48   
## Haridwar Hyderabad Indore Jaipur   
## 48 536 160 768   
## Jaisalmer Jodhpur Kanpur Kochi   
## 264 224 16 608   
## Kolkata Lucknow Madurai Manali   
## 512 128 112 288   
## Mangalore Mumbai Munnar Mysore   
## 104 712 328 160   
## Nainital Ooty Panchkula Pune   
## 144 136 64 600   
## Puri Rajkot Rishikesh Shimla   
## 56 128 88 280   
## Srinagar Surat Thiruvanthipuram Thrissur   
## 40 80 392 32   
## Udaipur Varanasi   
## 456 264

## one-way contingency tables:

mytable2<-with(cities.df,table(IsTouristDestination))  
View(mytable2)  
round(prop.table(mytable2)\*100,2)

## IsTouristDestination  
## 0 1   
## 30.28 69.72

mytable3<-with(cities.df,table(IsWeekend))  
View(mytable3)  
round(prop.table(mytable3)\*100,2)

## IsWeekend  
## 0 1   
## 37.72 62.28

mytable4<-with(cities.df,table(IsNewYearEve))  
View(mytable4)  
round(prop.table(mytable4)\*100,2)

## IsNewYearEve  
## 0 1   
## 87.56 12.44

mytable5<-with(cities.df,table(StarRating))  
View(mytable5)  
round(prop.table(mytable5)\*100,2)

## StarRating  
## 0 1 2 2.5 3 3.2 3.3 3.4 3.5 3.6 3.7 3.8   
## 0.12 0.06 3.33 4.78 44.99 0.06 0.12 0.06 13.24 0.06 0.18 0.12   
## 3.9 4 4.1 4.3 4.4 4.5 4.7 4.8 5   
## 0.24 18.61 0.18 0.12 0.06 2.84 0.06 0.12 10.64

mytable6<-with(cities.df,table(FreeWifi))  
View(mytable6)  
round(prop.table(mytable6)\*100,2)

## FreeWifi  
## 0 1   
## 7.41 92.59

mytable7<-with(cities.df,table(FreeBreakfast))  
View(mytable7)  
round(prop.table(mytable7)\*100,2)

## FreeBreakfast  
## 0 1   
## 35.09 64.91

mytable8<-with(cities.df,table(HasSwimmingPool))  
View(mytable8)  
round(prop.table(mytable8)\*100,2)

## HasSwimmingPool  
## 0 1   
## 64.42 35.58

mytable9<-with(cities.df,table(CityName))  
View(mytable9)  
round(prop.table(mytable9)\*100,2)

## CityName  
## Agra Ahmedabad Amritsar Bangalore   
## 3.26 3.20 1.03 4.96   
## Bhubaneswar Chandigarh Chennai Darjeeling   
## 0.91 2.54 3.14 1.03   
## Delhi Gangtok Goa Guwahati   
## 15.48 0.97 4.72 0.36   
## Haridwar Hyderabad Indore Jaipur   
## 0.36 4.05 1.21 5.80   
## Jaisalmer Jodhpur Kanpur Kochi   
## 2.00 1.69 0.12 4.59   
## Kolkata Lucknow Madurai Manali   
## 3.87 0.97 0.85 2.18   
## Mangalore Mumbai Munnar Mysore   
## 0.79 5.38 2.48 1.21   
## Nainital Ooty Panchkula Pune   
## 1.09 1.03 0.48 4.53   
## Puri Rajkot Rishikesh Shimla   
## 0.42 0.97 0.67 2.12   
## Srinagar Surat Thiruvanthipuram Thrissur   
## 0.30 0.60 2.96 0.24   
## Udaipur Varanasi   
## 3.45 2.00

## Two-way contingency tables:

library(gmodels)  
CrossTable(cities.df$IsMetroCity,cities.df$IsTouristDestination)

##   
##   
## Cell Contents  
## |-------------------------|  
## | N |  
## | Chi-square contribution |  
## | N / Row Total |  
## | N / Col Total |  
## | N / Table Total |  
## |-------------------------|  
##   
##   
## Total Observations in Table: 13232   
##   
##   
## | cities.df$IsTouristDestination   
## cities.df$IsMetroCity | 0 | 1 | Row Total |   
## ----------------------|-----------|-----------|-----------|  
## 0 | 3352 | 6120 | 9472 |   
## | 81.543 | 35.419 | |   
## | 0.354 | 0.646 | 0.716 |   
## | 0.837 | 0.663 | |   
## | 0.253 | 0.463 | |   
## ----------------------|-----------|-----------|-----------|  
## 1 | 655 | 3105 | 3760 |   
## | 205.419 | 89.226 | |   
## | 0.174 | 0.826 | 0.284 |   
## | 0.163 | 0.337 | |   
## | 0.050 | 0.235 | |   
## ----------------------|-----------|-----------|-----------|  
## Column Total | 4007 | 9225 | 13232 |   
## | 0.303 | 0.697 | |   
## ----------------------|-----------|-----------|-----------|  
##   
##

CrossTable(cities.df$IsWeekend,cities.df$IsNewYearEve)

##   
##   
## Cell Contents  
## |-------------------------|  
## | N |  
## | Chi-square contribution |  
## | N / Row Total |  
## | N / Col Total |  
## | N / Table Total |  
## |-------------------------|  
##   
##   
## Total Observations in Table: 13232   
##   
##   
## | cities.df$IsNewYearEve   
## cities.df$IsWeekend | 0 | 1 | Row Total |   
## --------------------|-----------|-----------|-----------|  
## 0 | 4989 | 2 | 4991 |   
## | 87.637 | 616.864 | |   
## | 1.000 | 0.000 | 0.377 |   
## | 0.431 | 0.001 | |   
## | 0.377 | 0.000 | |   
## --------------------|-----------|-----------|-----------|  
## 1 | 6597 | 1644 | 8241 |   
## | 53.075 | 373.592 | |   
## | 0.801 | 0.199 | 0.623 |   
## | 0.569 | 0.999 | |   
## | 0.499 | 0.124 | |   
## --------------------|-----------|-----------|-----------|  
## Column Total | 11586 | 1646 | 13232 |   
## | 0.876 | 0.124 | |   
## --------------------|-----------|-----------|-----------|  
##   
##

CrossTable(cities.df$FreeWifi,cities.df$FreeBreakfast)

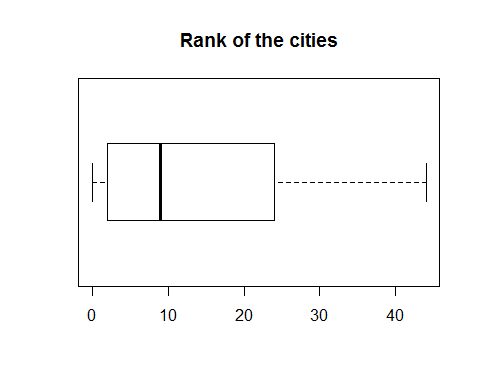
##   
##   
## Cell Contents  
## |-------------------------|  
## | N |  
## | Chi-square contribution |  
## | N / Row Total |  
## | N / Col Total |  
## | N / Table Total |  
## |-------------------------|  
##   
##   
## Total Observations in Table: 13232   
##   
##   
## | cities.df$FreeBreakfast   
## cities.df$FreeWifi | 0 | 1 | Row Total |   
## -------------------|-----------|-----------|-----------|  
## 0 | 606 | 375 | 981 |   
## | 199.074 | 107.614 | |   
## | 0.618 | 0.382 | 0.074 |   
## | 0.131 | 0.044 | |   
## | 0.046 | 0.028 | |   
## -------------------|-----------|-----------|-----------|  
## 1 | 4037 | 8214 | 12251 |   
## | 15.941 | 8.617 | |   
## | 0.330 | 0.670 | 0.926 |   
## | 0.869 | 0.956 | |   
## | 0.305 | 0.621 | |   
## -------------------|-----------|-----------|-----------|  
## Column Total | 4643 | 8589 | 13232 |   
## | 0.351 | 0.649 | |   
## -------------------|-----------|-----------|-----------|  
##   
##

CrossTable(cities.df$FreeWifi,cities.df$HasSwimmingPool)

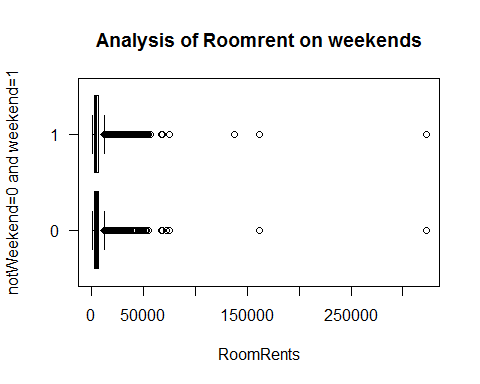
##   
##   
## Cell Contents  
## |-------------------------|  
## | N |  
## | Chi-square contribution |  
## | N / Row Total |  
## | N / Col Total |  
## | N / Table Total |  
## |-------------------------|  
##   
##   
## Total Observations in Table: 13232   
##   
##   
## | cities.df$HasSwimmingPool   
## cities.df$FreeWifi | 0 | 1 | Row Total |   
## -------------------|-----------|-----------|-----------|  
## 0 | 592 | 389 | 981 |   
## | 2.526 | 4.574 | |   
## | 0.603 | 0.397 | 0.074 |   
## | 0.069 | 0.083 | |   
## | 0.045 | 0.029 | |   
## -------------------|-----------|-----------|-----------|  
## 1 | 7932 | 4319 | 12251 |   
## | 0.202 | 0.366 | |   
## | 0.647 | 0.353 | 0.926 |   
## | 0.931 | 0.917 | |   
## | 0.599 | 0.326 | |   
## -------------------|-----------|-----------|-----------|  
## Column Total | 8524 | 4708 | 13232 |   
## | 0.644 | 0.356 | |   
## -------------------|-----------|-----------|-----------|  
##   
##

## Boxplots

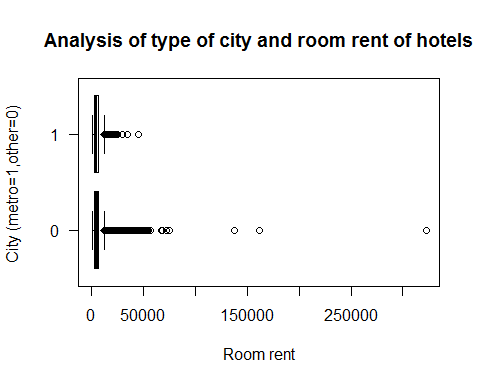
boxplot(cities.df$CityRank , horizontal =TRUE,main="Rank of the cities")



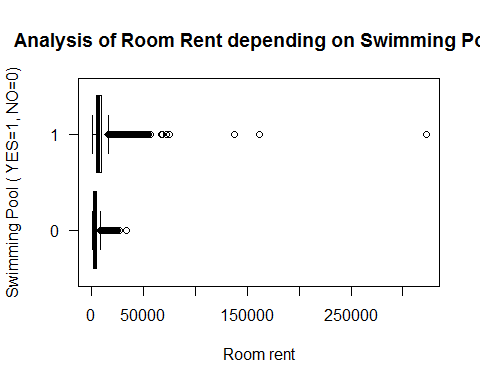
boxplot(cities.df$RoomRent ~ cities.df$IsWeekend, horizontal=TRUE,ylab=" notWeekend=0 and weekend=1 ", xlab="RoomRents", las=1,main="Analysis of Roomrent on weekends",col=c("black","white"))



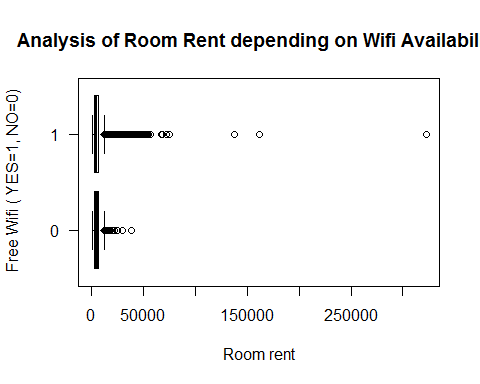
boxplot(cities.df$RoomRent ~ cities.df$IsMetroCity, horizontal=TRUE,ylab="City (metro=1,other=0) ", xlab="Room rent", las=1,main="Analysis of type of city and room rent of hotels",col=c("black","white"))



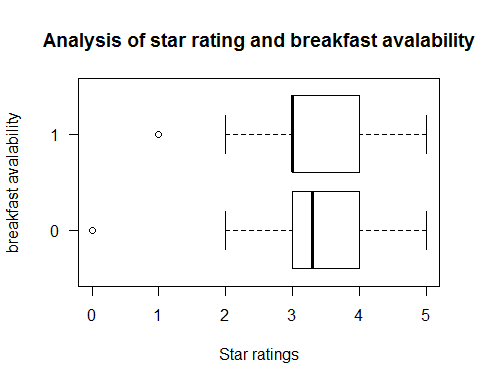
boxplot(cities.df$RoomRent ~ cities.df$HasSwimmingPool, horizontal=TRUE,main="Analysis of Room Rent depending on Swimming Pool",ylab="Swimming Pool ( YES=1, NO=0)",las=1,xlab="Room rent")



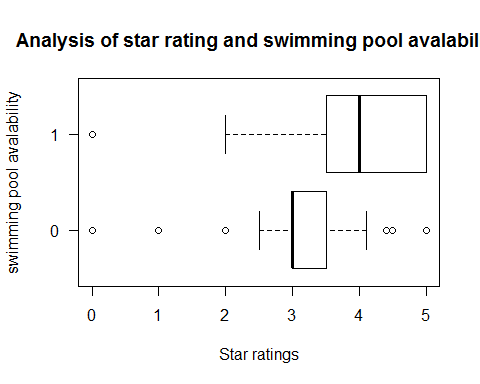
boxplot(cities.df$RoomRent ~ cities.df$FreeWifi, horizontal=TRUE,main="Analysis of Room Rent depending on Wifi Availabiliy ",ylab=" Free Wifi ( YES=1, NO=0)",las=1,xlab="Room rent")



boxplot(cities.df$StarRating ~ cities.df$FreeBreakfast, horizontal=TRUE,ylab="breakfast avalability", xlab="Star ratings", las=1,main="Analysis of star rating and breakfast avalability")

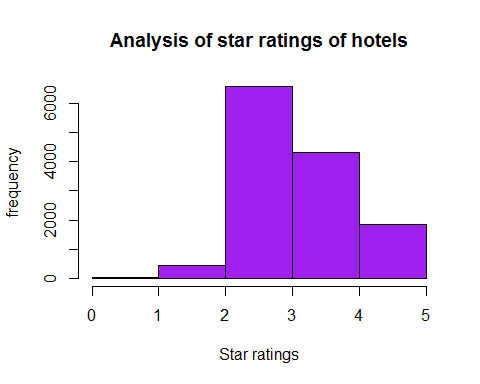


boxplot(cities.df$StarRating ~ cities.df$HasSwimmingPool, horizontal=TRUE,ylab="swimming pool avalability", xlab="Star ratings", las=1,main="Analysis of star rating and swimming pool avalability")



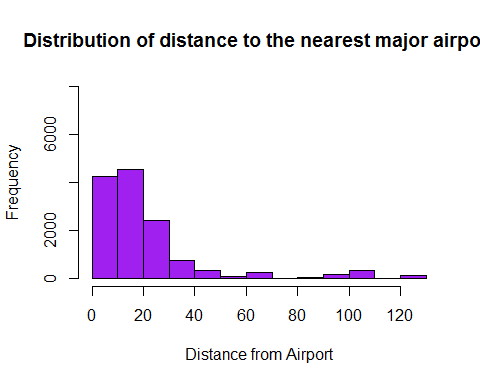
## Analysis of star rating of hotel

hist(cities.df$StarRating,main="Analysis of star ratings of hotels",xlab="Star ratings", ylab="frequency", breaks=5, col="purple", freq=TRUE)



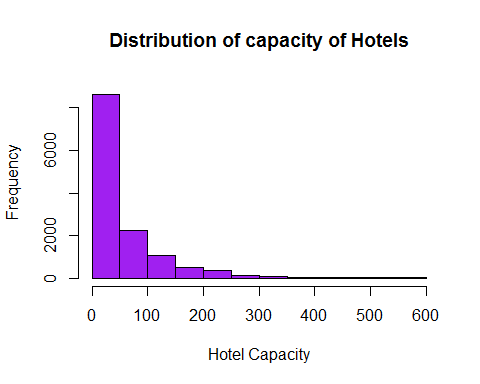
## Analysis of distance of airport from hotel

hist(cities.df$Airport,  
 xlab="Distance from Airport", col="purple",main="Distribution of distance to the nearest major airport", ylim = c(0,8000))



## Anaysis of hotel capacity on pricing

hist(cities.df$HotelCapacity,  
 xlab="Hotel Capacity", main="Distribution of capacity of Hotels",ylim=c(0,9000),col="purple")

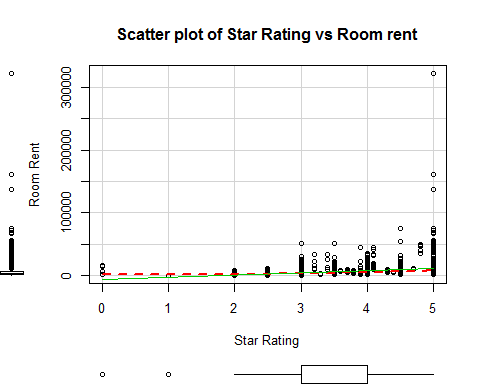
 ##Scatterplots

library(car)

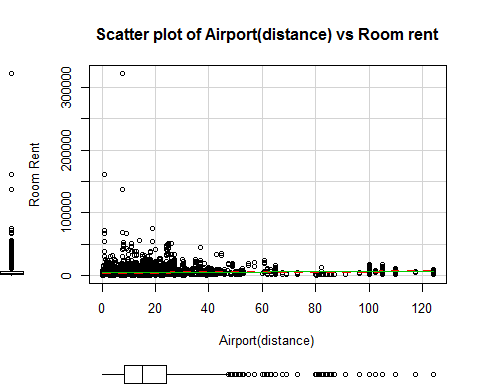
##   
## Attaching package: 'car'

## The following object is masked from 'package:psych':  
##   
## logit

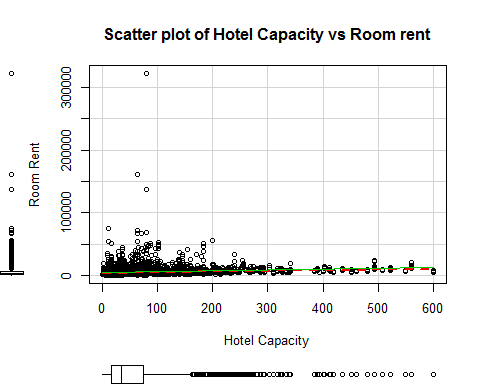
scatterplot(RoomRent~StarRating,data=cities.df,spread=FALSE, smoother.args=list(lty=2),main="Scatter plot of Star Rating vs Room rent",ylab="Room Rent", xlab="Star Rating")



scatterplot(RoomRent~Airport,data=cities.df,spread=FALSE, smoother.args=list(lty=2),main="Scatter plot of Airport(distance) vs Room rent",ylab="Room Rent", xlab="Airport(distance)")



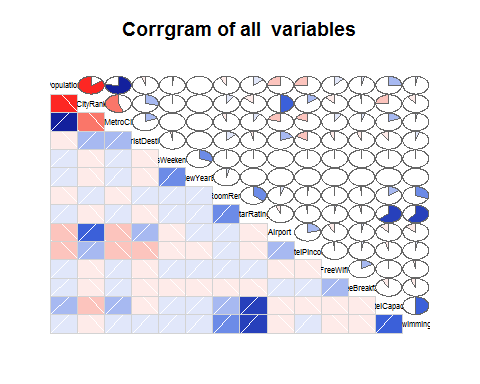
scatterplot(RoomRent~HotelCapacity,data=cities.df,spread=FALSE, smoother.args=list(lty=2),main="Scatter plot of Hotel Capacity vs Room rent",ylab="Room Rent", xlab="Hotel Capacity")



#scatterplot(RoomRent~cityRank,data=cities.df,spread=FALSE, smoother.args=list(lty=2),main="Scatter plot of City rank vs Room rent", ylab="Room Rent",xlab="Rank of City")  
  
#scatterplot(Population~cityRank,data=cities.df,spread=FALSE, smoother.args=list(lty=2),main="Scatter plot of City rank vs Population", ylab="Population",xlab="Rank of City")

## Corrgram

library(corrgram)  
corrgram(cities.df, lower.panel = panel.shade, upper.panel = panel.pie, text.panel = panel.txt, main = "Corrgram of all variables")



## Correlation matrix

cor.df <- cities.df[, c(2,3,4,5,6,7,10,11,12,18)]  
cor(cor.df)

## Population CityRank IsMetroCity  
## Population 1.0000000000 -0.8353204432 0.7712260105  
## CityRank -0.8353204432 1.0000000000 -0.5643937903  
## IsMetroCity 0.7712260105 -0.5643937903 1.0000000000  
## IsTouristDestination -0.0482029722 0.2807134520 0.1763717063  
## IsWeekend 0.0115926802 -0.0072564766 0.0018118005  
## IsNewYearEve 0.0007332482 -0.0006326444 0.0006464753  
## RoomRent -0.0887280632 0.0939855292 -0.0668397705  
## StarRating 0.1341365933 -0.1333810133 0.0776028661  
## Airport -0.2597010198 0.5059119892 -0.2073586125  
## HotelCapacity 0.2599830516 -0.2561197059 0.1871502153  
## IsTouristDestination IsWeekend IsNewYearEve  
## Population -0.048202972 0.011592680 0.0007332482  
## CityRank 0.280713452 -0.007256477 -0.0006326444  
## IsMetroCity 0.176371706 0.001811801 0.0006464753  
## IsTouristDestination 1.000000000 -0.019481101 -0.0022663884  
## IsWeekend -0.019481101 1.000000000 0.2923820508  
## IsNewYearEve -0.002266388 0.292382051 1.0000000000  
## RoomRent 0.122502963 0.004580134 0.0384912269  
## StarRating -0.040554998 0.006378436 0.0023608970  
## Airport 0.194422049 -0.002724756 0.0004598872  
## HotelCapacity -0.094356091 0.006306507 0.0013526790  
## RoomRent StarRating Airport HotelCapacity  
## Population -0.088728063 0.134136593 -0.2597010198 0.259983052  
## CityRank 0.093985529 -0.133381013 0.5059119892 -0.256119706  
## IsMetroCity -0.066839771 0.077602866 -0.2073586125 0.187150215  
## IsTouristDestination 0.122502963 -0.040554998 0.1944220492 -0.094356091  
## IsWeekend 0.004580134 0.006378436 -0.0027247555 0.006306507  
## IsNewYearEve 0.038491227 0.002360897 0.0004598872 0.001352679  
## RoomRent 1.000000000 0.369373425 0.0496532442 0.157873308  
## StarRating 0.369373425 1.000000000 -0.0609191837 0.637430337  
## Airport 0.049653244 -0.060919184 1.0000000000 -0.117672072  
## HotelCapacity 0.157873308 0.637430337 -0.1176720722 1.000000000

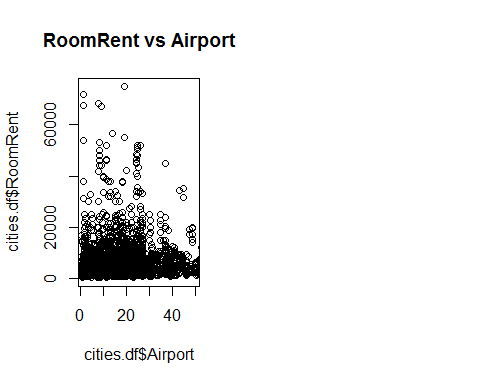
# RoomRent and istouristdestination, starRating, hotelCapacity are highly correlated. So prices peak due to these factors. RoomRent and population,IsMetroCity are negatively correlated.

## OLS model

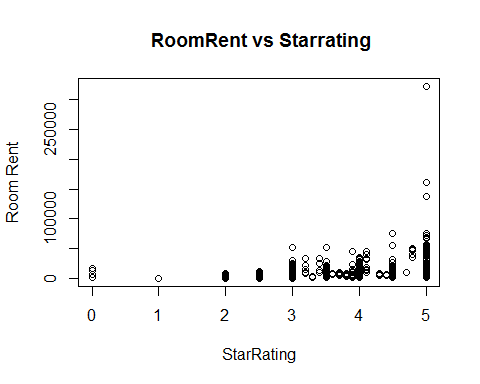
# OLS Model  
fit<- glm(RoomRent ~HasSwimmingPool+IsNewYearEve+IsTouristDestination+IsMetroCity+StarRating+Airport+HotelCapacity+FreeWifi+FreeBreakfast, data = cities.df)  
summary(fit)

##   
## Call:  
## glm(formula = RoomRent ~ HasSwimmingPool + IsNewYearEve + IsTouristDestination +   
## IsMetroCity + StarRating + Airport + HotelCapacity + FreeWifi +   
## FreeBreakfast, data = cities.df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -11696 -2375 -701 1063 309539   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -8906.418 405.396 -21.970 < 2e-16 \*\*\*  
## HasSwimmingPool 2227.069 159.327 13.978 < 2e-16 \*\*\*  
## IsNewYearEve 844.123 174.046 4.850 1.25e-06 \*\*\*  
## IsTouristDestination 2113.725 134.336 15.735 < 2e-16 \*\*\*  
## IsMetroCity -1548.328 138.527 -11.177 < 2e-16 \*\*\*  
## StarRating 3564.570 110.489 32.262 < 2e-16 \*\*\*  
## Airport 11.265 2.710 4.157 3.24e-05 \*\*\*  
## HotelCapacity -10.990 1.026 -10.714 < 2e-16 \*\*\*  
## FreeWifi 485.597 224.134 2.167 0.0303 \*   
## FreeBreakfast 182.992 123.296 1.484 0.1378   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 43657476)  
##   
## Null deviance: 7.1149e+11 on 13231 degrees of freedom  
## Residual deviance: 5.7724e+11 on 13222 degrees of freedom  
## AIC: 270339  
##   
## Number of Fisher Scoring iterations: 2

par(mfrow= c(1,2))  
plot(cities.df$RoomRent ~ cities.df$Airport, ylim= c(100 , 75000) , xlim=c(1,50) , main="RoomRent vs Airport")



plot(cities.df$RoomRent ~ cities.df$StarRating , main= " RoomRent vs Starrating" ,ylab= "Room Rent" , xlab="StarRating")



## cor test between RoomRent and IsTouristDestnation

cor.test(cities.df$RoomRent,cities.df$IsTouristDestination)

##   
## Pearson's product-moment correlation  
##   
## data: cities.df$RoomRent and cities.df$IsTouristDestination  
## t = 14.197, df = 13230, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.1056846 0.1392512  
## sample estimates:  
## cor   
## 0.122503

## Cor test between StarRating and RoomRent

cor.test(cities.df$RoomRent,cities.df$StarRating)

##   
## Pearson's product-moment correlation  
##   
## data: cities.df$RoomRent and cities.df$StarRating  
## t = 45.719, df = 13230, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3545660 0.3839956  
## sample estimates:  
## cor   
## 0.3693734

## Cor test between HotelCapacity and RoomRent

cor.test(cities.df$RoomRent, cities.df$HotelCapacity)

##   
## Pearson's product-moment correlation  
##   
## data: cities.df$RoomRent and cities.df$HotelCapacity  
## t = 18.389, df = 13230, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.1412142 0.1744430  
## sample estimates:  
## cor   
## 0.1578733

## T- test between RoomRent and HasSwimmingPool

2 Tail test Ho:-There is no significant difference between the Room Rent of Hotels with swimmin pool and hotels without swimmin pool H1:-There is a significant difference between the Room Rent of Hotels with swimmin pool and hotels without swimmin pool

t.test(cities.df$RoomRent[cities.df$HasSwimmingPool==0],cities.df$RoomRent[cities.df$HasSwimmingPool==1])

##   
## Welch Two Sample t-test  
##   
## data: cities.df$RoomRent[cities.df$HasSwimmingPool == 0] and cities.df$RoomRent[cities.df$HasSwimmingPool == 1]  
## t = -29.013, df = 5011.3, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -5096.030 -4450.942  
## sample estimates:  
## mean of x mean of y   
## 3775.566 8549.052

Inference: Since p value < .05, we reject null hypothesis H0 and accept that there is a significant difference bewtween the RoomRent of hotels with swimming pool and hotels without swimming pool.

## T-test between RoomRent and FreeWifi

Ho:-There is no significant difference between the Room Rent of Hotels providing free wifi and those which do not H1:-There is a significant difference between the Room Rent of Hotels providing free wifi and those which do not

t.test(cities.df$RoomRent[cities.df$FreeWifi==0],cities.df$RoomRent[cities.df$FreeWifi==1])

##   
## Welch Two Sample t-test  
##   
## data: cities.df$RoomRent[cities.df$FreeWifi == 0] and cities.df$RoomRent[cities.df$FreeWifi == 1]  
## t = -0.76847, df = 1804.7, p-value = 0.4423  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -360.5977 157.5701  
## sample estimates:  
## mean of x mean of y   
## 5380.004 5481.518

## T-test between RoomRent and FreeBreakfast

Ho: There is no significant difference between the Room Rent of Hotels with free breakfast and hotels without free breakfast. H1: There is a significant difference between the Room Rent of Hotels with free breakfast and hotels without free breakfast

t.test(cities.df$RoomRent[cities.df$FreeBreakfast==0],cities.df$RoomRent[cities.df$FreeBreakfast==1])

##   
## Welch Two Sample t-test  
##   
## data: cities.df$RoomRent[cities.df$FreeBreakfast == 0] and cities.df$RoomRent[cities.df$FreeBreakfast == 1]  
## t = 0.98095, df = 6212.3, p-value = 0.3267  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -153.5017 460.9935  
## sample estimates:  
## mean of x mean of y   
## 5573.790 5420.044

Inference: Since p-value>0.05, we accept H0,hence, There is no significant difference between the Room Rent of Hotels with free breakfast and hotels without free breakfast.

## T-test between RoomRent and IsNewYearEve

Ho: There is no significant difference between the Room Rent of Hotels on normal Eve and New Year’s Eve. H1: The Room Rents of Hotels on normal Eve are cheaper than that on New Year’s Eve

t.test(cities.df$RoomRent[cities.df$IsNewYearEve==0],cities.df$RoomRent[cities.df$IsNewYearEve==1],alternative = "less")

##   
## Welch Two Sample t-test  
##   
## data: cities.df$RoomRent[cities.df$IsNewYearEve == 0] and cities.df$RoomRent[cities.df$IsNewYearEve == 1]  
## t = -4.1793, df = 2065, p-value = 1.523e-05  
## alternative hypothesis: true difference in means is less than 0  
## 95 percent confidence interval:  
## -Inf -518.4763  
## sample estimates:  
## mean of x mean of y   
## 5367.606 6222.826

Inference: Since p-value<0.05, we accept H1,hence,the Room Rents of Hotels on normal Eve are cheaper than that on New Year’s Eve.

## T-test between RoomRent and IsWeekend

Ho:-There is no significant difference between the Room Rent of Hotels on weekdays and weekends. H1:-There is a significant difference between the Room Rent of Hotels on weekdays and weekends.

t.test(cities.df$RoomRent[cities.df$IsWeekend==0],cities.df$RoomRent[cities.df$IsWeekend==1])

##   
## Welch Two Sample t-test  
##   
## data: cities.df$RoomRent[cities.df$IsWeekend == 0] and cities.df$RoomRent[cities.df$IsWeekend == 1]  
## t = -0.51853, df = 9999.4, p-value = 0.6041  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -331.2427 192.6559  
## sample estimates:  
## mean of x mean of y   
## 5430.835 5500.129

Inference:-Since p-vale>0.05, we accept Ho,hence there is no significant difference between the Room Rent of Hotels on weekdays and weekends.

## T-test between RoomRent and IsTouristDestination

Ho: There is no significant difference between the Room Rent of Hotels in Tourist destinations and non tourist destinations. H1: The Room Rents of Hotels in Tourist destinations are greater than that in non tourist destinations

t.test(cities.df$RoomRent[cities.df$IsTouristDestination==0],cities.df$RoomRent[cities.df$IsTouristDestination==1],alternative = "less")

##   
## Welch Two Sample t-test  
##   
## data: cities.df$RoomRent[cities.df$IsTouristDestination == 0] and cities.df$RoomRent[cities.df$IsTouristDestination == 1]  
## t = -19.449, df = 12888, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is less than 0  
## 95 percent confidence interval:  
## -Inf -1789.665  
## sample estimates:  
## mean of x mean of y   
## 4111.003 6066.024

Inference: Since p-value<0.05, we accept H1,hence,the Room.Rents of Hotels in Tourist destinations are greater than that in non tourist destinations

## T-test between RoomRent and IsMetroCity

Ho: There is no significant difference between the Room Rent of Hotels in non-metro cities and metro cities. H1: The Room Rent of hotels in non-metro cities are more expensive than that in metro cities.

t.test(cities.df$RoomRent[cities.df$IsMetroCity==0],cities.df$RoomRent[cities.df$IsMetroCity==1],alternative = "greater")

##   
## Welch Two Sample t-test  
##   
## data: cities.df$RoomRent[cities.df$IsMetroCity == 0] and cities.df$RoomRent[cities.df$IsMetroCity == 1]  
## t = 10.721, df = 13224, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is greater than 0  
## 95 percent confidence interval:  
## 919.9785 Inf  
## sample estimates:  
## mean of x mean of y   
## 5782.794 4696.073

Inference: Since p-value<0.05, we accept H1,hence,the Room Rents of Hotels in non-metro cities is more than that of metro cities.

## DISCUSSION

FROM THE ABOVE ANALYSIS, WE CAN SAY THAT THE FOLLOWING VARIABLES ARE SIGNIFICANT:

1.IsMetroCity

2.IsNewYearEve

3.HasSwimmingPool

4.IsTouristDestination

## CONCLUSIONS

The most significant factors include the location of a hotel whether it is in a tourist area or in a metropolitan city, the date of booking falls on a special occasion like New Year Eve, the review it has in terms of rating and the total capacity of the hotel that determines the price of a room.

facility of swimming pool and the star rating will affect the hotel price, positively.

Free wifi,free breakfast does not impact much.

## REFERENCES

1, www.youtube.com

1. www.kaggle.com
2. www.stackoverflow.com
3. www.udemy.com
4. www,coursera.org
5. R documentation

**This was the final project to be submitted under the internship of Prof SammerMathur (IIM Lucknow, CMU) as a part of his data analytics internship in R.**