Ankit Kashyap My Project Code.r

**Hotel Room Pricing In The Indian Market**

Ankit Kashyap

Mon Jul 10 03:58:03 2017

# Analysis of Hotel Room Pricing in Indian Market  
# NAME: Ankit Kashyap  
# EMAIL: ankitkashyap.te15@rvce.edu.in  
# COLLEGE : R.V.COllege Of Engineering, Bengaluru.

## 1. Reading the raw data into a data frame

##setting the directory and assigning a variabel to the data frame  
setwd("C:/Users/Ankit/Desktop/harvard")  
  
#Reading the dataset and creating a data frame  
hotel.df<-read.csv(paste("Cities42.csv",sep = ""))  
  
#Viewing the data  
View(hotel.df)

## 2. Changing the irregularity of dates in the data frame

Use of gsub() command to replace the wrong format of the date

#Removing the repeated date by gsub command  
  
hotel.df$Date<-gsub("18-Dec-16", "Dec 18 2016", hotel.df$Date)  
hotel.df$Date<-gsub("21-Dec-16", "Dec 21 2016", hotel.df$Date)  
hotel.df$Date<-gsub("24-Dec-16", "Dec 24 2016", hotel.df$Date)  
hotel.df$Date<-gsub("25-Dec-16", "Dec 25 2016", hotel.df$Date)  
hotel.df$Date<-gsub("28-Dec-16", "Dec 28 2016", hotel.df$Date)  
hotel.df$Date<-gsub("31-Dec-16", "Dec 31 2016", hotel.df$Date)  
hotel.df$Date<-gsub("4-Jan-17", "Jan 04 2017", hotel.df$Date)  
hotel.df$Date<-gsub("4-Jan-16", "Jan 04 2017", hotel.df$Date)  
hotel.df$Date<-gsub("8-Jan-16", "Jan 08 2017", hotel.df$Date)  
hotel.df$Date<-gsub("8-Jan-17", "Jan 08 2017", hotel.df$Date)  
hotel.df$Date<-gsub("Jan 4 2017", "Jan 04 2017", hotel.df$Date)  
hotel.df$Date<-gsub("Jan 8 2017", "Jan 08 2017", hotel.df$Date)  
  
#Checking the dates  
  
table(hotel.df$Date)

##   
## Dec 18 2016 Dec 21 2016 Dec 24 2016 Dec 25 2016 Dec 28 2016 Dec 31 2016   
## 1652 1655 1655 1655 1655 1655   
## Jan 04 2017 Jan 08 2017   
## 1652 1653

#Changing dates to factors for labelling   
  
hotel.df$Date<-factor(hotel.df$Date)  
  
##Arrange the factors in order  
is.factor(hotel.df$Date)

## [1] TRUE

#Checking the labelling  
levels(hotel.df$Date)

## [1] "Dec 18 2016" "Dec 21 2016" "Dec 24 2016" "Dec 25 2016" "Dec 28 2016"  
## [6] "Dec 31 2016" "Jan 04 2017" "Jan 08 2017"

# DATA SUMMARY

### 3. Summary Statistics - mean, sd, median, min, max of variables

#Analyzing the summary of the data and describing the variables  
  
library(psych)  
describe(hotel.df)

## vars n mean sd median trimmed  
## X 1 13232 6616.50 3819.89 6616.5 6616.50  
## CityName\* 2 13232 18.07 11.72 16.0 17.29  
## Population 3 13232 4416836.87 4258386.00 3046163.0 4040816.22  
## CityRank 4 13232 14.83 13.51 9.0 13.30  
## IsMetroCity 5 13232 0.28 0.45 0.0 0.23  
## IsTouristDestination 6 13232 0.70 0.46 1.0 0.75  
## IsWeekend 7 13232 0.62 0.48 1.0 0.65  
## IsNewYearEve 8 13232 0.12 0.33 0.0 0.03  
## Date\* 9 13232 4.50 2.29 4.0 4.50  
## HotelName\* 10 13232 841.19 488.16 827.0 841.18  
## RoomRent 11 13232 5473.99 7333.12 4000.0 4383.33  
## StarRating 12 13232 3.46 0.76 3.0 3.40  
## Airport 13 13232 21.16 22.76 15.0 16.39  
## HotelAddress\* 14 13232 1202.53 582.17 1261.0 1233.25  
## HotelPincode 15 13232 397430.26 259837.50 395003.0 388540.47  
## HotelDescription\* 16 13224 581.34 363.26 567.0 575.37  
## FreeWifi 17 13232 0.93 0.26 1.0 1.00  
## FreeBreakfast 18 13232 0.65 0.48 1.0 0.69  
## HotelCapacity 19 13232 62.51 76.66 34.0 46.03  
## HasSwimmingPool 20 13232 0.36 0.48 0.0 0.32  
## mad min max range skew  
## X 4904.44 1.0 13232 13231.0 0.00  
## 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 8 7.0 0.00  
## 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  
## X -1.20 33.21  
## 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\* -1.24 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

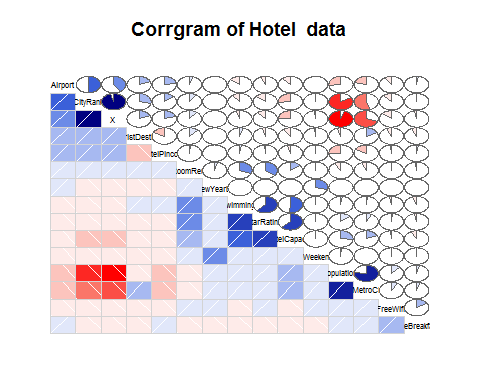
summary(hotel.df)

## X CityName Population CityRank   
## Min. : 1 Delhi :2048 Min. : 8096 Min. : 0.00   
## 1st Qu.: 3309 Jaipur : 768 1st Qu.: 744983 1st Qu.: 2.00   
## Median : 6616 Mumbai : 712 Median : 3046163 Median : 9.00   
## Mean : 6616 Bangalore: 656 Mean : 4416837 Mean :14.83   
## 3rd Qu.: 9924 Goa : 624 3rd Qu.: 8443675 3rd Qu.:24.00   
## Max. :13232 Kochi : 608 Max. :12442373 Max. :44.00   
## (Other) :7816   
## IsMetroCity IsTouristDestination IsWeekend IsNewYearEve   
## Min. :0.0000 Min. :0.0000 Min. :0.0000 Min. :0.0000   
## 1st Qu.:0.0000 1st Qu.:0.0000 1st Qu.:0.0000 1st Qu.:0.0000   
## Median :0.0000 Median :1.0000 Median :1.0000 Median :0.0000   
## Mean :0.2842 Mean :0.6972 Mean :0.6228 Mean :0.1244   
## 3rd Qu.:1.0000 3rd Qu.:1.0000 3rd Qu.:1.0000 3rd Qu.:0.0000   
## Max. :1.0000 Max. :1.0000 Max. :1.0000 Max. :1.0000   
##   
## Date HotelName RoomRent   
## Dec 21 2016:1655 Vivanta by Taj : 32 Min. : 299   
## Dec 24 2016:1655 Goldfinch Hotel : 24 1st Qu.: 2436   
## Dec 25 2016:1655 OYO Rooms : 24 Median : 4000   
## Dec 28 2016:1655 The Gordon House Hotel: 24 Mean : 5474   
## Dec 31 2016:1655 Apnayt Villa : 16 3rd Qu.: 6299   
## Jan 08 2017:1653 Bentleys Hotel Colaba : 16 Max. :322500   
## (Other) :3304 (Other) :13096   
## StarRating Airport   
## Min. :0.000 Min. : 0.20   
## 1st Qu.:3.000 1st Qu.: 8.40   
## Median :3.000 Median : 15.00   
## Mean :3.459 Mean : 21.16   
## 3rd Qu.:4.000 3rd Qu.: 24.00   
## Max. :5.000 Max. :124.00   
##   
## HotelAddress   
## The Mall, Shimla : 32   
## #2-91/14/8, White Fields, Kondapur, Hitech City, Hyderabad, 500084 India: 16   
## 121, City Terrace, Walchand Hirachand Marg, Mumbai, Maharashtra : 16   
## 14-4507/9, Balmatta Road, Near Jyothi Circle, Hampankatta : 16   
## 144/7, Rajiv Gandi Salai (OMR), Kottivakkam, Chennai, Tamil Nadu : 16   
## 17, Oliver Road, Colaba, Mumbai, Maharashtra : 16   
## (Other) :13120   
## HotelPincode HotelDescription FreeWifi FreeBreakfast   
## Min. : 100025 3 : 120 Min. :0.0000 Min. :0.0000   
## 1st Qu.: 221001 Abc : 112 1st Qu.:1.0000 1st Qu.:0.0000   
## Median : 395003 3-star hotel: 104 Median :1.0000 Median :1.0000   
## Mean : 397430 3.5 : 88 Mean :0.9259 Mean :0.6491   
## 3rd Qu.: 570001 4 : 72 3rd Qu.:1.0000 3rd Qu.:1.0000   
## Max. :7000157 (Other) :12728 Max. :1.0000 Max. :1.0000   
## NA's : 8   
## HotelCapacity HasSwimmingPool   
## Min. : 0.00 Min. :0.0000   
## 1st Qu.: 16.00 1st Qu.:0.0000   
## Median : 34.00 Median :0.0000   
## Mean : 62.51 Mean :0.3558   
## 3rd Qu.: 75.00 3rd Qu.:1.0000   
## Max. :600.00 Max. :1.0000   
##

## 4. Identifying the idependent variable Y and independent variables X1,X2 and X3 from the dataframe.

#Taking Y = RoomRent, identifying the most relevent predictor variables by correlation corrgram

#Corrgram  
  
library(corrgram)  
  
corrgram(hotel.df, order=TRUE, lower.panel=panel.shade,  
 upper.panel=panel.pie, text.panel=panel.txt,  
 main="Corrgram of Hotel data")



##through corrgram HasSwimming, StarRating, HotelCapital are very well correlated to RoomRent  
##so we can take them as predictors  
  
##Visualizing data for Y as Room rent and X1,X2,X3 as HasSwimmingPool, StarRating and HotelCapacity respectively

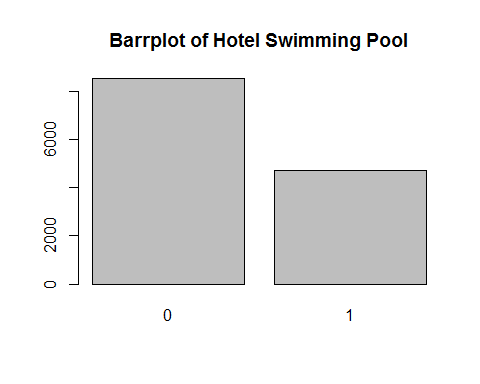
# VISUALIZATION

## 5. Visualizing the independent variables X1,X2 and X3 in the dataframe

#Table for HasSwimmingPool  
table(hotel.df$HasSwimmingPool)

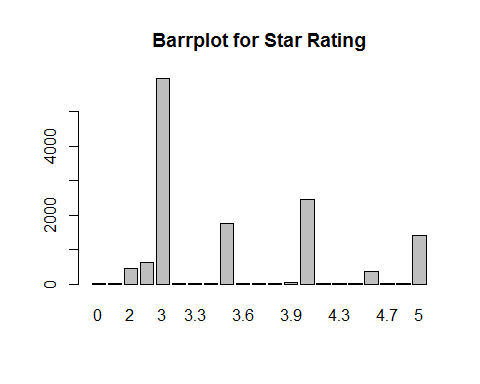
##   
## 0 1   
## 8524 4708

hasSwim<-table(hotel.df$HasSwimmingPool)  
barplot(hasSwim,main="Barrplot of Hotel Swimming Pool")



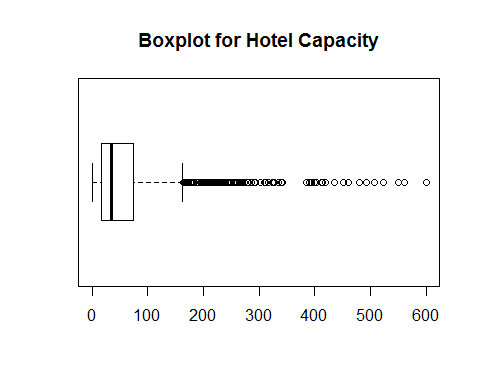
Result: The above visualization tells us that the number of hotel not having swimming pools is greater than the number of hotels having swimming pool.

#Table for StarRating  
hasstarRating<-table(hotel.df$StarRating)  
barplot(hasstarRating,main = "Barrplot for Star Rating")



#BoxPlot for HotelCapacity  
boxplot(hotel.df$HotelCapacity, main="Boxplot for Hotel Capacity",horizontal = TRUE)

Result: The above data reveals the class of hotels in India , with 3 star hotels at it’s maximum i.e., the number of 3 star hotels is India is too large.



Result: There are a lot of outlier to the hotel capacity data which makes the data quite uncertain about the mean and median

# ROLE OF DIFFERENT DEPENDENT VARIABLES ON THE PRICNG OF THE HOTEL ROOM.

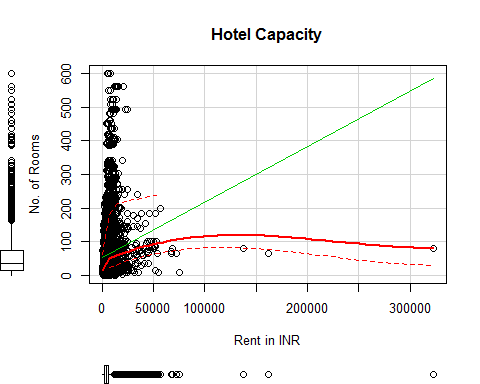
### 5a. Scatterplot distribution between Hotel Capacity and RoomRent

#Scatterplot pair wise for predictor variable  
library(car)

##   
## Attaching package: 'car'

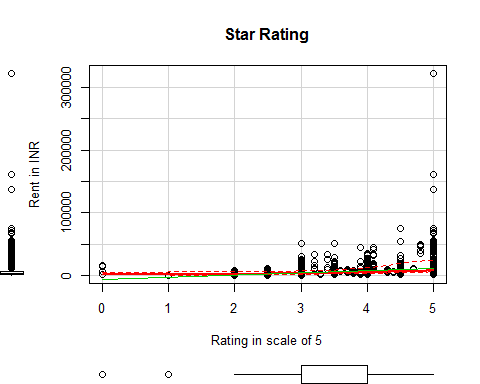
## The following object is masked from 'package:psych':  
##   
## logit

scatterplot(hotel.df$RoomRent,hotel.df$HotelCapacity,main="Hotel Capacity",ylab = "No. of Rooms", xlab="Rent in INR",cex=1.1)



### 5b. Scatterplot distribution between Star Rating and RoomRent

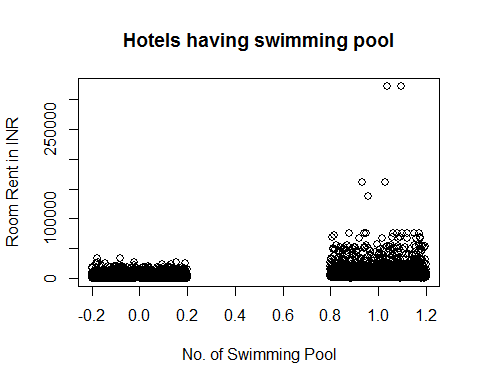
scatterplot(hotel.df$StarRating,hotel.df$RoomRent,main="Star Rating",ylab = "Rent in INR", xlab="Rating in scale of 5",cex=1.1)



### 5c. Plot and bwplot distribution between HasSwimmingPool and RoomRent

#RoomRent Vs HasSwimmingPool

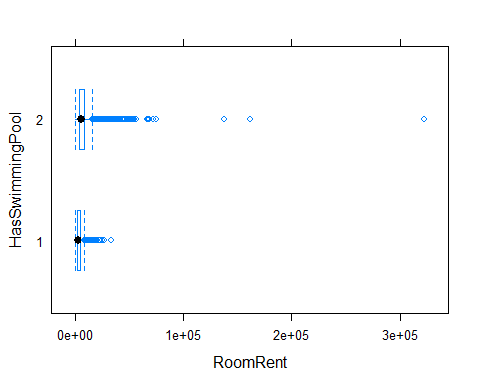
plot(jitter(hotel.df$HasSwimmingPool),jitter(hotel.df$RoomRent),main="Hotels having swimming pool",ylab = "Room Rent in INR", xlab="No. of Swimming Pool",cex=1.1)



library(lattice)

## Warning: package 'lattice' was built under R version 3.4.1

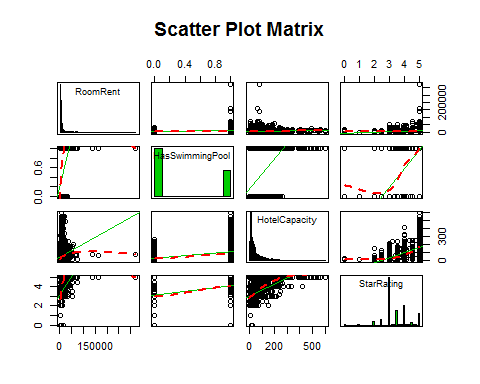
bwplot(HasSwimmingPool~RoomRent,data=hotel.df)



### 5d. Scattreplotmatrix distribution between Hotel Capacity, HasSwimmingPool, StarRating and RoomRent

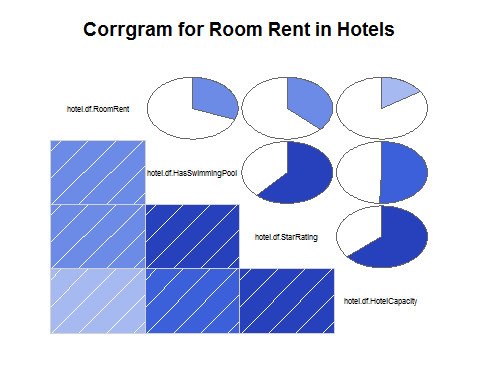
#Scatter plot matrix for RoomRent,HasswimmingPool,StarRating and Hotel Capacity  
library(car)  
scatterplotMatrix(  
 hotel.df[  
 ,c("RoomRent","HasSwimmingPool","HotelCapacity","StarRating")],  
 spread=FALSE, smoother.args=list(lty=2),  
 main="Scatter Plot Matrix", diagonal = "histogram")

## Warning in smoother(x, y, col = col[2], log.x = FALSE, log.y = FALSE,  
## spread = spread, : could not fit smooth



### 5e. Corrgram of Hotel Capacity, HasSwimmingPool, StarRating and RoomRent

#Corrgram for RoomRent,HasSwimmingPool,StarRating,HotelCapacity  
library(corrgram)  
hotelnew<-data.frame(hotel.df$RoomRent,hotel.df$StarRating,hotel.df$HotelCapacity,hotel.df$HasSwimmingPool)  
corrgram(hotelnew, order=TRUE, lower.panel=panel.shade,  
 upper.panel=panel.pie, text.panel=panel.txt,  
 main="Corrgram for Room Rent in Hotels")



### 6. Covariance and Varaince matrix between Independent variables and RoomRent

#Variance Matrix, Correlation Matrix and Covariance Matrix  
x<-hotel.df[,c("HasSwimmingPool","StarRating","HotelCapacity")]  
y<-hotel.df[,c("RoomRent")]  
cor(x,y)

## [,1]  
## HasSwimmingPool 0.3116577  
## StarRating 0.3693734  
## HotelCapacity 0.1578733

cov(x,y)

## [,1]  
## HasSwimmingPool 1094.202  
## StarRating 2048.375  
## HotelCapacity 88753.413

var(x,y)

## [,1]  
## HasSwimmingPool 1094.202  
## StarRating 2048.375  
## HotelCapacity 88753.413

#Forming a variable which is having RoomRent less than 1 lakh because the outliers effect the average

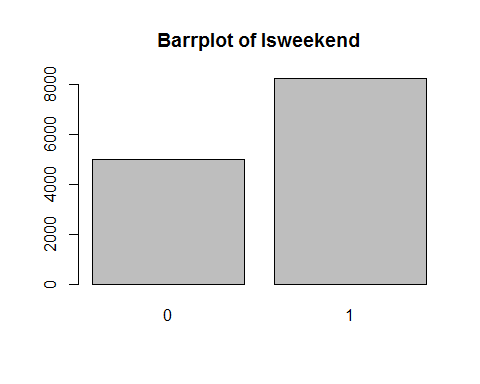
roomrent.df<-hotel.df[which(hotel.df$RoomRent<100000),]

This data frame containing the room rent of hotels less than 100k will help us to get a clear

View of how really is the mean of the data without getting affected by the extreme outliers.

### 7. Summary and Visualization of other factors which affect RoomRent

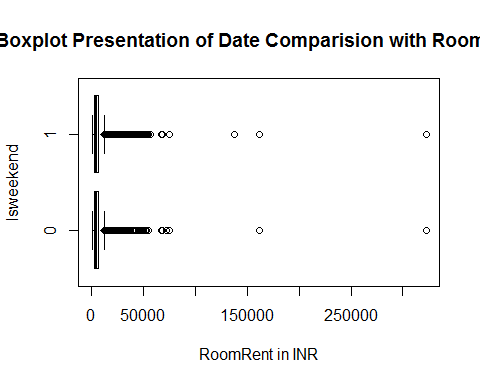
#Comparing other factors and their pattern using other trends with roomrent  
  
  
#Comparing IsWeekend with RoomRent  
#Visualization  
weekend1<-table(hotel.df$IsWeekend)  
library(lattice)  
barplot(weekend1,main="Barrplot of Isweekend")



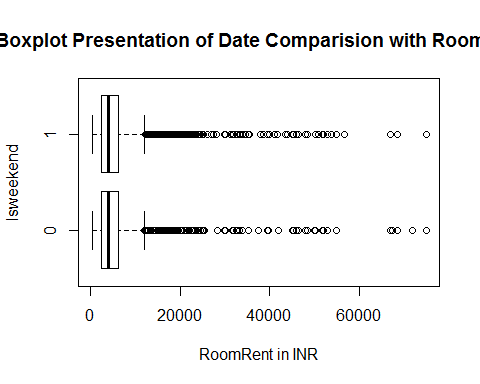
#Effect of Isweekend on RoomRent  
pd = aggregate(RoomRent ~ IsWeekend, data = hotel.df, mean)  
pd

## IsWeekend RoomRent  
## 1 0 5430.835  
## 2 1 5500.129

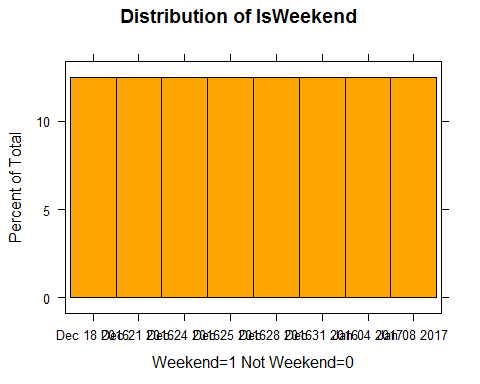
boxplot(RoomRent~IsWeekend,data=hotel.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="Isweekend",main="Boxplot Presentation of Date Comparision with Roomrent")



#Effect of Isweekend on RoomRent without extreme outliers  
boxplot(RoomRent~IsWeekend,data=roomrent.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="Isweekend",main="Boxplot Presentation of Date Comparision with Roomrent")



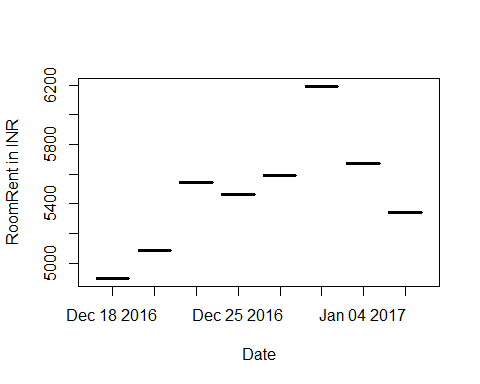
#Comparing RoomRent with Date  
library(lattice)  
histogram(~Date, data = hotel.df,main = "Distribution of IsWeekend", xlab= "Weekend=1 Not Weekend=0",col='orange')



#Effect of Date on RoomRent  
dd = aggregate(RoomRent ~ Date, data = hotel.df, mean)  
dd

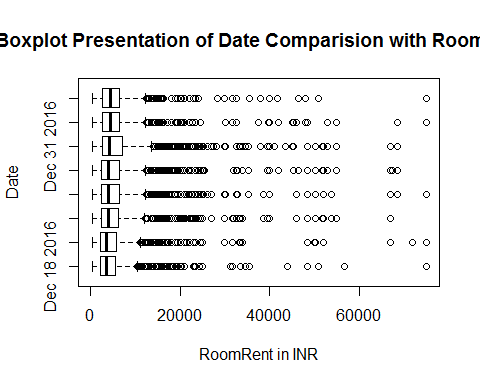
## Date RoomRent  
## 1 Dec 18 2016 4896.402  
## 2 Dec 21 2016 5085.315  
## 3 Dec 24 2016 5543.236  
## 4 Dec 25 2016 5464.143  
## 5 Dec 28 2016 5593.924  
## 6 Dec 31 2016 6191.776  
## 7 Jan 04 2017 5674.062  
## 8 Jan 08 2017 5342.234

library(car)  
scatterplot(dd$Date,dd$RoomRent,main="Effect of Date on RoomRent",xlab="Date",ylab="RoomRent in INR")

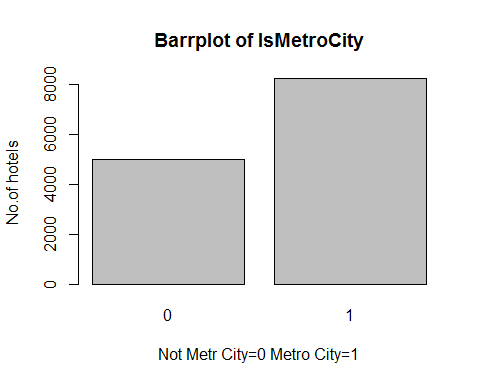


Result : The above Visualization of room rents according to the sold out dates tell us that the room rent on 31st December 2016 was the highest among all sold out dates. The average room rent on 31st December was around 6.1k.

#Effect of Date on RoomRent eliminating extreme outliers  
boxplot(RoomRent~Date,data=roomrent.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="Date",main="Boxplot Presentation of Date Comparision with Roomrent")



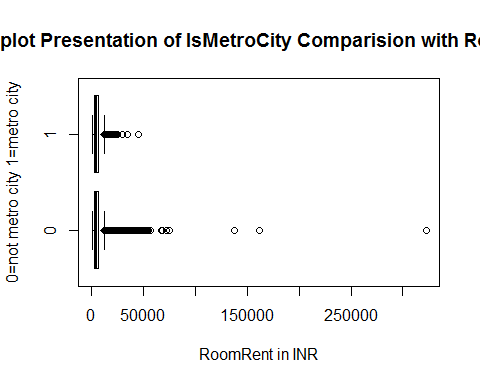
#Comparing RoomRent with IsMetroCity  
metro1<-table(hotel.df$IsMetroCity)  
barplot(weekend1,main="Barrplot of IsMetroCity",xlab="Not Metr City=0 Metro City=1",ylab="No.of hotels")



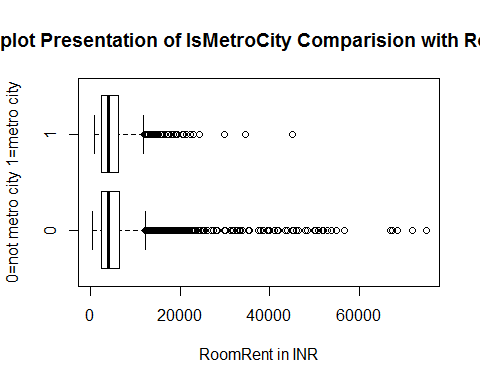
#Effect of IsMetroCity on RoomRent  
im = aggregate(RoomRent ~ IsMetroCity, data = hotel.df, mean)  
im

## IsMetroCity RoomRent  
## 1 0 5782.794  
## 2 1 4696.073

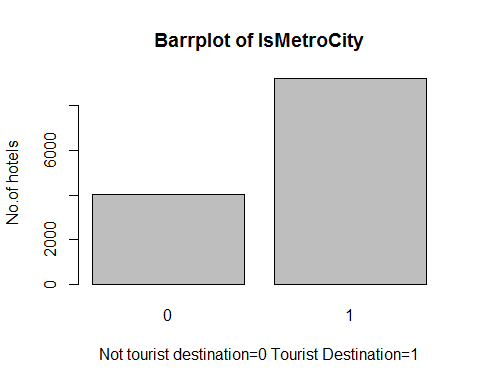
boxplot(RoomRent~IsMetroCity,data=hotel.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="0=not metro city 1=metro city",main="Boxplot Presentation of IsMetroCity Comparision with Roomrent")



#Effect of IsMetroCity on RoomRent eliminating extreme outliers  
boxplot(RoomRent~IsMetroCity,data=roomrent.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="0=not metro city 1=metro city",main="Boxplot Presentation of IsMetroCity Comparision with Roomrent")



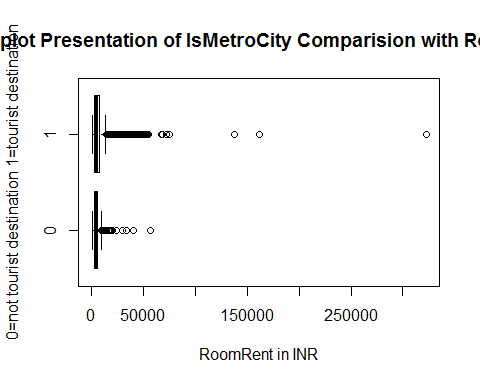
#Comparing RoomRent with IsTouristDestinatio  
tourist1<-table(hotel.df$IsTouristDestination)  
barplot(tourist1,main="Barrplot of IsMetroCity",xlab="Not tourist destination=0 Tourist Destination=1",ylab="No.of hotels")



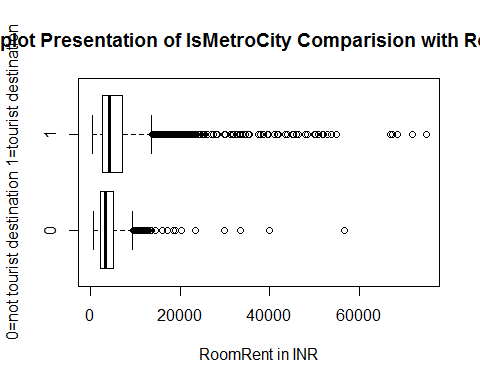
#Effect of IsTouristDestination on RoomRent  
id = aggregate(RoomRent ~ IsTouristDestination, data = hotel.df, mean)  
id

## IsTouristDestination RoomRent  
## 1 0 4111.003  
## 2 1 6066.024

boxplot(RoomRent~IsTouristDestination,data=hotel.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="0=not tourist destination 1=tourist destination",main="Boxplot Presentation of IsMetroCity Comparision with Roomrent")



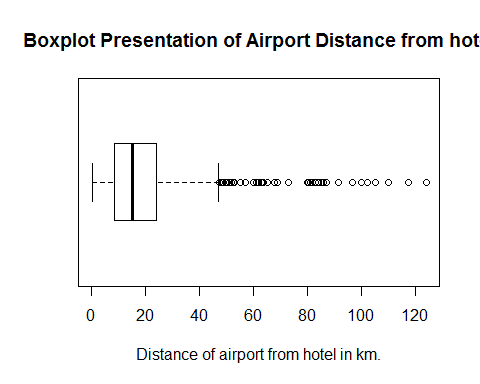
#Effect of IsTouristDestination on RoomRent eliminating extreme outliers  
boxplot(RoomRent~IsTouristDestination,data=roomrent.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="0=not tourist destination 1=tourist destination",main="Boxplot Presentation of IsMetroCity Comparision with Roomrent")



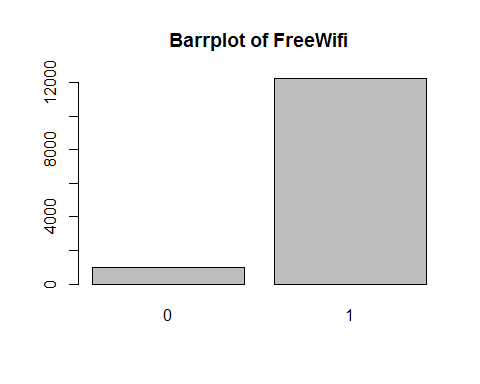
#Effect of Airport and Ismetrocity on RoomRent  
summary(hotel.df$Airport)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.20 8.40 15.00 21.16 24.00 124.00

boxplot(hotel.df$Airport,horizontal = TRUE,xlab="Distance of airport from hotel in km.",main="Boxplot Presentation of Airport Distance from hotel")  
  
#Effect of Airport and Ismerocity on roomrent eliminating extreme outliers  
boxplot(roomrent.df$Airport,horizontal = TRUE,xlab="Distance of airport from hotel in km.",main="Boxplot Presentation of Airport Distance from hotel")



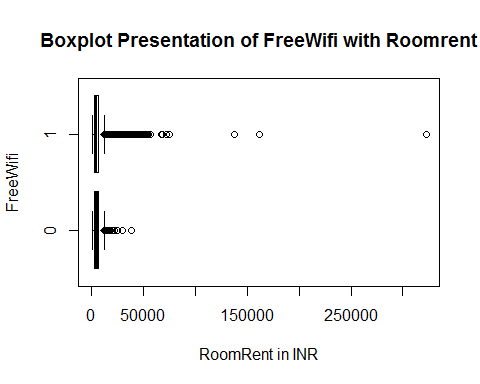
#Comparing FreeWifi and RoomRent  
free1<-table(hotel.df$FreeWifi)  
library(lattice)  
barplot(free1,main="Barrplot of FreeWifi")



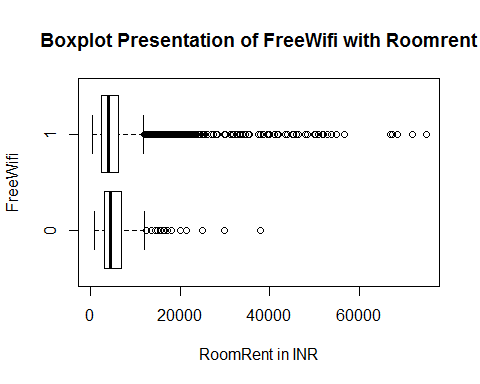
#Effect of FreeWifi on RoomRent  
fw = aggregate(RoomRent ~ FreeWifi, data = hotel.df, mean)  
fw

## FreeWifi RoomRent  
## 1 0 5380.004  
## 2 1 5481.518

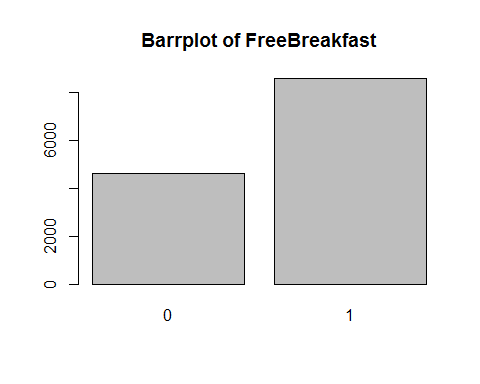
boxplot(RoomRent~FreeWifi,data=hotel.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="FreeWifi",main="Boxplot Presentation of FreeWifi with Roomrent")



#Effect of FreeWifi on RoomRent eliminating extreme outliers  
boxplot(RoomRent~FreeWifi,data=roomrent.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="FreeWifi",main="Boxplot Presentation of FreeWifi with Roomrent")



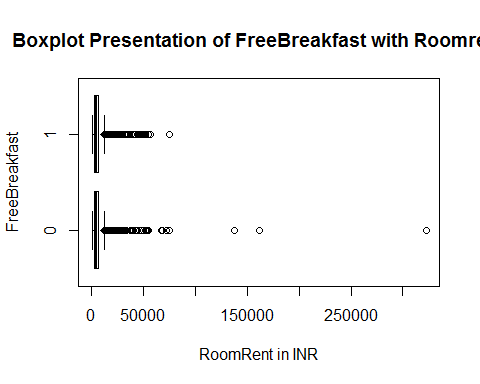
#Comparing FreeBreakfast and RoomRent  
free2<-table(hotel.df$FreeBreakfast)  
library(lattice)  
barplot(free2,main="Barrplot of FreeBreakfast")



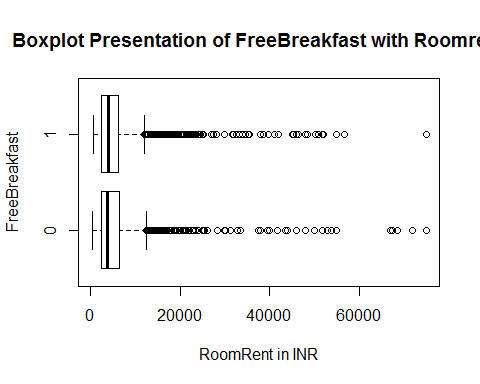
#Effect of FreeWifi on RoomRent  
fb = aggregate(RoomRent ~ FreeWifi, data = hotel.df, mean)  
fb

## FreeWifi RoomRent  
## 1 0 5380.004  
## 2 1 5481.518

boxplot(RoomRent~FreeBreakfast,data=hotel.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="FreeBreakfast",main="Boxplot Presentation of FreeBreakfast with Roomrent")

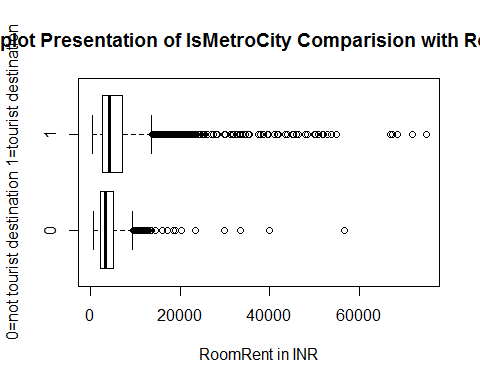


#Effect of FreeWifi on RoomRent eliminating extreme outlier  
boxplot(RoomRent~FreeBreakfast,data=roomrent.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="FreeBreakfast",main="Boxplot Presentation of FreeBreakfast with Roomrent")



boxplot(RoomRent~IsTouristDestination,data=roomrent.df,horizontal = TRUE,xlab="RoomRent in INR",ylab="0=not tourist destination 1=tourist destination",main="Boxplot Presentation of IsMetroCity Comparision with Roomrent")

Result : The RoomRent for Hotel changes according with the outlier when it comes to FreeBreakfast



Result: The prices of Room of Hotels in Tourist Places is far more and have more outliers as that of normal city.

# Hypothesis

### 8. Articulating hypothesis and conducting t-test to determine their p value

#Hpothesis  
#1.Average RoomRent in hotels having swimmingpool is more than that which don't have  
t.test(RoomRent~HasSwimmingPool,data=hotel.df,alternative="less")

##   
## Welch Two Sample t-test  
##   
## data: RoomRent by HasSwimmingPool  
## t = -29.013, df = 5011.3, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is less than 0  
## 95 percent confidence interval:  
## -Inf -4502.814  
## sample estimates:  
## mean in group 0 mean in group 1   
## 3775.566 8549.052

* Since the p-value is less than 0.05, we can reject the null hypothesis that the mean are equal

#2.Average RoomRent in hotels with high star rating is high as compared to one which has low star rating.  
t.test(hotel.df$RoomRent,hotel.df$StarRating)

##   
## Welch Two Sample t-test  
##   
## data: hotel.df$RoomRent and hotel.df$StarRating  
## t = 85.813, df = 13231, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 5345.575 5595.491  
## sample estimates:  
## mean of x mean of y   
## 5473.991838 3.458933

* Since the p-value is less than 0.05, we can reject the null hypothesis that they are equal

#3.Average RoomRent in hotels providing Free Breakfast is more than that which don't provide free breakfast  
t.test(RoomRent~FreeBreakfast,data=hotel.df,alternative="less")

##   
## Welch Two Sample t-test  
##   
## data: RoomRent by FreeBreakfast  
## t = 0.98095, df = 6212.3, p-value = 0.8367  
## alternative hypothesis: true difference in means is less than 0  
## 95 percent confidence interval:  
## -Inf 411.5844  
## sample estimates:  
## mean in group 0 mean in group 1   
## 5573.790 5420.044

* Since the p-value is more than 0.05, we fail to reject the null hypothesis that they are equal

#4.Average RoomRent in metro cities hotels is more than that of non metro cities.  
t.test(RoomRent~IsMetroCity,data=hotel.df,alternative="less")

##   
## Welch Two Sample t-test  
##   
## data: RoomRent by IsMetroCity  
## t = 10.721, df = 13224, p-value = 1  
## alternative hypothesis: true difference in means is less than 0  
## 95 percent confidence interval:  
## -Inf 1253.463  
## sample estimates:  
## mean in group 0 mean in group 1   
## 5782.794 4696.073

* Since the p-value is more than 0.05, we fail to reject the null hypothesis that they are equal

#5.Average RoomRent in hotels having higher capacity is more than than hotels having less capacity  
t.test(hotel.df$RoomRent,hotel.df$HotelCapacity,alternative="less")

##   
## Welch Two Sample t-test  
##   
## data: hotel.df$RoomRent and hotel.df$HotelCapacity  
## t = 84.882, df = 13234, p-value = 1  
## alternative hypothesis: true difference in means is less than 0  
## 95 percent confidence interval:  
## -Inf 5516.352  
## sample estimates:  
## mean of x mean of y   
## 5473.99184 62.51164

* Since the p-value is less than 0.05, we can reject the null hypothesis that the mean are equal

# Regression Model

### 9. Generating Regression models using lm() model and testing hypothesis

#Regression Model  
#1. ##Generating A Multi Variable Linear Regressional Model for Hotel RoomRent  
  
linear1.mod<- lm(RoomRent~ HasSwimmingPool + StarRating + HotelCapacity -1, data = hotel.df)  
summary(linear1.mod)

##   
## Call:  
## lm(formula = RoomRent ~ HasSwimmingPool + StarRating + HotelCapacity -   
## 1, data = hotel.df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -8039 -2448 -1249 461 312401   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## HasSwimmingPool 3719.6943 148.7835 25.001 < 2e-16 \*\*\*  
## StarRating 1396.8746 26.1320 53.455 < 2e-16 \*\*\*  
## HotelCapacity -7.6598 0.9415 -8.136 4.44e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6813 on 13229 degrees of freedom  
## Multiple R-squared: 0.4457, Adjusted R-squared: 0.4456   
## F-statistic: 3546 on 3 and 13229 DF, p-value: < 2.2e-16

#Coefficients of the model  
coefficients(linear1.mod)

## HasSwimmingPool StarRating HotelCapacity   
## 3719.694300 1396.874562 -7.659814

## Model: salary = b0 + b1\*HasSwimmingPool + b2\*StarRating + b3\*HotelCapacity  
## b0 = -1(assumption), b1 = 3719.694300,b2= 1396.874562,b3=-7.659814  
## Model: salary = -1 + 3719.694300\*HasSwimmingPool+1396.874562\*StarRating+ -7.659814\*HotelCapacity  
  
  
#2.  
  
linear2.mod<- lm(RoomRent~ IsWeekend + IsTouristDestination -1, data = hotel.df)  
summary(linear2.mod)

##   
## Call:  
## lm(formula = RoomRent ~ IsWeekend + IsTouristDestination - 1,   
## data = hotel.df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -6644 -2676 -156 2512 317844   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## IsWeekend 2286 109 20.98 <2e-16 \*\*\*  
## IsTouristDestination 4656 103 45.21 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7498 on 13230 degrees of freedom  
## Multiple R-squared: 0.3287, Adjusted R-squared: 0.3286   
## F-statistic: 3239 on 2 and 13230 DF, p-value: < 2.2e-16

#Coefficients of the model  
coefficients(linear2.mod)

## IsWeekend IsTouristDestination   
## 2286.297 4656.327

## Model: salary = b0 + b1\*IsWeekend + b2\*IsTouristDestination  
## b0 = -1(assumption), b1 = 2286.297,b2= 4656.327  
## Model: salary = -1 + 2286.297\*IsWeekend + 4656.327 \*IsTouristDestination  
  
  
#3.  
linear3.mod<- lm(RoomRent~ HasSwimmingPool + StarRating + HotelCapacity + Airport -1, data = hotel.df)  
summary(linear3.mod)

##   
## Call:  
## lm(formula = RoomRent ~ HasSwimmingPool + StarRating + HotelCapacity +   
## Airport - 1, data = hotel.df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -8240 -2380 -1224 384 312742   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## HasSwimmingPool 3903.7369 150.6728 25.909 < 2e-16 \*\*\*  
## StarRating 1248.4270 33.2220 37.578 < 2e-16 \*\*\*  
## HotelCapacity -6.7434 0.9482 -7.112 1.20e-12 \*\*\*  
## Airport 18.8697 2.6157 7.214 5.73e-13 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6800 on 13228 degrees of freedom  
## Multiple R-squared: 0.4479, Adjusted R-squared: 0.4477   
## F-statistic: 2683 on 4 and 13228 DF, p-value: < 2.2e-16

#Coefficients of the model  
coefficients(linear3.mod)

## HasSwimmingPool StarRating HotelCapacity Airport   
## 3903.736921 1248.426988 -6.743354 18.869726

## Model: salary = b0 + b1\*HasSwimmingPool + b2\*StarRating + b3\*HotelCapacity+b4\*Airport  
## b0 = -1(assumption), b1 = 3903.736921 b2=1248.426988 b3=-6.743354 b4= 18.869726   
## Model: salary = -1 + 3903.736921 \*HasSwimmingPool +1248.426988\*StarRating+ -6.743354\*HotelCapacity +18.869726 \*AirPort