DATA SCIENCE PROJECT HYATT GROUP OF HOTELS

MOO6-GROUP 3

Mudita Humar| Neha Majety| Sai Santosh Reddy Atla|
Sharathchandra Bangalore Munibaire Gowda

INTRODUCTION

Customer satisfaction plays an important role for the hotels like any other business in the hospitality industry. Not only is it the leading indicator to measure customer loyalty, identify unhappy customers, and increase revenue; it is also a key point that helps you to attract new customers in competitive business environments. Customer satisfaction is the best indicator to measure the success of the business and encourage return customers.

Customers feedback is a critical indicator for the reputation of hotels, eventually influencing the revenue of hotels through word of mouth. With the increase in websites that provide reviews on hotels, customer satisfaction has become an integral part of the business. Owing to the advancement of technology, there are many data mining tools that help in surveying customer data effectively and in the least amount of time.

OBJECTIVE

Hyatt group projects had stored around 13 GB of customer data which was given to various data analyst to study the data and identify factors that can help the Hyatt Hotel Chain to improve the customer experience they deliver to customers in the future. Customer experience is estimated through the calculation of Net Promoter Score, or NPS. The NPS assesses to what extent a respondent would recommend a certain company, product or service to his friends, relatives, or colleagues.

SCOPE

The dataset contains about 3 Million responses collected from the Hyatt Customer Survey from a time span of Feb 2014 to Jan 2015. We have considered all the 12 months of data. The data was appropriately cleaned and leveraged to gain insight into the Customer ratings for different factors concerning the hotel. In the document, we describe our understanding of the project, the questions that arose from the data and how we have answered them, the models we used to analyze the data and the implementation. We conclude by presenting some recommendations to help the Hyatt Chain of Hotels improve their Net Promoter Score.

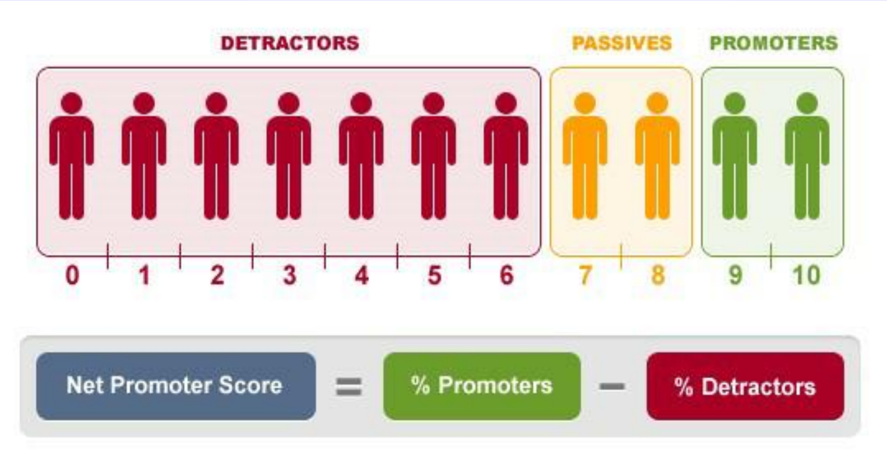
There are about 240 columns for each observation, and these columns contain data points about the customers such as purpose of visit, gender, guest title, preferred language, country to which they belong and some attributes about the hotel (example: Location, amenities and Type) and whether the responder is promoter, passive or detractor where if a customer is likely to be recommend the hotel, he is a “promoter” or else a “detractor”, with complacent people being labelled as “passive”.

• WHAT IS NPS?

Net Promoter Score, or NPS, measures customer experience and predicts business growth. This proven metric transformed the business world and now provides the core measurement for customer experience management programs the world round. Because NPS is a leading indicator for growth, it is an excellent tool to improve the customer experience management program(CEM). NPS alone is not sufficient to analyze customer experience but when supplemented with other metrics and insights from the customer's experience, it is possible to develop a comprehensive, actionable view of the program.

NPS CALCULATION:

In this project, we have taken into consideration the column 'NPS_TYPE' for our NPS calculations. We calculated the No of Promoters, passive and detractors for all countries.



The NPS Scale is divided into 3 categories:

- **Promoters** (score 9-10) are loyal enthusiasts who will keep buying and refer others, fueling growth.
- **Passives** (score 7-8) are satisfied but unenthusiastic customers who are vulnerable to competitive offerings.
- **Detractors** (score 0-6) are unhappy customers who can damage your brand and impede growth through negative word-of-mouth.

NPS can be calculated using the following formula:
Actual NPS = ((No. of Promoters – No. of Detractors) / (No. of respondents)) * 100

We can see the number of customers for each NPS TYPE from the following chart. As we can see, majority of the customers are likely to recommend the hotel to others.

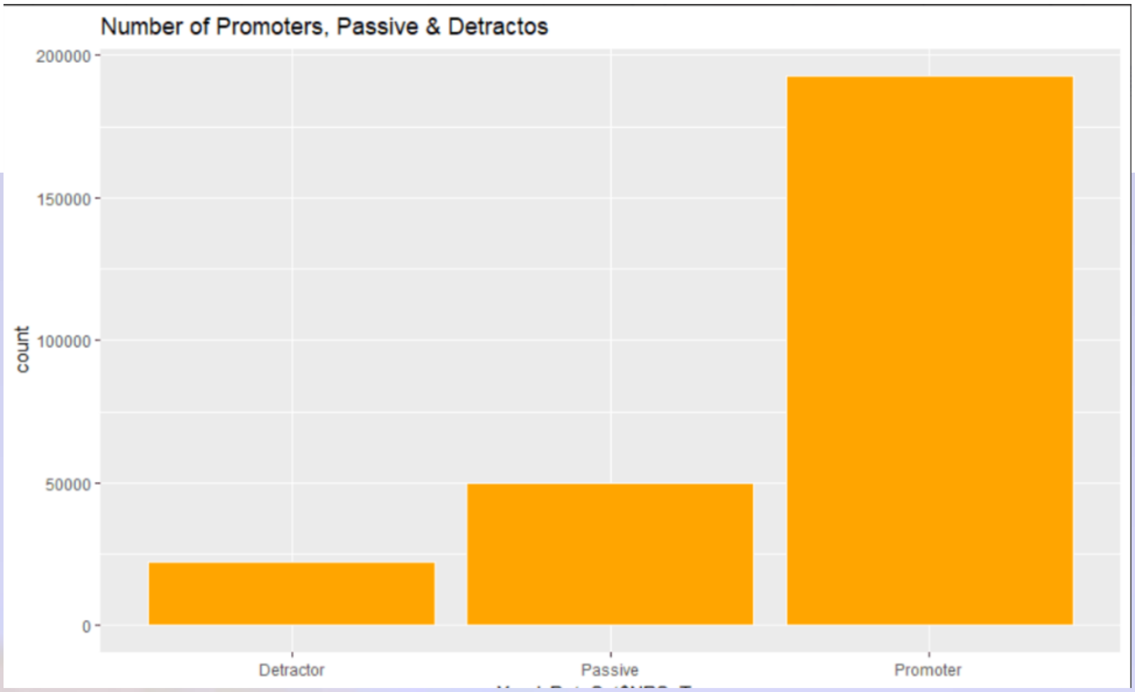


Fig 1.1 NPS Calculation

```
ggplot(data=YearlyDataSet,aes(x=YearlyDataSet$NPS_Type))+geom_bar(stat="count",colour="white",
fill="orange") + ggtitle("Number of Promoters, Passive & Detractors")
```

• DATA PREPARATION

EXTRACTION, TRANSFORMATION & LOADING

```
#loading data
FebDataHyattHotel <- fread(("out-201402.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
MarchDataHyattHotel <-  fread(("out-201403.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
AprilDataHyattHotel <-  fread(("out-201404.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
MayDataHyattHotel <-  fread(("out-201405.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
JuneDataHyattHotel <-  fread(("out-201406.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
JulyDataHyattHotel <-  fread(("out-201407.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
AugDataHyattHotel <-  fread(("out-201408.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
SepDataHyattHotel <-  fread(("out-201409.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
```

```
OctDataHyattHotel <-  fread(("out-201410.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
NovDataHyattHotel <-  fread(("out-201411.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232))
DecDataHyattHotel <- na.omit(fread(("out-201412.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232)))
JanDataHyattHotel <- na.omit(fread(("out-201501.csv"), select =
c(19,55,109,107,108,110,137:145,147,167,171,168,131,103,127,182,191,189,200:223,232)))
gc()

colnames(YearlyDataSet)
#omiting NA's
NovDataHyattHotel <- na.omit(NovDataHyattHotel)
OctDataHyattHotel <- na.omit(OctDataHyattHotel)
SepDataHyattHotel <- na.omit(SepDataHyattHotel)
AugDataHyattHotel <- na.omit(AugDataHyattHotel)
JulyDataHyattHotel <- na.omit(JulyDataHyattHotel)
JuneDataHyattHotel <- na.omit(JuneDataHyattHotel)
MayDataHyattHotel <- na.omit(MayDataHyattHotel)
AprilDataHyattHotel <- na.omit(AprilDataHyattHotel)
MarchDataHyattHotel <- na.omit(MarchDataHyattHotel)
FebDataHyattHotel <- na.omit(FebDataHyattHotel)



#combing data of all months
YearlyDataSet <-
rbind(JanDataHyattHotel,FebDataHyattHotel,MarchDataHyattHotel,AprilDataHyattHotel,MayDataHyattHotel,
JuneDataHyattHotel,JulyDataHyattHotel,AugDataHyattHotel,SepDataHyattHotel,OctDataHyattHotel,NovData
HyattHotel,DecDataHyattHotel)
row.names(YearlyDataSet) <- NULL
```

• BUSINESS QUESTIONS

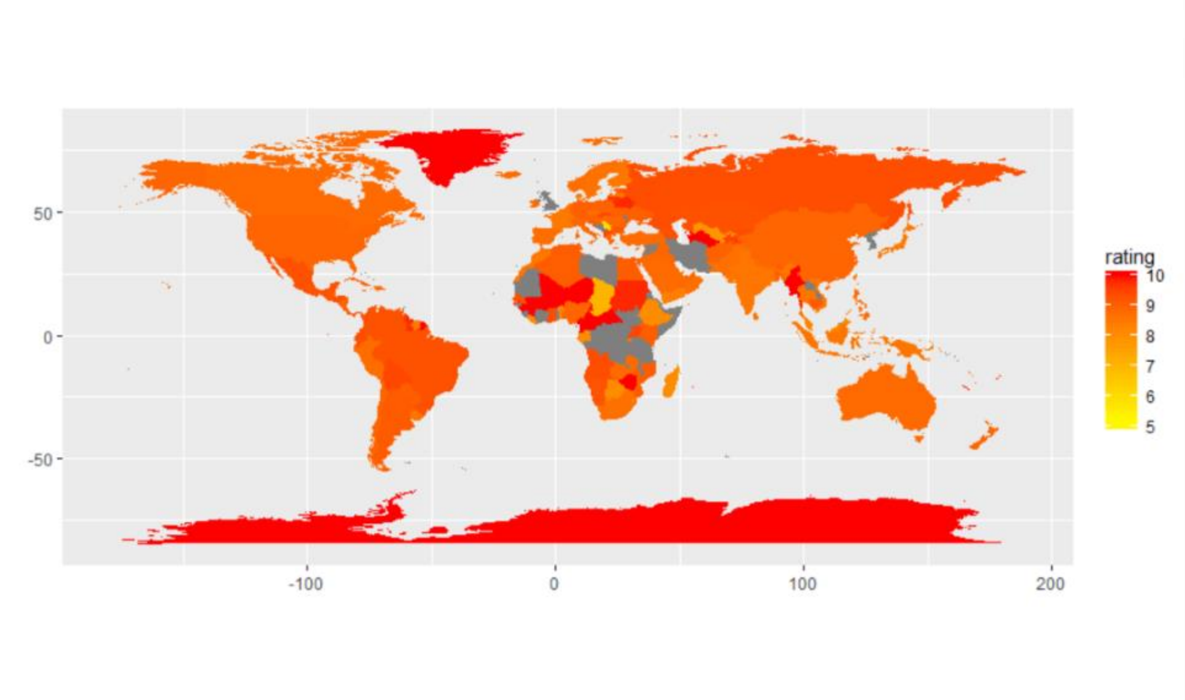
- Interesting patterns or association among columns affecting NPS Score?
- Which states with Hyatt hotels have the highest number of detractors?
- Which of the amenities play a crucial role in affecting the NPS?
- How does overall satisfaction affect Likelihood to recommend across Customer type?
- Which is the most popular brand among the Hyatt chain of hotels?

- ## UNDERSTANDING THE DATA

In order to get a better understanding of the data, we have analyzed some general as well as specific factors given in the dataset like age, gender, length of stay, purpose of visit etc. We have taken values from all the countries in the beginning to get an idea of whole picture and narrowed it down later on in the process for more specific analysis of the dataset.

DATA ANALYSIS:

We mapped the average rating given by customers for all countries and observed that some countries in Africa and Middle east received low ratings. The ratings vary across different locations. It is also important to note that, Antarctica has the highest customer rating of 10. This is because there was only one customer from Antarctica, which happened to give a rating of 10/10. We must carefully interpret the data to avoid any misleading conclusions.



7.1 Likelihood to Recommend Vs the World

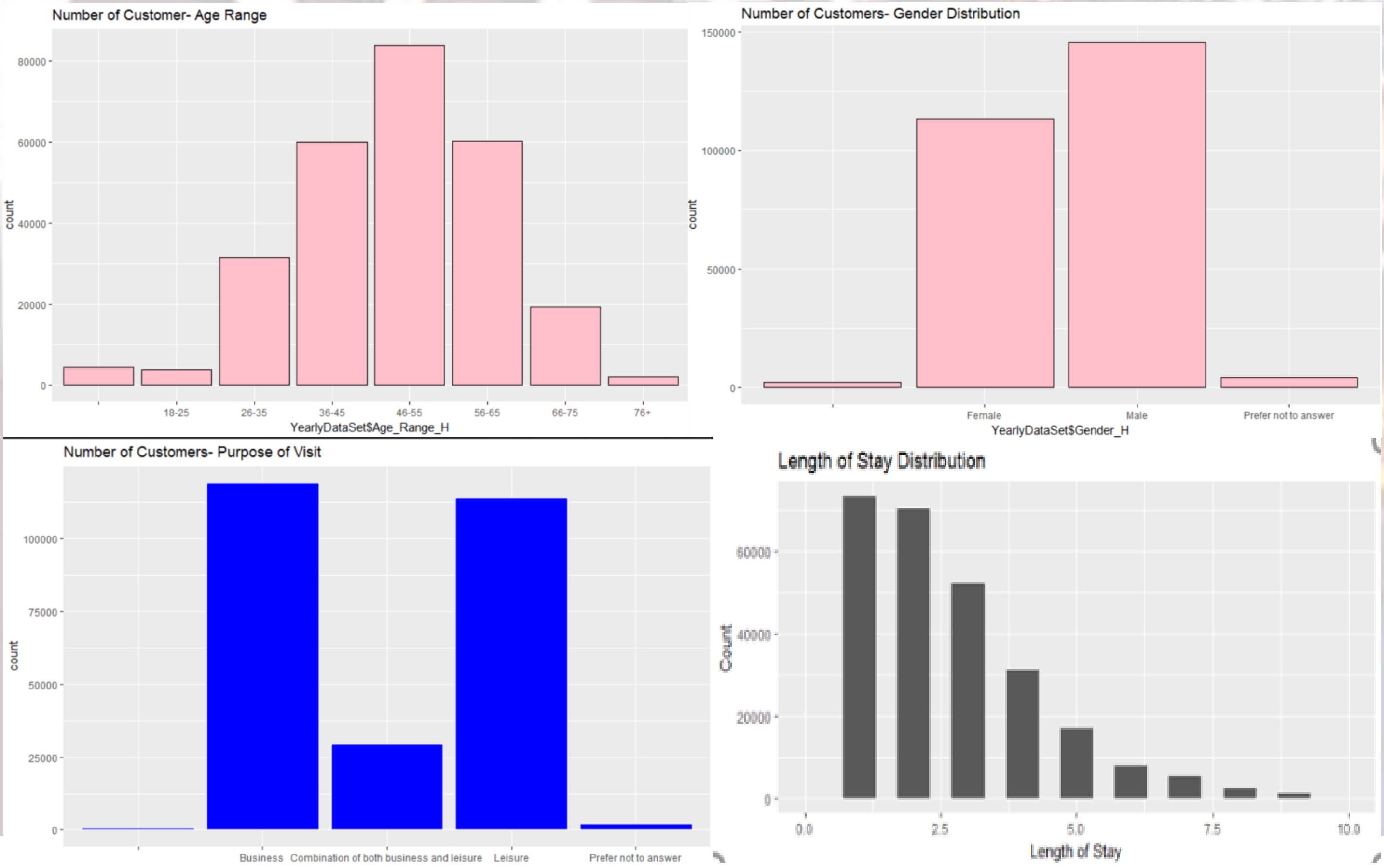
```
#World Map
worldMap <- map_data(map = "world")
SurveyQuestions <- YearlyDataSet
countryRating <-
as.data.frame(tapply(SurveyQuestions$Likelihood_Recommend_H,SurveyQuestions$Guest_Country_H,
mean))
colnames(countryRating) <- c("rating")
countryRating$country <- row.names(countryRating)
rownames(countryRating) <- NULL
worldMap$rating <- sapply(1:nrow(worldMap), function(i) countryRating$rating[countryRating$country
== worldMap$region[i]])
worldMap$rating <- as.numeric(worldMap$rating)
```

```
#Generating the world map based on the ratings. Yellow represents low ratings and red represents high ratings.
ggplot() + geom_map(data = worldMap, map = worldMap,aes(map_id = region, x = long, y = lat, fill = rating)) + scale_fill_gradient(low = "yellow", high = "red", guide = "colorbar") + coord_equal()
```

7.2: Demographic Analysis:

To understand the demographic distribution of our data, we carried out an analysis on the data to gain insight into the characteristics of the population. The following were the conclusions:

1. We can infer from Image 1 that the population from age range 36-45, 46-55 and 56-65 were prominent customers for the hotel. This gave us insight into what demographic we should target in our analysis.
2. From Image3, we see that majority of customers came to the hotel for either business or leisure purposes. Hence, the hotel must focus on the services we provide to customers who have come to the hotels with that purpose. Hence customer amenities would be important factors in our analysis.
3. Image 4 shows that The Length of Stay distribution revealed that most of the customers stay at the hotel for 1-3 days. So whichever facilities they would use, they would have to accommodate between these number of days.

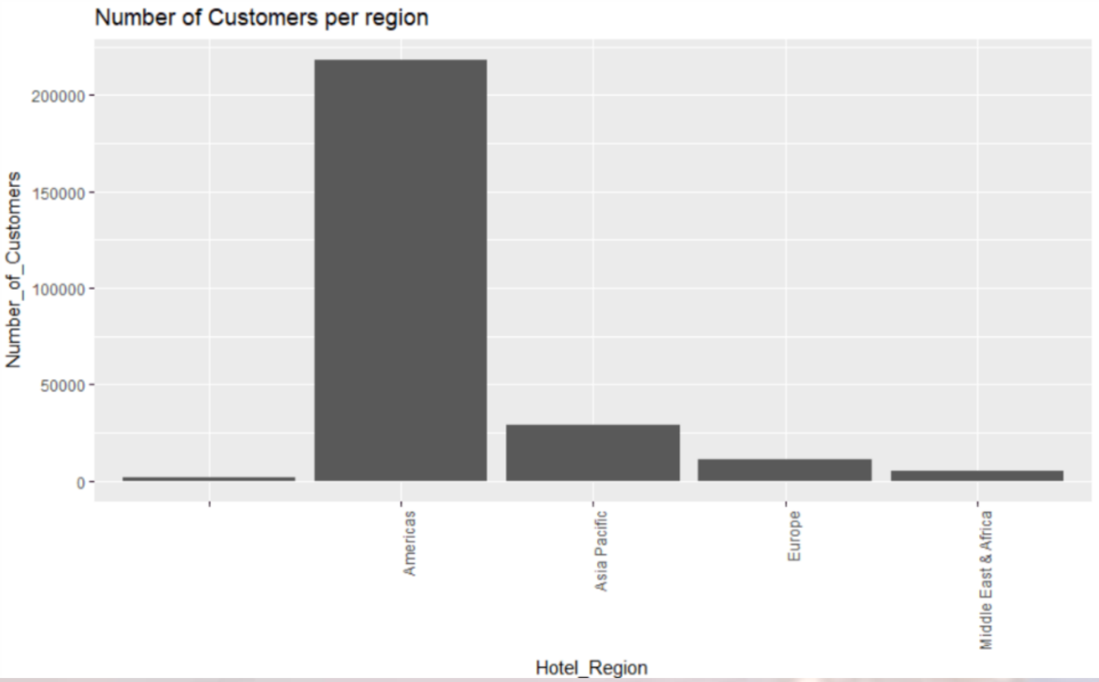


7.2 Demographic Analysis

```
#"Number of Customers- Gender Distribution"
ggplot(data=YearlyDataSet, aes(x=YearlyDataSet$Gender_H ))+geom_bar(stat = "count",colour="black", fill="pink") + ggtitle("Number of Customers- Gender Distribution")

#"Number of Customer- Age Range"
ggplot(data=YearlyDataSet, aes(x=YearlyDataSet$Age_Range_H ))+geom_bar(stat = "count",colour="black", fill="pink") + ggtitle("Number of Customer- Age Range")

#"Number of Customers- Purpose of Visit"
ggplot(data=YearlyDataSet, aes(x=YearlyDataSet$POV_H ))+geom_bar(stat = "count",colour="white",fill="Blue") + ggtitle("Number of Customers- Purpose of Visit")
#"Number of Customers- Length of Stay"
ggplot(data=YearlyDataSet, aes(x=YearlyDataSet$LENGTH_OF_STAY_C ))+geom_bar(stat = "count",colour="white",fill="green") + ggtitle("Number of Customers- Length of Stay")
```

7.3: Demographic distribution based on region:

We analyzed the data to see which region has the most number of customers. From the below graph, we found that most of the customers are from the Americas, followed by Asia Pacific and Europe

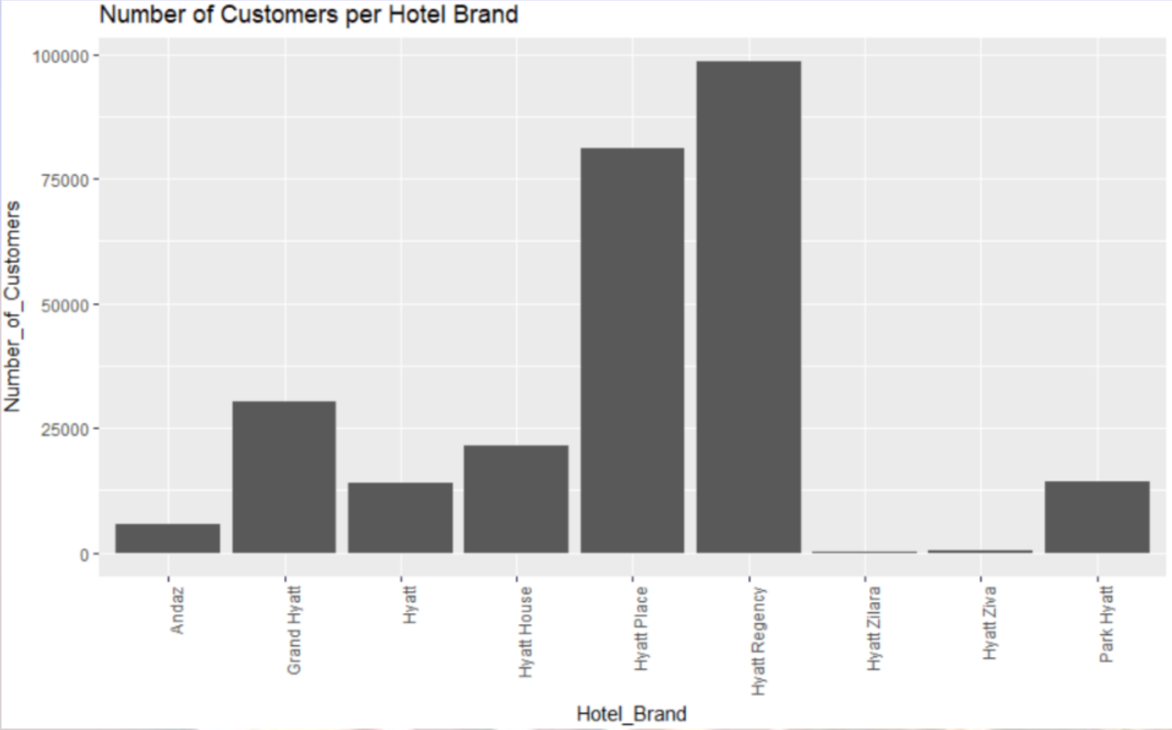


7.3 Regional Analysis

```
#"Number of Customers per region"
hotelRegion <- data.frame(table(YearlyDataSet$Region_PL), stringsAsFactors=FALSE)
colnames(hotelRegion) <- c("Hotel_Region", "Number_of_Customers")
hotelRegion$Hotel_Region <- as.character(hotelRegion$Hotel_Region)
hotelRegion$Number_of_Sample <- as.numeric((hotelRegion$Number_of_Customers))
g2 <- ggplot(data=hotelRegion, aes(x=Hotel_Region, y=Number_of_Customers))
g2 <- g2 + geom_bar(stat="identity")
g2 <- g2 + theme(axis.text.x=element_text(angle=90, hjust=1)) + ggtitle("Number of Customers per region")
g2
```

7.4: Popularity of Hyatt Brands:

Hyatt has several brands. We observed that Hyatt Regency is the most popular among the customers of our dataset. About a hundred thousand customers were from Hyatt Regency. The second most popular is the Hyatt Place, with about 80,000 customers. Hyatt Zilara and Hyatt Ziva attracted a few hundred customers.

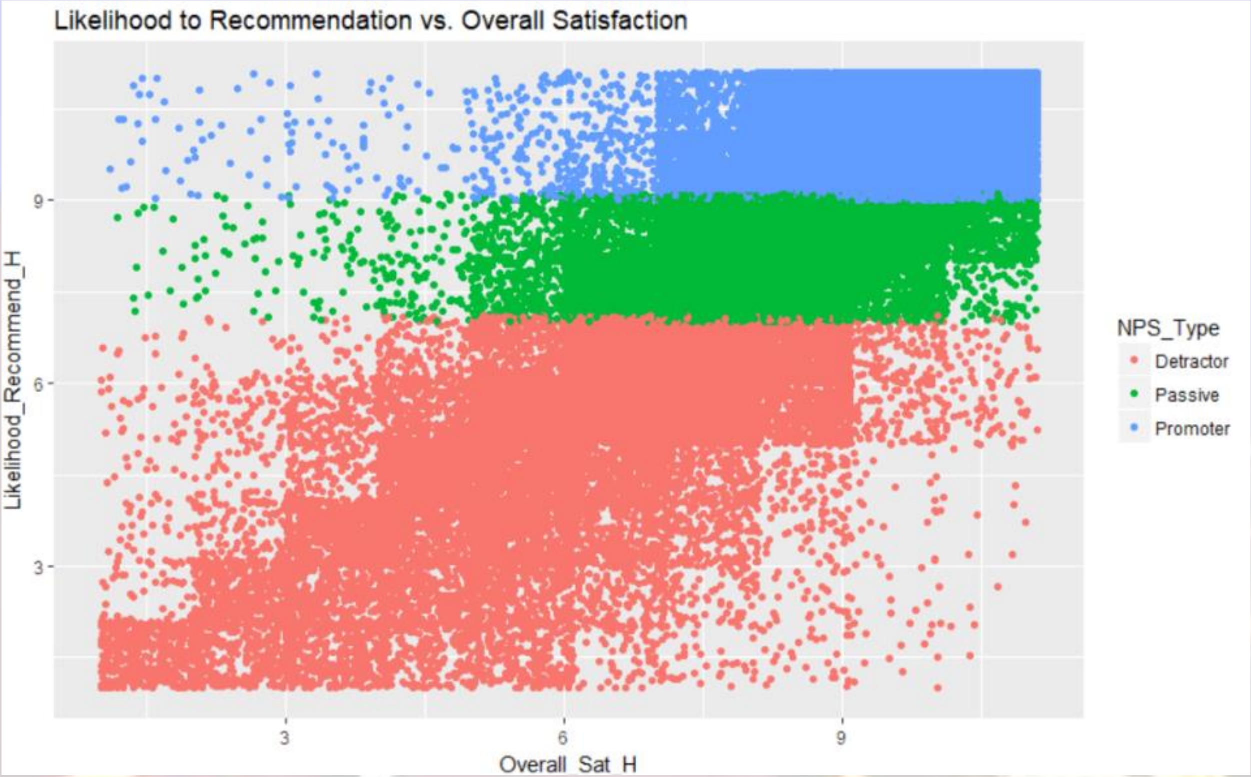


7.4 Brand Analysis

```
#"Number of Customers per Hotel Brand"
hotelBrand <- data.frame(table(YearlyDataSet$Brand_PL))
colnames(hotelBrand) <- c("Hotel_Brand", "Number_of_Customers")
g <- ggplot(data=hotelBrand, aes(x=Hotel_Brand, y=Number_of_Customers))
g <- g + geom_bar(stat="identity")
g <- g + theme(axis.text.x=element_text(angle=90, hjust=1)) + ggtitle("Number of Customers per Hotel Brand")
g
```

7.5: Correlation between LTR and Overall Satisfaction:

The following scatter plot shows the correlation between Likelihood to Recommend and Overall satisfaction for each of the NPS type i.e. Promoter, Passive and Detractor. This plot helps us in analyzing the effect of overall satisfaction on Likelihood to recommend. In the below chart we can see that the points lie very close to the line of best fit for both the promoters and passive which proves that the correlation between the variables is very strong for these customer types.

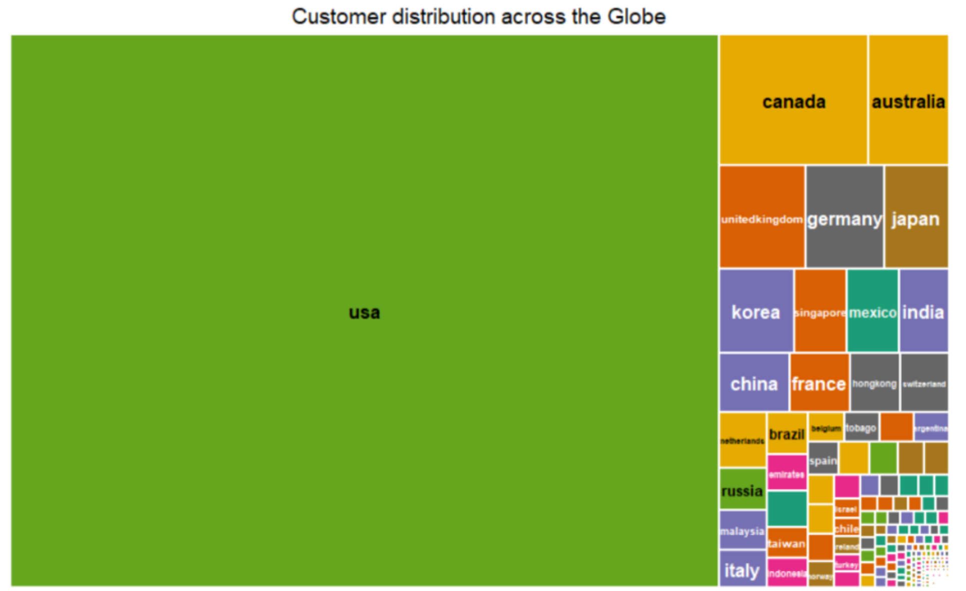


.5 Correlation Analysis

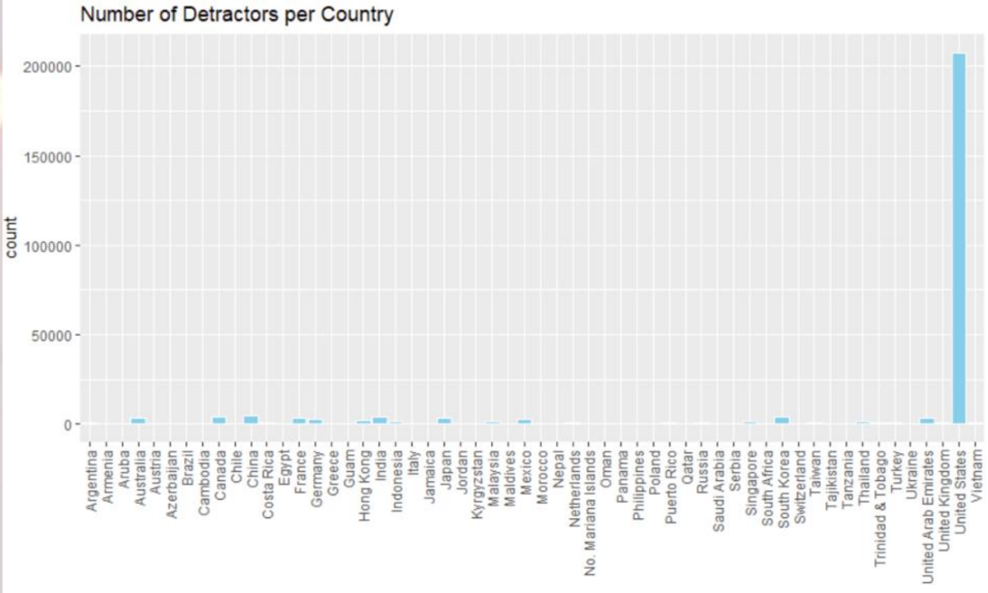
```
cor(YearlyDataSet$Likelihood_Recommend_H, YearlyDataSet$Overall_Sat_H)
# Scatter plot: x=overall satisfaction, y=likelihood to recommendation, color=NPS type
Scaterplot <- YearlyDataSet[, c("Likelihood_Recommend_H","Overall_Sat_H","NPS_Type")]
Scaterplot1$Likelihood_Recommend_H <- Scaterplot$Likelihood_Recommend_H +
runif(nrow(Scaterplot), min=0, max=1.1)
Scaterplot$Overall_Sat_H <- Scaterplot$Overall_Sat_H + runif(nrow(Scaterplot), min=0, max=1.1)
g3 <- ggplot(data=Scaterplot, aes(x=Overall_Sat_H))
g3 <- g3 + geom_point(aes(y=Likelihood_Recommend_H, color=NPS_Type))
g3 <- g3 + ggtitle("Likelihood to Recommendation vs. Overall Satisfaction")
g3
```

7.6: Why United States of America?

The following charts show that the number of customers were highest in USA. We also observed that the number of detractors where highest for USA. Hence, we decided to use USA to analyze the data so that the analytical output of this project can be applied to the majority of the customers.



7.6Text mining for all countries



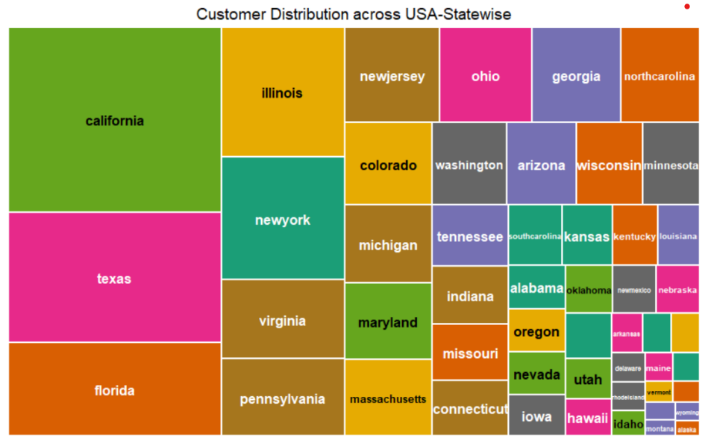
7.6.1 Detractors for all countries

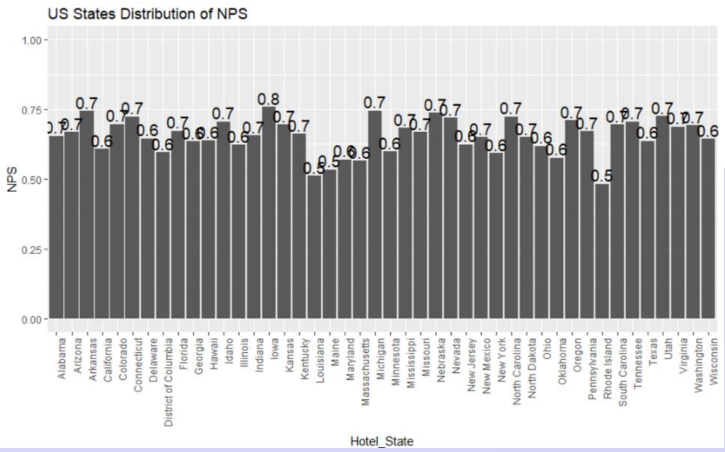
Since United States had highest number of customers and also highest detractors, we decided to further explore states in United States and their effect on overall NPS for United States

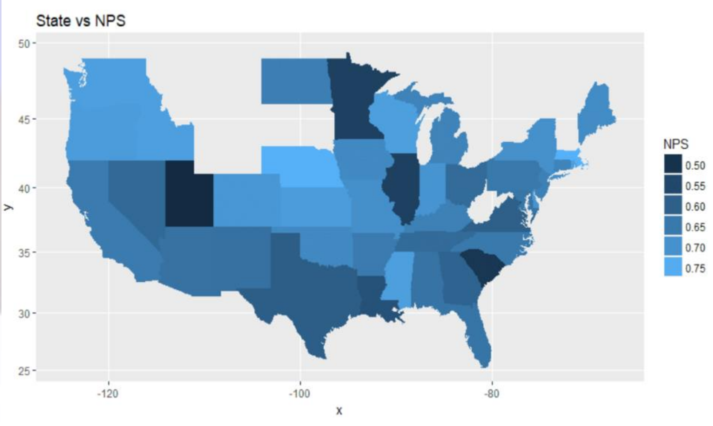
```
SpecifiedFreqMatrix <- function(x){
  VectorSource <- VectorSource(x)
  WordCorpus <- Corpus(VectorSource)
  TDM <- TermDocumentMatrix(WordCorpus)
  WordMatrix <- as.matrix(TDM)
  WordCount <- rowSums(WordMatrix)
  WordCount <- sort(WordCount,decreasing = TRUE)
  CloudFrame <- data.frame(Name=names(WordCount),freq=WordCount)
  return(CloudFrame)
}
YearlyDataSet$NPS_Type <- as.character(YearlyDataSet$NPS_Type)

UserInfoDataSetYearly <- subset(YearlyDataSet,select = c(1,10,2,4,50))
UserInfoDataSetYearly$StateofCustomerFullName <- abbr2state(UserInfoDataSetYearly$STATE_R)

UserInfoDataSetYearly$StateofCustomerFullName <-
as.character(UserInfoDataSetYearly$StateofCustomerFullName)
UserInfoDataSetYearly$CountryofCustomer <- as.character(UserInfoDataSetYearly$Guest_Country_H)
UserInfoDataSetYearly$StateofCustomer <- as.character(UserInfoDataSetYearly$STATE_R)

UserInfoDataSetYearly$CountryofCustomer <- str_replace(UserInfoDataSetYearly$CountryofCustomer,"
","")
UserInfoDataSetYearly$StateofCustomerFullName <-
str_replace_all(UserInfoDataSetYearly$StateofCustomerFullName," ","")

StateCloudFrame <- SpecifiedFreqMatrix(UserInfoDataSetYearly$StateofCustomerFullName)
UserStateTreeMap<- treemap(StateCloudFrame,index = c("Name"),vSize="freq",type="index",palette =
"Dark2",title = "Customer Distribution across USA-Statewise",fontsize.title = 14,fontsize.labels =
12,border.col = "white")

CountryCloudFrame <- SpecifiedFreqMatrix(UserInfoDataSetYearly$CountryofCustomer)
UserCountryTreeMap<- treemap(CountryCloudFrame,index = c("Name"),vSize="freq",type="index",palette =
"Dark2",title = "Customer distribution across the Globe",fontsize.title = 14,fontsize.labels = 12,border.col =
"white")
```

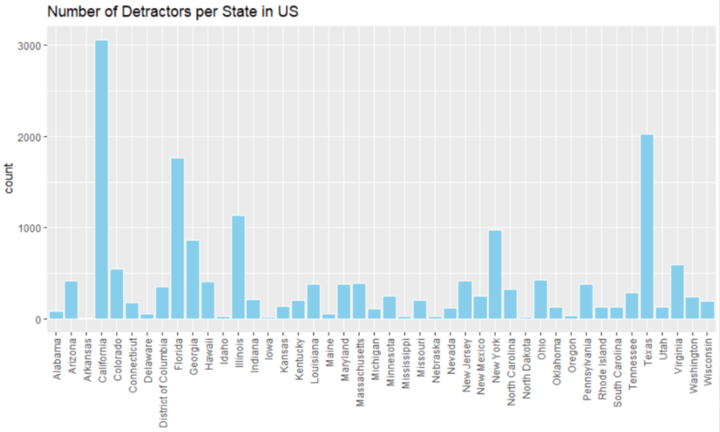
7.7: Why California?

After deciding that we are going to use United States based on their frequency of occurrence in the given data set, we found out that California is the state with highest number of customers followed by Texas. We also observed that the number of detractors is highest for California with 3060 followed by Texas with 2029 and it also has a lower NPS value of 0.6. Although the bar chart and heat map show that both California and Texas have the same NPS value with 0.6 rounded, we have chosen California for further analysis since it has higher number of customers, higher number of detractors and low NPS.







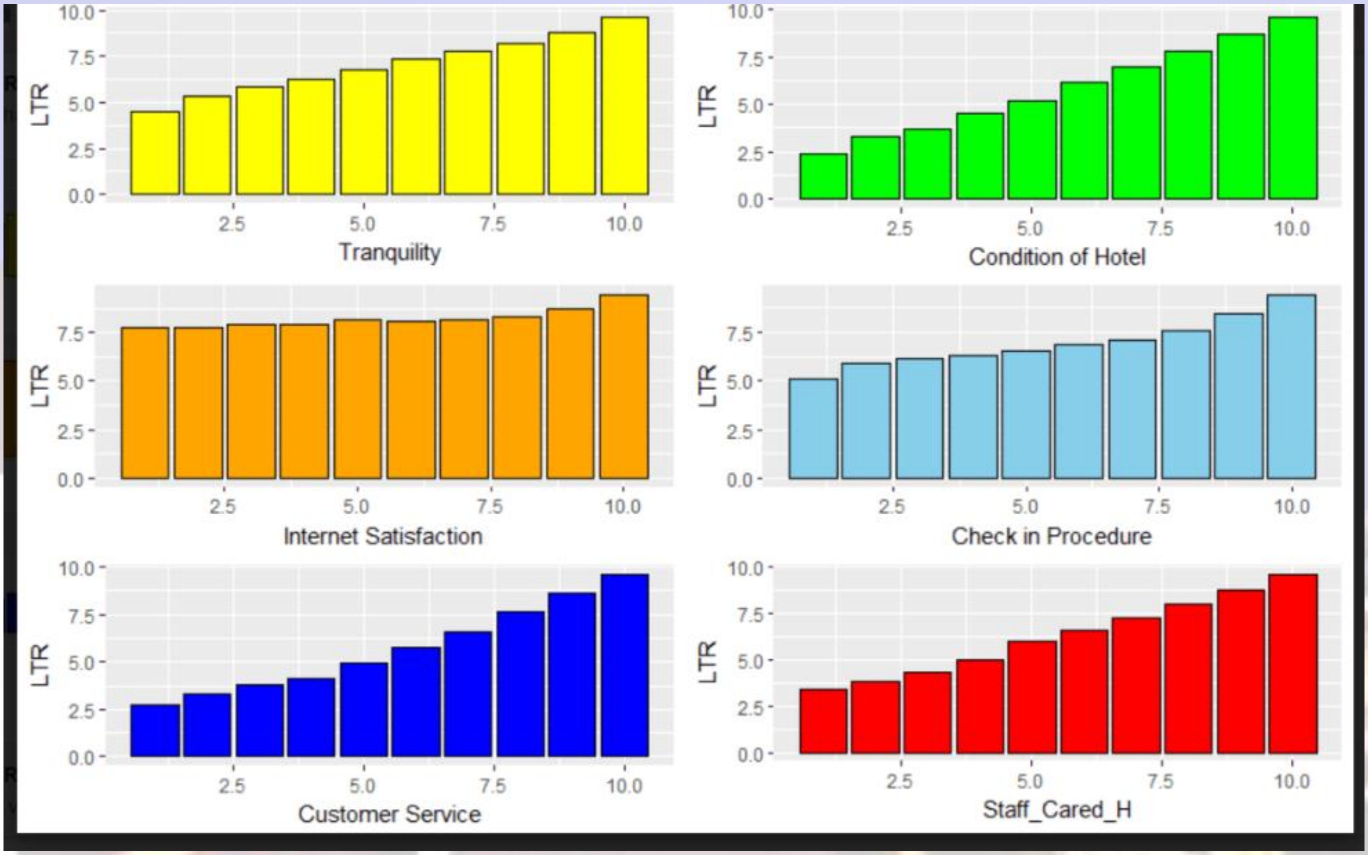


7.7.1 NPS values for all states in US

```
#"State vs NPS" Heat Map
us <- map_data("state")
map1 <- ggplot(NUSA, aes(map_id=NUSA$State_PL))+geom_map(map=us,aes(fill=NUSA$NPS))+
guides(fill=guide_legend(title = "NPS"))+expand_limits(x=us$long,y=us$lat)+labs(title="State vs NPS")+
coord_map()
map1

#"US States Distribution of NPS"
g4 <- ggplot(data=hotelUSStatesNPS, aes(x=Hotel_State, y=NPS))
g4 <- g4 + geom_bar(stat="identity")
g4 <- g4 + geom_text(aes(x=Hotel_State, y=NPS+0.05*mean(NPS), label=round(NPS,1)), size=5)
g4 <- g4 + ggtitle("US States Distribution of NPS")
g4 <- g4 + ylim(0,1) + theme(axis.text.x=element_text(angle=90, hjust=1))
g4
ggplot(data=df5,aes(x=df5$State_PL))+geom_bar(stat="count",colour="white",
fill="skyblue")+theme(axis.text.x = element_text(angle = 90,hjust = 1,vjust = 0.5)) + ggtitle("Number of
Detractors per State in US")

#Detractors per State in USA
Detractor <- USA[USA$NPS_Type=="Detractor"]
barplot(tapply(Detractor$NPS_Type,Detractor$State_PL,length))
```

7.8: Likelihood to Recommend Vs other rated parameters?

It is crucial to know the relationship between the most important parameter i.e. likelihood to recommend and several other rated parameters for e.g. tranquility, condition of the hotel, internet satisfaction, check in process, customer service and the staff cared. The below graph visually helps us to understand the same.



7.8 LTR Vs other parameters

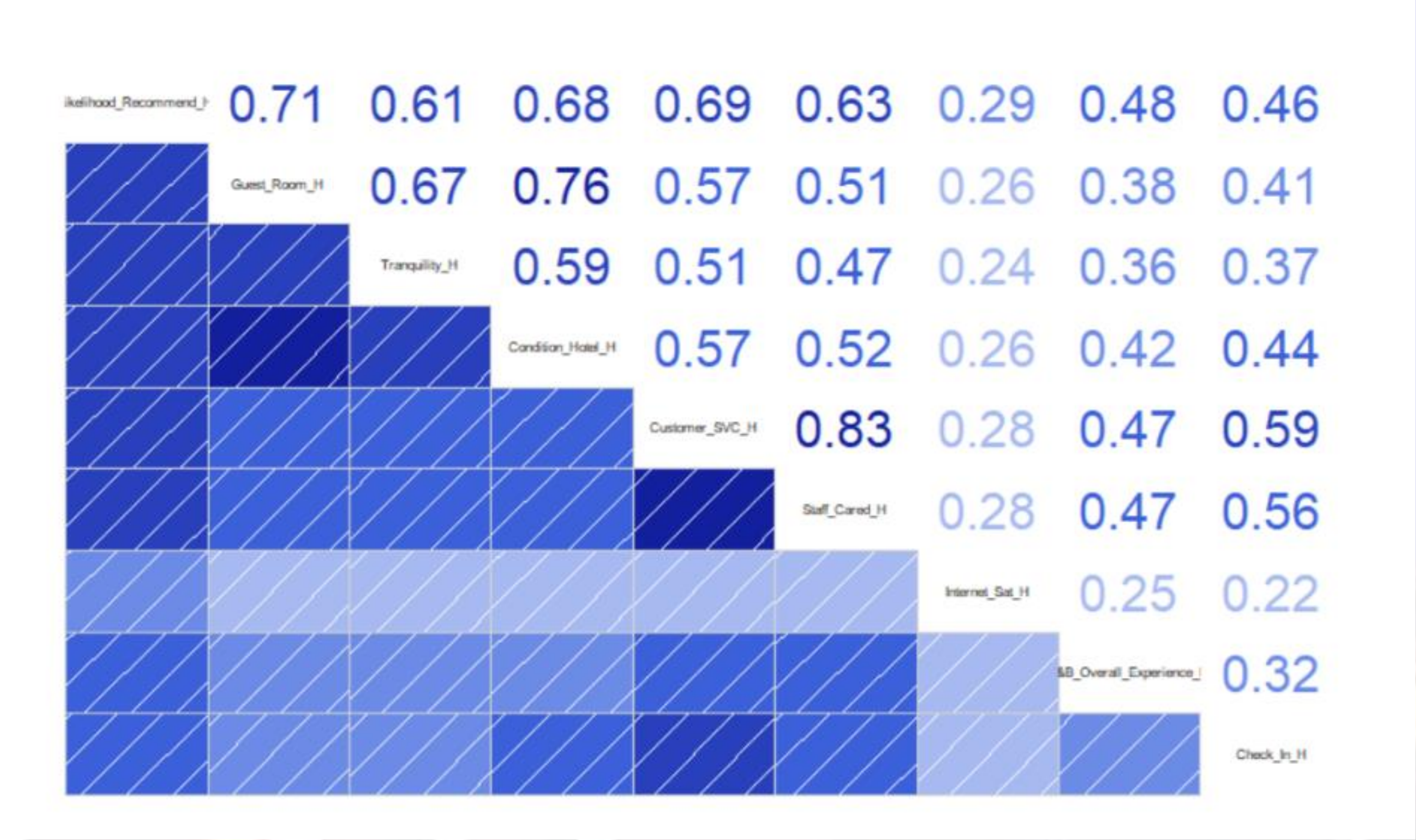
- More the ratings for tranquility, more is the likelihood to recommend
- Better the condition of the hotel, better are the ratings
- Internet satisfaction seems ambiguous in conclusion since whatsoever the quality of service, it affects the likelihood to recommend equally.
- Better the quality of check in process, better are the ratings for likelihood to recommend.
- Better the customer service, better the likelihood to recommend.
- The more caring the staff is, the more likely it is for customer to recommend the hotel.

```
t1<- ggplot(data=Cali, aes(x=as.numeric(Tranquility_H),y=Likelihood_Recommend_H)) +
  geom_bar(stat="summary",fun.data="mean_se", color="black", fill="yellow")  +
  labs(y="LTR",x="Tranquility")

t1
```

```
t2<- ggplot(data=Cali, aes(x=as.numeric(Condition_Hotel_H),y=Likelihood_Recommend_H)) +
  geom_bar(stat="summary",fun.data="mean_se", color="black", fill="green")  +
  labs(y="LTR",x="Condition of Hotel")
t2
t3<- ggplot(data=Cali, aes(x=as.numeric(Internet_Sat_H),y=Likelihood_Recommend_H)) +
  geom_bar(stat="summary",fun.data="mean_se", color="black", fill="orange") +
  labs(y="LTR",x="Internet Satisfaction")
t3
t4<- ggplot(data=Cali, aes(x=as.numeric(Check_In_H),y=Likelihood_Recommend_H)) +
  geom_bar(stat="summary",fun.data="mean_se", color="black", fill="#87ceeb")  +
  labs(y="LTR",x="Check in Procedure")
t4
t5<- ggplot(data=Cali, aes(x=as.numeric(Customer_SVC_H),y=Likelihood_Recommend_H)) +
  geom_bar(stat="summary",fun.data="mean_se", color="black", fill="blue") +
  labs(y="LTR",x="Customer Service")
t5
t6<- ggplot(data=Cali, aes(x=as.numeric(Staff_Cared_H),y=Likelihood_Recommend_H)) +
  geom_bar(stat="summary",fun.data="mean_se",color="black", fill="red") +
  labs(y="LTR",x="Staff_Cared_H")
t6
library(gridExtra)
grid.arrange(t1,t2,t3,t4,t5,t6)
```

7.9: CORRGRAM:

Using correlation, we found out metrics which affected the NPS most by assuming that LTR is directly proportional to NPS. With this corrgram, we can understand how LTR is influenced by other survey question variables like guest room, tranquility, hotel condition, internet satisfaction etc.



```
correlationSurvey <- subset(Cali,select=c(13,15:22))
correlation<-cor(correlationSurvey)
correlationPlotYearly<-corrgram(correlation,upper.panel = panel.cor)
```

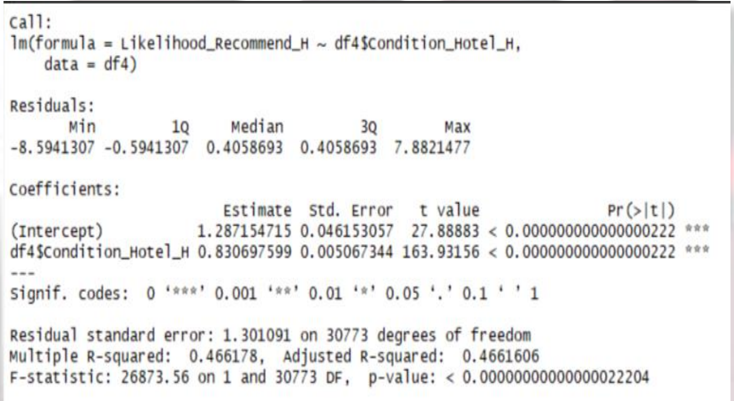
• MODELLING:

• LINEAR MODELLING:

We performed Linear Modelling by primarily focusing on detractors. That means, the likelihood to recommend (LTR)was our dependent variable. Following are the variables we have taken to calculate r-squared value for the dependent variable.

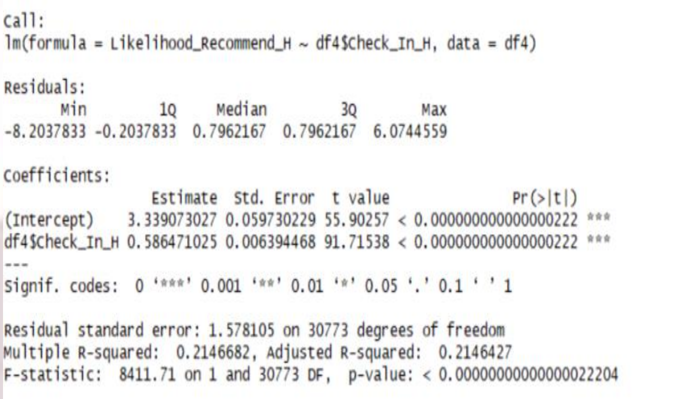
1. Condition of the Hotel
2. Quality of check in process
3. Food and beverage
4. Staff Caring factor
5. Customer Service
6. Guest room quality
7. Internet satisfaction
8. Staff service
9. Tranquility

Below is the output for the linear modelling model:



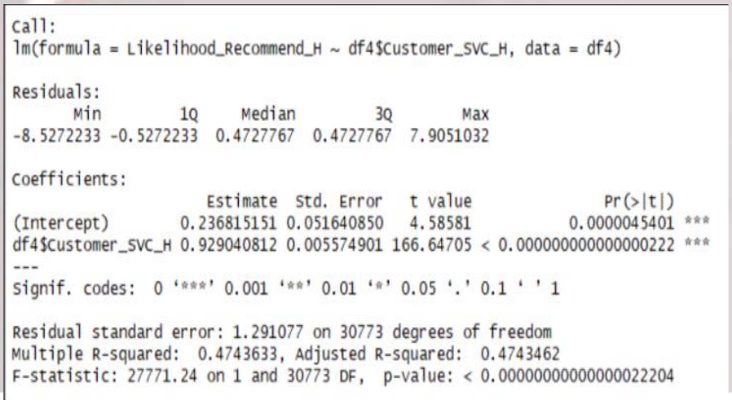
```
Call:
lm(formula = Likelihood_Recommend_H ~ df4$Condition_Hotel_H,
    data = df4)

Residuals:
       Min         1Q     Median         3Q        Max
-8.5941307 -0.5941307  0.4058693  0.4058693  7.8821477

Coefficients:
                        Estimate  Std. Error   t value                Pr(>|t|)
(Intercept)          1.287154715 0.046153057  27.88883 < 0.000000000000000222 ***
df4$Condition_Hotel_H 0.830697599 0.005067344 163.93156 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.301091 on 30773 degrees of freedom
Multiple R-squared:  0.466178,  Adjusted R-squared:  0.4661606
F-statistic: 26873.56 on 1 and 30773 DF,  p-value: < 0.00000000000000022204
```

8.1 LTR Vs Condition of the hotel



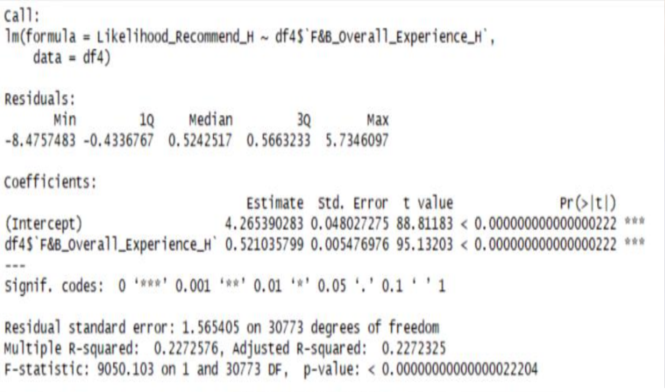
```
Call:
lm(formula = Likelihood_Recommend_H ~ df4$Check_In_H, data = df4)

Residuals:
       Min         1Q     Median         3Q        Max
-8.2037833 -0.2037833  0.7962167  0.7962167  6.0744559

Coefficients:
                  Estimate  Std. Error  t value                Pr(>|t|)
(Intercept)    3.339073027 0.059730229 55.90257 < 0.000000000000000222 ***
df4$Check_In_H 0.586471025 0.006394468 91.71538 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.578105 on 30773 degrees of freedom
Multiple R-squared:  0.2146682, Adjusted R-squared:  0.2146427
F-statistic:  8411.71 on 1 and 30773 DF,  p-value: < 0.00000000000000022204
```

8.2 LTR Vs Quality of the check in process



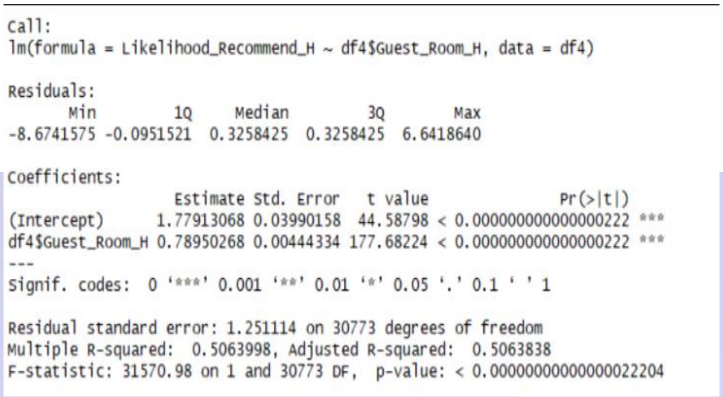
```
Call:
lm(formula = Likelihood_Recommend_H ~ df4$Customer_SVC_H, data = df4)

Residuals:
       Min         1Q     Median         3Q        Max
-8.5272233 -0.5272233  0.4727767  0.4727767  7.9051032

Coefficients:
                     Estimate  Std. Error   t value               Pr(>|t|)
(Intercept)       0.236815151 0.051640850   4.58581           0.0000045401 ***
df4$Customer_SVC_H 0.929040812 0.005574901 166.64705 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.291077 on 30773 degrees of freedom
Multiple R-squared:  0.4743633, Adjusted R-squared:  0.4743462
F-statistic: 27771.24 on 1 and 30773 DF,  p-value: < 0.00000000000000022204
```

8.3 LTR Vs Quality of the customer service



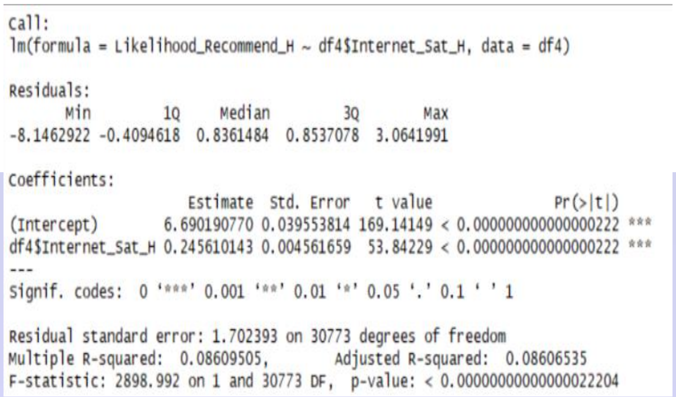
```
Call:
lm(formula = Likelihood_Recommend_H ~ df4$`F&B_Overall_Experience_H`,
    data = df4)

Residuals:
       Min         1Q     Median         3Q        Max
-8.4757483 -0.4336767  0.5242517  0.5663233  5.7346097

Coefficients:
                                  Estimate  Std. Error  t value                Pr(>|t|)
(Intercept)                    4.265390283 0.048027275 88.81183 < 0.000000000000000222 ***
df4$`F&B_Overall_Experience_H` 0.521035799 0.005476976 95.13203 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.565405 on 30773 degrees of freedom
Multiple R-squared:  0.2272576, Adjusted R-squared:  0.2272325
F-statistic: 9050.103 on 1 and 30773 DF,  p-value: < 0.00000000000000022204
```

8.4 LTR Vs Food &Beverage experience



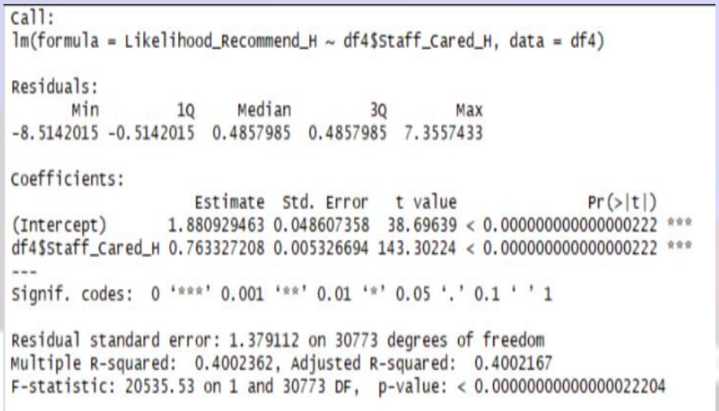
```
Call:
lm(formula = Likelihood_Recommend_H ~ df4$Guest_Room_H, data = df4)

Residuals:
      Min         1Q     Median         3Q        Max 
-8.6741575 -0.0951521  0.3258425  0.3258425  6.6418640 

Coefficients:
                   Estimate Std. Error   t value              Pr(>|t|)    
(Intercept)      1.77913068 0.03990158  44.58798 < 0.000000000000000222 ***
df4$Guest_Room_H 0.78950268 0.00444334 177.68224 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.251114 on 30773 degrees of freedom
Multiple R-squared:  0.5063998, Adjusted R-squared:  0.5063838 
F-statistic: 31570.98 on 1 and 30773 DF,  p-value: < 0.00000000000000022204
```

8.5 LTR Vs Guest room satisfaction



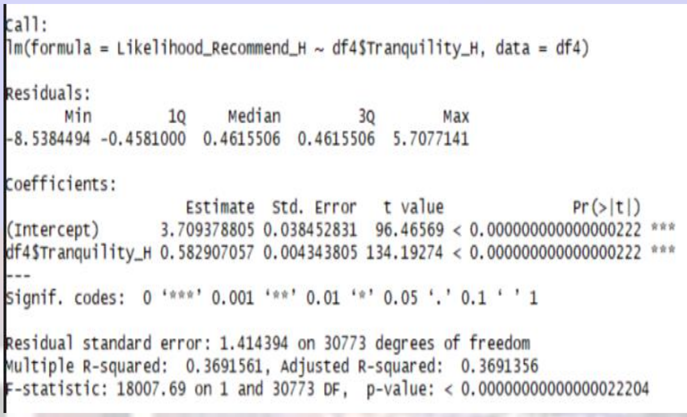
```
Call:
lm(formula = Likelihood_Recommend_H ~ df4$Internet_Sat_H, data = df4)

Residuals:
      Min         1Q     Median         3Q        Max 
-8.1462922 -0.4094618  0.8361484  0.8537078  3.0641991 

Coefficients:
                      Estimate  Std. Error   t value              Pr(>|t|)    
(Intercept)        6.690190770 0.039553814 169.14149 < 0.000000000000000222 ***
df4$Internet_Sat_H 0.245610143 0.004561659  53.84229 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.702393 on 30773 degrees of freedom
Multiple R-squared:  0.08609505,	Adjusted R-squared:  0.08606535 
F-statistic: 2898.992 on 1 and 30773 DF,  p-value: < 0.00000000000000022204
```

8.6 LTR Vs Internet satisfaction



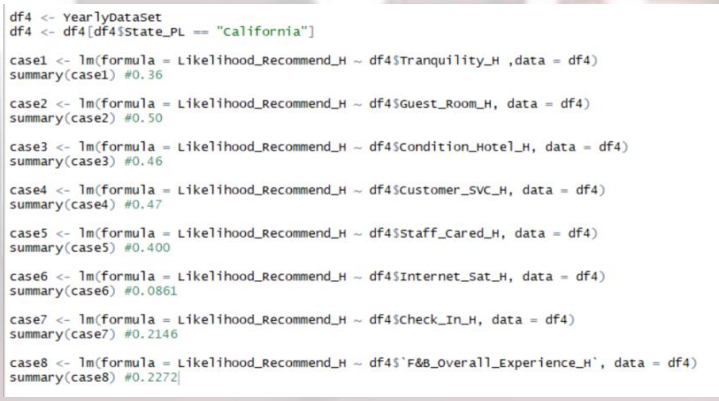
```
Call:
lm(formula = Likelihood_Recommend_H ~ df4$Staff_Cared_H, data = df4)

Residuals:
      Min         1Q     Median         3Q        Max 
-8.5142015 -0.5142015  0.4857985  0.4857985  7.3557433 

Coefficients:
                     Estimate  Std. Error   t value              Pr(>|t|)    
(Intercept)       1.880929463 0.048607358  38.69639 < 0.000000000000000222 ***
df4$Staff_Cared_H 0.763327208 0.005326694 143.30224 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.379112 on 30773 degrees of freedom
Multiple R-squared:  0.4002362, Adjusted R-squared:  0.4002167 
F-statistic: 20535.53 on 1 and 30773 DF,  p-value: < 0.00000000000000022204
```

8.7 LTR Vs Staff service



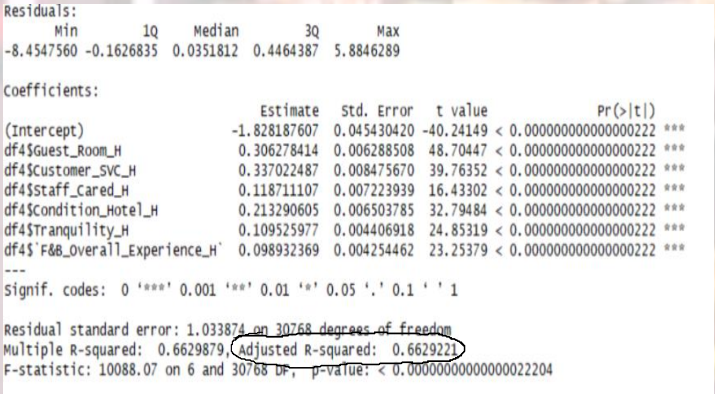
```
Call:
lm(formula = Likelihood_Recommend_H ~ df4$Tranquility_H, data = df4)

Residuals:
      Min         1Q     Median         3Q        Max 
-8.5384494 -0.4581000  0.4615506  0.4615506  5.7077141 

Coefficients:
                     Estimate  Std. Error   t value              Pr(>|t|)    
(Intercept)       3.709378805 0.038452831  96.46569 < 0.000000000000000222 ***
df4$Tranquility_H 0.582907057 0.004343805 134.19274 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.414394 on 30773 degrees of freedom
Multiple R-squared:  0.3691561, Adjusted R-squared:  0.3691356 
F-statistic: 18007.69 on 1 and 30773 DF,  p-value: < 0.00000000000000022204
```

8.8 LTR Vs tranquility



```
df4 <- YearlyDataSet
df4 <- df4[df4$State_PL == "California"]

case1 <- lm(formula = Likelihood_Recommend_H ~ df4$Tranquility_H ,data = df4)
summary(case1) #0.36

case2 <- lm(formula = Likelihood_Recommend_H ~ df4$Guest_Room_H, data = df4)
summary(case2) #0.50

case3 <- lm(formula = Likelihood_Recommend_H ~ df4$Condition_Hotel_H, data = df4)
summary(case3) #0.46

case4 <- lm(formula = Likelihood_Recommend_H ~ df4$Customer_SVC_H, data = df4)
summary(case4) #0.47

case5 <- lm(formula = Likelihood_Recommend_H ~ df4$Staff_Cared_H, data = df4)
summary(case5) #0.400

case6 <- lm(formula = Likelihood_Recommend_H ~ df4$Internet_Sat_H, data = df4)
summary(case6) #0.0861

case7 <- lm(formula = Likelihood_Recommend_H ~ df4$Check_In_H, data = df4)
summary(case7) #0.2146

case8 <- lm(formula = Likelihood_Recommend_H ~ df4$`F&B_Overall_Experience_H`, data = df4)
summary(case8) #0.2272
```

8.9 R-squared values for each model



```
Residuals:
      Min         1Q     Median         3Q        Max 
-8.4547560 -0.1626835  0.0351812  0.4464387  5.8846289 

Coefficients:
                                 Estimate  Std. Error  t value              Pr(>|t|)    
(Intercept)                  -1.828187607 0.045430420 -40.24149 < 0.000000000000000222 ***
df4$Guest_Room_H              0.306278414 0.006288508  48.70447 < 0.000000000000000222 ***
df4$Customer_SVC_H            0.337022487 0.008475670  39.76352 < 0.000000000000000222 ***
df4$Staff_Cared_H             0.118711107 0.007223939  16.43302 < 0.000000000000000222 ***
df4$Condition_Hotel_H         0.213290605 0.006503785  32.79484 < 0.000000000000000222 ***
df4$Tranquility_H             0.109525977 0.004406918  24.85319 < 0.000000000000000222 ***
df4$`F&B_Overall_Experience_H` 0.098932369 0.004254462  23.25379 < 0.000000000000000222 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.033874 on 30768 degrees of freedom
Multiple R-squared:  0.6629879, Adjusted R-squared:  0.6629221 
F-statistic: 10088.07 on 6 and 30768 DF,  p-value: < 0.00000000000000022204
```

8.10 Final Linear Model

```
#LM model for indivisual survey questions
case1 <- lm(formula = Likelihood_Recommend_H ~ Cali$Tranquility_H ,data = Cali)
summary(case1) #0.36

case2 <- lm(formula = Likelihood_Recommend_H ~ Cali$Guest_Room_H, data = Cali)
summary(case2) #0.50

case3 <- lm(formula = Likelihood_Recommend_H ~ Cali$Condition_Hotel_H, data = Cali)
summary(case3) #0.46

case4 <- lm(formula = Likelihood_Recommend_H ~ Cali$Customer_SVC_H, data = Cali)
```

```
summary(case4) #0.47

case5 <- lm(formula = Likelihood_Recommend_H ~ Cali$Staff_Cared_H, data = Cali)
summary(case5) #0.400
```

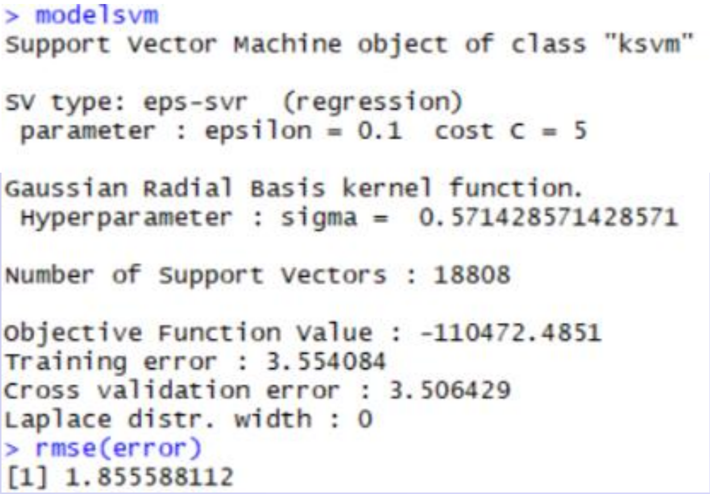
```
case6 <- lm(formula = Likelihood_Recommend_H ~ Cali$Internet_Sat_H, data = Cali)
summary(case6) #0.0861

case7 <- lm(formula = Likelihood_Recommend_H ~ Cali$Check_In_H, data = Cali)
summary(case7) #0.2146

case8 <- lm(formula = Likelihood_Recommend_H ~ Cali$`F&B_Overall_Experience_H`, data = Cali)
summary(case8) #0.2272

#Best LM model
case9 <- lm(formula = Likelihood_Recommend_H ~ Cali$Guest_Room_H + Cali$Customer_SVC_H  +
Cali$Staff_Cared_H + Cali$Condition_Hotel_H + Cali$Tranquility_H   +
Cali$`F&B_Overall_Experience_H`, data = Cali)
summary(case9)
#Check in and Internet satisfaction not affecting the lm value #0.6629
```

OBSERVATIONS:

- From fig 8.6, we observed that internet availability did not affect likelihood to recommend
- The top four variables influencing LTR are:
  Guest room, Customer Service, condition of hotel and Staff cared.
- Thus, the final linear model has 6 variables that influence LTR and the r-squared value is 0.67 which is closer to 1 and hence is a good model.

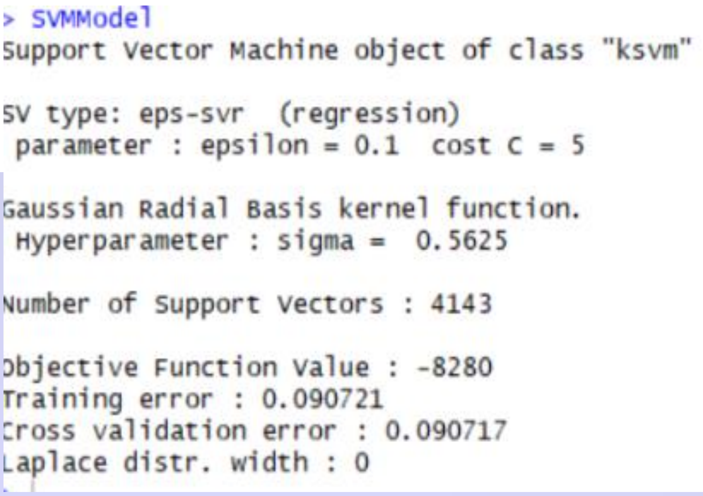
- ## KSVM:

  We used K Support Vector Machines (KSVM model) to create a classifier. This classifier will help us to see if a give a high value to the likelihood to recommend variable, or in other words, if the guest will promote our brand or not. To achieve this, we have split the data in the ratio of 1:3.



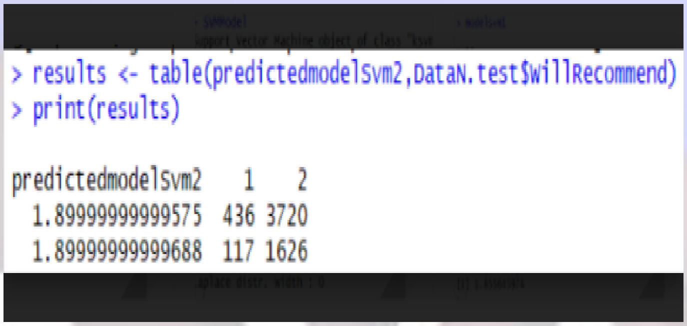
```
> modelsvm
Support Vector Machine object of class "ksvm"

SV type: eps-svr  (regression)
 parameter : epsilon = 0.1  cost C = 5

Gaussian Radial Basis kernel function.
 Hyperparameter : sigma =  0.571428571428571

Number of Support Vectors : 18808

Objective Function Value : -110472.4851
Training error : 3.554084
Cross validation error : 3.506429
Laplace distr. width : 0
> rmse(error)
[1] 1.855588112
```

8.11 KSVM



```
> SVMModel
Support Vector Machine object of class "ksvm"

SV type: eps-svr  (regression)
 parameter : epsilon = 0.1  cost C = 5

Gaussian Radial Basis kernel function.
 Hyperparameter : sigma =  0.5625

Number of Support Vectors : 4143

Objective Function Value : -8280
Training error : 0.090721
Cross validation error : 0.090717
Laplace distr. width : 0
```

8.12 Predicted model based on ksvm

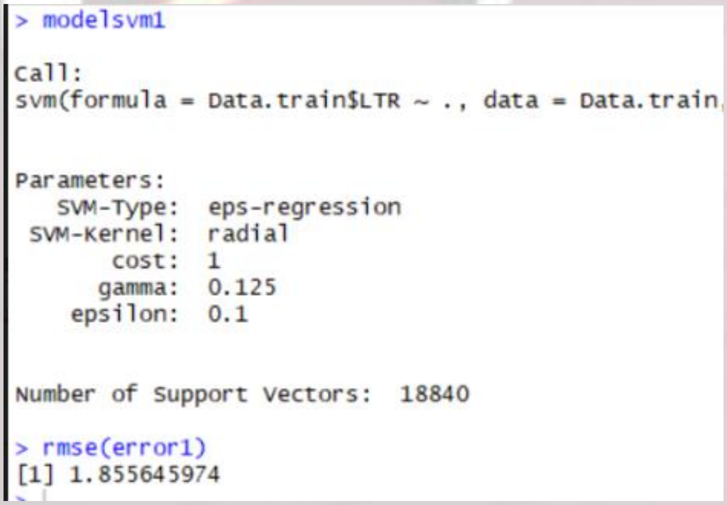


```
> results <- table(predictedmodelSvm2,DataN.test$willRecommend)
> print(results)

predictedmodelSvm2    1    2
  1.89999999999575  436 3720
  1.89999999999688  117 1626
```

8.13Predicted model’s confusion matrix

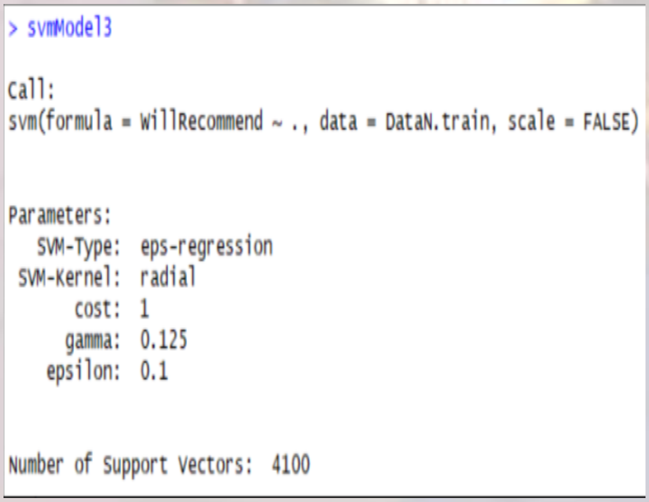
- ## SVM:

  We used Support Vector Machines (SVM model) to create a classifier. This classifier will help us to see if a give a high value to the likelihood to recommend variable, or in other words, if the guest will promote our brand or not. To achieve this, we have split the data in the ratio of 1:3.



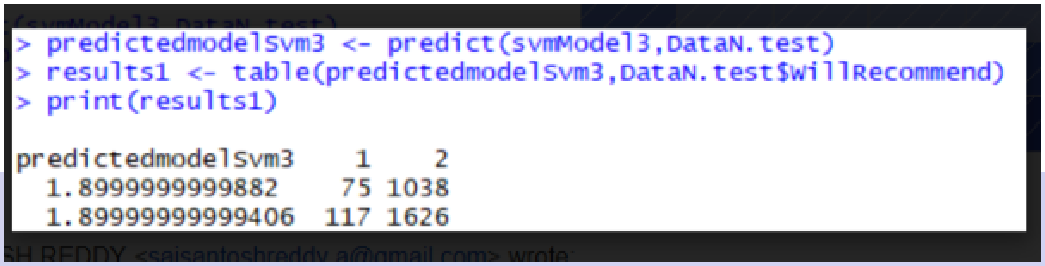
```
> modelsvm1

Call:
svm(formula = Data.train$LTR ~ ., data = Data.train,


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.125
    epsilon:  0.1


Number of Support Vectors:  18840

> rmse(error1)
[1] 1.855645974
```

8.14 SVM



```
> svmModel3

Call:
svm(formula = willRecommend ~ ., data = DataN.train, scale = FALSE)


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.125
    epsilon:  0.1


Number of Support Vectors:  4100
```

8.15 Predicted model based on SVM



```
> predictedmodelSvm3 <- predict(svmModel3,DataN.test)
> results1 <- table(predictedmodelSvm3,DataN.test$WillRecommend)
> print(results1)

predictedmodelSvm3    1    2
  1.8999999999882    75 1038
  1.89999999999406  117 1626
```

8.16 Predicted model's confusion matrix

OBSERVATIONS:

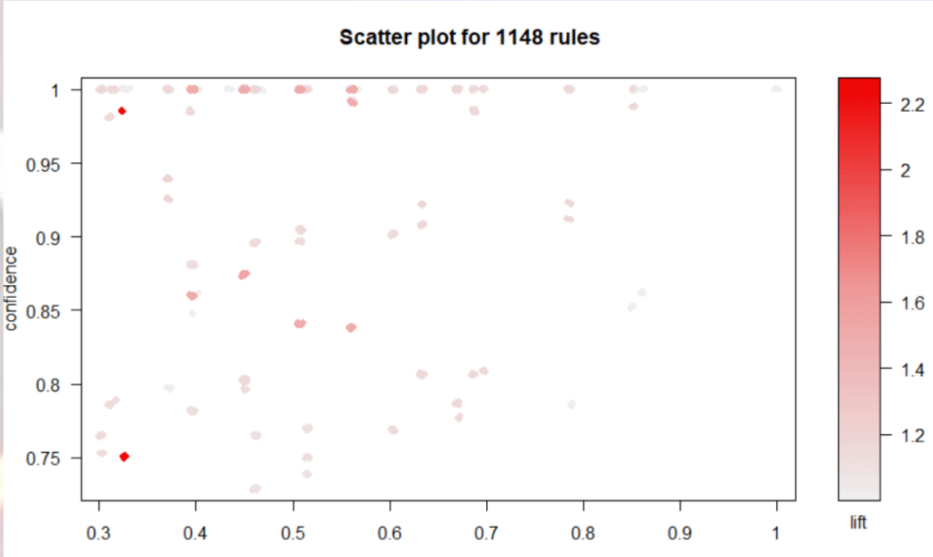
- From the fig 8.11, we found that the KSVM model designed is an underfitting model with a training error of 3.55.
- This being an underfitting model, it is beneficial for this kind of large dataset.
- Thus, the predicted model designed over this model will have less training error and will produce better results.
- We have designed a new variable "WillRecommend" which contains values 1 for all LTR values greater/equal to 7 and 0 for values lesser than 7
- The predicted model has a training error of 0.09 with accuracy of 55.4% calculated by the respective confusion matrix
- The predicted model designed over SVM model has 0.08 training error with 60.8 accuracy calculated by the respective confusion matrix
- The new variable with the help of confusion matrices will predict the future value of LTR for HYATT Hotel in California

```
Data <- Cali[,c(13,26,28,30:34,46)]

str(Data)
row.names(Data) <- NULL
colnames(Data) <- c("LTR","BS","BC","CO","CV","DC","EL","FC","SP")

Data$LTR <- as.numeric(Data$LTR)
Data$BS <- as.data.frame(ifelse(Data$BS=="Y",1,0))
Data$BC <- as.data.frame(ifelse(Data$BC=="Y",1,0))
Data$CO <- as.data.frame(ifelse(Data$CO=="Y",1,0))
Data$CV <- as.data.frame(ifelse(Data$CV=="Y",1,0))
Data$DC <- as.data.frame(ifelse(Data$DC=="Y",1,0))
Data$EL <- as.data.frame(ifelse(Data$EL=="Y",1,0))
Data$FC <- as.data.frame(ifelse(Data$FC=="Y",1,0))
Data$SP <- as.data.frame(ifelse(Data$SP=="Y",1,0))

Data$BS <- as.numeric(Data$BS)
Data$BC <- as.numeric(Data$BC)
Data$CO <- as.numeric(Data$CO)
Data$CV <- as.numeric(Data$CV)
Data$DC <- as.numeric(Data$DC)
Data$EL <- as.numeric(Data$EL)
Data$FC <- as.numeric(Data$FC)
Data$SP <- as.numeric(Data$SP)
```

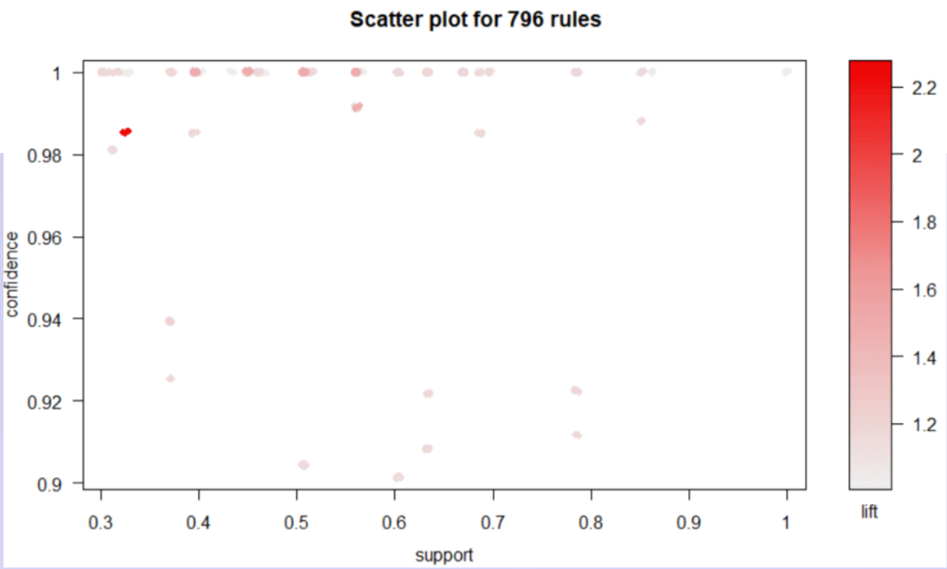
```
SvmData<-
cbind.data.frame(Data$LTR,Data$BS,Data$BC,Data$CO,Data$CV,Data$DC,Data$EL,Data$FC,Data$SP)
random.indexes<-sample(1:nrow(Data))

cutpoint<- floor(nrow(Data)/3*2)
Data.train <- Data[random.indexes[1:cutpoint],]
Data.test<-Data[random.indexes[(cutpoint+1):nrow(Data)],]

library(e1071)
library(kernlab)

modelsvm<-ksvm(Data.train$LTR ~.,data=Data.train, kernel="rbfdot",kpar="automatic",C=5, cross=3,
prob.model=TRUE, scale=FALSE)
modelsvm

predictedmodelSvm <- predict(modelsvm,Data.test)

modelsvm1 <- svm(Data.train$LTR~.,data=Data.train,scale = FALSE)
modelsvm1
predictedmodelSvm1 <- predict(modelsvm1,Data.test)
error1 <- Data.test$LTR - predictedmodelSvm1
rmse(error1)

error <- Data.test$LTR - predictedmodelSvm
rmse <- function(error){
  sqrt(mean(error^2))
}
rmse(error)

Data$WillRecommend <- as.factor(as.numeric(Data$LTR > 6))
Data$WillRecommend <- as.numeric(Data$WillRecommend)
str(Data)

Data$promteLTR <- NULL
DataN <- Data
DataN$LTR <- NULL
DataN$WillRecommend <- as.numeric(as.numeric(DataN$WillRecommend))
str(DataN)

dim(DataN)
random.indexesN <-sample(1:dim(DataN)[1])

cutpointN<- floor(2*dim(DataN)[1]/3)
DataN.train <- DataN[random.indexes[1:cutpointN],]
row.names(DataN.train) <- NULL
DataN.test<-DataN[random.indexes[(cutpoint+1):dim(DataN)[1]],]
row.names(DataN.test) <- NULL

svmModel2 <- ksvm(WillRecommend~.,data=DataN.train,scale=FALSE)
svmModel2
predictedmodelSvm2 <- predict(svmModel2,DataN.test)
predictedmodelSvm2
```

```
results <- table(predictedmodelSvm2,DataN.test$WillRecommend)
print(results)

svmModel3 <- svm(WillRecommend~.,data=DataN.train,scale=FALSE)
svmModel3
predictedmodelSvm3 <- predict(svmModel3,DataN.test)
predictedmodelSvm3
results1 <- table(predictedmodelSvm3,DataN.test$WillRecommend)
print(results1)

results <- table(predictedmodelSvm2,DataN.test$WillRecommend)
print(results)
```

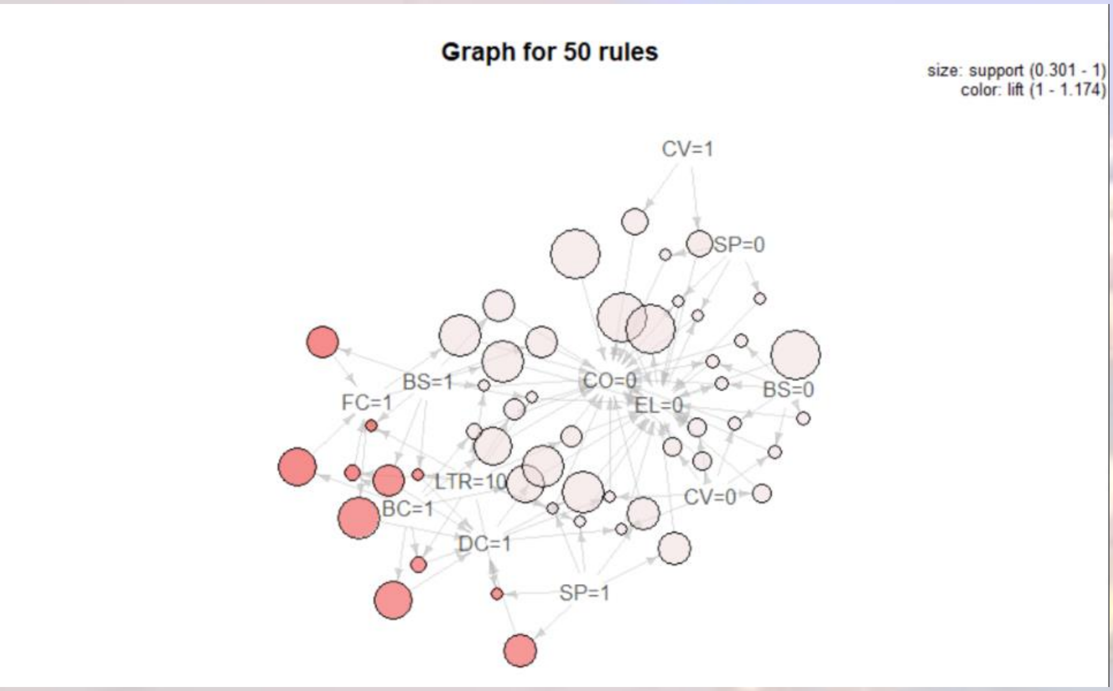
- ## ARULES:



8.16 Apriori scatter plot



8.17 Good rules based on confidence



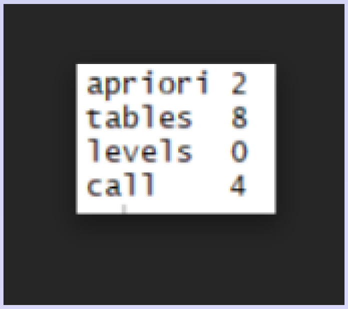
8.18 Graphical representation of arules

OBSERVATIONS:

According to Arules, designed we recommend that customers who visited for business purposes are the ones to influence the NPS score based on their experience with the amenities provided.

- ## NAIVE BAYES:

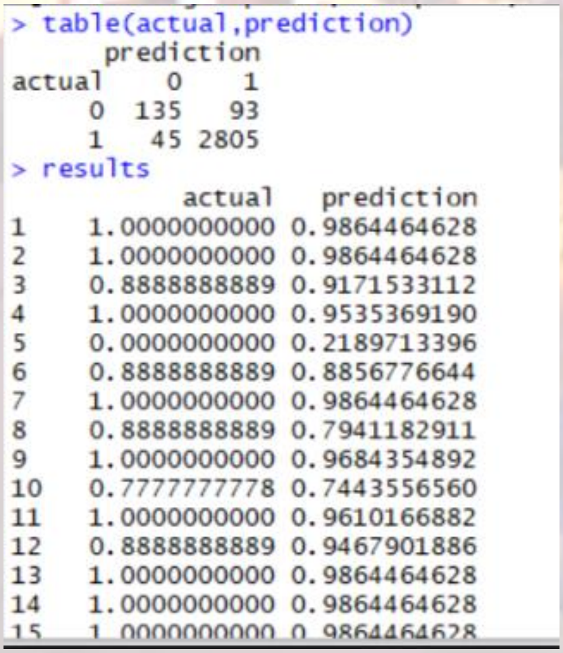
Naive Using naive bayes, we aim to implement a binary classifier based on probability. In this case, will recommend is an independent variable and all others are dependent variables. Using the training and testing datasets created before, we will apply the same models to understand how good it is performing.



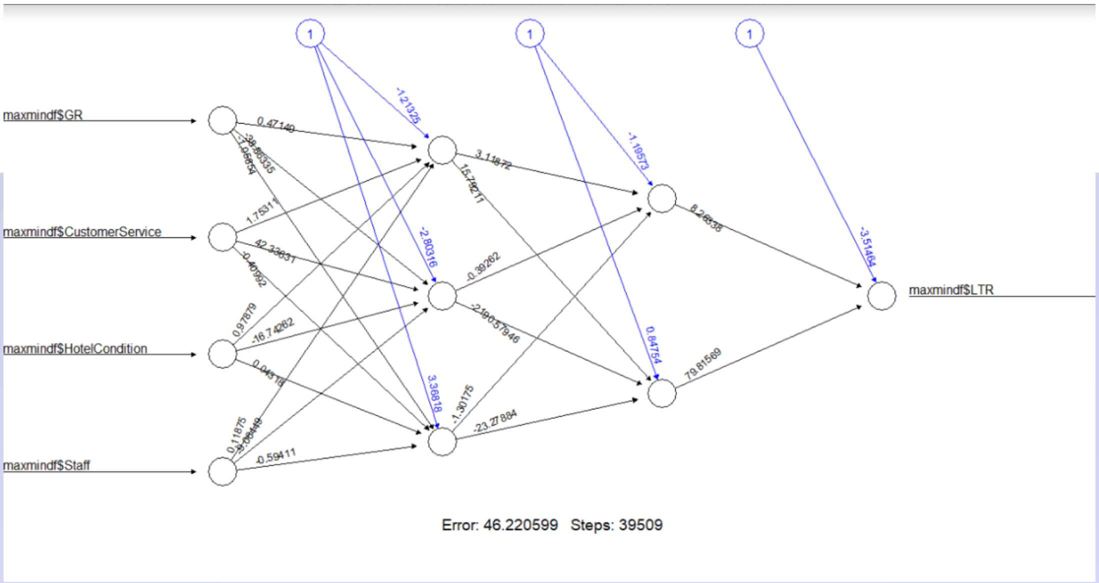
```
apriori 2
tables  8
levels  0
call    4
```

- ## NEURAL NETWORK

Neural networks are typically organized in layers. Layers are made up of a number of interconnected 'nodes' which contain an 'activation function'. Patterns are presented to the network via the 'input layer', which communicates to one or more 'hidden layers' where the actual processing is done via a system of weighted 'connections'



```
> table(actual,prediction)
      prediction
actual    0    1
     0  135   93
     1   45 2805
> results
             actual   prediction
1    1.0000000000 0.9864464628
2    1.0000000000 0.9864464628
3    0.8888888889 0.9171533112
4    1.0000000000 0.9535369190
5    0.0000000000 0.2189713396
6    0.8888888889 0.8856776644
7    1.0000000000 0.9864464628
8    0.8888888889 0.7941182911
9    1.0000000000 0.9684354892
10   0.7777777778 0.7443556560
11   1.0000000000 0.9610166882
12   0.8888888889 0.9467901886
13   1.0000000000 0.9864464628
14   1.0000000000 0.9864464628
15   1.0000000000 0.9864464628
```



OBSERVATIONS:

From the Neural network model, which has a training error of 46.22% with 39509 steps, the final output is LTR with guest room, customer service, hotel condition and staff being the inputs with (3,2) hidden layers. This model is 95.45% accurate in predicting future LTR values.

• RECOMMENDATIONS & CONCLUSIONS:

Likelihood to recommend being the word of mouth from people who have visited HYATT hotel is always influenced by the amenities and facilities provided by the hotel

- Our analysis shows that the condition of a hotel, customer service provided and guest room infrastructure influences the people recommending hotel in California.
- With the predicted models created using SVM, KSVM, Neural Network and Naive Bayes modelling techniques, the customers are currently happy with the provided amenities but with value of 3060 detractors the following amenities must be upgraded and provided in hotels in California.

  These influence the most:

- Business center
- Fitness center
- Bell staff
- Self-parking facility
- Dry Cleaning

- The Hotels who fail to reach their goal NPS score should focus on improving their amenities provided to the customers since our analysis has led us to believe that the amenities provided to the customer play a significant role in the customer's feedback.
- According to Arules, designed we recommend that customers who visited for business purposes are the ones to influence the NPS score based on their experience with the amenities provided.

• REFERENCES:

. Byers, T. (2015) Ggplot 2.0.0. Available at: https://blog.rstudio.org/2015/12/21/ggplot2-2-0-0/ (Accessed: April 27th, 2017).

2. CheckMarket (2011) Net promoter score (NPS) - use, application and pitfalls. Available at: https://www.checkmarket.com/blog/net-promoter-score/ (Accessed: April 27th, 2017).

3. Geom_boxplot. Ggplot2 2.1.0 (no date) Available at: http://docs.ggplot2.org/current/geom_boxplot.html (Accessed: April 27th, 2017).

4. Godwin, H. (2011) Merge all files in a directory using R into a single dataframe. Available at: https://www.r-bloggers.com/merge-all-files-in-a-directory-using-r-into-a-single-dataframe/ (Accessed: April 27th, 2017).

5. In R, how can I compute percentage statistics on a column in a dataframe (table function extended with percentages) (2016) Available at: http://stackoverflow.com/questions/9623763/in-r-how-can-i-compute-percentage-statistics-on-a-column-in-a-dataframe-tabl (Accessed: April 27th, 2017).

6. Legends (ggplot2) (no date) Available at: http://www.cookbook-r.com/Graphs/Legends_(ggplot2)/ (Accessed: April 27th, 2017).

7. Robk, R.K. - (2014) Quick-r: Pie charts. Available at: http://www.statmethods.net/graphs/pie.html (Accessed: April 27th, 2017).

8. Systems, S. (2016) What is net promoter? Available at: https://www.netpromoter.com/know/ (Accessed: April 27th, 2017).

HYATT

HYATT

HYATT

HYATT