**Final Report – Hyatt Hotel Data Analysis**



**Team Members:**

Anmol Handa

Khushabh Thakar

Shreyas Sambamurthy

Tanay Pardeshi

* **Problem statement:**

Hyatt has chains all around the world. Customers from varying demographics and age visit different brands of Hyatt for different purposes. Hyatt aims at increasing the number of Promoters. The Net Promoter Score is an index ranging from -100 to 100 that measures the willingness of customers to recommend a company's products or services to others. It is used as a proxy for gauging the customer's overall satisfaction with a company's product or service and the customer's loyalty to the brand. Hyatt is looking for suggestions to improve their customer experience.

* **Data Extraction:**

Initially we started off by cleaning the data given to us. We used the data from the months of March, June, September and December for our analysis. We narrowed down our analysis of the data set to United States since the majority of the data comprised of reviews from USA. We took a sample of 5000 records from all the four months and further used the analysis as given below. There was a total of 65 records taken into consideration.

**R code:**

marchData <- read.csv("sampleMarchData.csv")

juneData <- read.csv("sampleJuneData.csv")

septemberData <- read.csv("sampleSeptemberData.csv")

decemberData <- read.csv("sampleDecemberData.csv")

fieldNames <- read.csv("majorAttributes.csv")

marchData <- marchData[,which(colnames(marchData) %in% colnames(fieldNames))]

juneData <- juneData[,which(colnames(juneData) %in% colnames(fieldNames))]

septemberData <- septemberData[,which(colnames(septemberData) %in% colnames(fieldNames))]

decemberData <- decemberData[,which(colnames(decemberData) %in% colnames(fieldNames))]

View(marchData)

View(juneData)

View(septemberData)

View(decemberData)

#write.csv(marchData,file= 'filteredDataMarch.csv')

#write.csv(juneData,file='filteredData2.csv')

#write.csv(septemberData,file='filteredData3.csv')

#write.csv(decemberData,file='filteredData4.csv')

#Comparison to shortlist Countries

#Find Highest visited Country

install.packages("ggplot2")

library(ggplot2)

#install.packages("reshape2")

#library(reshape2)

#For March, the top countries are:

marchdataTable <- table(marchData$Country\_PL, marchData$State\_PL, marchData$City\_PL)

marchTableVector <- as.data.frame(marchdataTable)

class(marchTableVector$Var1)

marchTableVectorData <- c(marchTableVector$Var1)

headMarchData <- head(marchTableVector[order(-marchTableVector$Freq),],4)

headMarchData$month<-  rep("March",nrow(headMarchData)) # make new column

headMarchData

dummyDF <- data.frame(state.name, stringsAsFactors=FALSE)

dummyDF$state <- tolower(dummyDF$state.name)

#United States

#States- Flordia, New York,Texas, California, DC, California

#City- Orlando, New York, San Antonio, San Diego

#For June, the top countries are:

juneDataTable <- table(juneData$Country\_PL, juneData$State\_PL, juneData$City\_PL)

juneTableVector <- as.data.frame(juneDataTable)

headJuneData <- head(juneTableVector[order(-juneTableVector$Freq),],4)

headJuneData$month<-  rep("June",nrow(headJuneData)) # make new column

headJuneData

#United States

#States- New York, Florida, Texas, California, California

#City- New York, Orlando, San Antonio,San Diego

#For Sep, the top countries are:

sepDataTable <- table(septemberData$Country\_PL,septemberData$State\_PL, septemberData$City\_PL)

sepTableVector <- as.data.frame(sepDataTable)

headSepData <- head(sepTableVector[order(-sepTableVector$Freq),],4)

headSepData$month<-  rep("Sep",nrow(headSepData)) # make new column

headSepData

#United States

#States- New York, Florida, Texas, Illinois

#City- New York, Orlando, San Antonio, Chicago

#For Dec, the top countries are:

decDataTable <- table(decemberData$Country\_PL,decemberData$State\_PL, decemberData$City\_PL)

decTableVector <- as.data.frame(decDataTable)

headDecData<- head(decTableVector[order(-decTableVector$Freq),],4)

headDecData$month<-  rep("Dec",nrow(headDecData)) # make new column

headDecData

#United States

#State- Illinois, Flordia, New York, California, Texas

#City- Chicago, Orlando, New York

install.packages("reshape2")

library(reshape2)

combinedResults <- rbind(headMarchData, headJuneData, headSepData, headDecData)

colnames(combinedResults)[2]<- 'States'

combinedResults

ggplot(combinedResults, aes(reorder(factor(month), -Freq), Freq, fill = States)) +

  geom\_bar(stat="identity", position = "dodge") +

  scale\_fill\_brewer(palette = "Set1")+ylim(0, 170) + coord\_cartesian(ylim=c(90,170))+labs(x="Months") +labs(y="No. of Customers")

The above code combines the results from the data set of all four months.

Graph displays statistics of different states in the USA

A picture containing screenshot

Description generated with very high confidence

* **Major business questions answered:**

1. Why was analysis confined to guests staying at Hyatt group of hotels in the state of Texas?

From our observations as shown below we got to know that Texas has a staggering NPS score and Average Likelihood to recommend. Hence we focused our major analysis on the state of Texas.

marchTexasData <- read.csv("filteredData1.csv")

juneTexasData <- read.csv("filteredData2.csv")

septemberTexasData <- read.csv("filteredData3.csv")

decemberTexasData <- read.csv("filteredData4.csv")

combinedTexasData <- rbind(marchTexasData, juneTexasData, septemberTexasData, decemberTexasData)

write.csv(combinedTexasData,file= 'combinedTexasData.csv')

combinedTexasData

#Lets look at highest hotel entries along with net promoter score for the hotels

#install.packages("sqldf")

library(sqldf)

#install.packages("ggplot")

library(ggplot2)

#install.packages("gridExtra")

library(gridExtra)

avgHotelNetPromoterScore <- sqldf("select AVG(Likelihood\_Recommend\_H), Count(X), Brand\_PL from combinedTexasData GROUP BY Brand\_PL ")

avgHotelNetPromoterScore

#To check NPS Score for Texas over 4 months

npsMarch <- sqldf("select Count(NPS\_Type) from  marchTexasData Where NPS\_TYPE='Promoter'and City\_PL='San Antonio'")

npsMarchOverall <-sqldf("select Count(NPS\_Type) from  marchTexasData Where City\_PL='San Antonio'")

npsMarch <- npsMarch/npsMarchOverall\*100

npsMarch

npsJune <- sqldf("select Count(NPS\_Type) from  juneTexasData Where NPS\_TYPE='Promoter' and City\_PL='San Antonio' ")

npsJuneOverall <-sqldf("select Count(NPS\_Type) from  juneTexasData Where City\_PL='San Antonio'")

npsJune <- npsJune/npsJuneOverall\*100

npsJune

npsSep <- sqldf("select Count(NPS\_Type) from  septemberTexasData Where NPS\_TYPE='Promoter' and City\_PL='San Antonio' ")

npsDecOverall <-sqldf("select Count(NPS\_Type) from  septemberTexasData Where City\_PL='San Antonio'")

npsSep <- npsSep/npsDecOverall\*100

npsSep

npsDec <- sqldf("select Count(NPS\_Type) from  decemberTexasData Where NPS\_TYPE='Promoter' And City\_PL='San Antonio' ")

npsDecOverall <-sqldf("select Count(NPS\_Type) from  decemberTexasData Where City\_PL='San Antonio'")

npsDec <- npsDec/npsDecOverall\*100

npsDec

#Now same calculation of AVG likelihood to recommend for march month only

avgHotelNetPromoterScore3 <- sqldf("select AVG(Likelihood\_Recommend\_H), Count(X), Brand\_PL from  marchTexasData GROUP BY Brand\_PL ")

avgHotelNetPromoterScore3

#Now same calculation of AVG likelihood to recommend for june month only

avgHotelNetPromoterScore4 <- sqldf("select AVG(Likelihood\_Recommend\_H), Count(X), Brand\_PL from  juneTexasData GROUP BY Brand\_PL ")

avgHotelNetPromoterScore4

#Now same calculation of AVG likelihood to recommend for sep month only

avgHotelNetPromoterScore5 <- sqldf("select AVG(Likelihood\_Recommend\_H), Count(X), Brand\_PL from  septemberTexasData GROUP BY Brand\_PL ")

avgHotelNetPromoterScore5

#Now same calculation of AVG likelihood to recommend for december month only

avgHotelNetPromoterScore2 <- sqldf("select AVG(Likelihood\_Recommend\_H), Count(X), Brand\_PL from decemberTexasData GROUP BY Brand\_PL ")

avgHotelNetPromoterScore2

A picture containing LEGO

Description generated with high confidence

1. Why Hyatt Regency needs special focus amongst other Hyatt brands?

As visualized below, Hyatt Regency accumulates major amount of customer base along with the decreasing value of Likelihood to recommend across months. Hence, we decided to focus on this erratic behavior and consider Hyatt Regency for further analysis.

#Grouping the entire data set based on Brand\_PL

combined\_data <- sqldf("select AVG(Likelihood\_Recommend\_H), Count(X), Brand\_PL from combinedTexasData GROUP BY Brand\_PL ")

combined\_data

View(combined\_data)

#Grouping the data set for march based on Brand\_PL

march\_data <- sqldf("select AVG(Likelihood\_Recommend\_H),Count(X), Brand\_PL from  marchTexasData GROUP BY Brand\_PL ")

march\_data <- data.frame(march\_data)

colnames(march\_data) <- c("AvgLikelihoodRecommend","CountMarch","HotelChain")

View(march\_data)

#Grouping the data set for june based on Brand\_PL

june\_data <- sqldf("select AVG(Likelihood\_Recommend\_H),Count(X), Brand\_PL from  juneTexasData GROUP BY Brand\_PL ")

june\_data

june\_data <- data.frame(june\_data)

colnames(june\_data) <- c("AvgLikelihoodRecommend","CountJune","HotelChain")

View(june\_data)

#Grouping the data set for september based on Brand\_PL

sep\_data <- sqldf("select AVG(Likelihood\_Recommend\_H),Count(X), Brand\_PL from  septemberTexasData GROUP BY Brand\_PL ")

sep\_data

sep\_data <- data.frame(sep\_data)

colnames(sep\_data) <- c("AvgLikelihoodRecommend","CountSep","HotelChain")

View(sep\_data)

#Grouping the data set for december based on Brand\_PL

dec\_data <- sqldf("select AVG(Likelihood\_Recommend\_H),Count(X), Brand\_PL from decemberTexasData GROUP BY Brand\_PL ")

dec\_data

dec\_data <- data.frame(dec\_data)

colnames(dec\_data) <- c("AvgLikelihoodRecommend","CountDec","HotelChain")

View(dec\_data)

#Data frame to store month values

Months <- c("March","March","March","March","March","June","June","June","June","June", "September","September","September","September", "September","December","December","December", "December","December")

Months < data.frame(Months)

View(Months)

#Data frame to store the count values

CountValues <- c(march\_data[2],june\_data[2],sep\_data[2],dec\_data[2])

CountValues <- stack(CountValues)

View(CountValues)

CountValues <- CountValues[-2]

colnames(CountValues) <- "Count"

View(CountValues)

#Data frame to store the Brand\_PL values

hotelchain <- c("Grand Hyatt","Hyatt","Hyatt House","Hyatt Place","Hyatt Regency","Grand Hyatt","Hyatt","Hyatt House","Hyatt Place","Hyatt Regency","Grand Hyatt","Hyatt","Hyatt House","Hyatt Place","Hyatt Regency","Grand Hyatt","Hyatt","Hyatt House","Hyatt Place","Hyatt Regency")

hotelchain <- data.frame(hotelchain)

View(hotelchain)

#Combined data frame

Newdf <- data.frame(hotelchain,Months,CountValues)

View(Newdf)

Months <- factor(Months, levels = c('March', 'June', 'September','December'))

#Comparing count data for all 4 months using bar chart:

g\_bar <- ggplot(Newdf,aes(x=hotelchain,y=CountValues,fill=Months))+ geom\_bar(stat="identity",position="dodge")+ scale\_fill\_manual(name="Months",values=c("grey", "yellow", "brown", "orange")) + xlab("Hotel Chain")+ylab("Number of guests")+ coord\_cartesian(ylim=c(300,3000)) + ggtitle("Number of guests based on Hotel chain")

g\_bar

#Generating a Bar plot for comparing the Avg of likelihood to recommend field of different hotel chains in the months of March, June, September and December-->

#Data frame to store the average likelihood to recommend values

AvgLikelihood <- c(march\_data[1],june\_data[1],sep\_data[1],dec\_data[1])

AvgLikelihood <- stack(AvgLikelihood)

View(AvgLikelihood)

AvgLikelihood <- AvgLikelihood[-2]

colnames(AvgLikelihood) <- "AvgLikelihood"

View(AvgLikelihood)

#New combined data frame

Newdf2 <- data.frame(hotelchain,Months,AvgLikelihood)

View(Newdf2)

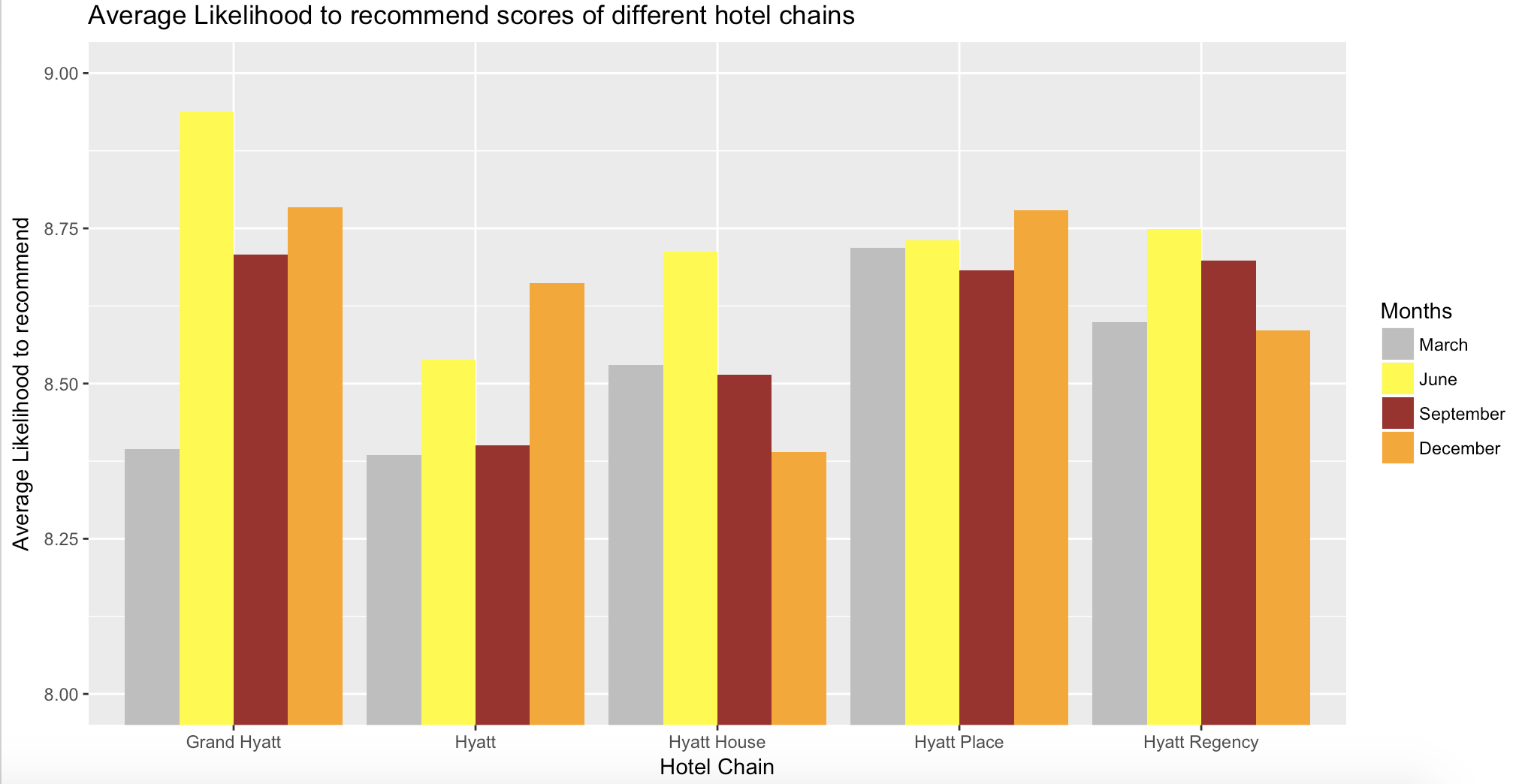
#Comparing Avg likelihood to recommend data for all 4 months using bar chart:

g\_avgLTR <- ggplot(Newdf2,aes(x=hotelchain,y=AvgLikelihood,fill=factor(Months)))+ geom\_bar(stat="identity",position="dodge")+ scale\_fill\_manual(name="Months",values=c("grey", "yellow", "brown", "orange"))  + xlab("Hotel Chain")+ylab("Average Likelihood to recommend")+ coord\_cartesian(ylim=c(8,9)) + ggtitle("Average Likelihood to recommend scores of different hotel chains")

g\_avgLTR

A picture containing screenshot

Description generated with high confidence



**Data refining for Hyatt Regency:**

We filtered our attributes based on the summary functions to check if the values of attributes weren’t changing throughout the data. We replaced the NA values present in the columns having ratings on the scale of 1-10 with the mean values in order to maintain data consistency. Also, there were some empty fields with no values which were replaced by NA. For club lounge and spa use services, we replaced the NA values with “I don’t know” which was one of the already existing factors. Later, a subset of the relevant columns was only taken and also the remaining NA values among the flag attributes were omitted. After cleaning the data, we filtered down to 4600 records with 37 variables.

#HyattRegency

hyattRegencyDataSet <- sqldf("select \* from combinedTexasData WHERE Brand\_PL ='Hyatt Regency' ")

hyattRegencyDataSet

#replacing null values with NA

hyattRegencyDataSet[hyattRegencyDataSet==''] <- NA

hyattRegencyDataSet$Clublounge\_Used\_H <- as.character(hyattRegencyDataSet$Clublounge\_Used\_H, stringsAsFactors=FALSE)

hyattRegencyDataSet$Spa\_Used\_H <- as.character(hyattRegencyDataSet$Spa\_Used\_H, stringsAsFactors=FALSE)

hyattRegencyDataSet$MEMBER\_STATUS\_R <- as.character(hyattRegencyDataSet$MEMBER\_STATUS\_R, stringsAsFactors=FALSE)

hyattRegencyDataSet[["Clublounge\_Used\_H"]][is.na(hyattRegencyDataSet[["Clublounge\_Used\_H"]])] <-"I don't know"

hyattRegencyDataSet[["Spa\_Used\_H"]][is.na(hyattRegencyDataSet[["Spa\_Used\_H"]])] <-"I don't know"

hyattRegencyDataSet[["MEMBER\_STATUS\_R"]][is.na(hyattRegencyDataSet[["MEMBER\_STATUS\_R"]])] <-"None"

#Replacing na Values in F.b\_overall\_h, Check\_in\_h, internet service and few other with mean values

for(col in 1:ncol(hyattRegencyDataSet)){

  hyattRegencyDataSet[is.na(hyattRegencyDataSet[,col]), col] <- round(mean(na.omit(hyattRegencyDataSet[,col])))

}

#Checking summary of fields to figure out which fields are not changing

#

summary(hyattRegencyDataSet$Mini.Bar\_PL)

#

summary(hyattRegencyDataSet$All.Suites\_PL)

summary(hyattRegencyDataSet$Bell.Staff\_PL)

#

summary(hyattRegencyDataSet$Boutique\_PL)

summary(hyattRegencyDataSet$Business.Center\_PL)

#

summary(hyattRegencyDataSet$Casino\_PL)

#

summary(hyattRegencyDataSet$Conference\_PL)

summary(hyattRegencyDataSet$Convention\_PL)

summary(hyattRegencyDataSet$Dry.Cleaning\_PL)

#

summary(hyattRegencyDataSet$Elevators\_PL)

#

summary(hyattRegencyDataSet$Fitness.Center\_PL)

#

summary(hyattRegencyDataSet$Fitness.Trainer\_PL)

summary(hyattRegencyDataSet$Golf\_PL)

#

summary(hyattRegencyDataSet$Indoor.Corridors\_PL)

summary(hyattRegencyDataSet$Laundry\_PL)

summary(hyattRegencyDataSet$Limo.Service\_PL)

#

summary(hyattRegencyDataSet$Pool.Indoor\_PL)

#

summary(hyattRegencyDataSet$Pool.Outdoor\_PL)

summary(hyattRegencyDataSet$Regency.Grand.Club\_PL)

summary(hyattRegencyDataSet$Resort\_PL)

#

summary(hyattRegencyDataSet$Restaurant\_PL)

summary(hyattRegencyDataSet$Self.Parking\_PL)

summary(hyattRegencyDataSet$Shuttle.Service\_PL)

#

summary(hyattRegencyDataSet$Ski\_PL)

summary(hyattRegencyDataSet$Spa\_PL)

summary(hyattRegencyDataSet$Valet.Parking\_PL)

summary(hyattRegencyDataSet$Spa\_Used\_H)

summary(hyattRegencyDataSet$Clublounge\_Used\_H)

#Dropping columns by names since these fields are not contributing here

hyattRegencyDataSet = subset(hyattRegencyDataSet, select = -c(Spa.services.in.fitness.center\_PL,Spa.online.booking\_PL,Spa.F.B.offering\_PL) )

hyattRegencyDataSetFiltered = subset(hyattRegencyDataSet, select = -c(Mini.Bar\_PL,All.Suites\_PL,Bell.Staff\_PL, Boutique\_PL,Casino\_PL,Conference\_PL,Elevators\_PL,Fitness.Center\_PL,

                                                              Fitness.Trainer\_PL,Indoor.Corridors\_PL,Pool.Indoor\_PL,Pool.Outdoor\_PL,Restaurant\_PL,Ski\_PL,ROOM\_NUM\_C,Club.Type\_PL,MAJOR\_MARKET\_CODE\_C,ROOM\_TYPE\_CODE\_C,

                                                              PMS\_ROOM\_REV\_USD\_C,RESERVATION\_STATUS\_R,MAJOR\_MARKET\_CODE\_R,PAST\_VS\_FUTURE\_R,State\_PL,Country\_PL,Guest.NPS.Goal\_PL,Brand\_PL,Hotel.Inventory\_PL,Floors\_PL,Union\_PL,Relationship\_PL) )

#Cleaning Further by just keeping genders as male or females

hyattRegencyDataSetFiltered <- subset(hyattRegencyDataSetFiltered,Gender\_H %in% c("Male","Female")) # only keep male and female gender

hyattRegencyDataSetFiltered$F.B\_Overall\_Experience\_H <- round(hyattRegencyDataSetFiltered$F.B\_Overall\_Experience\_H )

#Omitting few records where adult nos are not mentioned

hyattRegencyDataSetFiltered <-  hyattRegencyDataSetFiltered[!is.na(hyattRegencyDataSetFiltered$ADULT\_NUM\_C),]

hyattRegencyDataSetFiltered<- na.omit(hyattRegencyDataSetFiltered)

hyattRegencyDataSetFiltered <- hyattRegencyDataSetFiltered[hyattRegencyDataSetFiltered$X!=605,]

Changing data types of the attributes:

#Giving data types to each field properly

hyattRegencyDataSetFiltered$LENGTH\_OF\_STAY\_C <- as.numeric(hyattRegencyDataSetFiltered$LENGTH\_OF\_STAY\_C)

hyattRegencyDataSetFiltered$Gender\_H <- as.factor(hyattRegencyDataSetFiltered$Gender\_H)

hyattRegencyDataSetFiltered$Age\_Range\_H <- as.factor(hyattRegencyDataSetFiltered$Age\_Range\_H)

hyattRegencyDataSetFiltered$POV\_CODE\_C <- as.factor(hyattRegencyDataSetFiltered$POV\_CODE\_C)

hyattRegencyDataSetFiltered$ADULT\_NUM\_C <- as.numeric(hyattRegencyDataSetFiltered$ADULT\_NUM\_C)

hyattRegencyDataSetFiltered$REVENUE\_USD\_R <- as.numeric(hyattRegencyDataSetFiltered$REVENUE\_USD\_R)

hyattRegencyDataSetFiltered$Spa\_PL <- as.factor(hyattRegencyDataSetFiltered$Spa\_PL)

hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H <- as.numeric(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H)

hyattRegencyDataSetFiltered$Guest\_Room\_H <- as.numeric(hyattRegencyDataSetFiltered$Guest\_Room\_H)

hyattRegencyDataSetFiltered$Tranquility\_H <- as.numeric(hyattRegencyDataSetFiltered$Tranquility\_H)

hyattRegencyDataSetFiltered$Condition\_Hotel\_H <- as.numeric(hyattRegencyDataSetFiltered$Condition\_Hotel\_H)

hyattRegencyDataSetFiltered$Customer\_SVC\_H <- as.numeric(hyattRegencyDataSetFiltered$Customer\_SVC\_H)

hyattRegencyDataSetFiltered$Staff\_Cared\_H <- as.numeric(hyattRegencyDataSetFiltered$Staff\_Cared\_H)

hyattRegencyDataSetFiltered$Internet\_Sat\_H <- as.numeric(hyattRegencyDataSetFiltered$Internet\_Sat\_H)

hyattRegencyDataSetFiltered$Check\_In\_H <- as.numeric(hyattRegencyDataSetFiltered$Check\_In\_H)

hyattRegencyDataSetFiltered$Clublounge\_Used\_H <- as.factor(hyattRegencyDataSetFiltered$Clublounge\_Used\_H)

hyattRegencyDataSetFiltered$Spa\_Used\_H <- as.factor(hyattRegencyDataSetFiltered$Spa\_Used\_H)

hyattRegencyDataSetFiltered$MEMBER\_STATUS\_R <- as.factor(hyattRegencyDataSetFiltered$MEMBER\_STATUS\_R)

Converting NPS type to an integral value with three levels:

#Adding a new field to check whether people recommend

hyattRegencyDataSetFiltered$willRecommend <- as.character(as.numeric(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H > 7))

hyattRegencyDataSetFiltered <- within(hyattRegencyDataSetFiltered, willRecommend[Likelihood\_Recommend\_H >8 & Likelihood\_Recommend\_H <=10] <- 1)

hyattRegencyDataSetFiltered <- within(hyattRegencyDataSetFiltered, willRecommend[Likelihood\_Recommend\_H >=7 & Likelihood\_Recommend\_H <= 8] <-  0)

hyattRegencyDataSetFiltered <- within(hyattRegencyDataSetFiltered, willRecommend[Likelihood\_Recommend\_H >=0 & Likelihood\_Recommend\_H < 7] <- -1)

hyattRegencyDataSetFiltered$willRecommend <- as.factor(hyattRegencyDataSetFiltered$willRecommend)

1. Which class and category of guests constitute the major source of revenue for Hyatt group of hotels within the United States?

As shown below, the analysis around the customer type (Business and Leisure) represented that the major customers as well as the revenue generated was from business class. This helped us to confine our interest around business class customers.

#Calculating Avg revenue each customer giving us from Business and Leisure class

avgRevenue<- aggregate(hyattRegencyDataSetFiltered$REVENUE\_USD\_R, by= list(hyattRegencyDataSetFiltered$POV\_CODE\_C), mean)

avgRevenue

#Calculating Overall revenue from different groups of customers

netRevenue<- tapply(hyattRegencyDataSetFiltered$REVENUE\_USD\_R, hyattRegencyDataSetFiltered$POV\_CODE\_C, sum)

netRevenue

#Calculating Overall Customers from Leisure and Business

overallCustomers <- tapply(hyattRegencyDataSetFiltered$REVENUE\_USD\_R, hyattRegencyDataSetFiltered$POV\_CODE\_C, length)

overallCustomers

#Calculating Avg Length of stay for Business and Leisure Class

avgLengthStay<- tapply(hyattRegencyDataSetFiltered$LENGTH\_OF\_STAY\_C, hyattRegencyDataSetFiltered$POV\_CODE\_C, mean)

avgLengthStay

#Calculating Avg likelihood to Recommend for business and Leisure Class

avgLikeLihood<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$POV\_CODE\_C, mean)

avgLikeLihood

medianLikeLihood<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$POV\_CODE\_C, median)

medianLikeLihood

comparisonRowNames <- c("Leisure", "Business")

#For Avg Revenue

pieSlices <- c(avgRevenue[2,2],avgRevenue[1,2])

pct <- round(pieSlices/sum(pieSlices)\*100)

label <- paste(comparisonRowNames, pct) # add percents to labels

lbls <- paste(label,"%",sep="") # ad % to labels

pieChart1 <- pie(pieSlices,labels = lbls, col=rainbow(length(lbls)),main="Average Revenue for different Type of Guests in $")

#For total Revenue

pieSlices2 <- c(netRevenue[2],netRevenue[1])

pct2 <- round(pieSlices2/sum(pieSlices2)\*100)

label2 <- paste(comparisonRowNames, pct2) # add percents to labels

lbls <- paste(label2,"%",sep="")

pieChart2 <- pie(pieSlices2,labels = lbls, col=rainbow(length(lbls)),main="Net Revenue for different Type of Guests in $")

#For overall Customers

pieSlices3 <- c(overallCustomers[2],overallCustomers[1])

pct3 <- round(pieSlices3/sum(pieSlices3)\*100)

label3 <- paste(comparisonRowNames, pct3) # add percents to labels

lbls <- paste(label3,"%",sep="")

pieChart3 <- pie(pieSlices3,labels = lbls, col=rainbow(length(lbls)),main="Frequency of Guests")

#Conclusion - Avg $ spent by Business visitor is less than the Leisure Vistor but netrevenue from

#Business Vistor is 5 times the revenue from Leisure with proven fact both are staying for almost 2 days on an average

#Also, no of Business Vistors are 5 times the Leisure Visitors

#Main source of income for hyatt regency in Texas is Business Class

A close up of graphics

Description generated with high confidenceA close up of a logo

Description generated with high confidence

* **Creating training and test data sets:**

#Splitting data into training and test datasets

#2/3 training dataset and 1/3 test dataset

datasetFormation <- function(df){

  randIndex <- sample(1:nrow(df))

  cutPoint <- round(2 \* nrow(df) / 3)

  trainingData <- df[randIndex[1:cutPoint],]

  testData <- df[randIndex[(cutPoint + 1): nrow(df)],]

  return(list(trainingData, testData))

}

#Calling datasetFormation for traindataSet and testDataSet

#All the importatn attributes taken here except X, and NPS\_Type and Likelihood\_Recommend\_H

#tempDataFrame <- datasetFormation(hyattRegencyDataSetFiltered[,c(2:13,15:36,38)])

tempDataFrame <- datasetFormation(hyattRegencyDataSetFiltered[,c(5,8,10:13,15:22,38)])

#all the age, sex, date of checkin, adults no, class,guest country, membership, gender, length of stay

#tempDataFrame2 <- datasetFormation(hyattRegencyDataSetFiltered[,c(2:8,10:11,38)])

trainingData <- as.data.frame(tempDataFrame[1])

testingData <- as.data.frame(tempDataFrame[2])

rm(tempDataFrame)

* **KSVM model:**

  ############ KSVM Model

  ksvmModel <- ksvm(willRecommend~ ., data = trainingData, kernel = "rbfdot",

                    kpar = "automatic", C = 5, cross = 3, prob.model = TRUE)

  ksvmModel

  ksvmPrediction <- predict(ksvmModel, testingData, type = "response")

  ksvmPrediction

  newDf1 = data.frame(v1=testingData$willRecommend, v2=ksvmPrediction)

newDf1

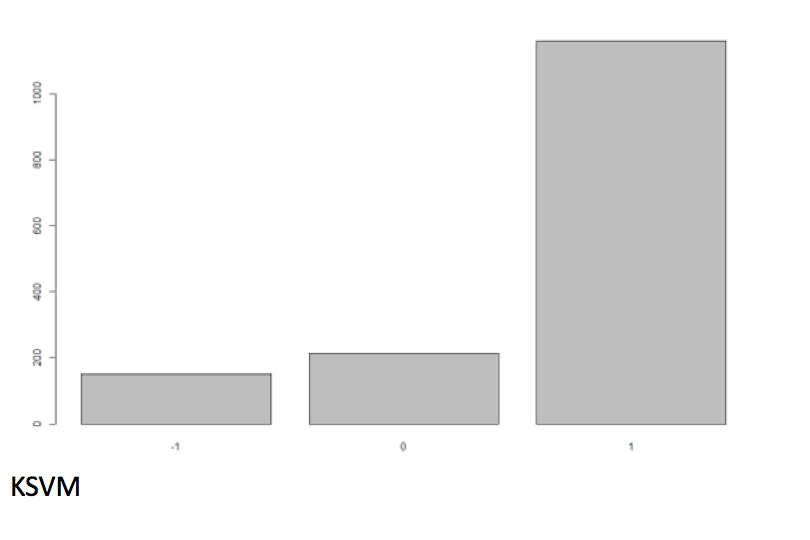
  ksvmError <- sum(testingData[,aim] != ksvmPrediction) \* 100 /length(ksvmPrediction)

  ksvmError

  #Error is only 20.24902%

#Cross Validation Error is 0.198824

#Training Error is 0.0985



* **SVM model:**

############ SVM Model

  svmModel <- svm(willRecommend ~ ., data = trainingData, kernel = "linear",

                  kpar = "automatic", C = 5, cross = 3, prob.model = TRUE)

  svmModel

  svmPrediction <- predict(svmModel, testingData, type = "response")

  svmPrediction

  newDf2 = data.frame(v1=testingData$willRecommend, v2=svmPrediction)

  newDf2

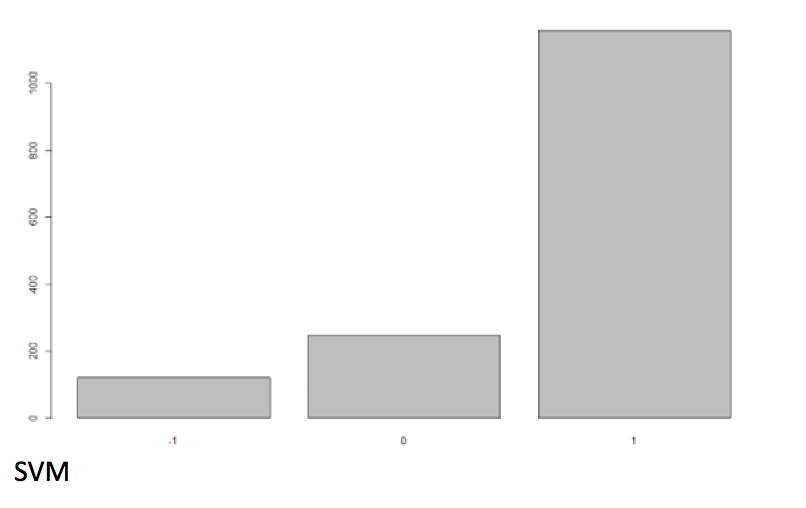
  svmError <- sum(testingData[,aim] != svmPrediction) \* 100 /length(svmPrediction)

  svmError

  # Error is 19.39712% by SVM

# Gamma: 0.035

# support vectors: 1309



* **Naïve Bayes model:**

############ Naive Bayes Model

  nbModel <- naiveBayes(trainingData[,aim] ~ ., data = trainingData, kernel = "linear",

                        kpar = "automatic", C = 5, cross = 3, prob.model = TRUE)

  nbModel

  nbPrediction <- predict(nbModel, testingData, type = "class")

  nbPrediction

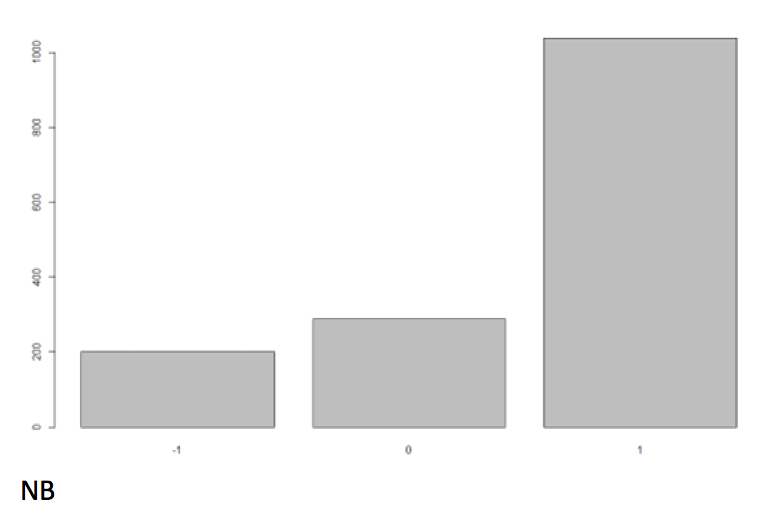
  newDf3 = data.frame(v1=testingData$willRecommend, v2=nbPrediction)

  newDf3

  nbError <- sum(testingData[,aim] != nbPrediction) \* 100 /length(nbPrediction)

  nbError

  #Error is 4.390564% by Naive Bayes



* **Linear model (On all attributes):**

tempDataFrame <- datasetFormation(hyattRegencyDataSetFiltered[,c(5,10:22)])

  trainingData <- as.data.frame(tempDataFrame[1])

  testingData <- as.data.frame(tempDataFrame[2])

  rm(tempDataFrame)

  ############ Linear Model

  lmodel <- lm(as.numeric(Likelihood\_Recommend\_H) ~ ., data = trainingData)

  summary(lmodel)

  plot(lmodel)

  lPred <- predict(lmodel, testingData, type = "response")

  lPred

  rootMeanSquareErrorLM <- mean((testingData$Likelihood\_Recommend\_H - lPred) ^ 2)

  rootMeanSquareErrorLM

  #Adjusted R-squared value = 0.6215 <<<1 and hence this model is not useful

  #Residual Standard Error = 1.157 on 3032 degrees of freedom

  #Hence Linear Model is not useful here and automatically dumped

* **Descriptive Analysis:**

As we observed in the beginning that we used the summary function to rule out some attributes which were not affecting the likelihood to recommend score, we used tapply and sqldf library to understand important attributes affecting the likelihood score. The analysis revealed the importance of few attributes like Customer Service, Hotel Condition, Customer’s Age and Golf Course Reachability were major factors in deciding promoters and detractors. Guest country was not considered since it did not contribute to the analysis. Attributes like business center availability, dry cleaning services and resort facility, limo and laundry services were not important for analysis, hence removed. Golf facility and self- parking were essential as they impact the net promoter score. Internet does not contribute much to the analysis.

 #####################################################

  #Checking Guest\_country and NPS

  guestslikelihoodCountrywise<- sqldf("select AVG(Likelihood\_Recommend\_H) as avgLikelihood, Count(X), GUEST\_COUNTRY\_R from  hyattRegencyDataSetFiltered GROUP BY GUEST\_COUNTRY\_R ORDER BY avgLikelihood")

  guestslikelihoodCountrywise

  #avgLikelihood below 7 is just for 5 customers which is very small compared 4600 customers present in database and hence customer's country

  #doensn't play any role in determing good NPS\_SCORE

  #######################Age and Gender , Tanay is doing

  #Yes and No factors

  #Remove this one, all are yes

  avgBusinessCenterNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Business.Center\_PL, mean)

  avgBusinessCenterNPS

  #Convention\_PL

  conventionNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Convention\_PL, mean)

  conventionNPS

  #Remove it , all are yes

  dryCleaningNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Dry.Cleaning\_PL, mean)

  dryCleaningNPS

  #Golf

  avgGolfNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Golf\_PL, mean)

  avgGolfNPS

  ######################

  #Write sql query to see how many business class played golf and how many didnt and how many overall played golf

  golfPlayedCount <- sqldf("select Count(Golf\_PL), POV\_CODE\_C from  hyattRegencyDataSetFiltered GROUP BY POV\_CODE\_C")

  golfPlayedCount

  golfPlayedNPSscore <- sqldf("select Count(Golf\_PL) FROM hyattRegencyDataSetFiltered WHERE Golf\_PL='N'")

  golfPlayedNPSscore

  golfPlayedNPSscore2 <- sqldf("select Count(Golf\_PL), CITY\_PL FROM hyattRegencyDataSetFiltered GROUP BY CITY\_PL HAVING Golf\_PL='N'")

  golfPlayedNPSscore2

  #Laundry\_PL

  laundryNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Laundry\_PL, mean)

  laundryNPS

  #Limo.Service\_PL

  limoNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Limo.Service\_PL, mean)

  limoNPS

  #Regency.Grand.Club\_PL

  regencyGrandNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Regency.Grand.Club\_PL, mean)

  regencyGrandNPS

  #Resort\_PL

  resortNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Resort\_PL, mean)

  resortNPS

  #Self.Parking\_PL ------ Important

  selfParkingNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Self.Parking\_PL, mean)

  selfParkingNPS

  #Y -NPS - 8.7444559

  #N- NPS- 7.841860

  #Shuttle.Service\_PL

  shuttleNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Shuttle.Service\_PL, mean)

  shuttleNPS

  #Spa\_PL

  spaNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Spa\_PL, mean)

  spaNPS

  #Valet.Parking\_PL

  valetNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Valet.Parking\_PL, mean)

  valetNPS

  #Booking\_Channel

  channelNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Booking\_Channel, mean)

  channelNPS

  #City\_PL

  cityNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$City\_PL, mean)

  cityNPS

  cityNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$City\_PL, median)

  cityNPS

  cityNPS<- tapply(hyattRegencyDataSetFiltered$X, hyattRegencyDataSetFiltered$City\_PL, length)

  cityNPS

  #Dallas has least Likelihood to recommend, maybe there is some problem in hyatt regency at dallas

  #Clublounge\_Used\_H

  clubUsedNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Clublounge\_Used\_H, mean)

  clubUsedNPS

  #Spa\_Used\_H

  spaUsedNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Spa\_Used\_H, mean)

  spaUsedNPS

  #Age\_Range\_H

  ageRangeNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Age\_Range\_H, mean)

  ageRangeNPS

  ageRangeNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Age\_Range\_H, median)

  ageRangeNPS

  ageRangeNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Age\_Range\_H, length)

  ageRangeNPS

  #NPS is low for age 26-35 , 36-45 with overall  , given customers count is similar to other ranges, 66-75 has 368 records and highest score

  #Gender\_H

  genderNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Gender\_H, mean)

  genderNPS

  genderNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Gender\_H, length)

  genderNPS

  genderNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$Gender\_H, median)

  genderNPS

  #NPS for male is less and customers count are almost equal

  #MEMBER\_STATUS\_R

  memberStatusNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$MEMBER\_STATUS\_R, mean)

  memberStatusNPS

  #LENGTH\_OF\_STAY\_C

  lengthStayNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$LENGTH\_OF\_STAY\_C, mean)

  lengthStayNPS

  lengthStayNPS<- tapply(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H, hyattRegencyDataSetFiltered$LENGTH\_OF\_STAY\_C, length)

  lengthStayNPS

* **Plots to understand factors affecting NPS:**

 ######################### GGPLOT2 #######################

  #Some plots again using gglplot2

  #Length of stay vs likelihoodRecommend

  projectScatteredBehaviour = ggplot(data = hyattRegencyDataSetFiltered,aes(x=as.numeric(hyattRegencyDataSetFiltered$Likelihood\_Recommend\_H), y=as.numeric(hyattRegencyDataSetFiltered$LENGTH\_OF\_STAY\_C))) + geom\_point(data=hyattRegencyDataSetFiltered, aes(size=Tranquility\_H,color= Condition\_Hotel\_H)) + labs(title = "Factors Affecting Likelihood to Recommend",y="Length of Stay",x="Likelihood To Recommend")

  projectScatteredBehaviour

  #this shows that likelihood\_recommend\_h is more when hotel condition is between 7.5-10. Tranquility is consistent for higher likelihood to recommend irrespective of length of stay

  #Guest Country vs Likelihood\_recommend\_H

  p <- ggplot(hyattRegencyDataSetFiltered, aes(GUEST\_COUNTRY\_R, Likelihood\_Recommend\_H))

  p + geom\_boxplot(outlier.colour = "blue", outlier.shape = 1,fill = "white", colour = "green") +theme(axis.text.x = element\_text(angle=90, hjust=1,vjust=0.3))

  #Plotting Factors like Tranquility, Guest\_room\_H, Condition\_H, Customer\_SVC\_H,staff\_Cared\_H against Likelihood\_Recommend\_H

  plot1<- ggplot(data=hyattRegencyDataSetFiltered, aes(x=Tranquility\_H,y=Likelihood\_Recommend\_H)) +

    geom\_bar(stat="summary",fun.data="mean\_se")  +

    labs(y="Likelihood to Recommend",x="Tranquility")

  plot1

  #Its almost nearly increasing with tranquility, that means likelihood is directly linked with tranquility

  #Internet Satisfaction vs Likelihood

  plot2<- ggplot(data=hyattRegencyDataSetFiltered, aes(x=Internet\_Sat\_H,y=Likelihood\_Recommend\_H)) +

    geom\_bar(stat="summary",fun.data="mean\_se") +

    labs(y="Likelihood to Recommend",x="Internet Satisfaction")

  plot2

  #It doesn't play a big role in determining the Overall Likelihood to Recommend

  #Guest\_Room\_H

  plot3<- ggplot(data=hyattRegencyDataSetFiltered, aes(x=Guest\_Room\_H,y=Likelihood\_Recommend\_H)) +

    geom\_bar(stat="summary",fun.data="mean\_se") +

    labs(y="Likelihood to Recommend",x="Guest Room Satisfaction")

  plot3

  # Guest\_Room\_H Plays an important role, linear trend

  #Condition\_Hotel\_H

  plot4<- ggplot(data=hyattRegencyDataSetFiltered, aes(x=Condition\_Hotel\_H,y=Likelihood\_Recommend\_H)) +

    geom\_bar(stat="summary",fun.data="mean\_se") +

    labs(y="Likelihood to Recommend",x="Condition\_Hotel\_H")

  plot4

  # Condition\_Hotel\_H Plays an important role, linear trend

  #Customer\_SVC\_H

  plot5<- ggplot(data=hyattRegencyDataSetFiltered, aes(x=Customer\_SVC\_H,y=Likelihood\_Recommend\_H)) +

    geom\_bar(stat="summary",fun.data="mean\_se") +

    labs(y="Likelihood to Recommend",x="Customer\_SVC\_H")

  plot5

  # Customer\_SVC\_H Plays an important role, linear trend

  #Staff\_Cared\_H

  plot6<- ggplot(data=hyattRegencyDataSetFiltered, aes(x=Staff\_Cared\_H,y=Likelihood\_Recommend\_H)) +

    geom\_bar(stat="summary",fun.data="mean\_se") +

    labs(y="Likelihood to Recommend",x="Staff\_Cared\_H")

  plot6

  # Staff\_Cared\_H Plays an important role, linear trend

  #Check\_In\_H

  plot7<- ggplot(data=hyattRegencyDataSetFiltered, aes(x=Check\_In\_H,y=Likelihood\_Recommend\_H)) +

    geom\_bar(stat="summary",fun.data="mean\_se") +

    labs(y="Likelihood to Recommend",x="Check\_In\_H")

  plot7

  # Check\_In\_H is not showing a proper trend and might not be involved directly in increasing likelihood\_recommend\_h

  #F.B\_Overall\_Experience\_H

  plot8<- ggplot(data=hyattRegencyDataSetFiltered, aes(x=F.B\_Overall\_Experience\_H,y=Likelihood\_Recommend\_H)) +

    geom\_bar(stat="summary",fun.data="mean\_se") +

    labs(y="Likelihood to Recommend",x="F.B\_Overall\_Experience\_H")

  plot8

  # F.B\_Overall\_Experience\_H Plays#Step 4- Look at all the data via a Heatmap

  install.packages("reshape2")

  library(reshape2)

  #Reusing modified Dataset named dfAirQualityNew formed above by melt function, putting y=variable

  # and x= Date, fill will show value which is value with each variable

  #Variable = O

  hyattHeatMapData <- hyattRegencyDataSetFiltered[,c(2,15,17,18,21,22)]

  hyattHeatMapData<- melt(hyattHeatMapData, measure.vars = c("Guest\_Room\_H", "Condition\_Hotel\_H","Customer\_SVC\_H","Check\_In\_H",'F.B\_Overall\_Experience\_H'))

  hyattHeatMapData$CHECK\_IN\_DATE\_C <- as.Date(hyattHeatMapData$CHECK\_IN\_DATE\_C)

  str( hyattHeatMapData$CHECK\_IN\_DATE\_C )

  heatMap <- ggplot(hyattHeatMapData, aes(x=CHECK\_IN\_DATE\_C, y=variable, fill=value)) + geom\_tile()

  heatMap <- heatMap +xlab("Dates in 2014") + ylab("Deciding Factors for customers retention") + ggtitle("Factors affecting NPS\_Score over period of months")

  heatMap

  #Check In service has improved over period of time and Customer\_SVC has gone down and remain bad

  #F.B service has improved over time Condition of Hotel has remain bad over period of time

  #Also, the guest\_room\_service has varied over period of time but has not been top notch

  hyattHeatMapData <- hyattRegencyDataSetFiltered[,c(2,27,32)]

  hyattHeatMapData<- melt(hyattHeatMapData, measure.vars = c("Golf\_PL",'Self.Parking\_PL'))

  hyattHeatMapData$CHECK\_IN\_DATE\_C <- as.Date(hyattHeatMapData$CHECK\_IN\_DATE\_C)

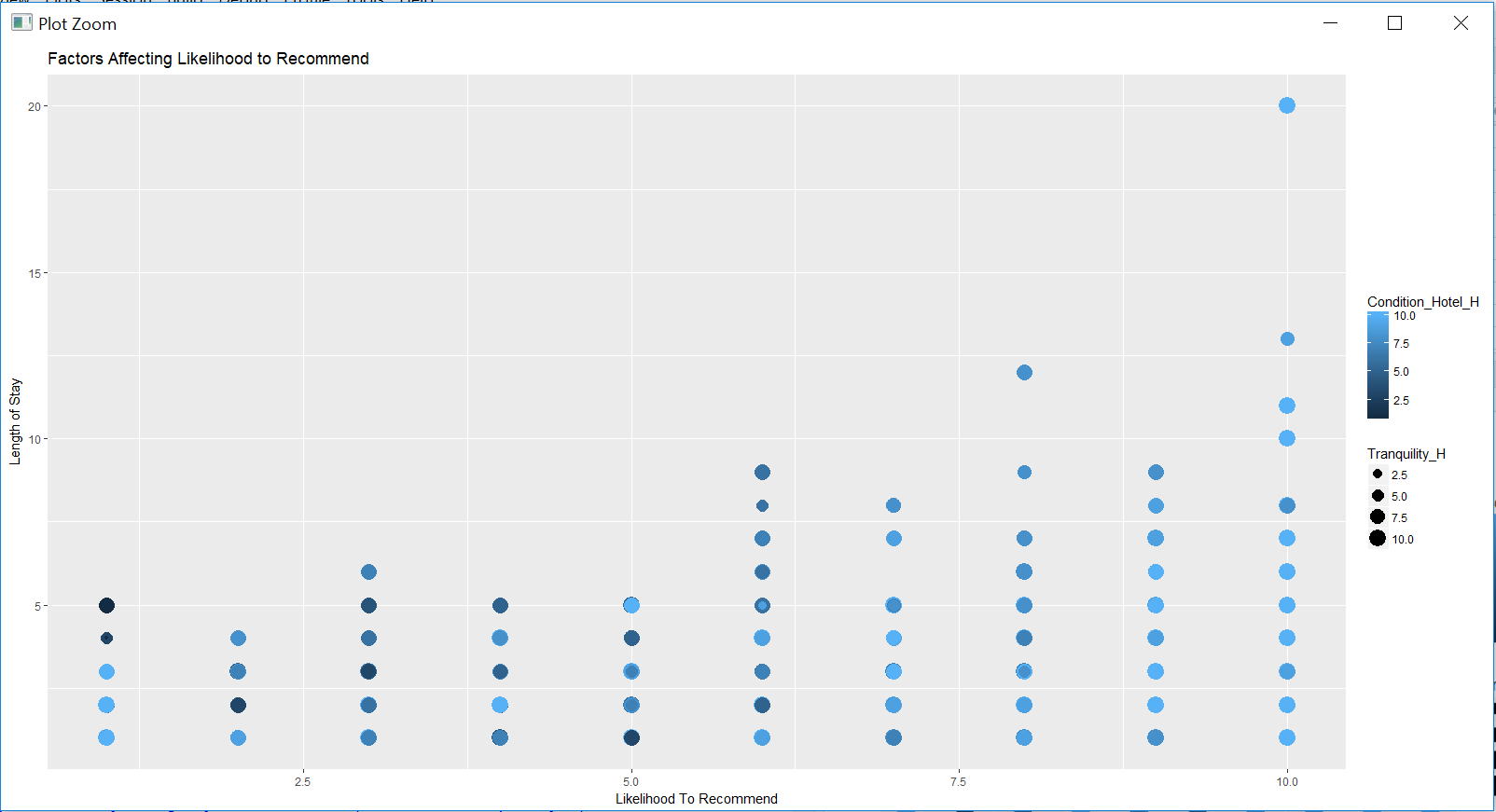
  str( hyattHeatMapData$CHECK\_IN\_DATE\_C )

  heatMap <- ggplot(hyattHeatMapData, aes(x=CHECK\_IN\_DATE\_C, y=variable, fill=value)) + geom\_tile()

  heatMap <- heatMap +xlab("Dates in 2014") + ylab("Deciding Factors for customers retention") + ggtitle("Factors affecting NPS\_Score over period of months")

  heatMap

  #Golf Facilities and Self-Parking facilities are staggering by the end of the year



A picture containing screenshot

Description generated with high confidence

A close up of a building

Description generated with high confidence

A screenshot of a cell phone

Description generated with very high confidence

A close up of a door

Description generated with high confidence

* **Linear modelling for the selected attributes with Likelihood\_Recommend\_H:**

#Linear Model for all the above observed factors

  ####################Linear Modeling############################# on the above factors

  #Likelihood on F.B\_Overall\_Experience\_H + Condition\_Hotel\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ F.B\_Overall\_Experience\_H + Condition\_Hotel\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value= 0.4837

  ##############F.B\_Overall\_Experience\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ F.B\_Overall\_Experience\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is very low = 0.08294

  ############Condition\_Hotel\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ Condition\_Hotel\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is 0.4586  <--- good

  ############Staff\_Cared\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ Staff\_Cared\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is 0.2025

  ############Customer\_SVC\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ Customer\_SVC\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is 0.437  <--- good

  ############Guest\_Room\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ Guest\_Room\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is 0.430  <--- good

  ############Tranquility\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ Tranquility\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is 0.149  <--- less

  ############Customer\_SVC\_H + Condition\_Hotel\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ Customer\_SVC\_H + Condition\_Hotel\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is 0.5748  <--- good

  ############Customer\_SVC\_H + Guest\_Room\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ Customer\_SVC\_H + Guest\_Room\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is 0.5744  <--- good

  ############Customer\_SVC\_H + Guest\_Room\_H

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ Customer\_SVC\_H + Tranquility\_H +Condition\_Hotel\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value is 0.5865  <--- good

  #Overall Linear Modeling based on important factors identified

  Likelihood\_lm <- lm(formula = Likelihood\_Recommend\_H ~ F.B\_Overall\_Experience\_H + Condition\_Hotel\_H+Staff\_Cared\_H+Customer\_SVC\_H+Guest\_Room\_H+Tranquility\_H, data= hyattRegencyDataSetFiltered)

  plot(Likelihood\_lm)

  summary(Likelihood\_lm)

  #R^2 value = 0.624 ---> combination of these attributes is strong

  #Important factors are Customer\_SVC\_H, Condition\_Hotel\_H, Guestroom\_H

* **Validation of results using arules:**

#Creating a subset again

  hyattRegencyDataSetFiltered2 = subset(hyattRegencyDataSetFiltered, select = -c(Business.Center\_PL,Convention\_PL,Dry.Cleaning\_PL,Laundry\_PL,

                                                                                 Limo.Service\_PL,Regency.Grand.Club\_PL,Resort\_PL,Shuttle.Service\_PL,

                                                                                 Spa\_PL,Valet.Parking\_PL,Booking\_Channel,NPS\_Type,Clublounge\_Used\_H,Spa\_Used\_H,REVENUE\_USD\_R,MEMBER\_STATUS\_R,

                                                                                 GUEST\_COUNTRY\_R,PMS\_FOOD\_BEVERAGE\_REV\_USD\_C,ADULT\_NUM\_C,LENGTH\_OF\_STAY\_C,CHECK\_IN\_DATE\_C, City\_PL,POV\_CODE\_C, Gender\_H, Check\_In\_H, Internet\_Sat\_H) )

  View(hyattRegencyDataSetFiltered2)

  #Use Apriori on this

  install.packages('arules')

  library(arules)

  install.packages('arulesViz')

  library(arulesViz)

  str(hyattRegencyDataSetFiltered2)

  #taking detractors to understand the data better

  #hyattDetractorData<- hyattRegencyDataSetFiltered2[which(hyattRegencyDataSetFiltered2$Likelihood\_Recommend\_H < 8,arr.ind = TRUE),]

  hyattDataFinal <-subset(hyattRegencyDataSetFiltered2, select = -c(X) )

  head(hyattDataFinal)

  ################Change Range of values   >=8 and <=10 = Good  :  <8 and >=7 = Average : < 7 Bad

  hyattDataFinal <- within(hyattDataFinal, Guest\_Room\_H[Guest\_Room\_H >=0 & Guest\_Room\_H < 7] <- -1)

  hyattDataFinal <- within(hyattDataFinal, Guest\_Room\_H[Guest\_Room\_H >8 & Guest\_Room\_H <=10] <- 1)

  hyattDataFinal <- within(hyattDataFinal, Guest\_Room\_H[Guest\_Room\_H >=7 & Guest\_Room\_H <= 8] <-  0)

  hyattDataFinal <- within(hyattDataFinal, Tranquility\_H[Tranquility\_H >=0 & Tranquility\_H < 7] <- -1)

  hyattDataFinal <- within(hyattDataFinal, Tranquility\_H[Tranquility\_H >8 & Tranquility\_H <=10] <- 1)

  hyattDataFinal <- within(hyattDataFinal, Tranquility\_H[Tranquility\_H >=7 & Tranquility\_H <=8] <-  0)

  hyattDataFinal <- within(hyattDataFinal, Condition\_Hotel\_H[Condition\_Hotel\_H >=0 & Condition\_Hotel\_H < 7] <- -1)

  hyattDataFinal <- within(hyattDataFinal, Condition\_Hotel\_H[Condition\_Hotel\_H >8 & Condition\_Hotel\_H <=10] <- 1)

  hyattDataFinal <- within(hyattDataFinal, Condition\_Hotel\_H[Condition\_Hotel\_H >=7 & Condition\_Hotel\_H <=8] <-  0)

  hyattDataFinal <- within(hyattDataFinal, Customer\_SVC\_H[Customer\_SVC\_H >=0 & Customer\_SVC\_H < 7] <- -1)

  hyattDataFinal <- within(hyattDataFinal, Customer\_SVC\_H[Customer\_SVC\_H >8 & Customer\_SVC\_H <=10] <- 1)

  hyattDataFinal <- within(hyattDataFinal, Customer\_SVC\_H[Customer\_SVC\_H >=7 & Customer\_SVC\_H <=8] <-  0)

  hyattDataFinal <- within(hyattDataFinal, Staff\_Cared\_H[Staff\_Cared\_H >=0 & Staff\_Cared\_H < 7] <- -1)

  hyattDataFinal <- within(hyattDataFinal, Staff\_Cared\_H[Staff\_Cared\_H >8 & Staff\_Cared\_H <=10] <- 1)

  hyattDataFinal <- within(hyattDataFinal, Staff\_Cared\_H[Staff\_Cared\_H >=7 & Staff\_Cared\_H <=8] <-  0)

  hyattDataFinal <- within(hyattDataFinal, F.B\_Overall\_Experience\_H[F.B\_Overall\_Experience\_H >=0 & F.B\_Overall\_Experience\_H < 7] <- -1)

  hyattDataFinal <- within(hyattDataFinal, F.B\_Overall\_Experience\_H[F.B\_Overall\_Experience\_H >8 & F.B\_Overall\_Experience\_H <=10] <- 1)

  hyattDataFinal <- within(hyattDataFinal, F.B\_Overall\_Experience\_H[F.B\_Overall\_Experience\_H >=7 & F.B\_Overall\_Experience\_H <=8] <-  0)

  #Since the columns selected were either characters or integers, it was necessary to convert them as factors in order to perform apriori

  hyattDataFinal$Age\_Range\_H <- as.factor(hyattDataFinal$Age\_Range\_H)

  hyattDataFinal$Guest\_Room\_H <- as.factor(hyattDataFinal$Guest\_Room\_H)

  hyattDataFinal$Tranquility\_H <- as.factor(hyattDataFinal$Tranquility\_H)

  hyattDataFinal$Condition\_Hotel\_H <- as.factor(hyattDataFinal$Condition\_Hotel\_H)

  hyattDataFinal$Customer\_SVC\_H <- as.factor(hyattDataFinal$Customer\_SVC\_H)

  hyattDataFinal$Staff\_Cared\_H <- as.factor(hyattDataFinal$Staff\_Cared\_H)

  hyattDataFinal$F.B\_Overall\_Experience\_H <- as.factor(hyattDataFinal$F.B\_Overall\_Experience\_H)

  hyattDataFinal$Self.Parking\_PL <- as.factor(hyattDataFinal$Self.Parking\_PL)

  hyattDataFinal$Golf\_PL <- as.factor(hyattDataFinal$Golf\_PL)

  #Removing Likelihood\_recommend\_h because we have will recommend field based on thats

  hyattDataFinal <-subset(hyattDataFinal, select = -c(Likelihood\_Recommend\_H) )

  #Performing Data association rules using apriori

  #Having the confidence as 0.8 and support as 0.01 gives the rules which are likely to predict one another

  rulesSet <- apriori(hyattDataFinal,parameter=list(support=0.01, confidence=0.8),

                     appearance=list(default='lhs', rhs=('willRecommend=-1')))

  goodRules <- rulesSet[quality(rulesSet)$lift>8]

  goodRules <- sort(goodRules,by='lift',decreasing=T)

  goodRules

  inspect(head(goodRules, 10))

  #Plotting the rule1

  plot(rulesSet)

  ################# Top 10 rules for detractors

  ## for Detractors

  #{Tranquility\_H=-1,Customer\_SVC\_H=-1} 8.658487    51

  #{Guest\_Room\_H=-1,Condition\_Hotel\_H=-1,Customer\_SVC\_H=-1} 8.417517    87

  #{Guest\_Room\_H=-1,Customer\_SVC\_H=-1,Golf\_PL=Y} 8.387999    55

  #{Guest\_Room\_H=-1, Customer\_SVC\_H=-1,Golf\_PL=Y, Self.Parking\_PL=Y} 8.387999   55

  #{Guest\_Room\_H=-1, Condition\_Hotel\_H=-1, Customer\_SVC\_H=-1, F.B\_Overall\_Experience\_H=1} 8.346004    64

  #Having the confidence as 0.8 and support as 0.01 gives the rules which are likely to predict one another

  #For Passives

   rulesSet <- apriori(hyattDataFinal,parameter=list(support=0.01, confidence=0.8),

                      appearance=list(default='lhs', rhs=('willRecommend=0')))

  goodRules <- rulesSet[quality(rulesSet)$lift>4]

  goodRules <- sort(goodRules,by='lift',decreasing=T)

  goodRules

  inspect(head(goodRules, 10))

  #Plotting the rule1

  plot(rulesSet)

  ####Passives Rules

  #{Guest\_Room\_H=0,Tranquility\_H=0,Condition\_Hotel\_H=0,Customer\_SVC\_H=0,Staff\_Cared\_H=0, Self.Parking\_PL=Y} 4.421862    59

  #{Guest\_Room\_H=0,Tranquility\_H=0, Condition\_Hotel\_H=0,Customer\_SVC\_H=0, Staff\_Cared\_H=0}  4.371898    60

  #{Guest\_Room\_H=0,Condition\_Hotel\_H=0,Customer\_SVC\_H=0,Staff\_Cared\_H=0,Self.Parking\_PL=Y} 4.362694    79

  #{Condition\_Hotel\_H=0,Customer\_SVC\_H=0,Staff\_Cared\_H=0,Golf\_PL=N,Self.Parking\_PL=Y}  4.357078 49

  #{Guest\_Room\_H=0,Tranquility\_H=0,Condition\_Hotel\_H=0,Staff\_Cared\_H=0,Self.Parking\_PL=Y}  4.350571    68

  #For Promoters

  rulesSet <- apriori(hyattDataFinal,parameter=list(support=0.01, confidence=0.8),

                      appearance=list(default='lhs', rhs=('willRecommend=1')))

  goodRules <- rulesSet[quality(rulesSet)$lift>1.4]

  goodRules <- sort(goodRules,by='lift',decreasing=T)

  goodRules

  inspect(head(goodRules, 10))

  #Plotting the rule1

  plot(rulesSet)

  # Promoters

  #{Age\_Range\_H=46-55,Guest\_Room\_H=1,Tranquility\_H=1,Customer\_SVC\_H=1,Staff\_Cared\_H=1, F.B\_Overall\_Experience\_H=0} => 1.402309    47

  #{Age\_Range\_H=46-55,Guest\_Room\_H=1,Tranquility\_H=1,Customer\_SVC\_H=1,Staff\_Cared\_H=1, F.B\_Overall\_Experience\_H=0,Self.Parking\_PL=Y} 1.402309    47

  #{Age\_Range\_H=46-55, Tranquility\_H=1, Condition\_Hotel\_H=1,Customer\_SVC\_H=1,Staff\_Cared\_H=1, F.B\_Overall\_Experience\_H=0} => 1.401674    46

  #{Age\_Range\_H=46-55, Guest\_Room\_H=1, Tranquility\_H=1, Condition\_Hotel\_H=1, Customer\_SVC\_H=1,Staff\_Cared\_H=1,F.B\_Overall\_Experience\_H=0} => 1.401674    46

  #{Age\_Range\_H=46-55,Tranquility\_H=1,Condition\_Hotel\_H=1,Customer\_SVC\_H=1,Staff\_Cared\_H=1, F.B\_Overall\_Experience\_H=0, Self.Parking\_PL=Y} 1.401674    46