HYATT HOTELS SURVEY DATA ANALYSIS

BATCH M006 TEAM 4

Aatif Desai

Ankita Biswas

Bhavik Lalwani

Rahul Jairaj

Rushabh Shah

Table of Contents

➢Introduction

Hyatt Hotels, a franchiser of hotels, resorts, and vacation properties performed a comprehensive annual survey for 679 properties in 54 countries from February 2014 to January 2015. The survey gathered information about various characteristics of hotels from all over the world, and quality of stay, which influenced the overall customer experience. This data could potentially give useful insights as to what factors are responsible to attract the customers and what are the areas which need improvement. Some of these factors were aggregated to determine the Net Promoter Score (NPS) type of a customer, categorizing them as “Promoter”, “Passive” or “Detractor” based on their experience with the hotel. Using the concise results provided by the NPS, Hyatt can strategize its endeavors to enhance the customer experience and maximize profit.

This survey resulted in a voluminous dataset consisting of over 15 million customers’ data. Using the concepts that we have studied through the course we plan to perform analysis on this dataset and determine which factors influenced a customer’s to be a Promoter, Passive or Detractor. Out of the 12 months of data, we had initially planned to use 4 months evenly interspersed through the year - February, May, August and December. These 4 months were chosen strategically to cover a variety of times - vacations to peak working months. However, later, due to want of more data for analysis, we decided to go with the entire dataset of 12 months.

➢Business Rules and Assumptions:

During the analysis of the data, we assumed certain entities that helped us predict the output. The assumptions are as follows:

- Likelihood to Recommend has a substantial role in determining the NPS type.
- Increasing the customer satisfaction on services and the quality of stay of detractors can convert them to promoters.
- Improving the array of amenities available to customers can convert detractors into promoters.
- We are to focus on detractors, as they take away future business, so the scope of the project would be decided by areas with less NPS and more number of detractors.

We have assumed these conditions while analyzing the data.

The aforementioned services may include Internet satisfaction, or the Customer Service satisfaction and the amenities mentioned include restaurants or conference centers near the hotel etc.

➢Business Questions:

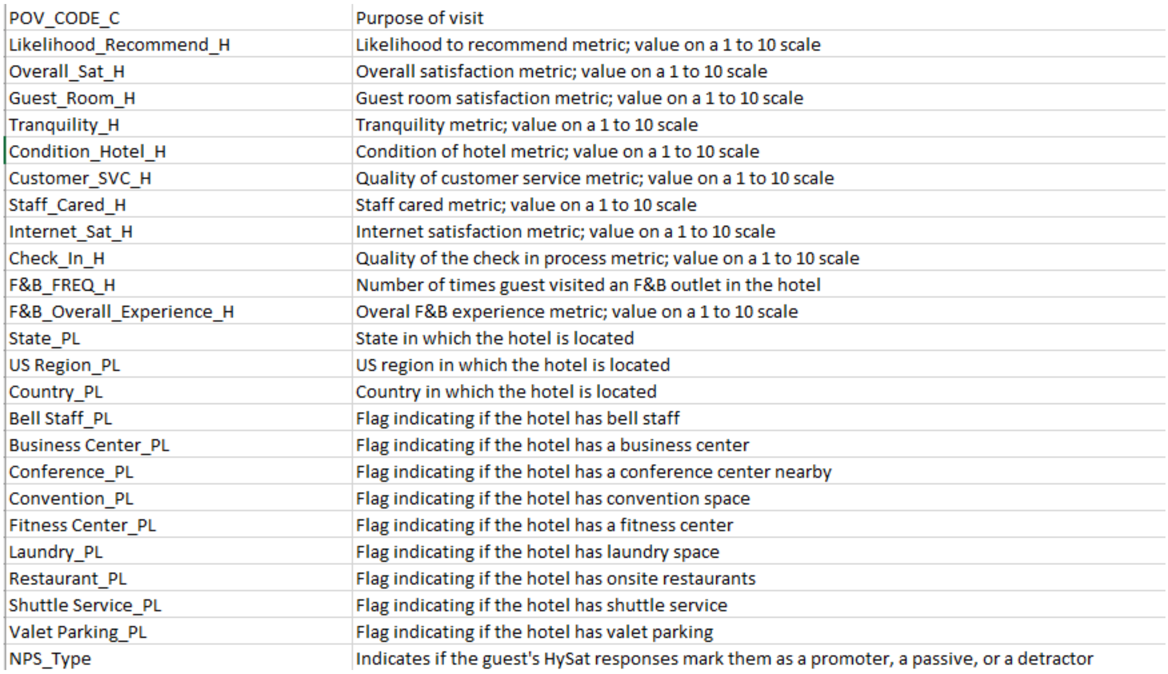
- What is the spread of customers with different Purpose of Visit across the three NPS types?
- What is the average Likelihood to recommend across different countries in the world?
- What is the average NPS across different countries in the world?
- Which countries have the most number of detractors?
- Which states within the United States have the lowest NPS?
- Which states within the United States have the highest number of detractors?
- Which survey factors have the most impact on Likelihood to Recommend?
- Which survey factors have most impact on NPS type across different purpose of visits?
- The presence of which amenities have most impact on NPS type across different purposes of visits?

➢Data Acquisition/Cleaning/Transformation/Munging:

The data provided to us was of 12 months and there were 12 csv files in all; one for every month. Since the amount of data was very large, we thought it best to perform our analysis on 4 out of 12 months. The months which we chose were February, May, August and December. However, due to a want of more data after cleaning, we decided to expand on all 12 months for our analysis. The dataset consists of 237 variables, out of which many of them were not relevant for the analysis.

Steps:

- We commenced our data analysis selecting the months to work on.
- We decided the columns based on the relevance of each column that will influence the NPS type as follows:



POV_CODE_C	Purpose of visit
Likelihood_Recommend_H	Likelihood to recommend metric; value on a 1 to 10 scale
Overall_Sat_H	Overall satisfaction metric; value on a 1 to 10 scale
Guest_Room_H	Guest room satisfaction metric; value on a 1 to 10 scale
Tranquility_H	Tranquility metric; value on a 1 to 10 scale
Condition_Hotel_H	Condition of hotel metric; value on a 1 to 10 scale
Customer_SVC_H	Quality of customer service metric; value on a 1 to 10 scale
Staff_Cared_H	Staff cared metric; value on a 1 to 10 scale
Internet_Sat_H	Internet satisfaction metric; value on a 1 to 10 scale
Check_In_H	Quality of the check in process metric; value on a 1 to 10 scale
F&B_FREQ_H	Number of times guest visited an F&B outlet in the hotel
F&B_Overall_Experience_H	Overal F&B experience metric; value on a 1 to 10 scale
State_PL	State in which the hotel is located
US Region_PL	US region in which the hotel is located
Country_PL	Country in which the hotel is located
Bell Staff_PL	Flag indicating if the hotel has bell staff
Business Center_PL	Flag indicating if the hotel has a business center
Conference_PL	Flag indicating if the hotel has a conference center nearby
Convention_PL	Flag indicating if the hotel has convention space
Fitness Center_PL	Flag indicating if the hotel has a fitness center
Laundry_PL	Flag indicating if the hotel has laundry space
Restaurant_PL	Flag indicating if the hotel has onsite restaurants
Shuttle Service_PL	Flag indicating if the hotel has shuttle service
Valet Parking_PL	Flag indicating if the hotel has valet parking
NPS_Type	Indicates if the guest's HySat responses mark them as a promoter, a passive, or a detractor

- We read the relevant data for the selected months and combined them.

- We cleaned the dataset the following way:
    - For NPT type, Likelihood to Recommend and Countries:

      We removed all customer data which was blank in these column to perform analysis on them.
    - For numerical variables – 1 to 10 (survey options)

      We looked at the number of entries in the variable that were blank. If a vast majority of them were blank, like in the case of Internet Satisfaction, we dropped that variable from our analysis.

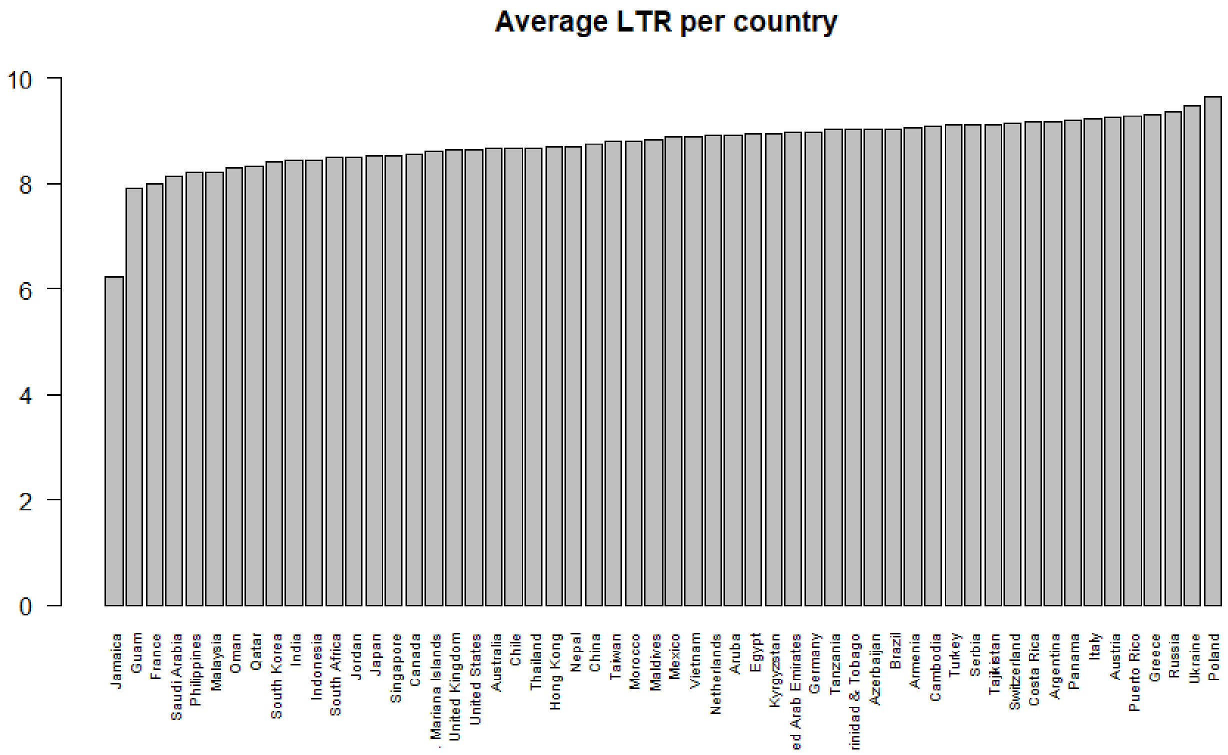
      For the rest, we substituted the mean of the non-blank data into the blank spaces, so that we can use that customer's data and not omit it.
    - For flag variables – "Y" and "N" (amenities):

      We removed all the rows which were blank so that we can perform analysis on them.

➢Descriptive Statistics

● Statistical Distribution of Average LTR per Country

We performed an analysis on the factor – "Likelihood to Recommend" for each country and calculated the average value for the same. The graph below shows the plotting of Likelihood to Recommend, ranging from 0 to 10, against each of the country.



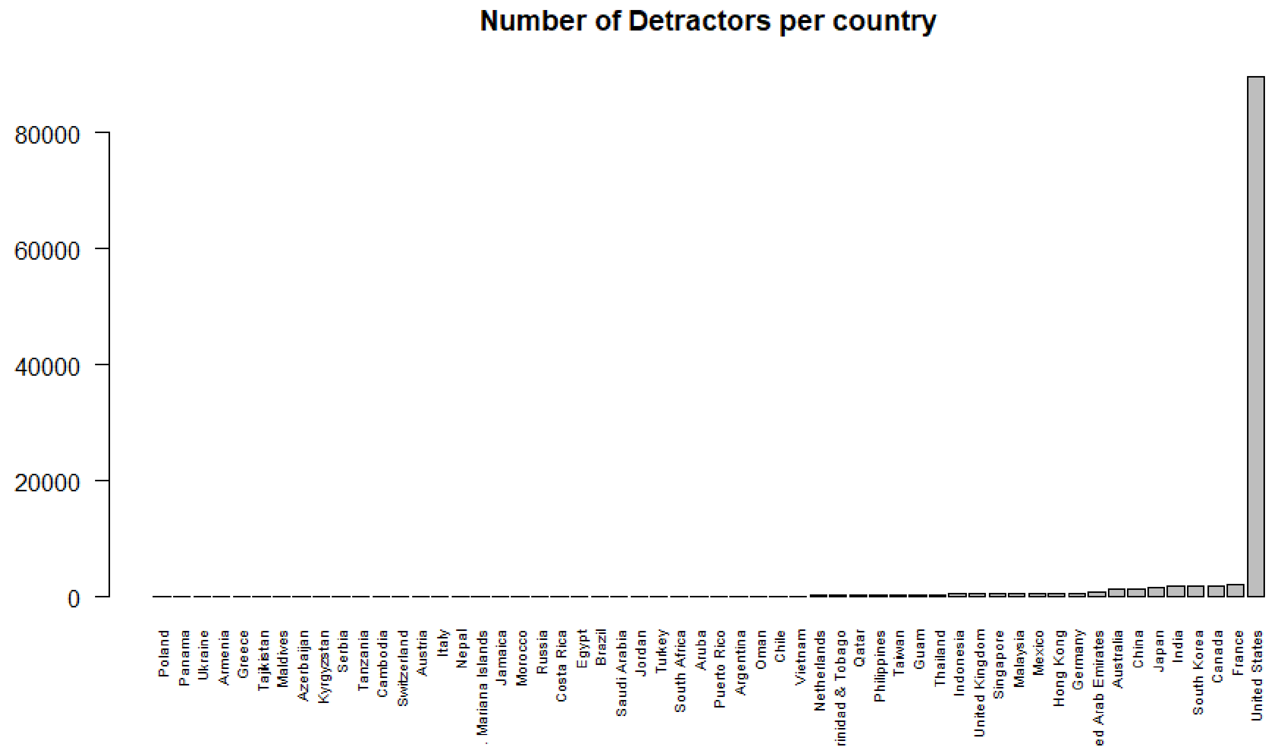
Inference from the graph:

- Jamaica has the lowest average LTR per country at **6.22**
- Poland has the highest average LTR per country at **9.64**
- US is at around the bottom one third at **8.64**

This by itself, however, doesn’t provide enough evidence to focus our analysis on one area.

- **Statistical Distribution on the number of Detractors per Country**

This graph shows the number of detractors per country:



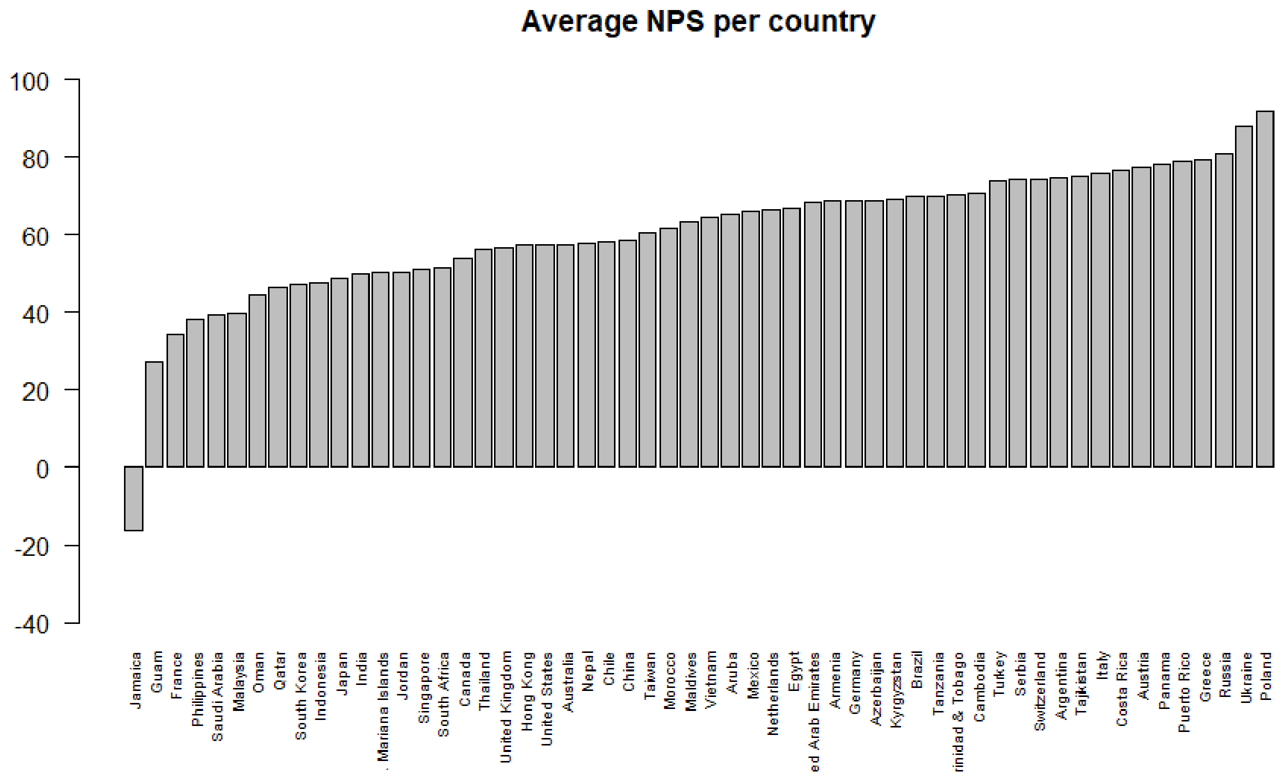
Inference from the Graph:

- USA has the most number of detractors by a large margin at **89565**.

This find is interesting, but not enough to choose USA as our focus. Large number of detractors doesn’t mean much by itself – it could be accompanied with a large number of promoters, there may be a large number of hotels in the US, and the population may be higher than the other countries.

Statistical Distribution Of Average NPS per Country

The below bar-plot represents the average NPS calculated per country. NPS is calculated by taking a difference between the percentage of promoters and the percentage of detractors per country.



Inference from the graph:

- Jamaica has the lowest NPS at **-16%**
- Poland has the highest NPS at **91%**
- US has NPS of **57.36%,** it's still in the bottom half.

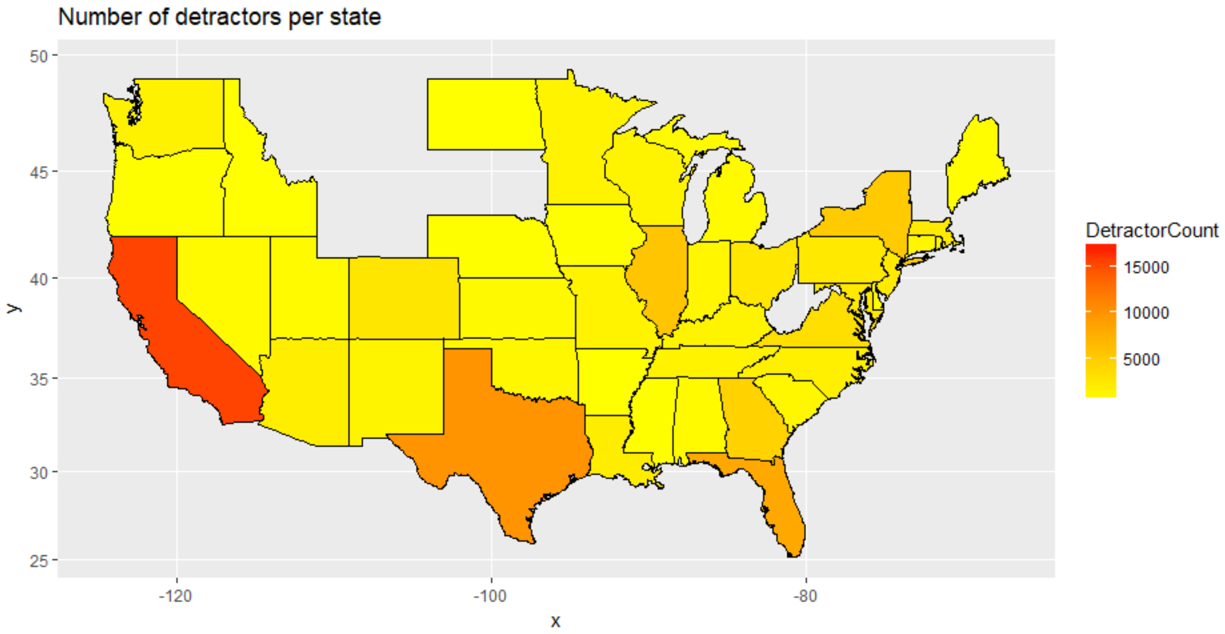
This information in combination with the previous graph, showing the number of detractors, has compelled us to take a closer look at the data in the United States – which has the most number of detractors and low average NPS.

Even a small increase in NPS can lead to many promoters being created from detractors and hence large profits for the organization.

So, we decide to focus on USA:

- ## Statistical Distribution on the Number of Detractors per State

The map displays the number of detractors per state:

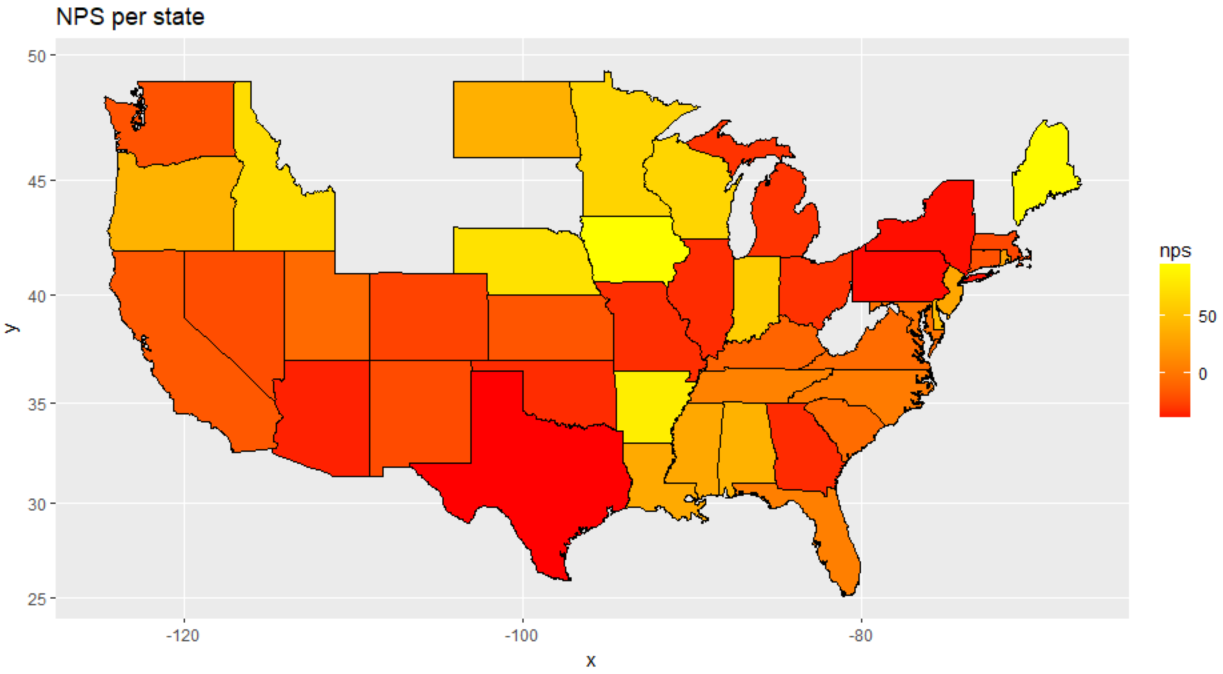


Inference from the map:

- California has the highest number of detractors at **15619**
- Texas has the **2nd** at **9905**.
- Arkansas had the least at **45**.

Statistical Distribution On NPS per State

The following map shows a distribution of average NPS per state:



Inference from the map:

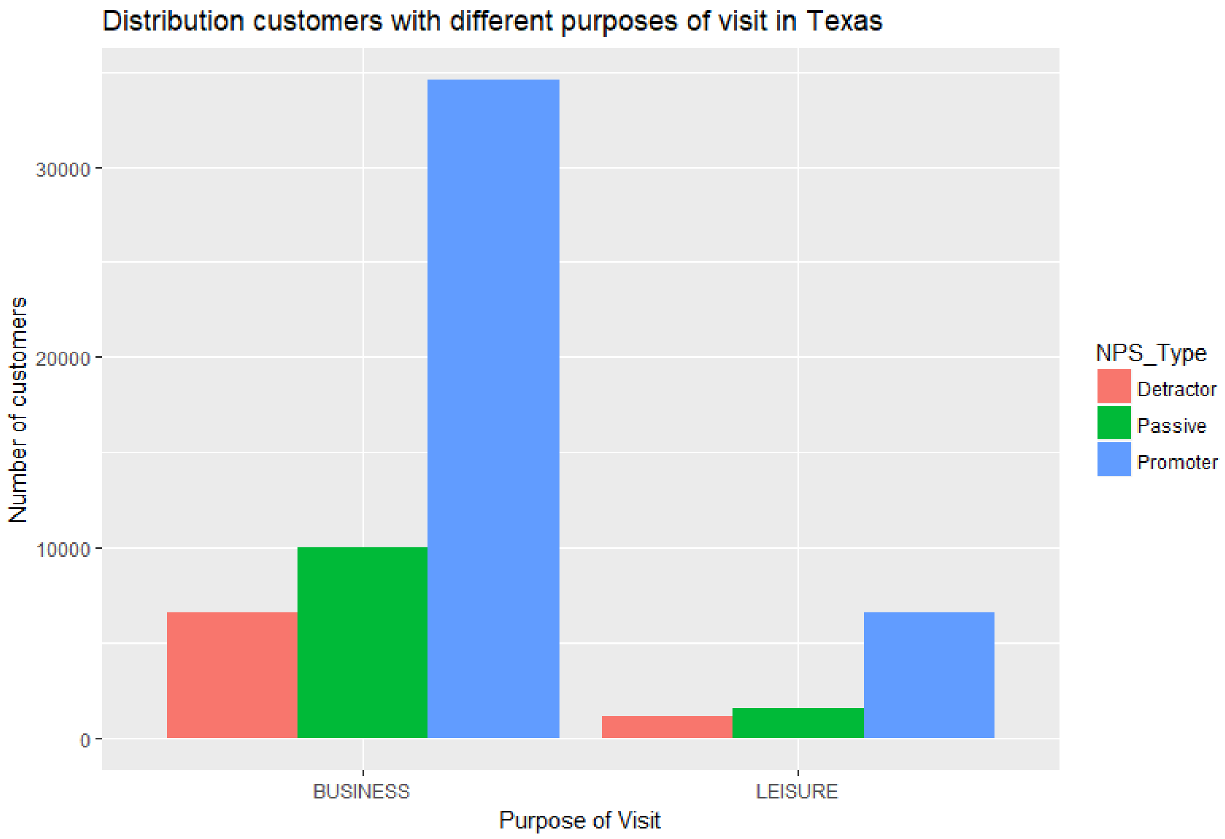
- Iowa has the highest NPS of **93.9%.**
- Texas has the least NPS at **-38.5%.**

Since it has the least value of NPS and one of the highest detractor count, it requires a lot of improvement, and hence we have based our further analysis on the state of Texas.

Going by the same logic of country selection, due to the 2 factors mentioned, conversion of detractors to passive and passive to promoters in Texas could help considerably further profits for the company.

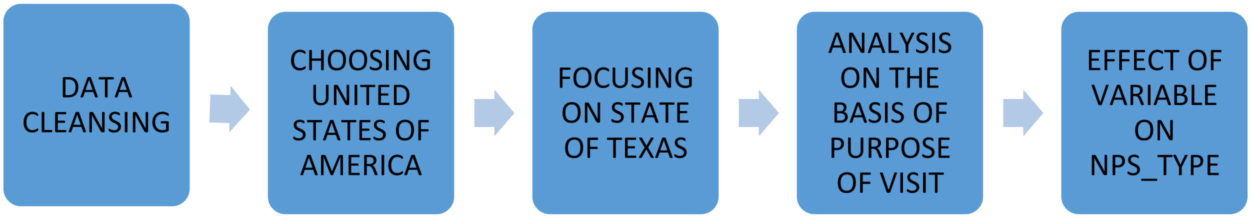
Statistical Distribution between Business and Leisure visitors in bar chart

Below is the distribution of customers based on purpose of visit – Business or Leisure.



Inference from the bar graph:

- There are far more business customers than leisure (**51,101** compared to **9239**).
- The distribution of promoters, passives and detractors is similar for both business and leisure customers – i.e. the shape of the graphs is similar

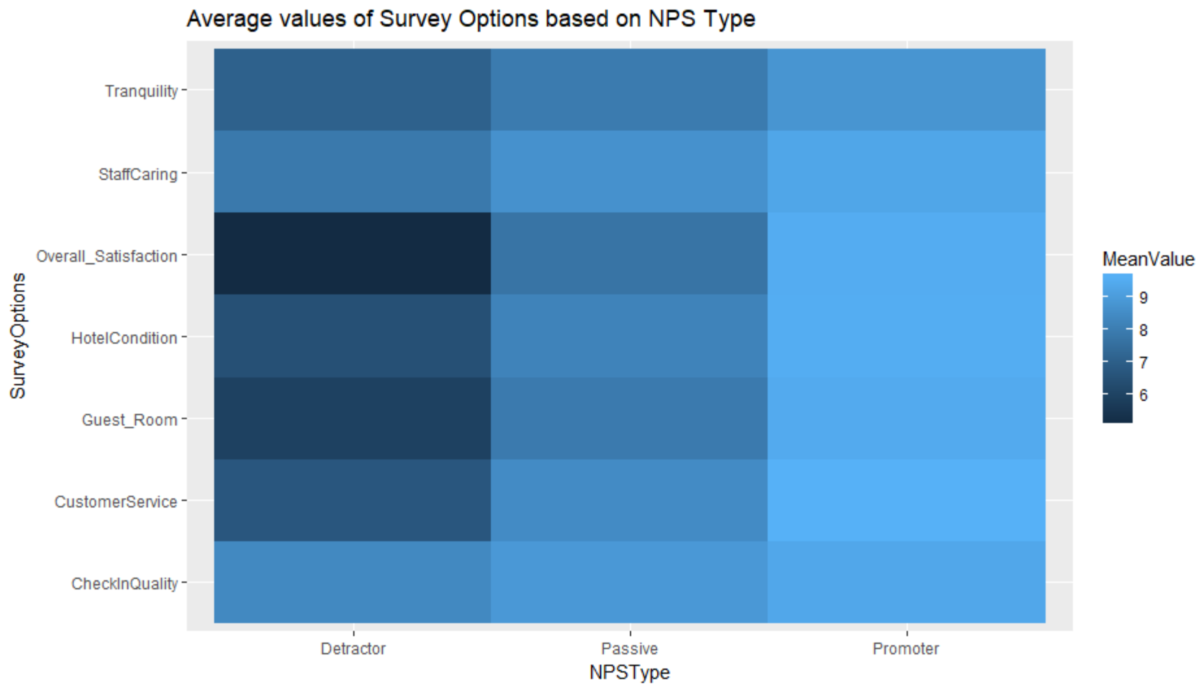


DATA CLEANSING → CHOOSING UNITED STATES OF AMERICA → FOCUSING ON STATE OF TEXAS → ANALYSIS ON THE BASIS OF PURPOSE OF VISIT → EFFECT OF VARIABLE ON NPS_TYPE

➢Statistical Distribution using HeatMap

To find the spread of certain variables across the NPS type, we have considered the following survey factors:

- Overall Satisfaction
- Guest Room Satisfaction
- Tranquility
- Hotel Condition
- Customer Service
- Caring of Staff
- Check-In Quality



Inference from the Heatmap:

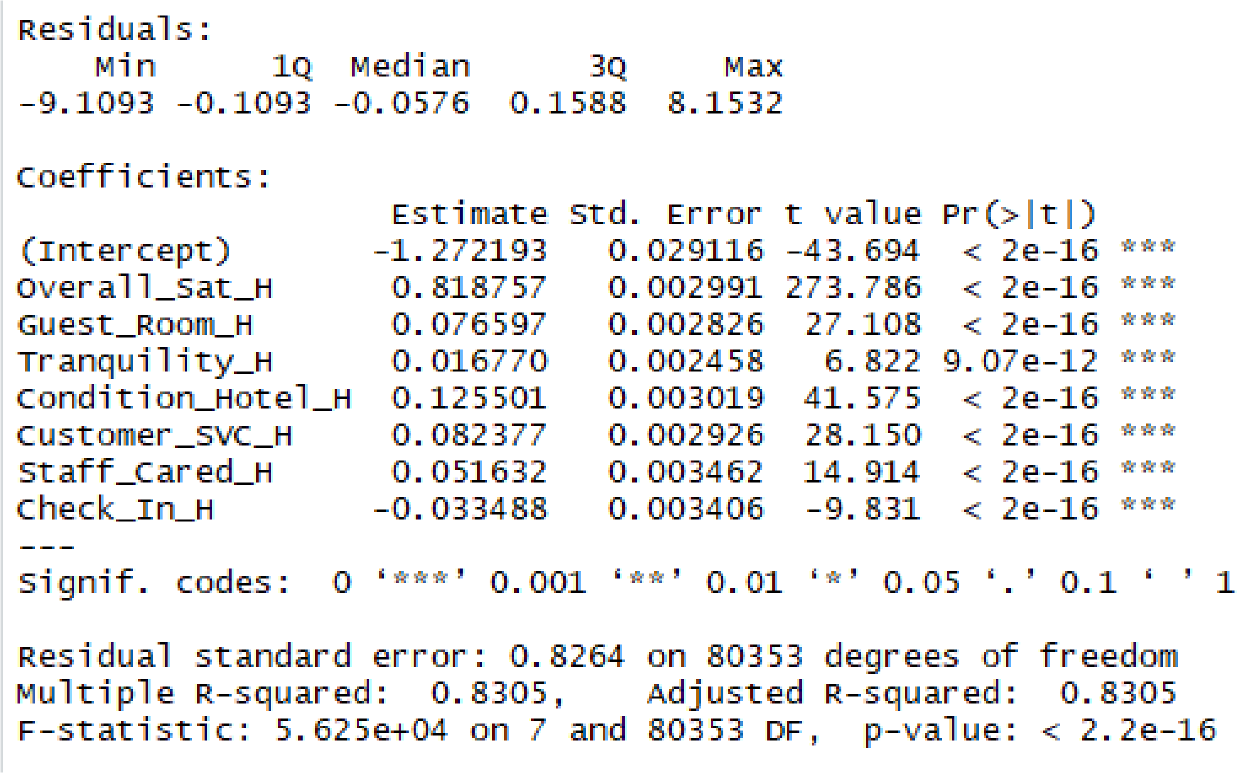
- Overall satisfaction is the most consistent metric to show satisfaction among promoters (**9.51**), detractors (**5.08**) and passive (**7.7**).
- Tranquility has the lowest score at **8.77**, followed by Check-In Quality at **9.36** among promoters, compared to other factors.
- Guest Room at **7.91** and Tranquility at **7.93** have the lowest average rating for passives, apart from Overall Satisfaction.
- Guest Room (**5.93**) and Customer Service (**6.69**) had the lowest mean value among detractors, apart from Overall Satisfaction.

➢Linear Modelling

We have performed linear modelling on factors used above as to evaluate its impact on Likelihood to recommend:

Fields chosen:

- Overall Satisfaction
- Guest Room Satisfaction
- Tranquility
- Condition of the Hotel
- Customer Service
- Caring of Staff
- Check in Quality



```
Residuals:
    Min      1Q  Median      3Q     Max
-9.1093 -0.1093 -0.0576  0.1588  8.1532

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       -1.272193   0.029116 -43.694  < 2e-16 ***
Overall_Sat_H      0.818757   0.002991 273.786  < 2e-16 ***
Guest_Room_H       0.076597   0.002826  27.108  < 2e-16 ***
Tranquility_H      0.016770   0.002458   6.822 9.07e-12 ***
Condition_Hotel_H  0.125501   0.003019  41.575  < 2e-16 ***
Customer_SVC_H     0.082377   0.002926  28.150  < 2e-16 ***
Staff_Cared_H      0.051632   0.003462  14.914  < 2e-16 ***
Check_In_H        -0.033488   0.003406  -9.831  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8264 on 80353 degrees of freedom
Multiple R-squared:  0.8305,    Adjusted R-squared:  0.8305
F-statistic: 5.625e+04 on 7 and 80353 DF,  p-value: < 2.2e-16
```

Inferences from the Linear Model:

- Excellent R squared value of **0.8305** indicates that the selected fields can determine Likelihood to Recommend with a confidence of **83%**
- Low p values (**under 2.2e-16**) show that they are all statistically significant as well.
- Low standard errors for all coefficients is a good sign.

➢ Analysis of Variables using Association Rules

We have performed Association Rules separately for both purposes of visit - Business and Leisure.

Apart from the previous data consideration of within Texas, we have done the following as well:

- Creating categorical variables from numeric ones (survey options) for processing. For example, consider Overall Satisfaction. We have created a variable O_Type, which is high for Overall Satisfaction higher than 8, Medium for Overall Satisfaction between 5 and 7, and Low for Overall Satisfaction 4 and under.

  These are as follows:

  - I_Type(Check In Process)
  - F_Type(Staff Cared)
  - S_Type(Customer Service)
  - C_Type(Condition of Hotel)
  - T_Type(Tranquility)
  - G_Type(Guest Room)

- O_Type(Overall Satisfaction)

- Splitting the dataset into two – one for business users and the other for leisure users.

  Business users specifically have

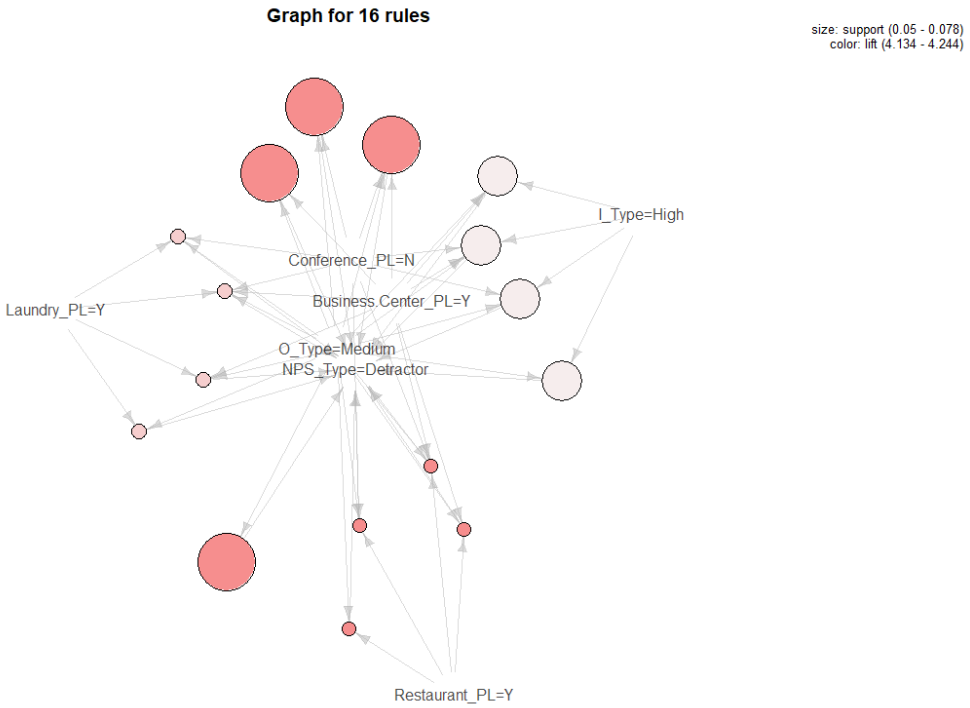
  - Valet parking
  - Business Center
  - Conference
  - Convention Center
  - Restaurant
  - Valet Parking
  - Laundry

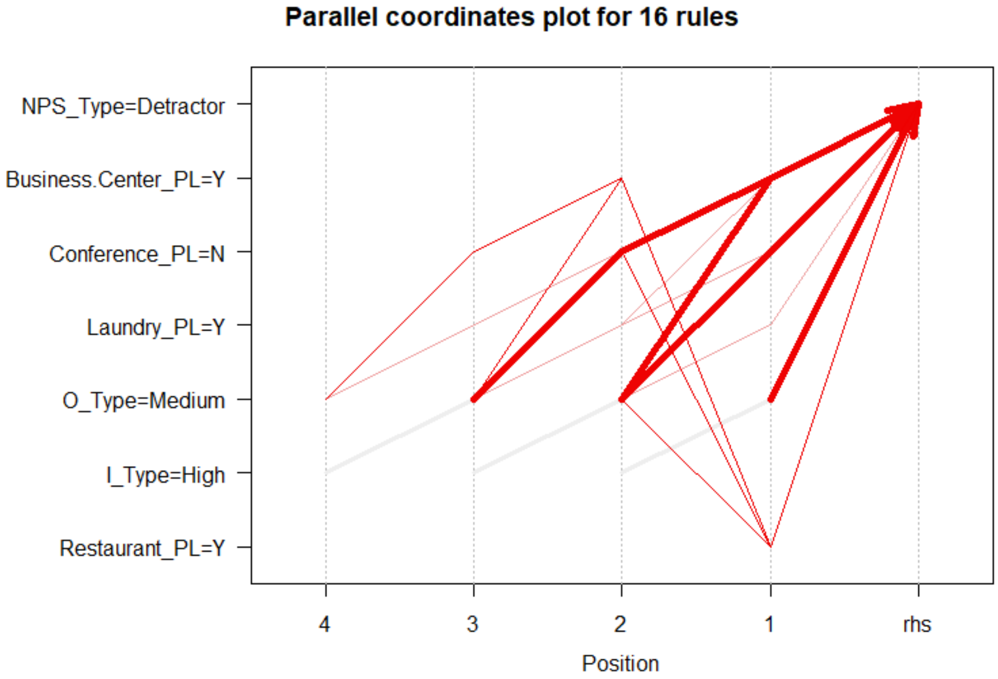
  Leisure users specifically have

  - Shuttle Service
  - Valet Parking
  - Bell Staff
  - Laundry
  - Restaurant

• Purpose of Visit = Business Type and NPS type = Detractor

The following are plots for rules mined for Business Detractors.





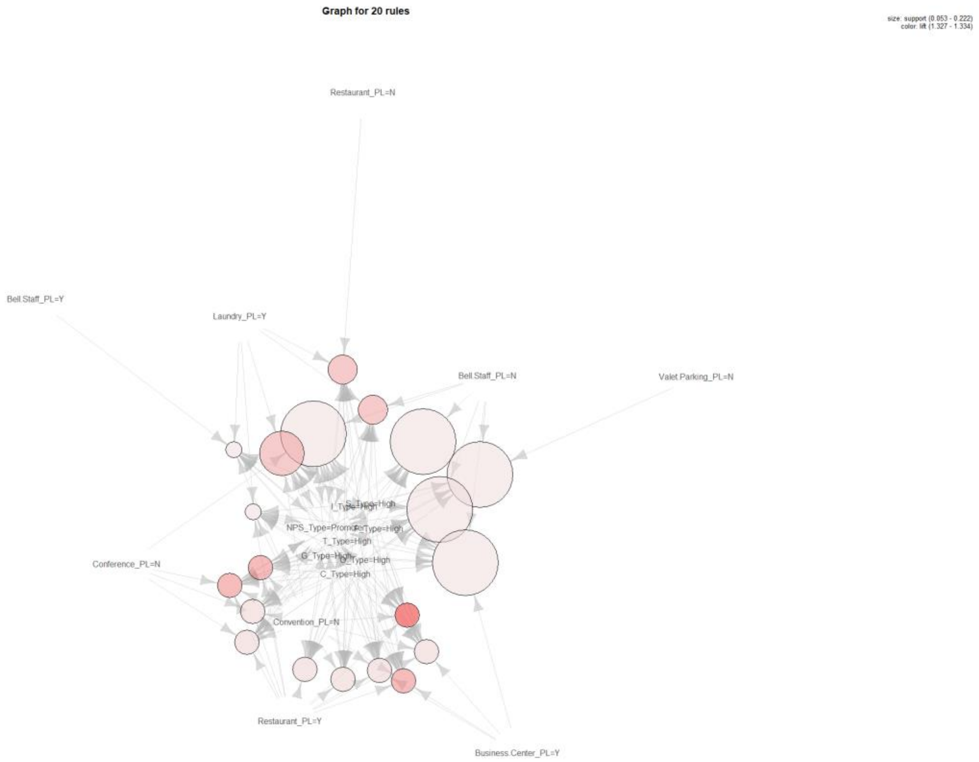
Inferences from the above the plotted rules:

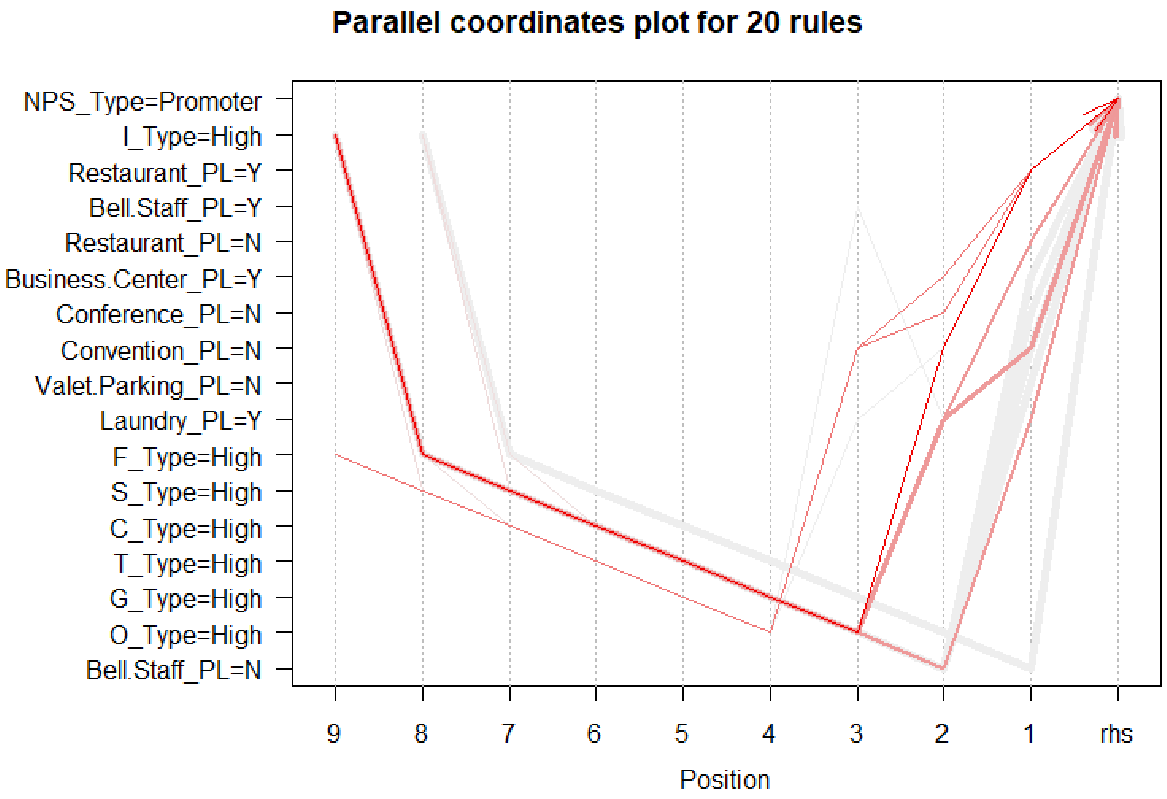
- O_Type (Categorization of Overall Satisfaction) being medium, Business customers are generally detractors.
- If there are no Conference Centers nearby, Business customers are likely to be detractors.
- Even though there were restaurants and laundry service and a business center at the hotel, business users are likely to be detractors.

  This means possibly that these services are not up to customer expectations - hence they were detractors.

● Purpose of Visit = Business Type and NPS type = Promoter

This following are graphs for Business Promoters:



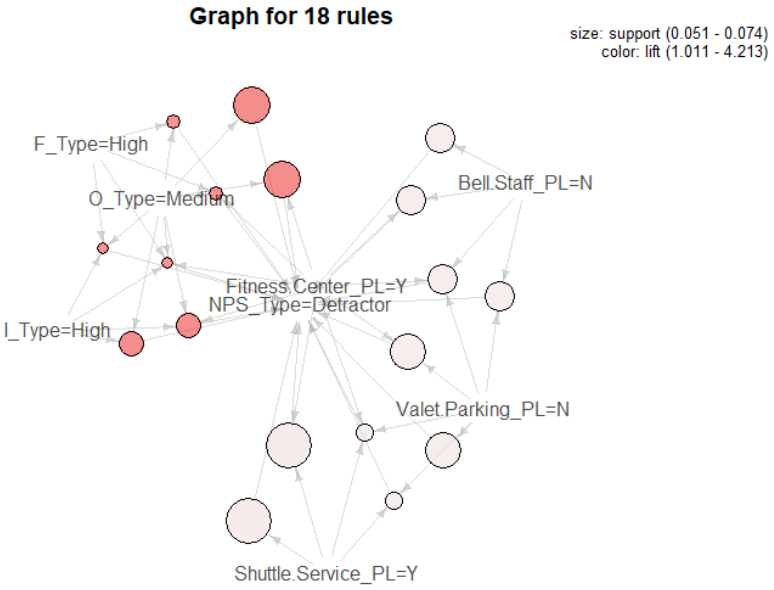


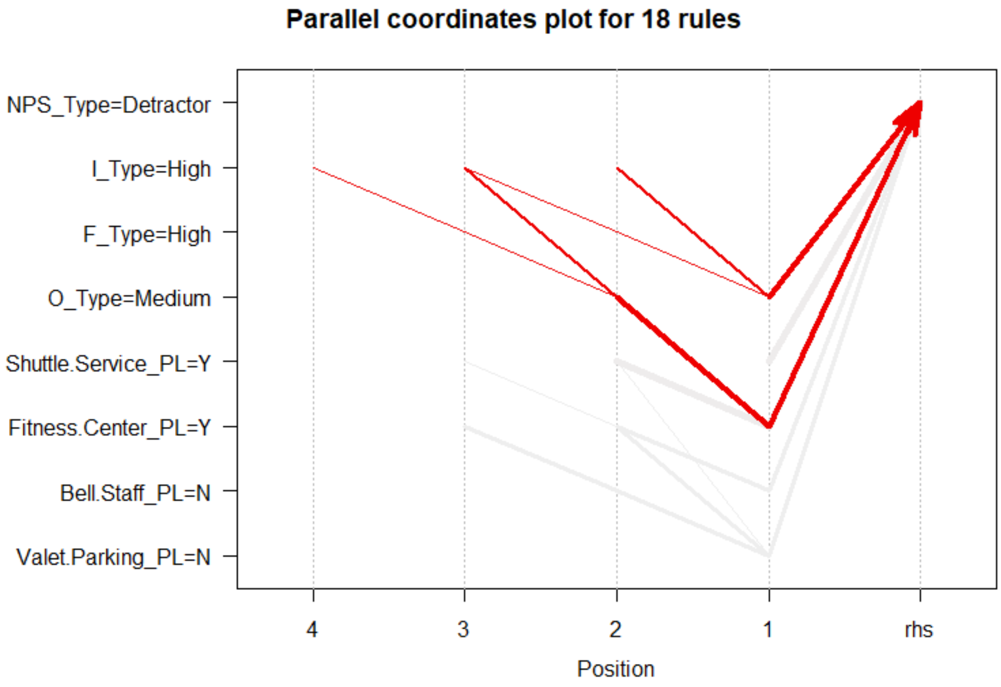
Inferences from the above the plotted rules:

- I_Type(Check In Process), F_Type(Staff Cared), S_Type(Customer Service), C_Type(Condition of Hotel), T_Type(Tranquility), G_Type(Guest Room), O_Type(Overall Satisfaction) being high implies that a business user is likely a promoter.
- Having Bell Staff or restaurant doesn't seem to affect NPS type of a business customer.
- Having no Convention center likely means that the business user is a promoter - showing that Business users might not really care for Convention centers.

- **Purpose of Visit = Leisure Type**
  - **Purpose of Visit = Leisure Type and NPS type = Detractor**

The following are graphs for Leisure detractors:

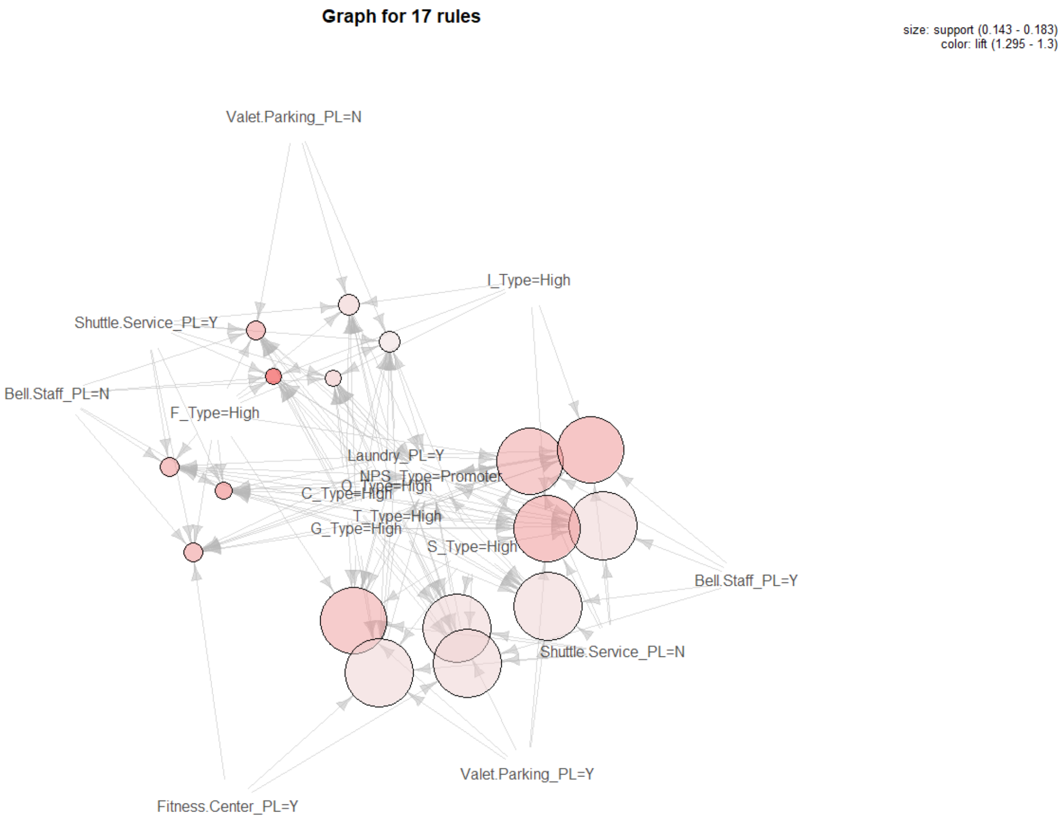


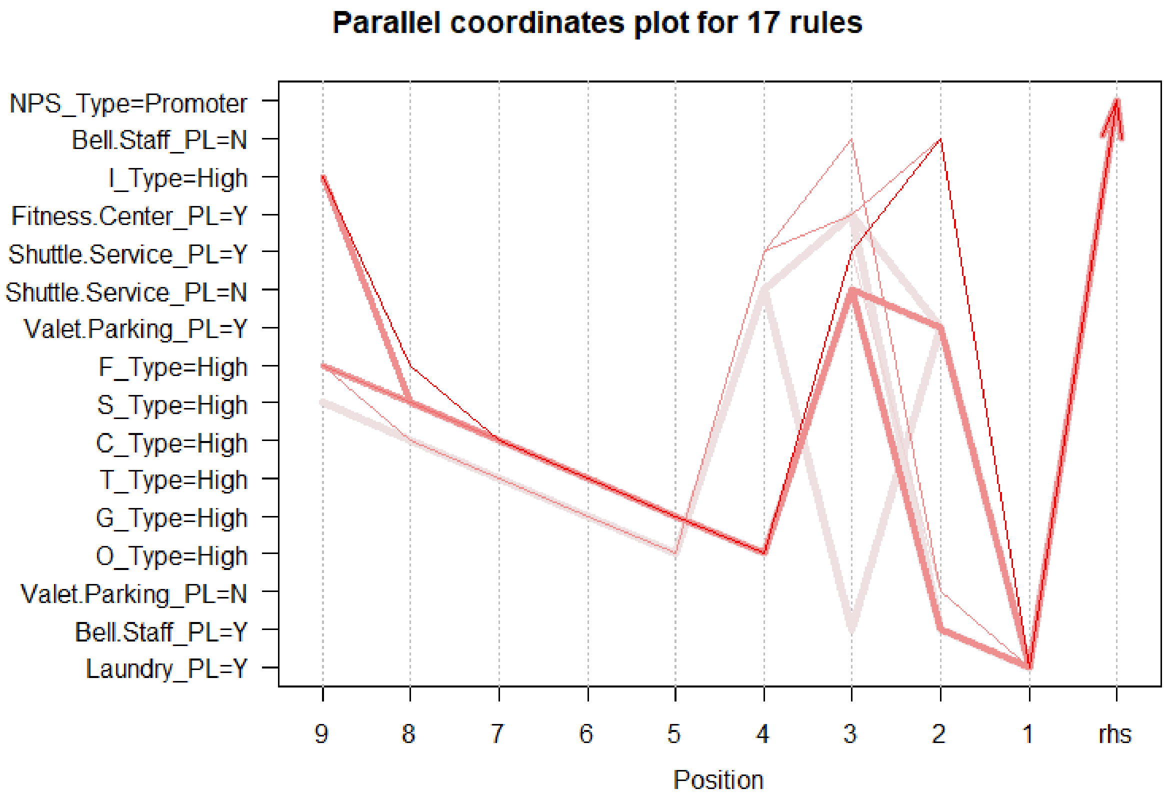


Inferences from the above plotted rules:

- No bell staff and no Valet parking likely means that the leisure customer is a detractor.
- Medium O_Type (Overall Satisfaction) means that the leisure customer is likely a detractor.
- Despite having a Shuttle service and Fitness center, leisure customers tend to be detractors.
- This may be due to poor quality of service or equipment at these places.

- **Purpose of Visit = Leisure Type and NPS type = Promoter**
- The following are graphs for Leisure promoters:





Inferences from the above plotted rules:

- I_Type(Check In Process), F_Type(Staff Cared),S_Type(Customer Service),C_Type(Condition of Hotel),T_Type(Tranquility),G_Type(Guest Room),O_Types (Overall Satisfaction) being high implies that a leisure user is likely a promoter.
- Having laundry generally means that a leisure user is a promoter.

Overall Inference from the above plotted rules:

1) For factors appear in both plots – promoter and detractor, it can be concluded that we cannot determine NPS type using these – like I_Type being high for Leisure.

2) Similarly, for variables which have both options appearing in a single plot, we cannot conclusively state that they help determine NPS Type – like Valet Parking for Leisure Promoter.

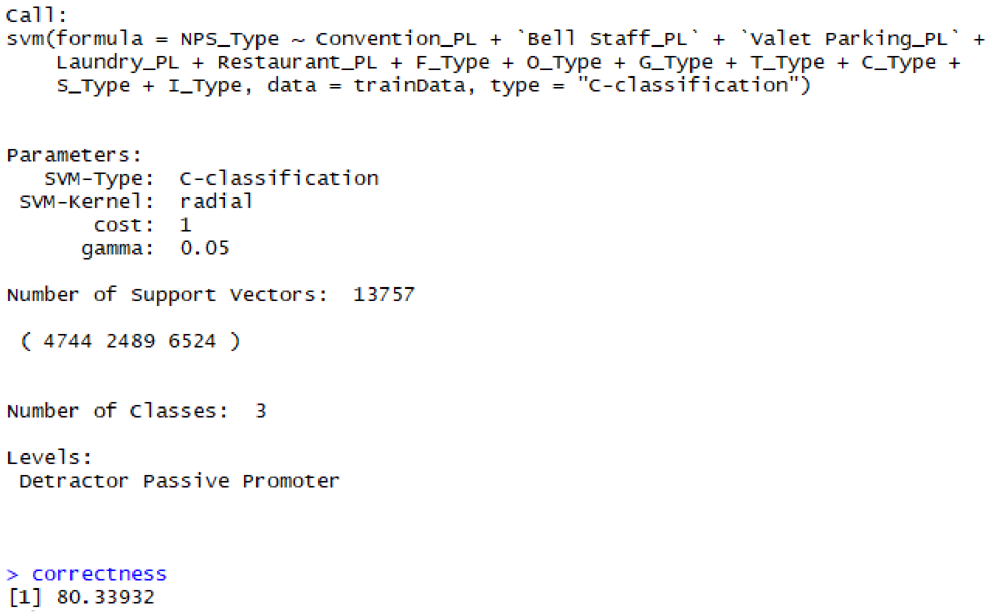
➢ Analysis of Variables using Support Vector Machine (SVM)

We have performed SVM separately for both purposes of visit - Business and Leisure.

We have processed the data the same way as was done in aRules – we have categorical variables corresponding to numeric fields (survey options) and we've split the data into Leisure and Business.

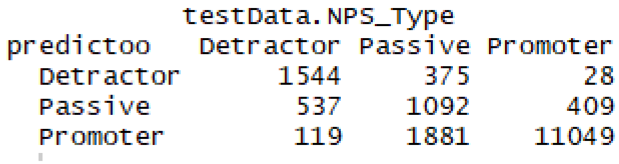
● Purpose of Visit = Business Type

Below is the result for performing SVM to predict NPS Type from other factors for a Business customer.



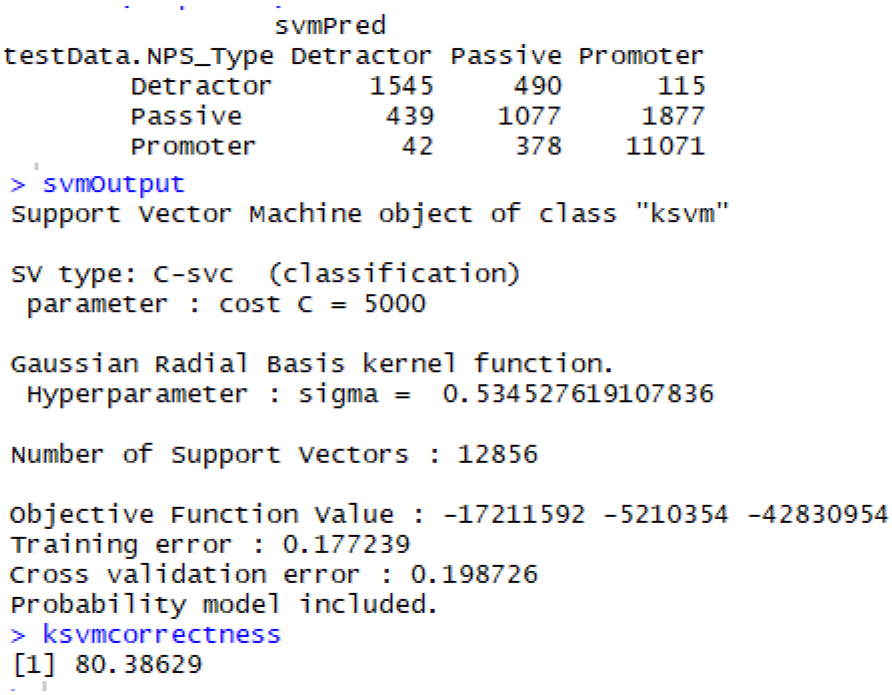
```
Call:
svm(formula = NPS_Type ~ Convention_PL + `Bell Staff_PL` + `Valet Parking_PL` +
    Laundry_PL + Restaurant_PL + F_Type + O_Type + G_Type + T_Type + C_Type +
    S_Type + I_Type, data = trainData, type = "C-classification")


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.05

Number of Support Vectors:  13757

 ( 4744 2489 6524 )


Number of Classes:  3

Levels:
 Detractor Passive Promoter


> correctness
[1] 80.33932
```

Below is the distribution of predicted NPS type and actual NPS type – the diagonal represents the correct predictions and the others represent incorrect ones.



```
             testData.NPS_Type
predictoo    Detractor Passive Promoter
  Detractor       1544     375       28
  Passive          537    1092      409
  Promoter         119    1881    11049
```

We also performed KSVM, with below training and cross-validation errors, and had similar results of over 80%:



```
                  svmPred
testData.NPS_Type Detractor Passive Promoter
        Detractor      1545     490      115
        Passive         439    1077     1877
        Promoter         42     378    11071
> svmOutput
Support Vector Machine object of class "ksvm" 

SV type: C-svc  (classification) 
 parameter : cost C = 5000 

Gaussian Radial Basis kernel function. 
 Hyperparameter : sigma =  0.534527619107836 

Number of Support Vectors : 12856 

Objective Function Value : -17211592 -5210354 -42830954 
Training error : 0.177239 
Cross validation error : 0.198726 
Probability model included. 
> ksvmcorrectness
[1] 80.38629
```

Inferences:

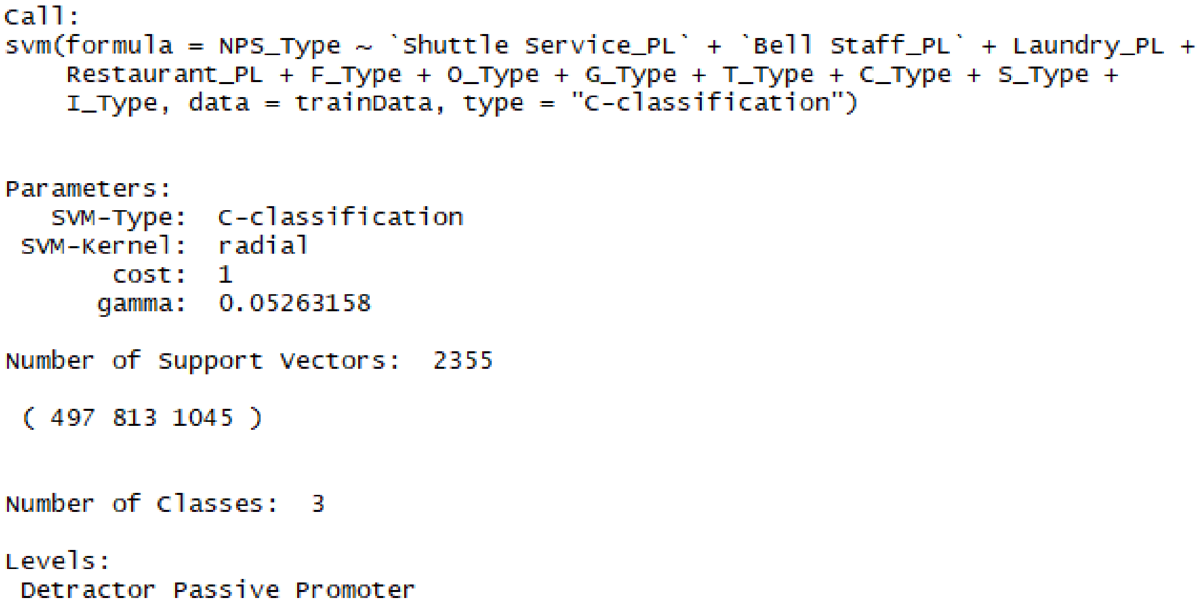
- Correctness of **80.33** indicates that the fields chosen for Business i.e.
    - Convention center
    - Bell Staff
    - Valet Parking
    - Laundry

  - I_Type(Check In Process)
  - F_Type(Staff Cared)
  - S_Type(Customer Service)
  - C_Type(Condition of Hotel)
  - T_Type(Tranquility)
  - G_Type(Guest Room)
  - O_Type(Overall Satisfaction)

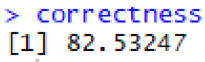
Helped predict NPS_Type of a customer with an accuracy of **80.33%,** which is very good.

- ### Purpose of Visit = Leisure Type

Below is the result for performing SVM to predict NPS Type from other factors for a Leisure customer.

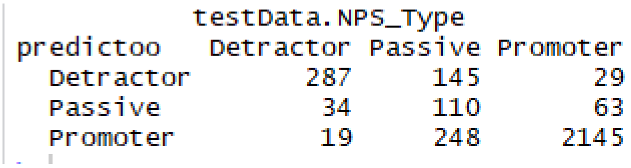


```
Call:
svm(formula = NPS_Type ~ `Shuttle Service_PL` + `Bell Staff_PL` + Laundry_PL +
    Restaurant_PL + F_Type + O_Type + G_Type + T_Type + C_Type + S_Type +
    I_Type, data = trainData, type = "C-classification")


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.05263158

Number of Support Vectors:  2355

 ( 497 813 1045 )


Number of Classes:  3

Levels:
 Detractor Passive Promoter
```



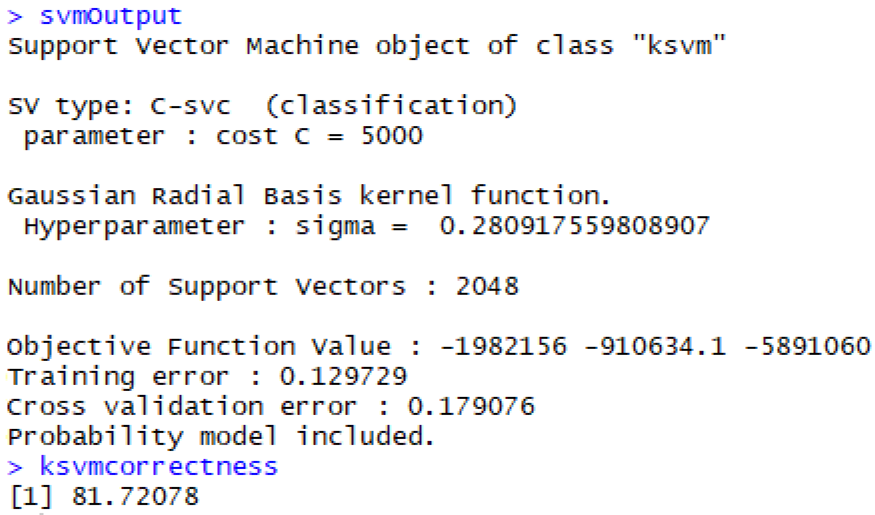
```
> correctness
[1] 82.53247
```

Below is the coefficient matrix:

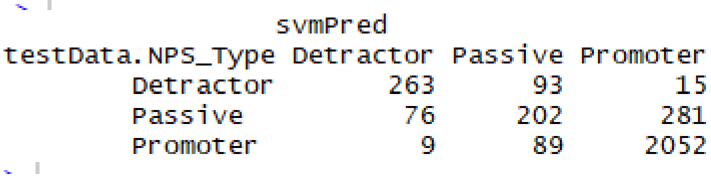


```
            testData.NPS_Type
predictoo   Detractor Passive Promoter
  Detractor       287     145       29
  Passive          34     110       63
  Promoter         19     248     2145
```

Again, we performed KSVM and obtained similar results:



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

SV type: C-svc  (classification)
 parameter : cost C = 5000

Gaussian Radial Basis kernel function.
 Hyperparameter : sigma =  0.280917559808907

Number of Support Vectors : 2048

Objective Function Value : -1982156 -910634.1 -5891060
Training error : 0.129729
Cross validation error : 0.179076
Probability model included.
> ksvmcorrectness
[1] 81.72078
```



```
                 svmPred
testData.NPS_Type Detractor Passive Promoter
        Detractor       263      93       15
        Passive          76     202      281
        Promoter          9      89     2052
```

Inferences:

- Correctness of **82.53** indicates that the fields chosen for Leisure i.e.
    - Shuttle Service
    - Bell Staff
    - Laundry
    - Restaurant
    - I_Type(Check In Process)
    - F_Type(Staff Cared)
    - S_Type(Customer Service)
    - C_Type(Condition of Hotel)

- T_Type(Tranquility)
- G_Type(Guest Room)
- O_Type(Overall Satisfaction)

have helped predict NPS type of a customer with an accuracy of **82.53**, which is, again, very good.

➢Validation

Choice of data

We decided to choose a region which has low NPS and high detractor count as even a small change in NPS in such a region could lead to large profits, due to large number of detractors shifting to promoters. We narrowed down on Texas in USA by plotting the number of detractors, NPS per country for USA, number of detractors per state and NPS per state for each state in the USA.

Survey Factors

We generated a heatmap on the survey factors, and ended up with the conclusion that many fields like Guest Room quality, Tranquility, Customer Service and Check-In Quality show a distinct spread across different NPS types. By running linear modelling on Likelihood to Recommend, and getting an “accuracy” of 83.05%, we could validate that these variables certainly count towards NPS type.

Amenities

We ran aRules to determine whether any amenities (or categorizations of survey characteristic) had a role to play in determining NPS type of a customer. We got some data here to support our assumption, but some data that doesn’t as well. Some factors like O_Type (Overall Satisfaction) showed clear signs that if it is Medium, customer is a detractor, while some variables like I_Type remains high regardless of a leisure user being a Promoter or Detractor.

SVM was more conclusive – performing it based on certain amenities like Shuttle service for leisure and Conference Center for business showed that those can indeed predict NPS type with a high precision of above 80% in each case, reinforcing those same facts from aRules.

➢Recommendations/Actionable insight

Based on the information gained from analyzing the data, we can recommend the following for increasing NPS of hotels in Texas, USA:

- Looking at the heatmap and linear modelling, Guest Room and Customer Service had a big role in determining Likelihood to recommend, and had the lowest mean value among detractors. Working on improving customer service and the condition of the guest rooms is a good idea to improve NPS.
- Looking at Business Visitors from aRules and SVM, we have determined that having Conference centers near the hotel is imperative for keeping them from being detractors. So, we suggest setting up conference centers in hotels.

  Business users who are exposed to Business Center in a hotel tend to be detractors – this implies that they are dissatisfied with them. So, we suggest improving the service in Business Centers across hotels in Texas.
- As for leisure customers, Laundry is an important factor for them being promoters – so we suggest hotels concentrate on setting up and maintaining this.

  Leisure customers also tend to react negatively with fitness center and the shuttle service – so we suggest improving these services to increase NPS.
- We've also concluded that the quality of the check in process, staff caring, customer service, condition of the hotel and guest room are all important factors for both business or leisure customers to be promoters – so we suggest keeping them high across all hotels.

R-Code

```
library("ggplot2")

library("maps")

library('ggmap')

library("ggtern")

library("gdata")

library("data.table")

#Specifying which columns to read

coltoread<-
c("POV_CODE_C","State_PL","Country_PL","NPS_Type","Likelihood_Recommend_H","Overall_Sat_H","Guest_Roo
m_H","Tranquility_H","Condition_Hotel_H","Customer_SVC_H","Staff_Cared_H","Internet_Sat_H","Check_In_H","
F&B_FREQ_H","F&B_Overall_Experience_H","Business
Center_PL","Conference_PL","Laundry_PL","Restaurant_PL","Valet Parking_PL","Bell Staff_PL","Fitness
Center_PL","Shuttle Service_PL")


#Reading feb data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data
Science/Project/Dataset/out-201402.csv", header=TRUE, select=coltoread, verbose=TRUE)

feb<- datta


#Reading mar data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data
Science/Project/Dataset/out-201403.csv", header=TRUE, select=coltoread, verbose=TRUE)

mar<- datta


#Reading apr data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data
Science/Project/Dataset/out-201404.csv", header=TRUE, select=coltoread, verbose=TRUE)

apr<- datta


#Reading may data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data
Science/Project/Dataset/out-201405.csv", header=TRUE, select=coltoread, verbose=TRUE)
```

```
may<-datta


#Reading jun data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data Science/Project/Dataset/out-201406.csv", header=TRUE, select=coltoread, verbose=TRUE)

jun<- datta


#Reading jul data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data Science/Project/Dataset/out-201407.csv", header=TRUE, select=coltoread, verbose=TRUE)

jul<- datta


#Reading aug data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data Science/Project/Dataset/out-201408.csv", header=TRUE, select=coltoread, verbose=TRUE)

aug<-datta


#Reading sep data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data Science/Project/Dataset/out-201409.csv", header=TRUE, select=coltoread, verbose=TRUE)

sep<- datta


#Reading oct data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data Science/Project/Dataset/out-201410.csv", header=TRUE, select=coltoread, verbose=TRUE)

oct<- datta


#Reading mov data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data Science/Project/Dataset/out-201411.csv", header=TRUE, select=coltoread, verbose=TRUE)

nov<- datta


#Reading dec data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data Science/Project/Dataset/out-201412.csv", header=TRUE, select=coltoread, verbose=TRUE)
```

```
dec<-datta


#Reading jan data

datta <- fread(file="D:/Rahul/Documents/Syracuse University/1st Semester/IST 687 - Applied Data Science/Project/Dataset/out-201501.csv", header=TRUE, select=coltoread, verbose=TRUE)

jan<- datta


#Combining to one dataset

fulldataset<-rbind(feb,mar,apr,may,jun,jul,aug,sep,oct,nov,dec,jan)


#Removing rows with blank LTR, NPS type and Country

fulldataset <- fulldataset[fulldataset$Likelihood_Recommend_H!="",]

fulldataset <- fulldataset[fulldataset$NPS_Type!="",]

fulldataset <- fulldataset[fulldataset$Country_PL!="",]


#Function to count how many detractors

detcountfn <- function(x)

{

  return(length(which(x=="Detractor")))

}


#Function to count how many promoters

promcountfn<-function(x)

{

  return(length(which(x=="Promoter")))

}


#Function to count how many passives

passcountfn <- function(x)

{

  return(length(which(x=="Passive")))

}
```

```
#Detractor/Promoter/Passive numbers per country
detpercountry<-data.frame(tapply(fulldataset$NPS_Type,fulldataset$Country_PL , detcountfn))
prompercountry<-data.frame(tapply(fulldataset$NPS_Type,fulldataset$Country_PL , promcountfn))
passpercountry<-data.frame(tapply(fulldataset$NPS_Type,fulldataset$Country_PL , passcountfn))

#Declaring nps for calculating NPS per country
nps<-NA
nps$Country<-data.frame(rownames(detpercountry))
nps$Detractor<-detpercountry$tapply.fulldataset.NPS_Type..fulldataset.Country_PL..detcountfn.
nps$Promoter<-prompercountry$tapply.fulldataset.NPS_Type..fulldataset.Country_PL..promcountfn.
nps$Passive<-passpercountry$tapply.fulldataset.NPS_Type..fulldataset.Country_PL..passcountfn.
nps$nps<NA

#Calculate NPS for each country
for (i in 1:length(nps$Promoter)) {
  total<-nps$Passive[i]+nps$Detractor[i]+nps$Promoter[i]
  prom <- nps$Promoter[i]/total
  det<-nps$Detractor[i]/total
  nps$nps[i] <-(prom-det)*100
}

#Making dataframe to display data in barplot and sorting
NPSCountry<-NA
NPSCountry$Country<-rownames(detpercountry)
NPSCountry$NPS<-nps$nps
NPSCountry<-data.frame(NPSCountry)
NPSCountry<-NPSCountry [order(NPSCountry$NPS),]

#Average Likelihood to recommend per country
avgltr <- data.frame(tapply(fulldataset$Likelihood_Recommend_H,fulldataset$Country_PL , mean))
#Barplot of the same
```

```
barplot(sort(avgltr$tapply.fulldataset.Likelihood_Recommend_H..fulldataset.Country_PL..),cex.names=0.6,las=2,ylim = c(0,10),main=" Average LTR per country")

#Inference

#Jamaica has the lowest average LTR per country at 6.22

#Poland has the highest average LTR per country at 9.64

#US is still in the bottom one third at 8.64


#Plot showing NPS across different  with our dataset

barplot(NPSCountry$NPS,names.arg = NPSCountry$Country,cex.names=0.6,las=2,ylim = c(-40,100),main="Average NPS per country")

#Inference

#Jamaica has the lowest NPS at -16%

#Poland has the highest NPS at 91%.

#US has NPS of 57.36%, it's in the bottom half.


#Finding detractor count per country and sorting

countdet<-data.frame(tapply(fulldataset$NPS_Type,fulldataset$Country_PL , detcountfn))

countdet$region<-rownames(countdet)

countdet<-countdet[order(countdet$tapply.fulldataset.NPS_Type..fulldataset.Country_PL..detcountfn.),]


#Detractor count per country barchart

barplot(countdet$tapply.fulldataset.NPS_Type..fulldataset.Country_PL..detcountfn.,cex.names=0.6,las=2,main="Number of Detractors per country")

#Inference

#USA has the most number of detractors by a longshot at 89565.

#So we decided to chose USA for our analysis.


#Getting US data

usdata<-fulldataset[fulldataset$Country_PL=="United States",]


#Statistics per state

detperstate<-data.frame(tapply(usdata$NPS_Type,usdata$State_PL , detcountfn))

promperstate<-data.frame(tapply(usdata$NPS_Type,usdata$State_PL , promcountfn))
```

```
passperstate<-data.frame(tapply(usdata$NPS_Type,usdata$State_PL , passcountfn))


#npsstate declared for calculating nps per state
npsstate<-NA
npsstate$State<-data.frame(rownames(detperstate))
npsstate$Detractor<-detperstate$tapply.usdata.NPS_Type..usdata.State_PL..detcountfn.
npsstate$Promoter<-prompercountry$tapply.fulldataset.NPS_Type..fulldataset.Country_PL..promcountfn.
npsstate$Passive<-passperstate$tapply.usdata.NPS_Type..usdata.State_PL..passcountfn.
npsstate$nps<NA


#Calculating NPS per state
for (i in 1:length(npsstate$Promoter)) {
  total<-npsstate$Passive[i]+npsstate$Detractor[i]+npsstate$Promoter[i]
  prom <- npsstate$Promoter[i]/total
  det<-npsstate$Detractor[i]/total
  npsstate$nps[i] <-(prom-det)*100
}



#New dataframe for NPS per state for displaying in map
NPState<-data.frame(state=rownames(detperstate),nps=npsstate$nps[1:44])
NPState$state<-tolower(NPState$state)


#Displaying a map of the US with average NPS per state
us <- map_data("state")
npsmapus <- ggplot(NPState, aes(map_id = state))
npsmapus <- npsmapus +     geom_map(map = us, color="black",aes(fill=nps))
npsmapus <- npsmapus +      expand_limits(x = us$long, y = us$lat)
npsmapus <- npsmapus +      coord_map() + ggtitle("NPS per state") + scale_fill_gradient(low = "red", high = "yellow")
npsmapus
#Inference
```

```
#Iowa has the highest NPS of 93.9%.

#Texas has the least NPS at -38.5%.


#Detractor count per state

countdetus<-data.frame(tapply(fulldataset$NPS_Type,fulldataset$State_PL,detcountfn))

colnames(countdetus)[1] <- "DetractorCount"

countdetus$state<-tolower(rownames(countdetus))


#Number of detractors by state

detractorcountmapus <- ggplot(countdetus, aes(map_id = state))

detractorcountmapus <- detractorcountmapus +     geom_map(map = us, color="black",aes(fill=DetractorCount))

detractorcountmapus <- detractorcountmapus +      expand_limits(x = us$long, y = us$lat)

detractorcountmapus <- detractorcountmapus +      coord_map() + ggtitle("Number of detractors per state") + scale_fill_gradient(low = "yellow", high = "red")

detractorcountmapus

#Inference from map

#California has the highest number of detractors at 15619.

#Texas was 2nd at 9905

#Arkansas had the lease at 45.


#Due to a combination of lowest NPS and high detractor count, We decided to choose Texas for our analysis


#Choosing Texas

texdata<-usdata[usdata$State_PL=="Texas",]




#For below numeric fields, we replace NAs with the mean - we truncate precision as we don't want to deal with decimals.

#We do not consider the other fields like Internet and F&B Frequency have too many nulls, so can't substitute with NAs.

texdata[is.na(texdata$Overall_Sat_H)]$Overall_Sat_H<-mean(texdata$Overall_Sat_H, na.rm = TRUE)

texdata[is.na(texdata$Guest_Room_H)]$Guest_Room_H<-mean(texdata$Guest_Room_H, na.rm = TRUE)

texdata[is.na(texdata$Tranquility_H)]$Tranquility_H<-mean(texdata$Tranquility_H, na.rm = TRUE)
```

```
texdata[is.na(texdata$Condition_Hotel_H)]$Condition_Hotel_H<-mean(texdata$Condition_Hotel_H, na.rm = TRUE)
texdata[is.na(texdata$Customer_SVC_H)]$Customer_SVC_H<-mean(texdata$Customer_SVC_H, na.rm = TRUE)
texdata[is.na(texdata$Staff_Cared_H)]$Staff_Cared_H<-mean(texdata$Staff_Cared_H, na.rm = TRUE)
texdata[is.na(texdata$Check_In_H)]$Check_In_H<-mean(texdata$Check_In_H, na.rm = TRUE)


#Removing rows with No data for the following flag variables.
newtexdata<-texdata[texdata$`Valet Parking_PL`!="",]
newtexdata<-newtexdata[newtexdata$`Business Center_PL`!="",]
newtexdata<-newtexdata[newtexdata$Conference_PL!="",]
newtexdata<-newtexdata[newtexdata$Laundry_PL!="",]
newtexdata<-newtexdata[newtexdata$Restaurant_PL!="",]
newtexdata<-newtexdata[newtexdata$`Bell Staff_PL`!="",]
newtexdata<-newtexdata[newtexdata$`Fitness Center_PL`!="",]
newtexdata<-newtexdata[newtexdata$`Shuttle Service_PL`!="",]
newtexdata<-newtexdata[newtexdata$POV_CODE_C!="",]


#Displaying customers by Purpose of visit in Texas.


gg <- ggplot(data = newtexdata, aes(x=POV_CODE_C, fill = NPS_Type)) + geom_bar(position = "dodge") + ggtitle("Distribution customers with different purposes of visit in Texas")
gg <- gg+ xlab("Purpose of Visit")+ylab("Number of customers")
gg
#Inferences
#There are far more business customers than leisure (51,101 compared to  9239).
#The distribution of promoters, passives and detractors is similar for both business and leisure customers.



#Gives the average on each flag based on NPS type and binds it to a vector
Overall_Satisfaction<-tapply(newtexdata$Overall_Sat_H, newtexdata$NPS_Type, mean)
Guest_Room<-tapply(newtexdata$Guest_Room_H, newtexdata$NPS_Type, mean)
Tranquility<-tapply(newtexdata$Tranquility_H, newtexdata$NPS_Type, mean)
HotelCondition<-tapply(newtexdata$Condition_Hotel_H, newtexdata$NPS_Type, mean)
```

```
CustomerService<-tapply(newtexdata$Customer_SVC_H, newtexdata$NPS_Type, mean)

StaffCaring<-tapply(newtexdata$Staff_Cared_H, newtexdata$NPS_Type, mean)

CheckInQuality<-tapply(newtexdata$Check_In_H, newtexdata$NPS_Type, mean)

d<-rbind(Overall_Satisfaction,Guest_Room,Tranquility,HotelCondition,CustomerService,StaffCaring,CheckInQuality)


#Dataframe for heatmap

heatdf<-NA

heatdf$SurveyOptions <-c("Overall_Satisfaction","Overall_Satisfaction","Overall_Satisfaction","Guest_Room","Guest_Room","Guest_Room","Tranquility","Tranquility","Tranquility","HotelCondition","HotelCondition","HotelCondition","CustomerService", "CustomerService","CustomerService","StaffCaring","StaffCaring","StaffCaring","CheckInQuality","CheckInQuality", "CheckInQuality")

heatdf$NPSType <-c("Detractor","Passive","Promoter","Detractor","Passive","Promoter","Detractor","Passive","Promoter","Detractor","Passive","Promoter","Detractor","Passive","Promoter","Detractor","Passive","Promoter","Detractor","Passive", "Promoter")

heatdf$MeanValue <-c(d[1,1],d[1,2],d[1,3],d[2,1],d[2,2],d[2,3],d[3,1],d[3,2],d[3,3],d[4,1],d[4,2],d[4,3],d[5,1],d[5,2],d[5,3],d[6,1],d[6,2],d[6,3],d[7,1],d[7,2],d[7,3])

heatdf <- data.frame(heatdf)


#Heatmap for the selected variables

texheatmap <- ggplot(heatdf, aes(NPSType,SurveyOptions)) + geom_tile(aes(fill = MeanValue))+ggtitle("Average values of Survey Options based on NPS Type")

texheatmap

#Inferences

#Overall satisfaction is the most consistent metric to show satisfaction among promoters (9.51), detractors (5.08) and passive (7.7).

#Tranquility has the lowest score at 8.77, followed by Check In Quality at 9.36 among promoters, compared to other factors.

#Guest Room at 7.91 and Tranquility at 7.93 have the lowest average rating for passives, apart from Overall Satisfaction.

#Guestroom (5.93) and Customer Service (6.69) had the lowest mean value among detractors, apart from Overall Satisfaction.


#Linear model based on selected fields
```

```
model <-
lm(formula=Likelihood_Recommend_H~Overall_Sat_H+Guest_Room_H+Tranquility_H+Condition_Hotel_H+Customer_SVC_H+Staff_Cared_H+Check_In_H,data=texdata)

summary(model)

#Inference

#Excellent R squared value of 0.8305 indicates that the selected fields can determine Likelihood to Recommend well.

#Low p values show that they are all statistically significant as well.


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

#aRules and SVM

library("ggplot2")

library("maps")

library('ggmap')

library("ggtern")

library("gdata")

library("data.table")

library("e1071")

library("kernlab")

library("arules")

library("arulesViz")



#Read data

coltoread<-
c("Overall_Sat_H","Guest_Room_H","Tranquility_H","Condition_Hotel_H","Customer_SVC_H","Staff_Cared_H","Check_In_H", "Likelihood_Recommend_H","City_PL","State_PL","NPS_Type", "Country_PL", "POV_CODE_C", "Restaurant_PL", "Business Center_PL", "Conference_PL","Convention_PL", "Laundry_PL", "Valet Parking_PL", "Bell Staff_PL", "Fitness Center_PL", "Shuttle Service_PL")


#Reading feb data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201402.csv", header=TRUE, select=coltoread, verbose=TRUE)

feb<- datta
```

```
#Reading mar data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201403.csv", header=TRUE,
select=coltoread, verbose=TRUE)

mar<- datta


#Reading apr data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201404.csv", header=TRUE,
select=coltoread, verbose=TRUE)

apr<- datta


#Reading may data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201405.csv", header=TRUE,
select=coltoread, verbose=TRUE)

may<-datta


#Reading jun data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201406.csv", header=TRUE,
select=coltoread, verbose=TRUE)

jun<- datta


#Reading jul data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201407.csv", header=TRUE,
select=coltoread, verbose=TRUE)

jul<- datta


#Reading aug data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201408.csv", header=TRUE,
select=coltoread, verbose=TRUE)

aug<-datta


#Reading sep data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201409.csv", header=TRUE,
select=coltoread, verbose=TRUE)

sep<- datta
```

```
#Reading oct data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201410.csv", header=TRUE, select=coltoread, verbose=TRUE)

oct<- datta


#Reading mov data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201411.csv", header=TRUE, select=coltoread, verbose=TRUE)

nov<- datta


#Reading dec data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201412.csv", header=TRUE, select=coltoread, verbose=TRUE)

dec<-datta


#Reading jan data

datta <- fread(file="//hd.ad.syr.edu/03/bed1ee/Documents/Downloads/Dataset/out-201501.csv", header=TRUE, select=coltoread, verbose=TRUE)

jan<- datta


datta<-rbind(feb,mar,apr,may,jun,jul,aug,sep,oct,nov,dec,jan)


#Remove blank NPS types and filter with US country and Texas state

newnewdatta <- datta[datta$NPS_Type!="",]

newnewdatta <- newnewdatta[newnewdatta$Country_PL == "United States", ]

newnewdatta <- newnewdatta[newnewdatta$State_PL =="Texas", ]


#Removing blanks for some variables, not doing for others as they are not being used for this particular analysis

newnewdatta<-newnewdatta[newnewdatta$Convention_PL!="",]

newnewdatta<-newnewdatta[newnewdatta$Laundry_PL!="",]

newnewdatta<-newnewdatta[newnewdatta$`Valet Parking_PL`!="",]

newnewdatta<-newnewdatta[newnewdatta$`Bell Staff_PL`!="",]

newnewdatta<-newnewdatta[newnewdatta$Restaurant_PL!="",]

newnewdatta<-newnewdatta[newnewdatta$`Shuttle Service_PL`!="",]
```

```
#We make a new column which maps numerical variables to categorical - 8 and up to High, 5-7 as Medium and below 4 as Low

#We want to use these columns for further analysis - aRules and SVM

#Substituting NA with mean

newnewdatta[is.na(newnewdatta$Overall_Sat_H)]$Overall_Sat_H <- mean(newnewdatta$Overall_Sat_H, na.rm = TRUE)

newnewdatta$O_Type <- NA


#Changing to categorical

for (i in 1:nrow(newnewdatta)){

  if(newnewdatta$Overall_Sat_H[i] == 1){

    newnewdatta$O_Type[i] = "Low"


  }

  else if(newnewdatta$Overall_Sat_H[i] == 2) {

    newnewdatta$O_Type[i] = "Low"


  }


  else if(newnewdatta$Overall_Sat_H[i] == 3) {

    newnewdatta$O_Type[i] = "Low"


  }


  else if(newnewdatta$Overall_Sat_H[i] == 4) {

    newnewdatta$O_Type[i] = "Low"


  }


  else if(newnewdatta$Overall_Sat_H[i] == 5) {

    newnewdatta$O_Type[i] = "Medium"
```

```
 }


 else if(newnewdatta$Overall_Sat_H[i] == 6) {
   newnewdatta$O_Type[i] = "Medium"


 }


 else if(newnewdatta$Overall_Sat_H[i] == 7) {
   newnewdatta$O_Type[i] = "Medium"


 }
 else
   newnewdatta$O_Type[i] = "High"


}




newnewdatta[is.na(newnewdatta$Guest_Room_H)]$Guest_Room_H <- mean(newnewdatta$Guest_Room_H, na.rm = TRUE)
newnewdatta$G_Type <- NA


for (i in 1:nrow(newnewdatta)){
 if(newnewdatta$Guest_Room_H[i] == 1){
   newnewdatta$G_Type[i] = "Low"


 }
 else if(newnewdatta$Guest_Room_H[i] == 2) {
   newnewdatta$G_Type[i] = "Low"


 }


 else if(newnewdatta$Guest_Room_H[i] == 3) {
   newnewdatta$G_Type[i] = "Low"

 }

 else if(newnewdatta$Guest_Room_H[i] == 4) {
   newnewdatta$G_Type[i] = "Low"

 }

 else if(newnewdatta$Guest_Room_H[i] == 5) {
   newnewdatta$G_Type[i] = "Medium"

 }

 else if(newnewdatta$Guest_Room_H[i] == 6) {
   newnewdatta$G_Type[i] = "Medium"

 }

 else if(newnewdatta$Guest_Room_H[i] == 7) {
   newnewdatta$G_Type[i] = "Medium"

 }
 else
   newnewdatta$G_Type[i] = "High"

}


newnewdatta[is.na(newnewdatta$Tranquility_H)]$Tranquility_H <- mean(newnewdatta$Tranquility_H, na.rm = TRUE)

newnewdatta$T_Type <- NA
```

```
for (i in 1:nrow(newnewdatta)){
 if(newnewdatta$Tranquility_H[i] == 1){
  newnewdatta$T_Type[i] = "Low"

 }
 else if(newnewdatta$Tranquility_H[i] == 2) {
  newnewdatta$T_Type[i] = "Low"

 }

 else if(newnewdatta$Tranquility_H[i] == 3) {
  newnewdatta$T_Type[i] = "Low"

 }

 else if(newnewdatta$Tranquility_H[i] == 4) {
  newnewdatta$T_Type[i] = "Low"

 }

 else if(newnewdatta$Tranquility_H[i] == 5) {
  newnewdatta$T_Type[i] = "Medium"

 }

 else if(newnewdatta$Tranquility_H[i] == 6) {
  newnewdatta$T_Type[i] = "Medium"

 }

 else if(newnewdatta$Tranquility_H[i] == 7) {
    newnewdatta$T_Type[i] = "Medium"


  }
  else
    newnewdatta$T_Type[i] = "High"


}


newnewdatta[is.na(newnewdatta$Condition_Hotel_H)]$Condition_Hotel_H <-
mean(newnewdatta$Condition_Hotel_H, na.rm = TRUE)
newnewdatta$C_Type <- NA


for (i in 1:nrow(newnewdatta)){
  if(newnewdatta$Condition_Hotel_H[i] == 1){
    newnewdatta$C_Type[i] = "Low"


  }
  else if(newnewdatta$Condition_Hotel_H[i] == 2) {
    newnewdatta$C_Type[i] = "Low"


  }


  else if(newnewdatta$Condition_Hotel_H[i] == 3) {
    newnewdatta$C_Type[i] = "Low"


  }


  else if(newnewdatta$Condition_Hotel_H[i] == 4) {
    newnewdatta$C_Type[i] = "Low"


  }
  else if(newnewdatta$Condition_Hotel_H[i] == 5) {
    newnewdatta$C_Type[i] = "Medium"

  }

  else if(newnewdatta$Condition_Hotel_H[i] == 6) {
    newnewdatta$C_Type[i] = "Medium"

  }

  else if(newnewdatta$Condition_Hotel_H[i] == 7) {
    newnewdatta$C_Type[i] = "Medium"

  }
  else
    newnewdatta$C_Type[i] = "High"

}



newnewdatta[is.na(newnewdatta$Customer_SVC_H)]$Customer_SVC_H <-
mean(newnewdatta$Customer_SVC_H, na.rm = TRUE)
newnewdatta$S_Type <- NA


for (i in 1:nrow(newnewdatta)){
  if(newnewdatta$Customer_SVC_H[i] == 1){
    newnewdatta$S_Type[i] = "Low"

  }
  else if(newnewdatta$Customer_SVC_H[i] == 2) {
    newnewdatta$S_Type[i] = "Low"
```

```
 }

 else if(newnewdatta$Customer_SVC_H[i] == 3) {
   newnewdatta$S_Type[i] = "Low"

 }

 else if(newnewdatta$Customer_SVC_H[i] == 4) {
   newnewdatta$S_Type[i] = "Low"

 }

 else if(newnewdatta$Customer_SVC_H[i] == 5) {
   newnewdatta$S_Type[i] = "Medium"

 }

 else if(newnewdatta$Customer_SVC_H[i] == 6) {
   newnewdatta$S_Type[i] = "Medium"

 }

 else if(newnewdatta$Customer_SVC_H[i] == 7) {
   newnewdatta$S_Type[i] = "Medium"

 }
 else
   newnewdatta$S_Type[i] = "High"

}
```

```
newnewdatta[is.na(newnewdatta$Staff_Cared_H)]$Staff_Cared_H <- mean(newnewdatta$Staff_Cared_H, na.rm =
TRUE)

newnewdatta$F_Type <- NA


for (i in 1:nrow(newnewdatta)){
 if(newnewdatta$Staff_Cared_H[i] == 1){
  newnewdatta$F_Type[i] = "Low"


 }
 else if(newnewdatta$Staff_Cared_H[i] == 2) {
  newnewdatta$F_Type[i] = "Low"


 }


 else if(newnewdatta$Staff_Cared_H[i] == 3) {
  newnewdatta$F_Type[i] = "Low"


 }


 else if(newnewdatta$Staff_Cared_H[i] == 4) {
  newnewdatta$F_Type[i] = "Low"


 }


 else if(newnewdatta$Staff_Cared_H[i] == 5) {
  newnewdatta$F_Type[i] = "Medium"


 }


 else if(newnewdatta$Staff_Cared_H[i] == 6) {
  newnewdatta$F_Type[i] = "Medium"
  }


  else if(newnewdatta$Staff_Cared_H[i] == 7) {
    newnewdatta$F_Type[i] = "Medium"


  }
  else
    newnewdatta$F_Type[i] = "High"


}


newnewdatta[is.na(newnewdatta$Check_In_H)]$Check_In_H <- mean(newnewdatta$Check_In_H, na.rm = TRUE)
newnewdatta$I_Type <- NA


for (i in 1:nrow(newnewdatta)){
  if(newnewdatta$Check_In_H[i] == 1){
    newnewdatta$I_Type[i] = "Low"


  }
  else if(newnewdatta$Check_In_H[i] == 2) {
    newnewdatta$I_Type[i] = "Low"


  }


  else if(newnewdatta$Check_In_H[i] == 3) {
    newnewdatta$I_Type[i] = "Low"


  }


  else if(newnewdatta$Check_In_H[i] == 4) {
    newnewdatta$I_Type[i] = "Low"
  }


  else if(newnewdatta$Check_In_H[i] == 5) {
    newnewdatta$I_Type[i] = "Medium"


  }


  else if(newnewdatta$Check_In_H[i] == 6) {
    newnewdatta$I_Type[i] = "Medium"


  }


  else if(newnewdatta$Check_In_H[i] == 7) {
    newnewdatta$I_Type[i] = "Medium"


  }
  else
    newnewdatta$I_Type[i] = "High"


}


#Splitting into business and leisure
businessdatta<-newnewdatta[newnewdatta$POV_CODE_C=="BUSINESS",]
leisuredatta<-newnewdatta[newnewdatta$POV_CODE_C=="LEISURE",]


#SVM for Leisure - using some leisure vairables like Shutle Service from leisure dataset
#Creating index for training and test data and creating the same from our data
randIndex <- sample(1:dim(leisuredatta)[1])
cutPoint2_3 <-floor(2*dim(leisuredatta)[1]/3)
length(randIndex)
head(randIndex)
cutPoint2_3
```

```
trainData <- leisuredatta[randIndex[1:cutPoint2_3],] #Creating training dataset
testData <- leisuredatta[randIndex[(cutPoint2_3+1):dim(leisuredatta)[1]],] #creating testing dataset

#Performing svm using below variables and predicting using test dataset
svmo <- svm(NPS_Type~`Shuttle Service_PL`+`Bell Staff_PL`+Laundry_PL+Restaurant_PL+F_Type+O_Type+G_Type+T_Type+C_Type+S_Type+I_Type, data = trainData, type = "C-classification")
summary(svmo)
predicto <- predict(svmo, testData)
resulto <- table(predicto, testData$NPS_Type)
#Saving into another variable for confusion matrix
predictoo<-predicto
#Finding the count of correct and incorrectly predicted variables.
predicto<-data.frame(predicto)
predicto$Correct<-NA
for(i in 1:length(testData$NPS_Type))
{
  predicto$Correct[i]<-predicto$predicto[i]==testData$NPS_Type[i]
}

#Number of entries that were correcly and incorrectly predicted and calculating percentage correctness
truecount<-length(predicto$predicto[predicto$Correct==TRUE])
falsecount<-length(predicto$predicto[predicto$Correct==FALSE])
correctness<-(truecount*100)/(truecount+falsecount)
correctness

compTable<-data.frame(predictoo,testData$NPS_Type)
table(compTable)
#This shows the confusion matrix - however, we have chosen to calculate out value as above - this is just for visalization



#Inference
```

```
#Correctness of 82.53 indicates that the fields chosen for Leisure ie.
#`Shuttle Service_PL`+`Bell Staff_PL`+Laundry_PL+Restaurant_PL+F_Type+O_Type+G_Type+T_Type+C_Type+S_Type+I_Type
#helped predict NPS_Type with an accuracy of 82.04 percent.


#Performing KSVM also for the same
svmOutput <- ksvm(NPS_Type ~ `Shuttle Service_PL`+`Bell Staff_PL`+Laundry_PL+Restaurant_PL+F_Type+O_Type+G_Type+T_Type+C_Type+S_Type+I_Type, data=trainData,kernel="rbfdot",kpar="automatic",C=5000,cross=3, prob.model=TRUE)
svmOutput
svmPred<-predict(svmOutput,testData)
compTable<-data.frame(testData$NPS_Type,svmPred)
conf<-table(compTable)

truepositives<-conf[1,1]+conf[2,2]+conf[3,3]
falsepositives<-conf[1,2]+conf[1,3]+conf[2,1]+conf[2,3]+conf[3,1]+conf[3,2]
ksvmcorrectness<-(truepositives*100)/(truepositives+falsepositives)
ksvmcorrectness
#Inferences
#We find similar results as we found in svm - 81% accuracy


#SVM for Business - considering business variables - like Convention, Valet Parking for businessdataset
#Creating index for training and test data and creating the same from our data
randIndex <- sample(1:dim(businessdatta)[1])
cutPoint2_3 <-floor(2*dim(businessdatta)[1]/3)
length(randIndex)
head(randIndex)
cutPoint2_3
trainData <- businessdatta[randIndex[1:cutPoint2_3],] #Creating training dataset
testData <- businessdatta[randIndex[(cutPoint2_3+1):dim(businessdatta)[1]],] #creating testing dataset
```

```
#Performing svm using below variables and predicting using test dataset
svmo <- svm(NPS_Type~Convention_PL+`Bell Staff_PL`+`Valet Parking_PL`+Laundry_PL+Restaurant_PL+F_Type+O_Type+G_Type+T_Type+C_Type+S_Type+I_Type, data = trainData, type = "C-classification")
summary(svmo)
predicto <- predict(svmo, testData)
resulto <- table(predicto, testData$NPS_Type)
#Copying to new variable for confusion matrix
predictoo<-predicto
#Finding the count of correct and incorrectly predicted variables.
predicto<-data.frame(predicto)
predicto$Correct<-NA
for(i in 1:length(testData$NPS_Type))
{
  predicto$Correct[i]<-predicto$predicto[i]==testData$NPS_Type[i]
}


#Number of entries that were correcly and incorrectly predicted and calculating percentage correctness
truecount<-length(predicto$predicto[predicto$Correct==TRUE])
falsecount<-length(predicto$predicto[predicto$Correct==FALSE])
correctness<-(truecount*100)/(truecount+falsecount)
correctness


compTable<-data.frame(predictoo,testData$NPS_Type)
table(compTable)
#This shows the confusion matrix - however, we have chosen to calculate out value as above - this is just for visalization



#Inference
#Correctness of 80.33 indicates that the fields chosen for Business ie.
#Convention_PL+`Bell Staff_PL`+`Valet Parking_PL`+Laundry_PL+Restaurant_PL+F_Type+O_Type+G_Type+T_Type+C_Type+S_Type+I_Type
#helped predict NPS_Type with an accuracy of 80.13 percent.
```

```
#Performing KSVM also for the same

svmOutput <- ksvm(NPS_Type ~ Convention_PL+`Bell Staff_PL`+`Valet
Parking_PL`+Laundry_PL+Restaurant_PL+F_Type+O_Type+G_Type+T_Type+C_Type+S_Type+I_Type,
data=trainData,kernel="rbfdot",kpar="automatic",C=5000,cross=3, prob.model=TRUE)

svmOutput

svmPred<-predict(svmOutput,testData)

compTable<-data.frame(testData$NPS_Type,svmPred)

conf<-table(compTable)


truepositives<-conf[1,1]+conf[2,2]+conf[3,3]

falsepositives<-conf[1,2]+conf[1,3]+conf[2,1]+conf[2,3]+conf[3,1]+conf[3,2]

ksvmcorrectness<-(truepositives*100)/(truepositives+falsepositives)

ksvmcorrectness

#Inferences

#We find similar results as we found in svm - 80% accuracy



#aRules


#Business  aRules

#Removing rows unnecessary for arules

businessreduced<-businessdatta[,-1:-10]

businessreduced<-businessreduced[,-2:-3]

businessreduced<-businessreduced[,-9:-10]


#Make factor for aRule and filter rule based on lift

newdattaf <- data.frame(sapply(businessreduced,as.factor))

rulesa <- apriori(newdattaf,parameter=list(minlen=1,support=0.05,confidence=0.5),appearance =
list(rhs=c("NPS_Type=Detractor"),default="lhs"), control=list(verbose=F))

goodrules<-rulesa[quality(rulesa)$lift>4.1]

summary(goodrules)
```

```
inspect(goodrules)


#Scatterplot and parallel plot Business Detractors

plot(goodrules, method="graph", control=list(type="items"))

plot(goodrules, method="paracoord",control=list(reorder=TRUE))

#Inference

#

#O_Type (Catergorization of Overall Satisfaction) being medium, people are generally detractors, which is logical

#If there are no Conference Centeres nearby, Business customers are likely to be detractors.

#Eventhough there were restaurants and laundry service and a business center at the hotel, business users are likely to be detractors.

#This means possibly that these services are not up to customer expectations - hence they were detractors.


#aRule and filter rule based on lift

rulesa <- apriori(newdattaf,parameter=list(minlen=1,support=0.05,confidence=0.5),appearance = list(rhs=c("NPS_Type=Promoter"),default="lhs"), control=list(verbose=F))

goodrules<-rulesa[quality(rulesa)$lift>1.327]

summary(goodrules)

inspect(goodrules)


#Scatterplot and parallel plot Business Promoter

plot(goodrules, method="graph", control=list(type="items"))

plot(goodrules, method="paracoord", control=list(reorder=TRUE))

#Inferences

#I (Check In Process), F (Staff Cared),S (Customer Service),C (Condition of Hotel),T (Tranquility),G (Guest Room),O_Types (Overall Satisfaction) being high implies that a business user is likely a promoter.

#Having Bell Staff or restaurant  doesn't seem to affect NPS type of a business customer.

#Having no Convention center likely means that the business user is a promoter - showing that Business users might not really care for Convention centers.


#aRules for Leisure


#Removing rows unnecessary for arules

leisurereduced<-leisuredatta[,-1:-10]
```

```
leisurereduced<-leisurereduced[,-2:-3]

leisurereduced<-leisurereduced[,-2:-3]

leisurereduced<-leisurereduced[,-3]

leisurereduced<-leisurereduced[,-2]


#Make factor for aRule and filter rule based on lift

newdattaf <- data.frame(sapply(leisurereduced,as.factor))

rulesa <- apriori(newdattaf,parameter=list(minlen=1,support=0.05,confidence=0.05),appearance =
list(rhs=c("NPS_Type=Detractor"),default="lhs"), control=list(verbose=F))

goodrules<-rulesa[quality(rulesa)$lift>1]

summary(goodrules)

inspect(goodrules)


#Scatterplot and parallel plot Leisure Detractor

plot(goodrules, method="graph", control=list(type="items"))

plot(goodrules, method="paracoord",control=list(reorder=TRUE))

#Inferences

#No bell stadd and no Valet paring likely means that the leisure customer is a detractor.

#Medium O_Type (Overall Satisfaction) means that the leisure customer is likely a detractor.

#Despite having a Shuttle service and Fitness center, leisure customers tend to be detractors.

#This may be due to poor quality of service or equipment at these places.


#aRule and filter rule based on lift

rulesa <- apriori(newdattaf,parameter=list(minlen=1,support=0.05,confidence=0.5),appearance =
list(rhs=c("NPS_Type=Promoter"),default="lhs"), control=list(verbose=F))

goodrules<-rulesa[quality(rulesa)$lift>1.295]

summary(goodrules)

inspect(goodrules)


#Scatterplot and parallel plot Leisure Promoter

plot(goodrules, method="graph", control=list(type="items"))

plot(goodrules, method="paracoord", control=list(reorder=TRUE))

#Inferences
```

#I (Check In Process), F (Staff Cared),S (Customer Service),C (Condition of Hotel),T (Tranquility),G (Guest Room),O_Types (Overall Satisfaction) being high implies that a leisure user is likely a promoter.

#Having laundry generally means that a leisure user is a promoter.

#For factors appear in both plots, it can be concluded that we cannot determine NPS type using these.

#Similarly for variables which have both options appearing in a single plot, w cannot conclusively state

#that they help determine NPS Type.