**IST 687 – Applied Data Science**

**Project Report**

# NET PROMOTER SCORE ANALYSIS

Project team:

**Dhruv Kharwar**
**Sweeney Pandit**
**Tushar Mundodu**
**Xichen Yao**
**Yiming Wang**

# Table of Contents


| Sr. No | Topic | Page Nos. |
|---|---|---|
| 1 | Introduction | 3 |
| 2 | Business Questions | 4 |
| 3 | Data Munging | 5-9 |
| 4 | Descriptive Statistics | 10-16 |
| 5 | Data Modelling | 17-31 |
| 6 | Validation of Data Modelling Outcomes | 32-34 |
| 7 | Business Recommendations | 35 |
| 8 | Appendix A – R Code | 36-56 |

# Introduction

The aim of the project is to analyse the customer data provided by the Hyatt chain of hotels and provide them with actionable insights that will enable them to improve customer satisfaction. Since their revenue stream depends solely on their customers, it is important to understand why customers get classified as promoters, passives or detractors.

The dataset that has been provided for analysis is very large (16.3 gigabytes). The data is spread over a year and divided up by month – starting with February 2014 and ending with January 2015. It comprises of 15.7 million records that contain 237 variables. It would be extremely difficult, time consuming and possibly unproductive to analyse data of this size. Thus, before delving into the actual analysis of the data, we must think at a higher level about what data might be relevant to us. In order to this, we will start by developing a set of business questions.

The general approach to producing actionable intelligence is outlined below:

1. Brainstorming for pertinent business questions
2. Identifying a set of relevant variables based on the business questions
3. Importing selective portions of the dataset based on the chosen variables
4. Statistical mapping of the data
5. Refining and reducing the dataset based on observations made from the statistical maps
6. Using descriptive statistics to deduce a set of variables that drive NPS
7. Using validation models to verify that the significance of deductions from statistical models
8. Proposing a set of solutions to the business questions

# Business Questions

Below, we have compiled a set of business questions that will enable Hyatt to understand how to improve their business by converting customers to promoters.

1. Who are the biggest detractors?
2. What is causing the detraction?
3. What can be improved to increase the number of promoters?
4. How does the age range and purpose of visit affect the likelihood of recommendation for hotels in United States?
5. Which is the most frequently visited hotel brand for business purposes in California?
6. Which age group gives a better accumulative recommendation and for which hotel brand?
7. How does NPS % vary based on Number of responses in the USA?
8. How do promoters that are visiting for business reasons affect the revenue for cities?
9. How do detractors that are visiting for business reasons affect the revenue for cities?
10. How is the age range affecting survey result for business visitors?
11. How does age range and gender of business related customers affect the NPS type for California state?

# Data Munging

At the outset of this process, since the data available to us was of high volume, we will be using data from three of the twelve months. Trying to assess data for all 12 months caused major performance issues on the software (RStudio) and hardware (laptops, configurations) that we will be using to perform our analysis. Three out of twelve months represents roughly 25% of the data that was made available to us and that is a significant proportion.

## Choosing a Country for Analysis

In order to drill-down on the pertinent data, our first step was find out where Hyatt's biggest market was located. We achieved this by performing:

1. Reduction of data by only including single instances per hotel based on the country that they were located in, their unique property ID and their geographical coordinates
2. Sorting the data alphabetically by country
3. Tabulating the number of occurrences of a country appeared in the data
4. Plotting the locations of each hotel, based on their IDs, on a map of the world

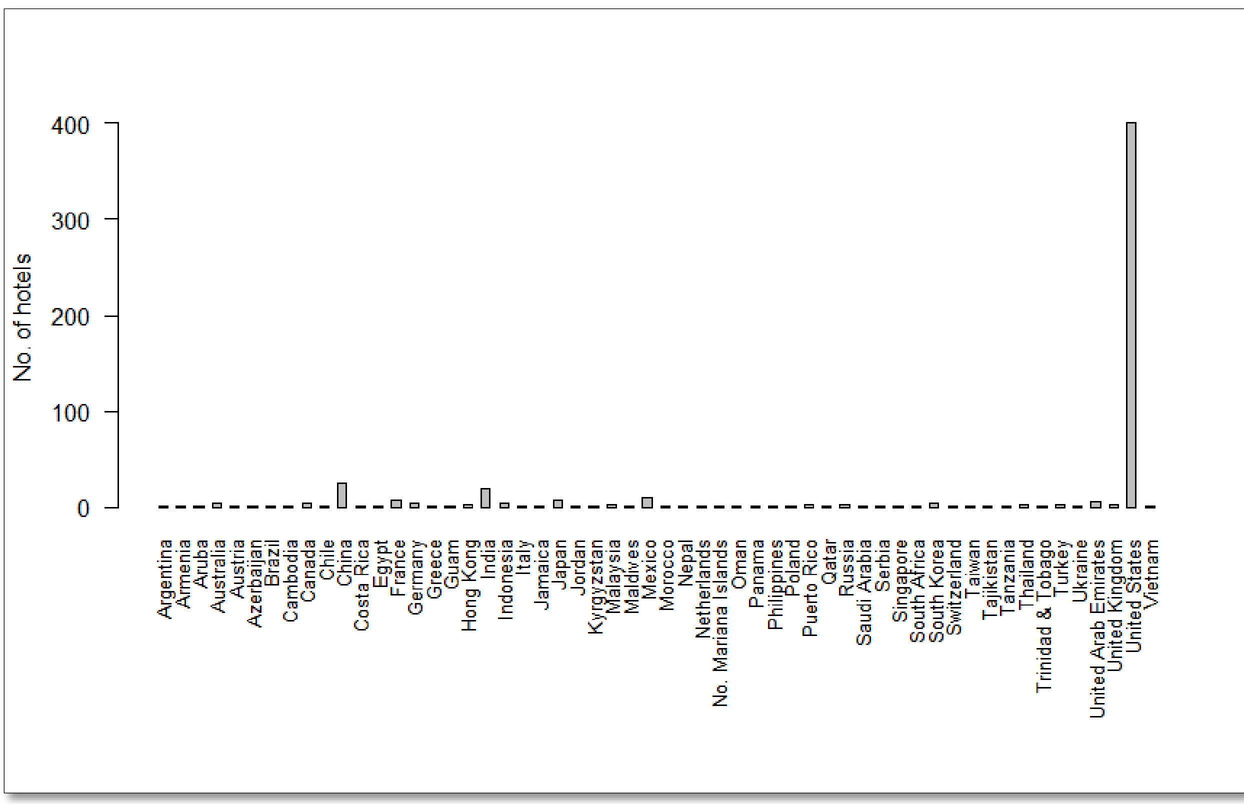



*Figure 1: Plot of the no. of hotels per country*

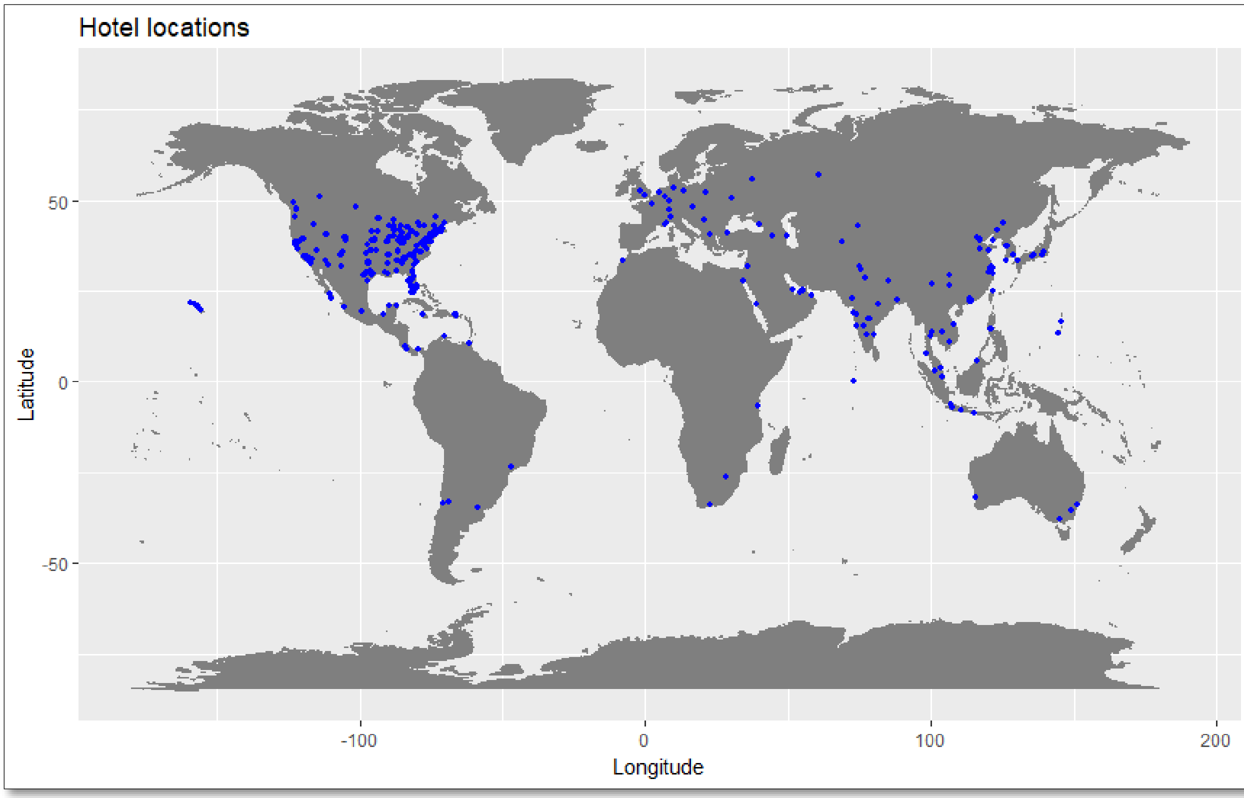



*Figure 2: Global map of the Hyatt hotels*

From the 2 visualizations presented above, it becomes clear that United States of America is the biggest market for Hyatt. Thus, we eliminated records from all other countries. As a result, we were able to reduce the size of the dataset by 25% (approx.: 3.8 million entries to 2.8 million entries).

### Choosing a State within the United States

Within the United States itself, there were approximately 3.8 million entries for analysis. We felt this number was still too large to come up insightful analysis. So, to reduce the size of our data, we decided to focus on just one state.

The first step in deciding this was to plot the number of completed surveys per state and assess what percentage of those surveyed were detractors.

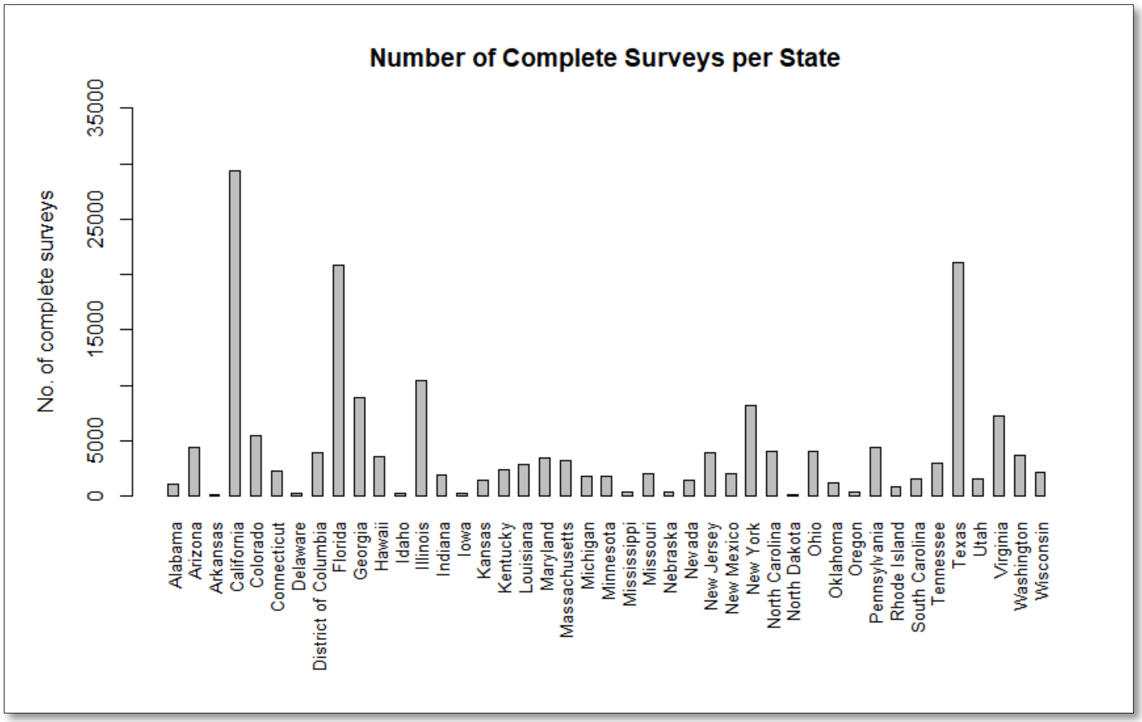



*Figure 3: Number of complete surveys per state*

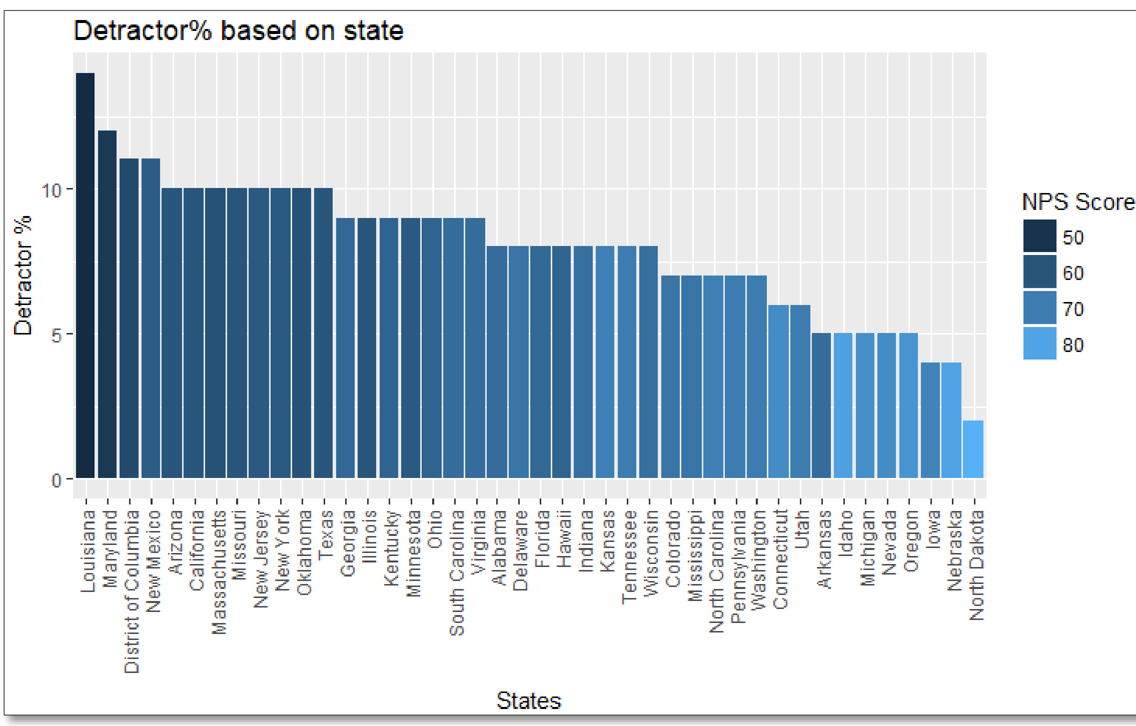



*Figure 4: Percentage of detractors per state*

From the representation in figure no. 3, we can see the most number of complete surveys come from the state of California. In fig. 4 it is seen that Louisiana has the highest percentage of detractors among all the states, however the number of completed surveys available to us for that state is far too low for us to consider it.

Thus, we will choose California as the state for analysis for two primary reasons:

1. It has the largest amount of survey data available to be analysed by almost 10000 entries
2. The percentage of detractors, while not the highest, ranks among the top 5 most "detracted" states in the USA

We feel that the combination of these two factors is sufficient to perform our analysis on it. By removing data for all other states, the size of our dataset

### Choosing Relevant Variables

From the 237 variables available to us, we were able to narrow them down to **30** variables. They are tabulated below:

| Column (Variable) Number | Variable Name |
|---|---|
| 12 | ROOM_TYPE_DESCRIPTION_C |
| 19 | LENGTH_OF_STAY |
| 28 | PMS_TOTAL_REV_USD_C |
| 54 | NT_RATE_R |
| 67 | MEMBER_STATUS_R |
| 107 | Guest_Country_H |
| 108 | Gender_H |
| 109 | Age_Range_H |
| 110 | POV_H |
| 137 | Likelihood_Recommend_H |
| 138 – 147 | Performance Metrics |
| 167 | City_PL |
| 168 | State_PL |
| 170 | Postal.Code_PL |

| | |
|---|---|
| 171 | Country_PL |
| 175 | Property.Latitude_PL |
| 176 | Property.Longitude_PL |
| 179 | Guest.NPS.Goal_PL |
| 182 | BRAND_PL |
| 200 – 227 | Amenity Availabiliy |
| 232 | NPS_Type |

These variables were chosen based on their relevance to the various modelling approaches used by the team.

# Descriptive Statistics

To understand our data and, more specifically, its distribution we came up with the following statistical visualizations. Some of the visualizations are just some observations that we made about the data.

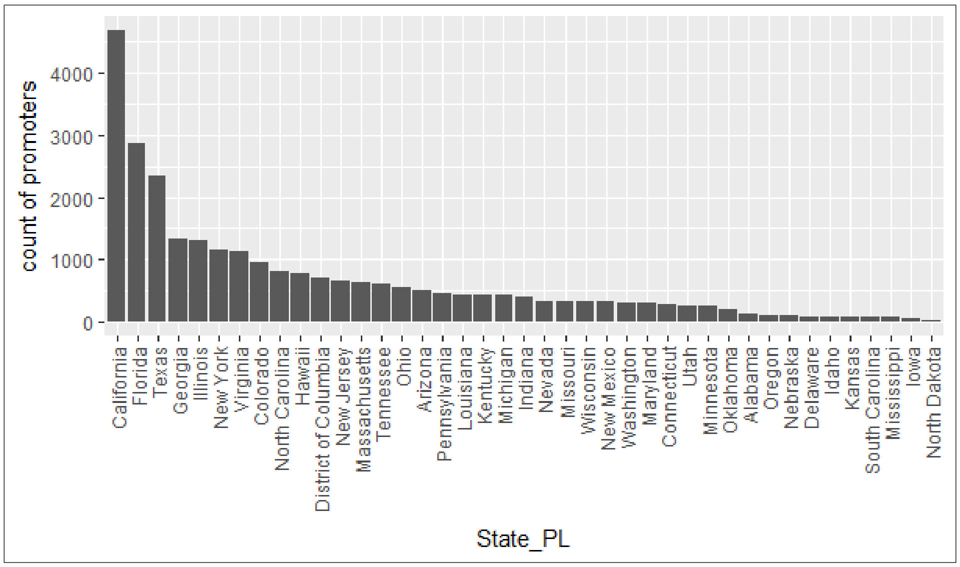



*Figure 5: Distribution of promoters per state*

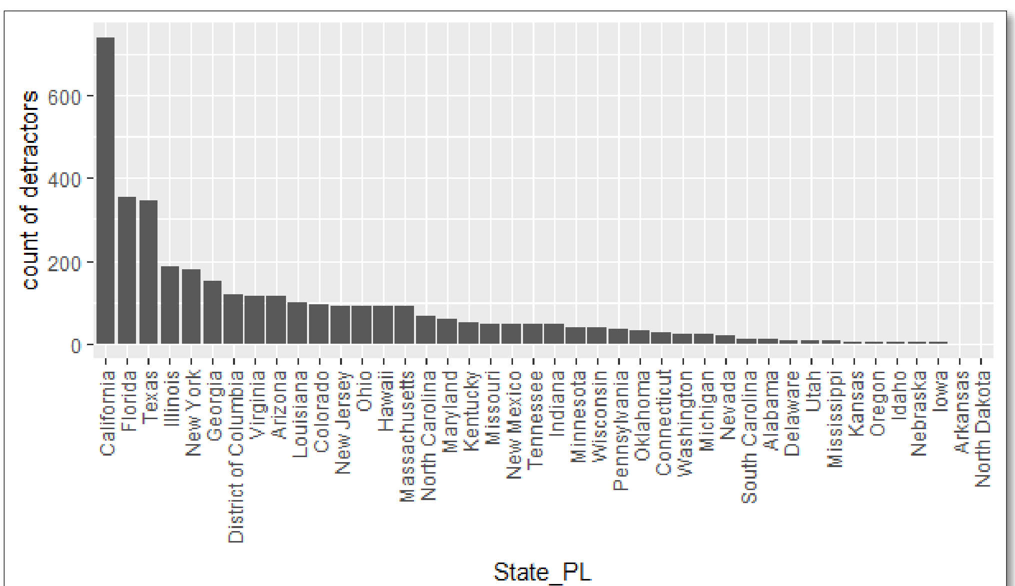



*Figure 6: Distribution of detractors per state*

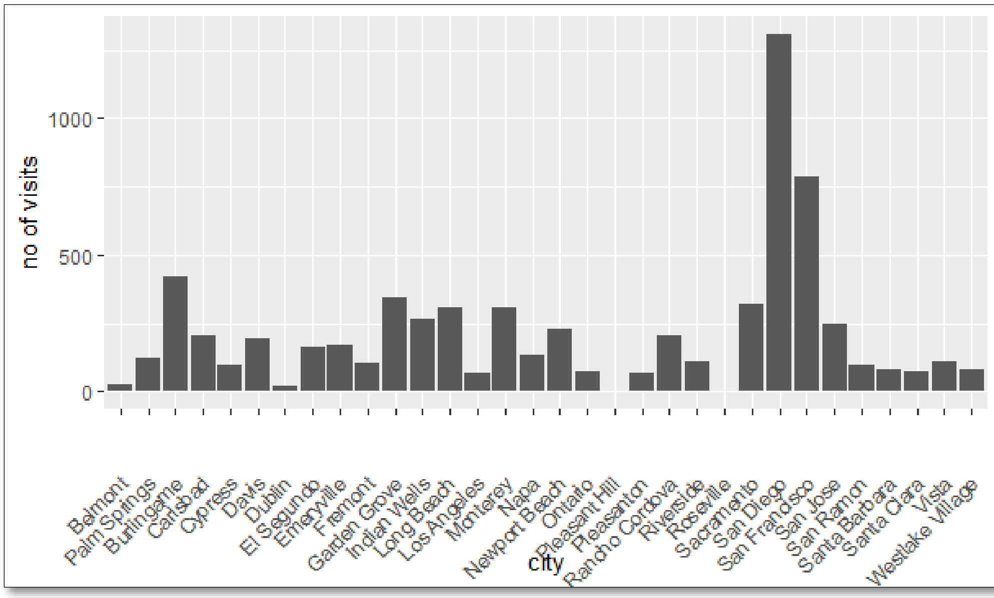



*Figure 7: Distribution of visitors throughout the cities of California*

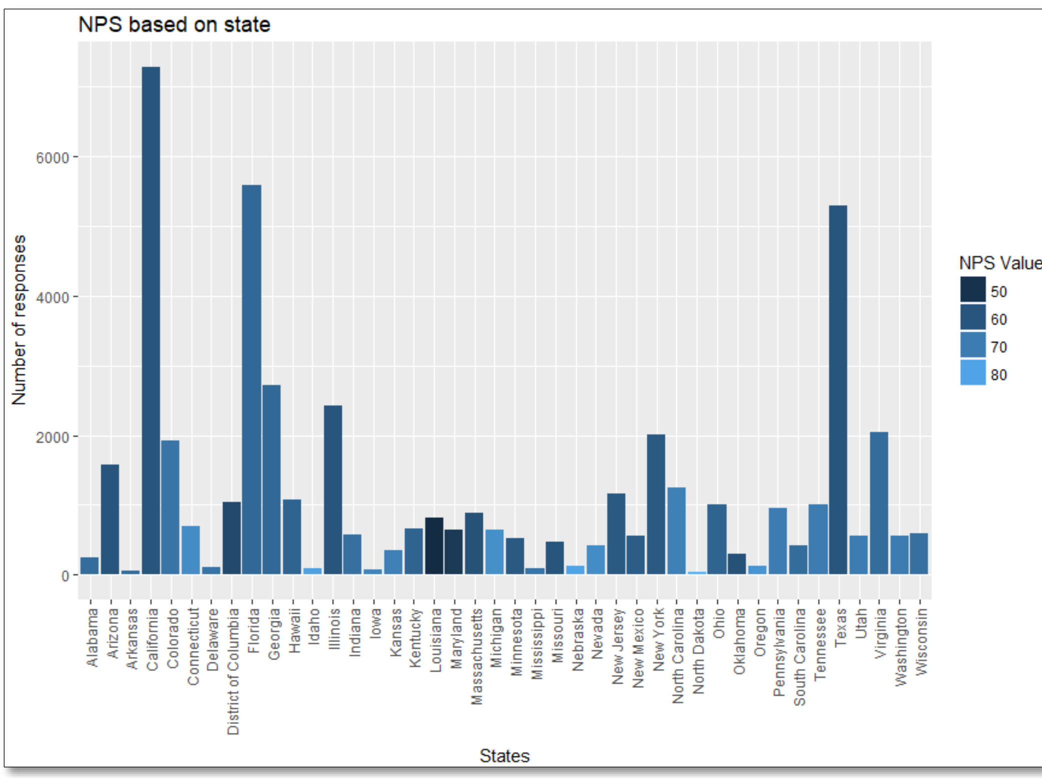



*Figure 8: NPS% per state*

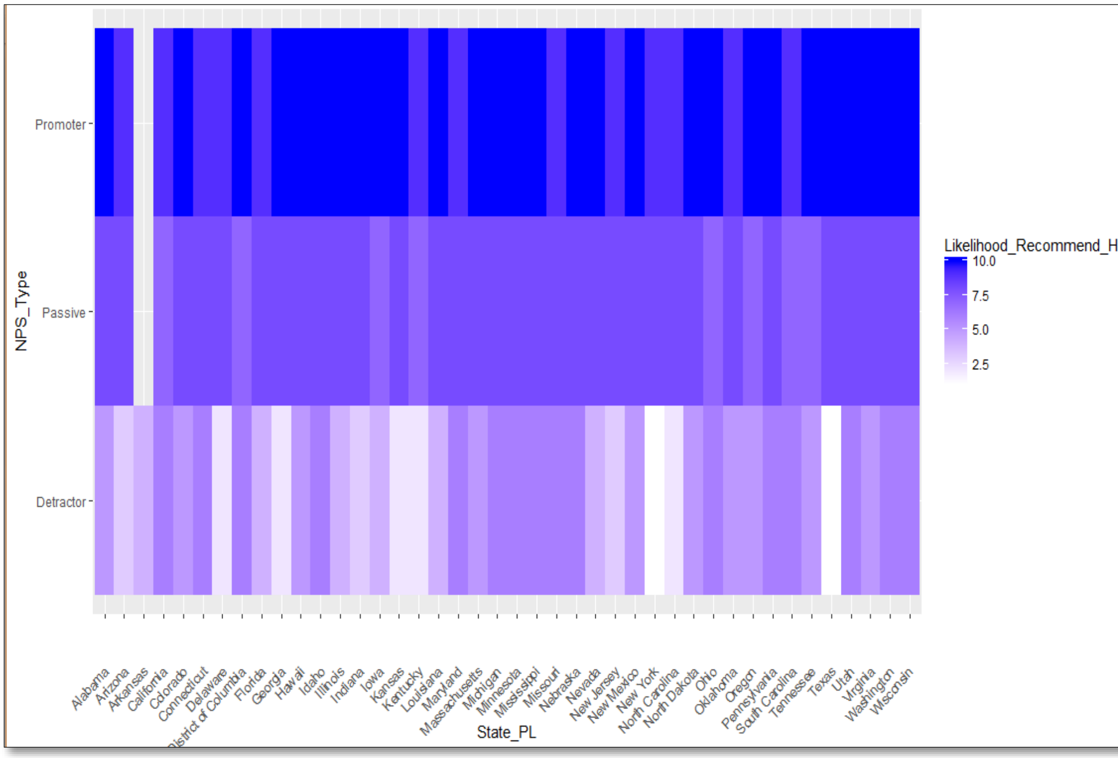



*Figure 9: Heat-map of the NPS types per state*

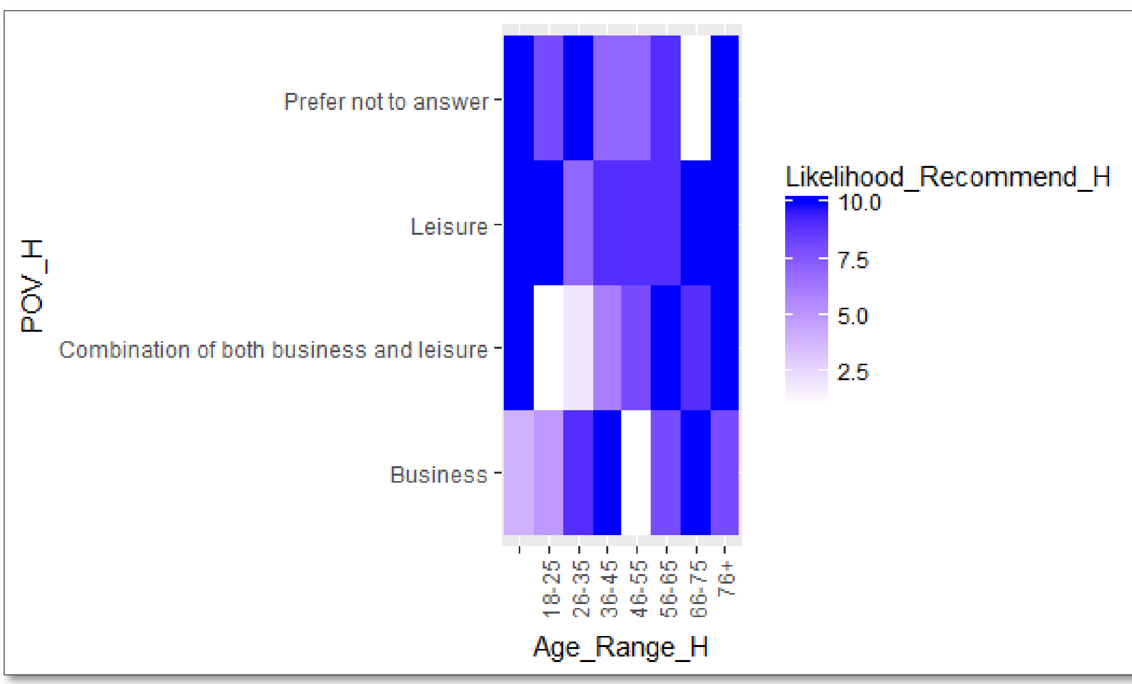



*Figure 10: Customer age vs purpose vs likelihood to recommend*

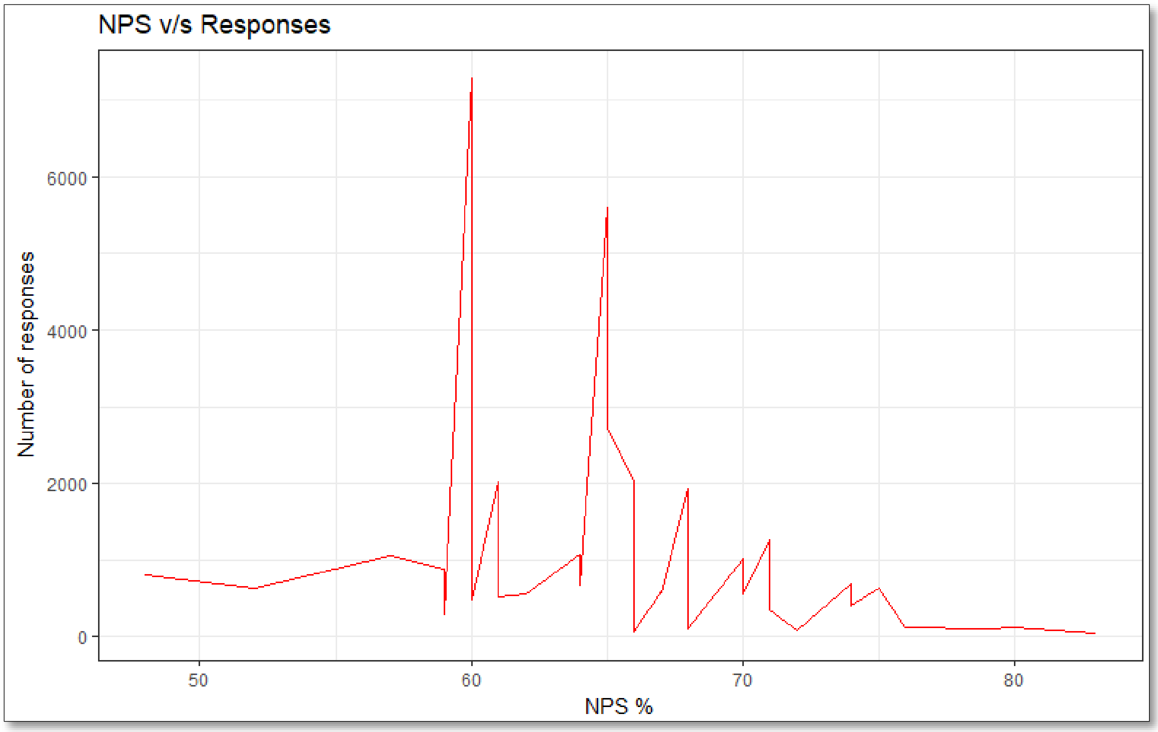



*Figure 11: Distribution of responses per NPS %*

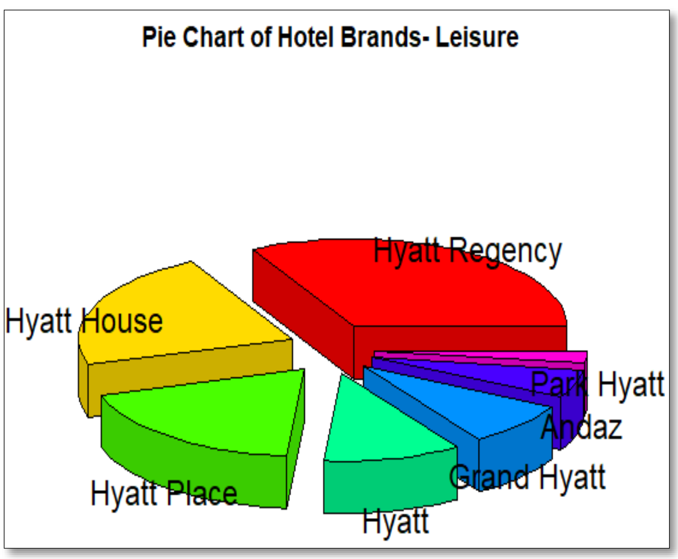



*Figure 12: Number of "leisure" visitors per brand in California*

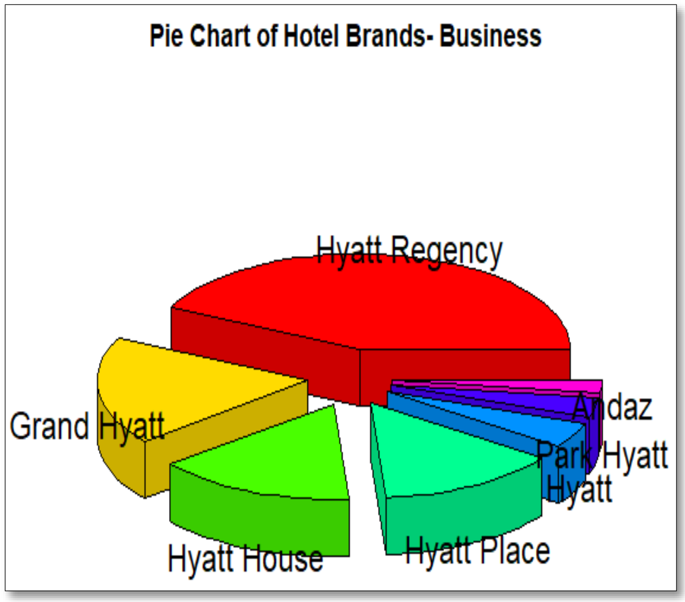



*Figure 13: Number of "business" visitors per brand in California*

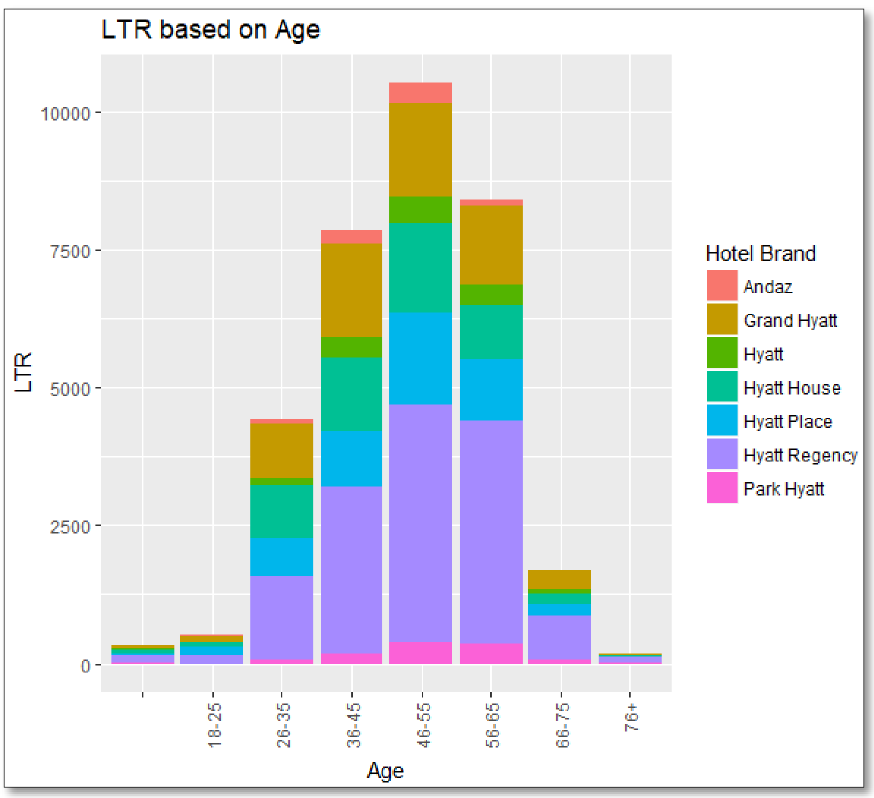



*Figure 14: NPS type versus age per brand*

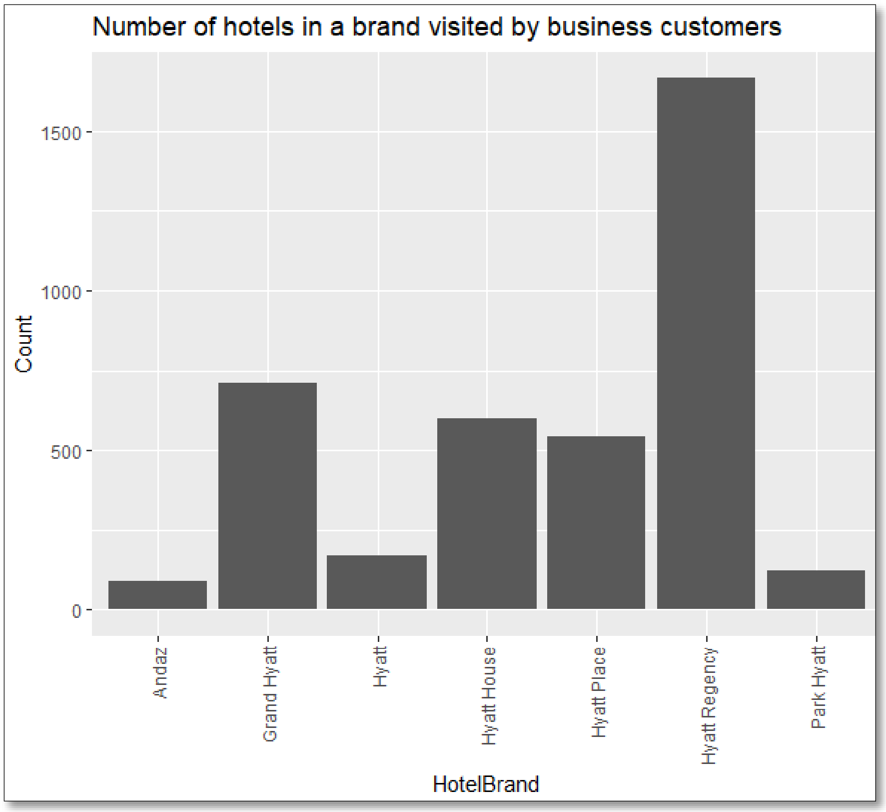



*Figure 15: Number of business visitors per brand*

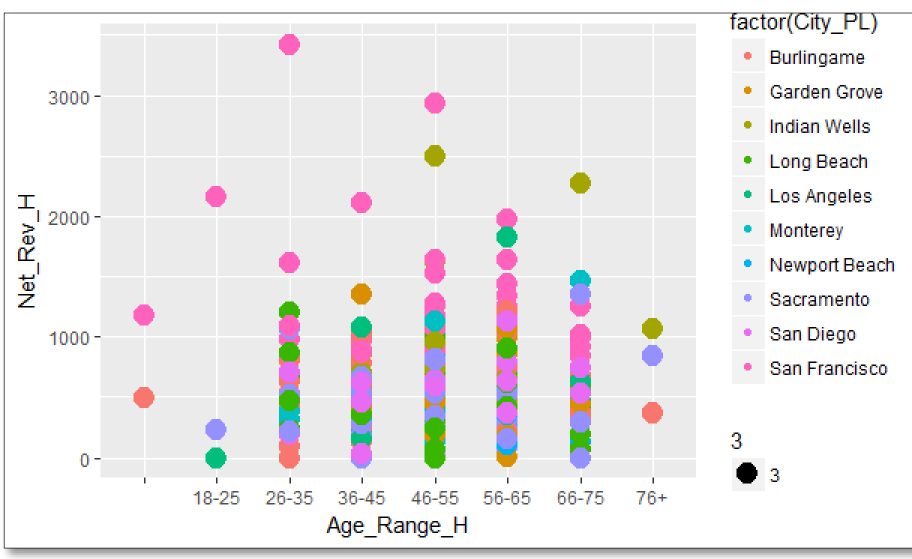



*Figure 16: Revenue generated per age bracket for detractors*

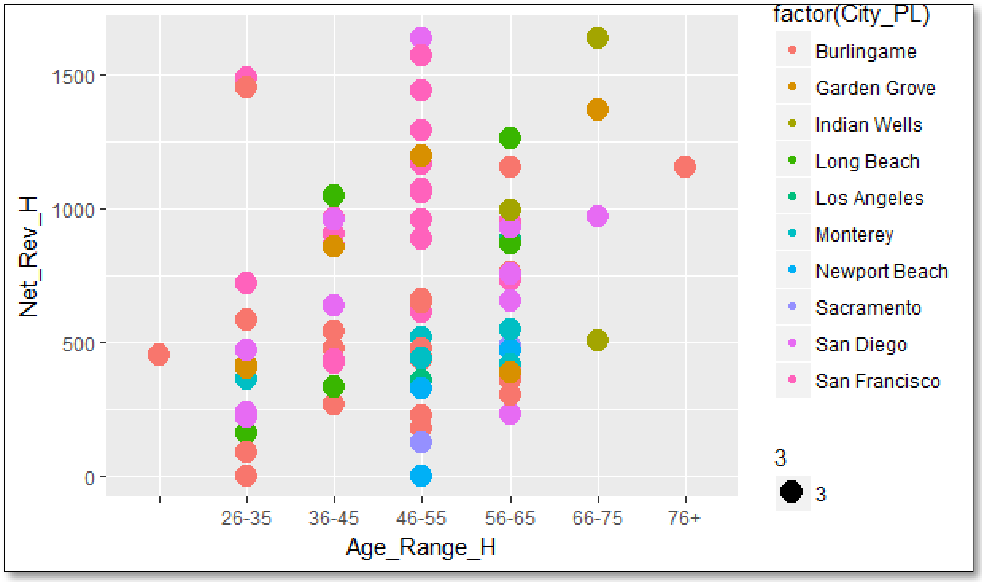



*Figure 17: Revenue generated per age bracket for promoters*

# Data Modelling

The first step in obtaining actionable intelligence is to understand what performance metrics drive the classification of a customer as a promoter, passive or a detractor.

### Association Rules

We have used association rules (A-rules) to determine how the rating of certain facilities in the hotels can affect a customer's likelihood to recommend the hotel. The customers that we have considered for this analysis are those whose purpose for visiting the hotel was business reasons. To determine which rules are of interest, we have used the "lift" value generated as a result of the evaluation as a criterion. The higher the lift value, the higher is that chance of co-occurrence of the dependent variable given the occurrence of the independent variable.

```
rule1 <- apriori(svy_p, parameter = list(support = 0.4, confidence = 0.8))
plot(rule1)
```

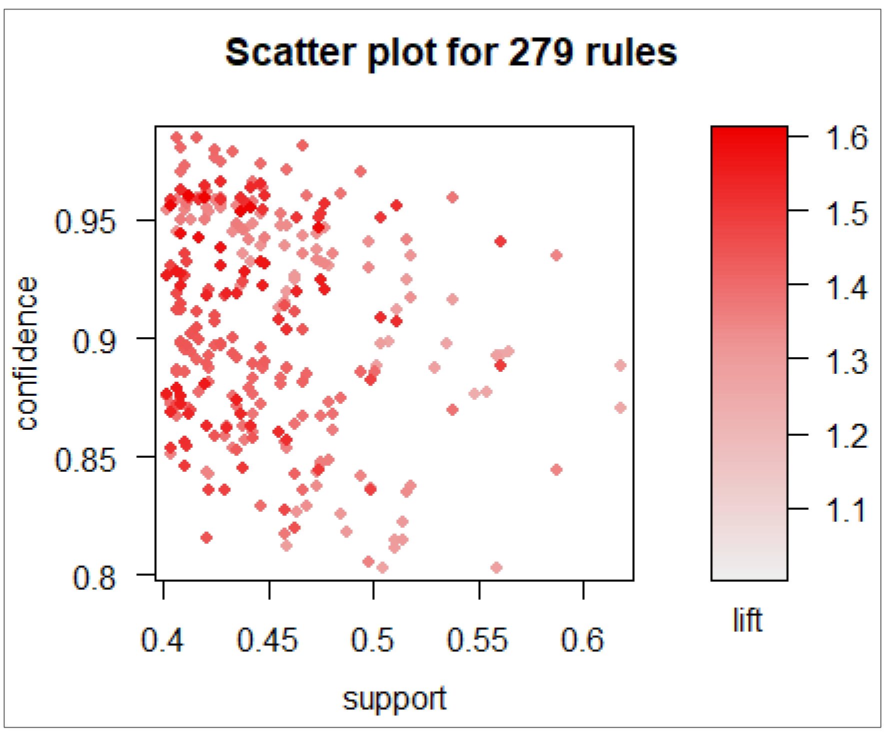



*Figure 18: Scatterplot of the support and confidence values of different rules*

From this scatterplot, we observe that most of the interesting rules with a high lift value and support value close to 0.4 have a high confidence value (up to 0.96). Thus, from the rules inspected in this analysis, we found the following to be the most interesting:

```
# {Room_SV=10, HotelCondition_SV=10, CustomerService_SV=10,
StaffCare_SV=10} => {LikelihoodToRecommend_SV=10} 0.4087673 0.9145013
1.461559
```

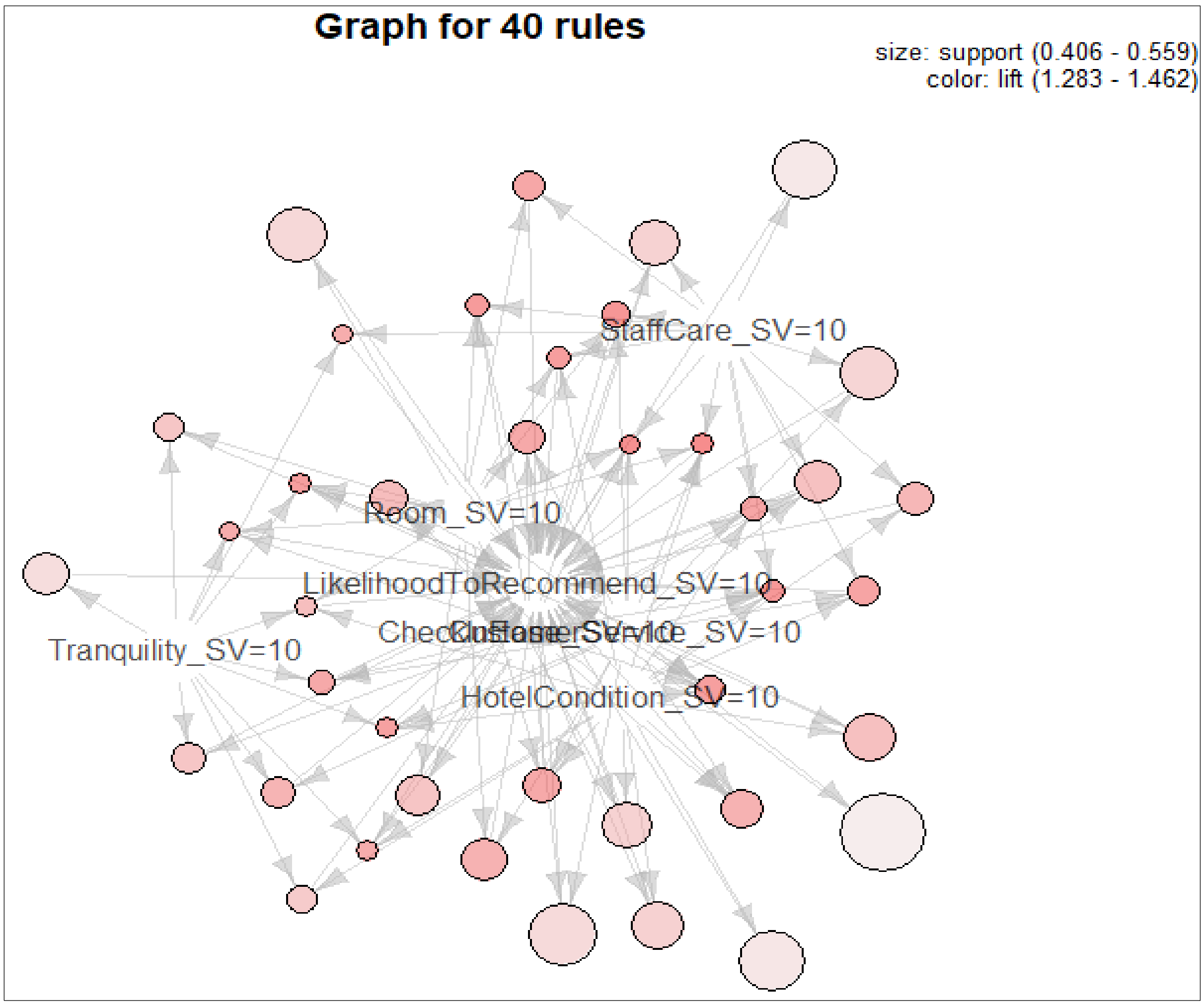



*Figure 19: Graph plotting the most interesting rules*

## Linear Modelling

In order to determine which performance metrics were the primary drivers in determining the NPS classification (promoter, detractor or passive), we employed the linear modelling technique. We used the good rules obtained from the A-rules analysis to determine the most parsimonious model.

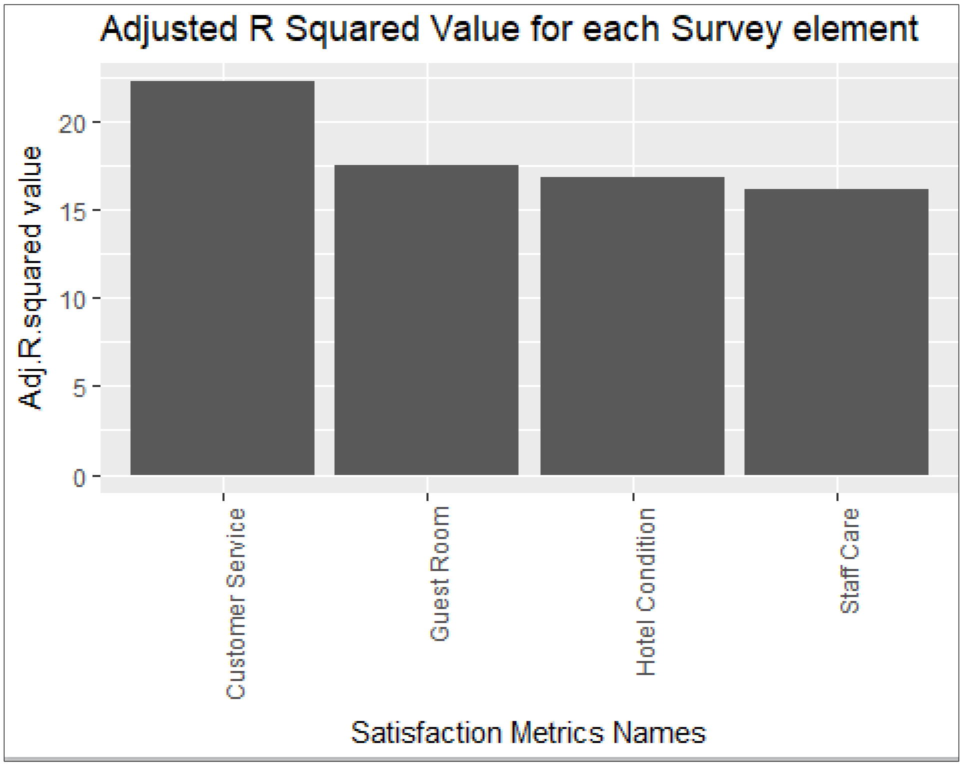
```
Start:  AIC=-4864.94
LikelihoodToRecommend_SV ~ Room_SV + HotelCondition_SV + CustomerService_SV +
    StaffCare_SV

                     Df Sum of Sq    RSS     AIC
<none>                            429.64 -4864.9
- StaffCare_SV        1    4.4270 434.07 -4839.6
- HotelCondition_SV   1    8.0054 437.65 -4817.7
- Room_SV             1   10.9321 440.58 -4799.9
- CustomerService_SV  1   25.5046 455.15 -4713.0

Call:
lm(formula = LikelihoodToRecommend_SV ~ Room_SV + HotelCondition_SV +
    CustomerService_SV + StaffCare_SV, data = svy_ppp)

Coefficients:
       (Intercept)             Room_SV   HotelCondition_SV  CustomerService_SV        StaffCare_SV
           5.10129             0.10615             0.09655             0.20824             0.06296
```

The model was run for an AIC value of -4864.94, which was the least.

```
Call:
lm(formula = LikelihoodToRecommend_SV ~ Room_SV + HotelCondition_SV +
    CustomerService_SV + StaffCare_SV, data = svy_ppp)

Residuals:
    Min      1Q  Median      3Q     Max
-0.8402 -0.3663  0.1598  0.1598  1.5158

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         5.10129    0.13199  38.648  < 2e-16 ***
Room_SV             0.10615    0.01289   8.233 2.82e-16 ***
HotelCondition_SV   0.09655    0.01370   7.045 2.35e-12 ***
CustomerService_SV  0.20824    0.01656  12.575  < 2e-16 ***
StaffCare_SV        0.06296    0.01202   5.239 1.74e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4016 on 2664 degrees of freedom
Multiple R-squared:  0.3127,    Adjusted R-squared:  0.3116
F-statistic: 302.9 on 4 and 2664 DF,  p-value: < 2.2e-16
```

An adjusted R-squared value of 31.16% was obtained for this parsimonious model. As was mentioned in the book by Professors Stanton and Saltz, "in the analysis of human behavior, which is notoriously unpredictable, an r-squared of 20% or 30% may be very good." For this result, the most chances of co-occurrence of the dependent variables on likelihood to recommend were: satisfaction with the guest room, rating of the condition of the hotel, rating of customer services and if the staff was perceived to care.

A plot of the adjusted R squared values against these survey columns can be seen below:



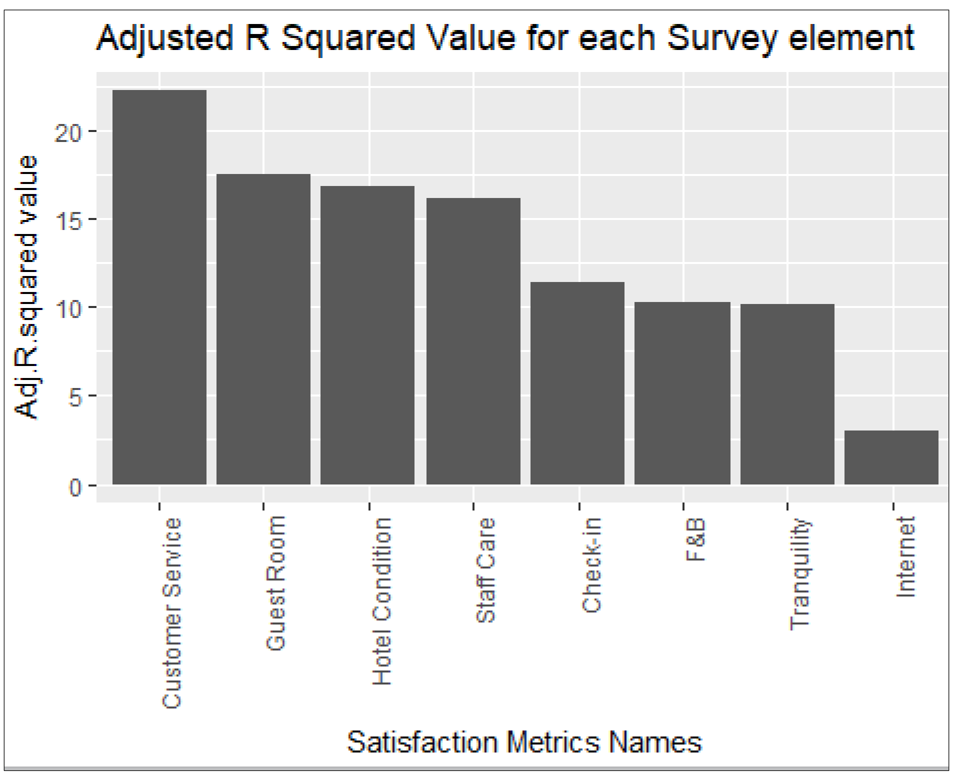

We then ran an LM against all the survey columns to determine the most parsimonious model again:

```
Start:  AIC=-4981.16
LikelihoodToRecommend_SV ~ Room_SV + Tranquility_SV + HotelCondition_SV +
    CustomerService_SV + StaffCare_SV + Internet_SV + CheckInEase_SV +
    F.B_SV

                     Df Sum of Sq    RSS     AIC
<none>                            410.11 -4981.2
- StaffCare_SV        1    1.8533 411.96 -4971.1
- Internet_SV         1    1.9705 412.08 -4970.4
- Tranquility_SV      1    2.3518 412.46 -4967.9
- CheckInEase_SV      1    2.8300 412.94 -4964.8
- HotelCondition_SV   1    4.9038 415.01 -4951.4
- Room_SV             1    6.2812 416.39 -4942.6
- F.B_SV              1    9.5544 419.66 -4921.7
- CustomerService_SV  1   14.9776 425.08 -4887.4

Call:
lm(formula = LikelihoodToRecommend_SV ~ Room_SV + Tranquility_SV +
    HotelCondition_SV + CustomerService_SV + StaffCare_SV + Internet_SV +
    CheckInEase_SV + F.B_SV, data = svy_pp)

Coefficients:
       (Intercept)             Room_SV      Tranquility_SV   HotelCondition_SV  CustomerService_SV
           4.84108             0.08403             0.02705             0.07647             0.16639
      StaffCare_SV         Internet_SV      CheckInEase_SV              F.B_SV
           0.04136             0.01421             0.04389             0.05272
```

The model with the least AIC value in this case was -4981.16.

```
Call:
lm(formula = LikelihoodToRecommend_SV ~ Room_SV + Tranquility_SV +
    HotelCondition_SV + CustomerService_SV + StaffCare_SV + Internet_SV +
    CheckInEase_SV + F.B_SV, data = svy_pp)

Residuals:
     Min       1Q   Median       3Q      Max
-0.90226 -0.32510  0.09774  0.21810  1.39948

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        4.841077   0.132502  36.536  < 2e-16 ***
Room_SV            0.084026   0.013164   6.383 2.04e-10 ***
Tranquility_SV     0.027048   0.006925   3.906 9.63e-05 ***
HotelCondition_SV  0.076467   0.013559   5.640 1.88e-08 ***
CustomerService_SV 0.166391   0.016882   9.856  < 2e-16 ***
StaffCare_SV       0.041359   0.011929   3.467 0.000534 ***
Internet_SV        0.014211   0.003975   3.575 0.000356 ***
CheckInEase_SV     0.043895   0.010245   4.284 1.90e-05 ***
F.B_SV             0.052722   0.006697   7.872 5.03e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3927 on 2660 degrees of freedom
Multiple R-squared:  0.3439,    Adjusted R-squared:  0.3419
F-statistic: 174.3 on 8 and 2660 DF,  p-value: < 2.2e-16
```

A plot of the dependent variables in the most parsimonious model is seen below:



But this model had an adjusted R-squared value of 34.19% as against the model above (run against a-rules result) which had a value of 31.16%, indicating a difference of ~3%. As can be observed from the plot above too, the columns that mostly influence a customer's likelihood to recommend the hotel are: satisfaction with the guest room, rating of the condition of the hotel, rating of customer services and if the staff was perceived to care.

**The result obtained from linear modelling of the data is validated in a later section using Support Vector Machine and Naïve Bayes modelling techniques.**

**Secondary Amenities Affecting Promoter Ratings**

To understand which secondary amenities are responsible for a customer being classified as a "Promoter", we performed more association rule mining to determine strong relationships between the presence of amenities and the NPS type classification of a customer.

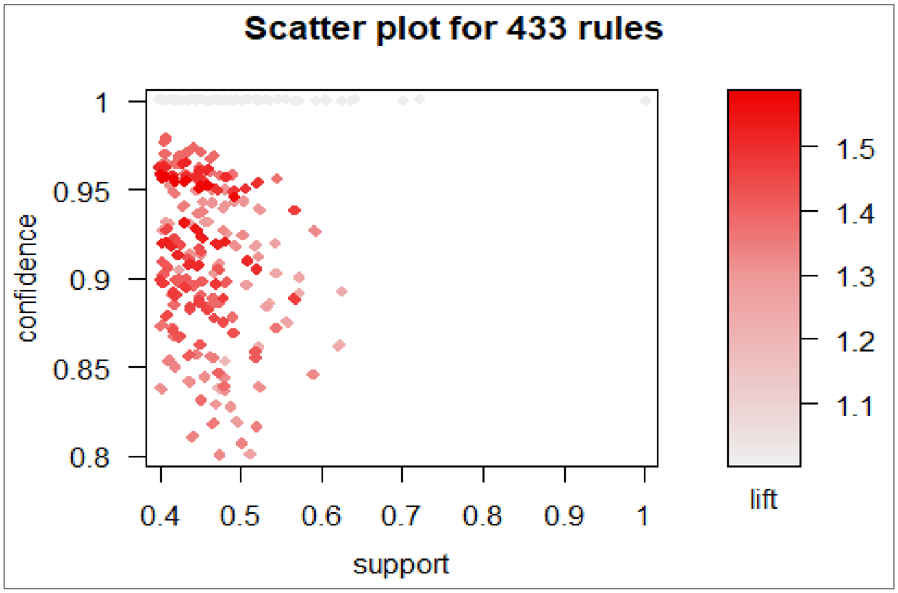
For better understanding and to obtain interesting rules, the secondary amenities were further divided into 4 categories:

1. Spa and fitness
2. Business
3. Transportation (vehicular) arrangements
4. Recreational facilities

To get stronger and more interesting categorical rules, the survey columns were considered after being converted from numeric to strings as per criterion below:

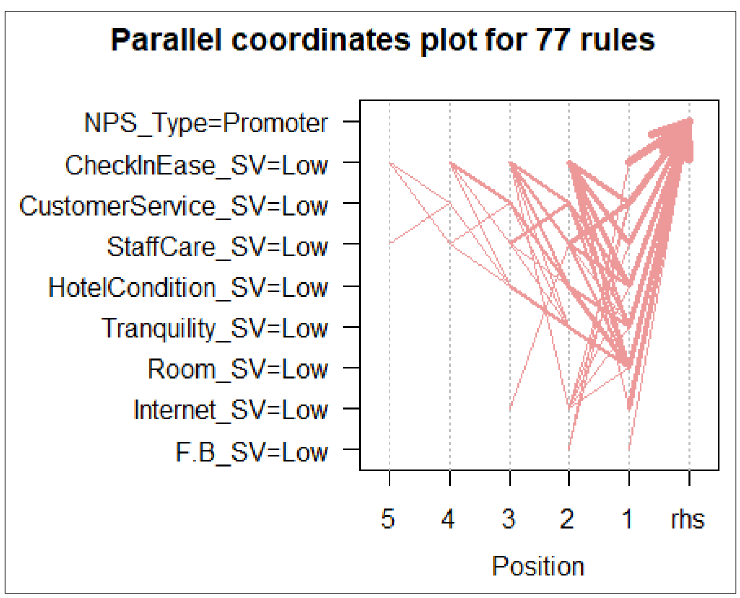
1. Ratings of 9 or 10 → "High" (corresponds to 'PROMOTER')
2. Ratings of 7 or 8 → "Medium" (corresponds to 'PASSIVE')
3. Ratings from 1 through to 6 → "Low" (corresponds to 'DETRACTOR')

```
#ruleset1<-apriori(part1,parameter = list(support= 0.4 ,confidence= 0.8))
```



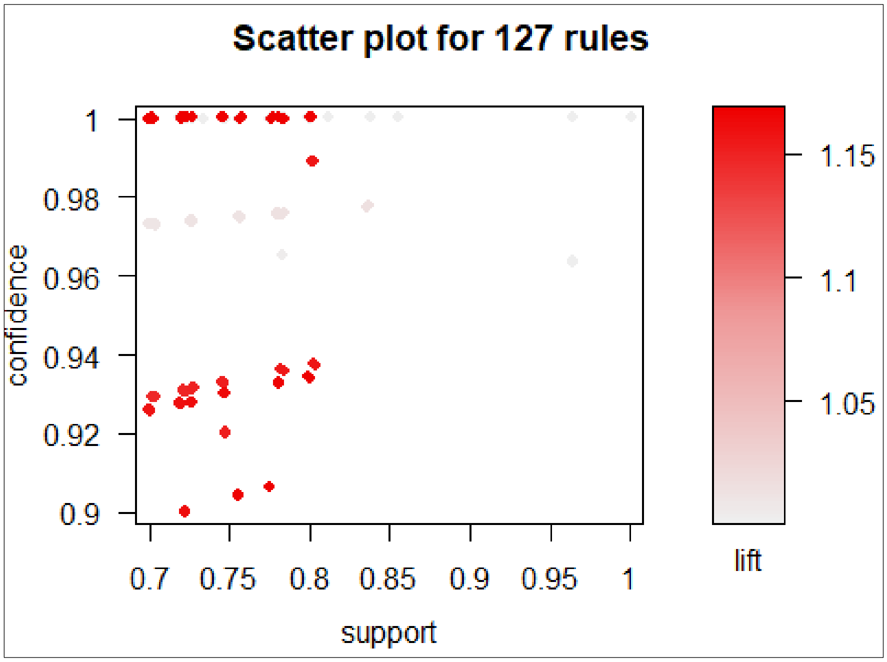

From this scatterplot, we notice that most of the interesting rules with a high lift value and support close to 0.4 have a high confidence (of up to 0.96). From the inspected rules, the co-occurrence of the data items below was the most interesting:

```
#{Room_SV = Low, HotelCondition_SV=Low, CustomerService_SV=Low,
StaffCare_SV=Low} => {NPS_Type=Promoter} 0.4518546 1 1
```



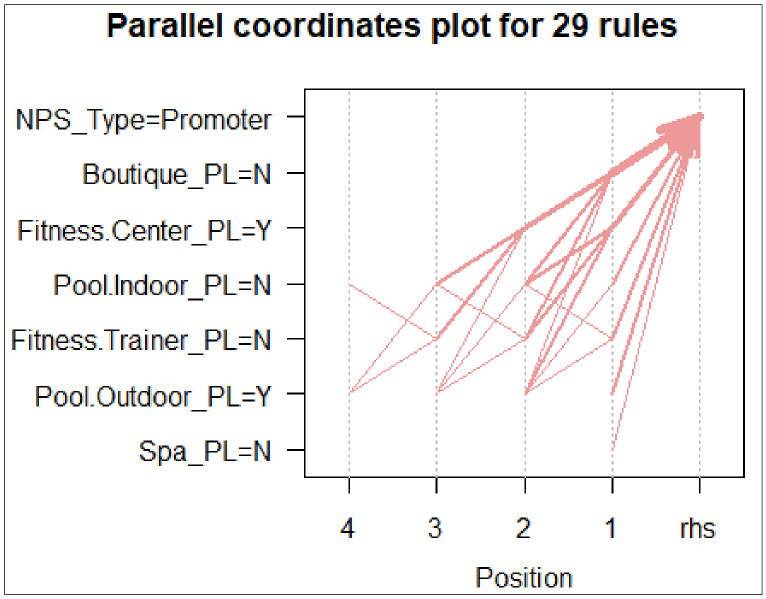
For spa and fitness:

```
# ruleset2 <- apriori(part2.s, parameter = list(support= 0.7, confidence=
0.9))
```



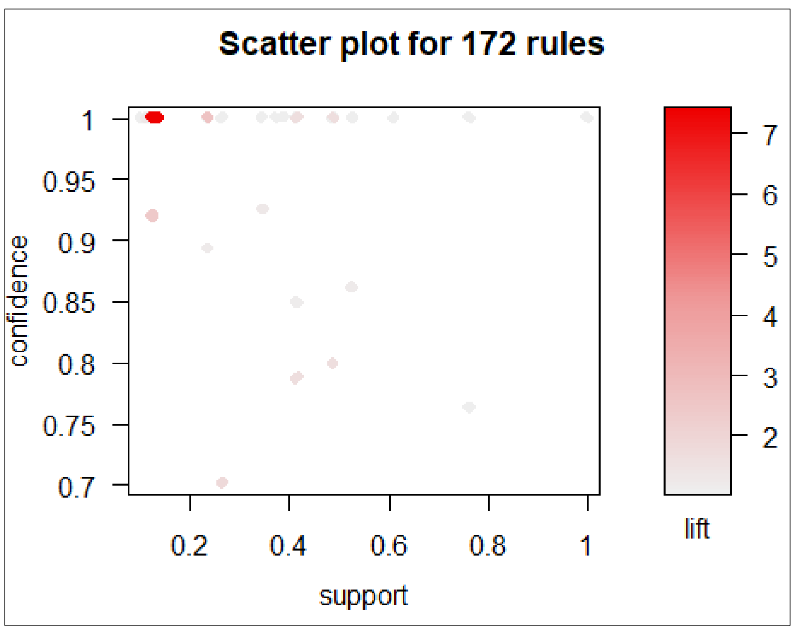

From this scatterplot, we notice that most of the interesting rules with a high lift value and support close to 0.7 have a high confidence (of up to 1). From the inspected rules, the co-occurrence of the data items below was the most interesting rule:

```
#{Boutique_PL=N, Fitness.Center_PL=Y, Fitness.Trainer_PL=N,
Pool.Indoor_PL=N, Pool.Outdoor_PL=Y}    => {NPS_Type=Promoter} 0.6614839
1    1
```



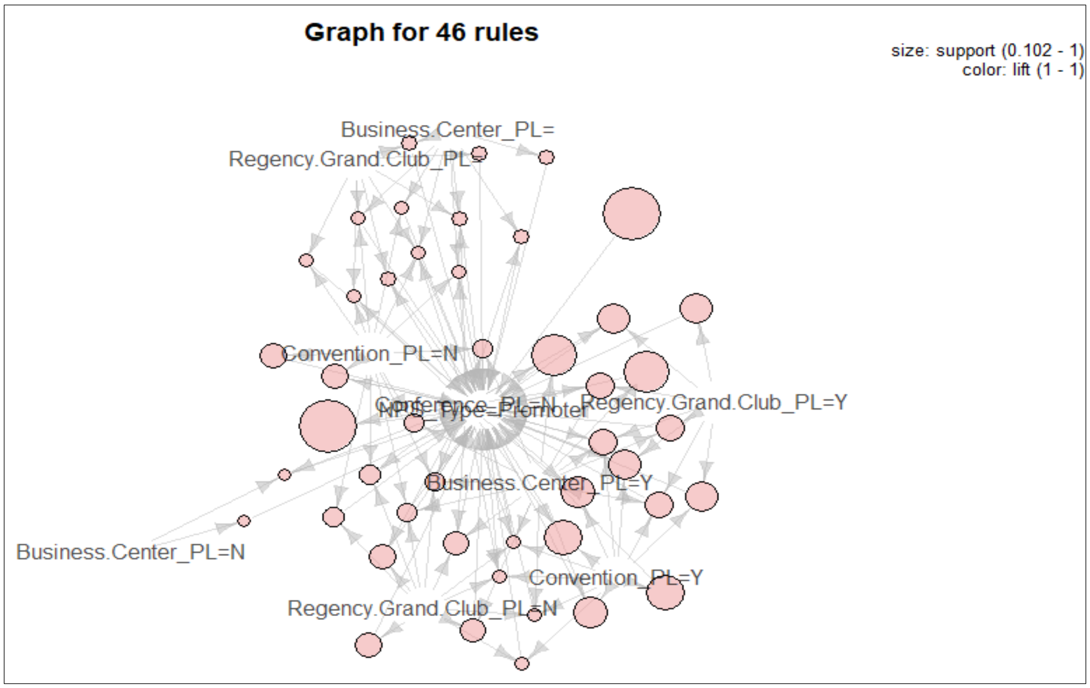

For business:

```
#ruleset4 <- apriori(part2.b, parameter = list(support= 0.1, confidence=
0.7))
```



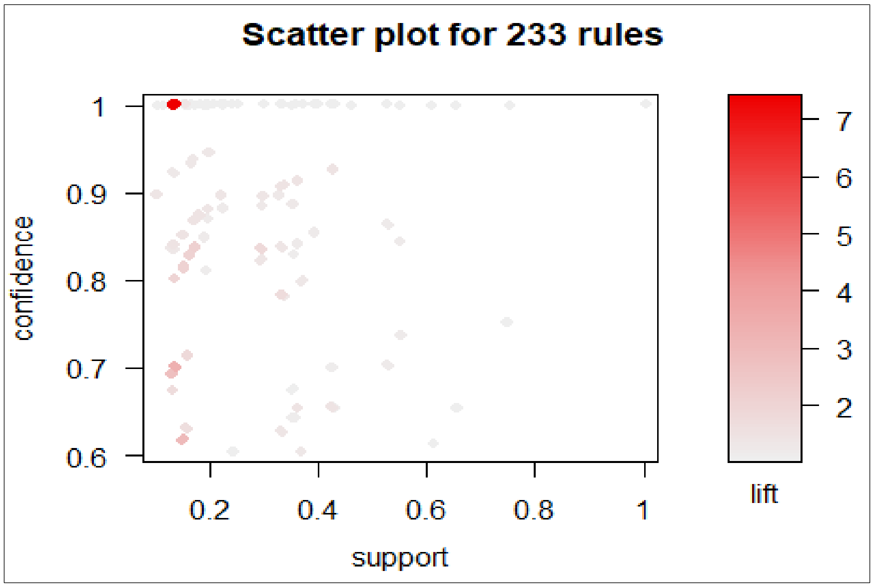
From this scatterplot, we notice that most of the interesting rules with a high lift value and support close to 0.1 have a high confidence of upto 1. From the inspected rules, the co-occurrence of the data items below was the most interesting rule:

```
#{Business.Center_PL=Y, Conference_PL=N, Convention_PL=Y,
Regency.Grand.Club_PL=N} => {NPS_Type=Promoter}
```



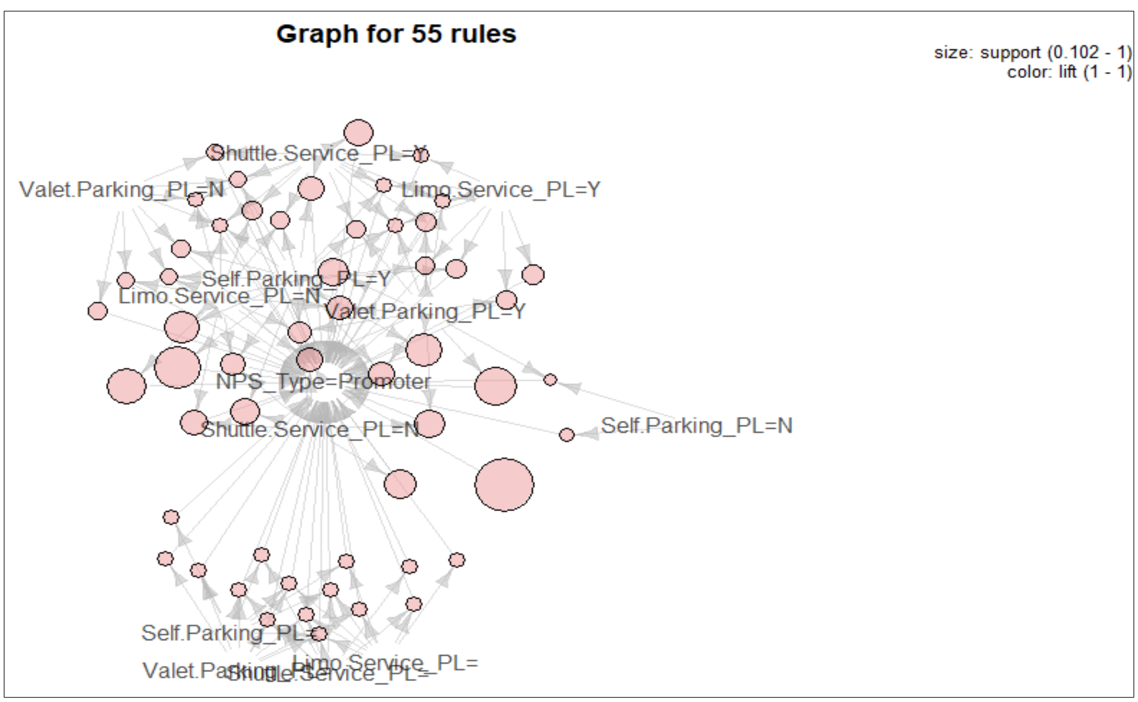

For transportation arrangements:

```
#ruleset3<-apriori(part2.v,parameter = list(support= 0.1 ,confidence=
0.6))
```



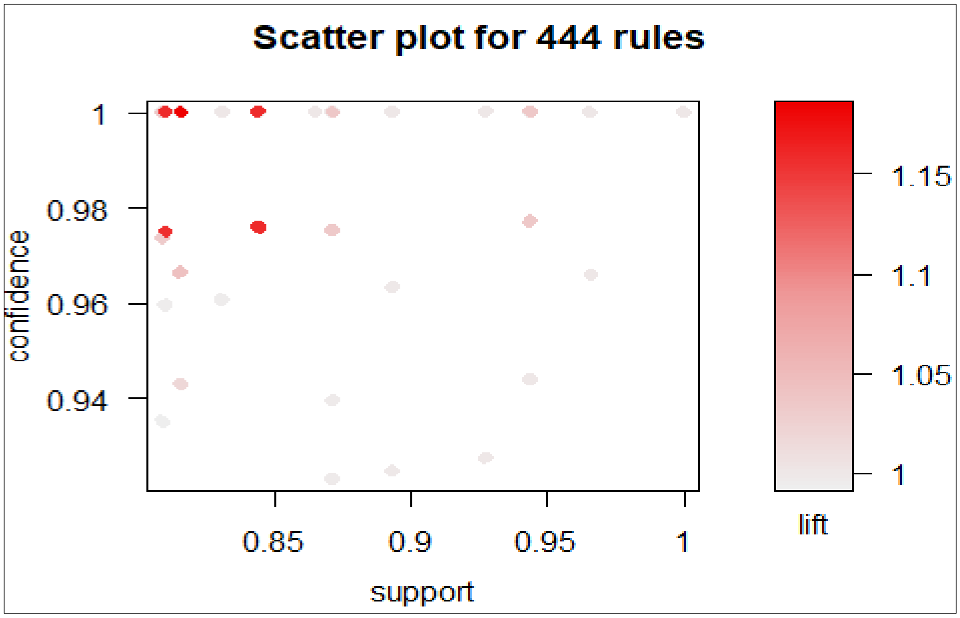

From this scatterplot, we notice that most of the interesting rules with a high lift value and support close to 0.1 have a high confidence (of up to 1). From the inspected rules, the co-occurrence of the data items below was the most interesting rule:

```
#{Limo.Service_PL=Y, Self.Parking_PL=Y,
Shuttle.Service_PL=Y,Valet.Parking_PL=Y}   => {NPS_Type=Promoter}
0.1330086          1    1
```



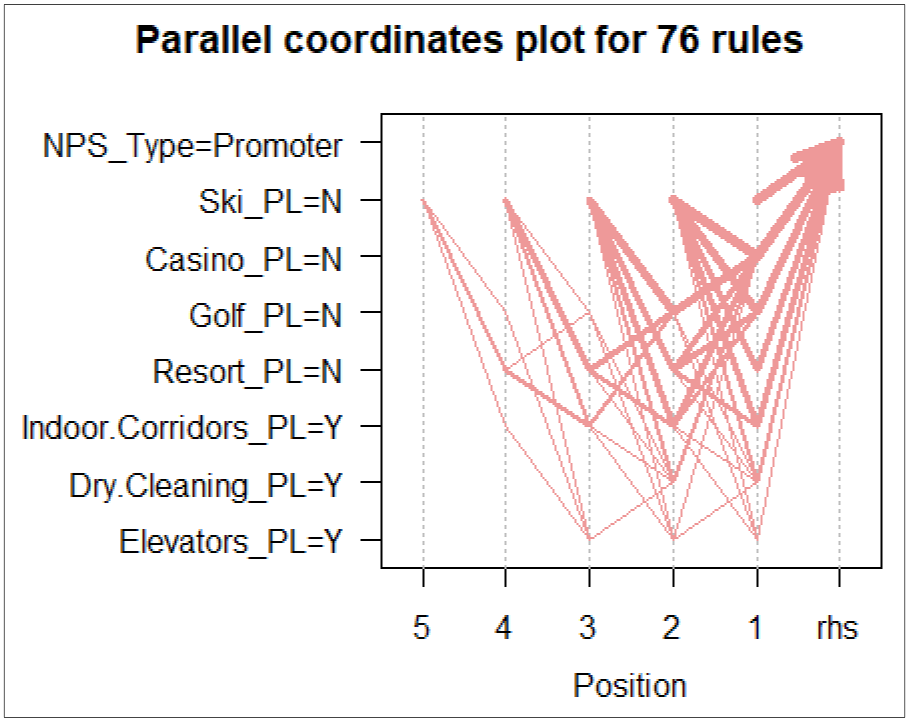
For recreational facilities:

```
# ruleset5<-apriori(part2.r,parameter = list(support= 0.8 ,confidence=
0.9))
```




From this scatterplot, we notice that most of the interesting rules with a high lift value and support close to 0.8 have a high confidence (of up to 1). From the inspected rules, the co-occurrence of the data items below was the most interesting rule:

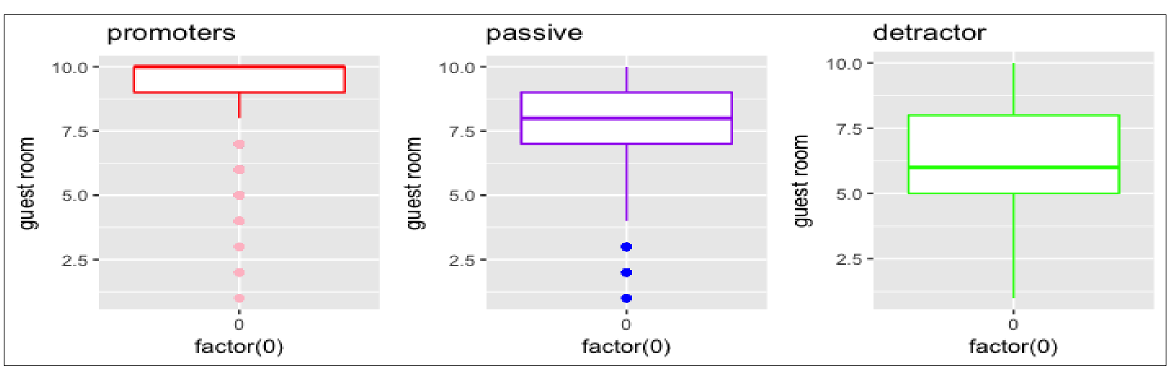
```
#{Casino_PL=N,Dry.Cleaning_PL=Y,Golf_PL=N,Resort_PL=N,Ski_PL=N}
=> {NPS_Type=Promoter} 0.8085425          1    1
```

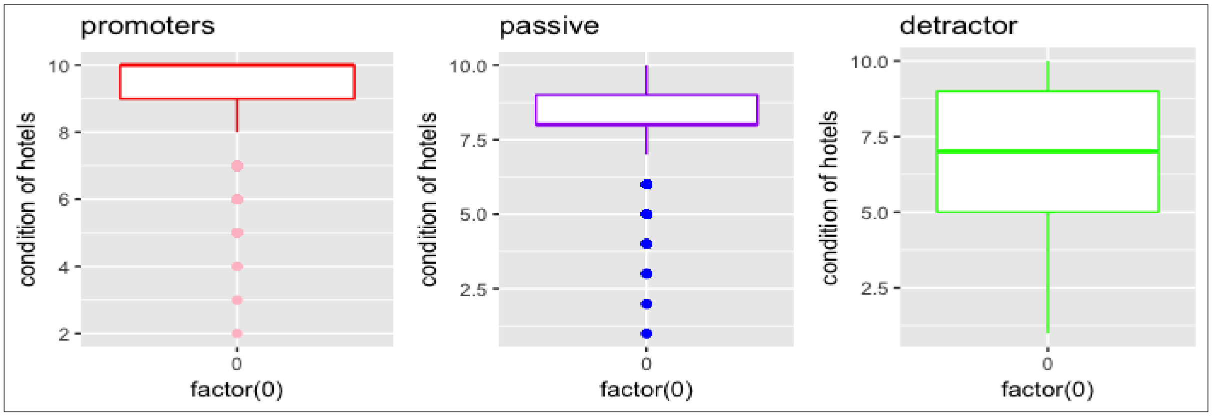


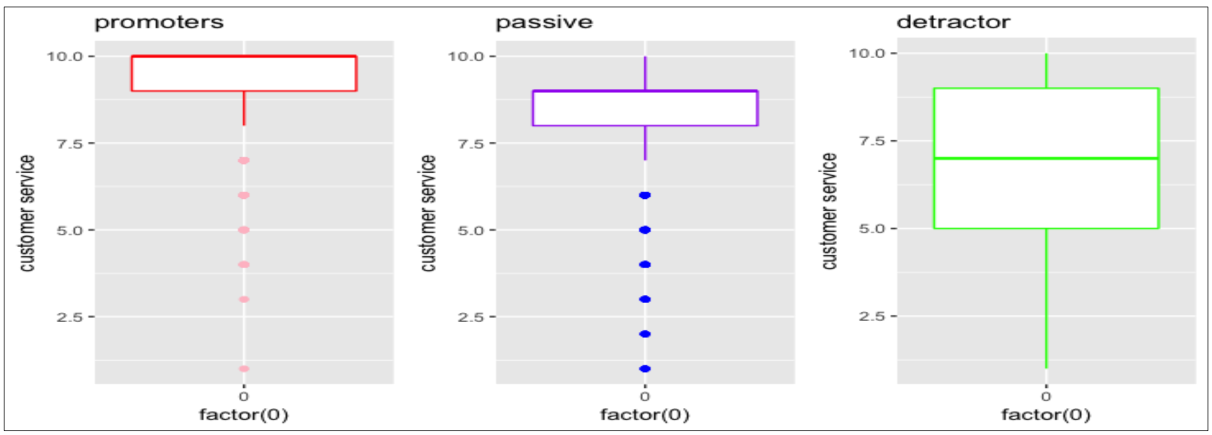

As can be seen from the results above, the columns that most influence NPS_Type to be a 'Promoter' are:

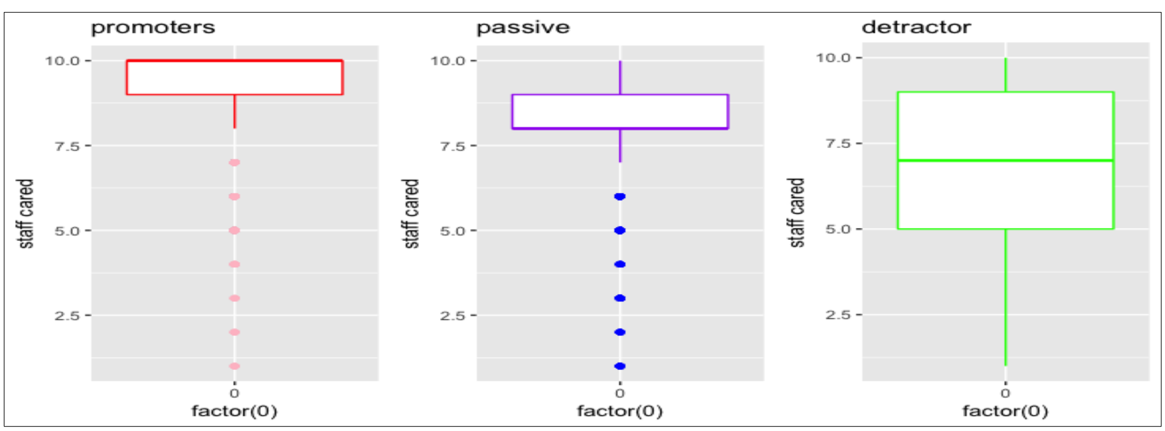
- **Guest Room, Hotel Condition, Customer Service, Staff Care (sustain and improve these services)**
- **Casino, Golf, Resort, Ski, Conference, Regency Grand Club, Boutique, Pool Indoor, Fitness Trainer (provide these facilities)**
- **Limo Service, Self-Parking, Shuttle Service, Valet Parking, Dry Cleaning, Business Center, Convention, Pool Outdoor, Fitness Center (sustain these facilities)**

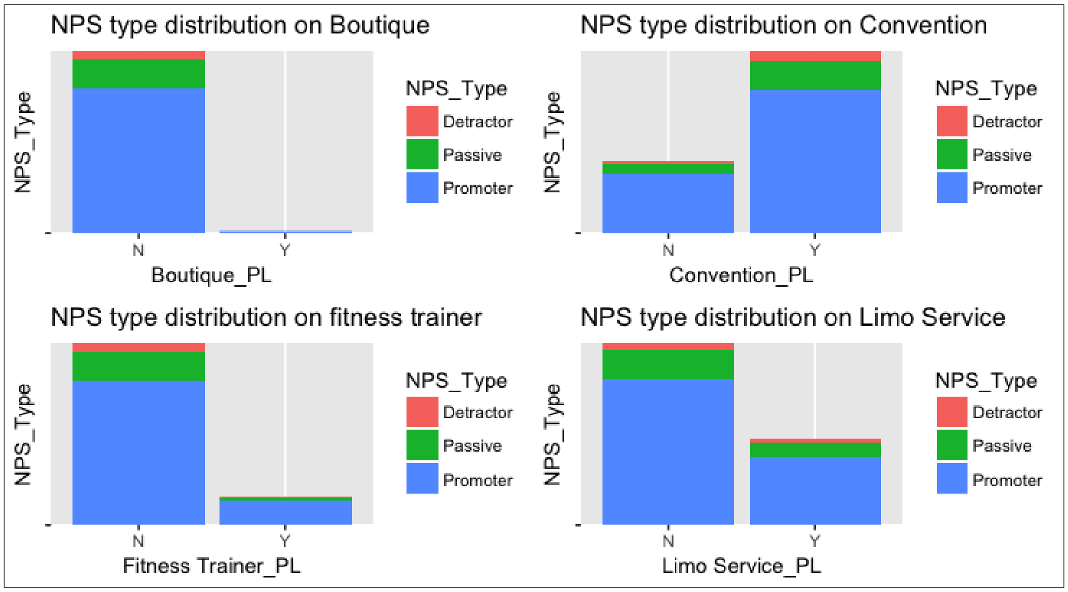
Plotting the occurrence frequency of these dependencies with respect to 'NPS_Type':

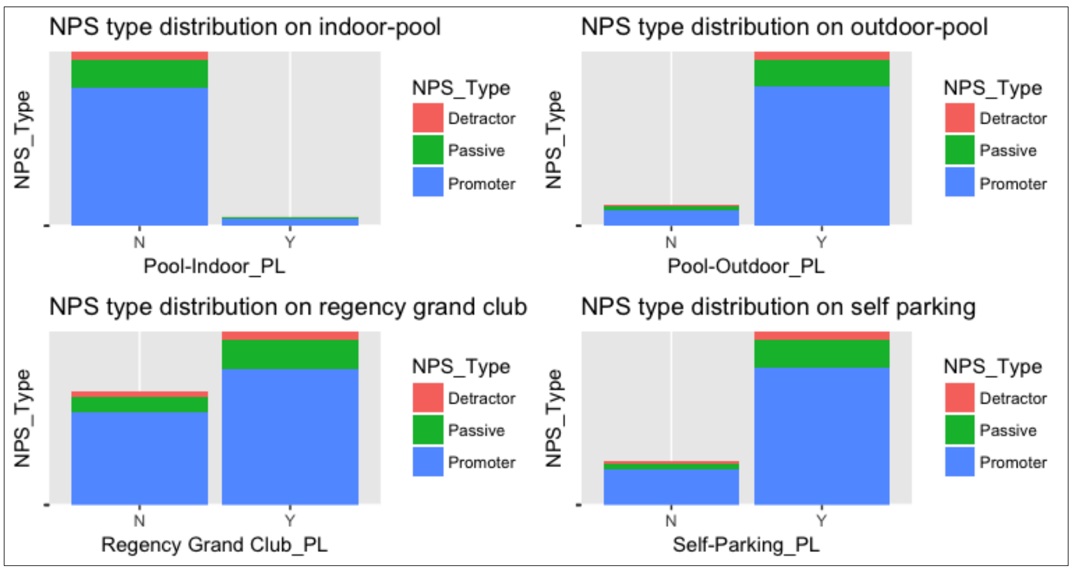



promoters
passive
detractor
condition of hotels
factor(0)


promoters
passive
detractor
customer service
factor(0)


promoters
passive
detractor
staff cared
factor(0)




# Validation of Data Modelling Outcomes

As a result of performing linear modelling, we found that 4 factors affect the NPS type of a customer more than other. In order to validate our results, we will use two more modelling techniques:

1. KSVM (K-Support Vector Machines)
2. Naïve Bayes model

Before performing either of the aforementioned modelling techniques, the test data was divided into 2 parts – one to be used as a training set and the other to be used as a testing set.

**KSVM**

By training the algorithm based on the training data provided (subset of the complete data set), the following model is obtained:

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

SV type: C-svc  (classification)
 parameter : cost C = 1

Gaussian Radial Basis kernel function.
 Hyperparameter : sigma =  0.606071257389265

Number of Support Vectors : 2517

Objective Function Value : -829.215 -512.1651 -1594.834
Training error : 0.204192
```

The next step was to assess the model's accuracy based on the testing data.

From the output obtained as a result testing the model, we can make the following deductions:

- There is a total of 2,863 cases to be evaluated
- Of the total, 2,313 cases were predicted correctly
- Percentage accuracy: 80.79%

```
> #Comparing results
> compTable <- data.frame(testingSet[, "NPS_Type"], testModel)
> table(compTable)
           testModel
NPS_Type    Detractor Passive Promoter
  Detractor       168     112       48
  Passive          31     283      243
  Promoter         10     106     1862
```

**Naïve Bayes**

```
> nbModel

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
Detractor   Passive  Promoter 
0.1215721 0.2099563 0.6684716 

Conditional probabilities:
           Condition_Hotel_H
Y               [,1]      [,2]
  Detractor 6.843391 2.1770317
  Passive   8.437604 1.1469179
  Promoter  9.520512 0.7536529

           Guest_Room_H
Y               [,1]      [,2]
  Detractor 6.186782 2.3971561
  Passive   8.108153 1.3000801
  Promoter  9.455448 0.8264375

           Customer_SVC_H
Y               [,1]      [,2]
  Detractor 7.027299 2.2339697
  Passive   8.594842 1.1641638
  Promoter  9.637314 0.6345579

           Staff_Cared_H
Y               [,1]      [,2]
  Detractor 6.800287 2.3843736
  Passive   8.365225 1.4002100
  Promoter  9.472171 0.8845816
```

Similar to KSVM, the first step is to train the algorithm using the training data. The resulting model is displayed above.

Now the model is constructed, we test it against the testing data:

```
> #Comparing results
> compTable <- data.frame(nbTestingSet[, "NPS_Type"], testModel)
> table(compTable)
         testModel
NPS_Type    Detractor Passive Promoter
  Detractor       187     109       49
  Passive          57     339      216
  Promoter         20     158     1728
```

From the results above, we can make the following inferences:

- There is a total of 2,863 cases to be evaluated
- Of the total, 2,254 cases were predicted correctly
- Percentage accuracy: 78.73%

By conducting these analyses, we have seen a high value for correct predictions, 80.79% and 78.73%, based on the variables deduced from linear modelling. The KSVM model showed the better accuracy of the two models. To conclude, we can state that the variables that we're basing our suggestions are of great importance.

## Business Recommendations

From the analysis that we have conducted, we discovered that the largest market for Hyatt is in California, USA. There is good spread of promoters and detractors which enables us to understand what drives them to be classified as such. The business visitors form a large part of the customer base in California and they are the ones on whom our analysis was based.

Through linear modelling we were able to deduce that the condition of the hotel, perception of whether the staff cared, the condition of the rooms and the customer service were major factors affecting the NPS type for business visitors. This conclusion was validated by the KSVM and Naïve Bayes models.

1. The design of the rooms could be altered to be slightly more conducive to businessmen. Adding small workspaces in the rooms could help to achieve this goal.
2. Customer service can be trained to cater to businessmen better. Businessmen normally are short on time and are usually looking for clear, concise responses to their questions. A knowledge of business hub in the locality of the hotel might be useful knowledge to have.

3. The hotel space in general could be modified to suit professionals. Availability of conference rooms, spaces for collaborative work, etc. can be added.
4. The staff could be better trained to suit the needs to businessmen, as mentioned earlier, they are normally in a hurry and are look for quick service, knowledge of their requirements might be imbued by the staff. If they were able to offer services like dry-cleaning of work clothes, etc. it could be a benefit.
5. 'Guest Room Double' was given the highest recommendation. These customers aged between 46-55 stayed for a longer duration as well (>=10 days) and have certainly brought in a lot of revenue to Hyatt Regency. One possible reason for this could be because the nightly rate of these rooms was <120$/day, as can be seen in the plot. Guests in this age group also tend to spend from 90$/day-350$/day on the room. This age group should be targeted for building revenue as they have stayed in almost all the 6 room types, predominantly- Guest Room Double, Guest Room King, Guest Room Double/Double, High Floor King, in this order.

# Appendix – R Code

## Descriptive Statistics

```
#How is survey affecting the  nps type for USA all brands

melt_df<-data.frame(quarter1map_us[,c(11:20,28)])
melt1<-melt(melt_df,id="NPS_Type")
ggplot(melt1 ,aes(x=value, y= variable ,
group=1))+geom_point(aes(shape=variable,size=4,color=NPS_Type))


#How is age range and purpose of visit affect the likelihood for hotels in United States

heatmap<- data.frame(quarter1[,c(7,8,12)])
heatmap<-heatmap[heatmap$Likelihood_Recommend_H>0,]
heatmap<-heatmap[heatmap$POV_H!="",]
ggplot(heatmap,aes(x=Age_Range_H,y=POV_H))+geom_tile(aes(fill=Likelihood_Recommend_H),stat="id
entity")+scale_fill_gradient(low="white",high="blue")+theme(axis.text.x=element_text(angle=90,
hjust=1,vjust=0.5))


#How are native residents affecting the Likelihood to Recommend.

quarter1_us_us<-quarter1_us[quarter1_us$Guest_Country_H=="USA",]
ggplot(quarter1_us_us,aes(x=State_PL,y=NPS_Type))+geom_tile(aes(fill=Likelihood_Recommend_H))+
scale_fill_gradient(low="white",high="blue")+theme(axis.text.x=element_text(angle=45,hjust=1,v
just=0.5))


#How are foreign  residents affecting the Likelihood to Recommend.

quarter1_us_fg<-quarter1_us[quarter1_us$Guest_Country_H!="USA",]
ggplot(quarter1_us_fg,aes(x=State_PL,y=NPS_Type))+geom_tile(aes(fill=Likelihood_Recommend_H))+
scale_fill_gradient(low="white",high="blue")

#Map visualization of likelihood to recommend in southern california

lamap<- get_map(location = 'la', zoom = 8, color = 'bw')
map4<- ggmap(lamap) + geom_point(aes(x=longitude, y = latitude, color =
Likelihood_Recommend_H, size = 3), data = quarter1map)+ scale_color_gradient(low= "blue", high
= "red")


#What is the distribution of number of detractors per state in United States

detractors<-sqldf('select count(quarter1map_us.NPS_Type),quarter1map_us.State_PL from
quarter1map_us where quarter1map_us.NPS_Type=="Detractor" group by quarter1map_us.State_PL')
detractors_df<-data.frame(detractors)
detractors_df$State_PL<-reorder(detractors_df$State_PL,-
detractors_df$count.quarter1map_us.NPS_Type.)
ggplot(detractors_df,aes(State_PL,count.quarter1map_us.NPS_Type.))+
  geom_bar(stat="identity")+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))+
ylab("count of detractors")
```

```
#what is the distribution of number of promoters per state in US

promoters<-sqldf('select count(quarter1map_us.NPS_Type),quarter1map_us.State_PL from
quarter1map_us where quarter1map_us.NPS_Type=="Promoter" group by quarter1map_us.State_PL')
View(promoters)
promoters_df<-data.frame(promoters)

promoters_df$State_PL<-reorder(promoters_df$State_PL,-
promoters_df$count.quarter1map_us.NPS_Type)
ggplot(promoters_df,aes(State_PL,count.quarter1map_us.NPS_Type.))+
  geom_bar(stat="identity")+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))+
ylab("count of promoters")

#What is the distribution of hotels visits  in the  cities of California.

NO_OF_VISITS<-sqldf('select count(quarter1_us_final_ca.Brand_PL) from quarter1_us_final_ca
group by quarter1_us_final_ca.City_PL')
city<-c(" Belmont","Burlingame","Carlsbad","Cypress","Davis","Dublin","El
Segundo","Emeryville","Fremont","Garden Grove","Indian Wells","Long Beach","Los
Angeles","Monterey","Napa","Newport Beach","Ontario"," Palm Springs","Pleasant
Hill","Pleasanton","Rancho Cordova","Riverside","Roseville","Sacramento","San Diego","San
Francisco","San Jose","San Ramon","Santa Barbara","Santa Clara","Vista","Westlake Village")
hotel_count_df<-data.frame(city,NO_OF_VISITS)
View(hotel_count_df)
ggplot(hotel_count_df,aes(city,NO_OF_VISITS))+
  +
geom_bar(stat="identity")+theme(axis.text.x=element_text(angle=45,hjust=1,vjust=0.5))+ylab("no
of visits")

#How does age range and gender affect the nps type/ltr for california state, business pov,
hyatt regency?

cal<-quarter1_us_final_ca[quarter1_us_final_ca$POV_H=="Business" &
quarter1_us_final_ca$NPS_Type=="Promoter" quarter1_us_final_ca$Brand_PL=="Hyatt Regency" ,]
cal<-cal[cal$Likelihood_Recommend_H==1 | cal$Likelihood_Recommend_H==3 |
cal$Likelihood_Recommend_H==5 | cal$Likelihood_Recommend_H==7 | cal$Likelihood_Recommend_H==9|
cal$Likelihood_Recommend_H==10,]
ggplot(cal ,aes(x=Age_Range_H, y=NPS_Type,
group=1))+geom_point(aes(shape=factor(Likelihood_Recommend_H),color=factor(Gender_H), size=3))

#How is the age range affecting survey result for  business pov, hyatt regency

cal1<-cal[,c(3,6:8,10,11:20,59)]
melt_df<-data.frame(cal1[,c(3:16)])
melt1<- melt(melt_df,id="Age_Range_H")
melt1_df<-cal1[,c(3,7:16)]
melt1<- melt(melt_df,id="Age_Range_H")
melt1<- melt(melt1_df,id="Age_Range_H"
melt1<- melt(melt1_df,id="Age_Range_H")
melt1<-melt1[-c(9298:10330), ]
ggplot(melt1 ,aes(x=Age_Range_H, y= variable , group=1))+geom_point(aes(size=4,color=value))

#How are amenities affecting the NPS Type that is detractor for hyatt regency, business
```

```
ca1<-quarter1_us_final_ca[quarter1_us_final_ca$NPS_Type=="Detractor"&
quarter1_us_final_ca$MEMBER_STATUS_R=="Gold" & quarter1_us_final_ca$Brand_PL=="Hyatt Regency"
& quarter1_us_final_ca$POV_H=="Business",]

ca1_f<- ca1[,c(6,22,28,30,33,34,35,36,40,42,45,46,47,51,52,53,54,60)]

ca1_f$Gender_H<-as.factor(as.character(ca1_f$Gender_H))
ca1_f$City_PL<-as.factor(as.character(ca1_f$City_PL))
ca1_f$All.Suites_PL<-as.factor(as.character(ca1_f$All.Suites_PL))
ca1_f$Business.Center_PL<-as.factor(as.character(ca1_f$Business.Center_PL))
ca1_f$Casino_PL<-as.factor(as.character(ca1_f$Casino_PL))
ca1_f$Conference_PL<-as.factor(as.character(ca1_f$Conference_PL))
ca1_f$Fitness.Center_PL<-as.factor(as.character(ca1_f$Fitness.Center_PL))
ca1_f$Limo.Service_PL<-as.factor(as.character(ca1_f$Limo.Service_PL))
ca1_f$Mini.Bar_PL<-as.factor(as.character(ca1_f$Mini.Bar_PL))
ca1_f$Pool.Indoor_PL<-as.factor(as.character(ca1_f$Pool.Indoor_PL))
ca1_f$Self.Parking_PL<-as.factor(as.character(ca1_f$Self.Parking_PL))
ca1_f$Shuttle.Service_PL<-as.factor(as.character(ca1_f$Shuttle.Service_PL))
ca1_f$Spa_PL<-as.factor(as.character(ca1_f$Spa_PL))
ca1_f$Valet.Parking_PL<-as.factor(as.character(ca1_f$Valet.Parking_PL))
ca1_f$NPS_Type<-as.factor(as.character(ca1_f$NPS_Type))

ruleset<-apriori(ca1_f,parameter = list(support= 0.8 ,confidence= 0.8))
inspect(ruleset)

What are the count range of amenities that are not available for detractors in gold members in
hyatt regency for california

count_cas<-length(grep("N",ca1_f$Casino.Center_PL))
count_con<-length(grep("N",ca1_f$Conference.Center_PL))
count_fit<-length(grep("N",ca1_f$Fitness.Center_PL))
count_limo<-length(grep("N",ca1_f$Limo.Center_PL))
count_bar<-length(grep("N",ca1_f$Mini.Bar_PL))
count_park<-length(grep("N",ca1_f$Self.Parking_PL))
count_pool<-length(grep("N",ca1_f$Pool.Indoor_PL))
count_spa<-length(grep("N",ca1_f$Spa_PL))
ca1_f_df<-data.frame(count_cas,count_bar,count_bus,
count_con,count_fit,count_limo,count_park,count_pool,count_spa)
ca1_f_df<-data.frame(amenities = c("casino", "mini bar", "business
center","conference","fitness center","limo","self park","pool","spa"), nonavailability =
c(count_cas,count_bar,count_bus,
count_con,count_fit,count_limo,count_park,count_pool,count_spa))

ggplot(ca1_f_df, aes(amenities, value)) +  geom_col()



What are the count range of amenities that are available for detractors in gold members in
hyatt regency for california
count_cas<-length(grep("Y",ca1_f$Casino.Center_PL))
count_con<-length(grep("Y",ca1_f$Conference.Center_PL))
count_fit<-length(grep("Y",ca1_f$Fitness.Center_PL))
count_fit<-length(grep("Y",ca1_f$Limo.Center_PL))
count_bar<-length(grep("Y",ca1_f$Mini.Bar_PL))
count_bar<-length(grep("Y",ca1_f$Pool.Indoor_PL))
count_park<-length(grep("Y",ca1_f$Self.Parking_PL))
count_pool<-length(grep("Y",ca1_f$Pool.Indoor_PL))
```

```
count_bar<-length(grep("Y",ca1_f$Mini.Bar_PL))
count_spa<-length(grep("Y",ca1_f$Spa_PL))
count_fit<-length(grep("Y",ca1_f$Fitness.Center_PL))

count_limo<-length(grep("Y",ca1_f$Limo.Center_PL))
ca1_f_df<-data.frame(count_cas,count_bar,count_bus,
count_con,count_fit,count_limo,count_park,count_pool,count_spa)
ca1_f_df<-data.frame(amenities = c("casino", "mini bar", "business
center","conference","fitness center","limo","self park","pool","spa"), availability =
c(count_cas,count_bar,count_bus,
count_con,count_fit,count_limo,count_park,count_pool,count_spa))

ggplot(ca1_f_df, aes(amenities, availability)) +  geom_col()
```

How are length of stay , city , gender along with its distribution with ltr affecting the NPS_Type that is detractor for hyatt regency, business purpose of visit?

How is NPS_Type that is detractor for hyatt regency, business pov affecting the revenue for cities ?

```
ca1<-quarter1_us_final_ca[quarter1_us_final_ca$NPS_Type=="Detractor"&
quarter1_us_final_ca$MEMBER_STATUS_R=="Gold" & quarter1_us_final_ca$Brand_PL=="Hyatt Regency"
& quarter1_us_final_ca$POV_H=="Business",]

ggplot(ca1,aes(x=Age_Range_H, y= Net_Rev_H,
group=1))+geom_point(aes(size=3,color=factor(City_PL)))
```

How is NPS Type that is promoter of hyatt regency, business pov affecting the revenue for cities ?

```
ca1<-quarter1_us_final_ca[quarter1_us_final_ca$NPS_Type=="Promoter"&
quarter1_us_final_ca$MEMBER_STATUS_R=="Gold" & quarter1_us_final_ca$Brand_PL=="Hyatt Regency"
& quarter1_us_final_ca$POV_H=="Business",]

ggplot(ca1,aes(x=Age_Range_H, y= Net_Rev_H,
group=1))+geom_point(aes(size=3,color=factor(City_PL)))
```

## Descriptive Statistics and A-rules

```
library(data.table)
library(ggplot2)
library(gridExtra)
library(arules)
library(arulesViz)
Feb_14 <- fread("C:/Users/xyao0/Desktop/out-201402.csv", select =
c(10,12,14,19,20,23,24,27,43,67,129,137:147,167:169,171,179,182,183,191,232))
Mar_14 <- fread("C:/Users/xyao0/Desktop/out-201403.csv", select =
c(10,12,14,19,20,23,24,27,43,67,129,137:147,167:169,171,179,182,183,191,232))
Apr_14 <- fread("C:/Users/xyao0/Desktop/out-201404.csv", select =
c(10,12,14,19,20,23,24,27,43,67,129,137:147,167:169,171,179,182,183,191,232))
Feb_14$Month <- "2"
Mar_14$Month <- "3"
Apr_14$Month <- "4"
Feb_14 <- na.omit(Feb_14)
```

```
Mar_14 <- na.omit(Mar_14)
Apr_14 <- na.omit(Apr_14)
Quarter <- rbind(Feb_14,Mar_14,Apr_14)
Feb_14_FL <- Feb_14[Feb_14$State_PL == "Florida"]
Mar_14_FL <- Mar_14[Mar_14$State_PL == "Florida"]
Apr_14_FL <- Apr_14[Apr_14$State_PL == "Florida"]
Quarter_FL <- rbind(Feb_14_FL,Mar_14_FL,Apr_14_FL)
#Clean for LM
Quarter_FL_lm <- Quarter_FL[,-1:-10]
Quarter_FL_lm <- Quarter_FL_lm[,-1]
Quarter_FL_lm <- Quarter_FL_lm[,-12:-20]
Quarter_FL_lm <- Quarter_FL_lm[,-10]
View(Quarter_FL_lm)
#LM Likelihood_to_recommend with one variable
model_1 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$Overall_Sat_H, data
= Quarter_FL_lm)
a_1 <- summary(model_1)$r.squared
model_2 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$Guest_Room_H, data
= Quarter_FL_lm)
a_2 <- summary(model_2)$r.squared
model_3 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$Tranquility_H, data
= Quarter_FL_lm)
a_3 <- summary(model_3)$r.squared
model_4 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$Condition_Hotel_H,
data = Quarter_FL_lm)
a_4 <- summary(model_4)$r.squared
model_5 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$Customer_SVC_H,
data = Quarter_FL_lm)
a_5 <- summary(model_5)$r.squared
model_6 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$Staff_Cared_H, data
= Quarter_FL_lm)
a_6 <- summary(model_6)$r.squared
model_7 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$Internet_Sat_H,
data = Quarter_FL_lm)
a_7 <- summary(model_7)$r.squared
model_8 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$Check_In_H, data =
Quarter_FL_lm)
a_8 <- summary(model_8)$r.squared
model_9 <- lm(formula = Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$`F&B_FREQ_H`, data
= Quarter_FL_lm)
a_9 <- summary(model_9)$r.squared
model_10 <- lm(formula =
Quarter_FL_lm$Likelihood_Recommend_H~Quarter_FL_lm$`F&B_Overall_Experience_H`, data =
Quarter_FL_lm)
a_10 <- summary(model_10)$r.squared
rsvalue <- c(a_1,a_2,a_3,a_4,a_5,a_6,a_7,a_8,a_10)
names <-
c("Overall_sat","Guest_Romm","Tranquility","Condition","Customer_Svc","Staff_Cared","Internet"
,"Check_In","F&B_Overall")
graph_1 <- data.frame(names,rsvalue)
plot_1 <- ggplot(graph_1, aes(x=graph_1$names,y=graph_1$rsvalue))+geom_col()+theme(axis.text.x
= element_text(angle = 90,hjust = 1))+xlab("Satisfaction Metrics Names")+ylab("R.squared
value")+ggtitle("R Squared Value for each Survey element")
plot_1
#According to R-squared value, the most reason that influence Likelihood_to_recommend is
Overall satisfaction to the hotel
```

```
#Further more, customers always care about condition of the room, and customer service, and
guest room
mydata_1 <- Quarter_FL_lm[,-2]
mydata_1 <- mydata_1[,-10]
model_all <- lm(formula = Likelihood_Recommend_H~. ,data=mydata_1)
step(model_all, data=mydata_1, direction = "backward")
# what i can get for the lowest AIC value is -1044.12
model_lowest_AIC <- lm(formula = Likelihood_Recommend_H ~ Guest_Room_H + Tranquility_H +
                         Condition_Hotel_H + Customer_SVC_H + Staff_Cared_H + Internet_Sat_H +
                         `F&B_Overall_Experience_H`, data = mydata_1)
summary(model_lowest_AIC)$adj.r.squared
#Adjusted-R-Squared value is 0.6752 which is the maximum one i can find
#Promoter & Detractor
promoter_Quarter_FL <- subset(mydata_1)[which(mydata_1$Likelihood_Recommend_H == 9 |
mydata_1$Likelihood_Recommend_H == 10 )]
detractor_Quarter_FL <- subset(mydata_1)[which(mydata_1$Likelihood_Recommend_H < 7  )]
lm_model_for_promoter <- lm(formula = promoter_Quarter_FL$Likelihood_Recommend_H~., data =
promoter_Quarter_FL)
list(step(lm_model_for_promoter))
#The lowest AIC is -9929.94,promoter_Quarter_FL$Likelihood_Recommend_H ~ Overall_Sat_H +
#Guest_Room_H + Tranquility_H + Condition_Hotel_H + Customer_SVC_H +
#  Staff_Cared_H + Internet_Sat_H + Check_In_H + `F&B_Overall_Experience_H`+
#  Month

#Length graph for promoters, passives, and detractors
graph_3_2 <- data.frame(Feb_14_FL$NPS_Type, Feb_14_FL$Month)
graph_3_3 <- data.frame(Mar_14_FL$NPS_Type, Mar_14_FL$Month)
graph_3_4 <- data.frame(Apr_14_FL$NPS_Type, Apr_14_FL$Month)
a <- length(graph_3_3$Mar_14_FL.NPS_Type[which(graph_3_3$Mar_14_FL.NPS_Type=="Promoter")])
length(graph_3_3$Mar_14_FL.NPS_Type[which(graph_3_3$Mar_14_FL.NPS_Type=="Passive")])
b <- length(graph_3_3$Mar_14_FL.NPS_Type[which(graph_3_3$Mar_14_FL.NPS_Type=="Detractor")])
value_3_3 <- c(1680,425,200)
names_3_3 <- c("Promoter", "Passive", "Detractor")
df_3_3 <- data.frame(value_3_3,names_3_3)
plot_3_3 <- ggplot(df_3_3,
aes(x=df_3_3$names_3_3,y=df_3_3$value_3_3))+geom_col()+theme(axis.text.x = element_text(angle
= 0,hjust = 1))+ xlab("NPS_Type")+ylab("Numbers in March")+ggtitle("March")
plot_3_3
c <- length(graph_3_2$Feb_14_FL.NPS_Type[which(graph_3_2$Feb_14_FL.NPS_Type=="Promoter")])
length(graph_3_2$Feb_14_FL.NPS_Type[which(graph_3_2$Feb_14_FL.NPS_Type=="Passive")])
d <- length(graph_3_2$Feb_14_FL.NPS_Type[which(graph_3_2$Feb_14_FL.NPS_Type=="Detractor")])
value_3_2 <- c(1520,384,191)
names_3_2 <- c("Promoter", "Passive", "Detractor")
df_3_2 <- data.frame(value_3_2,names_3_2)
plot_3_2 <- ggplot(df_3_2,
aes(x=df_3_2$names_3_2,y=df_3_2$value_3_2))+geom_col()+theme(axis.text.x = element_text(angle
= 0,hjust = 1))+ xlab("NPS_Type")+ylab("Numbers in February")+ggtitle("February")
plot_3_2
e <- length(graph_3_4$Apr_14_FL.NPS_Type[which(graph_3_4$Apr_14_FL.NPS_Type=="Promoter")])
length(graph_3_4$Apr_14_FL.NPS_Type[which(graph_3_4$Apr_14_FL.NPS_Type=="Passive")])
f <- length(graph_3_4$Apr_14_FL.NPS_Type[which(graph_3_4$Apr_14_FL.NPS_Type=="Detractor")])
value_3_4 <- c(1382,364,180)
names_3_4 <- c("Promoter", "Passive", "Detractor")
df_3_4 <- data.frame(value_3_4,names_3_4)
plot_3_4 <- ggplot(df_3_4,
aes(x=df_3_4$names_3_4,y=df_3_4$value_3_4))+geom_col()+theme(axis.text.x = element_text(angle
= 0,hjust = 1)) + xlab("NPS_Type")+ ylab("Numbers in April")+ggtitle("April")
```

```
grid.arrange(plot_3_2,plot_3_3,plot_3_4,nrow=2,ncol=2)
#Ratio of Promoter for each months
Ratio_of_Feb_P <- c/length(graph_3_2$Feb_14_FL.NPS_Type)
Ratio_of_Feb_D <- d/length(graph_3_2$Feb_14_FL.NPS_Type)
Ratio_of_Mar_P <- a/length(graph_3_3$Mar_14_FL.NPS_Type)
Ratio_of_Mar_D <- b/length(graph_3_3$Mar_14_FL.NPS_Type)
Ratio_of_Apr_P <- e/length(graph_3_4$Apr_14_FL.NPS_Type)
Ratio_of_Apr_D <- f/length(graph_3_4$Apr_14_FL.NPS_Type)
Three_month <- c(2,3,4)
Ratio_P <- c(Ratio_of_Feb_P,Ratio_of_Mar_P,Ratio_of_Apr_P)
Ratio_D <- c(Ratio_of_Feb_D,Ratio_of_Mar_D,Ratio_of_Apr_D)
line_data <- data.frame(Three_month,Ratio_P,Ratio_D)
line_graph <- ggplot(line_data, aes(x=Three_month))+
geom_line(aes(y=Ratio_P),colour="red")+geom_line(aes(y=Ratio_D),colour="blue")+
  xlab("Month")+ylab("Ratio of Promoters & Detractors")+ggtitle("NPS_Type Ratio")
line_graph
grid.arrange(plot_3_2,plot_3_3,plot_3_4,line_graph,nrow=2,ncol=2)
#ARules
dataforarule <- Quarter_FL[,c(2,4,6,11:22,28)]
dataforarule <- dataforarule[,c(-4,-7,-8,-12:-15)]
View(dataforarule)
colnames(dataforarule) <-
c("Room_Type","Length_of_Stay","Purpose_of_visit","Likelihood_Recommend",
"Overall_sat","Hotel_Condition","Customer_service","Staff_Cared","Hotel_Brand")
dataforarule$Room_Type <- as.factor(dataforarule$Room_Type)
dataforarule$Length_of_Stay <- as.factor(dataforarule$Length_of_Stay)
dataforarule$Purpose_of_visit <- as.factor(dataforarule$Purpose_of_visit)
dataforarule$Likelihood_Recommend <- as.factor(dataforarule$Likelihood_Recommend)
dataforarule$Overall_sat <- as.factor(dataforarule$Overall_sat)
dataforarule$Hotel_Condition <- as.factor(dataforarule$Hotel_Condition)
dataforarule$Customer_service <- as.factor(dataforarule$Customer_service)
dataforarule$Staff_Cared <- as.factor(dataforarule$Staff_Cared)
dataforarule$Hotel_Brand <- as.factor(dataforarule$Hotel_Brand)
mydata_1$Likelihood_Recommend_H <- as.factor(mydata_1$Likelihood_Recommend_H)
mydata_1$Guest_Room_H <- as.factor(mydata_1$Guest_Room_H)
mydata_1$Tranquility_H <- as.factor(mydata_1$Tranquility_H)
mydata_1$Condition_Hotel_H <- as.factor(mydata_1$Condition_Hotel_H)
mydata_1$Customer_SVC_H <- as.factor(mydata_1$Customer_SVC_H)
mydata_1$Staff_Cared_H <- as.factor(mydata_1$Staff_Cared_H)
mydata_1$Internet_Sat_H <- as.factor(mydata_1$Internet_Sat_H)
mydata_1$Check_In_H <- as.factor(mydata_1$Check_In_H)
mydata_1$`F&B_Overall_Experience_H` <- as.factor(mydata_1$`F&B_Overall_Experience_H`)
data_1 <- dataforarule[,c(1:4,9)]
apriori(data_1)
aruleset <- apriori(data_1, parameter = list(support=0.1, confidence=0.5))
summary(aruleset)
inspect(aruleset)
plot(aruleset)

apriori(mydata_1)
aruleset_2 <- apriori(mydata_1, parameter = list(support=0.1, confidence=0.9))
summary(aruleset_2)
inspect(aruleset_2)
plot(aruleset_2)
high_recommend_rules <- subset(aruleset_2, rhs %in% "Likelihood_Recommend_H=10")
plot(high_recommend_rules)
```

```
#from aruleset, I found that combination of POV, Length of stay and likelihood recommend is
worth to study and find the relationship between them
#Purpose of visit, length of stay and likelihood recommend
ggplot(dataforarule, aes(x=dataforarule$Length_of_Stay, y=dataforarule$Purpose_of_visit))+
  geom_tile(aes(fill=dataforarule$Likelihood_Recommend),colour="purple")+
  ggtitle("Likelihood to Recommend by purpose of visit and length of stay")+
  xlab("Length of Stay") + ylab("Purpose of Visit")
#Purpose of visit and Room type because this combination has the higher confidence value
ggplot(dataforarule, aes(Room_Type,fill=Purpose_of_visit))+
  geom_bar()+
  theme(axis.text.x = element_text(angle=90,hjust = 0.5, size = 7))+
  ggtitle("Room Type and Purpose of Visit")+
  xlab("Room Type")
#Arule for guest satisfaction metrics
data_sat <- dataforarule[,4:8]
apriori(data_sat)
arulesat <- apriori(data_sat, parameter = list(support = 0.03, confidence = 0.6))
inspect(arulesat)
#the combination of cus_svc, hotel_codi, overall_sat, and likehood_recom has the lowest supp
value (0.051)
ggplot(data_sat, aes(x=data_sat$Likelihood_Recommend,y=data_sat$Overall_sat))+
  geom_point(aes(color = Hotel_Condition,size=Customer_service))+
  ggtitle("Scatter Chart for 4 satisfaction metrics")+
  xlab("Likelihood to Recommend")+ylab("Overall Satisfaction")
#Overall_sat, hotel_codi affect likelihood recommend directly, because they have larger
confidence (>0.98)
data_line <- Quarter_FL[,c(12,13,16,32)]
meltedlinedata <- melt(data_line, id='Month')
ggplot(meltedlinedata,aes(x=Month,y=variable,fill=value))+
  geom_tile()+
  scale_fill_gradient(low = "white",high = "orange")



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


Feb<- fread("~/Downloads/out-201402.csv", select =
c(23,139,141,145,168,201,203,204,205,208,209,213,215,216,217,220,222,232))
Mar<- fread("~/Downloads/out-201403.csv", select =
c(23,139,141,145,168,201,203,204,205,208,209,213,215,216,217,220,222,232))
Apr<- fread("~/Downloads/out-201404.csv", select =
c(23,139,141,145,168,201,203,204,205,208,209,213,215,216,217,220,222,232))
Feb$Month <- "2"
Mar$Month <- "3"
Apr$Month <- "4"
Feb <- na.omit(Feb)
Mar<- na.omit(Mar)
Apr<- na.omit(Apr)
Quarter <- rbind(Feb,Mar,Apr)


#Choose California
Feb_CA <- Feb[Feb$State_PL == "California"]
Mar_CA <- Mar[Mar$State_PL == "California"]
Apr_CA<- Apr[Apr$State_PL == "California"]
Quarter_CA <- rbind(Feb_CA,Mar_CA,Apr_CA)
```

```
View(Quarter)
#choose people travel in business purpose
Quarter_CA<- subset(Quarter_CA)[Quarter_CA$POV_CODE_C == "BUSINESS"]
#devide into three dataset : promoter, passive and detractor
Quarter_CA_Promoter<-subset(Quarter_CA)[Quarter_CA$NPS_Type=="Promoter"]
Quarter_CA_Passive<-subset(Quarter_CA)[Quarter_CA$NPS_Type=="Passive"]
Quarter_CA_Detractor<-subset(Quarter_CA)[Quarter_CA$NPS_Type=="Detractor"]
#box plot for guest room, hotel condition and check in ease
library(ggplot2)
boxR1<-
ggplot(Quarter_CA_Promoter,aes(x=factor(0),y=Quarter_CA_Promoter$Guest_Room_H))+geom_boxplot(c
ol="red",outlier.color = "pink")+ggtitle(" promoters")+ylab("guest room")
boxR2<-
ggplot(Quarter_CA_Passive,aes(x=factor(0),y=Quarter_CA_Passive$Guest_Room_H))+geom_boxplot(col
="purple",outlier.color = "blue")+ggtitle("passive")+ylab("guest room")
boxR3<-
ggplot(Quarter_CA_Detractor,aes(x=factor(0),y=Quarter_CA_Detractor$Guest_Room_H))+geom_boxplot
(col="green",outlier.color = "yellow")+ggtitle("detractor")+ylab("guest room")
grid.arrange(boxR1,boxR2,boxR3,nrow=1,ncol=3)

boxH1<-
ggplot(Quarter_CA_Promoter,aes(x=factor(0),y=Quarter_CA_Promoter$Condition_Hotel_H))+geom_boxp
lot(col="red",outlier.color = "pink")+ggtitle("promoters")+ylab("condition of hotels")
boxH2<-
ggplot(Quarter_CA_Passive,aes(x=factor(0),y=Quarter_CA_Passive$Condition_Hotel_H))+geom_boxplo
t(col="purple",outlier.color = "blue")+ggtitle("passive")+ylab("condition of hotels")
boxH3<-
ggplot(Quarter_CA_Detractor,aes(x=factor(0),y=Quarter_CA_Detractor$Condition_Hotel_H))+geom_bo
xplot(col="green",outlier.color = "yellow")+ggtitle("detractor")+ylab("condition of hotels")
grid.arrange(boxH1,boxH2,boxH3,nrow=1,ncol=3)

boxC1<-
ggplot(Quarter_CA_Promoter,aes(x=factor(0),y=Quarter_CA_Promoter$Check_In_H))+geom_boxplot(col
="red",outlier.color = "pink")+ggtitle("promoters")+ylab("check in ease")
boxC2<-
ggplot(Quarter_CA_Passive,aes(x=factor(0),y=Quarter_CA_Passive$Check_In_H))+geom_boxplot(col="
purple",outlier.color = "blue")+ggtitle("passive")+ylab("check in ease")
boxC3<-
ggplot(Quarter_CA_Detractor,aes(x=factor(0),y=Quarter_CA_Detractor$Check_In_H))+geom_boxplot(c
ol="green",outlier.color = "yellow")+ggtitle("detractor")+ylab("check in ease")
grid.arrange(boxC1,boxC2,boxC3,nrow=1,ncol=3)

boxS1<-
ggplot(Quarter_CA_Promoter,aes(x=factor(0),y=Quarter_CA_Promoter$Customer_SVC_H))+geom_boxplot
(col="red",outlier.color = "pink")+ggtitle("promoters")+ylab("customer service")
boxS2<-
ggplot(Quarter_CA_Passive,aes(x=factor(0),y=Quarter_CA_Passive$Customer_SVC_H))+geom_boxplot(c
ol="purple",outlier.color = "blue")+ggtitle("passive")+ylab("customer service")
boxS3<-
ggplot(Quarter_CA_Detractor,aes(x=factor(0),y=Quarter_CA_Detractor$Customer_SVC_H))+geom_boxpl
ot(col="green",outlier.color = "yellow")+ggtitle("detractor")+ylab("customer service")
grid.arrange(boxS1,boxS2,boxS3,nrow=1,ncol=3)

boxF1<-
ggplot(Quarter_CA_Promoter,aes(x=factor(0),y=Quarter_CA_Promoter$Staff_Cared_H))+geom_boxplot(
col="red",outlier.color = "pink")+ggtitle("promoters")+ylab("staff cared")
```

```
boxF2<-
ggplot(Quarter_CA_Passive,aes(x=factor(0),y=Quarter_CA_Passive$Staff_Cared_H))+geom_boxplot(co
l="purple",outlier.color = "blue")+ggtitle("passive")+ylab("staff cared")
boxF3<-
ggplot(Quarter_CA_Detractor,aes(x=factor(0),y=Quarter_CA_Detractor$Staff_Cared_H))+geom_boxplo
t(col="green",outlier.color = "yellow")+ggtitle("detractor")+ylab("staff cared")
grid.arrange(boxF1,boxF2,boxF3,nrow=1,ncol=3)

#bar chart for boutique, convention, fitness trainer, limo service, indoor pool, outdoor pool,
regency grand club and self parking
library(reshape2)
Quarter_CA <- Quarter_CA[!(Quarter_CA$`Fitness Center_PL`=="" |
Quarter_CA$NPS_Type==""|Quarter_CA$`Fitness Trainer_PL`=="")]
g1<-ggplot(Quarter_CA, aes(x=Boutique_PL, y=NPS_Type)) + geom_bar(aes(fill=NPS_Type), stat =
"identity")+ggtitle("NPS type distribution on Boutique")+theme(axis.text.y = element_blank())
g2<-ggplot(Quarter_CA, aes(x=Convention_PL, y=NPS_Type)) + geom_bar(aes(fill=NPS_Type), stat =
"identity")+ggtitle("NPS type distribution on Convention")+theme(axis.text.y =
element_blank())
g3<-ggplot(Quarter_CA, aes(x=`Fitness Trainer_PL`, y=NPS_Type)) + geom_bar(aes(fill=NPS_Type),
stat = "identity")+ggtitle("NPS type distribution on fitness trainer")+theme(axis.text.y =
element_blank())
g4<-ggplot(Quarter_CA, aes(x=`Limo Service_PL`, y=NPS_Type)) + geom_bar(aes(fill=NPS_Type),
stat = "identity")+ggtitle("NPS type distribution on Limo Service")+theme(axis.text.y =
element_blank())
g5<-ggplot(Quarter_CA, aes(x=`Pool-Indoor_PL`, y=NPS_Type)) + geom_bar(aes(fill=NPS_Type),
stat = "identity")+ggtitle("NPS type distribution on indoor-pool")+theme(axis.text.y =
element_blank())
g6<-ggplot(Quarter_CA, aes(x=`Pool-Outdoor_PL`, y=NPS_Type)) + geom_bar(aes(fill=NPS_Type),
stat = "identity")+ggtitle("NPS type distribution on outdoor-pool")+theme(axis.text.y =
element_blank())
g7<-ggplot(Quarter_CA, aes(x=`Regency Grand Club_PL`, y=NPS_Type)) +
geom_bar(aes(fill=NPS_Type), stat = "identity")+ggtitle("NPS type distribution on regency
grand club")+theme(axis.text.y = element_blank())
g8<-ggplot(Quarter_CA, aes(x=`Self-Parking_PL`, y=NPS_Type)) + geom_bar(aes(fill=NPS_Type),
stat = "identity")+ggtitle("NPS type distribution on self parking")+theme(axis.text.y =
element_blank())
grid.arrange(g1,g2,g3,g4,nrow=2,ncol=2)
grid.arrange(g5,g6,g7,g8,nrow=2,ncol=2)
```

## Descriptive Statistics and Linear Modelling

```
#libraries used
library(data.table)
library(stats)
library(zipcode)
library(kernlab)
library(datasets)
library(ggplot2)
library(sqldf)
library(gridExtra)
library(ggmap)
library(arules)
library(arulesViz)
library(plotrix)

memory.limit(size=12000)
```

```
#considering data for feb,march, april -- 1st quarter
dat_feb14 <- fread("C:/Users/Tushar/Desktop/IST687-data/out-201402.csv", select =
c(12,19,28,54,67,106:110,137:147,167:171,175,176,179,182,191,199:227,232))
dat_feb14_2<- data.frame(dat_feb14,stringsAsFactors = FALSE)
dat_feb14_2<- na.omit(dat_feb14_2)
feb14<-dat_feb14_2
feb14<-feb14[(feb14$NPS_Type=="Promoter" | feb14$NPS_Type=="Detractor" |
feb14$NPS_Type=="Passive")&& feb14$Likelihood_Recommend_H>0,]
feb14.1<- feb14[,-c(32:60)]
feb14.1$Month<-"February 2014"

dat_mar14 <- fread("C:/Users/Tushar/Desktop/IST687-data/out-201403.csv", select =
c(12,19,28,54,67,106:110,137:147,167:171,175,176,179,182,191,199:227,232))
dat_mar14_2<- data.frame(dat_mar14,stringsAsFactors = FALSE)
dat_mar14_2<- na.omit(dat_mar14_2)
mar14<-dat_mar14_2
mar14<-mar14[(mar14$NPS_Type=="Promoter" | mar14$NPS_Type=="Detractor" |
mar14$NPS_Type=="Passive")&& mar14$Likelihood_Recommend_H>0,]
mar14.1<- mar14[,-c(32:60)]
mar14.1$Month<-"March 2014"

dat_apr14 <- fread("C:/Users/Tushar/Desktop/IST687-data/out-201404.csv", select =
c(12,19,28,54,67,106:110,137:147,167:171,175,176,179,182,191,199:227,232))
dat_apr14_2<- data.frame(dat_apr14,stringsAsFactors = FALSE)
dat_apr14_2<- na.omit(dat_apr14_2)
apr14<-dat_apr14_2
apr14<-apr14[(apr14$NPS_Type=="Promoter" | apr14$NPS_Type=="Detractor" |
apr14$NPS_Type=="Detractor")&& apr14$Likelihood_Recommend_H>0,]
apr14.1<- apr14[,-c(32:60)]
apr14.1$Month<-"April 2014"

quarter<- rbind(feb14,mar14,apr14)
names_col<- colnames(quarter)
names_col
quarter1<- rbind(feb14.1,mar14.1,apr14.1)
row.names(quarter1)<- NULL
quarter1<-quarter1[,-c(20,24,31)]
colnames(quarter1)<-
c("RoomType","LengthOfStay","HotelRevenue","NightlyRate","MemberStatus","GuestState","GuestCou
ntry","GuestGender","GuestAgeRange","PurposeOfVisit","LikelihoodToRecommend_SV","OverallSatisf
action_SV","Room_SV","Tranquility_SV","HotelCondition_SV","CustomerService_SV","StaffCare_SV",
"Internet_SV","CheckInEase_SV","F.B_SV","City","State","Zipcode","Country","Latitude","Longitu
de","NPS_Goal","HotelBrand","NPS_Type","Month")
View(quarter1)

# worldmap
world<- borders("world",colour = "gray50",fill = "gray50")
locationplot<- ggplot()+world+geom_point(aes(x=quarter1$Longitude, y = quarter1$Latitude),
color = "blue", size = 1)
locationplot+labs(y="Latitude",x="Longitude",title= "Hotel locations")

#US map
map<-get_map(location='united states', zoom=4, maptype= "terrain", source='google',
color='bw')
# Map from URL :
http://maps.googleapis.com/maps/api/staticmap?center=united+states&zoom=4&size=640x640&scale=2
&maptype=terrain&language=en-EN&sensor=false
```

```
map2 <- ggmap(map) + geom_point(aes(x=Longitude, y = Latitude, color =
LikelihoodToRecommend_SV, size = 1.5), data = quarter1)+ scale_color_gradient(low= "blue",
high = "red")+labs(y="Latitude",x="Longitude",color= "LTR")
map2
#########################
#########################
#########################

# US country data
us_coun_hotels<- quarter1[quarter1$Country=="United States" & quarter1$HotelRevenue>0 &
(quarter1$GuestGender=="Female" | quarter1$GuestGender=="Male" | quarter1$GuestGender=="Prefer
not to answer"),]
us_coun_hotels$NightlyRate<-as.numeric(us_coun_hotels$NightlyRate)
us_coun_hotels$NightlyRate<- gsub("*","",us_coun_hotels$NightlyRate)
us_coun_hotels$NightlyRate<-round(us_coun_hotels$NightlyRate)
us_coun_hotels$HotelRevenue<-round(us_coun_hotels$HotelRevenue)
View(us_coun_hotels)


# nps calculation --> why Cali & Florida?
nps_st_pr<- sqldf("SELECT State,NPS_Type,COUNT(NPS_Type) AS 'Ctr1' FROM us_coun_hotels WHERE
NPS_Type IS 'Promoter' GROUP BY State ORDER BY State ASC")
nps_st_de<-sqldf("SELECT State,NPS_Type,COUNT(NPS_Type) AS 'Ctr2' FROM us_coun_hotels WHERE
NPS_Type IS 'Detractor' GROUP BY State ORDER BY State ASC")
nps_st_pa<-sqldf("SELECT State,NPS_Type,COUNT(NPS_Type) AS 'Ctr3' FROM us_coun_hotels WHERE
NPS_Type IS 'Passive' GROUP BY State ORDER BY State ASC")

nps_st<- data.frame(c(nps_st_pr,nps_st_de,nps_st_pa))
nps_st<- nps_st[,-c(4,7)]
nps_st$NumOfResponses<- as.numeric(nps_st$Ctr1+nps_st$Ctr2+nps_st$Ctr3)
nps_st$NPS_value_percentage <- round(((nps_st$Ctr1- nps_st$Ctr2)/(nps_st$Ctr1+ nps_st$Ctr2+
nps_st$Ctr3))*100)
nps_st<- nps_st[order(-nps_st$NumOfResponses),]
nps_st$detr_pr<- round(((nps_st$Ctr2)/(nps_st$Ctr1+ nps_st$Ctr2+ nps_st$Ctr3))*100)
row.names(nps_st)<- NULL
View(nps_st)

ggplot(nps_st, aes(x=NPS_value_percentage, y=NumOfResponses)) + geom_line(color="red") +
ggtitle("NPS v/s Responses") + labs(y="Number of responses",x="NPS %")+theme_bw()
gbar0<- ggplot(nps_st, aes(x=nps_st$State, y=nps_st$NumOfResponses)) + geom_bar(aes(fill =
nps_st$NPS_value_percentage), stat = "identity") + xlab('States') + ylab('Number of
responses') + guides(fill=guide_legend(title="NPS Value"))+ggtitle('NPS based on state')
gbar0+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))

gbar0<- ggplot(nps_st, aes(x=nps_st$State, y=nps_st$NumOfResponses)) + geom_bar(aes(fill =
nps_st$NPS_value_percentage), stat = "identity") + xlab('States') + ylab('Number of
responses') + guides(fill=guide_legend(title="NPS Value"))+ggtitle('NPS based on state')
gbar0+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))

#nps_st <- nps_st[order(-nps_st$detr_pr), ]
nps_st$State<-reorder(nps_st$State,-nps_st$detr_pr)
gbar9<- ggplot(nps_st, aes(x=nps_st$State, y=nps_st$detr_pr)) + geom_bar(aes(fill =
nps_st$NPS_value_percentage), stat = "identity") + xlab('States') + ylab('Detractor %') +
guides(fill=guide_legend(title="NPS Score"))+ggtitle('Detractor% based on state')
gbar9+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))

#################
```

```
#################
#################
svy<- quarter1[quarter1$Country=="United States" & quarter1$HotelRevenue>0 &
(quarter1$GuestGender=="Female" | quarter1$GuestGender=="Male" | quarter1$GuestGender=="Prefer
not to answer") & quarter1$State=="California" & quarter1$PurposeOfVisit=="Business",]
svy_p<- svy[svy$NPS_Type=="Promoter",]
svy_d<- svy[svy$NPS_Type=="Detractor",]
########promoters
svy_p<- svy_p[,c(11,13:20)]
row.names(svy_p)<- NULL
View(svy_p)
svy_p$Room_SV<- as.factor(as.character(svy_p$Room_SV))
svy_p$HotelCondition_SV<- as.factor(as.character(svy_p$HotelCondition_SV))
svy_p$Tranquility_SV<- as.factor(as.character(svy_p$Tranquility_SV))
svy_p$CheckInEase_SV<- as.factor(as.character(svy_p$CheckInEase_SV))
svy_p$CustomerService_SV<- as.factor(as.character(svy_p$CustomerService_SV))
svy_p$StaffCare_SV<- as.factor(as.character(svy_p$StaffCare_SV))
svy_p$Internet_SV<- as.factor(as.character(svy_p$Internet_SV))
svy_p$F.B_SV<- as.factor(as.character(svy_p$F.B_SV))
svy_p$LikelihoodToRecommend_SV<- as.factor(as.character(svy_p$LikelihoodToRecommend_SV))
rule1<-apriori(svy_p,parameter = list(support= 0.4 ,confidence= 0.8))
inspect(rule1)
rules2 <- subset(rule1, rhs %in% "LikelihoodToRecommend_SV=10")
inspect(rules2)
# {Room_SV=10,HotelCondition_SV=10,CustomerService_SV=10,StaffCare_SV=10}       =>
{LikelihoodToRecommend_SV=10} 0.4087673  0.9145013 1.461559
plot(rule1)
plot (rules2,method="graph",interactive=TRUE,shading="lift")
plot(rules2, method="graph", control=list(type="items"))


#linear modelling for promoters for all columns
svy_pp<- svy[svy$NPS_Type=="Promoter",]
svy_pp<-svy_pp[,c(11,13:20)]
row.names(svy_pp)<- NULL
a_1 <- lm(formula = svy_pp$LikelihoodToRecommend_SV ~ svy_pp$Room_SV, data = svy_pp)
a<-summary(a_1)$adj.r.squared*100
a_2 <- lm(formula = svy_pp$LikelihoodToRecommend_SV ~ svy_pp$CustomerService_SV, data =
svy_pp)
b<-summary(a_2)$adj.r.squared*100
a_3 <- lm(formula = svy_pp$LikelihoodToRecommend_SV ~ svy_pp$HotelCondition_SV, data = svy_pp)
c<-summary(a_3)$adj.r.squared*100
a_4 <- lm(formula = svy_pp$LikelihoodToRecommend_SV ~ svy_pp$StaffCare_SV, data = svy_pp)
d<-summary(a_4)$adj.r.squared*100
a_5 <- lm(formula = svy_pp$LikelihoodToRecommend_SV ~ svy_pp$Tranquility_SV, data = svy_pp)
e<-summary(a_5)$adj.r.squared*100
a_6 <- lm(formula = svy_pp$LikelihoodToRecommend_SV ~ svy_pp$Internet_SV, data = svy_pp)
f<-summary(a_6)$adj.r.squared*100
a_7 <- lm(formula = svy_pp$LikelihoodToRecommend_SV ~ svy_pp$CheckInEase_SV, data = svy_pp)
g<-summary(a_7)$adj.r.squared*100
a_7 <- lm(formula = svy_pp$LikelihoodToRecommend_SV ~ svy_pp$F.B_SV, data = svy_pp)
h<-summary(a_7)$adj.r.squared*100
rsvalue1 <- c(a,b,c,d,e,f,g,h)
names1 <- c("Guest Room","Customer Service","Hotel Condition","Staff
Care","Tranquility","Internet","Check-in","F&B")
graph.1 <- data.frame(names1,rsvalue1)
graph.1$names1<-reorder(graph.1$names1,-graph.1$rsvalue1)
```

```
plot.1 <- ggplot(graph.1, aes(x=graph.1$names,y=graph.1$rsvalue))+geom_col()+theme(axis.text.x
= element_text(angle = 90,hjust = 1))+xlab("Satisfaction Metrics Names")+ylab("Adj.R.squared
value")+ggtitle("Adjusted R Squared Value for each Survey element")
plot.1
#According to Adj.R-squared value, the most reason that influence Likelihood_to_recommend is
Customer service & Internet


#Modelling with combinations of survey -  Most parsimonious model
model_al <- lm(formula = LikelihoodToRecommend_SV~. ,data=svy_pp)
step(model_al, data=svy_pp, direction = "backward")
#AIC value is -4981.16

model_lowest_AIC <- lm(formula = LikelihoodToRecommend_SV~Room_SV + Tranquility_SV +
HotelCondition_SV + CustomerService_SV + StaffCare_SV + Internet_SV + CheckInEase_SV + F.B_SV,
data = svy_pp)
summary(model_lowest_AIC)
summary(model_lowest_AIC)$adj.r.squared
#Adjusted-R-Squared value is 0.342 which is the most parsimonious model


# based on a rule result, linear modelling for promoters for those 4 columns
svy_ppp<- svy[svy$NPS_Type=="Promoter",]
View(svy_ppp)
svy_ppp<-svy_ppp[,c(11,13,15:17)]
row.names(svy_ppp)<- NULL
b_1 <- lm(formula = svy_ppp$LikelihoodToRecommend_SV ~ svy_ppp$Room_SV, data = svy_ppp)
z<-summary(b_1)$adj.r.squared*100
b_2 <- lm(formula = svy_ppp$LikelihoodToRecommend_SV ~ svy_ppp$CustomerService_SV, data =
svy_ppp)
y<-summary(b_2)$adj.r.squared*100
b_3 <- lm(formula = svy_ppp$LikelihoodToRecommend_SV ~ svy_ppp$HotelCondition_SV, data =
svy_ppp)
x<-summary(b_3)$adj.r.squared*100
b_4 <- lm(formula = svy_ppp$LikelihoodToRecommend_SV ~ svy_ppp$StaffCare_SV, data = svy_ppp)
w<-summary(b_4)$adj.r.squared*100
rsvalue2 <- c(z,y,x,w)
names2 <- c("Guest Room","Customer Service","Hotel Condition","Staff Care")
graph.2 <- data.frame(names2,rsvalue2)
graph.2$names2<-reorder(graph.2$names2,-graph.2$rsvalue2)
plot.2 <- ggplot(graph.2, aes(x=graph.2$names,y=graph.2$rsvalue))+geom_col()+theme(axis.text.x
= element_text(angle = 90,hjust = 1))+xlab("Satisfaction Metrics Names")+ylab("Adj.R.squared
value")+ggtitle("Adjusted R Squared Value for each Survey element")
plot.2
#According to Adj.R-squared value, the most reason that influence Likelihood_to_recommend is
Customer service


#Modelling with combinations of survey -  Most parsimonious model
model_al2 <- lm(formula = LikelihoodToRecommend_SV~. ,data=svy_ppp)
step(model_al2, data=svy_ppp, direction = "backward")
#AIC value is -4964.94

model_lowest_AIC2 <- lm(formula = LikelihoodToRecommend_SV~Room_SV + HotelCondition_SV +
CustomerService_SV + StaffCare_SV, data = svy_ppp)
summary(model_lowest_AIC2)
summary(model_lowest_AIC2)$adj.r.squared
```

```
#Adjusted-R-Squared value is 0.312 which is the most parsimonious model

#difference b/w 2 parsimonious models is 3% which is pretty small. Hence, best 4 columns from both a rules data mining & modelling are : Room_SV, HotelCondition_SV, CustomerService_SV, StaffCare_SV
#####################

##################
temp_survey1<- quarter1[quarter1$Country=="United States" & quarter1$HotelRevenue>0 & (quarter1$GuestGender=="Female" | quarter1$GuestGender=="Male" | quarter1$GuestGender=="Prefer not to answer") & quarter1$State=="California" & quarter1$PurposeOfVisit=="Business" & quarter$NPS_Type=="Promoter",]
temp_survey<- temp_survey1[,c(12:20)]
row.names(temp_survey)<- NULL

temp_survey[temp_survey>="9"] <- "High"
temp_survey[temp_survey>="7" & temp_survey<"9"] <- "Medium"
temp_survey[temp_survey<"7"] <- "Low"
row.names(temp_survey)<- NULL

temp_amenity<- quarter[quarter$Country_PL=="United States" & quarter$PMS_TOTAL_REV_USD_C>0 & (quarter$Gender_H=="Female" | quarter$Gender_H=="Male" | quarter$Gender_H=="Prefer not to answer") & quarter$State_PL=="California" & quarter$POV_H=="Business" & quarter$NPS_Type=="Promoter",]
row.names(temp_amenity)<- NULL
temp_arul<- temp_amenity[,c(32:61)]
dummy1<-cbind(temp_survey,temp_arul)
View(dummy1)

head(dummy1,3)

# primary: Guest_Room_H + Condition_Hotel_H + Customer_SVC_H + Staff_Cared_H + Internet_Sat_H + Check_In_H + F.B_Overall_Experience_H
# to determine rules wrt NPS type
part1<- dummy1[,c(2:9,39)]
part1$Room_SV<- as.factor(as.character(part1$Room_SV))
part1$HotelCondition_SV<- as.factor(as.character(part1$HotelCondition_SV))
part1$Tranquility_SV<- as.factor(as.character(part1$Tranquility_SV))
part1$CheckInEase_SV<- as.factor(as.character(part1$CheckInEase_SV))
part1$CustomerService_SV<- as.factor(as.character(part1$CustomerService_SV))
part1$StaffCare_SV<- as.factor(as.character(part1$StaffCare_SV))
part1$Internet_SV<- as.factor(as.character(part1$Internet_SV))
part1$F.B_SV<- as.factor(as.character(part1$F.B_SV))
part1$NPS_Type<- as.factor(as.character(part1$NPS_Type))
ruleset1<-apriori(part1,parameter = list(support= 0.4 ,confidence= 0.8))
plot(ruleset1)
rulesets2 <- subset(ruleset1, rhs %in% "NPS_Type=Promoter")
inspect(rulesets2)
plot (ruleset1,method="graph",interactive=TRUE,shading="lift")
plot(rulesets2, method="paracoord", control=list(type="items"))
#{Room_SV=Low,HotelCondition_SV=Low,CustomerService_SV=Low,StaffCare_SV=Low}       => {NPS_Type=Promoter} 0.4518546          1    1

part2<- dummy1[,c(10:39)] #CONSIDERING COLUMNS RELEVANT FOR BUSINESS USERS
#SPA & FITNESS
View(part2)
part2.s<- part2[,c(3,10,11,17,18,25,26,30)]
```

```
part2.s$Boutique_PL<- as.factor(as.character(part2.s$Boutique_PL))
part2.s$Fitness.Center_PL<-as.factor(as.character(part2.s$Fitness.Center_PL))
part2.s$Fitness.Trainer_PL<-as.factor(as.character(part2.s$Fitness.Trainer_PL))
part2.s$Pool.Indoor_PL<-as.factor(as.character(part2.s$Pool.Indoor_PL))
part2.s$Pool.Outdoor_PL<-as.factor(as.character(part2.s$Pool.Outdoor_PL))
part2.s$Spa_PL<-as.factor(as.character(part2.s$Spa_PL))
part2.s$Spa.services.in.fitness.center_PL<-
as.factor(as.character(part2.s$Spa.services.in.fitness.center_PL))
part2.s$NPS_Type<- as.factor(as.character(part2.s$NPS_Type))
ruleset2<-apriori(part2.s,parameter = list(support= 0.7 ,confidence= 0.9))
inspect(ruleset2)
good2.s<-subset(ruleset2, rhs %in% "NPS_Type=Promoter")
inspect(good2.s)
plot(ruleset2)
plot(good2.s, method="paracoord", control=list(reorder=TRUE))
#{Boutique_PL=N,Fitness.Center_PL=Y,Fitness.Trainer_PL=N,Pool.Indoor_PL=N,Pool.Outdoor_PL=Y}
=> {NPS_Type=Promoter} 0.6614839          1    1

#VEHICLE ARRANGEMENT
part2.v<- part2[,c(15,22,23,29,30)]
View(part2)
part2.v$Limo.Service_PL<-as.factor(as.character(part2.v$Limo.Service_PL))
part2.v$Valet.Parking_PL<-as.factor(as.character(part2.v$Valet.Parking_PL))
part2.v$Shuttle.Service_PL<-as.factor(as.character(part2.v$Shuttle.Service_PL))
part2.v$Self.Parking_PL<-as.factor(as.character(part2.v$Self.Parking_PL))
part2.v$NPS_Type<- as.factor(as.character(part2.v$NPS_Type))
ruleset3<-apriori(part2.v,parameter = list(support= 0.1 ,confidence= 0.6))
inspect(ruleset3)
good2.v<-subset(ruleset3, rhs %in% "NPS_Type=Promoter")
inspect(good2.v)
plot(ruleset3)
plot(good2.v, method="graph", control=list(type="items"))
#{Limo.Service_PL=Y, Self.Parking_PL=Y, Shuttle.Service_PL=Y,Valet.Parking_PL=Y}   =>
{NPS_Type=Promoter} 0.1330086          1    1


# BUSINESS ORDEALS
part2.b<- part2[,-c(3,10,11,12,17,18,20,25,26)]
View(part2.b)
part2.b<- part2.b[,c(3,5,6,13,21)]
part2.b$Business.Center_PL<-as.factor(as.character(part2.b$Business.Center_PL))
part2.b$Conference_PL<-as.factor(as.character(part2.b$Conference_PL))
part2.b$Convention_PL<-as.factor(as.character(part2.b$Convention_PL))
part2.b$Regency.Grand.Club_PL<-as.factor(as.character(part2.b$Regency.Grand.Club_PL))
part2.b$NPS_Type<- as.factor(as.character(part2.b$NPS_Type))
ruleset4<-apriori(part2.b,parameter = list(support= 0.1 ,confidence= 0.7))
inspect(ruleset4)
good2.b<-subset(ruleset4, rhs %in% "NPS_Type=Promoter")
inspect(good2.b)
plot(ruleset4)
plot(good2.b, method="graph", control=list(type="items"))
# {Business.Center_PL=Y,Conference_PL=N,Convention_PL=Y,Regency.Grand.Club_PL=N} =>
{NPS_Type=Promoter}


# ROOM, HOTEL & OTHER F&B PROVISIONS
part2.r<- part2[-c(3,4,6,7,10,11,17:19,25:29)]
```

```
View(part2.r)
part2.r<- part2.r[,-c(9,13,14)]
part2.r$All.Suites_PL<- as.factor(as.character(part2.r$All.Suites_PL))
part2.r$Bell.Staff_PL<- as.factor(as.character(part2.r$Bell.Staff_PL))
part2.r$Casino_PL<-as.factor(as.character(part2.r$Casino_PL))
part2.r$Dry.Cleaning_PL<-as.factor(as.character(part2.r$Dry.Cleaning_PL))
part2.r$Elevators_PL<-as.factor(as.character(part2.r$Elevators_PL))
part2.r$Golf_PL<-as.factor(as.character(part2.r$Golf_PL))
part2.r$Indoor.Corridors_PL<-as.factor(as.character(part2.r$Indoor.Corridors_PL))
part2.r$Laundry_PL<-as.factor(as.character(part2.r$Laundry_PL))
part2.r$Resort_PL<-as.factor(as.character(part2.r$Resort_PL))
part2.r$Restaurant_PL<-as.factor(as.character(part2.r$Restaurant_PL))
part2.r$Ski_PL<-as.factor(as.character(part2.r$Ski_PL))
part2.r$Mini.Bar_PL<-as.factor(as.character(part2.r$Mini.Bar_PL))
part2.r$NPS_Type<- as.factor(as.character(part2.r$NPS_Type))
ruleset5<-apriori(part2.r,parameter = list(support= 0.8 ,confidence= 0.9))
good2.5<-subset(ruleset5, rhs %in% "NPS_Type=Promoter")
inspect(good2.5)
plot(ruleset5)
plot(good2.5, method="paracoord", control=list(type="items"))
# {Casino_PL=N,Dry.Cleaning_PL=Y,Golf_PL=N,Resort_PL=N,Ski_PL=N}              =>
{NPS_Type=Promoter} 0.8085425          1    1

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

#BEST hotel for leisure POV based on number of hotels
US_htl<- us_coun_hotels[us_coun_hotels$PurposeOfVisit=="Leisure" &
us_coun_hotels$State=="California",]
row.names(US_htl)<- NULL
brand2<- sqldf("select HotelBrand,COUNT(HotelBrand) AS 'Totals' FROM US_htl GROUP BY
HotelBrand ORDER BY Totals DESC")
View(brand2)
pie3D(brand2$Totals,labels=brand2$HotelBrand,explode=0.2,main="Pie Chart of Hotel Brands-
Leisure")


#CALI+business
#BEST hotel for business POV based on number of hotels
US_hyb<- us_coun_hotels[us_coun_hotels$PurposeOfVisit=="Business" &
us_coun_hotels$State=="California",]
row.names(US_hyb)<- NULL
brand<- sqldf("select HotelBrand,COUNT(HotelBrand) AS 'Total' FROM US_hyb GROUP BY HotelBrand
ORDER BY Total DESC")
View(brand)
pie3D(brand$Total,labels=brand$HotelBrand,explode=0.2,main="Pie Chart of Hotel Brands-
Business ")

#which age grp is LTR & for which hotel brand?
gbar1<- ggplot(US_hyb, aes(x=US_hyb$GuestAgeRange, y=US_hyb$LikelihoodToRecommend_SV)) +
geom_bar(aes(fill = US_hyb$HotelBrand), stat = "identity") + xlab('Age') + ylab('LTR') +
guides(fill=guide_legend(title="Hotel Brand"))+ggtitle('LTR based on Age')
gbar1+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))

#count of number of hotels visited for business.
st<- sqldf("select HotelBrand,count(HotelBrand) AS 'Count' from US_hyb GROUP BY HotelBrand")
```

```
ggplot(st,aes(HotelBrand,Count))+
geom_bar(stat="identity")+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))+ggtitle(
"Number of hotels in a brand visited by business customers")


#regency is the best for business.Why?
#where do promoters or detractors prefer to stay for long for business visits, in the US?
us_hot<- us_coun_hotels[us_coun_hotels$PurposeOfVisit=="Business",]
gbar1<- ggplot(us_hot, aes(x=us_hot$HotelBrand, y=us_hot$LengthOfStay)) + geom_bar(aes(fill =
us_hot$NPS_Type), stat = "identity") + xlab('Hotel Brand') + ylab('Length of Stay') +
ggtitle("Which hotel brand do the promoters reside in the longest?")+
guides(fill=guide_legend(title="NPS Type"))
gbar1+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))


#california + regency + business
# For Hyatt Regency, with POV as business
hyareg_d<- US_hyb[US_hyb$HotelBrand=="Hyatt Regency",]
row.names(hyareg_d)<- NULL
hyareg<- hyareg_d[,-c(10,19:22,26,27)]
View(hyareg)

#dataspread across the months
ggplot(hyareg,aes(Month,LengthOfStay))+
geom_bar(stat="identity",col="red",fill="green")+ggtitle("Monthly spread based on length of
stay")+theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))
# length of stay was highest in March

#which room type grossed highest average revenue/day & why?
hya<- hyareg[hyareg$Month=="March 2014",]
table(hya$RoomType)

#Guest Room Double,Guest Room King, Guest Room Queen/Queen, Guest Room Double/Double, High
Floor King, Bayview Balcony King
#considering the 6 most booked room types in March for this hotel brand for business purposes
subset<- hya[,c(1,2:5,8:10,17,21)]
subset<- subset[subset$RoomType=="Guest Room Double" | subset$RoomType=="Guest Room King" |
subset$RoomType=="Guest Room Queen/Queen" | subset$RoomType=="Guest Room Double/Double" |
subset$RoomType=="High Floor King" | subset$RoomType=="Bayview Balcony King",]

row.names(subset)<- NULL
View(subset)


# scatterplot to determine which age group stays for longer with a higher LTR and what type of
rooms do they prefer?
ggplot(subset,aes(LengthOfStay,GuestAgeRange))+geom_point(aes(shape=RoomType,color=LikelihoodT
oRecommend_SV))+scale_colour_gradient(low = "yellow", high = "dark
blue")+labs(title="SCATTERPLOT", y="Age Range", x="Length of Stay")+theme_bw()

# scatterplot to determine how length of stay is affected by nightly rate, and what is the LTR
by various age groups who have arrived for business visits?
sub<- subset
sub<- sub[sub$GuestAgeRange!="18-25" & sub$GuestAgeRange!="66-75" & sub$GuestAgeRange!="76+" &
sub$GuestAgeRange!="", ]
```

```
ggplot(sub,aes(LengthOfStay,NightlyRate))+geom_point(aes(shape=GuestAgeRange,color=LikelihoodT
oRecommend_SV))+scale_colour_gradient(low = "green", high = "red")+labs(title="SCATTERPLOT",
y="Nightly Rate", x="Length of Stay")+theme_bw()
```

## KSVM and NB

```
#Importing the required packages
library(data.table)

#Reading selective columns
feb2014 <- fread("C:/Users/dj_k9/Documents/Syracuse University/IST 687 - Applied Data
Science/Project/Data Set/out-201402.csv", select = c(162, 171, 168, 121, 179, 232, 133, 108,
110, 137:145, 147, 175, 176))
mar2014 <- fread("C:/Users/dj_k9/Documents/Syracuse University/IST 687 - Applied Data
Science/Project/Data Set/out-201403.csv", select = c(162, 171, 168, 121, 179, 232, 133, 108,
110, 137:145, 147, 175, 176))
apr2014 <- fread("C:/Users/dj_k9/Documents/Syracuse University/IST 687 - Applied Data
Science/Project/Data Set/out-201404.csv", select = c(162, 171, 168, 121, 179, 232, 133, 108,
110, 137:145, 147, 175, 176))


#Removing data from all countries other than the USA
usefulData <- mergedData[mergedData$Country_PL == 'United States', ]

#Removing incomplete surveys
usefulData <- usefulData[usefulData$Status_H == 'COMPLETED', ]

#Complete the state data code

#Selecting only data from California
usefulData <- usefulData[usefulData$State_PL == "California", ]

#Selecting reviews where the purpose of visit was "Business"
usefulData <- usefulData[usefulData$POV_H == 'Business']

#Importing the required packages
install.packages("kernlab")
library(kernlab)

#Useful variables as a result of LM:
#139 - Guest room satisfaction metric
#141 - Condition of hotel metric
#142 - Quality of customer service metric
#143 - Staff cared metric

#Reading the data
ksvmData <- usefulData[, c(2, 4, 5, 6, 19)]
ksvmData <- na.omit(ksvmData)

#All NAs have been omitted as we felt that substituting them with averages
#was not an ethical action to take as it may not be a true representation
#of the customer's opinion.

#Removing data where NPS type has not been assigned
ksvmData <- ksvmData[!(ksvmData$NPS_Type == ""), ]
```

```
#Creating a training set
randIndices <- sample(1:dim(ksvmData)[1])                #Generating the random indices
twoThirds <- floor(2 * dim(ksvmData)[1]/3)               #Calculating the index position at 2
thirds
trainingSet <- ksvmData[randIndices[1:twoThirds], ]      #Assigning values to the training
set

#Creating a testing set
testingSet <- ksvmData[randIndices[(twoThirds + 1):dim(ksvmData)[1]], ]

#Building the model
ksvmModel <- ksvm(NPS_Type ~ ., data = trainingSet, kernel = "rbfdot")

#Testing the model
testModel <- predict(ksvmModel, testingSet)

#Comparing results
compTable <- data.frame(testingSet[, "NPS_Type"], testModel)
table(compTable)

#Importing the required packages
install.packages("e1071")
library(e1071)

#Useful variables as a result of LM:
#139 - Guest room satisfaction metric
#141 - Condition of hotel metric
#142 - Quality of customer service metric
#143 - Staff cared metric

#Reading the data
nbData <- usefulData[, c(2, 4, 5, 6, 19)]
nbData <- na.omit(nbData)

#All NAs have been omitted as we felt that substituting them with averages
#was not an ethical action to take as it may not be a true representation
#of the customer's opinion.

#Removing data where NPS type has not been assigned
nbData <- nbData[!(nbData$NPS_Type == ""), ]

#Converting NPS type to factor
nbData$NPS_Type <- as.factor(nbData$NPS_Type)

#Creating a training set
randIndices <- sample(1:dim(nbData)[1])                #Generating the random indices
twoThirds <- floor(2 * dim(nbData)[1]/3)               #Calculating the index position at 2
thirds
nbTrainingSet <- nbData[randIndices[1:twoThirds], ]      #Assigning values to the training
set

#Creating a testing set
nbTestingSet <- nbData[randIndices[(twoThirds + 1):dim(nbData)[1]], ]

#Building the model
nbModel <- naiveBayes(NPS_Type ~ ., data = nbTrainingSet, na.action = na.omit)
```

```
#Testing the model
testModel <- predict(nbModel, nbTestingSet)

#Comparing results
compTable <- data.frame(nbTestingSet[, "NPS_Type"], testModel)
table(compTable)
```