INCREASING THE NPS SCORE FOR HYATT HOTELS CHINA: INSIGHTS & RECOMMENDATIONS

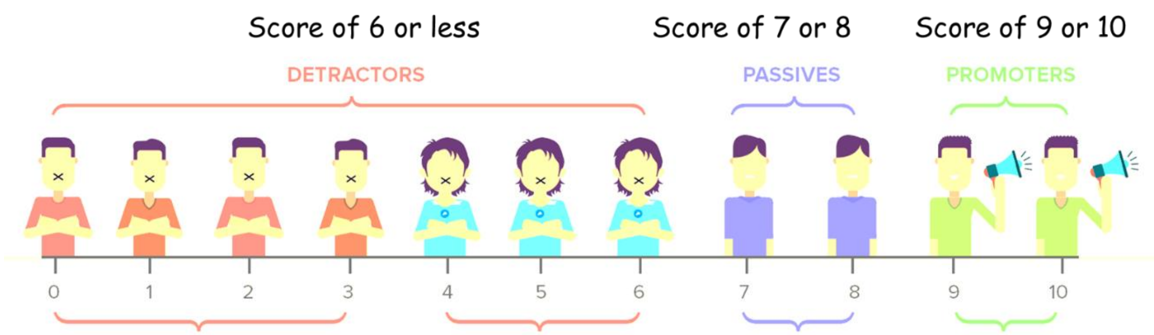
Rucha Kadam, Yuejia Peng, Mahnaz Valipour & Najaf Zahra

Table of Contents

Introduction and Problem Analysis

Hyatt Hotels Corporation is an American multinational owner, operator, and franchiser of hotels, resorts, and vacation properties. Being in the service industry with intense competition from other hotel chains requires hotels to continuously retain customers and minimize churn. Retention of existing customers and attracting more is dependent on the ability of the hotel chain to keep its customers happy. For this reason, hotel chains like Hyatt hotels value customer loyalty metrics for various reasons. Firstly, it is a powerful and effective technique, which can greatly increase a company's revenue if used properly. Predictions can be made on the kind of customers visiting the hotel and other factors having a direct or indirect correlation with the hotel growth.

The Net Promoter Score (NPS) is one such metric that popular amongst the service Industry. It is a customer loyalty metric that measures customers' willingness to not only return for another visit but also make a recommendation to their family, friends or colleagues. Calculation for the NPS involves three types of customers; Promoters, passives and detractors. Promoters are customer with high loyalty and would recommend to others enthusiastically. Passives are customers who



are satisfied, but may not refer to others and may be attracted by other offers from competitors. Lastly, detractors are customers who are not satisfied with the hotel and would not recommend it to others. The final NPS score is calculated by subtracting the percentage of promoters from detractors. Ideally, for sustained growth hotels would want a greater percentage of promoters and less detractors. In addition, they would want passive to move to the promotor category.

This report aims to suggest recommendations to Hyatt Hotels in China to increase their NPS scores by using data on hotel and guests; this report conducts various modelling techniques to determine which factors affect the likelihood to recommend and then proposes policy recommendations for the hotel to retain existing customers and attract new ones for a long run sustained growth.

Report Objectives

This report puts forth the following hypothesis. Analysis is done on data from 2014 to 2015 and analysis aims to answer the following the below states hypothesis questions. The analysis presented in this report aims to understand the effect of the following factors on the Likelihood to recommend;

Hypothesis 1	Supply side factors, i.e. the services provided
Hypothesis 2	Room pricing
Hypothesis 3	Satisfaction from customer care services

Hypothesis 4	Purpose of visit and Demographics

Data Cleaning, Munging and Variables for Analysis

Data Cleaning & Munging

Data files were obtained for all 12 months for analysis. The following rules were set for the first phase of data cleaning with the provided reasoning:

Phase 1: General Cleaning
- Data fields/columns not having data for more than 60% were deleted. This preliminary analysis was done in Excel. Once variables were skimmed down to 100, further cleaning was done in R

Phase 2: Narrowing Regional Scope
- China was chosen as the country to analyse given its growing importance as a business hub and tourist destination. All hotels in China were selected to obtain ample data for analysis

Phase 3: Narrowing Scope of Analysis
- Of the total variables 67 were selected for all types of modelling
- This data was then bought into R for further analysis

Phase 4: Getting Ready
- For Linear modelling, Likelihood to recommend was selected as the dependent variable of analysis

Once data files were ready in excel they were pulled into R for further analysis. The code for data cleaning is attached in Annex 1 as the code book

Variables for Analysis

The key variable of analysis in this report is the Likelihood to recommend. In addition, multiple service variables, satisfaction metrics and demographic variables were used in the analysis. The following variables were added as controls in the modelling. All control variables were coded as dummy variables, where female ,business visit and urban were coded as 1 and male, leisure visit and suburban were coded as 0. This was done to make linear analysis easy to interpret through one regression.

Service variables were skimmed down from 27 to 10. Variables having missing data for 35% or more were deleted to retain sample size. In addition, variables having "yes" or "no" for all instances were deleted since their effect on the likelihood to recommend could not be gauged without a comparison (not having/ having those services). Lastly, variables showing high multicollinearity were also taken out to avoid bias in the model. Satisfaction metrics were also included in the analysis. Lastly, a new variable "Revdiff" was created. This measured the difference between the quoted rate of a hotel room and the actual amount paid by the customer. A positive result for Revdiff would indicate that the customer paid less than the quoted and rate and hence, implying that he had a discount on his stay. On the other had, a negative Revdiff value would imply an error in entry since the actual cannot be greater than quoted. These negative values were deleted from the data set to avoid bias. The data set contained quoted rates in the local currency and hence, had to be converted in US dollars for comparison. Table 1 below gives a break down of all variables used for analysis. The code for data manipulation for linear modelling is attached in Annex 1

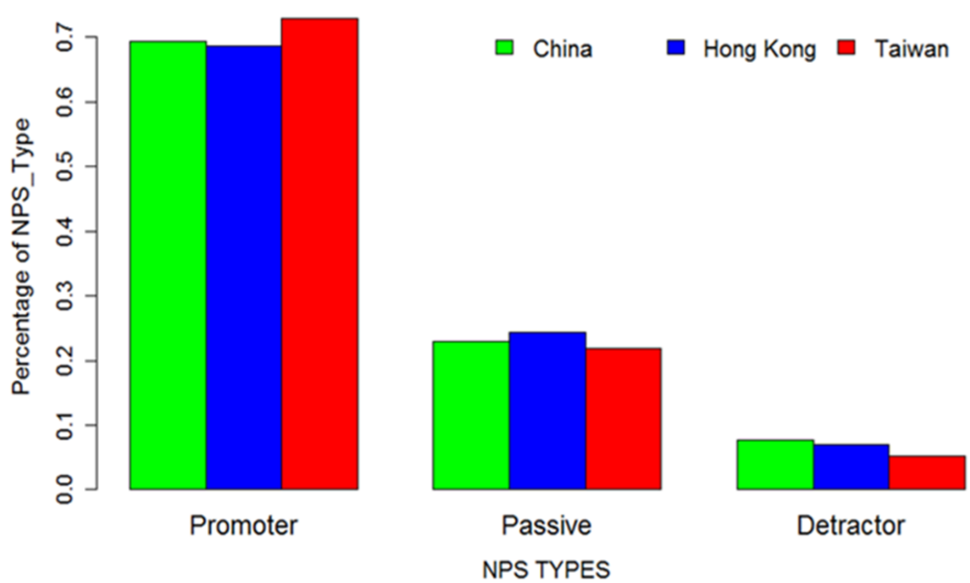
Table 1

Control Variables	**Service Variables**	**Metric Variables**	**Other variables**
Gender	Spa	Quality of Customer Service	Difference between quoted and actual rate paid
Location	Shuttle	Internet Satisfaction	
Purpose of Visit	Self-parking	Condition of Hotel	
	Outdoor pool	Quality of Staff Care	
	Indoor Pool	Overall F&B Experience	
	Valet Parking		
	Convention Centre		
	Regency grand Club		
	Golf		
	Fitness Trainer		

Descriptive statistics & Visualizations

The spread of Promotors, passives and detractors across China, Taiwan and Hong Kong show that promotors form the highest chunk, followed by passives and then by detractors. While the difference may be small, China has the greatest detractors in comparison to Taiwan and Hong Kong.

Figure 1: How NPS Varies Across China



Within China, Hong Kong and Taiwan, Jinan has the highest number of promotors followed by Cotai and Zhouliang Zhuang as shown in Fig 2a. Despite being a big city, Beijing has the highest percentage of detractors and passives

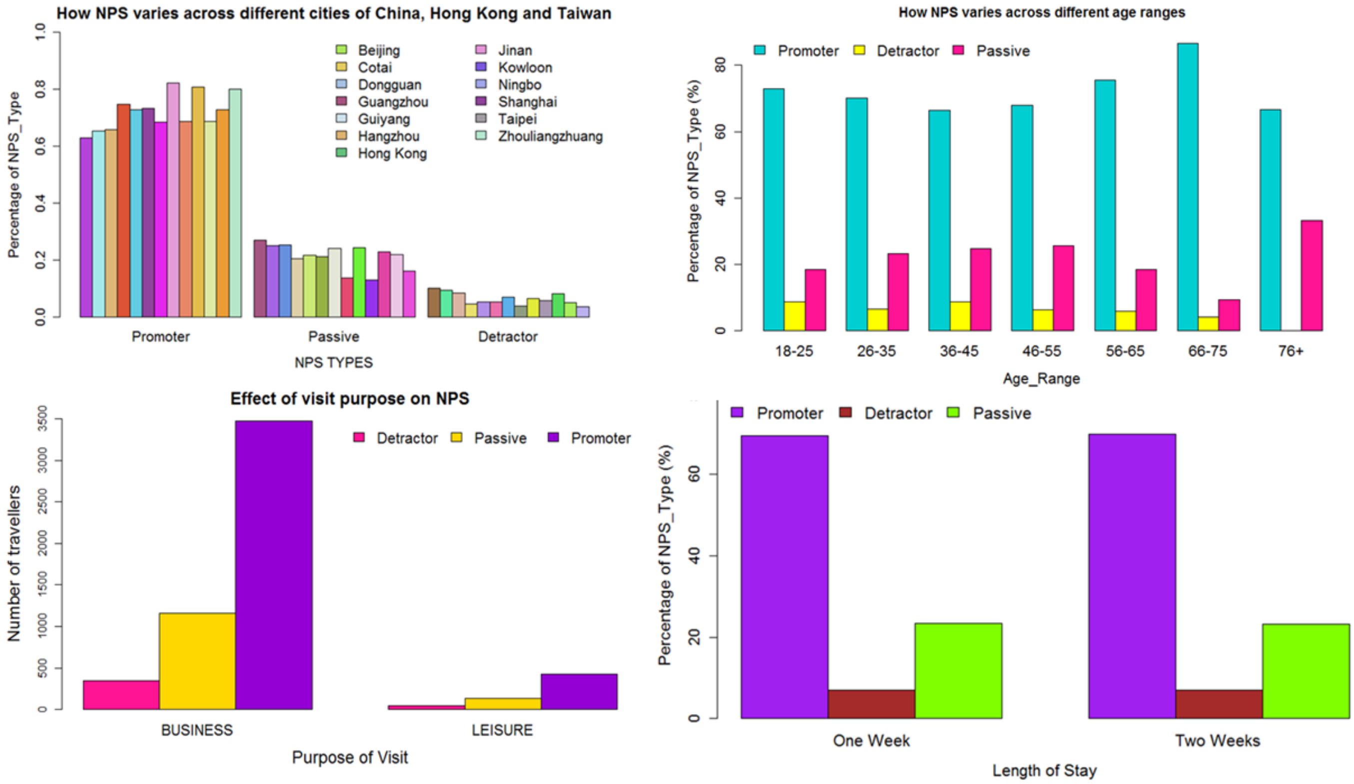
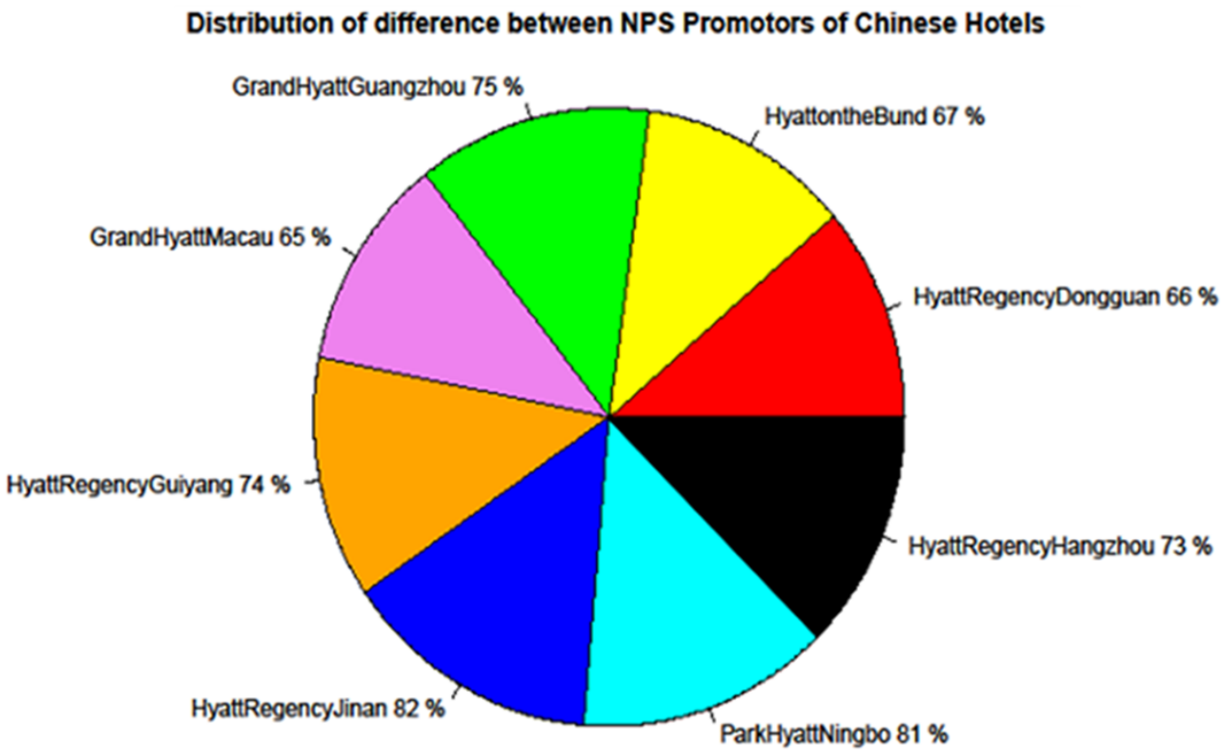


Fig 2a) How NPS varies across different cities in China, Fig2b) NPS across different ages, Fig 2c) Purpose of visit and NPS, Fig 2d) Length of Stay and NPS

Fig 2b shows that promotors form the greatest proportion for all age groups in the data. They are the highest for age 66-67 and lowest for age 76+. Detractors are highest for ages 18-25 and passives show an ambiguous trend, where they rise from 18-45 then decrease up to age 75 and rise again. Younger people from age 18-25 can also be considered to have fewer finances at their disposal for hotels and may be swayed easily by competitors. Fig 2c shows that most promoters visit hotels in China for business and less for leisure. This may be obvious from the fact that China is a new an upcoming economy with high prospects for new businesses and expansion for current ones. Fig 2d shows that most promoters stay for one to two weeks at the hotel.

The difference between NPS types of different Chinese hotels was also evaluated (Fig 3) to see if there is any significant different between Chinese hotels. Generally, the results showed indicating



there is was significant competition between Chinese hotels in terms of high NPS and all are in the same range. The codebook for descriptive statistics is attached in Annex 2.

Modeling techniques & Visualization

Linear Modelling

Linear modelling was used as the prime mode of analysis and its results were validated using Association rules and KSVM modelling. The following generic linear model was set up:

$Y_{LHR}= \beta_0 + \beta_1$ (Gender) + β_2 (Location) + β_3 (Purpose of Visit) + β_4 (Spa) + β_5 (Shuttle) + β_6 (Self-Parking) + β_7 (Outdoor Pool) + β_8 (Indoor Pool) + β_9 (Valet Parking) + β_{10} (Convention Centre) + β_{11} (Regency Grand Club) + β_{12} (Golf) + β_{13} (Fitness Trainer) + β_{14} (Quality of Customer Care) + β_{15} (Internet Satisfaction) + β_{16} (Condition of Hotel)+ β_{17} (Quality of Staff Cared) + + β_9 (Overall F&B Experience) + β_{10} (Difference Between Quoted and Actual Room Price) + e

Four different versions of the model were run. This is shown below in Table 2. Model 1 was run with only the control variables and model 2 was run with the control and service variables only. Both models reported low adjusted R-square values. Model 3 includes control, service, satisfaction metric and price variables. Model 4 runs the same model as 3 but without the service variables. Model 3 has the highest adjusted r-square value of 0.654, this means that 65% of the variation in the likelihood to recommend can be explained by the dependent variables in the model. Excluding the service variables in the model, reduces the adjusted r-square to 65.3. Since this change is small, it is better to keep the service variables in the model. Also, the Residual standard error is lowest for model 3 at 0.860. Hence, model 3 is the preferred model of choice. The code book for linear modelling is attached in Annex 3.

Table 2: Regression Output

Dependent variable: *Likelihood to Recommend*	Model 1	Model 2	Model 3	Model 4
Independent Variables				
Female	-0.049	-0.036	0.03	0.03
	(0.065)	(0.065)	(0.039)	(0.039)
Purpose Of Visit	0.041	-0.007	-0.044	-0.08
	(0.099)	(0.104)	(0.062)	(0.059)
Location	-0.06	-0.095	0.052	-0.065
	(0.073)	(0.12)	(0.071)	(0.043)
Spa		-0.17	-0.254	
		(0.385)	(0.228)	
Shuttle Service		0.157	0.043	
		(0.101)	(0.06)	
Self-Parking		0.365	0.234	
		(0.564)	(0.333)	
Pool Outdoor		-0.27	-0.15	
		(0.353)	(0.209)	
Pool Indoor		-0.02	0.045	
		(0.345)	(0.204)	
Valet Parking		0.390**	-0.003	
		(0.161)	(0.096)	
Convention Centre		-0.022	0.026	
		(0.147)	(0.087)	
Regency Grand Club		-0.421	-0.208	
		(0.463)	(0.274)	
Golf		0.118	-0.022	
		(0.392)	(0.232)	
Fitness Trainer		-0.027	-0.009	
		(0.294)	(0.174)	

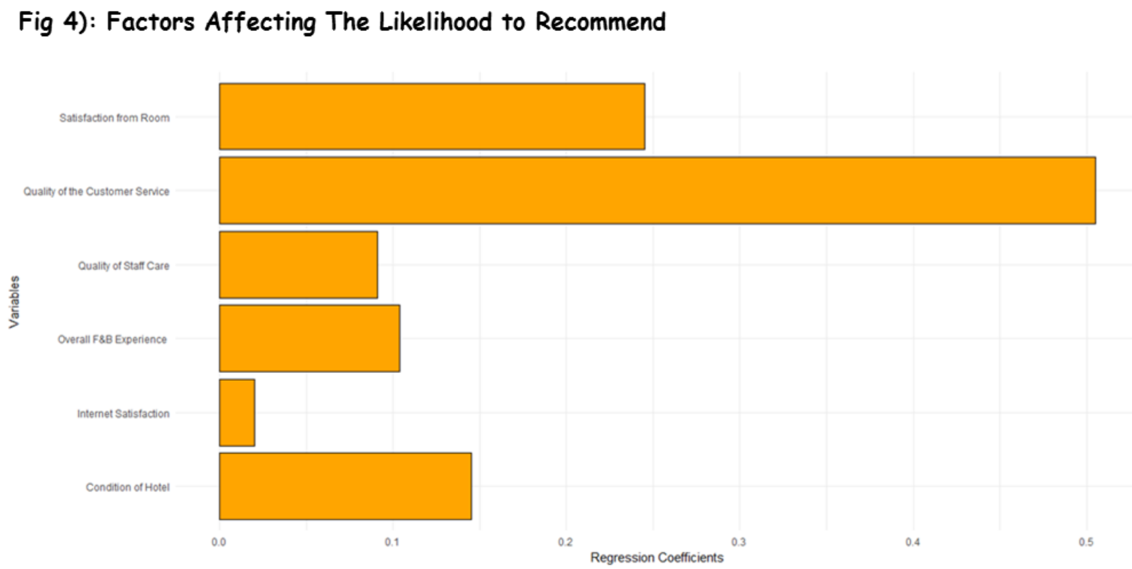
In Model 3, the significant variables are all metric variables except for quality of check in. The satisfaction from room variable suggests that, a unit increase in satisfaction from room leads to a 0.245 units increase in the likelihood to recommend, keeping all else constant. A unit increase in the quality of customer service increases likelihood to recommend by 0.505 units, keeping all else constant. An in increase in the condition of hotel (cleanliness etc) increases the likelihood to recommend by 0.145 units, keeping all else constant. Having a caring staff increases the likelihood to recommend by 0.091 units, keeping all else constant. Also, having a good food and beverage experience also increase the likelihood to recommend by 0.104 units. The above results are significant at the 1% level. Lastly, an increase in internet satisfaction increases the likelihood to recommend by 0.020 units and this result is significant at the 5% level. Control variables are insignificant in the model. This means that the likelihood to recommend is not different for men and women in the data set. This also holds true for the purpose of visit, i.e business or leisure and for location, i.e. urban or suburban. The results and their impact are summarized in Fig 4. The results show that customers experiencing high customer service and room service are more likely to recommend the hotel to others. In addition, having a good hotel condition adds to the appeal from a customer perspective, increasing his chances of recommending to friends and families.

	Model 1	Model 2	Model 3	Model 4
Satisfaction from Room			0.245***	0.244***
			(0.022)	(0.022)
Quality of the Customer Service			0.505***	0.501***
			(0.03)	(0.03)
Internet Satisfaction			0.020**	0.019*
			(0.01)	(0.01)
Condition of Hotel			0.145***	0.149***
			(0.026)	(0.026)
Quality of Staff Care			0.091***	0.093***
			(0.023)	(0.023)
Quality of Check in			-0.003	-0.006
			-0.017	-0.017
Overall F&B Experience			0.104***	0.104***
			(0.017)	(0.017)
Difference in Quoted and Actual Price			-0.0003	-0.0002
			(0.001)	(0.001)
Constant	8.895***	9.173***	-0.764*	-0.896***
	-0.111	-0.731	-0.46	-0.166
R^2	0.001	0.009	0.657	0.655
Adjusted R^2	-0.001	0.004	0.654	0.653
Residual Std. Error	1.462	1.459	0.860	0.861
F Statistic	0.481	1.809**	225.629***	428.094***
Note:	*p<0.1; **p<0.05; ***p<0.01			

KSVM

We are also using a KSVM model to predict different types of NPS and also validate our previous modeling techniques to evaluate if the results from different approaches are consistent or different. We selected different variables that we hypothesized to have most effects on NPS Types based on insights found from descriptive analysis and linear modelling. Several KSVM models were created and the best ones were selected for presentation. Each time independent variables were changed the model was rerun to see how accuracy is affected. Model arguments were set for lower cost of constraints and higher cross validations to make less mistakes and avoid overfitting. Finally, model with fewer variables was selected to make it parsimonious without compromising accuracy. Eventually we fixed KSVM model with 7 variables including Overall_Sat_H, Guest_Room_H, Condition_Hotel_H, Customer_SVC_H, Staff_Cared_H, Internet_Sat_H and F.B_Overall_Experience_H.

KSVM model results is showing the model is performing better for prediction of promotor than other two NPS Types (Fig 5). Model is underprediction detractors by 30%, overpredction passive by 14% while only underpredicting promotors by 1% for. It might show also the data for China is not normally distributed and the model is biased and skewed towards promotors. However, more surveys and data set from detractors and passive can increase validation.



We also

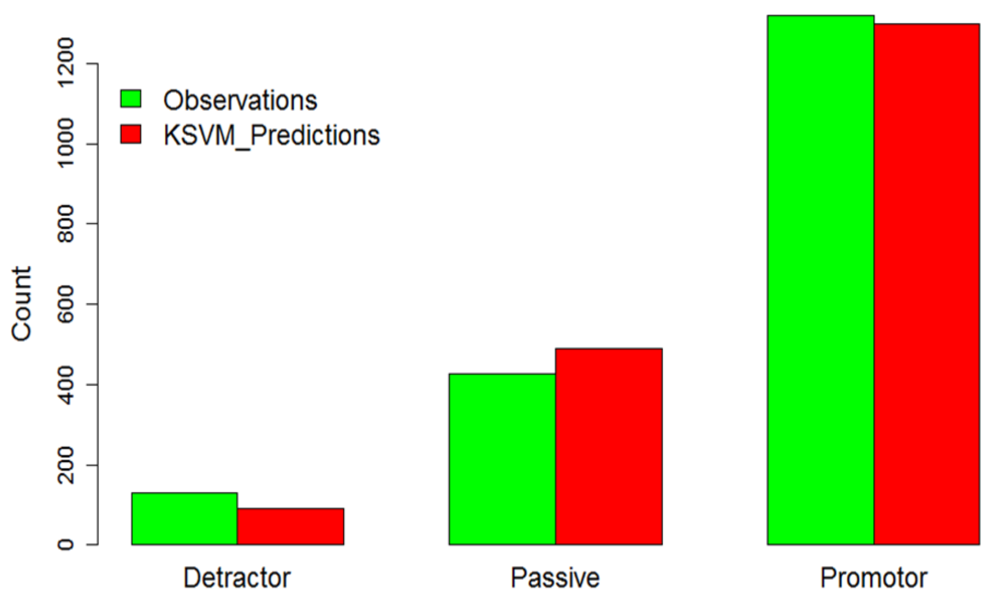


Fig 5) Comparison between KSVM Model predictions for NPS Types with observations.

developed a plot of KSVM Model results for two variables of overall satisfaction and staff care (Figure. 6) to get a deeper understanding of how the model is performing in predicting NPS Type when we have these two variables as and x and y on the plot (Figure .6). The plot is indicating the model could capture NPS types mainly for promotors and passive (the rectangles and triangles, respectively) but at high staff care and low overall satisfaction is prediction promotor as NPS Type while observations are showing NPS Type as detractors (circles).

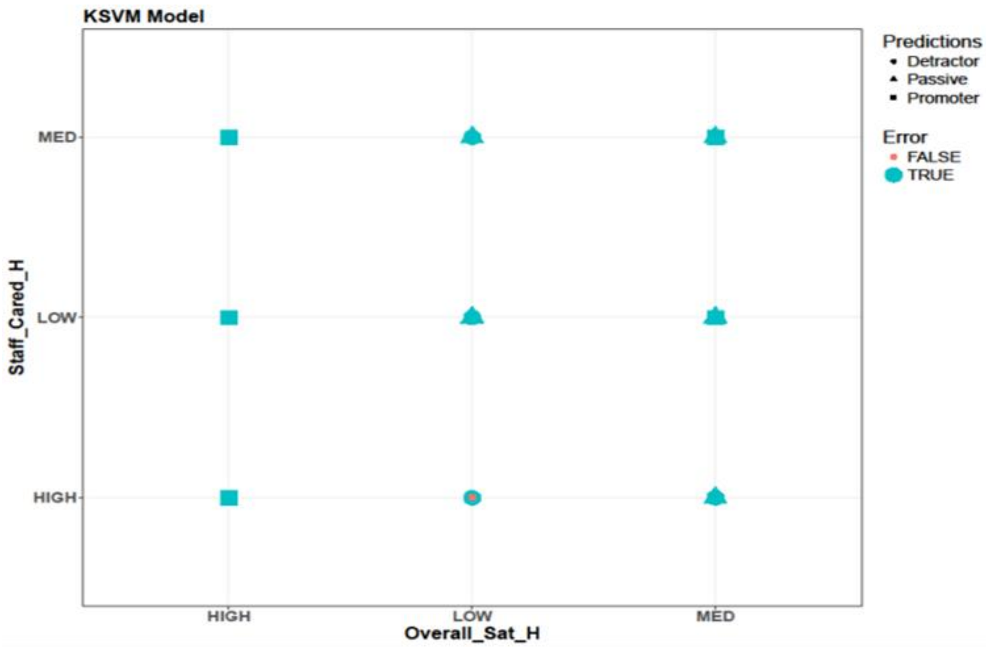


Figure 6. KSVM Model predictions for NPS Types versus overall satisfaction and staff care variables.

As result, KSVM technique indicates a well performing model with accuracy rate of 88%. Moreover, the results from KSVM model were consistent with the linear regression model and arules in terms of most important factors or variables that are affecting NPS Types and the likelihood to recommend. As above mentioned, having equal numbers of surveys for different categories of NPS types can increased model accuracy and validate our results. The code book for KSVM is attached in Annex 4. Moreover, there are more plots related to KSVM technique in appendix.

Association Rules

The key variables for analysis are those identified by linear modelling and are shown in figure 1. Association rules or arules uses read. The transaction function is sued to read the data from the data set and store it as a matrix. This matrix will count how many times each data has occurred in the data frame and will store it in matrix form. For analysis, we have considered scores from 8-10 are promoters, 6-7 are passive and 1-5 are detractors. In the code we changed each data to show the proper count as required for arules. For example, promoter data (range 8-10) for Guest_Room_ H is replaced as P_Guest_Room. Similarly, passive data (range 6-7) for Guest_Room_ H is replaced as N_Guest_Room detractor data (range 1-5) for Guest_Room_ H is replaced as D_Guest_Room.

Next we have divided the dataset further into separate datasets for promoters, detractors and passive. Using arules package, we have applied read.transaction function to each of the three datasets. Thus we have created matrix for data from promoters, detractors and passives.Using the summary function of arules, we can find the most frequent items occurring in that particular dataset and how many times it is occurring. We can also find the density of a particular item occurring in the matrix using the support function.The summary function also gives the distribution of data in the transaction matrix ranging from min-median-max. The itemFrequencyPlot function of arules will plot the items vs frequencies.In our code we have taken the most significant items to plot the graphs. The analysis is based on these graphs;

I. From the promoter's graph (fig 7) we can analyze that Overall Satisfaction, Condition of Hotel, Quality of customer service, Satisfaction from Room, Quality of Staff Care are the most significant ones and might be the strong reasons for the customers to be promoters.

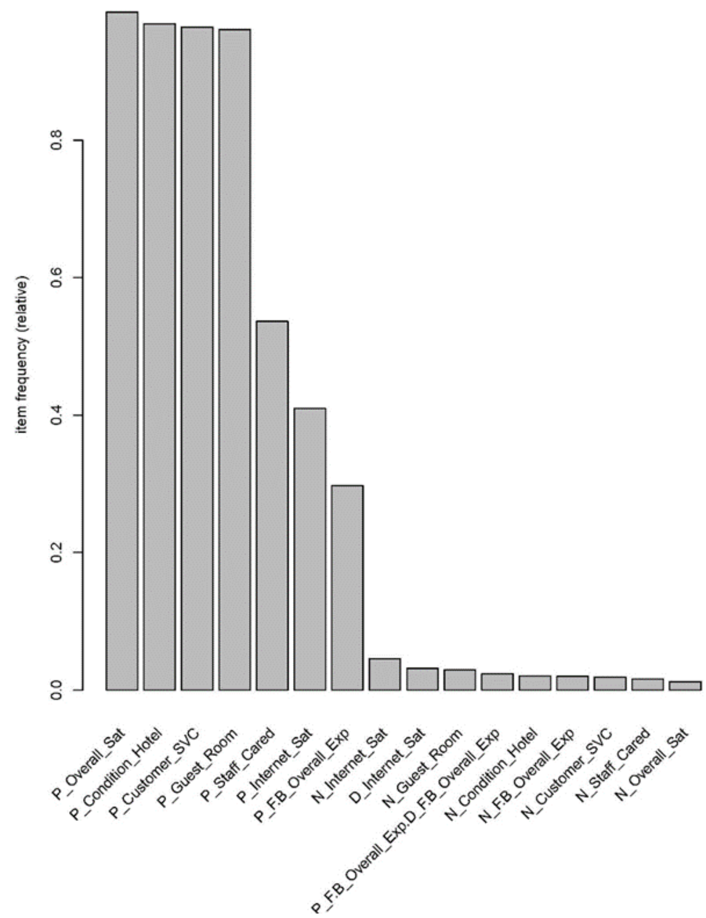
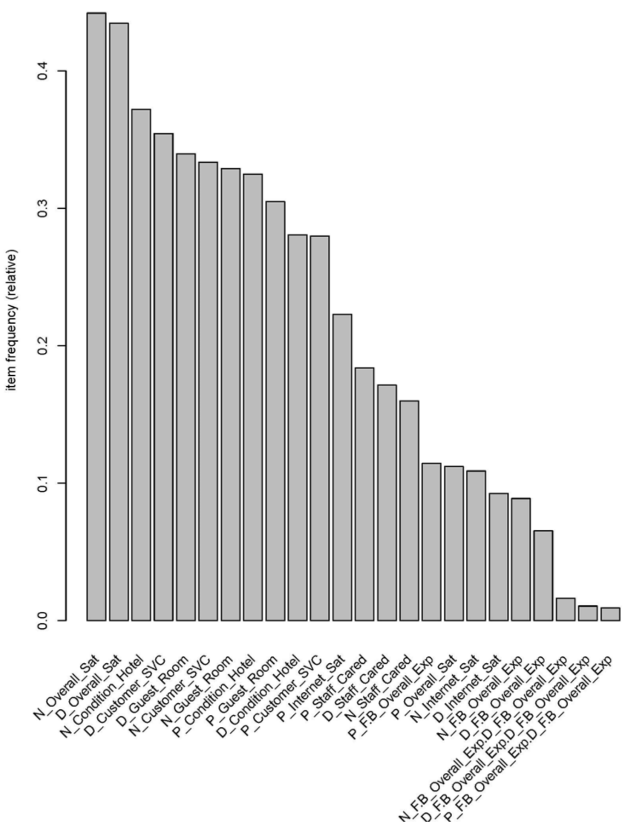


Fig. 7): Significant Variables for Promoters

II. From the detractor's graph (Fig8), we can see that there is a high frequency of detractors and passives for Overall satisfaction indicating that if we increase this variable, we can possibly convert a portion of detractors to promoters. The next significant frequency is of the detractors and passives for Quality of Customer Service, thirdly of Satisfaction from room, Condition of Hotel and Quality of staff care can be said to be somewhat in the middle.

III. From the passive's graph, a majority of positive response for Condition of Hotel, Quality of Customer Service, a slight decrease in the frequency can be seen in Overall Satisfaction and Satisfaction from Room, and a considerable amount of customers who were neutral about that Overall Satisfaction, Condition of Hotel, Quality of Customer Service, Satisfaction from Room



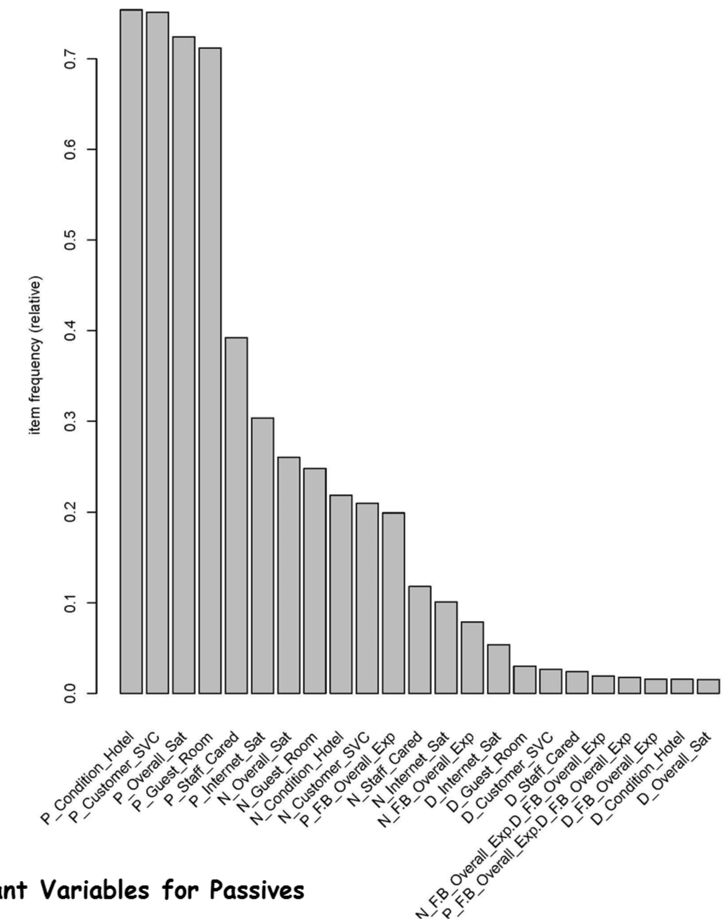


Fig 8a. Significant Variables for Detractors, Fig8b. Significant Variables for Passives

From the above data we can suggest that by increasing the quality of services by a small amount in Overall Satisfaction, Condition of Hotel, Quality of Customer Service, Satisfaction from Room, we can convert our passives to promoters and Overall Satisfaction, Quality of Customer Service can play a significant role in converting our detractors to promoters. The code book for association rules is attached in annex 4.

Validation of Results

The basic model for analysis in this report is Linear model, however KSVM and Association rules were used to validate results. Model 3 was taken as the preferred model and explained approximately 65% of the variation in the likely hood to recommend. Both validation methods consolidate the results shown my linear modelling that factors affecting metric variables significantly affect the likelihood to recommend and will be able to bring passives and detractors into the promoter category. The KSM model also shows that the prediction has the most power for promoters, i.e. the difference between current observations and predictions is 1%. Whereas, that for detractors is 30% and for passives it is 14%.

Policy Recommendations

Based on the analysis this report proposes four policy recommendations;

Trainings to Hotel Staff for Improved Customer and Room Service. Hotel should focus on providing customers with quality room and customer service. This can be achieved through trainings for hotel staff. Training mode should be based on real life simulations to better train hotel staff.

Improve Quality of Food and Beverage at the Hotel. Since most customers visiting the hotel for business, this means that the customer base is diverse. Different food and drink items should be included on the menu and staff should be trained how to effectively deal with diverse customers.

Improve Hotel Outlook. Resources from existing services, i.e. pool, spa etc should be diverted to maintenance of the hotels to keep a certain standard or outlook. Internally, hotel management can work on a maintenance manual that develops management indicators to quarterly monitor hotel outlook and provide prompt repairs where needed

Annexures

Annex 1

###CODE BOOK FOR DATA CLEANING AND MUNGING###

Bringing in Data Sets from CSV

```
China_Data_Dec <- read.csv("Dec.csv")
China_Data_Aug <- read.csv("Aug.csv")
China_Data_Sept <- read.csv("Sept.csv")
```

#Removing rows with empty likelihood to recommend value

```
LL <- function(x)
{
  x <- x[!(is.na(x$Likelihood_Recommend_H) | (x$Likelihood_Recommend_H == "")),]
  return(x)
}
```

#Applying function to all datasets

```
Dec <- LL(China_Data_Dec)
Sept <- LL(China_Data_Sept)
Aug <- LL(China_Data_Aug)
```

#Keeping only China

```
Deccountry <- Dec[which(Dec$Country_PL== "China" | Dec$Country_PL== "Hong Kong" |
Dec$Country_PL== "Taiwan"),]
Septcountry <- Sept[which(Sept$Country_PL== "China" | Sept$Country_PL== "Hong Kong" |
Sept$Country_PL== "Taiwan"),]
Augcountry <- Aug[which(Aug$Country_PL== "China" | Aug$Country_PL== "Hong Kong" |
Aug$Country_PL== "Taiwan"),]
```

#Keeping selected variables only

```
Cleandec <- data.frame(Deccountry$RESERVATION_CONFIRMATION_NUM_C,
Deccountry$CHECK_IN_DATE_C,Deccountry$CHECK_OUT_DATE_C,Deccountry$LENGT
H_OF_STAY_C,Deccountry$POV_CODE_C,Deccountry$QUOTED_RATE_C,Deccountry$P
MS_ROOM_REV_USD_C,Deccountry$CONFIRMATION_NUM_R,Deccountry$RESERVAT
ION_DATE_R,Deccountry$RESERVATION_STATUS_R,Deccountry$NUM_ROOMS_R,Dec
country$NT_RATE_R,Deccountry$COUNTRY_CODE_R,Deccountry$CALCULATED_NIGH
TS_NUM_R,Deccountry$ROOM_NIGHTS_R,Deccountry$STATUS_CALCULATION_R,Dec
country$GUEST_COUNTRY_R,Deccountry$REVENUE_R,Deccountry$REVENUE_USD_R,
Deccountry$Gender_H,Deccountry$Age_Range_H,Deccountry$Likelihood_Recommend_H,De
ccountry$Country_PL,Deccountry$Property.Latitude_PL,Deccountry$Property.Longitude_PL,D
eccountry$Currency_PL,Deccountry$Guest.NPS.Goal_PL,Deccountry$Location_PL,Deccountr
y$Bell.Staff_PL,Deccountry$Boutique_PL,Deccountry$Business.Center_PL,Deccountry$Casin
o_PL,Deccountry$Fitness.Center_PL,Deccountry$Fitness.Trainer_PL,Deccountry$Golf_PL,Dec
country$Regency.Grand.Club_PL,Deccountry$Ski_PL,Deccountry$Valet.Parking_PL,Deccount
```

ry$Convention_PL,Deccountry$Elevators_PL,Deccountry$Conference_PL,Deccountry$Indoor.
Corridors_PL,Deccountry$Laundry_PL,Deccountry$Limo.Service_PL,Deccountry$Mini.Bar_P
L,Deccountry$Pool.Indoor_PL,Deccountry$Pool.Outdoor_PL,Deccountry$Restaurant_PL,Decc
ountry$Self.Parking_PL,Deccountry$Shuttle.Service_PL,Deccountry$Spa_PL,Deccountry$NPS
_Type,Deccountry$LENGTH_OF_STAY_R,Deccountry$OFFER_FLG_R,Deccountry$Overall
_Sat_H,Deccountry$Guest_Room_H,Deccountry$Customer_SVC_H,Deccountry$Internet_Sat_
H,Deccountry$Dry.Cleaning_PL,Deccountry$GP_Tier_H,Deccountry$City_PL,Deccountry$Cla
ss_PL,Deccountry$Hotel.Name.Long_PL,Deccountry$Condition_Hotel_H,Deccountry$Staff_C
ared_H,Deccountry$Check_In_H,Deccountry$F.B_Overall_Experience_H)
Cleansept <- data.frame(Septcountry$RESERVATION_CONFIRMATION_NUM_C,
Septcountry$CHECK_IN_DATE_C,Septcountry$CHECK_OUT_DATE_C,Septcountry$LENG
TH_OF_STAY_C,Septcountry$POV_CODE_C,Septcountry$QUOTED_RATE_C,Septcountry$
PMS_ROOM_REV_USD_C,Septcountry$CONFIRMATION_NUM_R,Septcountry$RESERV
ATION_DATE_R,Septcountry$RESERVATION_STATUS_R,Septcountry$NUM_ROOMS_R,
Septcountry$NT_RATE_R,Septcountry$COUNTRY_CODE_R,Septcountry$CALCULATED_
NIGHTS_NUM_R,Septcountry$ROOM_NIGHTS_R,Septcountry$STATUS_CALCULATION
_R,Septcountry$GUEST_COUNTRY_R,Septcountry$REVENUE_R,Septcountry$REVENUE_
USD_R,Septcountry$Gender_H,Septcountry$Age_Range_H,Septcountry$Likelihood_Recomme
nd_H,Septcountry$Country_PL,Septcountry$Property.Latitude_PL,Septcountry$Property.Longit
ude_PL,Septcountry$Currency_PL,Septcountry$Guest.NPS.Goal_PL,Septcountry$Location_PL,
Septcountry$Bell.Staff_PL,Septcountry$Boutique_PL,Septcountry$Business.Center_PL,Septcou
ntry$Casino_PL,Septcountry$Fitness.Center_PL,Septcountry$Fitness.Trainer_PL,Septcountry$
Golf_PL,Septcountry$Regency.Grand.Club_PL,Septcountry$Ski_PL,Septcountry$Valet.Parking
_PL,Septcountry$Convention_PL,Septcountry$Elevators_PL,Septcountry$Conference_PL,Septc
ountry$Indoor.Corridors_PL,Septcountry$Laundry_PL,Septcountry$Limo.Service_PL,Septcoun
try$Mini.Bar_PL,Septcountry$Pool.Indoor_PL,Septcountry$Pool.Outdoor_PL,Septcountry$Rest
aurant_PL,Septcountry$Self.Parking_PL,Septcountry$Shuttle.Service_PL,Septcountry$Spa_PL,
Septcountry$NPS_Type,Septcountry$LENGTH_OF_STAY_R,Septcountry$OFFER_FLG_R,Se
ptcountry$Overall_Sat_H,Septcountry$Guest_Room_H,Septcountry$Customer_SVC_H,Septco
untry$Internet_Sat_H,Septcountry$Dry.Cleaning_PL,Septcountry$GP_Tier_H,Septcountry$City
_PL,Septcountry$Class_PL,Septcountry$Hotel.Name.Long_PL,Septcountry$Condition_Hotel_
H,Septcountry$Staff_Cared_H,Septcountry$Check_In_H,Septcountry$F.B_Overall_Experience
_H)

CleanAug <- data.frame(Augcountry$RESERVATION_CONFIRMATION_NUM_C,
Augcountry$CHECK_IN_DATE_C,Augcountry$CHECK_OUT_DATE_C,Augcountry$LENG
TH_OF_STAY_C,Augcountry$POV_CODE_C,Augcountry$QUOTED_RATE_C,Augcountry$
PMS_ROOM_REV_USD_C,Augcountry$CONFIRMATION_NUM_R,Augcountry$RESERVA
TION_DATE_R,Augcountry$RESERVATION_STATUS_R,Augcountry$NUM_ROOMS_R,A
ugcountry$NT_RATE_R,Augcountry$COUNTRY_CODE_R,Augcountry$CALCULATED_NI
GHTS_NUM_R,Augcountry$ROOM_NIGHTS_R,Augcountry$STATUS_CALCULATION_R,
Augcountry$GUEST_COUNTRY_R,Augcountry$REVENUE_R,Augcountry$REVENUE_US
D_R,Augcountry$Gender_H,Augcountry$Age_Range_H,Augcountry$Likelihood_Recommend

_H,Augcountry$Country_PL,Augcountry$Property.Latitude_PL,Augcountry$Property.Longitude_PL,Augcountry$Currency_PL,Augcountry$Guest.NPS.Goal_PL,Augcountry$Location_PL,Augcountry$Bell.Staff_PL,Augcountry$Boutique_PL,Augcountry$Business.Center_PL,Augcountry$Casino_PL,Augcountry$Fitness.Center_PL,Augcountry$Fitness.Trainer_PL,Augcountry$Golf_PL,Augcountry$Regency.Grand.Club_PL,Augcountry$Ski_PL,Augcountry$Valet.Parking_PL,Augcountry$Convention_PL,Augcountry$Elevators_PL,Augcountry$Conference_PL,Augcountry$Indoor.Corridors_PL,Augcountry$Laundry_PL,Augcountry$Limo.Service_PL,Augcountry$Mini.Bar_PL,Augcountry$Pool.Indoor_PL,Augcountry$Pool.Outdoor_PL,Augcountry$Restaurant_PL,Augcountry$Self.Parking_PL,Augcountry$Shuttle.Service_PL,Augcountry$Spa_PL,Augcountry$NPS_Type,Augcountry$LENGTH_OF_STAY_R,Augcountry$OFFER_FLG_R,Augcountry$Overall_Sat_H,Augcountry$Guest_Room_H,Augcountry$Customer_SVC_H,Augcountry$Internet_Sat_H,Augcountry$Dry.Cleaning_PL,Augcountry$GP_Tier_H,Augcountry$City_PL,Augcountry$Class_PL,Augcountry$Hotel.Name.Long_PL,Augcountry$Condition_Hotel_H,Augcountry$Staff_Cared_H,Augcountry$Check_In_H,Augcountry$F.B_Overall_Experience_H)

#Making column names the same

colnames(Cleandec) <-
c("RESERVATION_CONFIRMATION_NUM_C","CHECK_IN_DATE_C","CHECK_OUT_DATE_C","LENGTH_OF_STAY_C","POV_CODE_C","QUOTED_RATE_C","PMS_ROOM_REV_USD_C","CONFIRMATION_NUM_R","RESERVATION_DATE_R","RESERVATION_STATUS_R","NUM_ROOMS_R","NT_RATE_R","COUNTRY_CODE_R","CALCULATED_NIGHTS_NUM_R","ROOM_NIGHTS_R","STATUS_CALCULATION_R","GUEST_COUNTRY_R","REVENUE_R","REVENUE_USD_R","Gender_H","Age_Range_H","Likelihood_Recommend_H","Country_PL","Property.Latitude_PL","Property.Longitude_PL","Currency_PL","Guest.NPS.Goal_PL","Location_PL","Bell.Staff_PL","Boutique_PL","Business.Center_PL","Casino_PL","Fitness.Center_PL","Fitness.Trainer_PL","Golf_PL","Regency.Grand.Club_PL","Ski_PL","Valet.Parking_PL","Convention_PL","Elevators_PL","Conference_PL","Indoor.Corridors_PL","Laundry_PL","Limo.Service_PL","Mini.Bar_PL","Pool.Indoor_PL","Pool.Outdoor_PL","Restaurant_PL","Self.Parking_PL","Shuttle.Service_PL","Spa_PL","NPS_Type","LENGTH_OF_STAY_R","OFFER_FLG_R","Overall_Sat_H","Guest_Room_H","Customer_SVC_H", "Internet_Sat_H","Dry.Cleaning_PL","GP_Tier_H","City_PL","Class_PL","Hotel.Name.Long_PL","Condition_Hotel_H","Staff_Cared_H","Check_In_H","F.B_Overall_Experience_H")

colnames(CleanAug) <-
c("RESERVATION_CONFIRMATION_NUM_C","CHECK_IN_DATE_C","CHECK_OUT_DATE_C","LENGTH_OF_STAY_C","POV_CODE_C","QUOTED_RATE_C","PMS_ROOM_REV_USD_C","CONFIRMATION_NUM_R","RESERVATION_DATE_R","RESERVATION_STATUS_R","NUM_ROOMS_R","NT_RATE_R","COUNTRY_CODE_R","CALCULATED_NIGHTS_NUM_R","ROOM_NIGHTS_R","STATUS_CALCULATION_R","GUEST_COUNTRY_R","REVENUE_R","REVENUE_USD_R","Gender_H","Age_Range_H","Likelihood_Recommend_H","Country_PL","Property.Latitude_PL","Property.Longitude_PL","Currency_PL","Guest.NPS.Goal_PL","Location_PL","Bell.Staff_PL","Boutique_PL","Business.Center_PL","Casino_PL","Fitness.Center_PL","Fitness.Trainer_PL","Golf_PL","Regency.Grand.Club_PL","Ski_PL","Valet.Parking_PL","Convention_PL","Elevators_PL","Conference_PL","Indoor.Corridors

```
_PL","Laundry_PL","Limo.Service_PL","Mini.Bar_PL","Pool.Indoor_PL","Pool.Outdoor_PL",
"Restaurant_PL","Self.Parking_PL","Shuttle.Service_PL","Spa_PL","NPS_Type","LENGTH_O
F_STAY_R","OFFER_FLG_R","Overall_Sat_H","Guest_Room_H","Customer_SVC_H","Inter
net_Sat_H","Dry.Cleaning_PL","GP_Tier_H","City_PL","Class_PL","Hotel.Name.Long_PL","
Condition_Hotel_H","Staff_Cared_H","Check_In_H","F.B_Overall_Experience_H")
colnames(Cleansept) <-
c("RESERVATION_CONFIRMATION_NUM_C","CHECK_IN_DATE_C","CHECK_OUT_D
ATE_C","LENGTH_OF_STAY_C","POV_CODE_C","QUOTED_RATE_C","PMS_ROOM_R
EV_USD_C","CONFIRMATION_NUM_R","RESERVATION_DATE_R","RESERVATION_
STATUS_R","NUM_ROOMS_R","NT_RATE_R","COUNTRY_CODE_R","CALCULATED_
NIGHTS_NUM_R","ROOM_NIGHTS_R","STATUS_CALCULATION_R","GUEST_COUNT
RY_R","REVENUE_R","REVENUE_USD_R","Gender_H","Age_Range_H","Likelihood_Rec
ommend_H","Country_PL","Property.Latitude_PL","Property.Longitude_PL","Currency_PL","
Guest.NPS.Goal_PL","Location_PL","Bell.Staff_PL","Boutique_PL","Business.Center_PL","Ca
sino_PL","Fitness.Center_PL","Fitness.Trainer_PL","Golf_PL","Regency.Grand.Club_PL","Ski
_PL","Valet.Parking_PL","Convention_PL","Elevators_PL","Conference_PL","Indoor.Corridors
_PL","Laundry_PL","Limo.Service_PL","Mini.Bar_PL","Pool.Indoor_PL","Pool.Outdoor_PL",
"Restaurant_PL","Self.Parking_PL","Shuttle.Service_PL","Spa_PL","NPS_Type","LENGTH_O
F_STAY_R","OFFER_FLG_R","Overall_Sat_H","Guest_Room_H","Customer_SVC_H","Inter
net_Sat_H","Dry.Cleaning_PL","GP_Tier_H","City_PL","Class_PL","Hotel.Name.Long_PL","
Condition_Hotel_H","Staff_Cared_H","Check_In_H","F.B_Overall_Experience_H")
```

#Combining data other Datasets.

```
y <- read.csv("combined101101.csv")
R <- read.csv("combineddata.csv")
M <- read.csv("combineddata (1).csv")
#combined Data
combineddataN <- rbind(CleanAug,Cleansept,Cleandec)
combineddata <- rbind(combineddataN , R, M,y)
```

Annex 2

Conducting Descriptive Analysis#######

Comparison of NPS types between different countries (China, Taiwan and Hong Kong)

```
Promoter <-tapply(combineddata$NPS_Type=="Promoter",combineddata$
        Country_PL,sum)
Detractor <-tapply(combineddata$NPS_Type=="Detractor",combineddata$
        Country_PL,sum)
Passive <-tapply(combineddata$NPS_Type=="Passive",combineddata$
       Country_PL,sum)
countrydf <-data.frame(Promoter,Passive,Detractor)
countrydf$Total <- countrydf$Promoter+countrydf$Passive+countrydf$Detractor
countrydf$percPro<-countrydf$Promoter /countrydf$Total
countrydf$percPas<-countrydf$Passive /countrydf$Total
```

```
countrydf$percDet<-countrydf$Detractor /countrydf$Total
df1<-as.matrix(countrydf)
df1
par(mar=c(5,6,3,2))
barplot(as.matrix(countrydf)[,c(5,6,7)],beside = T,cex.axis = 1.2,cex.names = 1.5,
    col=c("green","blue","red"),names.arg=c("Promoter","Passive","Detractor"))
legend("topright", legend = rownames(df1),bty="n",
   fill=c("green","blue","red"),ncol=3,cex=1.3)
title(main = "How NPS varies across different countries", xlab =
    "NPS TYPES", ylab = "Percentage of NPS_Type",cex.lab=1.3,cex.main=1.5)
```

Comparison of NPS types between different cities of selected countries (China, Taiwan and Hong Kong)

```
Promoter <-tapply(combineddata$NPS_Type=="Promoter",combineddata$
            City_PL,sum)
Detractor <-tapply(combineddata$NPS_Type=="Detractor",combineddata$
             City_P,sum)
Passive <-tapply(combineddata$NPS_Type=="Passive",combineddata$
           City_P,sum)
citydf <-data.frame(Promoter,Passive,Detractor)
citydf$Total <- citydf$Promoter+citydf$Passive+citydf$Detractor
citydf$percPro<-citydf$Promoter /citydf$Total
citydf$percPas<-citydf$Passive /citydf$Total
citydf$percDet<-citydf$Detractor /citydf$Total
df1<-as.matrix(citydf)
df1
barplot(as.matrix(citydf)[,c(5,6,7)],beside = T,
    col=distinctColorPalette(50),names.arg=c("Promoter","Passive","Detractor"),

    cex.lab=1.3,cex.names = 1.3,cex.axis = 1.2,ylim=c(0,1))
legend("topright", legend = rownames(df1),bty="n",
   ncol=2,cex=1.3,fill=distinctColorPalette(50))
title(main = "How NPS varies across different cities of China, Hong Kong and Taiwan", xlab =
    "NPS TYPES", ylab = "Percentage of NPS_Type",cex.lab=1.3,cex.main=1.5)
```

Comparison of NPS_Promotors among Chinese Hotels###

```
chindata<-combineddata[combineddata$Country_PL=="China",]
unique(chindata$Hotel.Name.Long_PL)
HyattRegencyDongguan<-tapply(chindata$Hotel.Name.Long_PL=="Hyatt Regency
Dongguan",chindata$NPS_Type,sum)
HyattRegencyDongguan<-HyattRegencyDongguan/sum(HyattRegencyDongguan)*100
HyattontheBund<-tapply(chindata$Hotel.Name.Long_PL=="Hyatt on the
Bund",chindata$NPS_Type,sum)
HyattontheBund<-HyattontheBund/sum(HyattontheBund)*100
GrandHyattGuangzhou<-tapply(chindata$Hotel.Name.Long_PL=="Grand Hyatt
Guangzhou",chindata$NPS_Type,sum)
```

```
GrandHyattGuangzhou<-GrandHyattGuangzhou/sum(GrandHyattGuangzhou)*100
GrandHyattMacau<-tapply(chindata$Hotel.Name.Long_PL=="Grand Hyatt
Macau",chindata$NPS_Type,sum)
GrandHyattMacau<-GrandHyattMacau/sum(GrandHyattMacau)*100
HyattRegencyGuiyang<-tapply(chindata$Hotel.Name.Long_PL=="Hyatt Regency
Guiyang",chindata$NPS_Type,sum)
HyattRegencyGuiyang<-HyattRegencyGuiyang/sum(HyattRegencyGuiyang)*100
HyattRegencyJinan<-tapply(chindata$Hotel.Name.Long_PL=="Hyatt Regency
Jinan",chindata$NPS_Type,sum)
HyattRegencyJinan<-HyattRegencyJinan/sum(HyattRegencyJinan)*100
ParkHyattNingbo<-tapply(chindata$Hotel.Name.Long_PL=="Park Hyatt
Ningbo",chindata$NPS_Type,sum)
ParkHyattNingbo<-ParkHyattNingbo/sum(ParkHyattNingbo)*100
HyattRegencyHangzhou<-tapply(chindata$Hotel.Name.Long_PL=="Hyatt Regency
Hangzhou",chindata$NPS_Type,sum)
HyattRegencyHangzhou<-HyattRegencyHangzhou/sum(HyattRegencyHangzhou)*100
dfhotel<-
rbind(HyattRegencyDongguan,HyattontheBund,GrandHyattGuangzhou,GrandHyattMacau,
        HyattRegencyGuiyang,HyattRegencyJinan,ParkHyattNingbo,HyattRegencyHangzhou)
dfhotel<-as.data.frame(dfhotel)
length(dfhotel[,1])
library(RColorBrewer)
lbls <-rownames(dfhotel)[1:8]
colors = rainbow(c(1:14))
#pct<-round(x/sum(x)*100)
pct<-round(dfhotel$Promoter)
lbls <- paste(lbls, pct) # add percents to labels
lbls <-paste(lbls,"%")
pie(abs(dfhotel$Promoter), labels =lbls,
main="Distribution of difference between promotors and detractors of chinese hotels",
 col=c("red", "yellow", "green", "violet", "orange", "blue", "cyan",
      "black","pink",'chartreuse3', 'cornflowerblue', 'darkgoldenrod1', 'peachpuff3',
      'mediumorchid2'))

## or pie(dfhotel$Promoter-dfhotel$Detractor, labels =lbls,
library(randomcoloR)
pie(dfhotel$Promoter-dfhotel$Detractor, labels =lbls,
  col=distinctColorPalette(50))
```

Comparison of NPS Types among various age ranges####

```
# Delete rows that has NA in age range
combineddata <-
combineddata[!(is.na(combineddata$Age_Range_H)|(combineddata$Age_Range_H=="")),]
# Calculate sum of NPS types for each age range
agepromoter <-
tapply(combineddata$NPS_Type=="Promoter",combineddata$Age_Range_H,sum)
```

```
agedetractor <-
tapply(combineddata$NPS_Type=="Detractor",combineddata$Age_Range_H,sum)
agepassive <-tapply(combineddata$NPS_Type=="Passive",combineddata$Age_Range_H,sum)
agedf <- data.frame(agepromoter,agedetractor,agepassive)
str(agedf)
agedf2<-t(agedf)
agedf2[,1] <- (agedf2[,1]/sum(agedf2[,1])) * 100
agedf2[,2] <- (agedf2[,2]/sum(agedf2[,2])) * 100
agedf2[,3] <- (agedf2[,3]/sum(agedf2[,3])) * 100
agedf2[,4] <- (agedf2[,4]/sum(agedf2[,4])) * 100
agedf2[,5] <- (agedf2[,5]/sum(agedf2[,5])) * 100
agedf2[,6] <- (agedf2[,6]/sum(agedf2[,6])) * 100
agedf2[,7] <- (agedf2[,7]/sum(agedf2[,7])) * 100
# Barplot for Age_Range
par(mar=c(5,5,3,1))
barplot(agedf2,beside = TRUE,col=c("darkturquoise","yellow","deeppink"),
      cex.axis=1.2,cex.names = 1.3,
      ylim = c(0,90))
legend("topleft", legend = c("Promoter","Detractor","Passive"),bty="n",
     fill=c("darkturquoise","yellow","deeppink"),ncol=3,cex=1.3)
title(main = "How NPS varies across different age ranges", xlab =
      "Age_Range", ylab = "Percentage of NPS_Type (%)",cex.lab=1.3)
```

Effects of length day on NPS Types#####

```
# Calculate sum of NPS types for each length of stay for each country
staypro<-
tapply(combineddata$NPS_Type=="Promoter",combineddata$LENGTH_OF_STAY_R,sum)
staydet<-
tapply(combineddata$NPS_Type=="Detractor",combineddata$LENGTH_OF_STAY_R,sum)
staypas<-
tapply(combineddata$NPS_Type=="Passive",combineddata$LENGTH_OF_STAY_R,sum)
dfstay <- data.frame(staypro,staydet,staypas)
str(dfstay)
dfstay2<-t(dfstay)
dfstay2
stayonewpro<-sum(dfstay2[1,c(1:7)])
stayonedet<-sum(dfstay2[2,c(1:7)])
stayonepas<-sum(dfstay2[3,c(1:7)])
stayonew<-c(stayonewpro,stayonedet,stayonepas)
stayonewpro<-sum(dfstay2[1,c(1:14)])
stayonedet<-sum(dfstay2[2,c(1:14)])
stayonepas<-sum(dfstay2[3,c(1:14)])
staythreew<-c(stayonewpro,stayonedet,stayonepas)
stay<-data.frame(stayonew,staythreew)
stay[,1]<-stay[,1]/sum(stay[,1])*100
stay[,2]<-stay[,2]/sum(stay[,2])*100
```

```
par(mar=c(5,5,3,1))
colnames(stay)<-c("One Week","Two Weeks")
barplot(as.matrix(stay),beside = T,
      col=c("purple","brown","chartreuse"),
      cex.lab=1.3,cex.names = 1.3,cex.axis = 1.2,ylim=c(0,80)
    )
legend("topleft", legend = c("Promoter","Detractor","Passive"),bty="n",
     fill=c("purple","brown","chartreuse"),ncol=3,cex=1.3)
title(main = "Effect of length stay in hotels", xlab =
      "Length of Stay", ylab = "Percentage of NPS_Type (%)",cex.lab=1.3,cex.main=1.5)
```

######## Effects of purpose visit on NPS Types

```
### rechange Najaf code for 1 and 0
## nature of visit
a<-tapply(combineddata$POV_CODE_C=="BUSINESS", combineddata$NPS_Type, sum)
a
b<-tapply(combineddata$POV_CODE_C=="LEISURE", combineddata$NPS_Type,sum)
b
name<-c("BUSINESS","LEISURE")
df<-rbind(a,b)
df2<-t(df)
colnames(df2)<-name
df2
par(mar=c(6,6,3,1))
barplot(df2,col=c("deeppink","gold1","darkviolet"),
      beside = T,
      ylim=c(0,3500),cex.names = 1.3,cex.axis = 1.01,cex.lab=1.2
     )
legend("topright", legend = c("Detractor","Passive","Promoter"),bty="n",
     fill=c("deeppink","gold1","darkviolet"),ncol=3,cex=1.3)
title(main = "Effect of visit purpose on NPS", xlab="Purpose of Visit"
     , ylab = "Number of travellers",cex.lab=1.5,cex.main=1.5)
```

look at relationship between two variables######

```
plot(combineddata$Likelihood_Recommend_H,combineddata$Customer_SVC_H,cex=1.5,cex.a
xis=1.2,
pch=21,xlab="Likelihood_Recommend_H",ylab="Customer_SVC_H",cex.lab=1.5,bg=2,col="bl
ack"
   ,main="Relationship Between Likelihood and Customer Service")
```

Annex 3

####CODE BOOK FOR LINEAR MODELLING#####
##DEPENDENT VARIABLE: LIKELIHOOD TO RECOMMEND##

 # Changing Service Variables into Dummy variables

```
combineddata$Convention_PL <- (gsub("Y", "1",combineddata$Convention_PL))
```

```
combineddata$Convention_PL <- (gsub("N", "0",combineddata$Convention_PL))
combineddata$Convention_PL<- as.numeric((gsub("\\ ", "NA",combineddata$Convention_PL)))

combineddata$Fitness.Trainer_PL <- (gsub("Y", "1",combineddata$Fitness.Trainer_PL))
combineddata$Fitness.Trainer_PL <- (gsub("N", "0",combineddata$Fitness.Trainer_PL))
combineddata$Fitness.Trainer_PL<- as.numeric((gsub("\\ ",
"NA",combineddata$Fitness.Trainer_PL)))

combineddata$Golf_PL <- (gsub("Y", "1",combineddata$Golf_PL))
combineddata$Golf_PL <- (gsub("N", "0",combineddata$Golf_PL))
combineddata$Golf_PL<- as.numeric((gsub("\\ ", "NA",combineddata$Golf_PL)))

combineddata$Pool.Indoor_PL <- (gsub("Y", "1",combineddata$Pool.Indoor_PL))
combineddata$Pool.Indoor_PL <- (gsub("N", "0",combineddata$Pool.Indoor_PL))
combineddata$Pool.Indoor_PL<- as.numeric((gsub("\\ ",
"NA",combineddata$Pool.Indoor_PL)))

combineddata$Pool.Outdoor_PL <- (gsub("Y", "1",combineddata$Pool.Outdoor_PL))
combineddata$Pool.Outdoor_PL <- (gsub("N", "0",combineddata$Pool.Outdoor_PL))
combineddata$Pool.Outdoor_PL<- as.numeric((gsub("\\ ",
"NA",combineddata$Pool.Outdoor_PL)))

combineddata$Regency.Grand.Club_PL <- (gsub("Y",
"1",combineddata$Regency.Grand.Club_PL))
combineddata$Regency.Grand.Club_PL <- (gsub("N",
"0",combineddata$Regency.Grand.Club_PL))
combineddata$Regency.Grand.Club_PL<- as.numeric((gsub("\\ ",
"NA",combineddata$Regency.Grand.Club_PL)))

combineddata$Self.Parking_PL <- (gsub("Y", "1",combineddata$Self.Parking_PL))
combineddata$Self.Parking_PL <- (gsub("N", "0",combineddata$Self.Parking_PL))
combineddata$Self.Parking_PL<- as.numeric((gsub("\\ ",
"NA",combineddata$Self.Parking_PL)))

combineddata$Shuttle.Service_PL <- (gsub("Y", "1",combineddata$Shuttle.Service_PL))
combineddata$Shuttle.Service_PL <- (gsub("N", "0",combineddata$Shuttle.Service_PL))
combineddata$Shuttle.Service_PL<- as.numeric((gsub("\\ ",
"NA",combineddata$Shuttle.Service_PL)))

combineddata$Spa_PL <- (gsub("Y", "1",combineddata$Spa_PL))
combineddata$Spa_PL <- (gsub("N", "0",combineddata$Spa_PL))
combineddata$Spa_PL<- as.numeric((gsub("\\ ", "NA",combineddata$Spa_PL)))
```

```
combineddata$Valet.Parking_PL <- (gsub("Y", "1",combineddata$Valet.Parking_PL))
combineddata$Valet.Parking_PL <- (gsub("N", "0",combineddata$Valet.Parking_PL))
combineddata$Valet.Parking_PL<- as.numeric((gsub("\\ ",
"NA",combineddata$Valet.Parking_PL)))
```

Coding control variables as dummies
##Gender

```
combineddata$Gender_H <- (gsub("Female", "1",combineddata$Gender_H))
combineddata$Gender_H <- (gsub("Male", "0",combineddata$Gender_H))
combineddata$Gender_H <- as.numeric((gsub("Prefer not to answer",
"NA",combineddata$Gender_H)))
combineddata$Gender_H <- as.numeric((gsub("\\ ", "NA",combineddata$Gender_H)))
```

Nature of visit

```
combineddata$POV_CODE_C <- (gsub("BUSINESS", "1",combineddata$POV_CODE_C))
combineddata$POV_CODE_C <- (gsub("LEISURE", "0",combineddata$POV_CODE_C))
combineddata$POV_CODE_C <- as.numeric((gsub("\\ ",
"NA",combineddata$POV_CODE_C)))
```

##Region

```
combineddata$Location_PL <- (gsub("Urban", "1",combineddata$Location_PL))
combineddata$Location_PL <- (gsub("Suburban", "0",combineddata$Location_PL))
combineddata$Location_PL <- (gsub("Airport", "NA",combineddata$Location_PL))
combineddata$Location_PL <- (gsub("Resort", "NA",combineddata$Location_PL))
combineddata$Location_PL <- as.numeric((gsub("\\ ", "NA",combineddata$Location_PL)))
```

#remove NAs

```
combineddata <- combineddata[!is.na(combineddata$Fitness.Trainer_PL),]
combineddata <- combineddata[!is.na(combineddata$Golf_PL),]
combineddata <- combineddata[!is.na(combineddata$Regency.Grand.Club_PL),]
combineddata <- combineddata[!is.na(combineddata$Valet.Parking_PL),]
combineddata <- combineddata[!is.na(combineddata$Valet.Parking_PL),]
combineddata <- combineddata[!is.na(combineddata$Pool.Indoor_PL),]
combineddata <- combineddata[!is.na(combineddata$Pool.Outdoor_PL),]
combineddata <- combineddata[!is.na(combineddata$Self.Parking_PL),]
combineddata <- combineddata[!is.na(combineddata$Shuttle.Service_PL),]
combineddata <- combineddata[!is.na(combineddata$Spa_PL),]
```

##Adding other satisfaction metric variables
#removing all NAs

```
combineddata <- combineddata[!is.na(combineddata$Guest_Room_H),]
combineddata <- combineddata[!is.na(combineddata$Customer_SVC_H),]
combineddata <- combineddata[!is.na(combineddata$Internet_Sat_H),]
```

```
combineddata <- combineddata[!is.na(combineddata$Condition_Hotel_H),]
combineddata <- combineddata[!is.na(combineddata$Staff_Cared_H),]
combineddata <- combineddata[!is.na(combineddata$Check_In_H),]
combineddata <- combineddata[!is.na(combineddata$F.B_Overall_Experience_H),]
```

##Difference Between quoted room rate and actual paid rates
#Converting to local exchange

```
combineddata$exchangerate[combineddata$Currency_PL == "CNY"] <- 0.1611
combineddata$exchangerate[combineddata$Currency_PL == "HKD"] <- 0.12897
combineddata$exchangerate[combineddata$Currency_PL == "TWD"] <- 0.03164
combineddata$Revdiff <-
round(combineddata$QUOTED_RATE_C*combineddata$exchangerate -
combineddata$PMS_ROOM_REV_USD_C/combineddata$LENGTH_OF_STAY_C, digits = 2)
combineddata <- combineddata[which(combineddata$Revdiff > 0),]
```

#Regression modelling

```
model1 <- lm(formula=combineddata$Likelihood_Recommend_H ~ combineddata$Gender_H +
combineddata$POV_CODE_C + combineddata$Location_PL)
model2 <- lm(formula=combineddata$Likelihood_Recommend_H ~ combineddata$Gender_H +
combineddata$POV_CODE_C + combineddata$Location_PL + combineddata$Spa_PL +
combineddata$Shuttle.Service_PL + combineddata$Self.Parking_PL +
combineddata$Pool.Outdoor_PL + combineddata$Pool.Indoor_PL +
combineddata$Valet.Parking_PL + combineddata$Convention_PL +
combineddata$Valet.Parking_PL + combineddata$Regency.Grand.Club_PL +
combineddata$Golf_PL + combineddata$Fitness.Trainer_PL , data=combineddata)
model3 <- lm(formula=combineddata$Likelihood_Recommend_H ~ combineddata$Gender_H +
combineddata$POV_CODE_C + combineddata$Location_PL + combineddata$Spa_PL +
combineddata$Shuttle.Service_PL + combineddata$Self.Parking_PL +
combineddata$Pool.Outdoor_PL + combineddata$Pool.Indoor_PL +
combineddata$Valet.Parking_PL + combineddata$Convention_PL +
combineddata$Valet.Parking_PL + combineddata$Regency.Grand.Club_PL +
combineddata$Golf_PL + combineddata$Fitness.Trainer_PL + combineddata$Guest_Room_H
+ combineddata$Customer_SVC_H + combineddata$Internet_Sat_H +
combineddata$Condition_Hotel_H + combineddata$Staff_Cared_H +
combineddata$Check_In_H + combineddata$F.B_Overall_Experience_H +
combineddata$Revdiff , data=combineddata)
model4 <- lm(formula=combineddata$Likelihood_Recommend_H ~ combineddata$Gender_H +
combineddata$POV_CODE_C + combineddata$Location_PL + combineddata$Guest_Room_H
+ combineddata$Customer_SVC_H + combineddata$Internet_Sat_H +
combineddata$Condition_Hotel_H + combineddata$Staff_Cared_H +
combineddata$Check_In_H + combineddata$F.B_Overall_Experience_H +
combineddata$Revdiff , data=combineddata)
```

#Installing Stargate to obtain regression output in a tabular form
```
#install.packages("stargazer")
library(stargazer)
stargazer(model1, model2, model3, model4, type="html", out="models.htm")
```

#making a new dataframe with coefficients
```
plotdata.sig <- read.csv("plot_co2.csv")
```

#Making a boxplot
```
plot_coeffsig <- ggplot(data=plotdata.sig, aes(x=plotdata.sig$Dependent.Variables,
y=plotdata.sig$Coefficients)) + geom_bar(stat="identity",fill="orange",color="black") +
coord_flip() + theme_minimal()
plot_coeffsig <- plot_coeffsig + ggtitle("Factors Affecting The Likelihood to Recommend") +
labs(x = "Variables", y = "Regression Coefficients")
plot_coeffsig
```

Annex 4

KSVM Modeling analysis######
 # Create train and test data sets
```
set.seed(1000)
dim(combineddataCat)
randindex<-sample(1:dim(combineddataCat)[1])
randindex
summary(randindex)
length(randindex)
cutpoint_2_3<-floor(2*dim(combineddataCat)[1]/3)
cutpoint_2_3
traindata<-combineddataCat[randindex[1:cutpoint_2_3],]
testdata<-combineddataCat[randindex[(cutpoint_2_3+1):dim(combineddataCat)[1]],]
```
#################### Running for KSVM Model
KSVM 1
```
library(kernlab)
##### test for KSVM for NPS Type
independentvari<-c("Overall_Sat_H","Guest_Room_H",
"Condition_Hotel_H","Customer_SVC_H","Staff_Cared_H",
 "Internet_Sat_H","F.B_Overall_Experience_H")
dependentvari<-"NPS_Type"
## create ksvm model 1
svmoutput<-ksvm(NPS_Type~Overall_Sat_H+Condition_Hotel_H+
Customer_SVC_H+Staff_Cared_H+Internet_Sat_H+F.B_Overall_Experience_H,
kernel ="rbfdot",kpar="automatic",C=35,cross=5,
prob.model=TRUE
,data=traindata)
## make predictions
testdata$ksvm_pred<-predict(svmoutput,testdata[,independentvari])
## Percent accuracy:
```

```
testdata$ksvm_error <- ifelse(testdata[,dependentvari]==testdata$ksvm_pred,TRUE,FALSE)
sum(testdata$ksvm_error)/dim(testdata)[1]*100
## to look at summary of total detractor, promotors and passive in a table
table(testdata[,dependentvari])
## comparison btw model prediction and observation
b<-table(testdata[,dependentvari],testdata$ksvm_pred)
 library(ggplot2)
ksvm_plot<-ggplot(testdata, aes(x=Overall_Sat_H, y=Staff_Cared_H)) +
geom_point(aes(color=ksvm_error,shape=ksvm_pred,size=ksvm_error)) +
labs( x="Overall_Sat_H", y="Staff_Cared_H", size="Error",
col="Error", shape="Predictions",title="KSVM-regression")+
theme_bw()+
theme(legend.justification=c(1,1),
legend.position=c(0.99,0.99),axis.text=element_text(size=14,face="bold"),
plot.title=element_text(size=16,face="bold"),
axis.title=element_text(size=16,face="bold"),
 legend.text=element_text(size=14),legend.title=element_text(size=16))          # Position
legend in bottom right
 ksvm_plot
##### making bar plot to show KSVM model results
b
## for observations
Detractor<-sum(b[1,])
Passive<-sum(b[2,])
Promotor<-sum(b[3,])
Observations<-c(Detractor,Passive,Promotor)
## for predictions
Detractor<-sum(b[,1])
Passive<-sum(b[,2])
Promotor<-sum(b[,3])
KSVM_Predictions<-c(Detractor,Passive,Promotor)
c<-data.frame(Observations,KSVM_Predictions)
rownames(c)<-c("Detractor","Passive","Promotor")
c2<-t(c)
barplot(as.matrix(c2),beside = T,col=c("green","red"),
      ylim=c(0,1200),cex.names = 1.5,cex.axis = 1.2,ylab="Count",cex.lab=1.5)
legend("topleft",legend = rownames(c2), bty="n",fill =c("green","red"),cex=1.5)
```

################################ Second ksvm to get fewer mistake

```
## create ksvm model
## with increasing c ("cost of constraints")it incraese separation and margin, but with mistakes
## with decreasing c and incraesing cross validation we get less error
## cross validation avoids overfitting
independentvari<-c("Overall_Sat_H","Customer_SVC_H","Staff_Cared_H","Internet_Sat_H")
svmoutput<-ksvm(NPS_Type~Overall_Sat_H+
            Customer_SVC_H+Staff_Cared_H+Internet_Sat_H,
```

```
kernel ="rbfdot",kpar="automatic",C=25,cross=8,
prob.model=TRUE
,data=traindata)
## make predictions
testdata$ksvm_pred<-predict(svmoutput,testdata[,independentvari])
print(testdata$ksvm_pred)
## Percent accuracy:
testdata$ksvm_error  <- ifelse(testdata[,dependentvari]==testdata$ksvm_pred,TRUE,FALSE)
sum(testdata$ksvm_error)/dim(testdata)[1]*100
## to look at summary  of total detractor, promotors and passive in a table
table(testdata[,dependentvari])
## comparison btw model prediction and observation
table(testdata[,dependentvari],testdata$ksvm_pred)
# library(ggplot2)
ksvm_plot<-ggplot(testdata,  aes(x=Overall_Sat_H,  y=Staff_Cared_H))  +
geom_point(aes(color=ksvm_error,shape=ksvm_pred,size=ksvm_error))  +
labs( x="Overall_Sat_H",  y="Staff_Cared_H",  size="Error",
col="Error",  shape="Predictions",title="KSVM-regression")+
theme_bw()+
theme(legend.justification=c(1,1),
legend.position=c(0.99,0.99),axis.text=element_text(size=14,face="bold"),
plot.title=element_text(size=16,face="bold"),
 axis.title=element_text(size=16,face="bold"),
 legend.text=element_text(size=14),legend.title=element_text(size=16))              # Position
legend in bottom right
ksvm_plot
```

################################## If we DON'T not work with categorized variables, and work with numeric variables how is the results!

```
# Create train and test data sets
set.seed(1000)
dim(combineddata)
randindex<-sample(1:dim(combineddata)[1])
randindex
summary(randindex)
length(randindex)
cutpoint_2_3<-floor(2*dim(combineddata)[1]/3)
cutpoint_2_3
traindata<-combineddata[randindex[1:cutpoint_2_3],]
testdata<-combineddata[randindex[(cutpoint_2_3+1):dim(combineddata)[1]],]
combineddata$NPS_Type
############################
#library(kernlab)
##### test for KSVM for NPS Type
independentvari<-c("Overall_Sat_H","Guest_Room_H","Condition_Hotel_H",
"Condition_Hotel_H","Customer_SVC_H","Staff_Cared_H",
"Internet_Sat_H","F.B_Overall_Experience_H")
```

```
dependentvari<-"NPS_Type"
## create ksvm model
svmoutput<-ksvm(NPS_Type~Overall_Sat_H+Guest_Room_H+Condition_Hotel_H+
Customer_SVC_H+Staff_Cared_H+Internet_Sat_H+F.B_Overall_Experience_H,
kernel ="rbfdot",kpar="automatic",C=1,cross=0,
prob.model=TRUE
,data=traindata)
## make predictions
testdata$ksvm_pred<-predict(svmoutput,testdata[,independentvari])
## Percent accuracy:
testdata$ksvm_error <- ifelse(testdata[,dependentvari]==testdata$ksvm_pred,TRUE,FALSE)
sum(testdata$ksvm_error)/dim(testdata)[1]*100
## to look at summary of total detractor, promotors and passive in a table
table(testdata[,dependentvari])
## comparison btw model prediction and observation
table(testdata[,dependentvari],testdata$ksvm_pred)
# library(ggplot2)
ksvm_plot<-ggplot(testdata, aes(x=Overall_Sat_H, y=Staff_Cared_H)) +
geom_point(aes(color=ksvm_error,shape=ksvm_pred,size=ksvm_error)) +
labs( x="Overall_Sat_H", y="Staff_Cared_H", size="Error",
col="Error", shape="Predictions",title="KSVM-regression")+
theme_bw()+
theme(legend.justification=c(1,1),
legend.position=c(0.99,0.99),axis.text=element_text(size=14,face="bold"),
plot.title=element_text(size=16,face="bold"),
axis.title=element_text(size=16,face="bold"),
legend.text=element_text(size=14),legend.title=element_text(size=16))          # Position
legend in bottom right
ksvm_plot
```

Annex 5

##CODE BOOK FOR ASSOCIATION RULES##

##combineAll is the combined dataset for all the months

```
combineAll <- rbind(combine2_M, combine2_R, combine2_Y, combine2_N)
View(combineAll)
```

#Creating a subset of data set for the selected variables

```
dummyset <- combineAll[,c(52,55,56,57,58,64,65,67)]
View(dummyset)
```

#storing a backup of the data set

```
julyselect <- dummyset
```

#Changing the names of the data to get a clear result in the transaction matrix

```
dummyset$Overall_Sat_H <- (gsub("8",  "P_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("9",  "P_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("10",  "P_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("6",  "N_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("7",  "N_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("1",  "D_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("2",  "D_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("3",  "D_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("4",  "D_Overall_Sat",dummyset$Overall_Sat_H))
dummyset$Overall_Sat_H <- (gsub("5",  "D_Overall_Sat",dummyset$Overall_Sat_H))

dummyset$Guest_Room_H <- (gsub("8",  "P_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("9",  "P_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("10",  "P_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("6",  "N_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("7",  "N_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("1",  "D_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("2",  "D_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("3",  "D_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("4",  "D_Guest_Room",dummyset$Guest_Room_H))
dummyset$Guest_Room_H <- (gsub("5",  "D_Guest_Room",dummyset$Guest_Room_H))

dummyset$Customer_SVC_H<- (gsub("8",  "P_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("9",  "P_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("10",  "P_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("6",  "N_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("7",  "N_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("1",  "D_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("2",  "D_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("3",  "D_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("4",  "D_Customer_SVC", dummyset$Customer_SVC_H))
dummyset$Customer_SVC_H<- (gsub("5",  "D_Customer_SVC", dummyset$Customer_SVC_H))

dummyset$Internet_Sat_H<- (gsub("8",  "P_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("9",  "P_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("10",  "P_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("6",  "N_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("7",  "N_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("1",  "D_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("2",  "D_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("3",  "D_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("4",  "D_Internet_Sat", dummyset$Internet_Sat_H))
dummyset$Internet_Sat_H<- (gsub("5",  "D_Internet_Sat", dummyset$Internet_Sat_H))

dummyset$Condition_Hotel_H<-  (gsub("8",  "P_Condition_Hotel",  dummyset$Condition_Hotel_H))
dummyset$Condition_Hotel_H<-  (gsub("9",  "P_Condition_Hotel",  dummyset$Condition_Hotel_H))
```

```
dummyset$Condition_Hotel_H<- (gsub("10", "P_Condition_Hotel", dummyset$Condition_Hotel_H))
dummyset$Condition_Hotel_H<- (gsub("6", "N_Condition_Hotel", dummyset$Condition_Hotel_H))
dummyset$Condition_Hotel_H<- (gsub("7", "N_Condition_Hotel", dummyset$Condition_Hotel_H))
dummyset$Condition_Hotel_H<- (gsub("1", "D_Condition_Hotel", dummyset$Condition_Hotel_H))
dummyset$Condition_Hotel_H<- (gsub("2", "D_Condition_Hotel", dummyset$Condition_Hotel_H))
dummyset$Condition_Hotel_H<- (gsub("3", "D_Condition_Hotel", dummyset$Condition_Hotel_H))
dummyset$Condition_Hotel_H<- (gsub("4", "D_Condition_Hotel", dummyset$Condition_Hotel_H))
dummyset$Condition_Hotel_H<- (gsub("5", "D_Condition_Hotel", dummyset$Condition_Hotel_H))

dummyset$Staff_Cared_H<- (gsub("8", "P_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("9", "P_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("10", "P_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("6", "N_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("7", "N_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("1", "D_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("2", "D_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("3", "D_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("4", "D_Staff_Cared", dummyset$Staff_Cared_H))
dummyset$Staff_Cared_H<- (gsub("5", "D_Staff_Cared", dummyset$Staff_Cared_H))

dummyset$F.B_Overall_Experience_H<- (gsub("8", "P_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("9", "P_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("10", "P_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("6", "N_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("7", "N_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("1", "D_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("2", "D_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("3", "D_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("4", "D_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
dummyset$F.B_Overall_Experience_H<- (gsub("5", "D_F.B_Overall_Exp",
dummyset$F.B_Overall_Experience_H))
```

#Creating 3 different data sets for promoters detractors and passive
#Dataset for promoter's data

```
dummysetPro <- dummyset[which(dummyset$NPS_Type == "Promoter"),]
dummysetPro <- dummysetPro[,-1]
View(dummysetPro)
```

```
#Dataset for detractor's data
dummysetDetrac <- dummyset[which(dummyset$NPS_Type == "Detractor"),]
dummysetDetrac <- dummysetDetrac[,-1]
View(dummysetDetrac)

#dataset for passive data
dummysetPass <- dummyset[which(dummyset$NPS_Type == "Passive"),]
dummysetPass <- dummysetPass[,-1]
View(dummysetPass)


#creating comma separated excel sheets for each of the 3 data sets
write.csv(dummysetPro, file = "dummysetPro.csv")
write.csv(dummysetDetrac, file = "dummysetDetrac.csv")
write.csv(dummysetPass, file = "dummysetPass.csv")

#installing the arules package
install.packages("arules")
library(arules)
require(arules)

#reading the transactions and creating a data matrix for promoter data
choice <- read.transactions("dummysetPro.csv", sep = ",")
choice
#summary of the promoter's transaction matrix
summary(choice)
#plotting  the graph for the significant  data wrt support values
itemFrequencyPlot(choice,  topN = 16)


#reading the transactions and creating a data matrix for passive data
choicePass <- read.transactions("dummysetPass.csv", sep = ",")
choicePass
#summary of the passive's transaction matrix
summary(choicePass)
#plotting the graph for the significant data wrt support values
itemFrequencyPlot(choicePass, topN = 23)

#reading the transactions and creating a data matrix for detractor data
choiceDetrac <- read.transactions("dummysetDetrac.csv", sep = ",")
choiceDetrac
#summary of the detractor's transaction matrix
summary(choiceDetrac)
#plotting the graph for the significant data wrt support values
itemFrequencyPlot(choiceDetrac, topN = 24)
```

Appendix

Observations/*Prediction*	***Detractor***	***Passive***	***Promotor***
Detractor	72	53	6
Passive	17	346	65
Promotor	1	92	1228

Table A1. Confusion matrix indicating KSVM performance in depicting NPS Types

Figure A2. KSVM Model predictions for NPS Types versus overall satisfaction and staff care variables with the scenario if variables are numeric not categorized

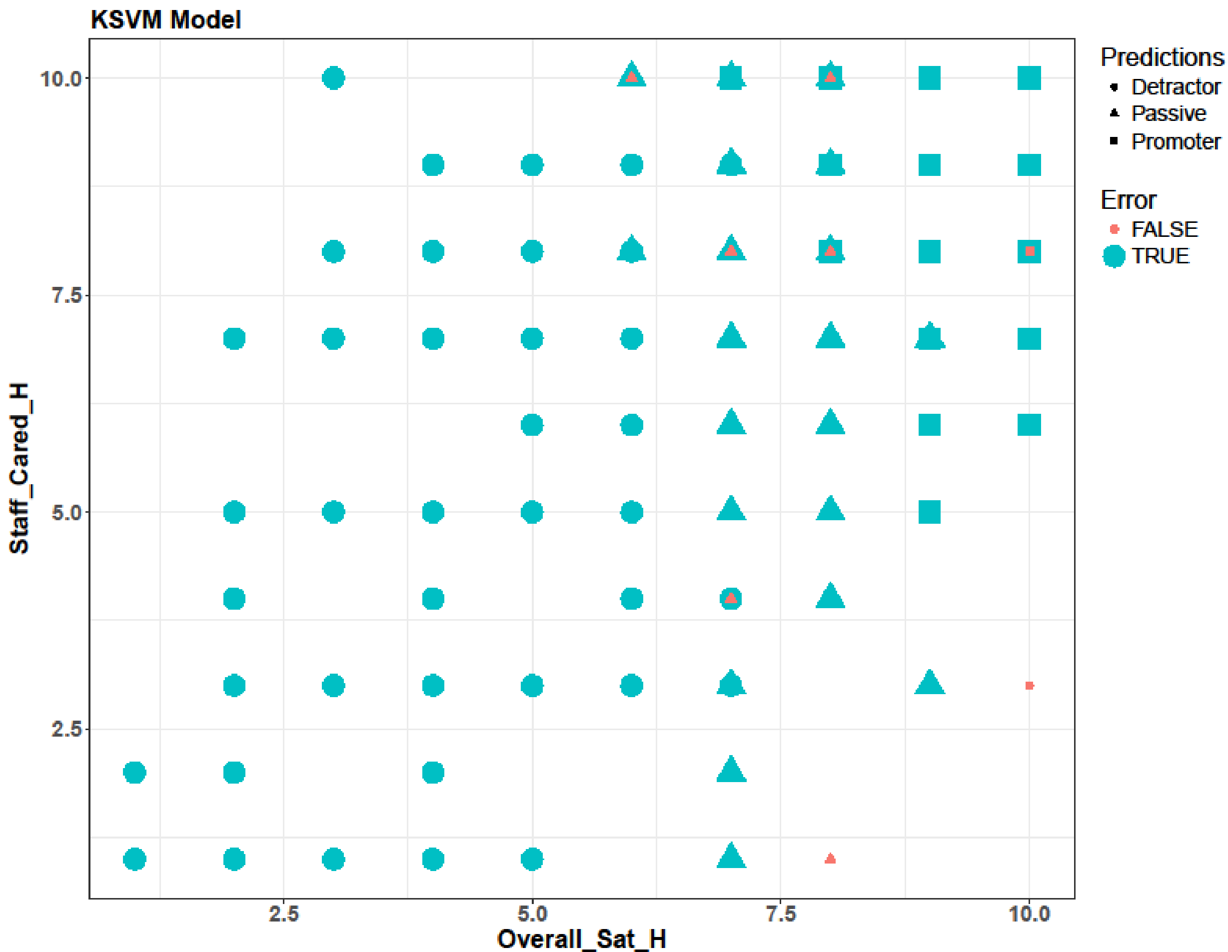


Figure A3. KSVM performance in terms of training error and cross validation

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

SV type: C-svc  (classification)
 parameter : cost C = 25

Gaussian Radial Basis kernel function.
 Hyperparameter : sigma =  0.571428571428571

Number of Support Vectors : 957

Objective Function Value : -4559.686 -2406.934 -14559.72
Training error : 0.101091
Cross validation error : 0.122106
Probability model included.
>
```