
HYATT HOTELS FINAL REPORT



DECEMBER 7, 2017
IST 687 FINAL PROJECT REPORT



Table of Contents

Part 1: Background Summary and Group Decision Making

Part 2: Business Rules and Assumptions

Part 3: Data Cleaning/Preparation

Part 4: Data Analysis and Descriptive Statistics

Part 5: Results Validation

Part 6: Recommendations

Part 7: Code Used

Background Summary and Group Decision Making

Hyatt Hotels decided to perform an extensive consumer survey that occurred from February 2014 until January 2015. The information gathered through the survey reflected several different categories including personal customer data, ratings of hotel personnel, hotel services, hotel amenities and geographical hotel locations. Customers were asked to rate the hotel as exceptional, adequate or lacking. From here Hyatt Hotels calculated the NPS, or Net Promoter Score of customers by labeling their submissions as a “promoter”, “detractor” or “passive”. Someone who is a promoter is classified as someone who would recommend the hotel to others, a person who is a detractor is someone who would not recommend the hotel to others and a passive person has no strong opinions in either direction about the hotel. This calculation of NPS type detailed a summary of customer’s attitudes towards the hotel. We used several statistical visualization and comparison techniques to gain a better understanding of the data collected in regards to the resulting calculation of the customers’ likelihood to recommend. First we took four months of the dataset, one from each season, to gain a better understanding of customer feelings throughout the year. After the analysis of these four months we found several connections between the hotel services, ratings of hotel personnel, hotel amenities, geographical hotel locations and customer differences that impacted the customers’ likelihood to recommend. Our group’s analysis work was done in R studio which is a data program used to clean, analyze and plot data.

Business Rules and Assumptions

In order to analyze the data successfully our group decided to make a few assumptions that helped us in the report.

Assumption 1: Everyone who took the survey completed it with honesty and integrity.

Assumption 2: It is possible to enhance the experience of detractors and passives to change them to promoters overtime.

Assumption 3: Analyzing four out of twelve months of data will give us enough information to provide solutions to the hotel's business practices.

Assumption 4: It is not possible to change the human variability of people staying in a given hotel, it is only possible to change the services and amenities of a given location.

Assumption 5: The categorical variables can be objectively used by people staying for either leisure or business or can be apparent in both.

Business Questions

Business Question #1: Which hotel services variables impact a "Business" customers' stay?

Business Question #2: Which hotel services variables impact a "Leisure" customers' stay?

Business Question #3: What state in the United States would be best to look at given their number of detractors and total number of respondents?

Business Question #4: Is the purpose of the respondents' visit a direct indicator on their NPS scores?

Data Preparation, Selection and Cleaning

Data Preparation/Selection

The hotel provided us with surveys over the course of 12 months. The files were very large individually and combined were definitely way more than enough information necessary to form proper analysis and insights. As a result of this our team decided to take 1/3 of the data given and analyzed the months of February, May, August and November. We selected these four months because they all reflect a different season in the calendar year.

Data Cleaning

The information was given to us in the form of .CSV, or excel based file which we read into R studio. We converted the excel files into workable R content by using the `read.csv` function in R studio. From there we pulled the necessary columns we decided to look at and pulled them into R studio. We selected 35 columns from the dataset. We only imported these 37 columns because we did not see the other columns as useful to our hotel analysis. After reading in the four months worth of data into R we needed to remove NAs from the likelihood to recommend column without replacement. To do this we used the `filter(febData, Likelihood_Recommmend_H != "NA")` function. We did this for all four months by substituting the appropriate month in the “febData” spot in the above example. From there we put the four months worth of data together using the “`rbind`” function in R. From there we determined that we were going to use the United States data since it seemed to give us the most consistent and abundant information. After combining the four months worth of data we ended up with 322,992 observations of 37 variables.

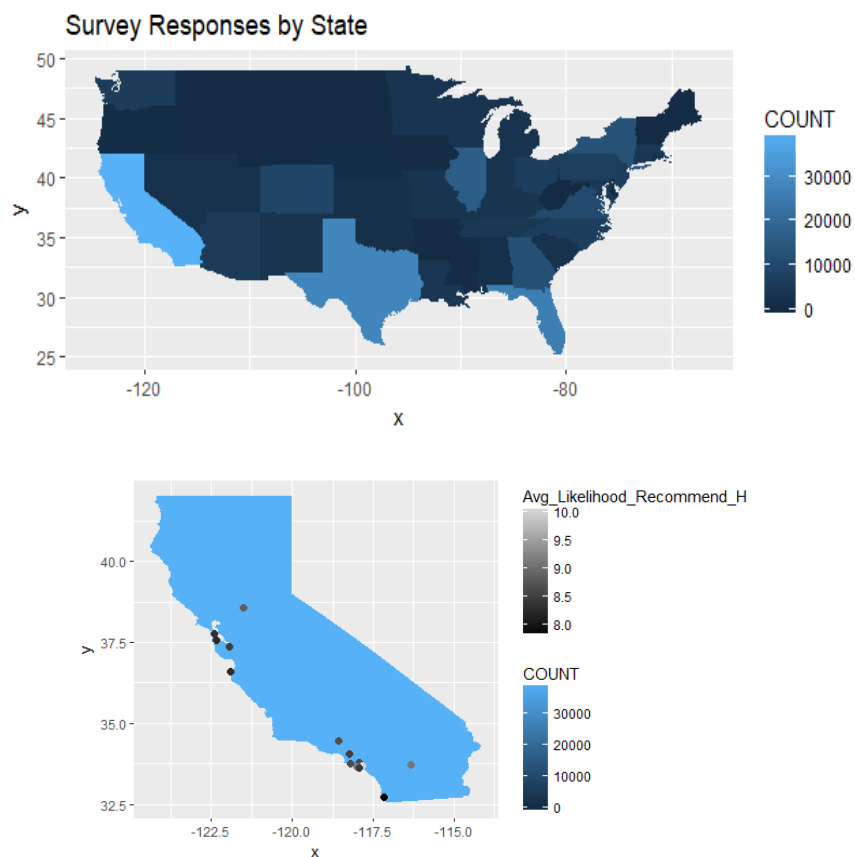
Column Selection and Definition

Column Name	Actual Definition
Brand_PL	Hotel's brand
Likelihood_Reommend_H	Likelihood to recommend metric; value on a 1 to 10 scale
Room_Type_H	Guest's room type code
Gender_H	Guest's gender
CHILDREN_NUM_C	Number of children indicated in the stay
Laundry_PL	Flag indicating if the hotel has laundry space
Conference_PL	Flag indicating if the hotel has a conference center nearby
NPS_Type	Indicates if the guests HySat responses mark them as a promoter, a passive or a detractor
Pool.Outdoor_PL	Flag indicating if the hotel has an outdoor pool
Resort_PL	Flag indicating if the hotel has a resort
Spa_PL	Flag indicating if the hotel has a spa
GP_Tier	GP tier of the guest
Staff_Cared_H	Staff cared metric; value on a 1 to 10 scale
Internet_Sat_H	Internet satisfaction metric; value on a 1 to 10 scale
Check_IN_H	Quality of the check in process metric; value on a 1 to 10 scale
Tranquility_H	Tranquility metric; value on a 1 to 10 scale
POV_CODE_C	Purpose of visit
Property.Latitude_PL	Latitude of the hotel's location
Property.Longitude_PL	Longitude of the hotel's location
Guest_Room_H	Guest room satisfaction metric, value on a 1 to 10 scale
Condition_Hotel_H	Condition of the hotel metric; value on a 1 to 10 scale
City_PL	City in which the hotel is located
State_PL	State in which the hotel is located
Country_PL	Country in which the hotel is located
COUNTRY_CODE_R	Country of the party making the reservation
Fitness.Center_PL	Flag indicating if the hotel has a fitness center
Golf_PL	Flag indicating if the hotel has a golfspace
Limo.Service_PL	Flag indicating if the hotel has limo service
Mini.Bar_PL	Flag indicating if the hotel has a minibar
NUM_ROOMS_R	Total number of rooms booked under the reservation
Pool.Indoor_PL	Flag indicating if the hotel has an indoor pool
Casino_PL	Flag indicating if the hotel has a casino
Convention_PL	Flag indicating if the hotel has a convention center
Dry.Cleaning_PL	Flag indicating if the hotel has dry cleaning
Business.Center_PL	Flag indicating if the hotel has a business center

Data Analysis using Descriptive Statistics

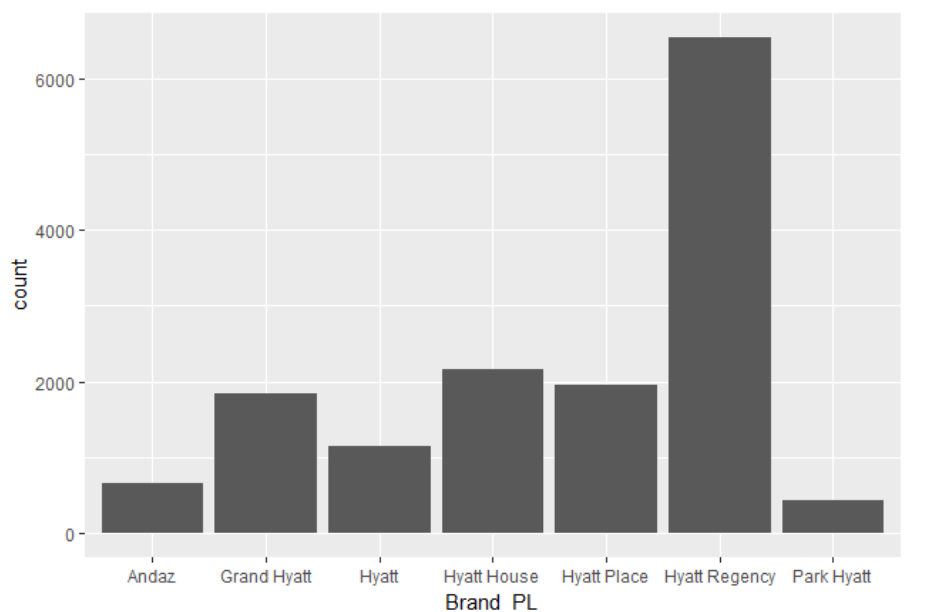
Statistical distribution of survey respondents in the United States

As a group we decided it would be best to look at the state with the most respondents to get the best understanding of the given data. In order to find this, we created a response map (shown below) which gave us the number of respondents across the United States. From the graph we saw California had the most respondents and decided to use them as a result. We also found that California had a large number of detractors in the dataset. These detractors are the ones who we are targeting in an effort to help improve the hotel's conditions later on. These numbers solidify our reasoning behind selecting California.



Statistical distribution of Hyatt Brand Hotels in California

Our group decided to examine which Hyatt Brand hotel was the most frequently surveyed hotel in California. The group decided to use a bar graph function to list all the respondents accordingly. Our bar graph shows that the Hyatt Regency gave us the greatest number of respondents. As a result we decided to further examine the Hyatt Regency trends throughout this study. Our group also took this one step further by taking the average likelihood to recommend numbers of each specific brand hotel. We did this by using the sqldf function in R. The results showed that the Hyatt Regency reviews were also the lowest likelihood to recommend and as a result we decided to continue our review as the Hyatt Regency brand in hopes of improving their NPS scores. We decided this because the only lower likelihood to recommend was the ambiguous “Hyatt” brand which does not exist.



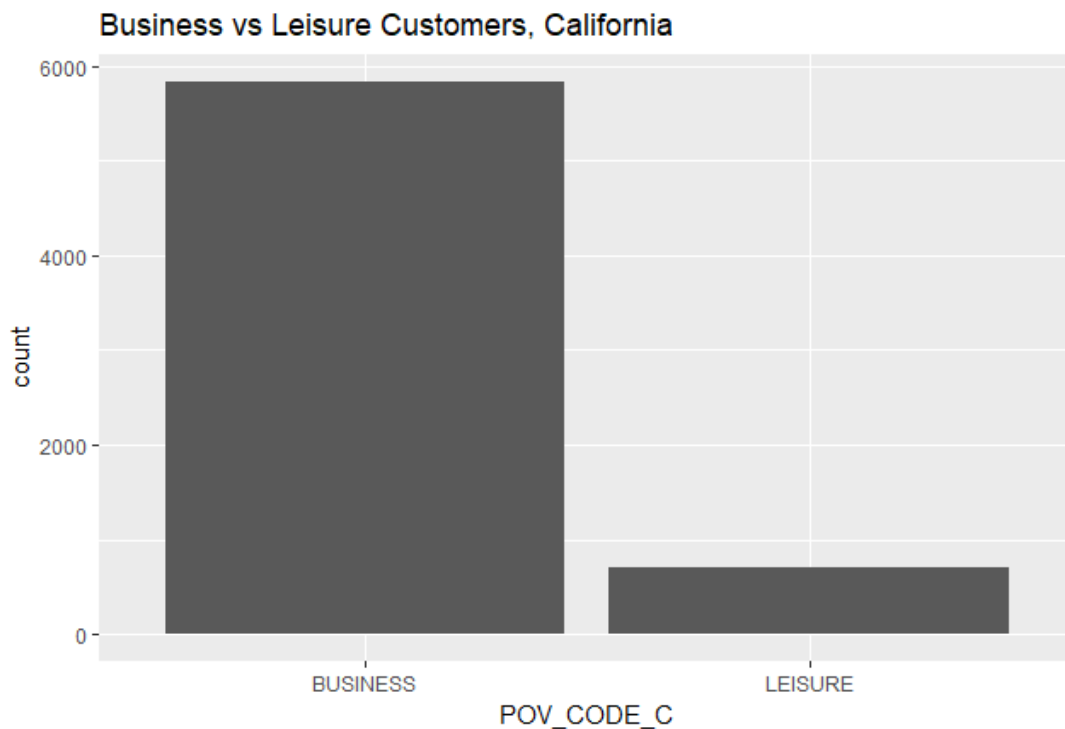
Brand_PL

AVG(Likelihood_Recommend_H)

1	Park Hyatt	9.043779
2	Hyatt House	8.935409
3	Grand Hyatt	8.935220
4	Hyatt Place	8.929887
5	Andaz	8.804511
6	Hyatt Regency	8.534180
7	Hyatt	8.216146

Distribution of Business v.s. Leisure & NPS in California

Our group then looked at the respondents' purpose of visit and compared them to their NPS Scores. After reviewing the data we found that a large majority of the people staying at the Hyatt Regency Hotel brand were doing so for business purposes as reflected by the graph below. We also ran the tapply function to show a comparison between their NPS scores and found there was not a significant different in the two's NPS. The people staying for business purposes reported a 8.518068 Likelihood to recommend while leisure stays reported a 8.668571. Leisure was slightly higher in comparison to business.



Purpose of Visit and Likelihood to Recommend Chart

	BUSINESS	LEISURE
Likelihood to Recommend	8.518068	8.668571

Linear Models Examples

Our group decided to use Linear Models as a way to if there were strong correlations between some of the variables. In linear models we look at the relationship of independent variables versus dependent variables. When running the models if there is a positive relationship in a linear model, as there was in our findings in our report, a high value for an independent variable will tend to indicate a high value for the dependent variable. In comparison, a low value for a dependent variable will tend to indicate a low value for the independent variable.

In our report we looked at likelihood to recommend as our dependent variable throughout the models. From there we classified the other variables we compared it to as the independent ones. For our independent variables we chose to look at the condition of the hotel, the amount of the staff cared, the internet, check in, tranquility and customer support variables. After running each of the variables against the likelihood to recommend variable we saw very strong r^2 values of 66%. Our model was able to successfully account for 66% of the variability in the data. The value of this is good when taking into human variability as it cannot always predetermine parameters.

Below is a snippet of our code output which will show the results of all of our independent variables.

```

Call:
lm(formula = Likelihood_Recommend_H ~ Condition_Hotel_H + Customer_SVC_H +
    Tranquility_H + Staff_Cared_H + Check_In_H + Internet_Sat_H,
    data = business)

Residuals:
    Min       1Q   Median       3Q      Max
-8.4545 -0.2905  0.0688  0.5074  8.7301

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   -1.866782   0.112575  -16.583 < 2e-16 ***
Condition_Hotel_H  0.403115   0.013172  30.603 < 2e-16 ***
Customer_SVC_H    0.399601   0.020923  19.099 < 2e-16 ***
Tranquility_H     0.217304   0.009662  22.490 < 2e-16 ***
Staff_Cared_H     0.120918   0.017456   6.927 4.77e-12 ***
Check_In_H       -0.009954   0.013282  -0.749  0.454
Internet_Sat_H    0.049566   0.006638   7.466 9.48e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.131 on 5701 degrees of freedom
Multiple R-squared:  0.6575, Adjusted R-squared:  0.6572
F-statistic: 1824 on 6 and 5701 DF, p-value: < 2.2e-16

```

A-Rules Analysis

The `apriori` function from the `aRules` packages in R Studio uses Bayes Theorem to find the probability of an event happening given that we know something else happens. In the case of our work The A-Rules helped us find the possibility of a passive, detractor or promoter given certain variables.

A-Rules gave us the possibility of the passive, or detractor score given certain variables. For example in our research we found that if a customer stays at a Hyatt Regency that has no business center, no mini-bar, no golf course and has limo service then there is a 17.6% chance that the customer will be a detractor. The rest of the output shows that this event occurred 98 times, or in 1.7% out of all business surveys for the California Hyatt Regency hotels.

Example 1

```
[4] {Business.Center_PL=N,  
    Golf_PL=N,  
    Limo.Service_PL=Y,  
    Mini.Bar_PL=N}    => {NPS_Type=Detractor} 0.017168886 0.1759425 1.390970 98
```

In a second example we looked at the event that the hotel has a convention center, does not have golf course, has limo service and does not have a minibar. As a result of this, there was a 16% chance that the NPS type of the customer will be a detractor. Again, the rest of our given output shows that this occurred 265 times, or 4.6% out of all the business surveys for the California Hyatt Regency hotels.

Example 2

```
[21] {Convention_PL=Y,  
    Golf_PL=N,  
    Limo.Service_PL=Y,  
    Mini.Bar_PL=N}    => {NPS_Type=Detractor} 0.046426069 0.1619804 1.280588 265
```

Naive Bayes Method

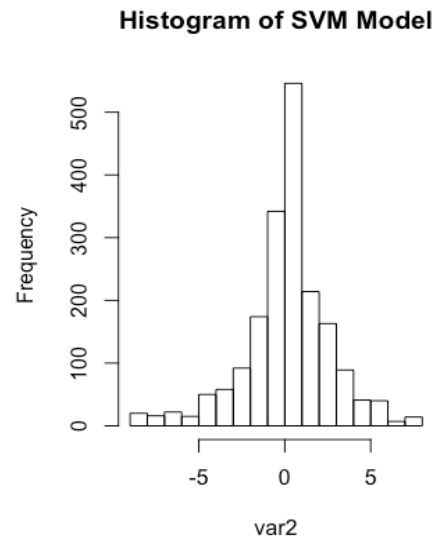
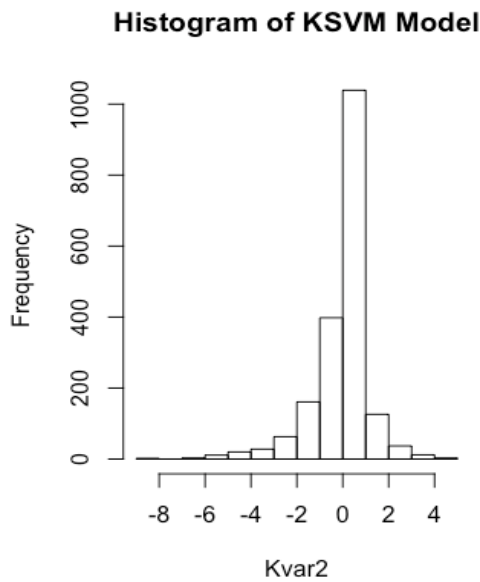
Naïve Bayes is used to see which predictors are assumed to be independent within a given class label. In our research we were trying to figure out the NPS Type from the existing patterns located in our data set. The end goal was to find out what the probability a customer's NPS Type would be based on variables. The variables we examined were Condition_Hotel_H, Num ,Customer_SVC_H_Num , Tranquility_H_Num , Staff_Cared_H_Num ,Check_In_H_Num , Internet_Sat_H_Num, Convention_PL_Num ,Business.Center_PL_Num , Spa_PL_Num and Mini.Bar_PL_Num. After running our predictive analysis, we were able to garner a "Errornb" or around 22.3% which is a good result.

```
> Errornb<- sum(testData1[, "NPS_Type"] != predAll) * 100 /length(predAll)
> Errornb
[1] 22.32925
> |
```

SVM and KSVM Plots

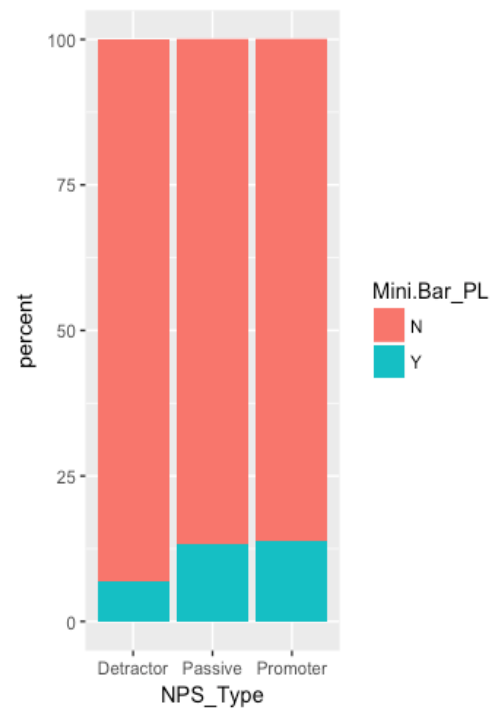
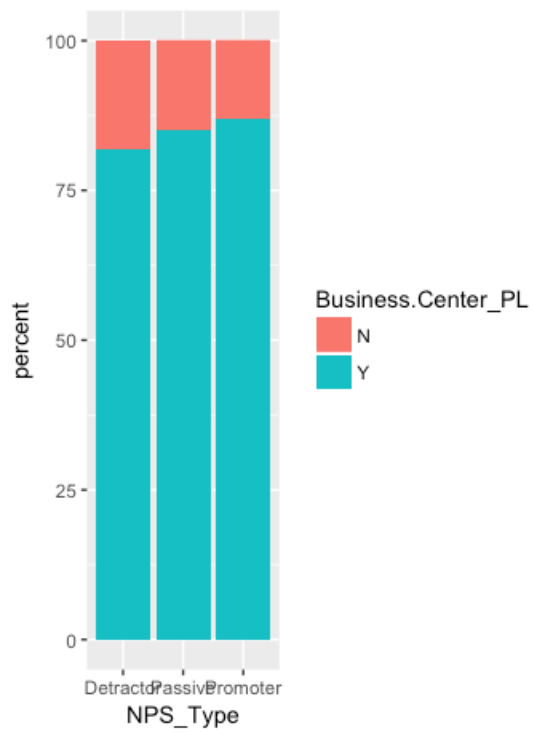
SVM and KSVM models are used to predict the values of a dependent variable based on training data and the predicted values of a test data set. The end goal of the plots is to receive a difference between the actual and predicted values that is low. When this value is low we are able to infer that the dependent variable will be accurately predicted by using independent variables in our data set.

We decided to use histograms of the different values in both the SVM and KSVM models. Our findings showed a lot of data centered around “0” on the histogram charts pictured below. This means that the models we ran will accurately predict the values of the dependent variable which for us is our NPS_Type. For specifics in our data set, the training error of the KSVM model we ran with the whole data set was 21%. This is considered a good value to predict data. In terms of our independent variables we looked at both categorical and numerical columns to ensure the model was predicted accurately.



Validation

In terms of overall project validation our group believes the models we included in this report were insightful and actionable. We ran several of the same models with a variety of different variables to ensure that all of our findings were supported effectively. In addition to this we accompanied several of our findings with visualizations to back up our results. In the examples below, these two charts regarding Business Centers and Mini Bars solidified our results that both had an effect on NPS score in our A-Rules results. In addition to the graphs, all of our findings showed very low error rates which ensure that the variables we were looking at will give Hyatt Hotels actionable insights.



Recommendations

After looking at the above data supplied to us by Hyatt Hotels we as a group have decided to make the following recommendations to Hyatt Hotels.

Recommendation #1: Improve Hotel Customer Service

Using the data we saw that there was a very high correlation between Hotel customer service and NPS type. Our recommendation would be to continue hotel customer service operations that are going well while potentially adding more representatives to ensure quality and speedy customer service.

Recommendation #2: Improve the condition of hotels

We saw a high correlation between hotel condition and NPS type. Our recommendation would be to ensure the condition of hotels meet or exceed those of the expectations of guests. With this Hyatt could run inspections every quarter to ensure all condition standards are met at a given location.

Recommendation #3: Ensure Mini-bars are available

Given the data we were able to see that certain hotel amenities affected the NPS scores of customers in the Hyatt Regency Hotels. Mini-Bars had a good effect on likelihood to recommend. We would recommend that Hyatt continues to keep mini-bars stocked and potentially even increase the supply of drinks in each mini-bar.

Recommendation #4: Make sure conference centers are available

Our data showed that conference centers had a great effect on likelihood to recommend in the hotels. We would recommend that conference centers are available in Hyatt Hotels. We would also recommend that the technology and furniture within these centers is up to par as well.

Complete Code

```
#install packages
install.packages("sqldf")
install.packages("ggplot2")
install.packages("plyr")
install.packages("dplyr")
install.packages("maps")
install.packages("arules")
install.packages("arulesViz")
install.packages("ggmap")
install.packages("gridExtra")
install.packages("kernlab")
install.packages("e1071")
install.packages("moments")

#Load Packages
library(sqldf)
library(ggplot2)
library(plyr)
library(dplyr)
library(maps)
library(arules)
library(arulesViz)
library(ggmap) #for geocoding
library(gridExtra)
library(kernlab)
library(e1071)
library(moments)

#WINDOWS
setwd("E:\\OneDrive\\Documents\\School Work\\Syracuse\\Applied Data
  Science\\project_data\\totalData")
#or...
setwd("E:\\FreyGeospatial\\Documents\\IST687-data\\IST687-data")

#MAC
setwd("/Users/jordanfrey/Documents/OneDrive/Documents/School Work/Syracuse/Applied Data
  Science/project_data/totalData")

#READ DATA

febData <- read.csv("out-201402.csv")[,c("Room_Type_H", "Gender_H", "CHILDREN_NUM_C",
  "Laundry_PL", "NUM_ROOMS_R", "Pool.Indoor_PL", "Casino_PL", "Convention_PL", "Dry.Cleaning_PL",
```

```
"Business.Center_PL",
      "Conference_PL", "NPS_Type", "Likelihood_Recommend_H", "Fitness.Center_PL",
"Golf_PL", "Limo.Service_PL", "Mini.Bar_PL",
      "Pool.Outdoor_PL", "Resort_PL", "Spa_PL", "GP_Tier", "COUNTRY_CODE_R",
"Country_PL", "State_PL", "City_PL", "Guest_Room_H", "Condition_Hotel_H", "Customer_SVC_H",
      "Staff_Cared_H", "Internet_Sat_H", "Check_In_H", "Tranquility_H",
"POV_CODE_C", "Brand_PL", "Property.Latitude_PL", "Property.Longitude_PL"]]
```

```
mayData <- read.csv("out-201405.csv")[,c("Room_Type_H", "Gender_H", "CHILDREN_NUM_C",
"Laundry_PL", "NUM_ROOMS_R", "Pool.Indoor_PL", "Casino_PL", "Convention_PL", "Dry.Cleaning_PL",
"Business.Center_PL",
      "Conference_PL", "NPS_Type", "Likelihood_Recommend_H", "Fitness.Center_PL",
"Golf_PL", "Limo.Service_PL", "Mini.Bar_PL",
      "Pool.Outdoor_PL", "Resort_PL", "Spa_PL", "GP_Tier", "COUNTRY_CODE_R",
"Country_PL", "State_PL", "City_PL", "Guest_Room_H", "Condition_Hotel_H", "Customer_SVC_H",
      "Staff_Cared_H", "Internet_Sat_H", "Check_In_H", "Tranquility_H",
"POV_CODE_C", "Brand_PL", "Property.Latitude_PL", "Property.Longitude_PL")]
```

```
augData <- read.csv("out-201408.csv")[,c("Room_Type_H", "Gender_H", "CHILDREN_NUM_C",
"Laundry_PL", "NUM_ROOMS_R", "Pool.Indoor_PL", "Casino_PL", "Convention_PL", "Dry.Cleaning_PL",
"Business.Center_PL",
      "Conference_PL", "NPS_Type", "Likelihood_Recommend_H", "Fitness.Center_PL",
"Golf_PL", "Limo.Service_PL", "Mini.Bar_PL",
      "Pool.Outdoor_PL", "Resort_PL", "Spa_PL", "GP_Tier", "COUNTRY_CODE_R",
"Country_PL", "State_PL", "City_PL", "Guest_Room_H", "Condition_Hotel_H", "Customer_SVC_H",
      "Staff_Cared_H", "Internet_Sat_H", "Check_In_H", "Tranquility_H",
"POV_CODE_C", "Brand_PL", "Property.Latitude_PL", "Property.Longitude_PL")]
```

```
novData <- read.csv("out-201411.csv")[,c("Room_Type_H", "Gender_H", "CHILDREN_NUM_C",
"Laundry_PL", "NUM_ROOMS_R", "Pool.Indoor_PL", "Casino_PL", "Convention_PL", "Dry.Cleaning_PL",
"Business.Center_PL",
      "Conference_PL", "NPS_Type", "Likelihood_Recommend_H", "Fitness.Center_PL",
"Golf_PL", "Limo.Service_PL", "Mini.Bar_PL",
      "Pool.Outdoor_PL", "Resort_PL", "Spa_PL", "GP_Tier", "COUNTRY_CODE_R",
"Country_PL", "State_PL", "City_PL", "Guest_Room_H", "Condition_Hotel_H", "Customer_SVC_H",
      "Staff_Cared_H", "Internet_Sat_H", "Check_In_H", "Tranquility_H",
"POV_CODE_C", "Brand_PL", "Property.Latitude_PL", "Property.Longitude_PL")]
```

#Interperet data

str(febData) #displays data frame structure, including the variables/factors

#make copies of data for distinct treatment of NAs and add month column

```
febData$Month <- "February"
```

```
mayData$Month <- "May"
```

```
augData$Month <- "August"
```

```
novData$Month <- "November"
```

```
#completely remove NA from likelihood to recommend without replacement
```

```
febData <- filter(febData, Likelihood_Recommend_H != "NA")  
mayData <- filter(mayData, Likelihood_Recommend_H != "NA")  
augData <- filter(augData, Likelihood_Recommend_H != "NA")  
novData <- filter(novData, Likelihood_Recommend_H != "NA")
```

```
#Union the dataframes (without removing duplicates, as a union would in SQL)  
totalData <- rbind(febData, mayData, augData, novData)
```

```
#Choose the location  
myMapData <- sqldf("SELECT State_PL, COUNT(*) FROM totalData GROUP BY State_PL ORDER BY  
COUNT(*) DESC")  
str(myMapData)  
myMapData <- myMapData[myMapData$State_PL %in% state.name,]  
emptyStates <- setdiff(state.name, myMapData$State_PL) #from dplyr?  
emptyStates <- data.frame(emptyStates, 0)  
colnames(emptyStates) <- c("State_PL", "COUNT(*)")  
myMapData <- rbind(myMapData, emptyStates)  
rownames(myMapData) <- NULL  
myMapData$State_PL <- tolower(myMapData$State_PL)  
colnames(myMapData) <- c("State_PL", "COUNT")  
USA <- map_data("state")
```

```
responseMap <- ggplot(data = myMapData, aes(map_id=State_PL))+  
  geom_map(map=USA, aes(fill=COUNT))+  
  expand_limits(x=USA$long, y=USA$lat)+  
  coord_fixed()+  
  ggtitle("Survey Responses by State")  
responseMap
```

```
california <- filter(USA, region=="california")  
california <- california[,-6]  
caliData <- filter(totalData, State_PL=="California")
```

```
caliData <- filter(caliData, Condition_Hotel_H != "NA")  
caliData <- filter(caliData, Staff_Cared_H != "NA")  
caliData <- filter(caliData, Internet_Sat_H != "NA")  
caliData <- filter(caliData, Check_In_H != "NA")  
caliData <- filter(caliData, Tranquility_H != "NA")  
caliData <- filter(caliData, Customer_SVC_H != "NA")
```

```

View(caliData)

unique(caliData$Brand_PL)

ggplot(data=caliData, aes(x=Brand_PL))+
  geom_bar()

#Will choose to examine Hyatt Regency b/c has highest response and lowest score (not counting
ambiguous 'Hyatt')
sqldf("SELECT Brand_PL, AVG(Likelihood_Recommend_H) FROM caliData GROUP BY Brand_PL ORDER
BY AVG(Likelihood_Recommend_H) DESC")

brandGG <- ggplot(data=caliData, aes(x=NPS_Type)) +
  geom_bar()+
  facet_grid(~Brand_PL)+
  ggtitle("NPS_Type by Brand")
brandGG

tapply(caliData$Likelihood_Recommend_H, caliData$Brand_PL, skewness)

row.names(caliData)<-1:nrow(caliData)

#filter by brand
caliData <- filter(caliData, Brand_PL == "Hyatt Regency")

#reorder months by factor levels
caliData$Month <- factor(caliData$Month, levels = month.name)

#replaces empty Limo values with "N"
caliData$Limo.Service_PL <- as.factor(gsub("^$", "N", caliData$Limo.Service_PL))

caliPoints <- data.frame(tapply(caliData$Likelihood_Recommend_H, caliData$City_PL, mean))
colnames(caliPoints) <- "Avg_Likelihood_Recommend_H"
caliPoints$City_PL <- rownames(caliPoints)
rownames(caliPoints) <- NULL
caliPoints$State_PL <- "california"
caliPoints <- na.omit(caliPoints)
caliPointsLatLong <- geocode(paste(caliPoints$City_PL, caliPoints$State_PL))
caliPoints$lon <- caliPointsLatLong$lon
caliPoints$lat <- caliPointsLatLong$lat

caliPoints

caliPoints$City_PL <- tolower(caliPoints$City_PL)

californiaMap <- ggplot(data = myMapData, aes(map_id=State_PL))+

```

```

geom_map(map=california, aes(fill=COUNT))+
expand_limits(x=california$long, y=california$lat)+
coord_fixed()+
geom_point(data = caliPoints, aes(x=caliPoints$lon, y=caliPoints$lat,
color=Avg_Likelihood_Recommend_H), size=2)+
scale_color_gradient(low="black", high="lightgrey")
californiaMap

```

#Analyze Data

```

#mean score likelihood_recommend score
tapply(caliData$Likelihood_Recommend_H, caliData$POV_CODE_C, mean) #leisure has slightly higher
LTR

```

```

ggplot(data=caliData, aes(x=POV_CODE_C))+
geom_bar()+
ggtitle("Business vs Leisure Customers, California")

```

```

unique(caliData$GP_Tier)
caliData[caliData$GP_Tier=="GOLD", which(colnames(caliData)=="GP_Tier")] <- "Gold"
caliData[caliData$GP_Tier=="PLAT", which(colnames(caliData)=="GP_Tier")] <- "Platinum"
caliData[caliData$GP_Tier=="DIAM", which(colnames(caliData)=="GP_Tier")] <- "Diamond"

```

```

unique(caliData$GP_Tier)

```

```

View(caliData)

```

```

#####

```

```

# aRules analysis
caliData[caliData$Casino_PL != "N", which(colnames(caliData)=="Casino_PL")] <- "Y"
caliData <- filter(caliData, Mini.Bar_PL != "")

```

```

business <- filter(caliData, POV_CODE_C == "BUSINESS")
leisure <- filter(caliData, POV_CODE_C == "LEISURE")

```

```

#in Apriori, only patterns up to a length of 10 are returned
#Also... No instance of casinos at Hyatt Regency Hotels in California...
#No instance of conference rooms at Hyatt Regency hotels in California...
businessRules <- business[,c("Room_Type_H", "Convention_PL", "Business.Center_PL",
" Golf_PL", "Limo.Service_PL", "Mini.Bar_PL",
"NPS_Type")]

```

```

leisureRules <- leisure[, c("Room_Type_H", "Pool.Indoor_PL", "Casino_PL",

```

```

"Golf_PL", "Limo.Service_PL", "Mini.Bar_PL",
"Pool.Outdoor_PL", "Resort_PL", "Spa_PL", "NPS_Type"]

businessRulesApri <- apriori(businessRules,
                             parameter = list(minlen=5, supp=0.005, conf=0.05), #minlen refers to minimum number of
                             variables being analyzed
                             appearance = list(rhs=c("NPS_Type=Passive", "NPS_Type=Detractor"), #restricting RHS to
                             Passive, or Detractor NPS types
                             default="lhs"), control = list(verbose=F))
businessRulesApri <- sort(businessRulesApri, by = "lift")
inspect(businessRulesApri[1:50])

#Proportion of services to NPS Type
businessProp <- businessRules
businessProp$one <- 1

businessProp <- ddply(businessProp, "NPS_Type", transform, percent = one / sum(one) * 100)

#Golf seems to be important
ggplot(businessProp, aes(x=NPS_Type, y=percent, fill=Golf_PL))+
  geom_bar(stat="identity")

ggplot(businessProp, aes(x=NPS_Type, y=percent, fill=Conference_PL))+
  geom_bar(stat="identity")

ggplot(businessProp, aes(x=NPS_Type, y=percent, fill=Limo.Service_PL))+
  geom_bar(stat="identity")

ggplot(businessProp, aes(x=NPS_Type, y=percent, fill=Casino_PL))+
  geom_bar(stat="identity")

#Mini Bar seems important
ggplot(businessProp, aes(x=NPS_Type, y=percent, fill=Mini.Bar_PL))+
  geom_bar(stat="identity")

#Business Center is important
ggplot(businessProp, aes(x=NPS_Type, y=percent, fill=Business.Center_PL))+
  geom_bar(stat="identity")

ggplot(businessProp, aes(x=NPS_Type, y=percent, fill=Convention_PL))+
  geom_bar(stat="identity")

ggplot(businessProp, aes(x=NPS_Type, y=percent, fill=Fitness.Center_PL))+
  geom_bar(stat="identity")

```

```
#not enough leisure surveys to make an informed decision using Apriori
leisureRulesApri <- apriori(leisureRules,
                           parameter = list(minlen=3, supp=0.005, conf=0.2), #minlen refers to minimum number
                           of variables being analyzed
                           appearance = list(rhs=c("NPS_Type=Passive", "NPS_Type=Detractor"), #restricting RHS to
                           Passive or Detractor NPS types
                           default="lhs"), control = list(verbose=F))
leisureRulesApri <- sort(leisureRulesApri, by = "lift")
inspect(leisureRulesApri[1:20])
```

```
#####
```

```
#Linear Modeling
```

```
#lm (single linear regression)
#lm_condition and lm_customerSVC have highest coefficients of determination
(lm_condition <- lm(data = business, formula = Likelihood_Recommend_H ~ Condition_Hotel_H))
summary(lm_condition)$r.squared
summary(lm_condition)$adj.r.squared
```

```
(lm_staff <- lm(data = business, formula = Likelihood_Recommend_H ~ Staff_Cared_H))
summary(lm_staff)$r.squared
summary(lm_staff)$adj.r.squared
```

```
(lm_internet <- lm(data = business, formula = Likelihood_Recommend_H ~ Internet_Sat_H))
summary(lm_internet)$r.squared
summary(lm_internet)$adj.r.squared
```

```
(lm_checkIn <- lm(data = business, formula = Likelihood_Recommend_H ~ Check_In_H))
summary(lm_checkIn)$r.squared
summary(lm_checkIn)$adj.r.squared
```

```
(lm_tranquility <- lm(data = business, formula = Likelihood_Recommend_H ~ Tranquility_H))
summary(lm_tranquility)$r.squared
summary(lm_tranquility)$adj.r.squared
```

```
(lm_customerSVC <- lm(data = business, formula = Likelihood_Recommend_H ~ Customer_SVC_H))
summary(lm_customerSVC)$r.squared
summary(lm_customerSVC)$adj.r.squared
```

```
#multiple linear regression
```

```
lm_multi1 <- lm(data = business, formula = Likelihood_Recommend_H ~ Condition_Hotel_H +
  Customer_SVC_H)
summary(lm_multi1)$r.squared
summary(lm_multi1)$adj.r.squared
```

```
lm_multi2 <- lm(data = business, formula = Likelihood_Recommend_H ~ Condition_Hotel_H +
  Customer_SVC_H + Tranquility_H)
```



```
summary(lm_multi2)$r.squared
summary(lm_multi2)$adj.r.squared
```

```
lm_multi3 <- lm(data = business, formula = Likelihood_Recommend_H ~ Condition_Hotel_H +
  Customer_SVC_H + Tranquility_H + Staff_Cared_H)
summary(lm_multi3)$r.squared
summary(lm_multi3)$adj.r.squared
```

```
lm_multi4 <- lm(data = business, formula = Likelihood_Recommend_H ~ Condition_Hotel_H +
  Customer_SVC_H + Tranquility_H + Staff_Cared_H + Check_In_H)
summary(lm_multi4)$r.squared
summary(lm_multi4)$adj.r.squared
```

```
lm_multi5 <- lm(data = business, formula = Likelihood_Recommend_H ~ Condition_Hotel_H +
  Customer_SVC_H + Tranquility_H + Staff_Cared_H + Check_In_H + Internet_Sat_H)
summary(lm_multi5)$r.squared
summary(lm_multi5)$adj.r.squared
summary(lm_multi5)
```

```
#strong P-values for all variables ***EXCEPT Check_In_H***
#Linear model explains >66% of variability in likelihood to recommend
summary(lm_multi5)
```

```
#####
```

```
#KSVM
business$NPS_Num <- business$NPS_Type
any(business$Convention_PL=="Y")
any(business$Pool=="Y")
```

```
business$NPS_Num <- factor(business$NPS_Num,
  labels = c(1, 2, 3))
```

```
business$Pool.Outdoor_PL_Num <- factor(business$Pool.Outdoor_PL,
  labels = c(0,1))
business[business$Pool.Outdoor_PL=="Y", which(colnames(business)=="Pool.Outdoor_PL_Num")] <- 1
business[business$Pool.Outdoor_PL=="N", which(colnames(business)=="Pool.Outdoor_PL_Num")] <- 0
```

```
business$Pool.Indoor_PL_Num <- factor(business$Pool.Indoor_PL,
  labels = c(0,1))
business[business$Pool.Indoor_PL=="Y", which(colnames(business)=="Pool.Indoor_PL_Num")] <- 1
business[business$Pool.Indoor_PL=="N", which(colnames(business)=="Pool.Indoor_PL_Num")] <- 0
```

```

business$Convention_PL_Num <- factor(business$Convention_PL,
                                     labels = c(0,1))
business[business$Convention_PL=="Y", which(colnames(business)=="Convention_PL_Num")] <- 1
business[business$Convention_PL=="N", which(colnames(business)=="Convention_PL_Num")] <- 0


business$Mini.Bar_PL_Num <- factor(business$Mini.Bar_PL,
                                   labels = c(0,1))
business[business$Mini.Bar_PL=="Y", which(colnames(business)=="Mini.Bar_PL_Num")] <- 1
business[business$Mini.Bar_PL=="N", which(colnames(business)=="Mini.Bar_PL_Num")] <- 0


business$Business.Center_PL_Num <- factor(business$Business.Center_PL,
                                           labels = c(0,1))
business[business$Business.Center_PL=="Y", which(colnames(business)=="Business.Center_PL_Num")]
  <- 1
business[business$Business.Center_PL=="N", which(colnames(business)=="Business.Center_PL_Num")]
  <- 0


business$Spa_PL_Num <- factor(business$Spa_PL,
                             labels = c(0,1))
business[business$Spa_PL=="Y", which(colnames(business)=="Spa_PL_Num")] <- 1
business[business$Spa_PL=="N", which(colnames(business)=="Spa_PL_Num")] <- 0


business[business$NPS_Num=="Promoter", which(colnames(business)=="NPS_Num")] <- 3
business[business$NPS_Num=="Passive", which(colnames(business)=="NPS_Num")] <- 2
business[business$NPS_Num=="Detractor", which(colnames(business)=="NPS_Num")] <- 1


#Set "low-medium-high" values for hotel operations as 0,1,2 respectively

#Condition_Hotel_H_Num
business$Condition_Hotel_H_Num <- as.numeric(business$Condition_Hotel_H)
business[business$Condition_Hotel_H_Num==1,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 0
business[business$Condition_Hotel_H_Num==2,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 0
business[business$Condition_Hotel_H_Num==3,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 0
business[business$Condition_Hotel_H_Num==4,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 0

business[business$Condition_Hotel_H_Num==5,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 1
business[business$Condition_Hotel_H_Num==6,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 1

```

```

business[business$Condition_Hotel_H_Num==7,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 1

business[business$Condition_Hotel_H_Num==8,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 2
business[business$Condition_Hotel_H_Num==9,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 2
business[business$Condition_Hotel_H_Num==10,
  which(colnames(business)=="Condition_Hotel_H_Num")] <- 2

#Customer_SVC_H
business$Customer_SVC_H_Num <- as.numeric(business$Customer_SVC_H)
business[business$Customer_SVC_H_Num==1, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 0
business[business$Customer_SVC_H_Num==2, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 0
business[business$Customer_SVC_H_Num==3, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 0
business[business$Customer_SVC_H_Num==4, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 0

business[business$Customer_SVC_H_Num==5, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 1
business[business$Customer_SVC_H_Num==6, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 1
business[business$Customer_SVC_H_Num==7, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 1

business[business$Customer_SVC_H_Num==8, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 2
business[business$Customer_SVC_H_Num==9, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 2
business[business$Customer_SVC_H_Num==10, which(colnames(business)=="Customer_SVC_H_Num")]
  <- 2

#Tranquility_H
business$Tranquility_H_Num <- as.numeric(business$Tranquility_H)
business[business$Tranquility_H_Num==1, which(colnames(business)=="Tranquility_H_Num")] <- 0
business[business$Tranquility_H_Num==2, which(colnames(business)=="Tranquility_H_Num")] <- 0
business[business$Tranquility_H_Num==3, which(colnames(business)=="Tranquility_H_Num")] <- 0
business[business$Tranquility_H_Num==4, which(colnames(business)=="Tranquility_H_Num")] <- 0

business[business$Tranquility_H_Num==5, which(colnames(business)=="Tranquility_H_Num")] <- 1
business[business$Tranquility_H_Num==6, which(colnames(business)=="Tranquility_H_Num")] <- 1
business[business$Tranquility_H_Num==7, which(colnames(business)=="Tranquility_H_Num")] <- 1

business[business$Tranquility_H_Num==8, which(colnames(business)=="Tranquility_H_Num")] <- 2
business[business$Tranquility_H_Num==9, which(colnames(business)=="Tranquility_H_Num")] <- 2

```

```

business[business$Tranquility_H_Num==10, which(colnames(business)=="Tranquility_H_Num")] <- 2

#Staff_Cared_H
business$Staff_Cared_H_Num <- as.numeric(business$Staff_Cared_H)
business[business$Staff_Cared_H_Num==1, which(colnames(business)=="Staff_Cared_H_Num")] <- 0
business[business$Staff_Cared_H_Num==2, which(colnames(business)=="Staff_Cared_H_Num")] <- 0
business[business$Staff_Cared_H_Num==3, which(colnames(business)=="Staff_Cared_H_Num")] <- 0
business[business$Staff_Cared_H_Num==4, which(colnames(business)=="Staff_Cared_H_Num")] <- 0

business[business$Staff_Cared_H_Num==5, which(colnames(business)=="Staff_Cared_H_Num")] <- 1
business[business$Staff_Cared_H_Num==6, which(colnames(business)=="Staff_Cared_H_Num")] <- 1
business[business$Staff_Cared_H_Num==7, which(colnames(business)=="Staff_Cared_H_Num")] <- 1

business[business$Staff_Cared_H_Num==8, which(colnames(business)=="Staff_Cared_H_Num")] <- 2
business[business$Staff_Cared_H_Num==9, which(colnames(business)=="Staff_Cared_H_Num")] <- 2
business[business$Staff_Cared_H_Num==10, which(colnames(business)=="Staff_Cared_H_Num")] <- 2

#Check_In_H
business$Check_In_H_Num <- as.numeric(business$Check_In_H)
business[business$Check_In_H_Num==1, which(colnames(business)=="Check_In_H_Num")] <- 0
business[business$Check_In_H_Num==2, which(colnames(business)=="Check_In_H_Num")] <- 0
business[business$Check_In_H_Num==3, which(colnames(business)=="Check_In_H_Num")] <- 0
business[business$Check_In_H_Num==4, which(colnames(business)=="Check_In_H_Num")] <- 0

business[business$Check_In_H_Num==5, which(colnames(business)=="Check_In_H_Num")] <- 1
business[business$Check_In_H_Num==6, which(colnames(business)=="Check_In_H_Num")] <- 1
business[business$Check_In_H_Num==7, which(colnames(business)=="Check_In_H_Num")] <- 1

business[business$Check_In_H_Num==8, which(colnames(business)=="Check_In_H_Num")] <- 2
business[business$Check_In_H_Num==9, which(colnames(business)=="Check_In_H_Num")] <- 2
business[business$Check_In_H_Num==10, which(colnames(business)=="Check_In_H_Num")] <- 2

#Internet_Sat_H
business$Internet_Sat_H_Num <- as.numeric(business$Internet_Sat_H)
business[business$Internet_Sat_H_Num==1, which(colnames(business)=="Internet_Sat_H_Num")] <- 0
business[business$Internet_Sat_H_Num==2, which(colnames(business)=="Internet_Sat_H_Num")] <- 0
business[business$Internet_Sat_H_Num==3, which(colnames(business)=="Internet_Sat_H_Num")] <- 0
business[business$Internet_Sat_H_Num==4, which(colnames(business)=="Internet_Sat_H_Num")] <- 0

business[business$Internet_Sat_H_Num==5, which(colnames(business)=="Internet_Sat_H_Num")] <- 1
business[business$Internet_Sat_H_Num==6, which(colnames(business)=="Internet_Sat_H_Num")] <- 1
business[business$Internet_Sat_H_Num==7, which(colnames(business)=="Internet_Sat_H_Num")] <- 1

business[business$Internet_Sat_H_Num==8, which(colnames(business)=="Internet_Sat_H_Num")] <- 2
business[business$Internet_Sat_H_Num==9, which(colnames(business)=="Internet_Sat_H_Num")] <- 2
business[business$Internet_Sat_H_Num==10, which(colnames(business)=="Internet_Sat_H_Num")] <- 2

View(business)

```

```
#####
#Set training and test data
randIndex <- sample(1:dim(business)[1]) #randomize index of rows to prevent any unintended bias
head(randIndex)

train_cutpoint2_3 <- floor(2*nrow(business)/3) #set the number of rows that will be used in the training
set (2/3 of data will be for training)
trainData <- business[randIndex[1:train_cutpoint2_3],] #applies 2/3 cut-off point for caliData
testData <- business[randIndex[train_cutpoint2_3+1:nrow(business)],] #applies the remainder for
testData

#####

ksvmOutputNum <- ksvm(NPS_Num ~ Condition_Hotel_H_Num + Customer_SVC_H_Num +
  Tranquility_H_Num + Staff_Cared_H_Num + Check_In_H_Num + Internet_Sat_H_Num,
  data=trainData,
  kernel = "rbfdot",
  kpar="automatic",
  C = 5,
  cross = 3,
  prob.model = TRUE)
ksvmOutputNum

ksvmOutputLR <- ksvm(Likelihood_Recommend_H ~ Condition_Hotel_H + Customer_SVC_H +
  Tranquility_H + Staff_Cared_H + Check_In_H + Internet_Sat_H,
  data=trainData,
  kernel = "rbfdot",
  kpar="automatic",
  C = 5,
  cross = 3,
  prob.model = TRUE)
ksvmOutputLR

#histogram of ksvm
ksvmpredLR <- predict(ksvmOutputLR,testData)
ksvmpredLR
table(ksvmpredLR)
actual <- testData$Likelihood_Recommend_H
length(actual)
length(ksvmpredLR)
Kvar2 <- actual - as.vector(ksvmpredLR)
Kvar2 <- na.omit(Kvar2)
View(Kvar2)
hist(Kvar2, main = "Histogram of KSVM Model")
```

```
ksvmOutputNPS <- ksvm(NPS_Num ~ Condition_Hotel_H + Customer_SVC_H + Tranquility_H +
  Staff_Cared_H + Check_In_H + Internet_Sat_H,
  data=trainData,
  kernel = "rbfdot",
  kpar="automatic",
  C = 5,
  cross = 3,
  prob.model = TRUE)
ksvmOutputNPS
```

```
ksvmOutputCat <- ksvm(NPS_Num ~ Pool.Indoor_PL_Num + Pool.Outdoor_PL_Num +
  Convention_PL_Num + Mini.Bar_PL_Num + Business.Center_PL_Num + Spa_PL_Num,
  data=trainData,
  kernel = "rbfdot",
  kpar="automatic",
  C = 5,
  cross = 3,
  prob.model = TRUE)
ksvmOutputCat
```

```
ksvmOutputCombined <- ksvm(NPS_Num ~ Condition_Hotel_H + Customer_SVC_H + Tranquility_H +
  Staff_Cared_H + Check_In_H + Internet_Sat_H +
  Convention_PL_Num + Business.Center_PL_Num + Spa_PL_Num + Mini.Bar_PL_Num,
  data=trainData,
  kernel = "rbfdot",
  kpar="automatic",
  C = 5,
  cross = 3,
  prob.model = TRUE)
ksvmOutputCombined
```

```
ksvmOutputCombined2 <- ksvm(NPS_Num ~ Condition_Hotel_H_Num + Customer_SVC_H_Num +
  Tranquility_H_Num + Staff_Cared_H_Num + Check_In_H_Num + Internet_Sat_H_Num +
  Convention_PL_Num + Business.Center_PL_Num + Spa_PL_Num + Mini.Bar_PL_Num,
  data=trainData,
  kernel = "rbfdot",
  kpar="automatic",
  C = 5,
  cross = 3,
  prob.model = TRUE)
ksvmOutputCombined2
```

```

#####
#SVM
#####
SVMOutput <- svm(Likelihood_Recommend_H ~ Condition_Hotel_H + Customer_SVC_H + Tranquility_H
+
      Staff_Cared_H + Check_In_H + Internet_Sat_H, data=trainData)
SVMOutput

svm(NPS_Num ~ Pool.Indoor_PL_Num + Pool.Outdoor_PL_Num + Convention_PL_Num +
Mini.Bar_PL_Num + Business.Center_PL_Num + Spa_PL_Num,
  data=trainData)

#trainData$Customer_SVC_H[is.na(trainData$Customer_SVC_H)] <- mean(trainData$Customer_SVC_H,
  na.rm = TRUE)
#any(is.na(testData$Internet_Sat_H))
#plot(scale(c),pch=16)

svmpred <- predict(SVMOutput,testData)
svmpred
table(svmpred)
actual <- testData$Likelihood_Recommend_H
actual
length(actual)
length(svmpred)
var2 <- actual - svmpred
View(var2)
var2 <- na.omit(var2)
hist(var2, main = "Histogram of SVM Model")

#####
#Naive Bayes
#####
#Naive Bayes
#Predicting Accuracy
row.names(business)<-1:nrow(business)
business[, "train"] <- ifelse(runif(nrow(business))<0.80,1,0) #Sampling 80% of the data
trainColNum <- grep("train",names(business))
trainData1 <- business[business$train==1,-trainColNum]
testData1 <- business[business$train==0,-trainColNum]

nb_model1 <- naiveBayes(NPS_Num ~ Condition_Hotel_H_Num + Customer_SVC_H_Num +
Tranquility_H_Num + Staff_Cared_H_Num + Check_In_H_Num + Internet_Sat_H_Num +

```

```

Convention_PL_Num + Business.Center_PL_Num + Spa_PL_Num + Mini.Bar_PL_Num,
data=trainData)
nb_model1

View(trainData1)

pred<- predict(nb_model1,testData1,type="class")
pred

table(pred=pred,true=testData1$NPS_Num)

mean(pred==testData1$NPS_Num)

MultipleRuns1 <- function(train_fraction,n){
  fraction_correct <- rep(NA,n)
  for (i in 1:n){
    business[, "train"] <- ifelse(runif(nrow(business))<0.80,1,0)
    trainColNum <- grep("train",names(business))
    trainData1 <- business[business$train==1,-trainColNum]
    testData1 <- business[business$train==0,-trainColNum]

    nb_model1 <- naiveBayes(NPS_Num ~ Condition_Hotel_H_Num + Customer_SVC_H_Num +
Tranquility_H_Num + Staff_Cared_H_Num + Check_In_H_Num + Internet_Sat_H_Num +
Convention_PL_Num + Business.Center_PL_Num + Spa_PL_Num + Mini.Bar_PL_Num,
data=trainData)

    pred<- predict(nb_model1,testData1,type="class")

    fraction_correct[i] <- mean(pred==testData1$NPS_Num)
  }
  return(fraction_correct)
}

fraction_correct_predictions <- MultipleRuns1(0.8,20)
fraction_correct_predictions

#Accuracy over 20 occurrences
# 0.7846154 0.7876950 0.7770979 0.7746858 0.7670886 0.7640351 0.7650709 0.7763636 0.7692308
0.7998296 0.7711864 0.7839286 0.7815050 0.7678869
# 0.7817214 0.7610390 0.7713115 0.7832168 0.7730116 0.7867384

summary(fraction_correct_predictions)

sd(fraction_correct_predictions)

#Naive Bayes

```



```

business[, "train"] <- ifelse(runif(nrow(business))<0.80,1,0)
trainColNum <- grep("train",names(business))
trainData1 <- business[business$train==1,-trainColNum]
testData1 <- business[business$train==0,-trainColNum]

nb_modelAll <- naiveBayes(NPS_Num ~ Condition_Hotel_H_Num + Customer_SVC_H_Num +
  Tranquility_H_Num + Staff_Cared_H_Num + Check_In_H_Num + Internet_Sat_H_Num +
    Convention_PL_Num + Business.Center_PL_Num + Spa_PL_Num + Mini.Bar_PL_Num,
  data=trainData)

predAll <- predict(nb_modelAll,testData1,type="class")
NBAll = data.frame(v1=testData1$NPS_Num, v2=predAll)

Errornb<- sum(testData1[, "NPS_Num"] != predAll) * 100 /length(predAll)
Errornb

#####

#####
#I experimented with these models, but are probably not too useful.

#Linear Model Based on Limo Service
Limo_conditionPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H,
  x=Condition_Hotel_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2) +
  geom_smooth(method="lm")+
  facet_grid(.~Limo.Service_PL)+
  ggtitle("Hotel Condition Rating")

Limo_staffPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Staff_Cared_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(.~Limo.Service_PL)+
  ggtitle("Staff Rating")

Limo_internetPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Internet_Sat_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(.~Limo.Service_PL)+
  ggtitle("Internet Rating")

Limo_checkInPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Check_In_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+

```

```
geom_smooth(method="lm")+
facet_grid(~Limo.Service_PL)+
ggtitle("Check-In Satisfaction Rating")
```

```
Limo_tranquilityPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Tranquility_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(~Limo.Service_PL)+
  ggtitle("Hotel Tranquility Rating")
```

```
Limo_customerPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, Customer_SVC_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(~Limo.Service_PL)+
  ggtitle("Customer Service Rating")
```

```
#look at Staff, check-in, cust.Service, tranquility, Internet.... All have lower scores w/out limo service
grid.arrange(Limo_conditionPlotRM, Limo_staffPlotRM, Limo_internetPlotRM, Limo_checkInPlotRM,
  Limo_tranquilityPlotRM, Limo_customerPlotRM, ncol=2, nrow=3)
```

```
#gender
business[business$Gender_H=="", which(colnames(caliData)=="Gender_H")] <- "Prefer not to answer"
```

```
GenderconditionPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H,
  x=Condition_Hotel_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2) +
  geom_smooth(method="lm")+
  facet_grid(~Gender_H)+
  ggtitle("Hotel Condition Rating")
```

```
GenderstaffPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Staff_Cared_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(~Gender_H)+
  ggtitle("Staff Rating")
```

```
GenderinternetPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Internet_Sat_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(~Gender_H)+
  ggtitle("Internet Rating")
```

```
GendercheckInPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Check_In_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(~Gender_H)+
  ggtitle("Check-In Rating")
```

```

facet_grid(.~Gender_H)+
ggtitle("Check-In Satisfaction Rating")

GendertranquilityPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Tranquility_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(.~Gender_H)+
  ggtitle("Hotel Tranquility Rating")

GendercustomerPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, Customer_SVC_H))
+
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(.~Gender_H)+
  ggtitle("Customer Service Rating")

grid.arrange(GenderconditionPlotRM, GenderstaffPlotRM, GenderinternetPlotRM,
  GendercheckInPlotRM, GendertranquilityPlotRM, GendercustomerPlotRM, ncol=2, nrow=3)

#Tier
Tier_conditionPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H,
  x=Condition_Hotel_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2) +
  geom_smooth(method="lm")+
  facet_grid(.~GP_Tier)+
  ggtitle("Hotel Condition Rating")

Tier_staffPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Staff_Cared_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(.~GP_Tier)+
  ggtitle("Staff Rating")

Tier_internetPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Internet_Sat_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(.~GP_Tier)+
  ggtitle("Internet Rating")

Tier_checkInPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Check_In_H)) +
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+
  geom_smooth(method="lm")+
  facet_grid(.~GP_Tier)+
  ggtitle("Check-In Satisfaction Rating")

```

```
Tier_tranquilityPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, x=Tranquility_H)) +  
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+  
  geom_smooth(method="lm")+  
  facet_grid(.~GP_Tier)+  
  ggtitle("Hotel Tranquility Rating")
```

```
Tier_customerPlotRM <- ggplot(data=business, aes(y=Likelihood_Recommend_H, Customer_SVC_H)) +  
  geom_point(position=position_jitter(width = .5, height = .5), alpha=0.2)+  
  geom_smooth(method="lm")+  
  facet_grid(.~GP_Tier)+  
  ggtitle("Customer Service Rating")
```

```
grid.arrange(Tier_conditionPlotRM, Tier_staffPlotRM, Tier_internetPlotRM, Tier_checkInPlotRM,  
  Tier_tranquilityPlotRM, Tier_customerPlotRM, ncol=2, nrow=3)
```