

*Final Project*

*Hotel Analysis*

*Net Promoter Score*

*In partial fulfillment of*

IST 687 (Summer 2017): Applied Data Science

Professor: Gary Krudys

The iSchool, Syracuse University

*Presented By*

Kellie Mosely, Zhaowei Jiang, Mason David, Jacob Dineen and Mohamed Khalifa

September 17, 2017

*[Any company information disclosed here is confidential and not for public consumption]*

***Table of Contents***

Introduction: …………………………………………………………………. 2

Target Region: ………………………………………………………………. 3

Audience Demographic: …………………………………………………….. 4

The Survey ……………………………………………………………………. 6

Derived Columns ……………………………………………………………... 7

NPS Score ………………………………………………………………….... 7

Correlation …………………………………………………………….………. 9

Linear Regression ……………………………………………………….….. 10

Support Vector Machines: ………………………………………..………... 11

Association Analysis …………………………………………….…..……… 13

Recommendation ………………………………………….…….………..….13

Appendix ………………………………………………………….……….…. 16

Work Cited ………………………………………………..…………….……. 28

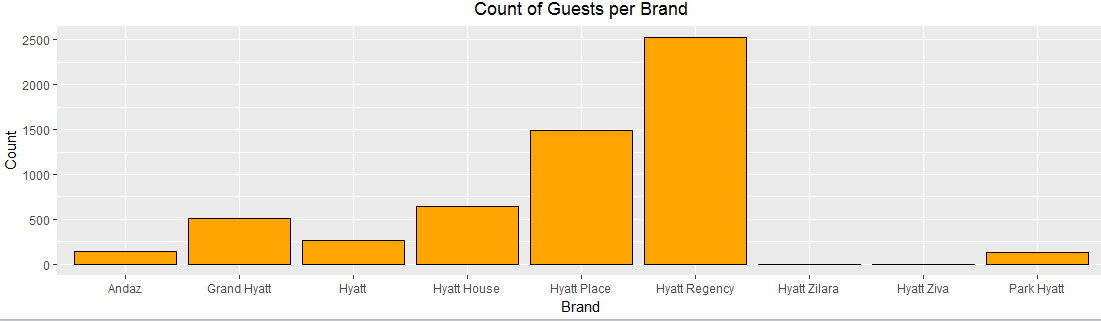
Data Munging Code ……………………………………….……….……….. 29

Package Glossary ……………………………………………….……..…… 32

***Introduction***

As consultants, we have been propositioned with the task of better understanding, analyzing and predicting net promoter score as it relates to our client’s business. “Net Promoter Score®, or NPS®, measures customer experience and predicts business growth. This proven metric transformed the business world and now provides the core measurement for customer experience management programs the world round” (What is Net Promoter?). By assigning a value to each individual consumer's experience with our brands, we can put ourselves in a better position to meet not only their goals, but our goals, as a company, as well. Our client is Hyatt, and the following chains of the Hyatt brand are the acting filter for our analysis: Hyatt Regency, Hyatt Place, Hyatt House, Hyatt and Andaz. Almost 80% of the dataset that we analyze will be inclusive of customers who stayed at either a Hyatt Regency or Hyatt Place location, so while our analysis can carry over to other branches within these regions, it would be most relevant as it pertains to renderings of these specific brands.

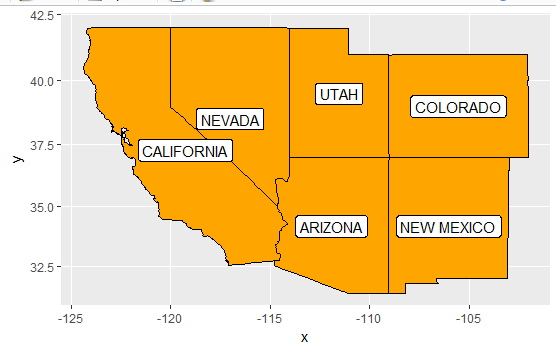
***Appendix A: Column Chart by Brand***



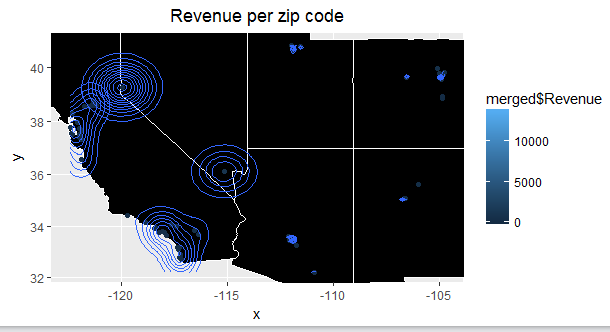
**Target Region**

Our focus, as consultants, is to look at the Southwest/Western United States. Because the survey response rate is too low, we will want an expanded geographical sample to better represent the population. Our goal is to run a statistical process many times, and we will take the law of large numbers into account. According to the law, we expect to see the distribution of sampling means starts to create a bell-shaped or normal distribution, and the center of that distribution, the mean of all those sample means get closer to the actual population mean. “The Law of Large Numbers says that in repeated, [independent](http://www.stat.berkeley.edu/%7Estark/SticiGui/Text/gloss.htm#independent) trials with the same probability p of success in each trial, the percentage of successes is increasingly likely to be close to the chance of success as the number of trials increases” (Glossary of Statistical Terms).

***Appendix B: Focused States***



***Appendix C: Revenue by Zip Code***



**The above graph shows that a majority of the total revenue driven from these ‘Southwestern’ states is from California, which coincides with the fact that most of the traffic to these states is represented by traffic to California hotels.**

**Audience Demographic**

Summary statistics on our data to this point, which contains 5713 total survey responses aggregated across the Southwestern United States, will provide us with notable trends, be they behavioral or demographic, about our current audience. This information won’t necessarily help with increasing NPS Score, but it will aid in potential marketing efforts if we are more targeted in gearing campaigns towards specific clusters of the population.

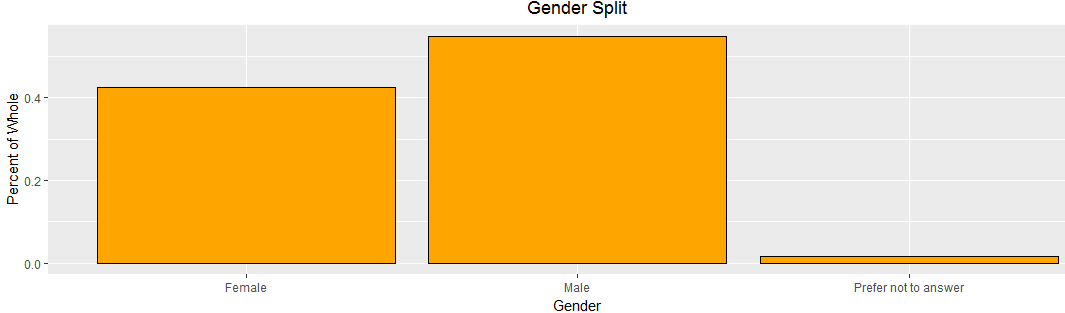
***Appendix D: Descriptive Statistics***

**Reason for visit :** Of the 5713 observations, we have a pretty even split between people staying at our hotels for reasons pertaining to either business and leisure.

**Guest Country:** 92% of our survey respondents were visiting from somewhere in the United States. This is either telling us that there is an international market to be tapped into, or we should simply focus on regionality.

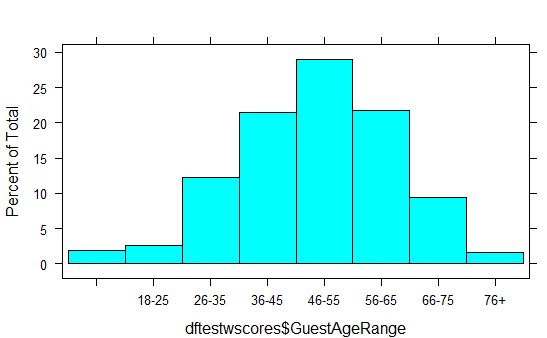
**Guest Gender:** Our visitors, as it relates to our sampled data, skews slightly male - as they represent 55% of our total survey respondents.

***Appendix E: Column Chart by Gender Percentage***

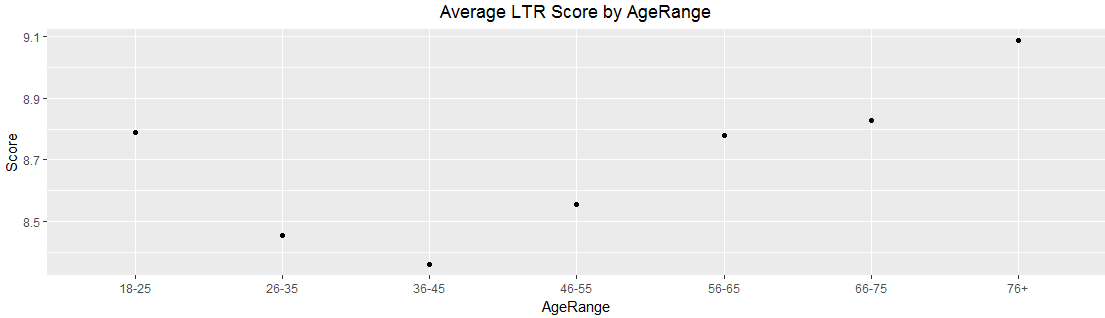


**Guest Age Range:** Of our survey respondents, this is likely one of our most significant pieces of data as it relates to audience demographics. We skew supremely heavy towards an older audience, which could speak to a number of things. 1. There could be some auto-correlation between age and willingness to respond to a survey. 2. There could be external economic class differentiators, as it relates to age, ie. income/financial stability.

***Appendix F: Histogram by Guest Age Range***

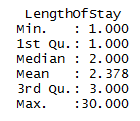
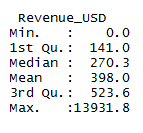


***Appendix G: Line Chart of Likelihood to Reco by Age Range***



**Guest Language:** 5600 of our 5713 observations of survey respondents registered English as their language. This could potentially be a sampling error where we might not have sent out bilingual or spanish surveys, and as such, our response rate was low amongst that demographic.

Our mean revenue generated by each of our observations was $398.00 per their respective stays, while our mean visit length was 2 days - Our quantile statistics on the customer's’ length of stay shows us that the 25%-75% is 1 to 3 days.

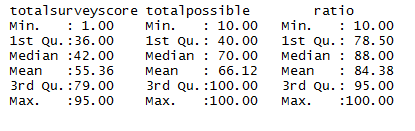


The above information provides guidance to know who our audience are in terms of raw counts, or percentages of the whole, but when we’re looking to find who our audience truly are, we’re really looking to see which particular group of people are driving the most revenue.

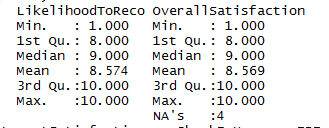
Males spend, on the average, about 15% more than females. The 18-25 Age demo spends the least, on average, and the 26-65 range is our sweet spot in terms of average revenue generated.

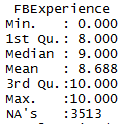
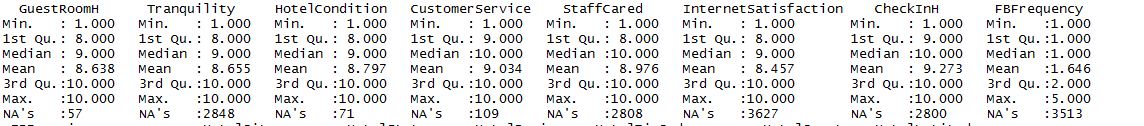
**The Survey**

One of these reasons we wanted to create a ratio column on the survey score against the total possible score is due to all of the null values present within those particular columns. This information may help those in charge better understand the audience that responds to surveys. The way to read the table below is that, on average, only 66% of the survey questions were responded to.



What we can see from the below statistics summarizing the survey responses, and their relevance to net promoter score, is that there were significant amounts of null responses on many of the questions, which could even ultimately save us a step on weighting our coefficients in a linear model, but we’ll get to that.





**Derived Columns**

**CheckInMonth\_Year:** This column is derived from various date columns and acts as a consolidated datetime stamp for each guest-id’s respective check-in. Because we are only looking at one month’s worth of data, this isn’t particularly necessary, but we can group by date, and create time series charts based on day of the month.

**NPSScore:** This column acts as a numerical value flag as it relates to the NPSType column. Promoters receive a +1, Detractors receive a -1, and Passives receive a 0.

**TotalSurveyScore:** This column acts as a summation of the total ‘score’ counted by each individual guest observation, as it relates to the 10 survey response questions.

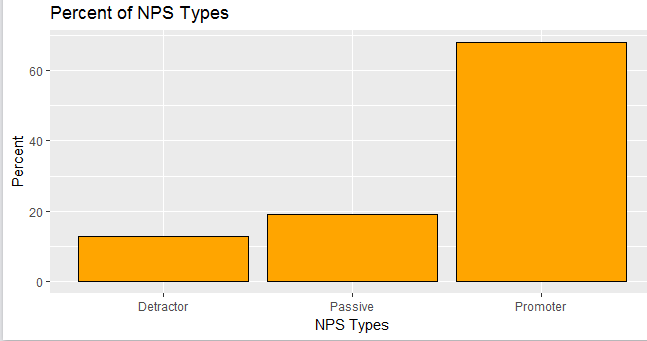
**TotalScorePossible:** This column acts as a summation of the total ‘score’ possible for each individual guest observation. If a guest did not answer one question, that question’s value was not added into this function.

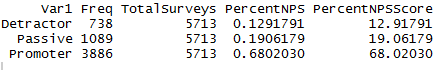
**Ratio:** This column takes the previous two derived columns and creates a scaled ratio that shows totalsurveyscore divided by totalscorepossible.

**NPS Score**

The total NPS Score for our data in question is determined by summing up the total count of responses rendered within the NPSType column. We secure counts for each of the three iterations of response types; Promoters, Detractors and Passives. We then calculate the percentage share of each of these response types. Passives do not play a role in our NPS Score other than representing part of the whole, so we focus on the percentage of Promoters of the whole against the percentage of Detractors of the whole- It is still important to include ‘Passive’ respondents as part of the total, aggregate grouping of respondents so we don’t distort the result. Upon subtracting these numbers and scaling into a whole number, we receive our true NPS Score. ***Appendix H: NPS Score Calculation***

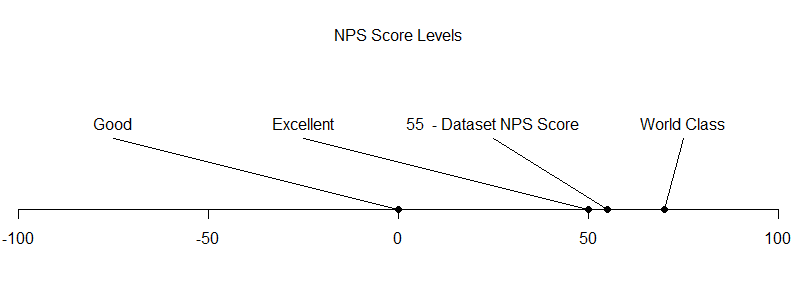
***Appendix I: Column Chart by Percent NPS Type***



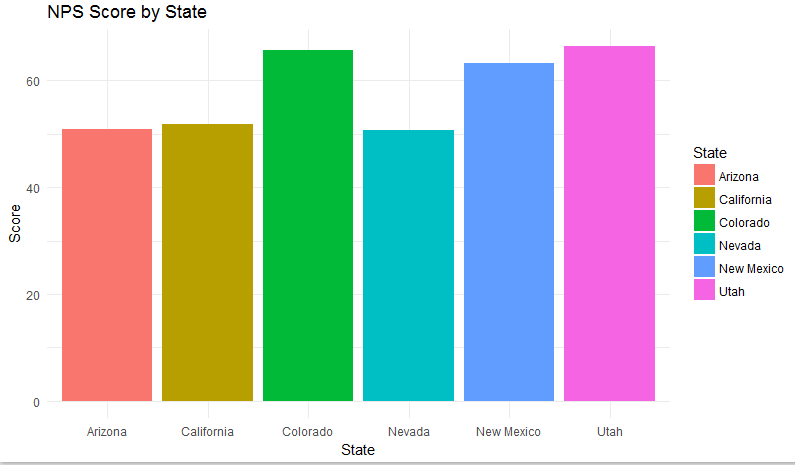


In the instance of our dataset, we see that there is a final NPS score of 55.1024. “Based on the global NPS standards, any score above 0 would be considered “good” (50 and above being excellent while 70 and above is considered “world class”)” (What is Net Promoter?).

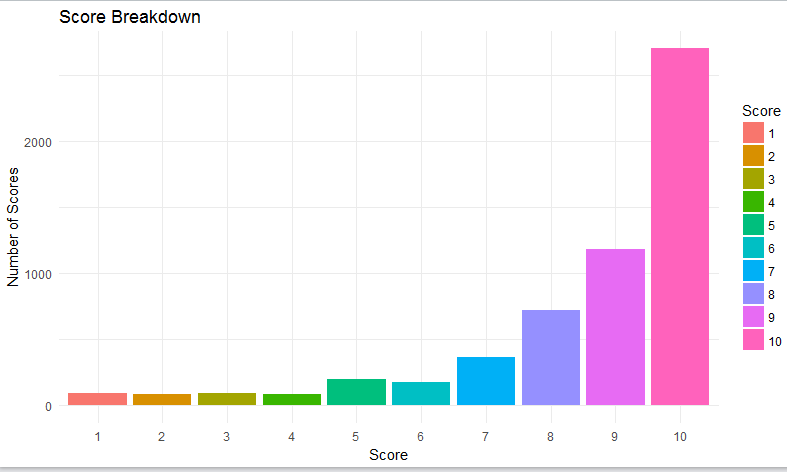
***Appendix J: Number line with NPS Score Plotted***



***Appendix K: NPS Score by State***



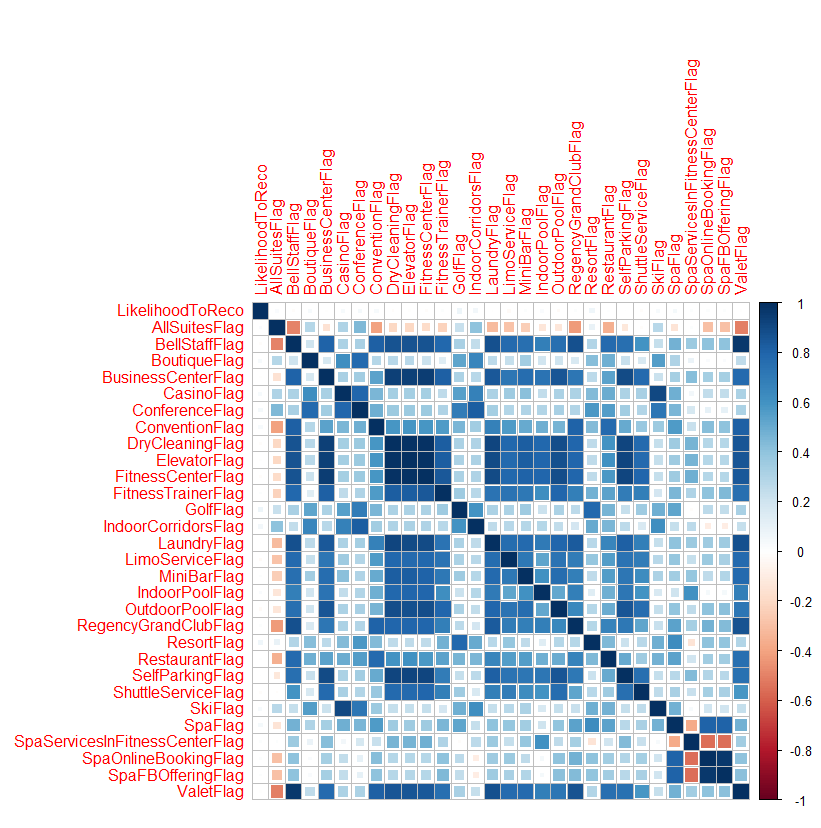
***Appendix L: Column Chart by Count of LTR Score***



**Correlation**

To better understand consumer behavior in relation to the amenities that a hotel has, or doesn’t have, we performed a correlation test and visualized the results. The below shows that there is no real relationship between a person’s Likelihood to Recommend Score and the specific offerings that a chain has on its grounds. What we can see, however, is the relationship between these amenities, themselves.

***Appendix M: Correlation Matrix***



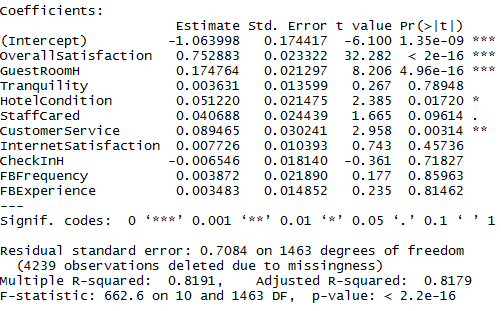
**Linear Regression**

Because our objective as consultants is essentially to advise our clients on achieving a higher NPS Score, our focus with linear regression is to understand the corresponding variance in our regressors, or x variables, as it relates to our y variable, which is our Likelihood to Recommend column. NPSType is simply a character derivation of the Likelihood to Reco column, with groupings of the latter associating to one of three results for our NPSType, or a main classifier of a customer.

Initially, it was thought to us that specific amenities would be the driving force of scoring likelihood to recommend, but it was discovered that our actual survey responses were instead the regressors. To expand on this, we tried to commit to a categorical regression, converting our amenity columns into binary, but that rendered a .03 adjusted R2, meaning the variance in what amenities a customer’s hotel had didn’t account for the variance in their scoring of their experience. Intuitively, this makes sense. When we, as consumers, book a hotel, there is already an understanding of what ‘amenities’ are included within the hotel’s grounds. As such, the likelihood that the absence of a service or extravagance, i.e. casino, is not prevalent, would seemingly have minor impact on our rating or overall experience.

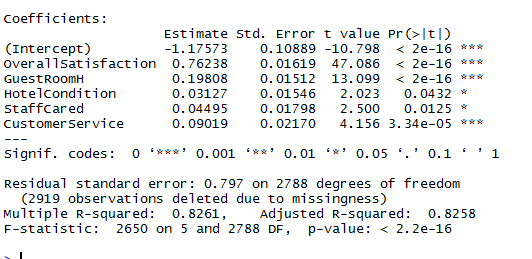
Running a multiple regression on the survey response scores, by column, against our likelihood to recommend score gives us the following model:

***Appendix N- Linear Modeling***



81.79% of our variance in Y is explained by our variance in these regressors, using adjusted R2. “Generally, it is better to look at adjusted R-squared rather than R-squared and to look at the standard error of the regression rather than the standard deviation of the errors” (Severson) If we rerun a similar model, but only take the significant variables - In this case anything that has a pvalue below .05 - We see the following:

***Appendix N- Linear Modeling***



Our adjusted R2 goes up to 82.58% and our variables are now more significant within this model.

We now know the drivers of our Likelihood to Recommend score, and in doing that, we better understand the drivers of our aggregated NPS Score, which would be Overall Satisfaction, Guest Room Rating,Hotel Condition,Staff Cared and Customer Service.

Not only do we know what drives variance in our NPS Score, but we can predict our NPS Score based on potential survey responses:

***Appendix O: Linear Modeling Prediction***



A person that would rate the hotel per the above would be classified as a promoter.

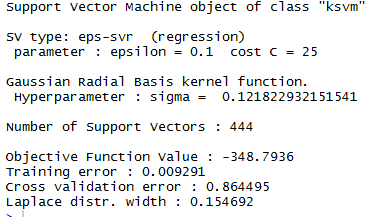
**Support Vector Machines**

To move on to more advanced modeling techniques, we will venture on to support vector machines, which are a form of supervised learning. “The reason support vector machines are considered a supervised learning technique is that we "train" the algorithm on an initial set of data (the "supervised" phase) and then we test it out on a brand-new set of data. If the training we accomplished worked well, then the algorithm should be able to predict the right outcome most of the time in the test data” (Saltz et al.). Before we start on building our models, we want to eliminate ‘Passive’ respondents from our dataset so that we are performing straightforward categorical analysis with two types of respondents. Then, we will remove all Nulls/NAs from the dataset so we are looking at complete cases only.

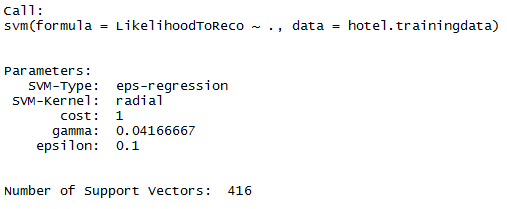
Summaries of Models Built:

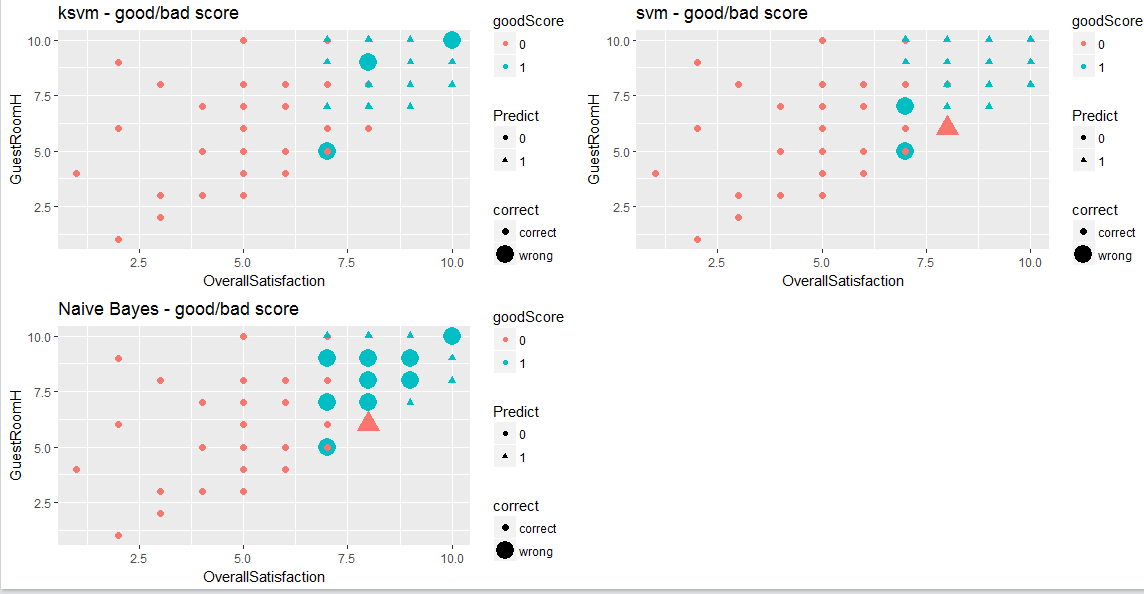
***Appendix P- Support Vector Machine and Plots***

**KSVM:**



**SVM:**





The KSVM model resulted in a rounded percent correct rate of 99%, missing the outcome on all but five predictions that were withheld with the testing data.

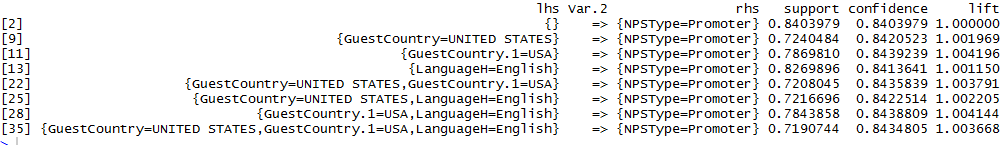
The SVM model resulted in a rounded percent correct rate of 99%, missing the outcome on all but four predictions that were withheld with the testing data.

The NB model resulted in a rounded percent correct rate of 96%, missing the outcome on 17 predictions that were withheld with the testing data.

**Association analysis**

Explained by its name, Association Rule Mining looks for associations between prevalent items within a group. Below, we show an example as it relates to LHS == Audience demographic data, and the RHS == Promoter, from the NPSType Column.

***Appendix Q: Association Rules Mining***



We can see that the highest levels of support and confidence in our specific inquiry do little but reinforce the summary statistics provided earlier. The issue with association rule mining, in this case, would be vast amount of similarities in our demographic data, as we are almost forcing an association between the inputs and the outputs due to a lack of variance amongst the two sides - Also, we are limiting the results to only show a single output of the ‘basket’..

**Recommendation**

We have several recommendations as it relates to our research on this data set, although it should first be noted that we are simply looking at one month’s worth of data, during a single year, with a geographical filter layered on top of that. Our first recommendation would be as it relates to marketing, and the audience that we are trying to either lure in through conquesting efforts directed at our competitors, or brand loyalist efforts to retain our existing customer base. If the former, we need to focus on the segments of the population that don’t currently represent a normal distribution of our sample. That could either reference a younger demographic, or potentially a foreign, international market that sees substantial air traffic to the Western United States. If, instead, we wanted to take the data we have and launch targeted advertising campaigns to prospective consumers that represent a majority of our total revenue, and a majority of our ‘Promoters’, then we could look at some of the summary statistics noted above. There aren’t 100% accuracy levels when predicting human behavior, so even if we were to target attributes, whether they be behavioral or economical, representative of our desired NPS Type, there could still be external factors that influence true satisfaction.

Through a series of modeling techniques, we can understand the drivers of net promoter score, as it relates to our data, and predict scores based on a set of values, be they factors or numbers.

To start, we conducted a multiple regression analysis on different sets of variables that we wanted to observe in relation to one another. “Multiple linear regression attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. Every value of the independent variable *x* is associated with a value of the dependent variable *y*” (Multiple Linear Regression). Our Likelihood to Recommend column will always be our dependent variable, or our Y variable, so we measured a series of audience segments, survey responses, and hotel amenities against that score. What we found was that the only real relationship between variables was seen when looking at the survey response data. Not all of the ratings were deemed significant in causing variance in the Likelihood To Recommend score however. Of the 10 ‘survey questions’, only the results of 5 proved to move the needle. Overall Satisfaction, Guest Room Rating, Customer Service, Staff Cared and Hotel Condition, in that order, were deemed the most integral part of the consumer experience. All subjective measures, but insightful nonetheless. Overall Satisfaction is closely linked with experience, which is defined by friendliness, and tidiness. Implementing a bottom-up business model would be a transformative solution here. “If employees are happy, satisfied, dedicated, and energetic, they'll take really diligent care of the customers. When the customers are happy, they come back. And that makes the shareholders happy” (The Rise of Southwest Airlines). Because our customer’s experience is reflective, at least in part, of their interactions with our hotel staff, implementing this model with increased focus on our employees would lead to better results.

Research was also conducted on the process of surveying guests, and ways to improve efficiency of that process. Of 200,000 GuestID records, we received responses from 2.5% regarding their stay, which puts us in the lower end - “Response rates can also fall below 2% (about 1 response for every 50 invitations sent) when the respondent population is less-targeted, when contact information is unreliable, or where there is less incentive or little motivation to respond” (Naicu et al.). If the sample were to increase, we would, in turn, reduce our margin of error within our models and increase the statistical relevance of our samples. One way to do this, as noted above, would be to offer incentives for valid survey completions, such as free points for their next booking. Not only does this motivate hotel-goers, but it acts as a loyalist technique to encourage further business with a specific chain. Another issue with the surveys rendered was that every question was on a scale, which makes it difficult to equate actual sentiment. We propose adding text fields into the incentivized surveys to better understand the consumer’s true feelings about their stay. Numerical classification works in scale, but it is challenging to change when you’re aren’t sure what exactly is broken within a process.

Regarding our correlation model, as stated earlier, we did not find any correlation between any of the amenities and the likelihood to recommendation score. We did find high correlations around a few different amenity columns, however. These were columns such as a business center, dry cleaning, fitness center and training, an elevator, laundry, outdoor pool, self-parking, and valet. We used these amenities in our models, rather than using all 20 ‘flag’ columns, to narrow down our number of factors. In doing so, we saw a lower rate of error. These amenities help support our linear model recommendations in high impact areas such as hotel condition, customer service, and staff cared. We recommend hotels, included within this analysis, to include these amenities, if possible, so that it can potentially increase hotel condition, thus increasing the rating of customer service to the guest. With amenities like valet, fitness training and dry cleaning, hotels could have more opportunity to give elevated levels of customer service which could turn into higher levels of recommendation from guest.

Our association rules mining model was utilized with our demographic data and yielded poor results, but this is likely due to not having more categories on the right-hand side to associate with. The association rule mining told us that English speaking Americans have a higher likelihood of being promoters, but English-speaking Americans happen to be a great portion of our data set. Therefore, it didn’t give us much insight compared to our other models.

With our supervised learning models, we can train, and then accurately predict a categorical outcome with miniscule error. As noted above, our Root Mean Squared Error falls below .9 on both our SVM and KSVM models, and we had a 96% or higher accuracy rating on predictions using all three of our models (SVM, KSVM, Naive Bayes).

***Appendix***

A. Column Chart by Brand- Page 2

*#Build Column Chart with Counts by Brand*

*brandtable <- table(dftestwscores$Brand)*

*branddf <- data.frame(brandtable)*

*str(branddf)*

*branddf$Freq <- as.numeric(branddf$Freq)*

*ggplot(branddf,aes(x=branddf$Var1, y=branddf$Freq)) + geom\_col(color="black",fill="orange") + ggtitle("Count of Guests per Brand") + xlab("Brand") + ylab("Count") #Column Graph Based on Percentage by NPSType*

B. Focused States- Page 3

*#Plot States Focused On from Sample*

*us<-map\_data("state")*

*US<- us[us$region=='california' | us$region=='arizona' | us$region=='nevada' | us$region=='utah' | us$region=='new mexico' | us$region=='colorado',]*

*rownames(US) <- NULL*

*US*

*cnames<-aggregate(cbind(US$long,US$lat)~US$region,data=US,FUN=function(x)mean(range(x)))*

*cnames$region<-cnames$`US$region`*

*cnames*

*US<-sqldf('select \* from US inner join cnames on US.region=cnames.region')*

*US*

*region<- ggplot(US, aes(map\_id=region))*

*region<-region + geom\_map(map=us,fill="orange",color="black")*

*region<-region + expand\_limits(x=US$long,y=US$lat) + coord\_map()*

*region<-region + geom\_label(aes(x=US$V1,y=US$V2),label=toupper(US$region),size=4)*

*Region*

C. Revenue by Zip Code- Page 3

*#Plotting revenue by zip code*

*meanlikelihood<- data.frame(dftestwscores$HotelZipCode, dftestwscores$Revenue\_USD)*

*meanlikelihood <- na.omit(meanlikelihood)*

*colnames(meanlikelihood) <- c("zip", "Revenue")*

*meanlikelihood$dftestwscores.HotelZipCode <- clean.zipcodes(meanlikelihood$zip)*

*str(meanlikelihood)*

*zipcodes <- data(zipcode) #saved as zipcode*

*merged <- merge(meanlikelihood, zipcode, by="zip")*

*str(merged)*

*score <- tapply(merged$Revenue, merged$state, sum) # calc mean of median by state*

*head(score)*

*head(merged)*

*merged$stateName <- state.name[match(merged$state,state.abb)]*

*merged$stateName <- tolower(merged$stateName)*

*head(merged)*

*us <- map\_data("state") # performed above, not adding anything new*

*minx <- min(merged$longitude)*

*maxx <- max(merged$longitude)*

*miny <- min(merged$latitude)*

*maxy <- max(merged$latitude)*

*mapZip <- ggplot(merged, aes(map\_id = stateName))*

*mapZip <- mapZip + geom\_map(map=us, fill="black", color="white")*

*mapZip <- mapZip + expand\_limits(x =maxx, y = maxy )*

*mapZip <- mapZip + geom\_point(data = merged,aes(x = merged$longitude, y = merged$latitude, color=merged$Revenue))*

*mapZip <- mapZip + coord\_map() + ggtitle("Revenue per zip code")*

*mapD <- mapZip + geom\_density\_2d(data = merged, aes(x = merged$longitude, y = merged$latitude))*

*mapD*

*theme\_update(plot.title = element\_text(hjust = 0.5))*

D. Descriptive Statistics- Page 4

*#Basis Descriptive Statistics*

*summary(dftestwscores)*

*str(dftestwscores)*

*sqldf("select GuestGender, count(GuestGender) from dftestwscores where HotelState = 'California' group by GuestGender")*

*sqldf("select AVG(LikelihoodToReco), GuestGender from dftestwscores group by GuestGender ")*

*sqldf("select AVG(LikelihoodToReco), GuestAgeRange from dftestwscores group by GuestAgeRange ")*

*sqldf("select AVG(LikelihoodToReco), HotelState from dftestwscores group by HotelState ")*

*sqldf("select AVG(LikelihoodToReco), Count(LikelihoodToReco), Brand from dftestwscores group by Brand ")*

*sqldf("select AVG(Revenue\_USD), GuestGender from dftestwscores group by GuestGender ")*

*sqldf("select Sum(Revenue\_USD), GuestGender from dftestwscores group by GuestGender ")*

*sqldf("select AVG(Revenue\_USD), GuestCountry from dftestwscores group by GuestCountry ")*

*sqldf("select Sum(Revenue\_USD), AVG(LikelihoodToReco) from dftestwscores group by LikelihoodToReco ")*

tapply(merged$Revenue, merged$state, sum)

E. Column Chart by Gender Percentage- Page 4

*#Build Column Chart with percentage by Gender*

*agedemo <- data.frame(sqldf("select Count(GuestGender), GuestGender from dftestwscores group by GuestGender "))*

*agedemo <- agedemo [-1,]*

*str(agedemo)*

*agedemo$percent <- agedemo$Count.GuestGender./5713*

*ggplot(agedemo,aes(x=agedemo$GuestGender, y=agedemo$percent)) + geom\_col(color="black",fill="orange") + ggtitle("Gender Split") + xlab("Gender") + ylab("Percent of Whole")*

F. Histogram by Guest Age Range- Page 5

histogram(dftestwscores$GuestAgeRange)

G. Line Chart Likelihood to Recommend Score by Age Range- Page 5

genderage <- data.frame(sqldf("select AVG(LikelihoodToReco), GuestAgeRange from dftestwscores group by GuestAgeRange "))

genderage <- genderage[-1,]

ggplot(genderage,aes(x=genderage$GuestAgeRange, y=genderage$AVG.LikelihoodToReco.)) + geom\_point(color="black",fill="orange") + ggtitle("LTR Score by AgeRange") + xlab("AgeRange") + ylab("Score")

H. NPS Score Calculation- Page 7

*#Calculate NPS Score*

*dfNPSScoreCalc<-data.frame(table(dftest$NPSType))*

*dfNPSScoreCalc<-dfNPSScoreCalc[-1,]*

*dfNPSScoreCalc*

*str(dfNPSScoreCalc)*

*dfNPSScoreCalc$TotalSurveys<-sum(dfNPSScoreCalc$Freq)*

*dfNPSScoreCalc$PercentNPS<-dfNPSScoreCalc$Freq / dfNPSScoreCalc$TotalSurveys*

*dfNPSScoreCalc$PercentNPSScore<- dfNPSScoreCalc$PercentNPS \* 100*

*rownames(dfNPSScoreCalc)<- NULL*

*dfNPSScoreCalc*

*ggplot(dfNPSScoreCalc,aes(x=dfNPSScoreCalc$Var1, y=dfNPSScoreCalc$PercentNPSScore)) + geom\_col(color="black",fill="orange") + ggtitle("Percent of NPS Types") + xlab("NPS Types") + ylab("Percent")*

I. Column Chart by Percent NPS Type- Page 8

*#Column Graph Based on Percentage by NPSType*

*dfNPSScoreCalc<-dfNPSScoreCalc[c(1,3),5]*

*dfNPSScoreCalc*

*NPSScore<-data.frame(dfNPSScoreCalc)*

*NPSScore<-apply(NPSScore,2,function(x)x-x[1])*

*NPSScore<-NPSScore[2,]*

*NPSScore #Run this to find total NPS Score*

J. Numberline with NPS score Plotted- Page 8

#Numberline showing how well the NPS score rates

xlim <-c(-100,100)

ylim<-c(0,100)

px<-c(0,50,round(NPSScore),70)

txtpx<-c('Good','Excellent',paste(round(NPSScore), ' - Dataset NPS Score'),'World Class')

py<-c(0,0,0,0)

lx.buf<-25

lx<-seq(xlim[1]+lx.buf,xlim[2]-lx.buf,len=length(px))

lx

ly<-20

par(xaxs='i',yaxs='i',mar=c(5,1,1,1))

plot(NA,xlim=xlim,ylim=ylim,axes=F,ann=F)

axis(1)

segments(px,py,lx,ly)

points(px,py,pch=16,xpd=NA)

text(lx,ly,txtpx,pos=3)

text(0,50,'NPS Score Levels') ## Levels based on website. "https://www.promoter.io/blog/good-net-promoter-score/"

K. NPS Score by State- Page 9

##NPS Score by State

dfNPSState<-data.frame(table(dftest$NPSType, dftest$HotelState))

dfNPSState$Freq<-as.numeric(dfNPSState$Freq)

dfNPSState

rownames(dfNPSState) <- NULL

totalSurveyed<-tapply(dfNPSState$Freq,dfNPSState$Var2, sum)

Var2<-rownames(totalSurveyed)

dfTotals<-data.frame(Var2,totalSurveyed)

rownames(dfTotals) <- NULL

dfTotals

dfNPSState<-merge(x=dfNPSState,y=dfTotals,by="Var2",all=TRUE)

dfNPSState

dfNPSState<-dfNPSState[dfNPSState$Freq!=0,]

rownames(dfNPSState)<-NULL

dfNPSState$PercentNPS<-dfNPSState$Freq / dfNPSState$totalSurveyed

dfNPSState$PercentNPSScore<- dfNPSState$PercentNPS \* 100

dfNPSState<-dfNPSState[dfNPSState$Var1!='Passive',]

dfNPSState$PercentNPSScore<- ifelse(dfNPSState$Var1=='Detractor',dfNPSState$PercentNPSScore\*-1,dfNPSState$PercentNPSScore\*1)

dfNPSState

NPSByState<-tapply(dfNPSState$PercentNPSScore,dfNPSState$Var2,sum)

NPSByState

dfNPSByState<-data.frame(rownames(NPSByState),NPSByState)

rownames(dfNPSByState)<-NULL

dfNPSByState<-na.omit(dfNPSByState)

rownames(dfNPSByState)<-NULL

dfNPSByState<-dfNPSByState[order(dfNPSByState$NPSByState,decreasing = TRUE),]

colnames(dfNPSByState)<-c('State','NPSScore')

rownames(dfNPSByState)<-NULL

dfNPSByState

gNPSbyState<-ggplot(dfNPSByState,aes(x=dfNPSByState$State, y=dfNPSByState$NPSScore,fill=dfNPSByState$State)) + geom\_col(stat="identity") + theme\_minimal() + ggtitle("NPS Score by State") + xlab("State") + ylab("Score") +scale\_fill\_discrete(name="State")#Column Graph Based on NPS Score by State

gNPSbyState

L. Column Chart by Count of Customers per Likelihood to Recommend Score- Page 9

##Likelihood to Recommend Score Breakdown

dfLikelihoodToRe<-data.frame(table(dftest$LikelihoodToReco))

dfLikelihoodToRe

gLikelihoodToRe<-ggplot(dfLikelihoodToRe,aes(x=dfLikelihoodToRe$Var1, y=dfLikelihoodToRe$Freq,fill=dfLikelihoodToRe$Var1)) + geom\_col() + theme\_minimal() + ggtitle("Score Breakdown") + xlab("Score") + ylab("Number of Scores") +scale\_fill\_discrete(name="Score")#Column Graph Based on NPS Score by State

gLikelihoodToRe

M. Correlation Matrix- Page 10

#Correlation Matrix

dfcorr <- dftestwscores[,c(10,30:58)]

for(i in 1:30)

{

dfcorr[,i] <- as.numeric(dfcorr[,i])

}

dfcorr$AllSuitesFlag <- as.numeric(dfcorr$AllSuitesFlag)

str(dfcorr)

cor(dfcorr)

dfcorr1 <- data.frame(cor(dfcorr))

dfcorr1 <- round(dfcorr1, 2)

install.packages("corrplot")

library("corrplot")

corrplot(cor(dfcorr), method= "square", title="Amenities Correlation Test")

N. Linear Modeling- Page 11

#Linear Modeling

hotelmodel1 <- lm(formula= LikelihoodToReco~OverallSatisfaction+GuestRoomH+Tranquility+HotelCondition+StaffCared+ CustomerService+InternetSatisfaction+CheckInH+FBFrequency+FBExperience, data= dftestwscores)

summary(hotelmodel1) #OverallSat, Guestroom,HotelCondition, CustomerService were the only statistically relevant variables.

hotelmodel2 <- lm(formula= LikelihoodToReco~OverallSatisfaction+GuestRoomH+HotelCondition+StaffCared +CustomerService, data= dftestwscores)

summary(hotelmodel2)

O. Linear Modeling Prediction- Page 12

newdata1 <- data.frame(OverallSatisfaction= 8, GuestRoomH=7, HotelCondition=6, StaffCared=8, CustomerService=9 )

P. Support Vector Models + Plots- Page 12

#FOR SUPPORT VECTORs

# IF we wanted to remove all of the passives from the data frame to just have promoter and detractors

dfwscoresfinal <- dftestwscores

dfwscoresfinal$NPSType <- as.character(dfwscoresfinal$NPSType)

str(dfwscoresfinal)

dfwscoresfinal$NPSType[dfwscoresfinal$NPSType == "Passive"] <- NA

dfwscoresfinal <- dfwscoresfinal[!is.na(dfwscoresfinal$NPSType),]

#Will want to remove all excess col that aren't need here

dfSVMS <- dfwscoresfinal[c(10:20,33,37,38,39,43,51,59)]

#Support Vector Machines

dfSVMS <- na.omit(dfSVMS)

nrows <- nrow(dfSVMS)

random.index <- sample(1:nrows)

head(random.index)

cutPoint <- floor(nrows/3\*2)

#Training Data (2/3 of total data sampled)

hotel.trainingdata <- dfSVMS[random.index[1:cutPoint],]

dim(hotel.trainingdata)

str(hotel.trainingdata)

#Testing Data (1/3 of total data sampled)

hotel.testingdata <- dfSVMS[random.index[(cutPoint+1):nrows],]

dim(hotel.testingdata)

str(hotel.testingdata)

#root mean squared error function

rmse <- function(error)

{

sqrt(mean(error^2))

}

require(kernlab)

require(e1071)

require(ggplot2)

##KSVM MODEL

model.ksvm.train <- ksvm(LikelihoodToReco ~., data=hotel.trainingdata, kernel = "rbfdot", kpar = "automatic", C = 25, cross = 3, prob.model = TRUE) #building the model

model.ksvm.train

model.ksvm.predict <- predict(model.ksvm.train, hotel.testingdata) #testing the model on the testing data

hotel.testingdata$error <- hotel.testingdata$LikelihoodToReco - model.ksvm.predict #computing the error between the predicted vs actual

head(hotel.testingdata)

rmse(hotel.testingdata$error) #Computing RMSE. RMSE = .87

##SVM MODEL

Model.svm.train <- svm(LikelihoodToReco ~., data=hotel.trainingdata) #building the model

Model.svm.train

model.svm.predict <- predict(Model.svm.train, hotel.testingdata)

hotel.testingdata$error <- hotel.testingdata$LikelihoodToReco - model.svm.predict #computing the error between the predicted vs actual

head(hotel.testingdata)

rmse(hotel.testingdata$error) #Computing RMSE. RMSE = .72

############################### Step 4 : Create a Variable

hotel.trainingdata$goodScore <- ifelse(hotel.trainingdata$NPSType == 'Detractor', 0, 1)

hotel.testingdata$goodScore <- ifelse(hotel.testingdata$NPSType == 'Detractor', 0, 1)

hotel.trainingdata$goodScore <- as.factor(hotel.trainingdata$goodScore)

hotel.testingdata$goodScore <- as.factor(hotel.testingdata$goodScore)

# remove "likelihood" from train data

hotel.trainingdata <- hotel.trainingdata[,-1]

# remove "likelihood" from test data

hotel.testingdata <- hotel.testingdata[,-1]

#Predicting Promoters V Detractors

#KSVM

model.ksvm.train <-ksvm(goodScore~., data=hotel.trainingdata, kernel = "rbfdot", kpar = "automatic", C = 50, cross = 3, prob.model = TRUE)

hotel.testingdata$predictedgoodScore <- predict(model.ksvm.train, hotel.testingdata, type = "response")

head(hotel.testingdata)

str(hotel.testingdata)

results <- table(hotel.testingdata$predictedgoodScore, hotel.testingdata$goodScore)

print(results)

percentCorrect <- (results[1,1]+results[2,2])/(results[1,1]+results[1,2]+results[2,1]+results[2,2])\*100

print(round(percentCorrect) )

#Plot KSVM Model

compgood1 <- data.frame(hotel.testingdata$goodScore, hotel.testingdata$predictedgoodScore)

colnames(compgood1) <- c("test", "pred")

compgood1$correct <- ifelse(compgood1$test==compgood1$pred,"correct","wrong")

Plot\_ksvm <- data.frame(compgood1$correct,hotel.testingdata$OverallSatisfaction,hotel.testingdata$GuestRoomH,hotel.testingdata$goodScore,compgood1$pred)

colnames(Plot\_ksvm) <- c("correct","OverallSatisfaction","GuestRoomH","goodScore","Predict")

ksvmgoodbadplot <- ggplot(Plot\_ksvm, aes(x=OverallSatisfaction,y=GuestRoomH)) +

geom\_point(aes(size=correct,color=goodScore,shape = Predict))+

ggtitle("ksvm - good/bad score")

ksvmgoodbadplot

#SVM

model.svm.train <-svm(goodScore~., data=hotel.trainingdata)

hotel.testingdata$predictedgoodScore <- predict(model.svm.train, hotel.testingdata)

head(hotel.testingdata)

str(hotel.testingdata)

results <- table(hotel.testingdata$predictedgoodScore, hotel.testingdata$goodScore)

print(results)

percentCorrect <- (results[1,1]+results[2,2])/(results[1,1]+results[1,2]+results[2,1]+results[2,2])\*100

print(round(percentCorrect) )

#Plot SVM Model

compgood2 <- data.frame(hotel.testingdata$goodScore, hotel.testingdata$predictedgoodScore)

colnames(compgood2) <- c("test", "pred")

compgood2$correct <- ifelse(compgood2$test==compgood2$pred,"correct","wrong")

Plot\_svm <- data.frame(compgood2$correct,hotel.testingdata$OverallSatisfaction,hotel.testingdata$GuestRoomH,hotel.testingdata$goodScore,compgood2$pred)

colnames(Plot\_svm) <- c("correct","OverallSatisfaction","GuestRoomH","goodScore","Predict")

svmgoodbadplot <- ggplot(Plot\_svm, aes(x=OverallSatisfaction,y=GuestRoomH)) +

geom\_point(aes(size=correct,color=goodScore,shape = Predict))+

ggtitle("svm - good/bad score")

svmgoodbadplot

#NAIVE BAYES

model.naivebayes.train <-naiveBayes(goodScore~., data=hotel.trainingdata)

hotel.testingdata$predictedgoodScore <- predict(model.naivebayes.train, hotel.testingdata)

head(hotel.testingdata)

str(hotel.testingdata)

results <- table(hotel.testingdata$predictedgoodScore, hotel.testingdata$goodScore)

print(results)

percentCorrect <- (results[1,1]+results[2,2])/(results[1,1]+results[1,2]+results[2,1]+results[2,2])\*100

print(round(percentCorrect) )

#Plot NB Model

compgood3 <- data.frame(hotel.testingdata$goodScore, hotel.testingdata$predictedgoodScore)

colnames(compgood3) <- c("test", "pred")

compgood3$correct <- ifelse(compgood3$test==compgood3$pred,"correct","wrong")

Plot\_NB <- data.frame(compgood3$correct,hotel.testingdata$OverallSatisfaction,hotel.testingdata$GuestRoomH,hotel.testingdata$goodScore,compgood3$pred)

colnames(Plot\_NB) <- c("correct","OverallSatisfaction","GuestRoomH","goodScore","Predict")

NBgoodbadplot <- ggplot(Plot\_NB, aes(x=OverallSatisfaction,y=GuestRoomH)) +

geom\_point(aes(size=correct,color=goodScore,shape = Predict))+

ggtitle("Naive Bayes - good/bad score")

NBgoodbadplot

grid.arrange(ksvmgoodbadplot,svmgoodbadplot,NBgoodbadplot, nrow=2)

Q. Association Rules Mining: Page 13

#Association Rule Mining

#Using Audience Demographic Data

dfARM <- dftestwscores

dfARM$NPSType <- as.character(dfARM$NPSType)

str(dfARM)

dfARM$NPSType[dfARM$NPSType == "Passive"] <- NA

dfARM <- dfARM[!is.na(dfARM$NPSType),]

dfARM <- dfARM[,c(1:9,59)]

dfARM <- na.omit(dfARM)

str(dfARM)

dfARM$Revenue\_USD <- as.factor(dfARM$Revenue\_USD)

dfARM$NPSType <- as.factor(dfARM$NPSType)

ARM <- apriori(dfARM, parameter = list(support=.5,confidence=.3))

ARMdf <- data.frame(inspect(ARM))

ARMdf1 <- ARMdf[ARMdf$rhs == '{NPSType=Promoter}',] #Filtering just for good score rhs

ARMdf1

***Work Cited***

“Glossary of Statistical Terms.” *Statistics at UC Berkeley*, [www.stat.berkeley.edu/~stark/SticiGui/Text/gloss.htm#law\_of\_large\_numbers](http://www.stat.berkeley.edu/~stark/SticiGui/Text/gloss.htm#law_of_large_numbers).Accessed September 2017

“Multiple Linear Regression”, *Yale Statistic*,[www.stat.yale.edu/Courses/1997-98/101/linmult.htm. Accessed September 2017](http://www.stat.yale.edu/Courses/1997-98/101/linmult.htm.%20Accessed%20September%202017)

Naicu, Octavian, et al. “What's a Good Survey Response Rate?” *SurveyGizmo*, [www.surveygizmo.com/survey-blog/survey-response-rates/](http://www.surveygizmo.com/survey-blog/survey-response-rates/). Accessed September 2017

Robert Nau, “A Good Value for R-Squared?”, *Duke University: The Fuqua School of Business*,<https://people.duke.edu/~rnau/rsquared.htm>, Accessed September 2017

Saltz, Jeffrey S., and Jeffrey M. Stanton. *An Introduction to Data Science*. SAGE, 2018.

“The Rise of Southwest Airlines”, *Advance Business Consulting*,

[www.advancebusinessconsulting.com/advance!/strategic-alignment/strategic-alignment-business-cases/the-rise-of-southwest-airlines.aspx. Accessed September 2017](http://www.advancebusinessconsulting.com/advance!/strategic-alignment/strategic-alignment-business-cases/the-rise-of-southwest-airlines.aspx.%20Accessed%20September%202017)

“What Is Net Promoter?” *Net Promoter Network*, [www.netpromoter.com/know/](http://www.netpromoter.com/know/). Accessed September 2017

Severson, Dana., “What's a Good NPS Score?” *Promoter.io Blog*, 14 Aug. 2017, [www.promoter.io/blog/good-net-promoter-score/](http://www.promoter.io/blog/good-net-promoter-score/). Accessed September 2017

**Data Munging Code**

#CLEAR ENVIRONMENT AND INSTALL INITIAL PACKAGES

rm(list = ls(all = TRUE))#Clear Enviroment

#specify the packages of interest

packages=c("maps","zipcode","mapproj","ggmap","ggplot2","gdata", "sqldf", "kernlab","e1071","gridExtra","ggplot2", "caret", "arules", "reshape2")

#use this function to check if each package is on the local machine if a package is installed, it will be loaded if any are not, the missing package(s) will be installed and loaded

package.check <- lapply(packages, FUN = function(x) {

if (!require(x, character.only = TRUE)) {

install.packages(x, dependencies = TRUE)

library(x, character.only = TRUE)

}

})

#BEGIN Munging

#Read File Into R. DF = All Data w/ preliminary column removal

df <- read.csv(file= "C:/Users/jdine/Desktop/out-201501.csv")[,c(9,12,17,18,23,26,54,59,65,73,76,89:92,106:111,137:147,167:171,175,176,179,182,199:227,232), ]

df <- data.frame(df, stringsAsFactors = FALSE)

#Entire Southwest

df1 <- sqldf("select \* from df WHERE State\_PL like '%California%' or State\_PL like '%Arizona%' or State\_PL like '%Nevada%' or State\_PL like '%Utah%' or State\_PL like '%New Mexico%' or State\_PL like '%Colorado%' ") #Filter for California only. DF1 = All data where state = Soutwest states w/ preliminary column removal

df1$CheckInMonth\_Year <- as.Date(paste(format(as.Date(df1$CHECK\_IN\_DATE\_C), "%Y-%m"), "-01", sep=""),format="%Y-%m-%d") #add M/Y Column based on check in date

df1 <- df1[-c(3,4,6,11:14)] #Remove check in/check out date variables + other dupes

#Plot States Focused On from Sample

us<-map\_data("state")

US<- us[us$region=='california' | us$region=='arizona' | us$region=='nevada' | us$region=='utah' | us$region=='new mexico' | us$region=='colorado',]

rownames(US) <- NULL

US

cnames<-aggregate(cbind(US$long,US$lat)~US$region,data=US,FUN=function(x)mean(range(x)))

cnames$region<-cnames$`US$region`

cnames

US<-sqldf('select \* from US inner join cnames on US.region=cnames.region')

US

region<- ggplot(US, aes(map\_id=region))

region<-region + geom\_map(map=us,fill="orange",color="black")

region<-region + expand\_limits(x=US$long,y=US$lat) + coord\_map()

region<-region + geom\_label(aes(x=US$V1,y=US$V2),label=toupper(US$region),size=4)

region

#Rename Columns

colnames(df1) <- c("GuestID", "RoomDescription", "POVCode", "NightlyRate", "LengthOfStay", "GuestCountry", "Revenue\_USD", "GuestRoomFloor", "GuestState", "GuestCountry", "GuestGender", "GuestAgeRange", "POV\_H", "LanguageH", "LikelihoodToReco", "OverallSatisfaction", "GuestRoomH", "Tranquility", "HotelCondition", "CustomerService", "StaffCared", "InternetSatisfaction", "CheckInH", "FBFrequency", "FBExperience", "HotelCity", "HotelState", "HotelRegion", "HotelZipCode", "HotelCountry","HotelLatitude", "HotelLongitude", "NPSGoal", "Brand", "AllSuitesFlag", "BellStaffFlag", "BoutiqueFlag", "BusinessCenterFlag", "CasinoFlag", "ConferenceFlag", "ConventionFlag","DryCleaningFlag", "ElevatorFlag", "FitnessCenterFlag", "FitnessTrainerFlag","GolfFlag", "IndoorCorridorsFlag", "LaundryFlag", "LimoServiceFlag", "MiniBarFlag","IndoorPoolFlag", "OutdoorPoolFlag", "RegencyGrandClubFlag", "ResortFlag", "RestaurantFlag", "SelfParkingFlag", "ShuttleServiceFlag", "SkiFlag", "SpaFlag", "SpaServicesInFitnessCenterFlag", "SpaOnlineBookingFlag", "SpaFBOfferingFlag", "ValetFlag", "NPSType", "CheckInMonth\_Year")

dftest <- df1[!is.na(df1$LikelihoodToReco),] #Filter based off likelihood to reco column

#NPSScore column + classification (need as.int)

dftest$NPSScore <- as.character(dftest$NPSType)

dftest$NPSScore[dftest$NPSScore == "Promoter"] <- "1"

dftest$NPSScore[dftest$NPSScore == "Detractor"] <- "-1"

dftest$NPSScore[dftest$NPSScore == "Passive"] <- "0"

#Create column that sums all survey response numbers divided by total response possible, run the regression on that value compared to flag columns

#Convert surveyscores to as.numeric

for(i in 15:25)

{

dftest[,i] <- as.numeric(dftest[,i])

}

#Column that adds all surveyscores together

dftest$totalsurveyscore <- apply(dftest[,c(16:25)], 1, sum, na.rm= TRUE)

#######Step To create updated column with total survey score possible, and column with ratio of total survey score/total survey score possible

#Create new df

dftestwscores <- dftest

#Converting Y/N Flags on amenities to binary

for(i in 35:63)

{

dftestwscores[,i] <- as.character(dftestwscores[,i])

dftestwscores[,i][dftestwscores[,i] == "Y"] <- "1"

dftestwscores[,i][dftestwscores[,i] == "N"] <- "0"

dftestwscores[,i][dftestwscores[,i] == NULL] <- "Null"

dftestwscores[,i] <- as.factor(dftestwscores[,i])

}

#override original survey columns with 0/1 to calculate total survey response possible

for(i in 16:25)

{

dftestwscores[,i][dftestwscores[,i] > 0] <- 1

record <- dftestwscores[[i]]

record[sapply(record, is.na)] <- 0

dftestwscores[[i]] <- record

}

dftestwscores$totalpossible <- rowSums(dftestwscores[,c(16:25)]) \*10 #perform row addition on survey responses not equal to 0

dftestwscores$ratio <- (dftestwscores$totalsurveyscore/dftestwscores$totalpossible)\*100 #create ratio column

dftestwscores[,c(15:25)] <- dftest[,c(15:25)] #Return survey columns to original values

#Fixing the Classifications on all variables that are messed up

dftestwscores$LikelihoodToReco <- as.numeric(dftestwscores$LikelihoodToReco)

dftestwscores$Revenue\_USD <- as.numeric(dftestwscores$Revenue\_USD)

dftestwscores$LengthOfStay <- as.numeric(dftestwscores$LengthOfStay)

dftestwscores$HotelZipCode <- as.factor(dftestwscores$HotelZipCode)

dftestwscores$NPSScore <- as.factor(dftestwscores$NPSScore)

#kill additional useless columns not needed for analysis

dftestwscores <- dftestwscores[-c(1,3,4,5,8)]

str(dftestwscores)

**Package Glossary**

**ggplot2** – is a system for ‘declaratively’ creating graphics, based on “the Grammar of Graphics”.

**map** – the r package for geographical maps

**zipcode** – this package contains a database of city, state, latitude and longitude information for U.S. ZIP codes from the CivicSpace DataBase (August 2004) and augmented by Daniel Coven’s federalgovernmentzipcodes.us web site (updated January 22,2012).

**mapproj** – converts latitude and longitude into projected coordinates.

**ggmap** – plots the raster object produced by get\_map

**gdata** – the r package for sample data

**sqldf** – SQL select on data frames

**kernlab** – functions that can be used in a pipeline implemented by magrittr

**e1071** – functions that can be used in a pipline implemented by magrittr

**gridExtra** – miscellaneous functions for “Grid” Graphics. It provides a number of user-level functions to work with “grid” graphics, notable to arrange multiple grid-based plots on a page, and draw tables.

**caret** – functions that can be used in a pipeline implemented by magrittr

**arules** – provides the infrastructure for representing, manipulating and analyzing transaction data and patterns. Also provides interfaces to C implementations of the association mining algorithms Apriori and Eclat by C. Borgelt

**reshape2** – flexibly restructure and aggregate data using just two functions: melt and ‘dcast’

\*All information were obtained from [www.rdocumentation.org](http://www.rdocumentation.org)