# Outmatch

## Priyanka Kalmane’s project Journal

Table of Contents

[Outmatch 1](#_Toc482346863)

[Priyanka Kalmane’s project Journal 1](#_Toc482346864)

[Meeting minutes and important discussions 9](#_Toc482346865)

[Code for classification models 12](#_Toc482346866)

[Tasks carried out by me: Data cleaning, RPART Classification models, RPART for Predictability problem statement. 12](#_Toc482346867)

[References: 18](#_Toc482346868)

**Introduction:**

Outmatch is a company based out of Texas that develops behavioral assessment applications for hiring employees. Outmatch aims at building winning teams by taking a data-driven approach to talent selection and development at all levels in its client’s organization. The service provided by Outmatch increases the Quality of hire, Exceptional guest service and performance. It decreases the turnover and time to hire.

**Problem Statement:**

If we closely study the Outmatch data given to us, the data consists of performance ratings given by supervisor to the subordinates based on various qualities. All this contributes to the overall rating.

* Outmatch currently follows linear regression in order to score the employees based on the weights. Our team’s hypothesis is that using the individual assessment item/ raw scores will yield better results compared to the existing approach.
* The team also aims at finding the accuracy of the existing model.

**Data source:**

Total number of observations:4197

Total number of variables: 588

16 variables are related to personality assessment raw scores and weighted scores. The rest are related to candidate information, supervisor ID and individual assessment items.

Various behavioral aspects such as Work intensity, positive attitude, sociability etc. are assessed on a 6-point scale (Agree, Disagree, Strongly Agree, Strongly Disagree, Slightly Agree, Slightly Disagree). (The dataset given to us is a 4-point scale). Each question pertains to assessment of a certain quality and these scores are what are referred to as “Individual Constructs” throughout this document. These individual constructs are further translated into raw scores for a specific quality. These are referred to as “Raw scores of the construct”. The qualities are referred to as “Constructs”. In order to distinguish good performers from bad, various “cuts” are made i.e, the raw scores are divided into various bins and a certain weight (Outmatch’s internal process) is assigned to each bin.

Example:

RECODE STD4\_HR\_ASSERT

(0 THRU 24 = 0.5 )

(25 THRU 26 = 3 )

(27 THRU 34 = 5 )

(35 THRU Highest = 1.5 )

INTO w\_Assertiveness.

The values 0.5,3,5,1.5 are called “Weighted scores of the constructs”. There are various constructs that are reverse coded such as Assertiveness thus making them non-linear. More explanation provided later.

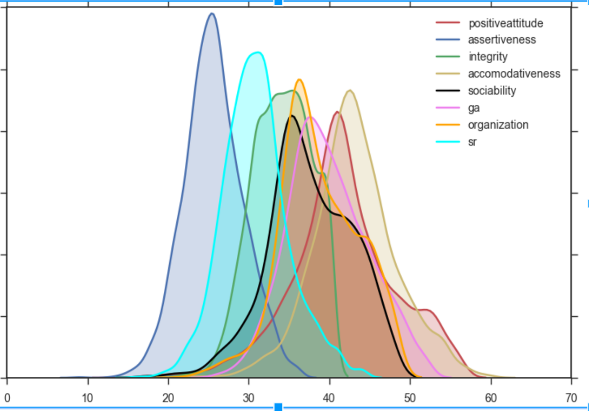
There is also another variable called “Overall Rating” provided by the supervisor which is between 1-5. In order to normalize the supervisor rating, (x-mean/std.dev) formula is applied and ZOverallRating is calculated.

**Data understanding:**

We have used the following Analytical techniques / methodology for analyzing the Data:

1. Summary of Statistics for each variable
2. Frequency of each factors
3. Using Box Plots to visually represent them
4. Tools used: R, Tableau & Excel
5. Techniques: Box Plot, Histogram, Bar Chart, Line Chart, Correlation Matrix, Multiple Linear Regression
6. We have used R Programming environment and Microsoft Excel for our analysis and Tableau for data

After carrying some initial exploratory data analysis, it was found that though most Raw scores distribution of constructs look normal, from carrying out normality tests like shapiro, f.test, we found that they do not follow a normal distribution. Thus models where parametric variables are used cannot be used for this problem. ZOverallRating ranges from -2.11 to 1.91 and follows a normal distribution. There are 8 constructs: STD4\_HR\_ASSERT-Assertiveness, STD4\_HR\_ACCOM-Accommodation, STD4\_HR\_GA-Work Intensity, INTEG10\_VAL\_STD4\_HR-Work Ethic, STD4\_HR\_SR- Work Independence, STD4\_HR\_ORG- Process Focused, STD4\_HR\_POSATT- Positive View People, STD4\_HR\_SOC-Sociability. The individual constructs all have missing values, except Assertiveness which is partially complete. Thus, we have considered only complete cases for our analysis. There are 605 observations.



**Approach followed by Outmatch:**

As mentioned before, Outmatch follows a manual approach to assigning the weights and making the raw scores bins. Using linear regression, they find the most significant variables, find the proportion of beta significance values for each construct and use that value to calculate the Overall Fit given by the formula,

Corr\_Relevance\_Overall\_Fit = (w\_Assertiveness\*.12 + w\_WorkIntensity\*.19 + w\_WorkEthic\*.17 + w\_WorkIndepend\*.13 + w\_ProcessFocused\*.24 + w\_PositiveViewPeople\*.15)

The weights shown in the beginning of document are plugged into the above formula and the fit is calculated. We have replicated the above technique to establish the accuracy of the baseline model.

**Analytics approach**

The Analytical Approach involved the following (not necessarily in the order) activities:

* Data quality check
* Data cleaning and data preparation
* Study each of the variables by exploring the data
* Study the variables for its relevance for the study
* Identifying Y variable(s). (ZOverall rating).
* Performing Univariate analysis for all variables.
* Division of data into train and test
* Model Development
* Final Model
* Model Validation & Model Validation on Test
* Recommendations and future scope

**Data management:**

We have 2 years data of 4196 candidates. The data consists of information about 7 personality assessment parameters such as Work Ethic, Accommodation, Assertiveness, Sociability, Work Intensity, Positive View of People, Process-Focused, Work Independence.

**Data Quality:**

1. Out of 4197 observations only 605 are complete observations (excluding the individual items variables. Total number of variables considered)
2. All the 743 observations are from Retail for the position of Manager and is 2015 data.
3. 1598 observations out of the total are from Restaurant industry, 825 from Retail were conducted in the month of April.

**Study each of the variables by exploring the data:**

After looking at the entire dataset that has over 588 variables, we concluded that though it is important to focus on the individual assessment constructs, we will do so at a later stage. Right now, we extracted the first 33 variables which consist of the raw scores, weights and other candidate related information along with Overall rating and ZOverall rating.

**Identifying Y variable(s):**

We could consider either Overall Rating or ZOverall rating as dependent variable. Overall rating is a factor variable between values 1 to 5 and ZOverall rating is a continuous variable between -2.62 to 3.10. Since we aim into grouping employees into Top 20, Middle 60 and Bottom 20 bins, ZOverall Rating is considered as the dependent variable. Also because, it is scaled and normalized depending on the supervisors.

**Performing Univariate linear regression analysis for all variables:**

In order to understand if we’re losing any predictive power as move from raw scores to weights and their relationship with Overall Rating, we carried out 2 Linear Regression models.

1. Using ZOverallRating as dependent variable. Raw scores as independent variables.
2. Using ZOverallRating as dependent variable. Weights as independent variables.

The results obtained were as follows:

|  |  |  |
| --- | --- | --- |
| **Model** | **Raw Scores R^2** | **Weighted Scores R^2** |
| **DP - Z Overall Rating** |  |  |
| Linear - All Constructs | 0.08605 | 0.08513 |
| Linear - Accommodative | 0.01086 | Not Significant |
| Linear - Work Intensity | 0.04806 | Not Significant |
| Linear - Organization | 0.05257 | Not Significant |
| Linear - Sociability | Not Significant | 0.000969 |
| **DP - Overall Rating** |  |  |
| Linear - All Constructs | 0.1096 | 0.09677 |
| Linear - Accommodation | 0.0009467 | Not Significant |
| Linear - Work Intensity | 0.05322 | 0.04926 |
| Linear - Sociability | 9.579e-05 | 0.0004143 |
| Linear Positive Attitude | 0.02462 | Not Significant |

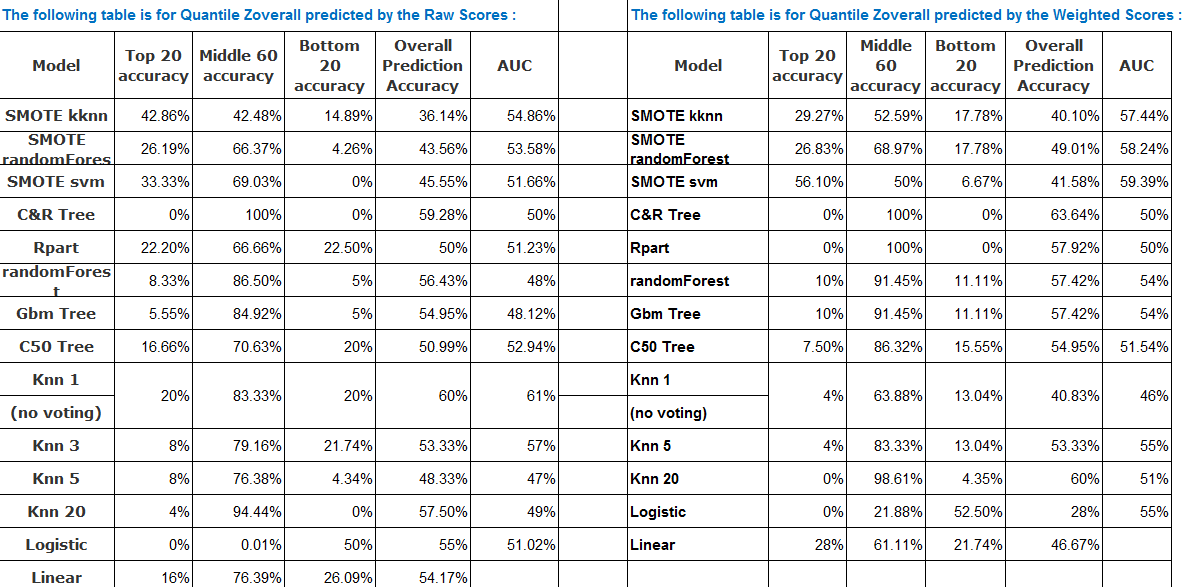
As we can see, there is a slight difference between Raw scores and Weighted scores, though not significantly different to arrive at a conclusion that raw scores are indeed better. Thus, our hypothesis still stands unverified.

**Division of data into train and test:**

We try classification techniques to identify top 20, middle 60 and bottom 20 employees. The data set is divided into train – 80% of the data and test – 20% of the data.

**Model development:**

* Using quantile function, we divide the ZOverallRating into Top 20, Middle 60 and Bottom 20. These values are then used to make 3 classes for Top 20, Middle 60 and Bottom 20. We train the train data set using classification algorithms and test the accuracy against the test data set.



It is important to note that Outmatch is looking for employees who fall under top 20 and bottom 20 bins more than the middle 60. Thus, we need higher accuracy in that section. Also, from the above results, though difficult to compare, we can say that SMOTE kknn performs the best with Raw scores as independent variables.

* The baseline model follows the formula as mentioned before:

Compute Corr\_Relevance\_Overall\_Fit = (w\_Assertiveness.12 + w\_WorkIntensity.19 + w\_WorkEthic.17 + w\_WorkIndepend.13 + w\_ProcessFocused.24 + w\_PositiveViewPeople.15).

The same was recreated from the beta values that we obtained after running linear regression. The results are as follows:

Outmatch Baseline (Regression all coefficients):

Top 20: 31.97%

Middle 60: 58.19%

Bottom 20% 36.75%

Outmatch Baseline (Regression minus negative coefficients)

Top 20: 35.25%

Middle 60: 62.84%

Bottom 20:  35.90%

* Individual items model

In this we considered the individual items as independent variables and conducted linear regression to check how significant its relationship is with Z scores. The results obtained are as follows:

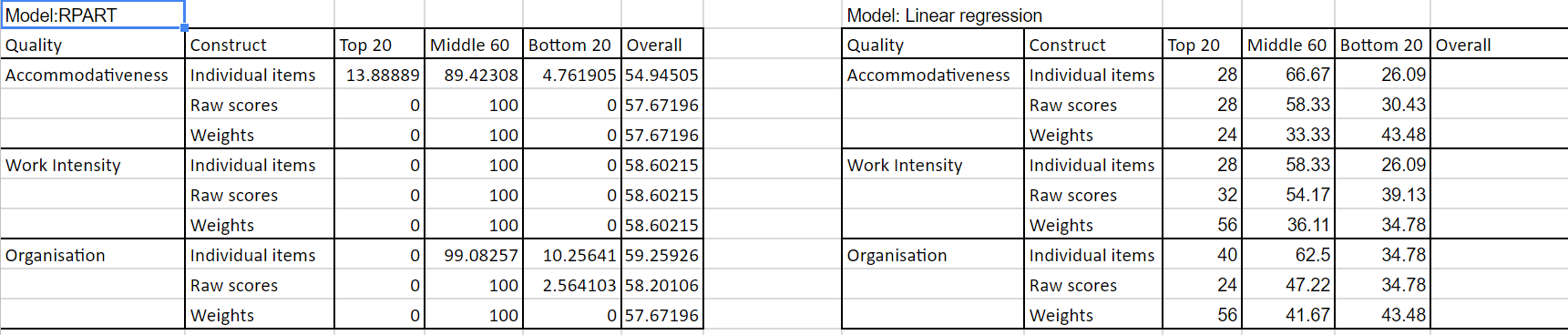
|  |  |  |
| --- | --- | --- |
| **Quality** | **Multiple R squared** | **p-value** |
| Assertiveness | 0.02651 | 0.06528 |
| Accommodativeness | 0.03873 | 0.07375 |
| Work Intensity | 0.07819 | 5.26E-06 |
| Sociability | 0.02358 | 0.2852 |
| Positive Attitude | 0.04248 | 0.02687 |
| Organization | 0.09095 | 8.16E-08 |
| Work Independence | 0.06351 | 4.89E-05 |

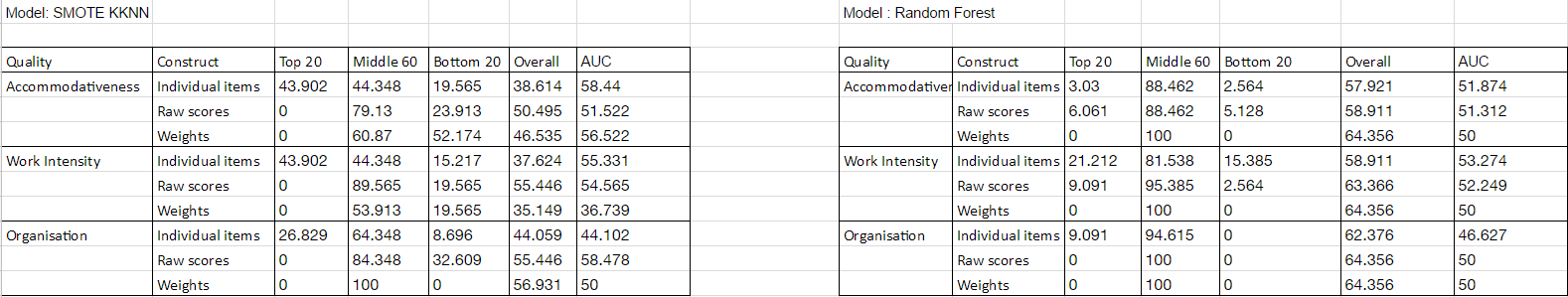
* Looking at Individual items, weighted scores and raw scores individually against ZOverall as dependent variable

In this case, we divide the data set into test and train, divide the continuous ZOverall into classes using quantile function and build classification models like before. We find the accuracy of the test set for 3 qualities : Accommodativeness, Work intensity and Organization.

We use the baseline model to replicate the same using individual items, raw scores and weights as independent variables to find the overall fit. After finding the value of the overall fit, we find the cuts by using the quantile function and check the accuracy of the model by comparing it against the existing ZOverall classes.

The results obtained for all models are as follows:



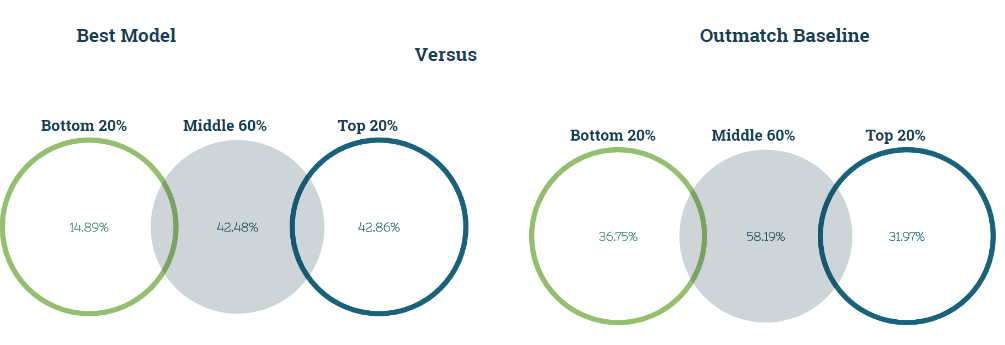


**Model validation**

Cross validation technique has been used. Due to a low number of records to work with, there may have been overfitting. This could be further validated via k-fold validation, dividing the data set into test, train and validation sets.

**Recommendations and future scope**

The major insights we obtained from this activity were:

* Both individual items and weights seem to have a high predictability power towards Z-Overall. However, we see that for Individual items, we have greater predictability compared to the others for SMOTE kknn. It depends on the client as to which one they would like to choose.
* 

Our model, SMOTE KKNN outperforms the Baseline for Top 20%. However, the same cannot be said for middle 60 and bottom 20. (We may be overfitting). Thus, it should be taken with a grain of salt.

* Our recommendation would be an ensemble model, which chooses the best of all models for all three classes.
* Also, for future scope we can look at demographics, ethnicity. More the data points, higher the accuracy.

## Meeting minutes and important discussions

**Date: 2/1/2017**

|  |  |
| --- | --- |
| **Task : Reading documentation and preliminary analysis** |  |
| **Questions** | **Answers as discussed** |
| Who defines the competencies needed for success in that job? | Generic constructs |
| How is online reference checking done? | Online tool |
| How are 'slightly agree' and 'slightly disagree' answers measured?  Which version is used in the questionairre? The 4 point or the 6 point | Reverse coded data |
| On slide 5, what does the blue range represent? |  |
| How is the comparison with the other candidates carried out?  Do you also consider age, gender, cultural background as a factor before comparison? | No. Can look into it. |
| What is Preliminary test battery? |  |

**Date: 2/11/2017**

|  |
| --- |
| **Questions to be discussed with the client** |
| How is the Overall Rating calculated? |
| What is the reason 0.5, 1.5, 3 and 5 have been used to assign weights? |
| Have the same been weights assigned to the other variables? If not, what has been used? |
| What is the significance of the other variables like Customer Focus, AA1079\_4pt and so on? Is there a document which describes what each of these variables are? |
| R\_sex has 4 values. Age Group has 4, Ethnic group has 9 - can we have descriptions of this? |
| What features would you like to see that would make your job easier for you? |
| What has been the most challenging aspect of the current model? |

|  |
| --- |
| **Answer obtained** |
| Rating is assigned by the supervisor |
| Internal company calculation |
| SPSS file shared for understanding |
| Those are individual items that are aggregated to form the raw scores |
| Given in the file |
| Classification model possiblities? |
| Manual approach to weighting |

**Date: 2/13/2017**

|  |
| --- |
| **Minutes of the meeting** |
| Manual approach - weighting |
| Binning - magnitude of the mean differences |
| We need to be able to understand relationship is linear and curvilinear |
| z-overall rating – dependent |
| Q to Y – raw personality variables Z through AG – Weighted values – Today’s model |
| Best scoring model for hiring process |
| Statistical significance - Across a wide range of jobs,the best combination of variables |
| Hybrid approach – personality and performance |
| Column K – ignore |

**Date: 2/20/2017**

|  |
| --- |
| **Questions to be discussed with the client** |
| From the hourly amalgamation csv file given, why isn't 'work independence' considered? |
| How do you decide which ones are significant? |
| How is the Zoverall Rating calculated from the model? |
| The weights assigned still not clear |
| Model run on raw scores match closely with the model in the excel given to us |

|  |
| --- |
| **Answers** |
| Only significant variables are considered |
| The beta value |
| Supervisor specific (x-mean/std.dev) |
| Another spss file to be sent by Keith |
| Use raw scores for analysis in that case |

**Date: 3/3/2017**

|  |
| --- |
| **Minutes of the meeting:** |
| Focus on establishing curvilinear relationships |
| Using ctrees can help in making cuts |
| Think about classification methods |
| Information about how raw scores are derived is important to understand.  We may be losing on the quality of data while deriving the raw scores from the individual item. |

**Action item: To meet with Prof. Jason**

**Date: 6/3/2017**

|  |
| --- |
| **Meeting with Prof. Jason** |
| Part 1 : Are you losing something when you aggregate? Did you lose predictive power or gain predictive power? |
| Things to do |
| R2 analysis one construct at a time |
| Find the most significant constructs (look at raw scores and the wt) |
| Overall z score for assertiveness - Dependent variable |
| Independent Variables - Items for assertiveness, Raw score assertiveness, Weight of assertiveness Raw Scores analysis. ALL THIS DONE INDIVIDUALLY. |
| Sum Non linear - sum value and sum value squared Weighted Non linear - weighted value and weighted value squared Classification - Item Classification - Sum Classification - Sum^2 Classification - Wt Classification - Wt^2 |

|  |
| --- |
| Part 2 : Improvement Weighting approach |
| 4 equal cuts and find corresponding y-hat at 0.2,0.4,.6 and 0.8 x value |

**Spring break work:**

|  |  |
| --- | --- |
| Spring Break work | Status |
| Recode values for classification model  Compute new zscores  Run a few classification models. (Anurag and I to split) | Complete |
| Non linearity check | Nothing worthwhile found.  Classification approach appears to be a better idea. |

**21/3/2017**

**Action items:**

**Prepare report for good understanding about deliverables, timeline and plan of action - Complete**

**24/3/2017**

|  |  |
| --- | --- |
| **Minutes** | **Action item** |
| Showed Keith the regression outputs and  classification model output |  |
| Sent him the files for his reference and further study |  |
| Discussed about the New Zoverall generation |  |
| Sent him the new Z file. Keith to confirm if the method of calculation was correct. |  |
| Sent Prof. Jason a brief documentation about the project. | Have been asked to build upon  previous batch's classification models for Part 1 |

Post this date, we only showed results of our models and fine-tuned our results.

## Code for classification models

### Tasks carried out by me: Data cleaning, RPART Classification models, RPART for Predictability problem statement.

**#CLEANING AND RPART for PREDICTABILITY**

hr\_data <- read.csv('C:/Users/USER/Documents/MS Stuff/RPI/Spring 2017/Business Capstone/FinalData.csv')

nrow(hr\_data)

#Eliminating mssing values

hr\_data\_complete <- hr\_data[complete.cases(hr\_data), ]

nrow(hr\_data\_complete)

head(hr\_data\_complete)

summary(hr\_data\_complete)

#Recoding 0 or 1

hr\_data\_complete$NewClass01HR\_ASSERT<-recode(hr\_data\_complete$STD4\_HR\_ASSERT,"15:24='0';25:34='1';else='0'")

hr\_data\_complete$NewClass01HR\_ACCOM<-recode(hr\_data\_complete$STD4\_HR\_ACCOM,"23:39='0';40:54='1';else='0'")

hr\_data\_complete$NewClass01HR\_GA<-recode(hr\_data\_complete$STD4\_HR\_GA,"26:34='0';else='1'")

hr\_data\_complete$NewClass01INTEG10\_VAL<-recode(hr\_data\_complete$INTEG10\_VAL\_STD4\_HR,"26:29='0';else='1'")

hr\_data\_complete$NewClass01HR\_SR<-recode(hr\_data\_complete$STD4\_HR\_SR,"17:28='0';else='1'")

hr\_data\_complete$NewClass01HR\_ORG<-recode(hr\_data\_complete$STD4\_HR\_ORG,"24:36='0';else='1'")

hr\_data\_complete$NewClass01HR\_POSATT<-recode(hr\_data\_complete$STD4\_HR\_POSATT,"18:38='0';else='1'")

hr\_data\_complete$NewClass01HR\_SOC<-recode(hr\_data\_complete$STD4\_HR\_SOC,"18:32='0';else='1'")

head(hr\_data\_complete)

write.csv(hr\_data\_complete,file="C:/Users/USER/Documents/MS Stuff/RPI/Spring 2017/Business Capstone/NewClass0or1.csv")

#Recoding 0,1,2

hr\_data\_complete1 <- hr\_data[complete.cases(hr\_data), ]

nrow(hr\_data\_complete1)

hr\_data\_complete1$NewClassHR\_ASSERT0123<-recode(hr\_data\_complete1$STD4\_HR\_ASSERT,"15:24='0';25:26='2';27:34='3';else='1'")

hr\_data\_complete1$NewClassHR\_ACCOM0123<-recode(hr\_data\_complete1$STD4\_HR\_ACCOM,"23:39='0';40:43='1';44:46='2';47:54='3';else='0'")

hr\_data\_complete1$NewClassHR\_GA0123<-recode(hr\_data\_complete1$STD4\_HR\_GA,"26:34='0';35:38='1';39:42='2';else='3'")

hr\_data\_complete1$NewClassINTEG10\_VAL012<-recode(hr\_data\_complete1$INTEG10\_VAL\_STD4\_HR,"26:29='0';30:35='1';else='2'")

hr\_data\_complete1$NewClassHR\_SR0123<-recode(hr\_data\_complete1$STD4\_HR\_SR,"17:28='0';26:28='1';29:40='3';else='2'")

hr\_data\_complete1$NewClassHR\_ORG0123<-recode(hr\_data\_complete1$STD4\_HR\_ORG,"24:36='0';30:36='1';37:42='2';else='3'")

hr\_data\_complete1$NewClassHR\_POSATT0123<-recode(hr\_data\_complete1$STD4\_HR\_POSATT,"18:30='0';31:38='1';39:43='2';else='3'")

hr\_data\_complete1$NewClassHR\_SOC0123<-recode(hr\_data\_complete1$STD4\_HR\_SOC,"18:32='0';33:38='1';39:45='3';else='2'")

head(hr\_data\_complete1)

write.csv(hr\_data\_complete1,file="C:/Users/USER/Documents/MS Stuff/RPI/Spring 2017/Business Capstone/NewClass0123.csv")

#Non linear regression

model1<-lm(hr\_data\_complete$ZOverallRating~poly(hr\_data\_complete$STD4\_HR\_ASSERT,3))

summary(model1)

plot(model1)

assert\_weight0<-hr\_data\_complete[which(hr\_data\_complete$w\_Assertiveness==3),]

head(assert\_weight0)

plot(assert\_weight0$STD4\_HR\_ASSERT\*assert\_weight0$STD4\_HR\_ASSERT\*assert\_weight0$STD4\_HR\_ASSERT)

var(hr\_data\_complete$STD4\_HR\_ASSERT)

plot(hr\_data\_complete$STD4\_HR\_ASSERT)

plot(hr\_data\_complete$STD4\_HR\_ACCOM)

plot(hr\_data\_complete$STD4\_HR\_SR)

plot(hr\_data\_complete$STD4\_HR\_SOC)

individual\_constructs<- read.csv('C:/Users/USER/Documents/MS Stuff/RPI/Spring 2017/Business Capstone/Capstone\_Constructs.csv')

nrow(individual\_constructs)

individual\_constructs<-individual\_constructs[complete.cases(individual\_constructs),]

model1\_assertiveness<-lm(individual\_constructs$ZOverallRating~individual\_constructs$AA1079\_4pt+individual\_constructs$AA1091\_4pt+individual\_constructs$AA1111\_4pt+individual\_constructs$AA2011\_4pt\_NEG+

individual\_constructs$AA2023\_4pt\_NEG+individual\_constructs$AA2039\_4pt\_NEG+individual\_constructs$AA2067\_4pt\_NEG+individual\_constructs$AA2087\_4pt\_NEG+individual\_constructs$AA2095\_4pt\_NEG)

summary(model1\_assertiveness)

model1\_accommodativeness<-lm(individual\_constructs$ZOverallRating~individual\_constructs$FF2126\_4pt\_NEG+individual\_constructs$FF2134\_4pt\_NEG+individual\_constructs$FF2138\_4pt\_NEG+individual\_constructs$FF2150\_4pt\_NEG+individual\_constructs$FF2154\_4pt\_NEG

+individual\_constructs$FF2158\_4pt\_NEG+individual\_constructs$FF2174\_4pt\_NEG+individual\_constructs$FF2182\_4pt\_NEG+individual\_constructs$FF2194\_4pt\_NEG+individual\_constructs$FF2206\_4pt\_NEG +

individual\_constructs$FF2214\_4pt\_NEG+individual\_constructs$FF2218\_4pt\_NEG+individual\_constructs$FF2222\_4pt\_NEG+individual\_constructs$FF2226\_4pt\_NEG+individual\_constructs$FF2234\_4pt\_NEG)

summary(model1\_accommodativeness)

model1\_workintensity<- lm(individual\_constructs$ZOverallRating~individual\_constructs$ENG104\_4pt+ individual\_constructs$ENG110\_4pt+ individual\_constructs$ENG111\_4pt+ individual\_constructs$ENG201\_4pt\_NEG+

individual\_constructs$ENG203\_4pt\_NEG+ individual\_constructs$ENG207\_4pt\_NEG+ individual\_constructs$ENG209\_4pt\_NEG+ individual\_constructs$GG1029\_4pt+

individual\_constructs$GG1037\_4pt+ individual\_constructs$GG1246\_4pt+ individual\_constructs$GG2025\_4pt\_NEG+ individual\_constructs$GG2049\_4pt\_NEG+individual\_constructs$GG2247\_4pt\_NEG)

summary(model1\_workintensity)

model1\_sociability<- lm(individual\_constructs$ZOverallRating~individual\_constructs$SS1005\_4pt+ individual\_constructs$SS1009\_4pt+ individual\_constructs$SS1021\_4pt+ individual\_constructs$SS1033\_4pt + individual\_constructs$SS1037\_4pt+ individual\_constructs$SS1093\_4pt+

individual\_constructs$SS2053\_4pt\_NEG+individual\_constructs$SS2057\_4pt\_NEG+ individual\_constructs$SS2069\_4pt\_NEG+ individual\_constructs$SS2073\_4pt\_NEG+ individual\_constructs$SS2089\_4pt\_NEG+ individual\_constructs$SS2105\_4pt\_NEG)

summary(model1\_sociability)

model1\_posatt<-lm(individual\_constructs$ZOverallRating~individual\_constructs$PP1203\_4pt+ individual\_constructs$PP2211\_4pt\_NEG+ individual\_constructs$PP2219\_4pt\_NEG+ individual\_constructs$PP2223\_4pt\_NEG+ individual\_constructs$PP2227\_4pt\_NEG+

individual\_constructs$PP2231\_4pt\_NEG+ individual\_constructs$PP2235\_4pt\_NEG+ individual\_constructs$SE16D36\_4pt\_NEG + individual\_constructs$SV01D05\_4pt\_NEG+ individual\_constructs$SV03E18\_4pt\_NEG + individual\_constructs$SV04D14\_4pt\_NEG+

individual\_constructs$SV05D33\_4pt\_NEG + individual\_constructs$WE01D23\_4pt\_NEG + individual\_constructs$WE04E15\_4pt\_NEG)

summary(model1\_posatt)

model1\_organization<-lm(individual\_constructs$ZOverallRating~individual\_constructs$OR1003\_4pt + individual\_constructs$OR1004\_4pt+ individual\_constructs$OR1005\_4pt+ individual\_constructs$OR1007\_4pt+

individual\_constructs$OR1008\_4pt+ individual\_constructs$OR1009\_4pt+ individual\_constructs$OR1010\_4pt+ individual\_constructs$OR1011\_4pt+

individual\_constructs$OR2001\_4pt\_NEG+ individual\_constructs$OR2002\_4pt\_NEG+ individual\_constructs$OR2006\_4pt\_NEG+ individual\_constructs$OR2012\_4pt\_NEG)

summary(model1\_organization)

model1\_selfreliance<-lm(individual\_constructs$ZOverallRating~ individual\_constructs$OO2098\_4pt\_NEG+individual\_constructs$SR2143\_4pt\_NEG+individual\_constructs$SR2147\_4pt\_NEG+ individual\_constructs$SR2175\_4pt\_NEG+

individual\_constructs$SR2183\_4pt\_NEG+ individual\_constructs$SR2187\_4pt\_NEG+ individual\_constructs$SR2195\_4pt\_NEG+individual\_constructs$SR2215\_4pt\_NEG+

individual\_constructs$SR2219\_4pt\_NEG+ individual\_constructs$SR2223\_4pt\_NEG+ individual\_constructs$SR2235\_4pt\_NEG)

summary(model1\_selfreliance)

individual\_constructs\_1<-read.csv('C:/Users/USER/Documents/MS Stuff/RPI/Spring 2017/Business Capstone/Capstone\_Constructs.csv')

individual\_constructs\_1[complete.cases(individual\_constructs\_1),]

nrow(individual\_constructs\_1)

na.omit(individual\_constructs\_1)

count=0

i=0

individual\_constructs\_1[3036,]

**#CLASSIFICATION MODELS**

#Reading file

hr\_data <- read.csv('C:/Users/USER/Documents/MS Stuff/RPI/Spring 2017/Business Capstone/FinalData\_605rows.csv')

nrow(hr\_data)

z\_score<-quantile(hr\_data$ZOverallRating,c(.2,.8))

z\_score

hr\_data$RecodedZOverall<-with(hr\_data, cut(hr\_data$ZOverallRating,breaks=quantile(hr\_data$ZOverallRating, probs=c(0,0.2,0.8,1),na.rm=TRUE),label = c("1","2","3"),include.lowest=TRUE))

head(hr\_data)

myvars<-c("INTEG10\_VAL\_STD4\_HR", "STD4\_HR\_ACCOM", "STD4\_HR\_ASSERT", "STD4\_HR\_SOC", "STD4\_HR\_GA", "STD4\_HR\_POSATT", "STD4\_HR\_ORG", "STD4\_HR\_SR","RecodedZOverall")

hr\_data\_complete<-hr\_data[myvars]

nrow(hr\_data\_complete)

head(hr\_data\_complete)

#Dividing into test and train

ind <- sample(2, nrow(hr\_data\_complete), replace=TRUE, prob=c(0.7, 0.3))

str(hr\_data\_complete)

hr\_data\_complete$RecodedZOverall<-factor(hr\_data\_complete$RecodedZOverall)

#Select our training data

train <- hr\_data\_complete[ind==1,]

test<-hr\_data\_complete[ind==2,]

test\_q<-hr\_data\_complete[ind==2,1:8]

str(train)

head(train)

head(test)

head(test\_q)

nrow(train)

nrow(test)

nrow(test\_q)

#C&R tree function using Raw scores as independent variables

model1<- tree(RecodedZOverall~., data=train,method="class")

summary(model1)

plot(model1)

text(model1)

predicted<-predict(model1,test\_q,type="class")

predicted

pred <- as.numeric(predicted)

pred

quantile(pred,c(.20,0.60,.80))

u = union(pred, test$RecodedZOverall)

t = as.matrix(table(Predicted = factor(pred, u), Actual = factor(test$RecodedZOverall, u)))

t

n <- sum(t)

diag <- diag(t)

acc <- sum(diag)/n

cat(acc\*100,"%", " values got classified accurately")

cat("Bottom 20% accuracy is", t[3,3]/sum(t[,3])\*100, "%")

cat("Middle 60% accuracy is", t[1,1]/sum(t[,1])\*100, "%")

cat("Top 20% accuracy is", t[2,2]/sum(t[,2])\*100, "%")

#C&R tree function using weighted scores as independent variables

myvars\_1<-c("w\_Assertiveness", "w\_Accommodation", "w\_WorkIntensity", "w\_WorkEthic", "w\_WorkIndepend", "w\_ProcessFocused", "w\_PositiveViewPeople","w\_Sociability","RecodedZOverall")

hr\_data\_wt<-hr\_data[myvars\_1]

ind <- sample(2, nrow(hr\_data\_wt), replace=TRUE, prob=c(0.7, 0.3))

str(hr\_data\_wt)

train <- hr\_data\_wt[ind==1,]

test<-hr\_data\_wt[ind==2,]

test\_q<-hr\_data\_wt[ind==2,1:8]

model2<- tree(RecodedZOverall~., data=train,method="class")

summary(model2)

plot(model2)

text(model2)

predicted<-predict(model2,test\_q,type="class")

predicted

str(predicted)

pred <- as.numeric(predicted)

quantile(pred,c(.20,0.60,.80))

u = union(pred, test$RecodedZOverall)

t = as.matrix(table(Predicted = factor(pred, u), Actual = factor(test$RecodedZOverall, u)))

t

n <- sum(t)

diag <- diag(t)

acc <- sum(diag)/n

cat(acc\*100,"%", " values got classified accurately")

cat("Bottom 20% accuracy is", t[3,3]/sum(t[,3])\*100, "%")

cat("Middle 60% accuracy is", t[1,1]/sum(t[,1])\*100, "%")

cat("Top 20% accuracy is", t[2,2]/sum(t[,2])\*100, "%")

#AUC – Area under curve

library(pROC)

auc\_CnRwt<-multiclass.roc(as.numeric(test$RecodedZOverall),pred)

plot(auc\_CnRwt, col="black", lwd=2,type="l", main=paste('C&R AUC:',round(auc\_CnRwt$auc[[1]],2)))

auc\_CnRwt

## References:

1. Outmatch presentation: RPI Overview: <https://drive.google.com/drive/folders/0B8yD5baAuYKXcmdpSmY5RVltQjQ>
2. Previous project code snippets: <https://drive.google.com/drive/folders/0B8yD5baAuYKXaEl6alFmR2cwbDQ>
3. Ethnicity and demographic data in machine learning:

Google Summary and Web App:

<https://research.google.com/bigpicture/attacking-discrimination-in-ml/>

Paper:

<https://drive.google.com/file/d/0B-wQVEjH9yuhanpyQjUwQS1JOTQ/view>