DAT 690- Capstone in Data Analytics

Final Capstone Project- General Electric Employee Attrition & Salary

Southern New Hampshire University

Timothy Harrison-Reyes

August 7, 2022

Contents

Introduction:

CRISP-DM Business Understanding Phase

Business Understanding

The General Electric Company, commonly known as GE, is a multinational organization from the United States that has been in business for over 125 years (FAQ- General Electric, 2022). GE has requested that the SNHU Data Analytics Team help their Human Resources (HR) department improve their employee retention rate. HR has spoken with mid-level managers and has identified that the company is losing capital and talent because of an increase in job postings across department, mainly regarding high potential employees. HR is requesting that the SNHU Data Analytics Team utilizes predictive analytic strategies based on the extensive data set they have provided to create a model to identify current employees that are at high risk of leaving the company. HR needs this information so they may work to retain them using various strategies based on the variables found from the analysis. Once the initial model is implemented and has shown success, HR is requesting that the Team predict the salary range needed to retain the employee.

There are many benefits to utilizing predictive analytics. One benefit is that GE already employs data collection methodologies across the entire business thus data collection would not be costly to the company and will increase the speed of the analysis. Another benefit is lowering training cost. For each employee that is retained, the hours and monetary value of the amount of training given to that employee will not have to be repeated regularly with new employees, which will ensure training costs will go down. One of the most beneficial outcomes of a successful predictive analysis model is the retention of quality staff. Quality staffing will improve business outcomes over time and help GE retain its title as a successful company. If

quality staffing becomes significantly affected by employee attrition rates, the company could be losing significant amount of capital.

Data Understanding

GE has given the team a robust data set, which will need to be examined for quality issues. Once data cleaning is conducted and the team feels that the data set is complete for its needs, the team will dig deeper into the raw data to ensure the important variables have been identified and pursue other avenues of analysis if needed (DAT650 Employee Attrition). The data will also be used to identify the relationships between the variables. The variables included in the data set are gender, income level, address, salary, education level and job role, annual income needed, and difference from salary. These are some of the most important variables, but not all of them since the data set includes 25 variables and 1470 entries (DAT650 Employee Attrition).

Data Preparation

To prep and analyze the data set, the Team will utilize R-Studio and the Rattle library. Rattle will allow the team to quickly identify variable focus and clean the data appropriately. The SNHU Data Analysis team combed the data set for outliers, missing or null values, and other issues that may cause issues during analysis (DAT650-Employee Attrition). The team removed gender, employee ID, and race to ensure that the model does not have any implicit bias and employee ID does not have any impact on the analysis and is unneeded.

Modeling

Once the data is prepared and meets the team's needs and integrity standards based on the CRISP-DM criteria, models will be created to solve GE's employee attrition issue. The team has decided that two models would best fit the needed solution: a random forest model and decision tree model. Decision tree model is best used to find the most important variables that have the

most impact on the current employee attrition rate. The random forest will then take those variables and determine the accuracy of each variable.

Evaluation

Once the variables are known that have the most impact, the results will be evaluated using the Area Under Curve (AUC) analysis and an error matrix. Depending on the results of both evaluation methods, the team will either re-evaluate current models and fix or scrap them as deemed appropriate or the team will deploy them.

Deployment

When the models are deemed highly accurate and appropriate solutions to General Electric's problem, the team will work with stakeholders and the Human Resources department to deploy across departments that have the highest attrition rates. This is a phased in deployment to help with model monitoring and continuous development as it is implemented in more and more departments. The SNHU Data Analytics Team will constantly monitor and assess the model to ensure it is working as intended. The data collected by HR once the model has been implemented on its level of success will allow the team to make necessary changes as needed down the road.

Plan Definition

CRISP-DM Data Understanding Phase*:*

Data Selection

General Electric's Human Resources department collected data for the project that included a total of 24 variables and 1,269 observations. It is the exact same as our previous analysis for attrition, but with extra variables added to help answer the research question. Each column is a variable, and each row is an employee. The variables included in the data set are gender, income

level, address, salary, education level and job role, annual income needed, and difference from salary. The added variables are what the employees think they should be paid, what they are actually paid, and the difference between the two values. These three new variables are numeric in nature.

The target variable will be ‘AnnualIncomeNeeded’ because it identifies the range of annual salary needed for said employee to remain with the company. This is the answer needed to General Electric’s and the SNHU Data Analytics Team’s research question. The team has also identified two other variables in relation to the ‘AnnualIncomeNeeded’ variable that may have significant impact on the analysis: ‘CurrentSalary’ and ‘DiffFromSalary’,, The team will ensure that these variables are closely monitored for impact on analysis. Other variables that will be part of the analysis will be education, environment satisfaction, job satisfaction, performance rating, and work life balance. These variables will be part of the analysis because they will give the team greater insight into the employees who believe they are underpaid based on levels of satisfaction and what is leading to said satisfaction.

The variables that have been removed are employee ID, percent of salary increase, training time, stock options, job level, and relationship satisfaction. Employee ID and relationship satisfaction have no relevance to the current question the team is trying to answer. GE did not give the proper description for non-GE employees to understand the description tied to stock options, job level and training time therefore it cannot be used with reliability. Identifying variables such as age, sex, and race were removed to ensure the removal of any possible bias.

GE dataset does raise some ethical concerns that an employer is asking and collecting data on employee relationship satisfaction. It is an irrelevant data point as noted earlier, and it’s an outlying variable that does not match the rest of the data set. It is also a personal piece of

information that has no impact on GE. It also makes employees feel pressured to give up this information because it is treated as relevant to their job, which is a major concern.

Statistical Analysis:

There were several observations once the descriptive analysis on the data occurred. First, when the team ran a summary of the data, the team noticed that the mean, median, and mode of each variable was wide ranging, which indicates a large enough range of employees to ensure the accuracy of the model. For example, the min of 'AnnualIncomeNeeded' is $62,400 with the max of $208,000 and a mean of $136,593 (Figure 1). These numbers indicate the large set of employees and the large amount of range in the annual salaries. As noted in figure 2, each variable adds to the skewness towards the right.

Figure 1

```
AnnualIncomeNeeded DistanceFromHome EnvironmentSatisfaction JobInvolvement
Min.   : 62400     Min.   :  8.00   Min.   :1.000           Min.   :1.000
1st Qu.:104000     1st Qu.: 36.00   1st Qu.:2.000           1st Qu.:2.000
Median :137280     Median : 65.00   Median :3.000           Median :3.000
Mean   :137990     Mean   : 64.81   Mean   :2.689           Mean   :2.715
3rd Qu.:174720     3rd Qu.: 93.00   3rd Qu.:4.000           3rd Qu.:3.000
Max.   :208000     Max.   :122.00   Max.   :4.000           Max.   :4.000
JobSatisfaction NumCompaniesWorked PercentSalaryHike PerformanceRating
Min.   :1.000   Min.   :0.00       Min.   :0.1100    Min.   :3.000
1st Qu.:2.000   1st Qu.:1.00       1st Qu.:0.1200    1st Qu.:3.000
Median :3.000   Median :2.00       Median :0.1400    Median :3.000
Mean   :2.707   Mean   :2.73       Mean   :0.1521    Mean   :3.153
3rd Qu.:4.000   3rd Qu.:4.00       3rd Qu.:0.1800    3rd Qu.:3.000
Max.   :4.000   Max.   :9.00       Max.   :0.2500    Max.   :4.000
TotalWorkingYears WorkLifeBalance YearsInCurrentRole YearsSinceLastPromotion
Min.   : 0.0      Min.   :1.000   Min.   : 0.000     Min.   : 0.000
1st Qu.: 6.0      1st Qu.:2.000   1st Qu.: 2.000     1st Qu.: 0.000
Median :10.0      Median :3.000   Median : 3.000     Median : 1.000
Mean   :11.3      Mean   :2.761   Mean   : 4.145     Mean   : 2.199
3rd Qu.:15.0      3rd Qu.:3.000   3rd Qu.: 7.000     3rd Qu.: 3.000
Max.   :40.0      Max.   :4.000   Max.   :18.000     Max.   :15.000
DiffFromSalary  CurrentSalary      Education
Min.   : 8506   Min.   : 45165   Min.   :1.000
1st Qu.:18173   1st Qu.: 84415   1st Qu.:2.000
Median :24799   Median :111730   Median :3.000
Mean   :25672   Mean   :112318   Mean   :2.905
3rd Qu.:32302   3rd Qu.:141244   3rd Qu.:4.000
Max.   :57408   Max.   :181418   Max.   :5.000
```

Figure 2

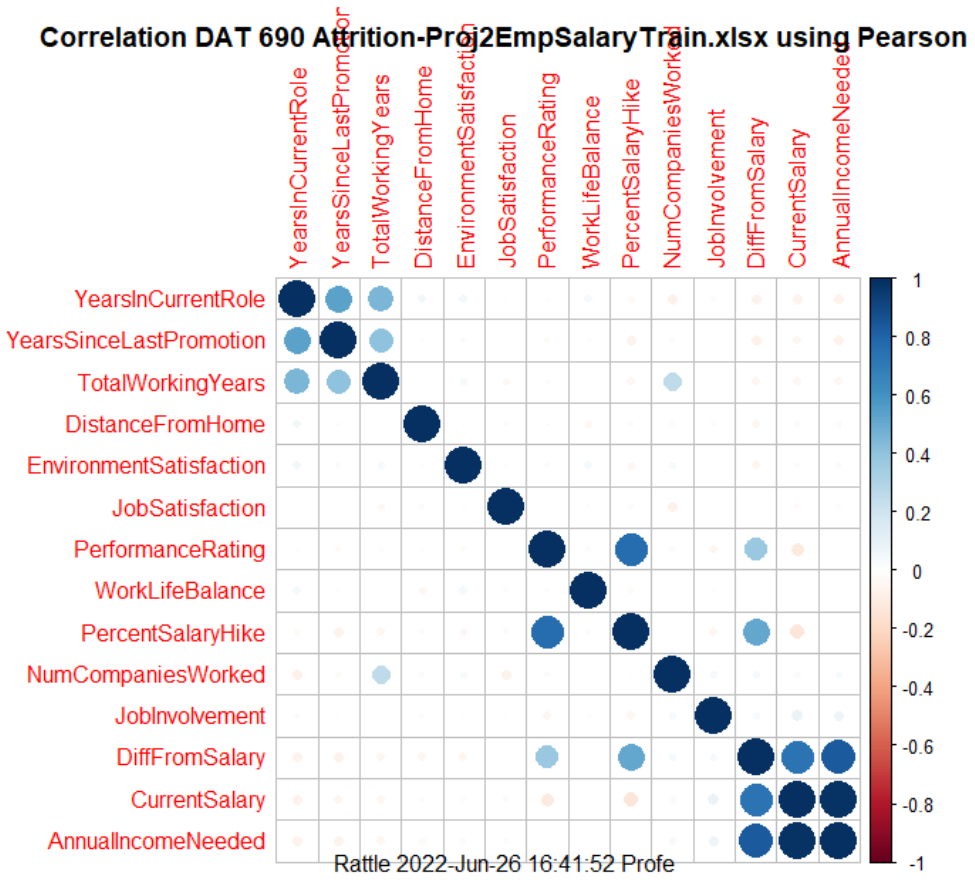
```
Skewness for each numeric variable of the dataset.
Positive means the right tail is longer.

     AnnualIncomeNeeded        DistanceFromHome EnvironmentSatisfaction         JobInvolvement         JobSatisfaction
            -0.05998882              0.01302129            -0.28537617            -0.45937442            -0.31089021
     NumCompaniesWorked       PercentSalaryHike       PerformanceRating      TotalWorkingYears      WorkLifeBalance
             1.00208110              0.78734220             1.92295145             1.07379594            -0.52654752
     YearsInCurrentRole YearsSinceLastPromotion         DiffFromSalary          CurrentSalary              Education
             0.94129180              2.00619574             0.50804980            -0.03292452            -0.31860280

Rattle timestamp: 2022-06-26 16:50:37 Profe
=====================================================================
```

Correlation Analysis

Figure 3



As shown in Figure 3, there are strong correlations between the variables the team initially hypothesized. The positive correlation is shown between years since last promotion,

work life balance, total working years, environment satisfaction and years at company. The variables that showed negative or no correlation were job involvement, number of companies worked, education, and job involvement. For the creation of our model, the team will remove these variables to ensure data viability and validity. The team knew that difference from salary, current salary and annual income needed would not show strong correlation because these are the target variables.

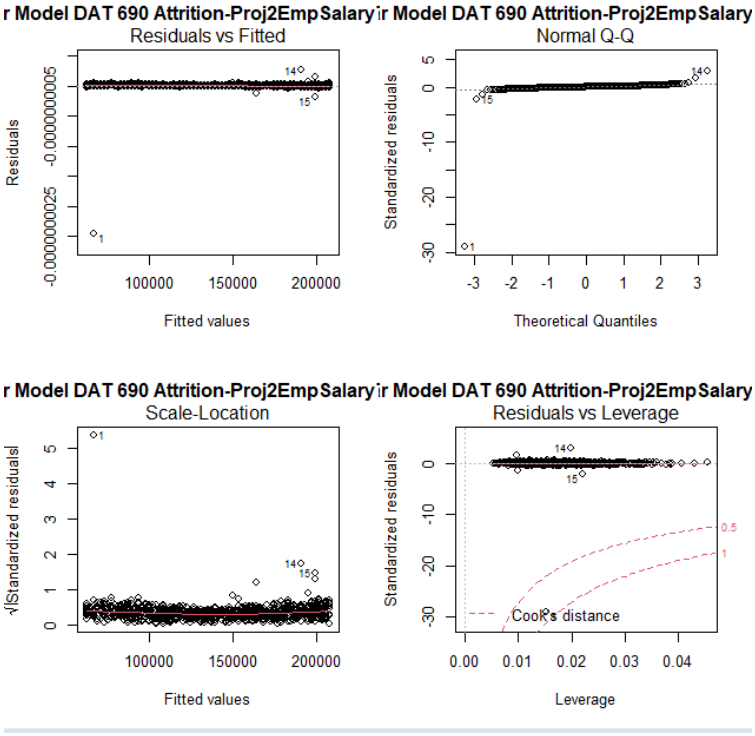
CRISP-DM Data Preparation Phase:

First the variables the team identified earlier to exclude were manually removed using Microsoft Excel. Next the team cleaned the data (i.e., removing duplicates, correcting misspellings, removing null values, and correcting formatting errors, etc.) by using the search and sort functions within Excel. GE did an outstanding job with collecting and maintaining this data set. It required minimal changes and cleaning. Once saved as a CSV file, the data set was uploaded to the Rattle library for analysis. Upon cursory glance, all variables were successfully assigned a numerical designation and listed as the correct variable type.

CRISP-DM Modeling Phase:

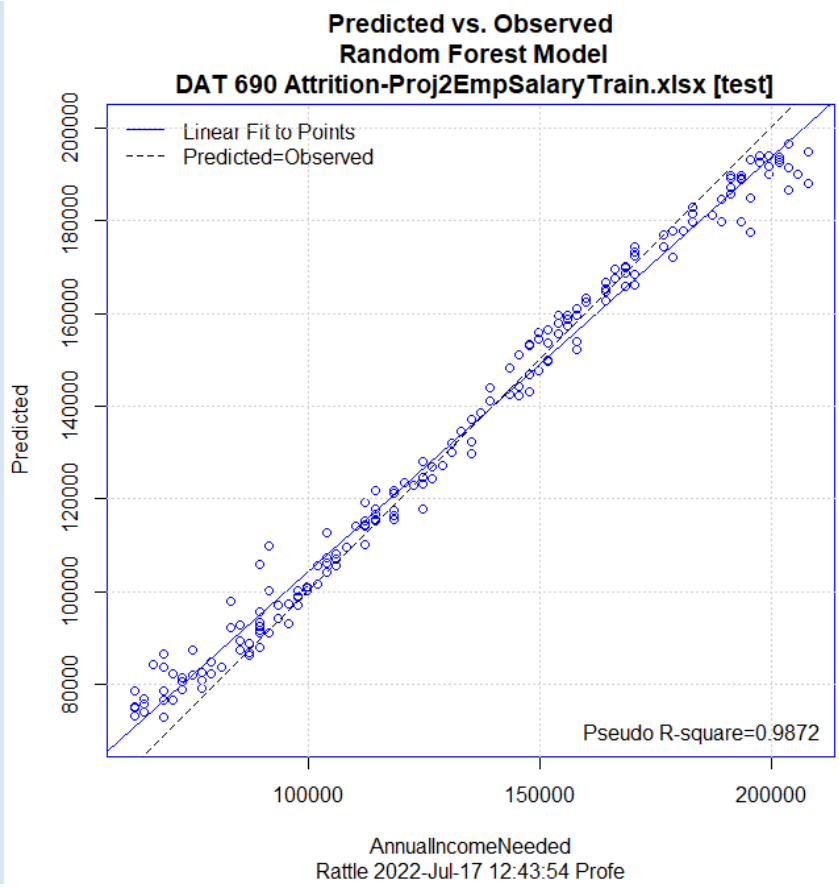
Preliminary Results

The team's preliminary results indicate that the current model is highly accurate. Using a Linear Aggression analysis, the data points are clustering close enough to the linear regression line as shown below:



Another analysis noted as the Area Under the Curve, commonly referred to as the AUC, that our model shows a score of 0.97. This score indicates that 97 % of the data points fit under the curve and is considerably high.

Another evaluation measure of the model is called a Predicted vs Observed plot. This evaluates the model's output (observed) against the predicted outcomes. As shown below, the overall error rate is considerably low and becomes significantly low as the forest is made.

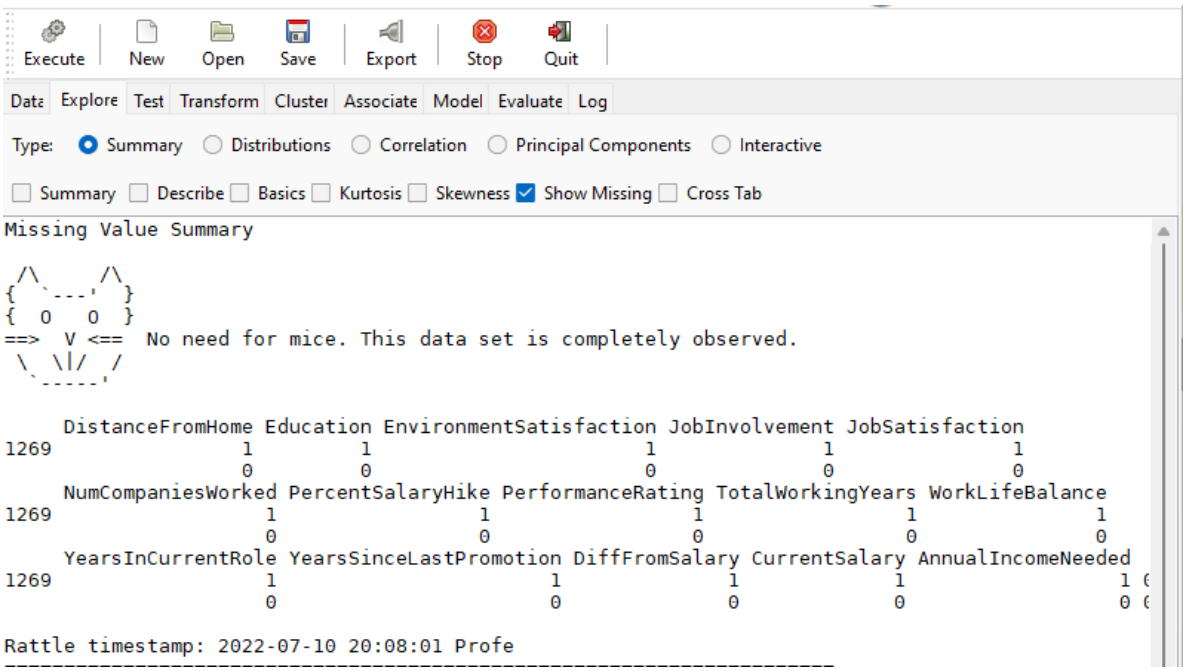


All three evaluation methods show a significant level of accuracy of the model's output, which means that the team is on track with the model.

Data Quality and Structure

The current output of the three evaluation methods shows that the team's initial analytical plan and methodology was solid. The data' s structure remains intact and according to the analytical plan. The team utilized the 70/15/15 partition method which means that the 70% of the data was partitioned into a training test and the other 30% was separated to create the validation and testing sets.

Data quality was evaluated using Rattle's Explore function for missing information as seen below:



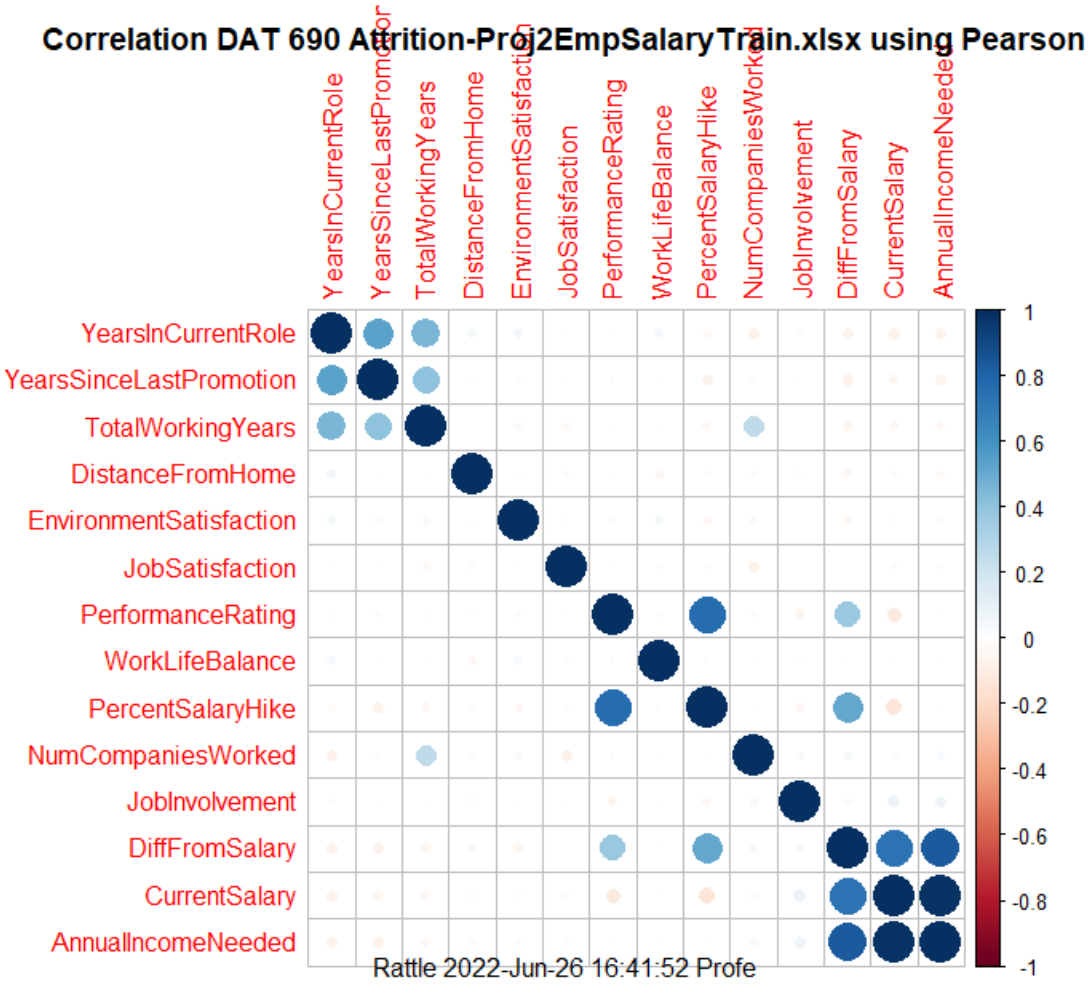
CRISP-DM Evaluation Phase:

Evaluation of Model

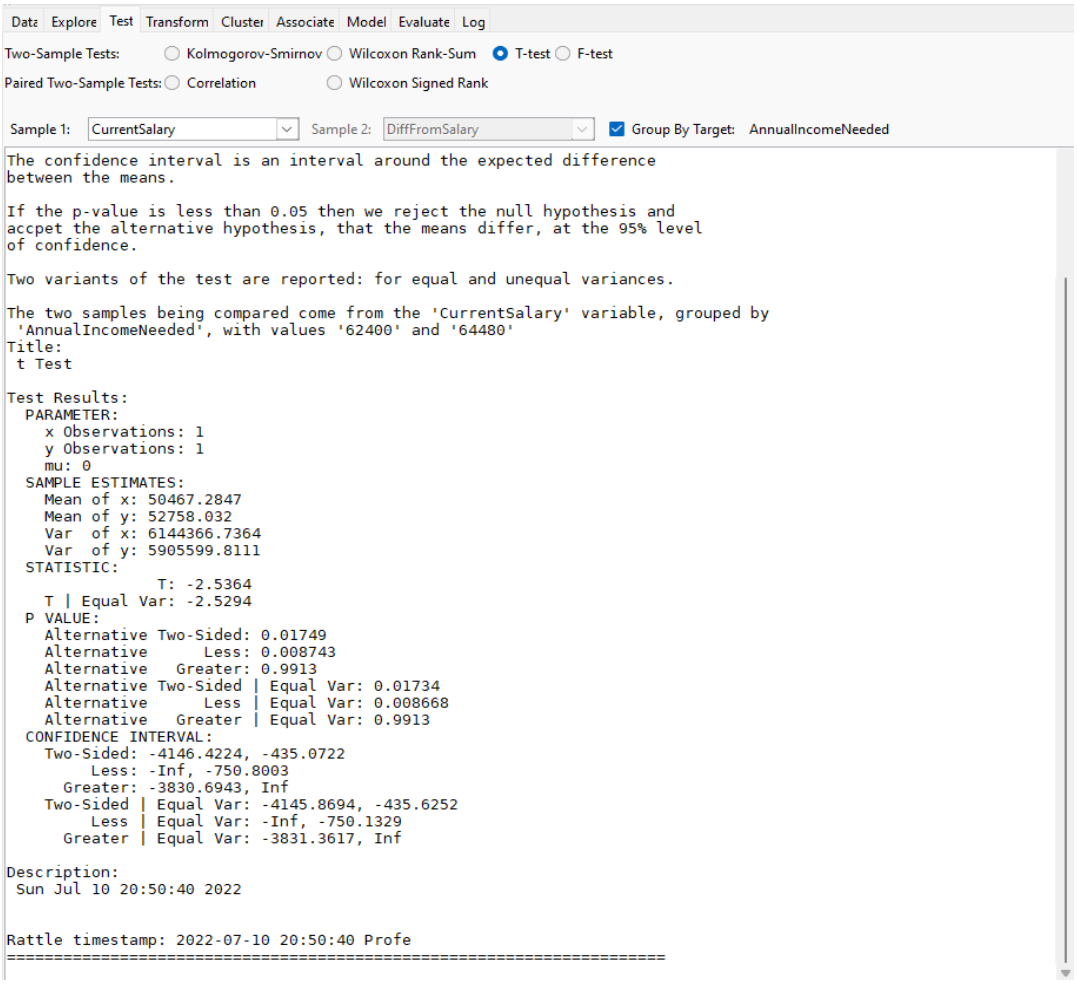
As stated earlier the model is producing a high level of accuracy, which means it is working as originally intended and is meeting the end goal of predictions needed. As this is the case, the model will not need to be overhauled.

Area(s) of Concern

The current area of concern is how to increase the model's accuracy by adding more possible variables that were already excluded. The team hypotheses that it would most likely be one or two more variables at most. As stated earlier the correlation graph using Pearson showed strong correlations between DiffFromSalary, CurrentSalary, and AnnualIncomeNeeded as shown below:


Correlation DAT 690 Attrition-Proj2EmpSalaryTrain.xlsx using Pearson
YearsInCurrentRole
YearsSinceLastPromotion
TotalWorkingYears
DistanceFromHome
EnvironmentSatisfaction
JobSatisfaction
PerformanceRating
WorkLifeBalance
PercentSalaryHike
NumCompaniesWorked
JobInvolvement
DiffFromSalary
CurrentSalary
AnnualIncomeNeeded
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
Rattle 2022-Jun-26 16:41:52 Profe

When a T-test is conducted (see below) it continues to show a strong correlation.



Overall, the model is working as intended with a high level of accuracy. This is great news for the SNHU Data Analytics team, and it could possibly be ready for full implementation following some experimentation with subtle variable additions.

Plan Implementation and Results

Deployment

The SNHU Data Analytics Team has concluded that the Random Forrest model should be used with the current data set from General Electric and is ready for full deployment. The Team has written a Production Turnover Report (See Appendix A) to be given to the head of the HR

department. The Production Turnover Report has the model's baseline performance and evaluation methodology. In order to Deploy the model regularly, the Team has also attached the R Code with full documentation to run the model regularly (See Appendix B). If HR does not have an individual or team training to run the model, follow up training will occur to ensure that the model is implemented and ran correctly.

Monitoring and Maintenance

The data set should continue to be maintained as new employees enter the workforce and other employees leave. The Team has recommendations on how to improve that data set over time (see Recommendations section below). Due to irregularity of employees leaving and other economic factors, the model should be monitored and evaluated at least once per fiscal quarter. This will keep the model updated and ensure that the model maintains its high level of accuracy. The code needed to implement the evaluation process is also part of the R Code (see Appendix B). As General Electric tweaks their data set based on the recommendations from the Team, evaluation and maintenance will need to occur. This will be completed as the recommended information is collected and stored in sufficient quantities. An end of the year report should also be completed to show how the model is doing as information changes as this will give new insights to achieving the business goal. If, at any point, the model's accuracy is less than 93%, the model should be re-evaluated and possibly changed to meet the goal.

Reporting Final Results

The SNHU Team will release this report to all stakeholders and stored as part of the project's full documentation procedures. A Production Turnover Report has been finalized (see Appendix A) and will be given to stakeholders during the final presentation and via email as a succinct version of the final report document. The R Code file will only be shared with the team or individual identified by General Electric who will be monitoring and maintaining the model

regularly. A final presentation will be made using Microsoft PowerPoint because it is a business standard and is used across many different fields. The presentation should have sufficient visuals that make the audience understand without being bogged down in mathematical jargon. The visuals may need to be created using Tableau or PowerBI just to create the appropriate visualizations of the data outside of what Rattle can do. Stakeholders will be appraised of the level of return on their investment, how the model works, what variables the model focuses on, and how accurate the model is, and the Team's recommendations regarding data set changes, monitoring and maintenance protocols.

Review Project

As part of the finalization of the project and its deployment, the SNHU Data Analytics Team will create Experience Documentation. Experience documentation will be used to document difficulties the Team as encountered, wrong approaches, and recommendations for picking the best techniques in similar situations in the future ("Deployment - Step 6 of the CRISP-Data Mining (DM) Process", 2022). This part of the project will remain internal as it is for the Team to improve, if General Electric wishes to have a copy, this can be submitted with all finalized paperwork as well.

Conclusions and Implications

The above analysis as conducted for General Electric to address employee attrition and salary needed to ensure they retain the employee. Initially the data set was clean using Excel and removing variables that were found to have little to no relevance to the business goal. It was also cleaned of all data that could create model bias. The Team decided that a Random Forrest Model and Decision Tree Model should first be conducted. The initial implementation and evaluation of both models showed high results, when the model was validated using the validated data set, a

linear aggression model was checked for viability. In the end, the Random Forrest model had the highest level of accuracy of all models.

Recommendations

General Electric's data set is very extensive but lacks enough data to ensure longevity of the current model and raises an ethical concern regarding collection of relationship satisfaction. It is the Team's recommendation that General Electric's Human Resources' department conduct exit surveys when any employee leaves. The exit survey should include updated job satisfaction levels and labeled as 'ExitJobSatisfaction' so that job satisfaction levels can be tracked to understand differences between initially collected and exiting the company. Also, qualitative data should be collected upon exit to understand why the employee leaving. The qualitative data should be analyzed to ensure that the current model's variables are what is contributing the employee attrition and ensure salary recommendations are accurate. When new employees are hired, initial data collection should occur to track differences over time. It is the Team's recommendation to utilize the Random Forrest model at this time. The Random Forrest model has a high accuracy rate of 97%.

References:

DAT650- Employee Attrition Use Case. DAT650 Southern New Hampshire University. Retrieved from: https://learn.snhu.edu/d2l/le/content/952001/Home.

Deployment - Step 6 of the CRISP-Data Mining (DM) Process. (2022). Retrieved 7 August 2022, from https://www.proglobalbusinesssolutions.com/deployment/

FAQ - General Electric. (2022). Retrieved 3 April 2022, from https://www.ge.com/faq

Appendix A: Production Turnover Report

Author: Timothy Harrison-Reyes

Date: July 26, 2022

Business Department: General Electric Human Resources Department

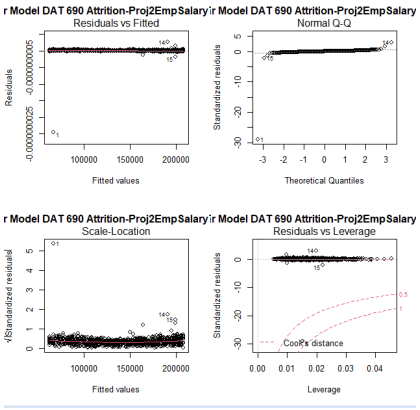
Project Name: General Electric Employee Retention Model

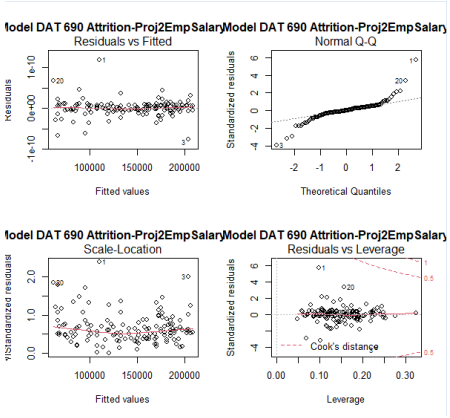
Project Description:

This model is designed to use General Electric's Human Resource Department's current and future data sets to help identify the salary needed to retain employees. Human Resources has spoken with mid-level managers and has identified that the company is losing capital and talent because of an increase in job postings across departments, mainly regarding high potential employees. This model will identify the salary range needed to retain employees that are highly likely to leave General Electric.

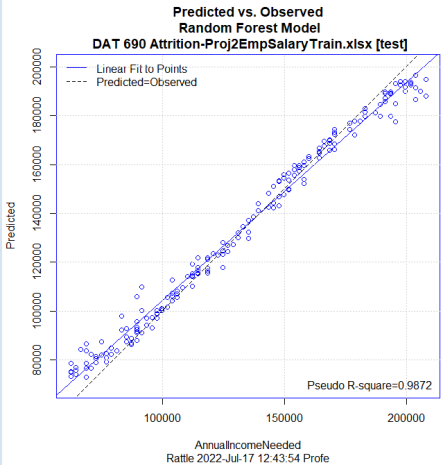
Model Baselines:

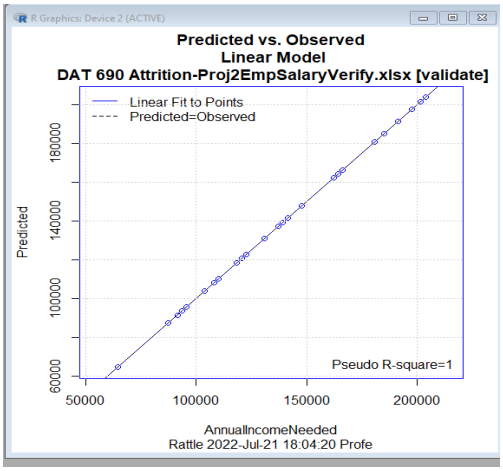
The team's preliminary results using training data set (left) and evaluation results (right) using the validation data set indicate that the current model is highly accurate. Using a Linear Aggression analysis, the data points are clustering close enough to the linear regression line as shown below:





Another analysis noted as the Area Under the Curve, commonly referred to as the AUC, that our model shows a score of 0.97. This score indicates that 97 % of the data points fit under the curve and is considerably high. The AUC for the validation data set is 98%, which is even higher. Another evaluation measure of the model is called a Predicted vs Observed plot. This evaluates the model's output (observed) against the predicted outcomes. As shown below, the overall error rate is considerably low and becomes significantly low as the forest is made.





All three evaluation methods show a significant level of accuracy of the model's output, which means that the team is on track with the model.

Model Performance:

As stated earlier, the model is showing a high level of accuracy using the training and verification data sets. The current model's variable focus is 'AnnualIncomeNeeded' because it identified the annual income needed to retain the identified employee within the company. Other variables that will be part of the analysis will be education, environment satisfaction, job satisfaction, performance rating, and work life balance. These variables are part of the analysis because they give the team greater insight into the employees who believe they are underpaid based on levels of satisfaction and what is leading to said satisfaction. GE dataset does raise some ethical concerns that an employer is asking and collecting data on employee relationship satisfaction. It is an irrelevant data point as noted earlier, and it's an outlying variable that does not match the rest of the data set. It is also a personal piece of information that has no impact on GE. It also makes employees feel pressured to give up this information because it is treated as relevant to their job, which is a major concern. It should be noted that an exit survey should be conducted and collected from any employee that is leaving to ensure that the model is constantly being refined to ensure high level of accuracy as the data set continues to grow.

Comments:

The stakeholders will use the outputs of the random forest results to identify those employees who need higher salaries to retain.

Appendix B: R Code

```
# Title: "General Electric Salary and Attrition Model"
# Class: DAT 690- Capstone in Data Analytics
# Author: Timothy Harrison-Reyes
# Date: July 28, 2022

================================================================================

## Statement of Problem:
# GE has requested that the SNHU Data Analytics Team help their Human Resources (HR) department improve their employee retention rate. HR has spoken with mid-level managers and has identified that the company is losing capital and talent because of an increase in job postings across department, mainly regarding high potential employees. HR is requesting that the SNHU Data Analytics Team utilizes predictive analytic strategies based on the extensive data set they have provided to create a model to identify current employees that are at high risk of leaving the company. HR needs this information so they may work to retain them using various strategies based on the variables found from the analysis. Once the initial model is implemented and has shown success, HR is requesting that the Team predict the salary range needed to retain the employee.
================================================================================
## Environment Setup & Packages
# OS: Windows 11 64 Bit
# R version 4.1.3 (2022-03-10) -- "One Push-Up"
# Rattle version 5.5.1

install.packages("https://access.togaware.com/RGtk2_2.20.36.2.zip", repos=NULL)
install.packages("https://access.togaware.com/cairoDevice_2.28.zip", repos=NULL)
install.packages("rattle")

Documentation for install: https://rattle.togaware.com

## Data Preparation
# file: "DAT 690 Attrition-Proj2EmpSalaryTrain"
# Using Excel, remove employeeID, trainingtime, stockoptions, joblevel, and relationshipsatisfaction.

##Initiate Rattle
packages("rattle")
rattle()

## Data Input - TrainingData
# Load TrainingData dataset from file: "DAT 690 Attrition-Proj2EmpSalaryTrain"

library(readxl, quietly=TRUE)
```

```
 crs$dataset <- read_excel("D:/Dropbox/Documents/College_Univserity/SNHU/DAT 690-
Advanced Data Analytics Capstone/DAT 690 Attrition-Proj2EmpSalaryTrain.xlsx",
guess_max=1e4)

 crs$dataset

# Build the train/validate/test datasets.

# nobs=1269 train=888 validate=190 test=191

set.seed(crv$seed)

crs$nobs <- nrow(crs$dataset)

crs$train <- sample(crs$nobs, 0.7*crs$nobs)

crs$nobs %>%
  seq_len() %>%
  setdiff(crs$train) %>%
  sample(0.15*crs$nobs) ->
crs$validate

crs$nobs %>%
  seq_len() %>%
  setdiff(crs$train) %>%
  setdiff(crs$validate) ->
crs$test

# The following variable selections have been noted.

crs$input     <- c("DistanceFromHome", "Education",
                   "EnvironmentSatisfaction", "JobInvolvement",
                   "JobSatisfaction", "NumCompaniesWorked",
                   "PercentSalaryHike", "PerformanceRating",
                   "TotalWorkingYears", "WorkLifeBalance",
                   "YearsInCurrentRole", "YearsSinceLastPromotion",
                   "DiffFromSalary", "CurrentSalary")

crs$numeric   <- c("DistanceFromHome", "Education",
                   "EnvironmentSatisfaction", "JobInvolvement",
                   "JobSatisfaction", "NumCompaniesWorked",
                   "PercentSalaryHike", "PerformanceRating",
                   "TotalWorkingYears", "WorkLifeBalance",
                   "YearsInCurrentRole", "YearsSinceLastPromotion",
                   "DiffFromSalary", "CurrentSalary")
```

```
crs$categoric <- NULL

crs$target    <- "AnnualIncomeNeeded"
crs$risk      <- NULL
crs$ident     <- NULL
crs$ignore    <- NULL
crs$weights   <- NULL

## Generate a correlation plot for the variables.

# The 'corrplot' package provides the 'corrplot' function.

library(corrplot, quietly=TRUE)

# Correlations work for numeric variables only.

crs$cor <- cor(crs$dataset[crs$train, crs$numeric], use="pairwise", method="pearson")

# Order the correlations by their strength.

crs$ord <- order(crs$cor[1,])
crs$cor <- crs$cor[crs$ord, crs$ord]

# Display the actual correlations.

print(crs$cor)

# Graphically display the correlations.

corrplot(crs$cor, mar=c(0,0,1,0))
title(main="Correlation DAT 690 Attrition-Proj2EmpSalaryTrain.xlsx using Pearson",
    sub=paste("Rattle", format(Sys.time(), "%Y-%b-%d %H:%M:%S"), Sys.info()["user"]))

================================================================================
========================================
# Build a Random Forest model using the traditional approach.

set.seed(crv$seed)

crs$rf <- randomForest::randomForest(AnnualIncomeNeeded ~ .,
  data=crs$dataset[crs$train, c(crs$input, crs$target)],
  ntree=500,
  mtry=3,
  importance=TRUE,
  na.action=randomForest::na.roughfix,
```

```
  replace=FALSE)

# Generate textual output of the 'Random Forest' model.

crs$rf

# List the importance of the variables.

rn <- crs$rf %>%
    randomForest::importance() %>%
    round(2)
    rn[order(rn[,1], decreasing=TRUE),]

============================================================================
========================================
# Decision Tree

# The 'rpart' package provides the 'rpart' function.

library(rpart, quietly=TRUE)

# Reset the random number seed to obtain the same results each time.

set.seed(crv$seed)

# Build the Decision Tree model.

crs$rpart <- rpart(AnnualIncomeNeeded ~ .,
    data=crs$dataset[crs$train, c(crs$input, crs$target)],
    method="anova",
    parms=list(split="information"),
    control=rpart.control(usesurrogate=0,
        maxsurrogate=0),
    model=TRUE)

# Generate a textual view of the Decision Tree model.

print(crs$rpart)
printcp(crs$rpart)
cat("\n")
============================================================================
========================================
# Regression model

# Build a Regression model.
```

```
crs$glm <- lm(AnnualIncomeNeeded ~ ., data=crs$dataset[crs$train,c(crs$input, crs$target)])

# Generate a textual view of the Linear model.

print(summary(crs$glm))
cat('==== ANOVA ====

')
print(anova(crs$glm))
print("
")
================================================================================
========================================
# Plot the model evaluation.

ttl <- genPlotTitleCmd("Linear Model",crs$dataname,vector=TRUE)
plot(crs$glm, main=ttl[1])

# Evaluate model performance on the training dataset.

# RPART: Generate a Predicted v Observed plot for rpart model on DAT 690 Attrition-
Proj2EmpSalaryTrain.xlsx [**train**].

crs$pr <- predict(crs$rpart, newdata=crs$dataset[crs$train, c(crs$input, crs$target)])

# Obtain the observed output for the dataset.

obs <- subset(crs$dataset[crs$train, c(crs$input, crs$target)], select=crs$target)

# Handle in case categoric target treated as numeric.

obs.rownames <- rownames(obs)
obs <- as.numeric(obs[[1]])
obs <- data.frame(AnnualIncomeNeeded=obs)
rownames(obs) <- obs.rownames

# Combine the observed values with the predicted.

fitpoints <- na.omit(cbind(obs, Predicted=crs$pr))

# Obtain the pseudo R2 - a correlation.

fitcorr <- format(cor(fitpoints[,1], fitpoints[,2])^2, digits=4)

# Plot settings for the true points and best fit.
```

```
op <- par(c(lty="solid", col="blue"))

# Display the observed (X) versus predicted (Y) points.

plot(fitpoints[[1]], fitpoints[[2]], asp=1, xlab="AnnualIncomeNeeded", ylab="Predicted")

# Generate a simple linear fit between predicted and observed.

prline <- lm(fitpoints[,2] ~ fitpoints[,1])

# Add the linear fit to the plot.

abline(prline)

# Add a diagonal representing perfect correlation.

par(c(lty="dashed", col="black"))
abline(0, 1)

# Include a pseudo R-square on the plot

legend("bottomright",  sprintf(" Pseudo R-square=%s ", fitcorr),  bty="n")

# Add a title and grid to the plot.

title(main="Predicted vs. Observed
 Decision Tree Model
 DAT 690 Attrition-Proj2EmpSalaryTrain.xlsx [**train**]",
    sub=paste("Rattle", format(Sys.time(), "%Y-%b-%d %H:%M:%S"), Sys.info()["user"]))
grid()

# GLM: Generate a Predicted v Observed plot for glm model on DAT 690 Attrition-
Proj2EmpSalaryTrain.xlsx [**train**].

crs$pr <- predict(crs$glm,
   type    = "response",
   newdata = crs$dataset[crs$train, c(crs$input, crs$target)])

# Obtain the observed output for the dataset.

obs <- subset(crs$dataset[crs$train, c(crs$input, crs$target)], select=crs$target)

# Handle in case categoric target treated as numeric.

obs.rownames <- rownames(obs)
obs <- as.numeric(obs[[1]])
```

```
obs <- data.frame(AnnualIncomeNeeded=obs)
rownames(obs) <- obs.rownames

# Combine the observed values with the predicted.

fitpoints <- na.omit(cbind(obs, Predicted=crs$pr))

# Obtain the pseudo R2 - a correlation.

fitcorr <- format(cor(fitpoints[,1], fitpoints[,2])^2, digits=4)

# Plot settings for the true points and best fit.

op <- par(c(lty="solid", col="blue"))

# Display the observed (X) versus predicted (Y) points.

plot(fitpoints[[1]], fitpoints[[2]], asp=1, xlab="AnnualIncomeNeeded", ylab="Predicted")

# Generate a simple linear fit between predicted and observed.

prline <- lm(fitpoints[,2] ~ fitpoints[,1])

# Add the linear fit to the plot.

abline(prline)

# Add a diagonal representing perfect correlation.

par(c(lty="dashed", col="black"))
abline(0, 1)

# Include a pseudo R-square on the plot

legend("bottomright",  sprintf(" Pseudo R-square=%s ", fitcorr),  bty="n")

# Add a title and grid to the plot.

title(main="Predicted vs. Observed
 Linear Model
 DAT 690 Attrition-Proj2EmpSalaryTrain.xlsx [**train**]",
    sub=paste("Rattle", format(Sys.time(), "%Y-%b-%d %H:%M:%S"), Sys.info()["user"]))
grid()

# Evaluate model performance on the training dataset.
```

```
# Risk Chart: requires the ggplot2 package.

library(ggplot2)

# Generate a risk chart.

# Rattle provides evaluateRisk() and riskchart().

crs$pr <- predict(crs$rf, newdata=na.omit(crs$dataset[crs$train, c(crs$input, crs$target)]))

crs$eval <- evaluateRisk(crs$pr, na.omit(crs$dataset[crs$train, c(crs$input,
crs$target)])$AnnualIncomeNeeded)
print(riskchart(crs$pr,
   na.omit(crs$dataset[crs$train, c(crs$input, crs$target)])$AnnualIncomeNeeded,
   title="Performance Chart Random Forest DAT 690 Attrition-Proj2EmpSalaryTrain.xlsx
[**train**] ", show.lift=FALSE, show.precision=FALSE, legend.horiz=FALSE))

=========================================================================
========================================

## Verification Data Set Model Evaluation
# Load verification dataset from file: DAT 690 Attrition-Proj2EmpSalaryVerify.xlsx.

library(readxl, quietly=TRUE)

 crs$dataset <- read_excel("D:/Dropbox/Documents/College_Univserity/SNHU/DAT 690-
Advanced Data Analytics Capstone/AttritionModelWorkspace_2022/DAT 690 Attrition-
Proj2EmpSalaryVerify.xlsx", guess_max=1e4)

 crs$dataset

#=========================================================================
===
# Action the user selections from the Data tab.

# Build the train/validate/test datasets.

# nobs=201 train=141 validate=30 test=30

set.seed(crv$seed)

crs$nobs <- nrow(crs$dataset)

crs$train <- sample(crs$nobs, 0.7*crs$nobs)

crs$nobs %>%
```

```
  seq_len() %>%
  setdiff(crs$train) %>%
  sample(0.15*crs$nobs) ->
crs$validate

crs$nobs %>%
  seq_len() %>%
  setdiff(crs$train) %>%
  setdiff(crs$validate) ->
crs$test

# The following variable selections have been noted.
# All new variables remained to check for variable correlation. This should be done whenever a new variable(s) is introduced to continuously update the model. The orginal variables that were left out in the training dataset were ignored in this evaluation.

crs$input     <- c("DistanceFromHome", "Education",
                   "EnvironmentSatisfaction", "JobInvolvement",
                   "JobLevel", "JobSatisfaction",
                   "NumCompaniesWorked", "AvgOverTime",
                   "PercentSalaryHike", "PerformanceRating",
                   "TotalWorkingYears", "WorkLifeBalance",
                   "YearsAtCompany", "YearsInCurrentRole",
                   "YearsSinceLastPromotion", "YearsWithCurrManager",
                   "DiffFromSalary", "CurrentSalary")

crs$numeric   <- c("DistanceFromHome", "Education",
                   "EnvironmentSatisfaction", "JobInvolvement",
                   "JobLevel", "JobSatisfaction",
                   "NumCompaniesWorked", "AvgOverTime",
                   "PercentSalaryHike", "PerformanceRating",
                   "TotalWorkingYears", "WorkLifeBalance",
                   "YearsAtCompany", "YearsInCurrentRole",
                   "YearsSinceLastPromotion", "YearsWithCurrManager",
                   "DiffFromSalary", "CurrentSalary")

crs$categoric <- NULL

crs$target    <- "AnnualIncomeNeeded"
crs$risk      <- NULL
crs$ident     <- NULL
crs$ignore    <- c("EMPID", "Age", "RelationshipSatisfaction", "StockOption",
"TrainingTimesLastYear")
crs$weights   <- NULL

# Generate a correlation plot for the variables.
```

```
# The 'corrplot' package provides the 'corrplot' function.

library(corrplot, quietly=TRUE)

# Correlations work for numeric variables only.

crs$cor <- cor(crs$dataset[crs$train, crs$numeric], use="pairwise", method="pearson")

# Order the correlations by their strength.

crs$ord <- order(crs$cor[1,])
crs$cor <- crs$cor[crs$ord, crs$ord]

# Display the actual correlations.

print(crs$cor)

# Graphically display the correlations.

corrplot(crs$cor, mar=c(0,0,1,0))
title(main="Correlation DAT 690 Attrition-Proj2EmpSalaryVerify.xlsx using Pearson",
    sub=paste("Rattle", format(Sys.time(), "%Y-%b-%d %H:%M:%S"), Sys.info()["user"]))

# Build a Random Forest model using the traditional approach.

set.seed(crv$seed)

crs$rf <- randomForest::randomForest(AnnualIncomeNeeded ~ .,
  data=crs$dataset[crs$train, c(crs$input, crs$target)],
  ntree=500,
  mtry=4,
  importance=TRUE,
  na.action=randomForest::na.roughfix,
  replace=FALSE)

# Generate textual output of the 'Random Forest' model.

crs$rf

# List the importance of the variables.

rn <- crs$rf %>%
    randomForest::importance() %>%
    round(2)
    rn[order(rn[,1], decreasing=TRUE),]
```

```
# Time taken: 0.09 secs

#=======================================================================
===
# Rattle timestamp: 2022-07-28 14:45:08 x86_64-w64-mingw32

# Evaluate model performance on the validation dataset.

# Risk Chart: requires the ggplot2 package.

library(ggplot2)

# Generate a risk chart.

# Rattle provides evaluateRisk() and riskchart().

crs$pr <- predict(crs$rf, newdata=na.omit(crs$dataset[crs$validate, c(crs$input, crs$target)]))

crs$eval <- evaluateRisk(crs$pr, na.omit(crs$dataset[crs$validate, c(crs$input,
crs$target)])$AnnualIncomeNeeded)
print(riskchart(crs$pr,
   na.omit(crs$dataset[crs$validate, c(crs$input, crs$target)])$AnnualIncomeNeeded,
   title="Performance Chart Random Forest DAT 690 Attrition-Proj2EmpSalaryVerify.xlsx
[validate] ", show.lift=FALSE, show.precision=FALSE, legend.horiz=FALSE))


#=======================================================================
===
# Rattle timestamp: 2022-07-28 14:45:11 x86_64-w64-mingw32

# Evaluate model performance on the validation dataset.

# RF: Generate a Predicted v Observed plot for rf model on DAT 690 Attrition-
Proj2EmpSalaryVerify.xlsx [validate].

crs$pr <- predict(crs$rf, newdata=na.omit(crs$dataset[crs$validate, c(crs$input, crs$target)]))

# Obtain the observed output for the dataset.

obs <- subset(na.omit(crs$dataset[crs$validate, c(crs$input, crs$target)]), select=crs$target)

# Handle in case categoric target treated as numeric.

obs.rownames <- rownames(obs)
obs <- as.numeric(obs[[1]])
```

```
obs <- data.frame(AnnualIncomeNeeded=obs)
rownames(obs) <- obs.rownames

# Combine the observed values with the predicted.

fitpoints <- na.omit(cbind(obs, Predicted=crs$pr))

# Obtain the pseudo R2 - a correlation.

fitcorr <- format(cor(fitpoints[,1], fitpoints[,2])^2, digits=4)

# Plot settings for the true points and best fit.

op <- par(c(lty="solid", col="blue"))

# Display the observed (X) versus predicted (Y) points.

plot(fitpoints[[1]], fitpoints[[2]], asp=1, xlab="AnnualIncomeNeeded", ylab="Predicted")

# Generate a simple linear fit between predicted and observed.

prline <- lm(fitpoints[,2] ~ fitpoints[,1])

# Add the linear fit to the plot.

abline(prline)

# Add a diagonal representing perfect correlation.

par(c(lty="dashed", col="black"))
abline(0, 1)

# Include a pseudo R-square on the plot

legend("bottomright",  sprintf(" Pseudo R-square=%s ", fitcorr),  bty="n")

# Add a title and grid to the plot.

title(main="Predicted vs. Observed
 Random Forest Model
 DAT 690 Attrition-Proj2EmpSalaryVerify.xlsx [validate]",
    sub=paste("Rattle", format(Sys.time(), "%Y-%b-%d %H:%M:%S"), Sys.info()["user"]))
grid()

#=====================================================================
===
```

```
# Rattle timestamp: 2022-07-28 14:45:35 x86_64-w64-mingw32

# Regression model

# Build a Regression model.

crs$glm <- lm(AnnualIncomeNeeded ~ ., data=crs$dataset[crs$train,c(crs$input, crs$target)])

# Generate a textual view of the Linear model.

print(summary(crs$glm))
cat('==== ANOVA ====

')
print(anova(crs$glm))
print("
")

# Time taken: 0.01 secs

# Plot the model evaluation.

ttl <- genPlotTitleCmd("Linear Model",crs$dataname,vector=TRUE)
plot(crs$glm, main=ttl[1])

#=====================================================================
===
# Rattle timestamp: 2022-07-28 14:45:44 x86_64-w64-mingw32

# Evaluate model performance on the validation dataset.

# Risk Chart: requires the ggplot2 package.

library(ggplot2)

# Generate a risk chart.

# Rattle provides evaluateRisk() and riskchart().

crs$pr <- predict(crs$glm,
  type    = "response",
  newdata = crs$dataset[crs$validate, c(crs$input, crs$target)])

crs$eval <- evaluateRisk(crs$pr, crs$dataset[crs$validate, c(crs$input,
crs$target)]$AnnualIncomeNeeded)
print(riskchart(crs$pr,
```

```
    crs$dataset[crs$validate, c(crs$input, crs$target)]$AnnualIncomeNeeded,
    title="Performance Chart Linear DAT 690 Attrition-Proj2EmpSalaryVerify.xlsx [validate] ",
show.lift=FALSE, show.precision=FALSE, legend.horiz=FALSE))


#=========================================================================
==

# Evaluate model performance on the validation dataset.

# GLM: Generate a Predicted v Observed plot for glm model on DAT 690 Attrition-
Proj2EmpSalaryVerify.xlsx [validate].

crs$pr <- predict(crs$glm,
   type    = "response",
   newdata = crs$dataset[crs$validate, c(crs$input, crs$target)])

# Obtain the observed output for the dataset.

obs <- subset(crs$dataset[crs$validate, c(crs$input, crs$target)], select=crs$target)

# Handle in case categoric target treated as numeric.

obs.rownames <- rownames(obs)
obs <- as.numeric(obs[[1]])
obs <- data.frame(AnnualIncomeNeeded=obs)
rownames(obs) <- obs.rownames

# Combine the observed values with the predicted.

fitpoints <- na.omit(cbind(obs, Predicted=crs$pr))

# Obtain the pseudo R2 - a correlation.

fitcorr <- format(cor(fitpoints[,1], fitpoints[,2])^2, digits=4)

# Plot settings for the true points and best fit.

op <- par(c(lty="solid", col="blue"))

# Display the observed (X) versus predicted (Y) points.

plot(fitpoints[[1]], fitpoints[[2]], asp=1, xlab="AnnualIncomeNeeded", ylab="Predicted")

# Generate a simple linear fit between predicted and observed.
```

```
prline <- lm(fitpoints[,2] ~ fitpoints[,1])

# Add the linear fit to the plot.

abline(prline)

# Add a diagonal representing perfect correlation.

par(c(lty="dashed", col="black"))
abline(0, 1)

# Include a pseudo R-square on the plot

legend("bottomright",  sprintf(" Pseudo R-square=%s ", fitcorr),  bty="n")

# Add a title and grid to the plot.

title(main="Predicted vs. Observed
 Linear Model
 DAT 690 Attrition-Proj2EmpSalaryVerify.xlsx [validate]",
    sub=paste("Rattle", format(Sys.time(), "%Y-%b-%d %H:%M:%S"), Sys.info()["user"]))
grid()
```