**R-Script Talking Points**

**HR Data Analysis**

**Libraries:**

***Ggplot2*** – Ggplot 2 is a R package that allows the user to create complex R data visuals. It is similar to built in R graphs, but allows much more customization through layers. I will use this to customize several graphics to visualize my distributions and explore my data to gain a better understanding

***Caret*** - It provides functions designed to simplify common tasks associated with predictive modeling. R has some of functionality built in, but using this package allows you to further analysis. Caret also has sample datasets such as the one I'm using called German Credit which demonstrates CARET's strengths.

***dplyr*** – dplyr is a package that lets you more easily work with tabular data. I like the glimpse() in this library that displays data in a cleaner easier to read manner. Also there is another useful function sample\_n() that allows you to grab random rows from a table or data frame

***PVclust*** – In order to conduct a cluster analysis I searched the web for clustering packages. Pvclust is interesting it allows me to store my cluster analysis as a variable and then plot it, which outputs a very clean looking tree of my clusters

***Ameila*** – Amelia is a cool package that lets you visualize how much of your data is missing via the missmap()

***MASS*** – This package has a particular useful function called confint()

***Dummies*** – This is a very useful package for modeling as it allows you to easily create dummy variables with a single function dummy(). I used this to convert some of my factors into dummy variables

***Catools*** – Catools library is a library that gives the user several functions to analyze a predictive model. In my script I have created a logistic regression to predict leave or stay of employees and this provided me with some tools to validated via AUC and ROC curves

***PVclust*** – In order to conduct a cluster analysis I searched the web for clustering packages. Pvclust is interesting it allows me to store my cluster analysis as a variable and then plot it, which outputs a very clean looking tree of my clusters

***randomForest*** – I wanted to predict employees leaving and staying with another method in addition to logistical regression

**Script Walkthrough:**

**Misc.**

After loading in the libraries, I plan to use in this analysis I set a random seed so I can reproduce the same results and then I load in the data and save it as hr.data. Since there were a lot of records in my original hr.data, I used the sample\_n() from dplyr to randomly select 4000 rows and use this new dataset for the remainder of my analysis

1. **Data Exploration Section**

Before doing any true statistical analysis I wanted to look at my variables and what type of data I have available to me. Using the glimpse() from dplyr I get an understanding of what each of my variables looks like and what type of data it is, number, factor, etc…

Next I was interested in seeing how many employees in my data left and how many stayed just to see if I had a roughly balanced dataset to create a model and conduct other analysis on.

Next I wanted to see if a certain department in the n had more employees than one another or if they were roughly the same. So I created a plot to see this.

I wanted to see if there was an overall satisfaction level at this company, or if it was balanced between high, med, and low

Then I was interested in seeing if employees with lower amounts of projects leave more, so I plotted this to see visually

Lastly I wanted to see how many records in my dataset were missing if any, so I could plan accordingly

1. **Logistic Regression**

This dataset had a class of leave or stay represented by 1 and 0 respectively. I wanted to develop a model to predict whether or not an employee stays or leaves.

First in order to create a model I need to convert my variables that are in factors into dummy variables. My sales and salary variables are in factors, so to create dummy variables, I use the dummy() from dummies package. After that I combine these new dummy variables back into my original dataset.

After I have this dataset I only select the columns I want to include and store this in hr.data.cleaned.

Now that I have my data, I create a training data set and a test data set to develop my model using an 80/20 split. So 80% of my data will be used to train my model, and the remaining 20% will be used to make predictions

Next I create a logistic model called hr.model that predicts left by all variables in the hr.train data

After the model is created I run summary() on it to see how it performed and what the significant variables were

Then I run confidence intervals on the model with a 95% confidence level

Now I want to find the best cutoff point to use to determine if the predicted probability represents leave or stay, so I create a cutoff point using the colAUC() from catools.

I then plot this roc curve as well as the optimal cutoff point and label it with the text()

After I have this optimal cutoff point I need to convert my predicted probabilities into 1 and 0s so I use the ifelse() to convert my probabilities into classes bases on the cutoff point

Now that all the conversion is one, I run the confusionMatrix() to see how my predicitons stacked up against the actual class from the test data set

1. **Cluster Analysis**

One of the most interesting parts of my project was seeing if there were similar groups of employees in my data.

To see this, I wanted to do a cluster analysis via the pvclust package.

I stored my clusters in hr.clusters using the pvclust() on the first 8 columns of my data, because id and class were not useful

Next I plotted those clusters to see how many different groups might exist in my data

1. **ANOVA**

In order to understand the behavior of these employees I wanted to see if on average there was a difference between time spent at the company and the department they worked in.

To test this, I ran an anova and ran a summary() on it.

Then I plotted this relationship using ggplot()

I also conducted an ANOVA to see if Satisfaction level was different on average by salary level

1. **Random Forest**

Finally I wanted to try to predict employees leaving or staying using another method, random forest.

To make sure this analysis was untouched by anything done previously I read in the original 4000 records and stored them in a new rf.data variable

I again created my 80/20 split of train and test data to develop this tree.

Next I created the rf model and for sake of computing time made it stop at 20 trees.

Then I ran summary on the model and used it to predict on my test data.

After I had my predictions, I ran a confusion matrix to see how my random forest did verses the actual class from the test data set.