**** 

**Post Graduate Program in Business Analytics and Business Intelligence (PGP-BABI) 2017-18**

**DATA MINING**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Student ID** | **Name** | **Team** |
| 1 | BACAUG17042 | Venkata Sai Sweta Kanchibhotla | 5 |
| 2 | BACAUG17031 | R Sarveshwaran |
| 3 | BACAUG17008 | Barathraj.T |
| 4 | BACAUG17058 | [Shruthi Ravi](https://classroom.greatlearning.in/groups/7684/users/6155) |
| 5 | BACAUG17030 | Sarenya Balasubramanian |

**Submitted by: Group 6**

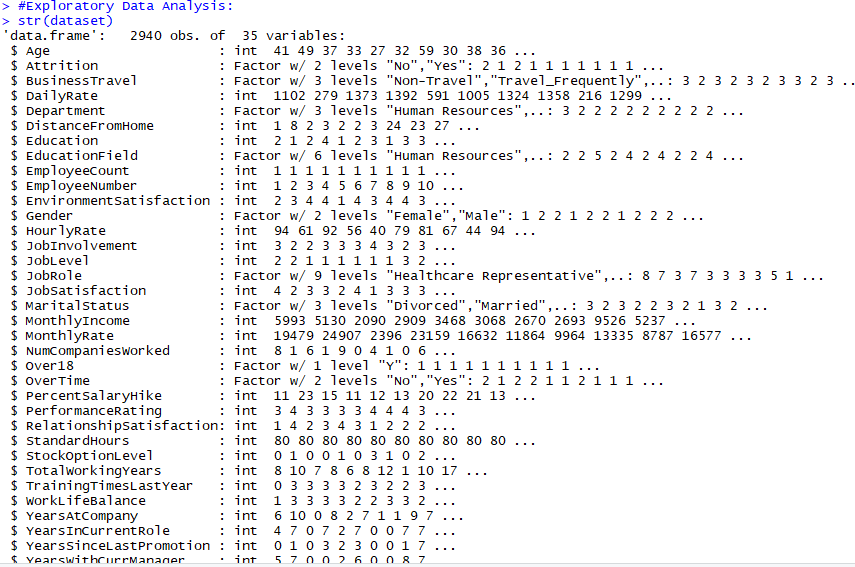
 ****

**Data Import (Target variable is "Attrition" column):**

The data has been imported from"HR\_Employee\_Attrition\_Data.csv" .

**Perform Exploratory Data Analysis:**

**1.Structure of the dataset:**

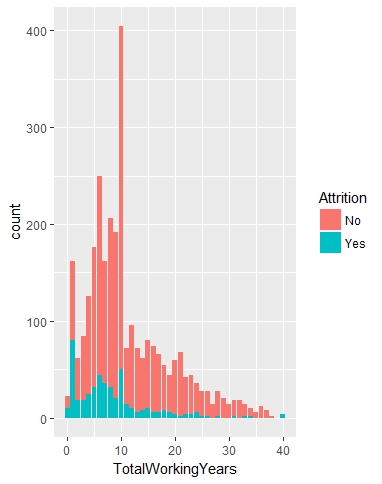
****

**2 .Descriptive Statistical Measures: Using summary (dataset)**

**3. Plots:**

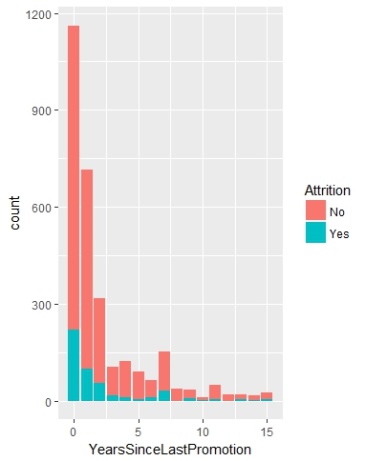
1. **Attrition:**

**The ratio of people in the company to people leaving the company is 1:5**



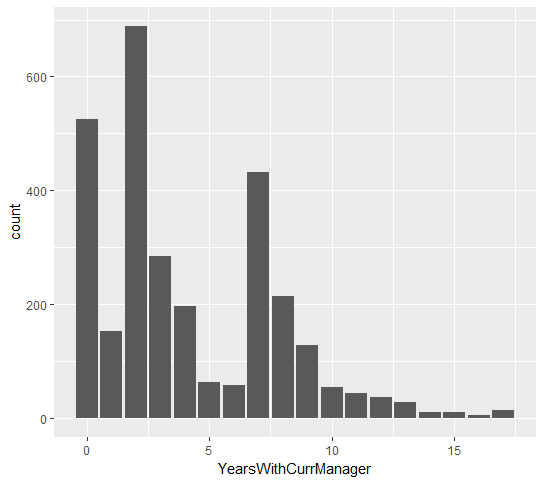
1. **Yearssincelastpromotion vs Attrition:**

**People recently promoted quit the company more than the ones not promoted.**



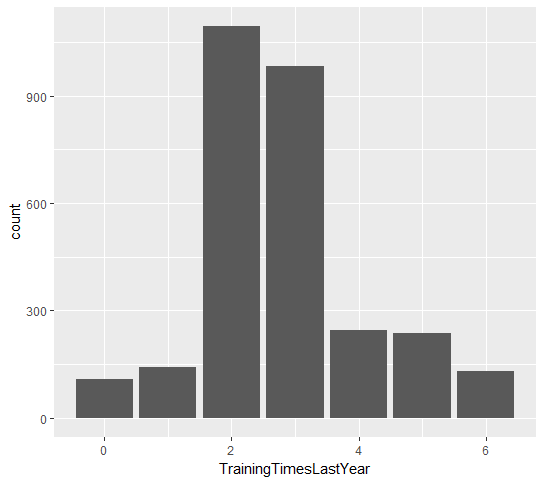
1. **YearsWithCurrentManager vs Attrition:**

**As the number of years with Current Manager Increases, Attrition decreases**



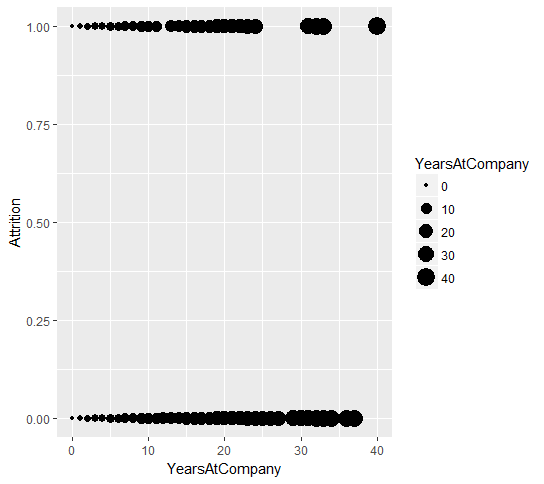
1. **TrainingTimeLastYear vs Attrition:**

**Attrition seen in employees trained between 2-4times last year.**



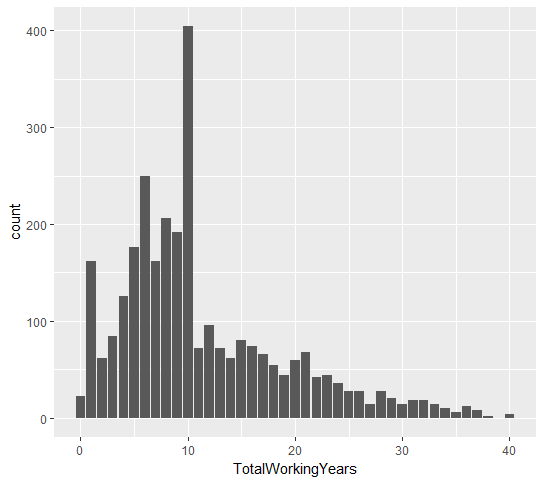
1. **YearsatCompany vs Attrition:**

**People with less no of years tend to quit the company more.**



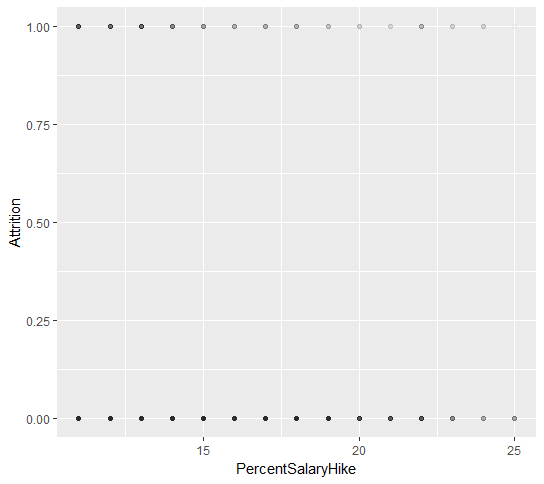
1. **Total Working Years vs Attrition:**

**People with less Experience are leaving the job more.**



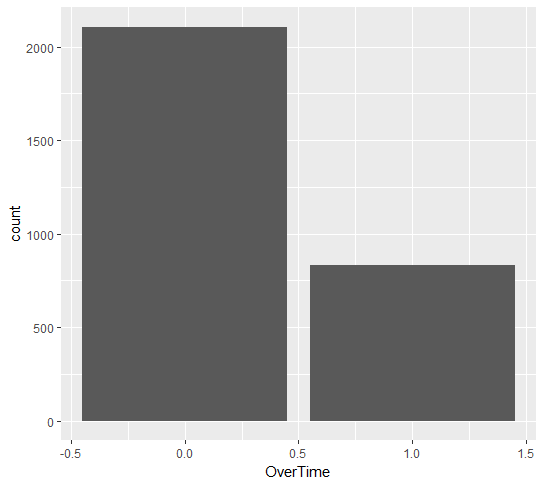
1. Present Salary Hike vs Attrition

People with Less Percent Hike leave the company.



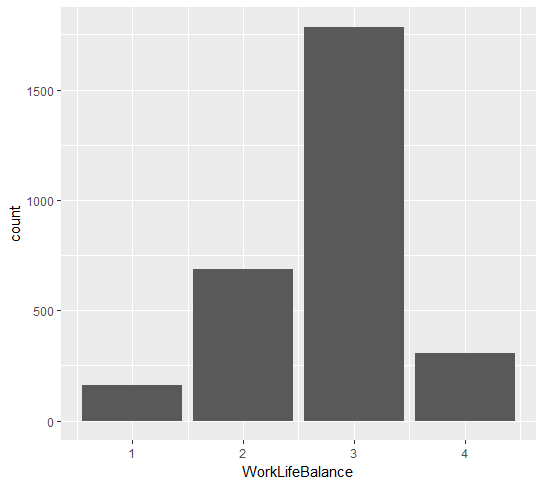
1. OverTime vs Attrition

Male employees working overtime leave the company more



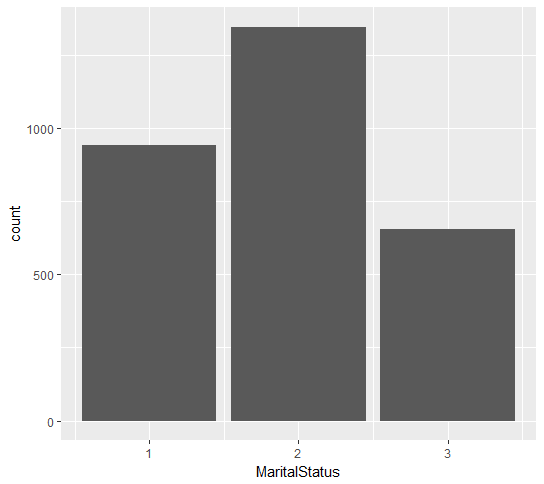
1. WorkLifeBalance vs Attrition:

People with better work life balance may tend to quit more.



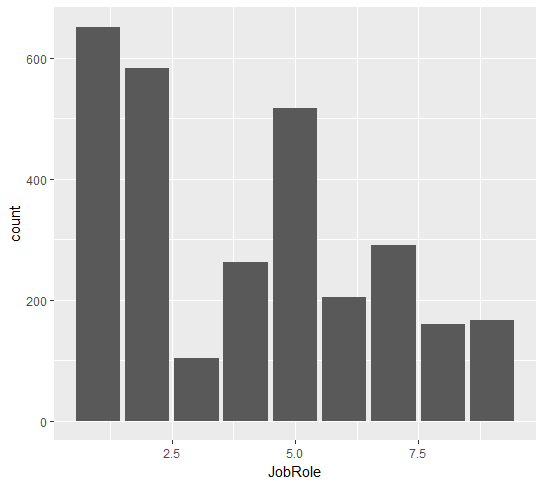
1. Marital Status vs Attrition:

Attrition highest in Employees who are single, medium in Employees who are married and least in Divorced Employees



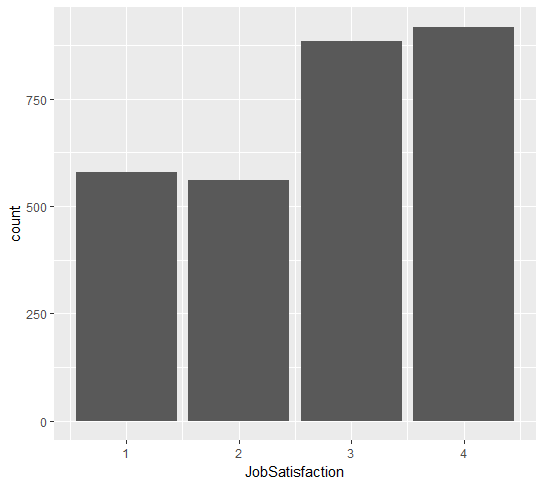
1. JobRole vs Attrition:

People leaving the company are highly involved in their jobs.



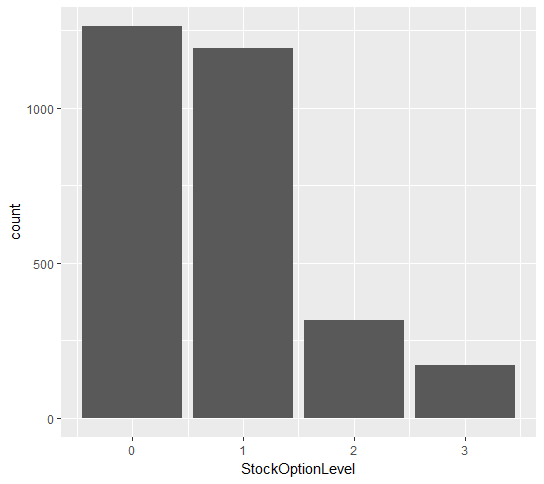
1. JobSatisfaction vs Attrition:

Low JobSatisfaction results in people leaving the company.



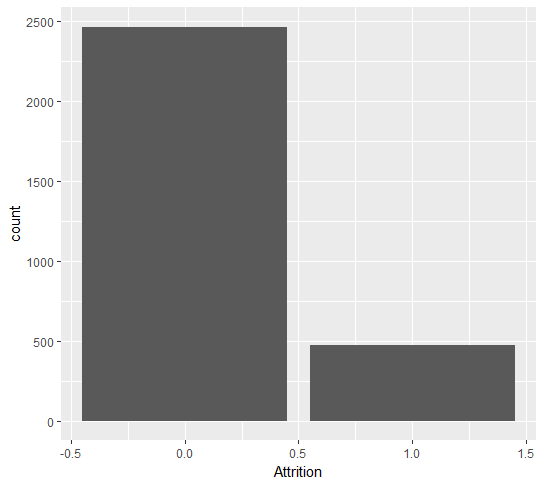
1. StockOptionLevels vs Attrition:

Attrition high in people with No or Less Stock Options



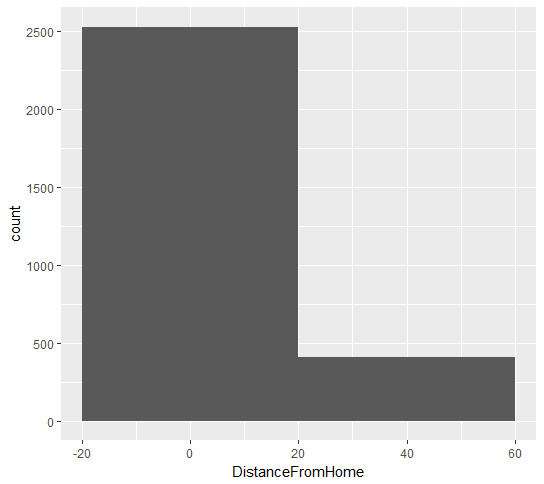
1. Gender vs Attrition

More in Male employees



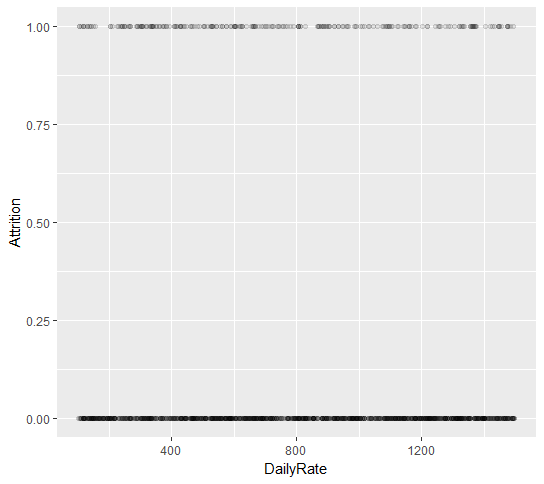
1. DistanceFromHome vs Attrition

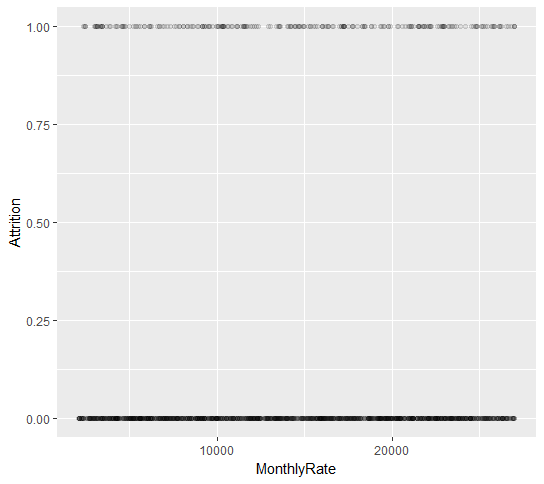
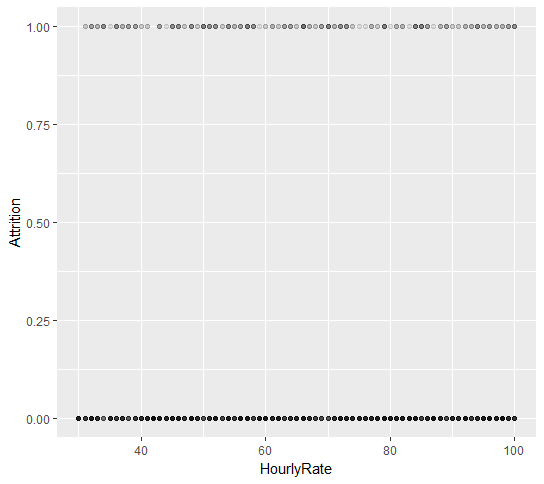
People living in shorter distances from office leave more.



1. Hourly, Daily & Monthly Rates vs Attrition:

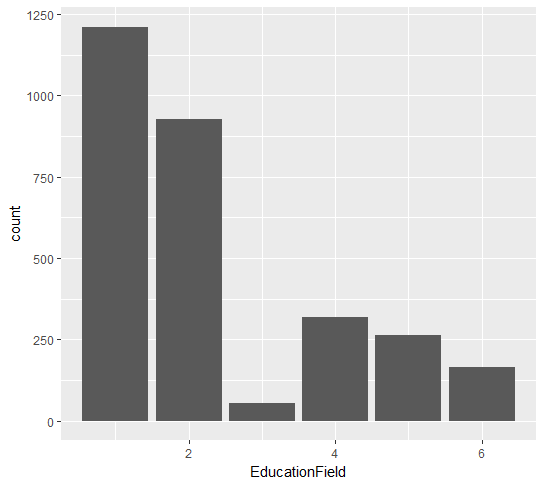
These factors do not influence the attrition.





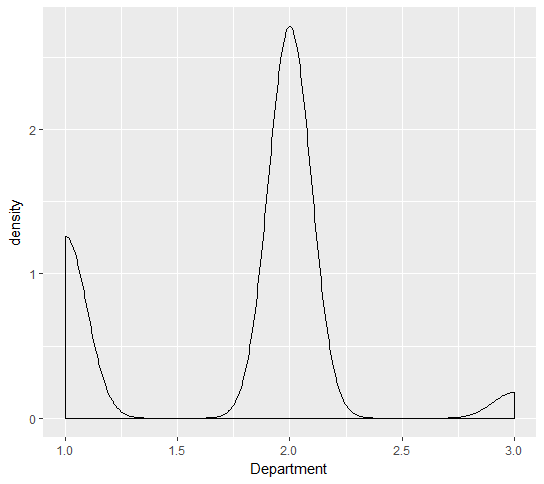
1. Education Field, Education vs Attrition

Attrition seems to be higher in Bachelors, Life Sciences and Medical

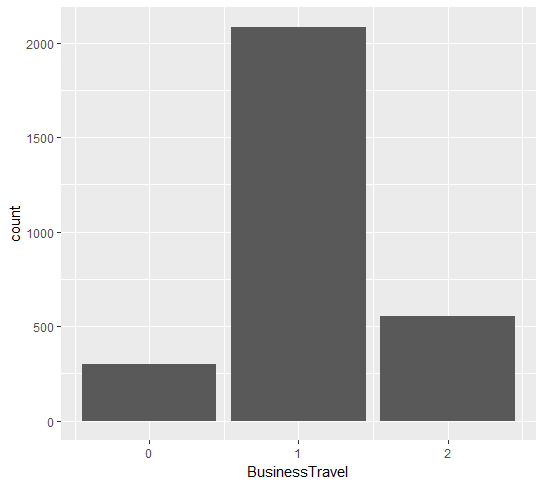
1. Department vs Attrition

People from Reasearch&Development and Sales tend to quit more compared to HR



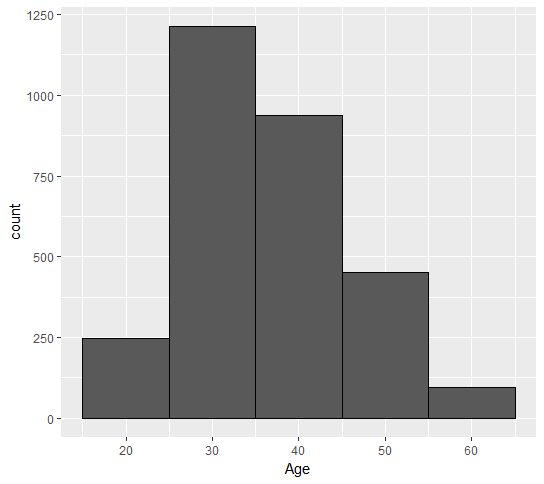
1. Business Travel vs Attrition

Attrition is directly proportional to Travel among the employees.



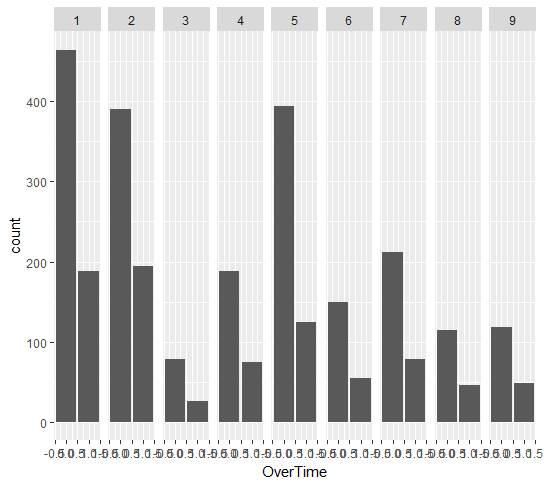
1. Age vs Attrition:

Employees around 30years leave the company more



1. Others:

High Attrition among Sales Representatives and Lab Technicians who work Overtime.



**Identify columns which are of no use. Drop those columns:**

**We have identified columns which need to be dropped in the following ways:**

**1. Correlated Variables: Since we have many independent variables, it is obvious that we will also have correlated variables, which we identified using corrplot function.**

From the Correlation plot, we have dropped the following variables:

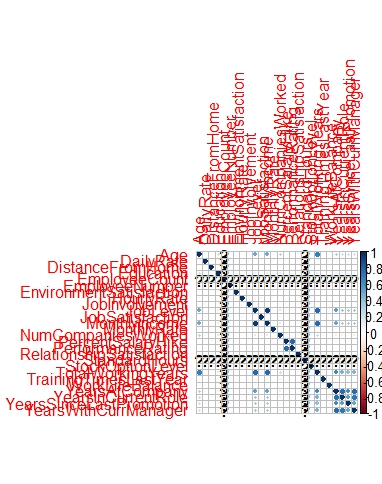
a)  YearsAtCompany

b) YearsInCurrentRole

c) YearsSinceLastPromotion

d) Monthly Income

e) Performance Rating



2. **Redundant/Least Importance variables**: We have dropped the following variables based on intuition and confirmed the same using Boruta .

Employee Count and Employee Number fields do not carry any useful information:

f) Employee Count

g) Employee Number

h) Over18   - Over18 field can be calculated from the Age column:

i) Standard Hours - Standard Hours is same across all the columns:

**Boruta output:**

Employee Count 0.000000 0.000000 0.000000 0.0000 0.0000 Rejected

Employee Number -5.177087 -4.938704 -6.649008 -4.0351 0.000 Rejected

Over18 0.0000 0.000000 0.000000 0.0000 0.000000 Rejected

Standard Hours 0.0000 0.000000 0.0000 0.0000 0.00000 Rejected

**Split the data in Dev & Hold Out sample (70:30):**

The imported data after dropping of redundant and correlated variables has been split into training and test sets using sample. Split on Attrition variable with split ratio=0.7 from “caTools” package.

All the models have been trained on training set and validated on test set.

The Target variable has been encoded as Yes-1 No-0.

**Write Hypothesis and validate the Hypothesis:**

**H0:** Attrition (dependent variable) is not affected by other independent variables.

**H1:** Attrition (dependent variable) is affected by other independent variables.

F-Statistic is ***significant*** and hence we **reject Null Hypothesis** and ***accept Alternative Hypothesis.*** i.e. Dependent variable Attrition is affected by the given independent variables.

Output:

Capturer

**CART Model:**

We have built a classification CART model with parameters:

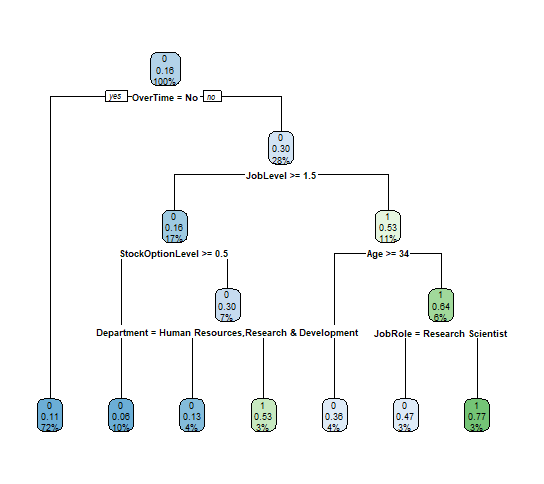
minsplit = 100; the minimum number of observations that must exist in a node in order for a split to be attempted.

minbucket = 10; the minimum number of observations in any terminal node (leaf node)

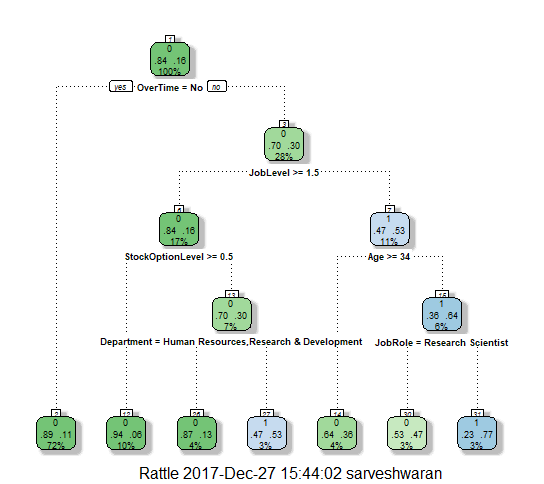
xval = 10,no of cross validations and to avoid Over-fitting

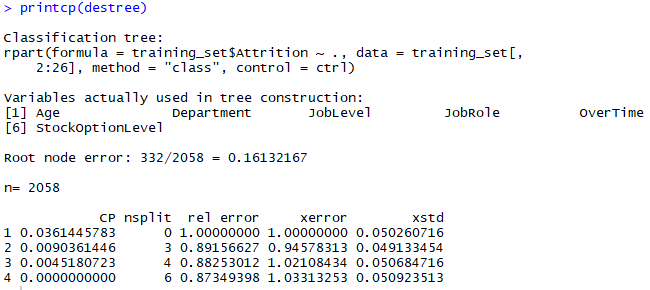
cp = 0 so that the cross-validated error rate is minimum

**rplot:**

****

**Fancy r-plot:**

****



**Validate CART Model:**

**Confusion Matrix:**

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | 722 | 18 |
| 1 | 108 | 34 |

**Summary of CART model:**

|  |  |
| --- | --- |
| **Confusion Matrix outcomes - CART** |  |
| Accuracy=(TP+TN)/total | 0.857142857 |
| Misclassification Rate or error rate =(FP+FN)/total | 0.142857143 |
| Sensitivity or recall or True Positive Rate = TP/actual yes | 0.23943662 |
| False Positive Rate = FP/actual no | 0.024324324 |
| Specificity = TN/actual no | 0.975675676 |
| Precision = TP/predicted yes | 0.653846154 |
| Prevalence = actual yes/total | 0.160997732 |

***Accuracy = 85.71%*** on testing set.

**Neural Networks:**

**Dataset been reordered as-Attrition followed by all numeric fields and categorical fields for easier cleaning of data for model building.**

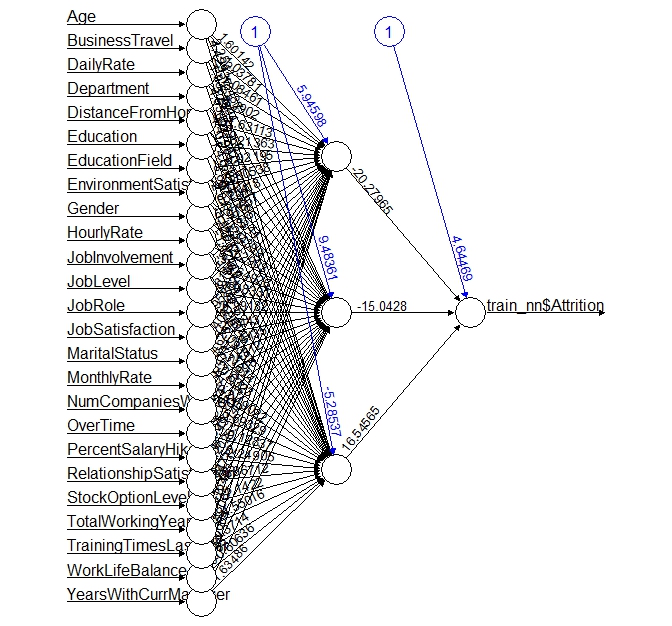
**All the categorical variables have been encoded as factors and all the variables are scaled to ensure easier and faster computations**

1. **Neural Network 1: Using Neural Net**

Attrition (Dependent variable) is compared to all Independent Variables .

Parameters considered:

* 2 hidden layers.
* Threshold  set to 0.1
* Stepmax: 2000.



**g) Validate NN model on Hold Out. If need be improvise:**

Threshold = 0.5, if predicted value is above 0.5 encoded as 1 else 0.

**Confusion matrix of Neural Network:**

|  |  |  |
| --- | --- | --- |
|  | 0 | 0 |
| 0 | 681 | 59 |
| 1 | 113 | 29 |

**Summary of Confusion Matrix:**

|  |  |
| --- | --- |
| **Confusion Matrix outcomes - Neural Network** |  |
| Accuracy=(TP+TN)/total | 0.804988662 |
| Misclassification Rate or error rate =(FP+FN)/total | 0.195011338 |
| Sensitivity or recall or True Positive Rate = TP/actual yes | 0.204225352 |
| False Positive Rate = FP/actual no | 0.07972973 |
| Specificity = TN/actual no | 0.92027027 |
| Precision = TP/predicted yes | 0.329545455 |
| Prevalence = actual yes/total | 0.160997732 |

***Accuracy Rate: 80.49%*** on test set using Neural Net.

On Improvising the Model using ***h2o.deeplearning:***

Parameters:2hidden layers with 13 neurons each over 100 epochs.

***Confusion Matrix for h2o Neural Network:***

|  |  |  |
| --- | --- | --- |
|  | 0 | 0 |
| 0 | 707 | 33 |
| 1 | 55 | 87 |

***Summary of Neural Network (h2o) model:***

|  |  |
| --- | --- |
| **Confusion Matrix outcomes - H2O Neural Network** |  |
| Accuracy=(TP+TN)/total | 0.900226757 |
| Misclassification Rate or error rate =(FP+FN)/total | 0.099773243 |
| Sensitivity or recall or True Positive Rate = TP/actual yes | 0.612676056 |
| False Positive Rate = FP/actual no | 0.044594595 |
| Specificity = TN/actual no | 0.955405405 |
| Precision = TP/predicted yes | 0.725 |
| Prevalence = actual yes/total | 0.160997732 |

***Accuracy = 90.02*%** has been achievedon test set using h2o

***Reasons for improved accuracy:***

1. Runs an instance of a computer system which will help the model learn efficiently along with lot of options.

2. Parameter tuning.

**j) Compare NN with CART:**

Comparing Improvised Neural Network with the CART model:

|  |  |  |
| --- | --- | --- |
| **Confusion Matrix** | **CART** | **h20 - Neural Network** |
| Accuracy=(TP+TN)/total | 0.857143 | 0.900226757 |
| Misclassification Rate or error rate =(FP+FN)/total | 0.142857 | 0.099773243 |
| Sensitivity or recall or True Positive Rate = TP/actual yes | 0.239437 | 0.612676056 |
| False Positive Rate = FP/actual no | 0.024324 | 0.044594595 |
| Specificity = TN/actual no | 0.975676 | 0.955405405 |
| Precision = TP/predicted yes | 0.653846 | 0.725 |
| Prevalence = actual yes/total | 0.160998 | 0.160997732 |

**k) Combine NN and CART into Ensemble Model:**

1. Using XGBoost:

Confusion matrix:

|  |  |  |  |
| --- | --- | --- | --- |
| Prediction | | | |
| Actual |  | 0 | 1 |
| 0 | 731 | 9 |
| 1 | 79 | 63 |

Summary of XGBOOST:

|  |  |
| --- | --- |
| **Confusion Matrix outcomes – XgBoost** |  |
| Accuracy=(TP+TN)/total | 0.900226757 |
| Misclassification Rate or error rate =(FP+FN)/total | 0.099773243 |
| Sensitivity or recall or True Positive Rate = TP/actual yes | 0.443661972 |
| False Positive Rate = FP/actual no | 0.012162162 |
| Specificity = TN/actual no | 0.987837838 |
| Precision = TP/predicted yes | 0.875 |
| Prevalence = actual yes/total | 0.160997732 |
|  |  |

***Reasons for High Accuracy:***

1. Parallel Processing

2. Automatic tree pruning and built in classification.

**2) Averaging:**

Formula: Taking the average of probabilities of the CART and Neural Network 1:

**Confusion Matrix:**

|  |  |  |  |
| --- | --- | --- | --- |
| Predicted Values | | | |
| Actual   Values |  | 0 | 1 |
| 0 | 493 | 247 |
| 1 | 96 | 46 |

**Summary of Averaging:**

|  |  |
| --- | --- |
| **Confusion Matrix outcomes Average of all models** |  |
| Accuracy=(TP+TN)/total | 0.611111111 |
| Misclassification Rate or error rate =(FP+FN)/total | 0.388888889 |
| Sensitivity or recall or True Positive Rate = TP/actual yes | 0.323943662 |
| False Positive Rate = FP/actual no | 0.333783784 |
| Specificity = TN/actual no | 0.666216216 |
| Precision = TP/predicted yes | 0.156996587 |
| Prevalence = actual yes/total | 0.160997732 |

**3.Random Forest:**

**Formula:**

**tclassifier <- tuneRF(x = training\_set[, 2:26],**

**y=training\_set$Attrition, mtryStart = 3, ntreeTry=100, stepFactor = 1.5, improve = 0.0001, trace=TRUE, plot = TRUE, doBest = TRUE,**

**nodesize = 10, importance=TRUE)**

**rantree = randomForest(x = training\_set[, 2:26], y = training\_set$Attrition,**

**ntree = 100, nodesize = 10, mtry=13, importance = TRUE)**

**We give the Optimal mtry value = 13**

**Confusion matrix:**

|  |  |  |  |
| --- | --- | --- | --- |
| Prediction | | | |
| Actual |  | 0 | 1 |
| 0 | 732 | 8 |
| 1 | 38 | 104 |

**Summary of Random Forest:**

|  |  |
| --- | --- |
| **Confusion Matrix outcomes - Random Forest** |  |
| Accuracy=(TP+TN)/total | 0.947845805 |
| Misclassification Rate or error rate =(FP+FN)/total | 0.052154195 |
| Sensitivity or recall or True Positive Rate = TP/actual yes | 0.732394366 |
| False Positive Rate = FP/actual no | 0.010810811 |
| Specificity = TN/actual no | 0.989189189 |
| Precision = TP/predicted yes | 0.928571429 |
| Prevalence = actual yes/total | 0.160997732 |

**Accuracy = 94.78% on running it in test set.**

**Conclusion: Random forest and XGBoost models are performing better than CART and Neural Network models.**