PROJECT NAME :
TITLE: *IBM EMPLOYEE ATTRITION*

PREDICTIVE MODELING

Batch:
DSP 19, Group 3

Submitted by:
Tejinder Singh Wadhwa & Qays Ibrahim Tole

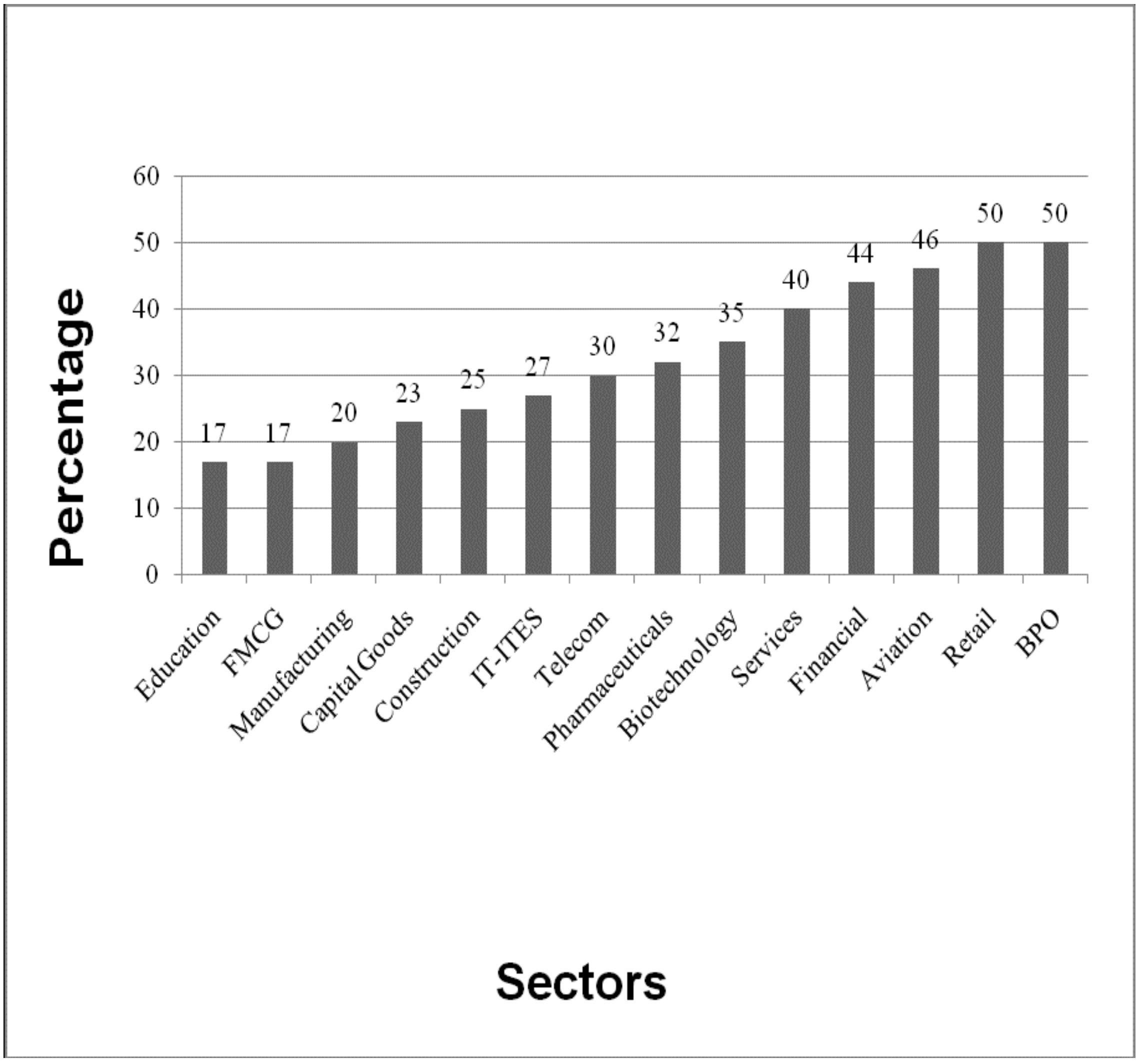
INTRODUCTION:

What is Attrition?

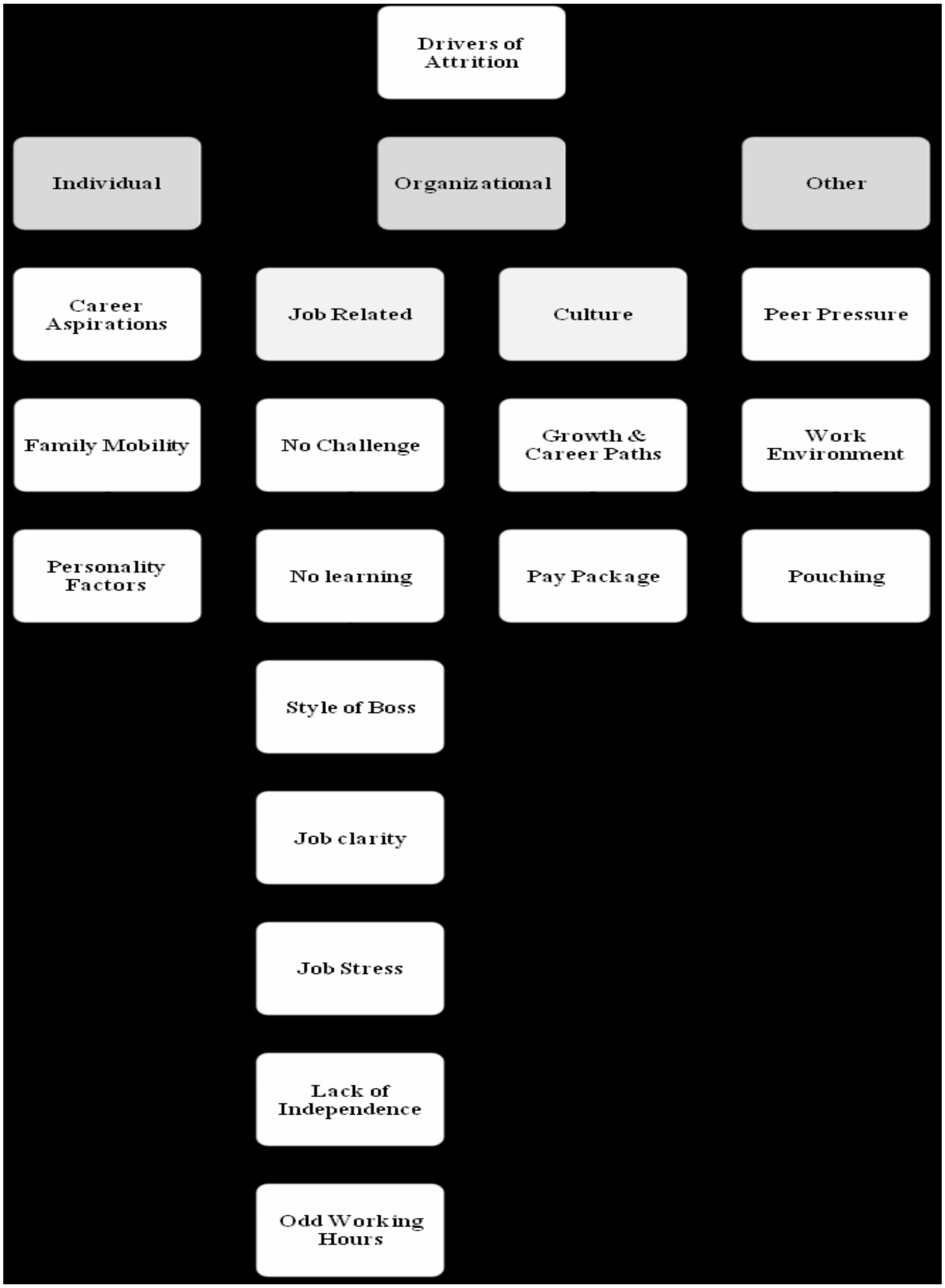
Attrition simply means "A reduction in the number of employees through retirement, resignation or death."

Attrition Scenario in India

Almost all the sectors in India are facing attrition, but the reasons and effects are unique to each sector. The attrition rates of different sector are illustrated below:



Drivers for Attrition:



Cost of Attrition

Employee attrition is a costly dilemma for all organizations. Employee attrition costs 12 to 18 months' salary for each leaving manager or professional.

OBJECTIVE:

OBJECTIVE 1:

Identifying the factors that cause employees to leave the organization and explore important questions such as ‘compare average monthly income by education and attrition’ or ‘is distance from home a case for attrition’?

OBJECTIVE 2:

Build classification model to predict which employee is likely to churn and help the business to devise policies and attract back the right talent.

TOOLS & PACKAGES USED:

Tools:

1. **R STUDIO:** We used this for importing the data, handling outliers, scaling, creating models & their summary as well as creating histogram charts

2. **Tableau:** We have used tableau for creating various pie charts and line graphs for presentation

Packages:

1. **caret:** The **caret** package (short for Classification and Regression Training) contains functions to streamline the model training process for complex regression and classification problems. One of the primary tools in the package is the `train` function which we used it in our coding

2. **e071:** Functions for latent class analysis, short time Fourier transform, fuzzy clustering, support vector machines, shortest path computation, bagged clustering, naive Bayes classifier. We used it for creating SVM models.

3. **ggplot2:** ggplot2 is a system for declaratively creating graphics, based on The Grammar of Graphics. Used for crating box plots, histograms etc.

4. **rminer:** We used this for finding important variables as per SVM model.

5. **randomForest:** Implements Breiman's random forest algorithm (based on Breiman and Cutler's original Fortran code) for classification and regression. We used this package to create Random Forest models

FEATURE SELECTION:

Deleting Variables:

Data file had **35 variables**. We deleted the following variables from the data set for the reasons mentioned below:

Variable Name	Reason for deletion
EmployeeCount	Variable had only 1 level which gives error when included in model creation
EmployeeNumber	Unique number for each employee. No significance in model creation
Over18	Variable had only 1 level which gives error when included in model creation
StandardHours	Variable had only 1 level which gives error when included in model creation

Creating New variable:

If we have a look at the datafile then you come across the below three variables that are related to the employee job satisfaction:

i. EnvironmentSatisfaction = Employee satisfaction score from 1 to 4 with 1 = Low to 4 = Very High
ii. JobSatisfaction = Employee Job satisfaction score from 1 to 4 with 1 = Low to 4 = Very High
iii. RelationshipSatisfaction = Employee Job satisfaction score from 1 to 4 with 1 = Low to 4 = Very High

If you use scores of the above three features in any combination then we can derive how satisfied that employee is with his/her job in the organization

Hence; we thought it would be a nice idea to derive a new variable which will combine the scores of the above three variables and provide us the employee happiness index with his/her job. We named it as "Work_Happiness_Index"

We use the below condition to derive Happiness index of the employees:

i. If JobSatisfaction >= 3 and EnvironmentSatisfaction >=3 AND RelationshipSatisfaction>=3) we categorized such employees "Work_Happiness_Index" as **"Very Happy"**
ii. If JobSatisfaction >= 3 AND (EnvironmentSatisfaction >=3 OR RelationshipSatisfaction>=3) we categorized such employees "Work_Happiness_Index" as **"Happy"**
iii. If JobSatisfaction <=2 AND (EnvironmentSatisfaction >=3 OR RelationshipSatisfaction >=3) we categorized such employees "Work_Happiness_Index" as **"Sad"**
iv. If JobSatisfaction <= 2 AND EnvironmentSatisfaction <=2 AND RelationshipSatisfaction <=2 we categorized such employees "Work_Happiness_Index" as **"Very Sad"**
v. Rest Employees who didn't fall in the above 4 categories are classified a "Work_Happiness_Index" as **"Somewhat Happy"**.

After Feature Selection the original data file is left with **32 Variables** only which were used for Model building.

ALGORTIHMS USED:

- **Logistic regression**

  Logistic regression measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function.

- **Decision tree**

  Decision Tree algorithm belongs to the family of supervised learning algorithms. Unlike other supervised learning algorithms, decision tree algorithm can be used for solving regression and classification problems too.

  The decision tree algorithm tries to solve the problem, by using tree representation. Each internal node of the tree corresponds to an attribute, and each leaf node corresponds to a class label.

  Decision Tree Algorithm Pseudocode:

i. Place the best attribute of the dataset at the root of the tree.
ii. Split the training set into subsets. Subsets should be made in such a way that each subset contains data with the same value for an attribute.
iii. Repeat step 1 and step 2 on each subset until you find leaf nodes in all the branches of the tree.

- **Random Forests:**

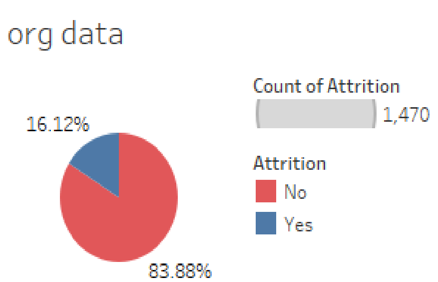
  Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random forest is like bootstrapping algorithm with Decision tree model.

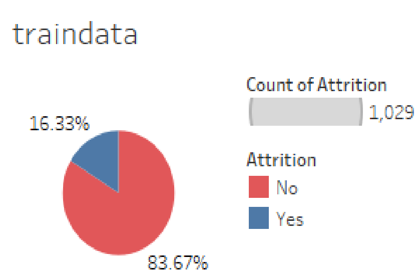
- **SVM:**

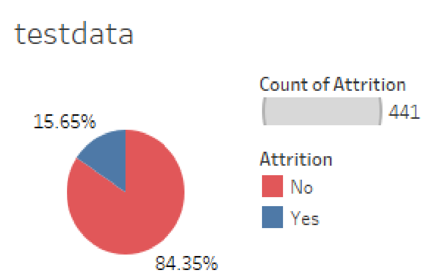
  Support-vector machines (SVMs, also support-vector networks are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like outliers detection

DATA SPLITTING:

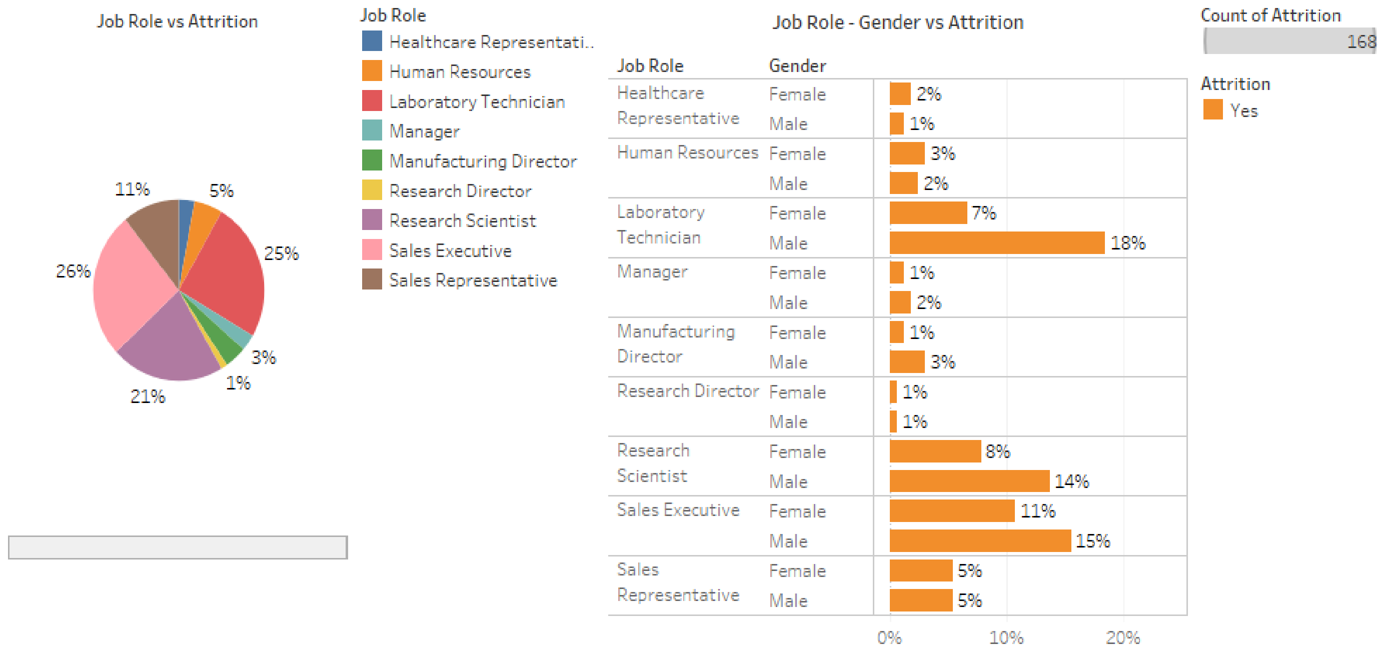
- We have made a random split of the ratio **70:30** respectively using the sample() function of R.
- Where in 70% of the data represents training data with **1029 observations and 32 variables**. We named this file as **"trainData.csv".**
- And 30% of the data represents the testing data with **441 observation and 32 variables**. We named this file as **"testData.csv".**
- Below graphs shows the percentage split of "Yes" and "No" of our target variable "Attrition" in Train and Test data is almost same as it was in the original data file. This implies that it is a **good data split**.
- As you can see from the below pie charts the ratio of count of "No" Attrition is way more than the "Yes" counts. It's almost 83:17 in the original file itself. This could lead to creation of models little bias towards predicting "No" Attrition more accurately than "Yes".

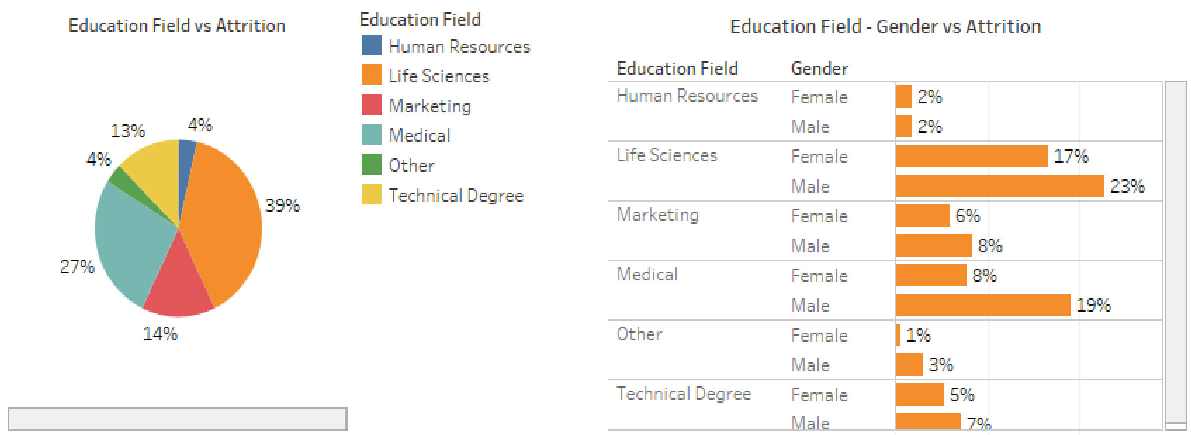




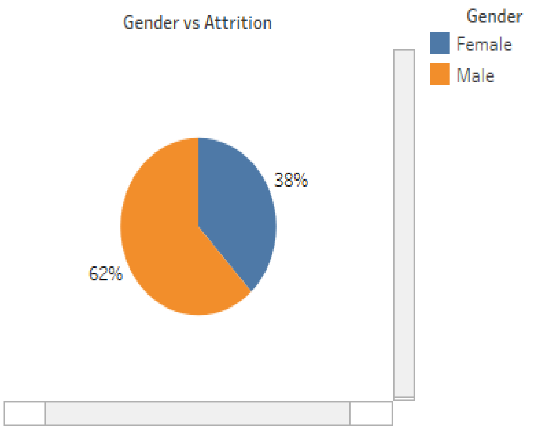


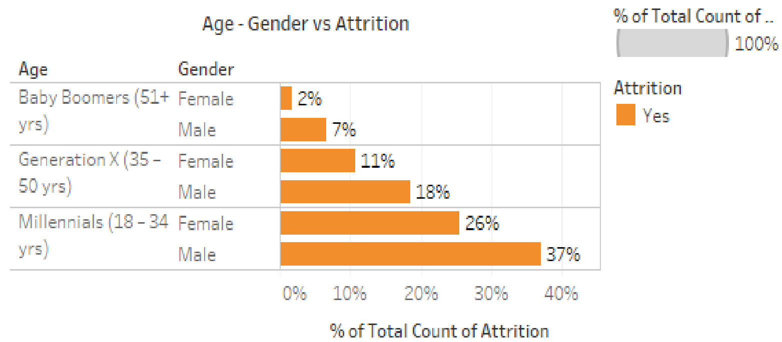
Training Data Visulization

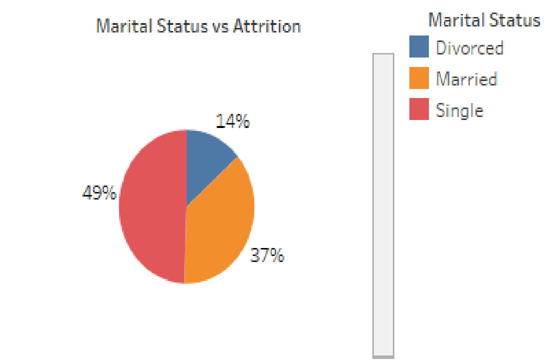


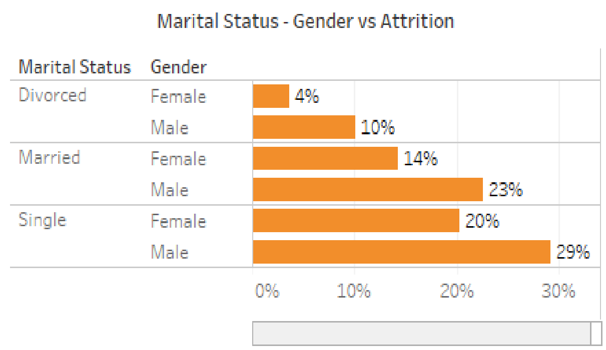


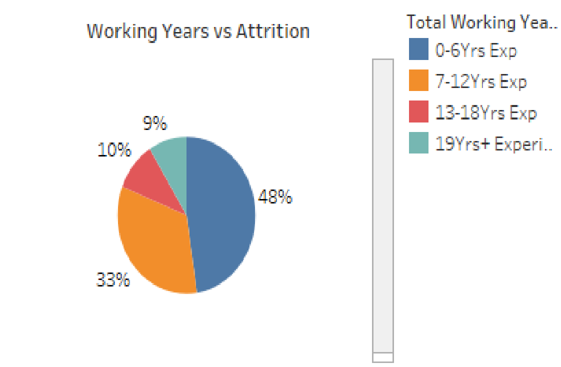
Training Data Visulization Contd...

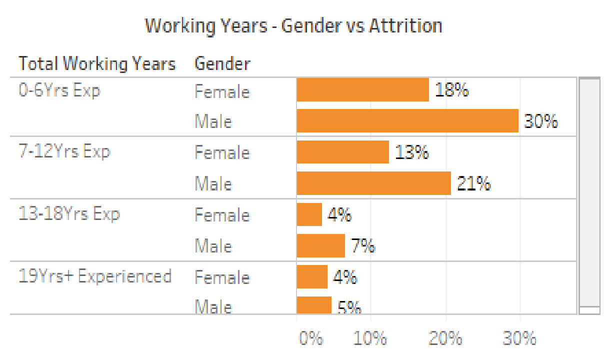












TECHNIQUE 1:
Model Building WITHOUT any data manipulation

1. **Step 1**
   First, we checked Train Data (trainData.csv) for **Blanks or '?'** (question marks) in the data. We were unable to find any.

2. **Step 2**
   We then created following 4 models: Logistic Regression, Decision Tree, Random Forest and Support Vector Machines to predict our target Variable "Attrition" comparing with all 32 columns (variables) of the "trainData.csv".

3. **Step 3**
   We then calculated the best threshold that gives the best accuracy and least total number of errors for each model. Details below:

Model#	**Algorithm used**	**Threshold Value**
MODEL 1	LOGISTIC REGRESSION (Attrition ~ .)	Any prediction value > **0.51,**"Yes","No"
MODEL 2	DECISION TREE (Attrition ~ .) WITH INFORMATION GAIN	tuneLength = 14
MODEL 3	DECISION TREE (Attrition ~ .) WITH GINI INDEX	tuneLength = 9
MODEL 3	RANDOM FOREST (Attrition ~ .)	tuneLength = 5
MODEL 4	SVM (Attrition ~ .)	cost = 9, gamma=0.015

Confusion Matrix and accuracy of all the 4 models are shown in the next page.

4. **Step 4**
   We then compared the accuracy as well as the Sensitivity and Specificity of all the 4 models to decide which model is best for predicting.

Confusion and Accuracy : Without any data manipulation

LOGISTIC REGRESSION OUTPUT

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  360  36
       Yes  12  33

               Accuracy : 0.8912
                 95% CI : (0.8583, 0.9186)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.0025818

                  Kappa : 0.5196
 Mcnemar's Test P-Value : 0.0009009

            Sensitivity : 0.47826
            Specificity : 0.96774
         Pos Pred Value : 0.73333
         Neg Pred Value : 0.90909
             Prevalence : 0.15646
         Detection Rate : 0.07483
   Detection Prevalence : 0.10204
      Balanced Accuracy : 0.72300

       'Positive' Class : Yes
```

DECISION TREE WITH GINI INDEX OUTPUT

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  356  48
       Yes  16  21

               Accuracy : 0.8549
                 95% CI : (0.8185, 0.8864)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.2809805

                  Kappa : 0.3222
 Mcnemar's Test P-Value : 0.0001066

            Sensitivity : 0.30435
            Specificity : 0.95699
         Pos Pred Value : 0.56757
         Neg Pred Value : 0.88119
             Prevalence : 0.15646
         Detection Rate : 0.04762
   Detection Prevalence : 0.08390
      Balanced Accuracy : 0.63067

       'Positive' Class : Yes
```

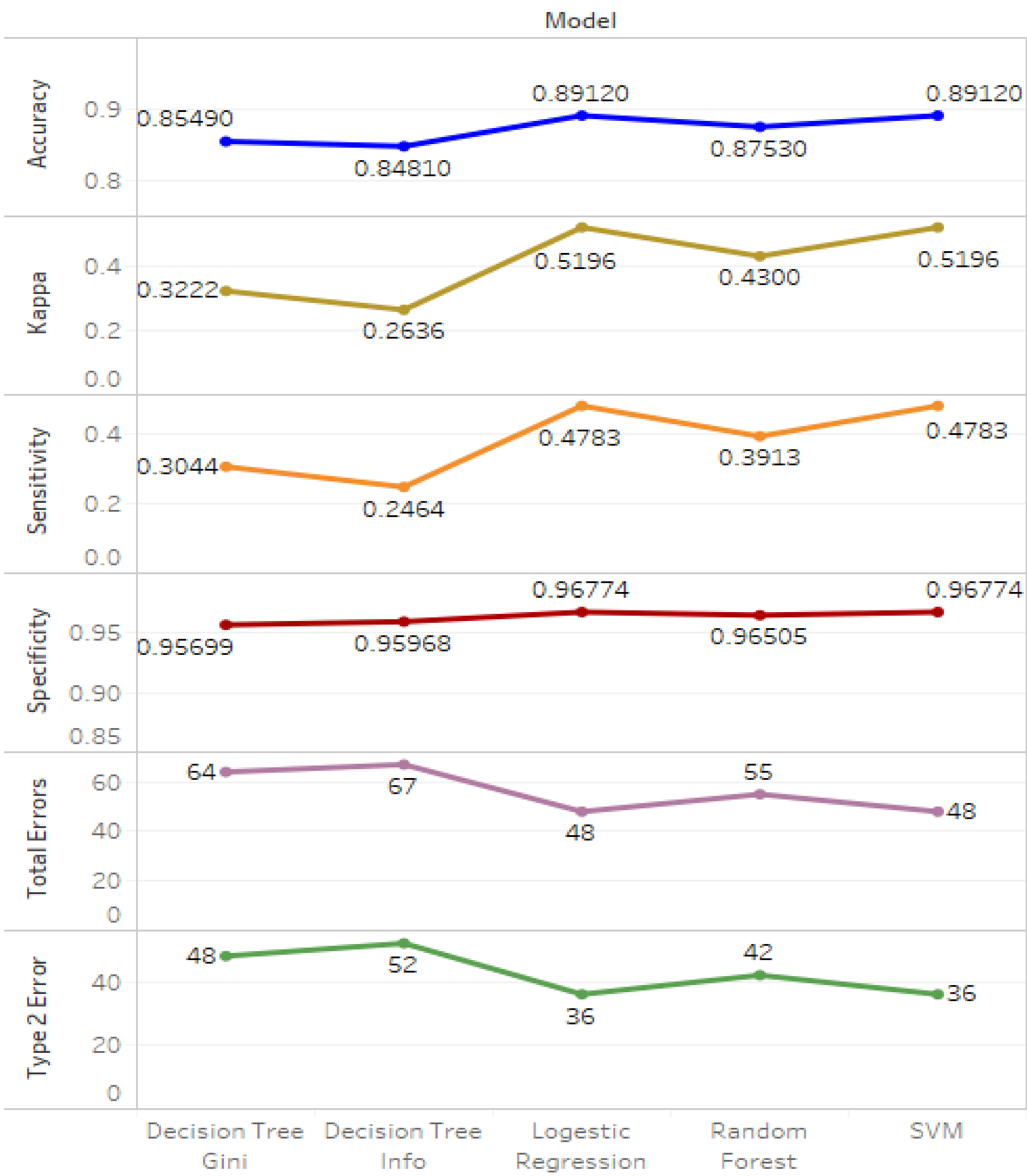
RANDOM FOREST OUTPUT

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  359  42
       Yes  13  27

               Accuracy : 0.8753
                 95% CI : (0.8408, 0.9046)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.0355092

                  Kappa : 0.43
 Mcnemar's Test P-Value : 0.0001597

            Sensitivity : 0.39130
            Specificity : 0.96505
         Pos Pred Value : 0.67500
         Neg Pred Value : 0.89526
             Prevalence : 0.15646
         Detection Rate : 0.06122
   Detection Prevalence : 0.09070
      Balanced Accuracy : 0.67818

       'Positive' Class : Yes
```

SVM OUTPUT

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  360  36
       Yes  12  33

               Accuracy : 0.8912
                 95% CI : (0.8583, 0.9186)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.0025818

                  Kappa : 0.5196
 Mcnemar's Test P-Value : 0.0009009

            Sensitivity : 0.47826
            Specificity : 0.96774
         Pos Pred Value : 0.73333
         Neg Pred Value : 0.90909
             Prevalence : 0.15646
         Detection Rate : 0.07483
   Detection Prevalence : 0.10204
      Balanced Accuracy : 0.72300

       'Positive' Class : Yes
```

Model Accuracy Graph :
Without any data manipulation



Technique 1 : Conclusion

Top 2 Best Models as per Technique 1

Technique#	Best Model	Threshold	Accuracy	Kappa	Sensitivity	Specificity
Technique 1	Logistic Regression (var=32)	0.51	0.8912	0.5196	0.47826	0.96774
Technique 1	SVM (Var=32)	c=9,g=0.015	0.8912	0.5196	0.47826	0.96774

After comparison, Logistic Regression and SVM model has the best **accuracy of 0.8912** and **Sensitivity of 0.4783**. Hence, these two are the best models to predict data if we compare Attrition with all the 32 remaining columns and without any data manipulation.

We will use these accuracies of each model as the base to tune and create a better model.

FEATURE ENGINERRING:

Data Manipulation:

1. Removing Outliers:

What are outliers?

- In statistics, an outlier is an observation point that is distant from other observations. An outlier may be due to variability in the measurement or it may indicate experimental error; the latter are sometimes excluded from the data set.

Why outliers treatment is important?

- Outliers in data can distort predictions and affect the accuracy, if you don't detect and handle them appropriately especially in regression models. Because, it can drastically bias/change the fit estimates and predictions.

Outlier technique used:

- We used the Winsorizing technique to handle the outliers.
- Winsorizing is the transformation of statistics by limiting extreme values in the statistical data to reduce the effect of possibly spurious outliers. A typical strategy is to set all outliers to a specified percentile of the data; for example, a 90% winsorization would see all data below the 5th percentile set to the 5th percentile, and data above the 95th percentile set to the 95th percentile.

2. Scaling (Z standardization scaling)

Why Scaling?

- The purpose of scaling is to put all the results onto a common scale. This is to avoid models biasness, this helps us to ensure models are not biased based on different ranges of numeric variables, this in turn helps in increasing the accuracy of the model

Use of scale() function:

- We have used the scale() function of R to scale all numeric variables that usese the Z standardization
- scale() is a generic function whose default method centers and/or scales the columns of a numeric matrix, within the range of +3 to -3.

e.g. See below the scaling for "Age" variable

Before scaling	After scaling

`> summary(orgData["Age"])`
`Age`
`Min.   :18.00`
`1st Qu.:30.00`
`Median :36.00`
`Mean   :36.92`
`3rd Qu.:43.00`
`Max.   :60.00`
	`> orgData["Age"] <- as.data.frame(scale(orgData["Age"]))`
	`> summary(orgData["Age"])`
	`Age`
	`Min.   :-2.0715`
	`1st Qu.:-0.7579`
	`Median :-0.1011`
	`Mean   : 0.0000`
	`3rd Qu.: 0.6651`
	`Max.   : 2.5260`

TECHNIQUE 2:
Model Building WITH All Features & AFTER Data manipulation

1. **Step 1:**

- We again created following 5 models: Logistic Regression, Decision Tree, Random Forest and Support Vector Machines to predict our target Variable "Attrition" comparing with all 32 columns (variables) of the "trainData.csv" with outliers and scaling done

2. **Step 2:**

- We then calculated the best threshold that gives the best accuracy and least total number of errors for each model. Details below:

Model#	**Algorithm used**	**Threshold Value**
MODEL 1	LOGISTIC REGRESSION (Attrition ~ .)	Any prediction value > **0.56,**"Yes","No"
MODEL 2	DECISION TREE (Attrition ~ .) WITH INFORMATION GAIN	tuneLength = 14
MODEL 3	DECISION TREE (Attrition ~ .) WITH GINI INDEX	tuneLength = 9
MODEL 4	RANDOM FOREST (Attrition ~ .)	tuneLength = 500
MODEL 5	SVM (Attrition ~ .)	cost = 7, gamma=0.015

Confusion and Accuracy:

All Features & AFTER Data manipulation

<table>
<tr><th>LOGISTIC REGRESSION OUTPUT</th><th>DECISION TREE WITH GINI INDEX OUTPUT</th></tr>
<tr><td><pre>
Confusion Matrix and Statistics

          Reference
Prediction No Yes
       No 368 40
       Yes   4 29

               Accuracy : 0.9002
                 95% CI : (0.8684, 0.9266)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.0003614

                  Kappa : 0.52
 Mcnemar's Test P-Value : 1.317e-07

            Sensitivity : 0.42029
            Specificity : 0.98925
         Pos Pred Value : 0.87879
         Neg Pred Value : 0.90196
             Prevalence : 0.15646
         Detection Rate : 0.06576
   Detection Prevalence : 0.07483
      Balanced Accuracy : 0.70477

       'Positive' Class : Yes
</pre></td><td><pre>
Confusion Matrix and Statistics

          Reference
Prediction No Yes
       No 356 48
       Yes 16 21

               Accuracy : 0.8549
                 95% CI : (0.8185, 0.8864)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.2809805

                  Kappa : 0.3222
 Mcnemar's Test P-Value : 0.0001066

            Sensitivity : 0.30435
            Specificity : 0.95699
         Pos Pred Value : 0.56757
         Neg Pred Value : 0.88119
             Prevalence : 0.15646
         Detection Rate : 0.04762
   Detection Prevalence : 0.08390
      Balanced Accuracy : 0.63067

       'Positive' Class : Yes
</pre></td></tr>
<tr><th>RANDOM FOREST OUTPUT</th><th>SVM OUTPUT</th></tr>
<tr><td><pre>
Confusion Matrix and Statistics

          Reference
Prediction No Yes
       No 370 55
       Yes   2 14

               Accuracy : 0.8707
                 95% CI : (0.8358, 0.9006)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.06318

                  Kappa : 0.2874
 Mcnemar's Test P-Value : 5.675e-12

            Sensitivity : 0.20290
            Specificity : 0.99462
         Pos Pred Value : 0.87500
         Neg Pred Value : 0.87059
             Prevalence : 0.15646
         Detection Rate : 0.03175
   Detection Prevalence : 0.03628
      Balanced Accuracy : 0.59876

       'Positive' Class : Yes
</pre></td><td><pre>
Confusion Matrix and Statistics

          Reference
Prediction No Yes
       No 361 38
       Yes 11 31

               Accuracy : 0.8889
                 95% CI : (0.8558, 0.9167)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.0039921

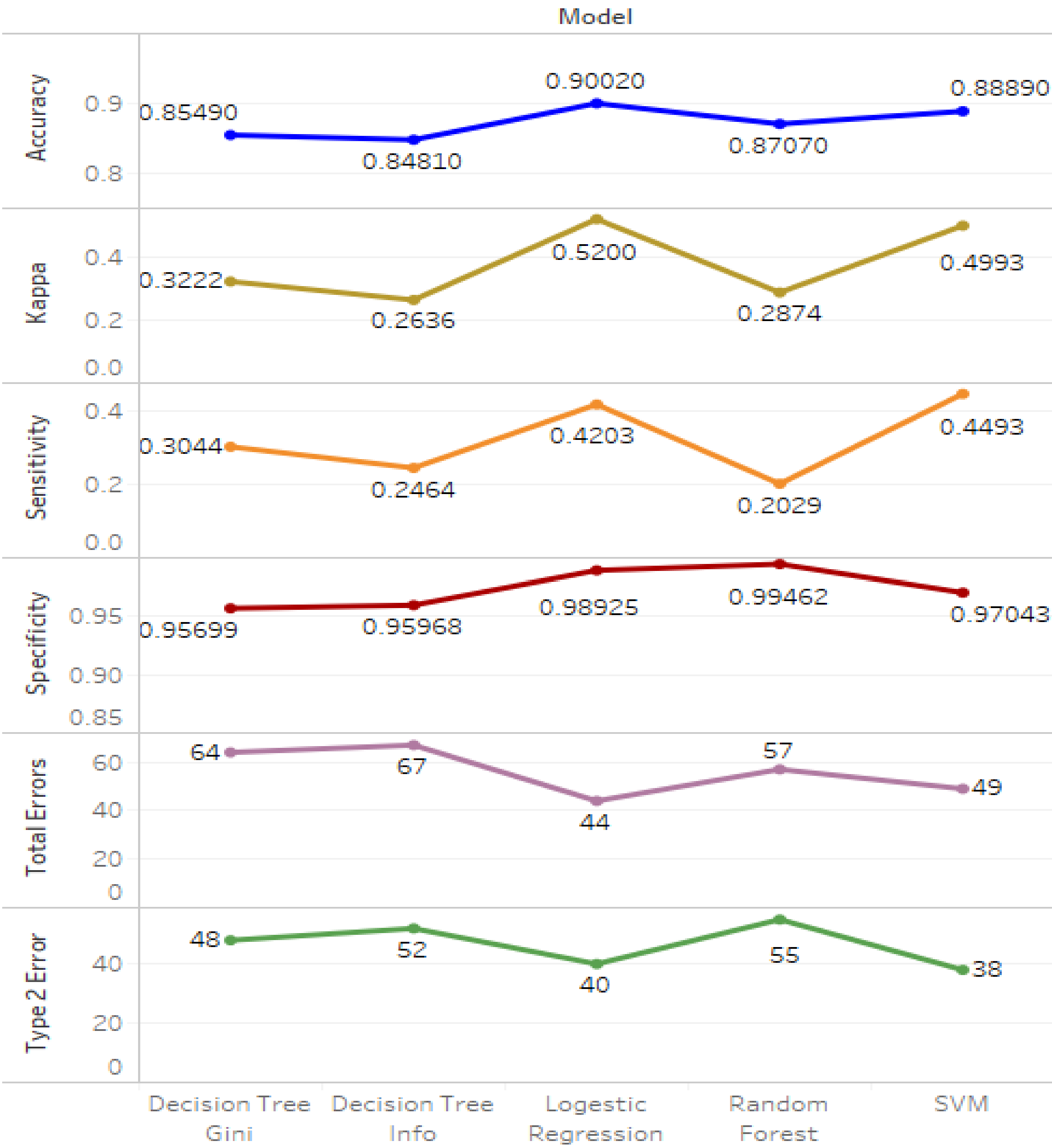
                  Kappa : 0.4993
 Mcnemar's Test P-Value : 0.0002038

            Sensitivity : 0.44928
            Specificity : 0.97043
         Pos Pred Value : 0.73810
         Neg Pred Value : 0.90476
             Prevalence : 0.15646
         Detection Rate : 0.07029
   Detection Prevalence : 0.09524
      Balanced Accuracy : 0.70985

       'Positive' Class : Yes
</pre></td></tr>
</table>

Model Accuracy Graph:

All Features & AFTER Data manipulation



Technique 2 : Conclusion

Top 2 Best Models as per Technique 2:

Technique#	**Best Model**	**Threshold**	**Accuracy**	**Kappa**	**Sensitivity**	**Specificity**
Technique 2	Logistic Regression (var=32)	0.56	0.9002	0.5200	0.4203	0.98925
Technique 2	SVM (var=32)	c=7, g=0.015	0.8889	0.4993	0.449275	0.97043

After, implementing Technique 2, now when we compare all the models we can conclude that the Logistic Regression model is giving the best **accuracy of 0.9002** among all.

If we compare this Logistic model with the previous one (Technique 1) then we see that the accuracy has improved marginally from **0.8912** to **0.9002** (increase of **0.009**) but there is a slight decrease of **0.0580** (approx. 5.8%) in Sensitivity (from **0.47826** to **0.42029**)

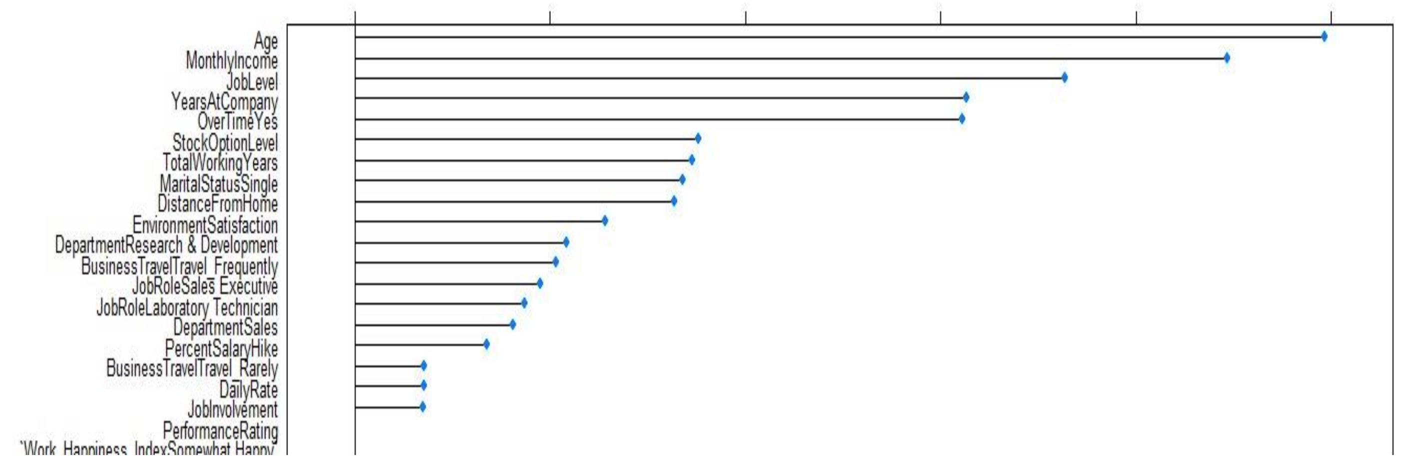
TECHNIQUE 3:

Model Building AFTER Data manipulation including only Important features

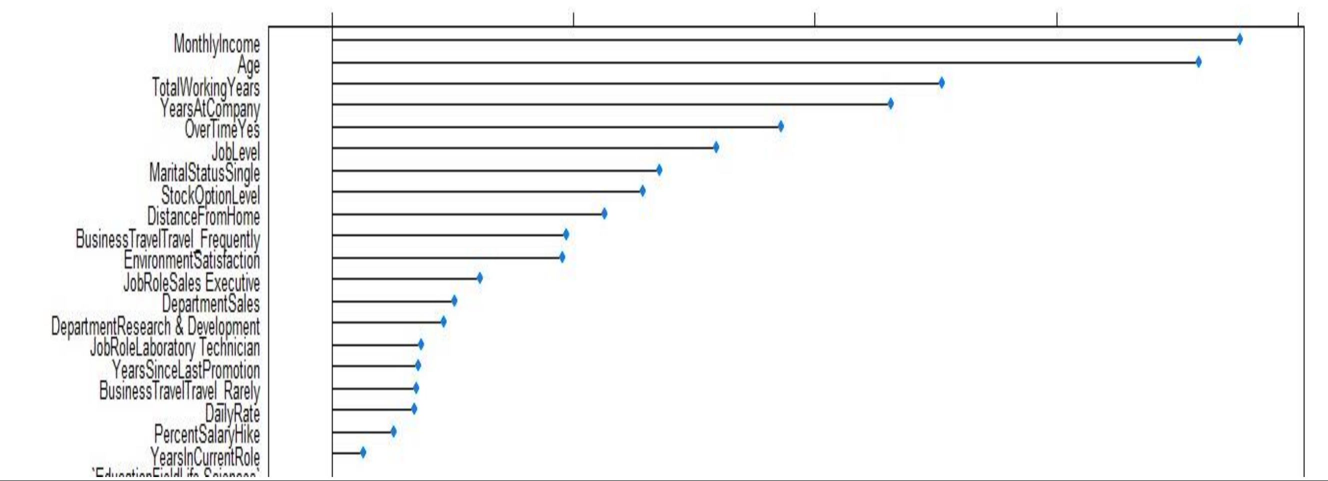
Important feature selection

- We used the VarImp() function in R for feature selection for Decision Tree, Random Forest and SVM.
- For Logistic we kept on iterating the model and eliminating the variables with high p-values until we had only features which had p-values greater than 0.05 or up to 0.1 if it affects the overall accuracy.

16 Important features as per Decision Tree model with Information Gain:

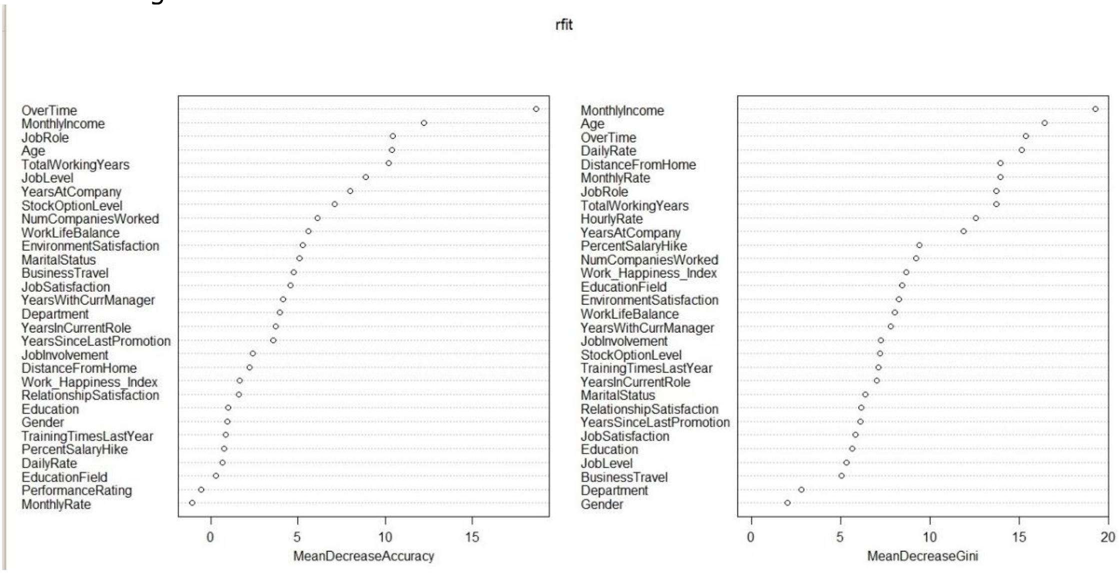


19 Important features as per Decision Tree model with Gini Index:



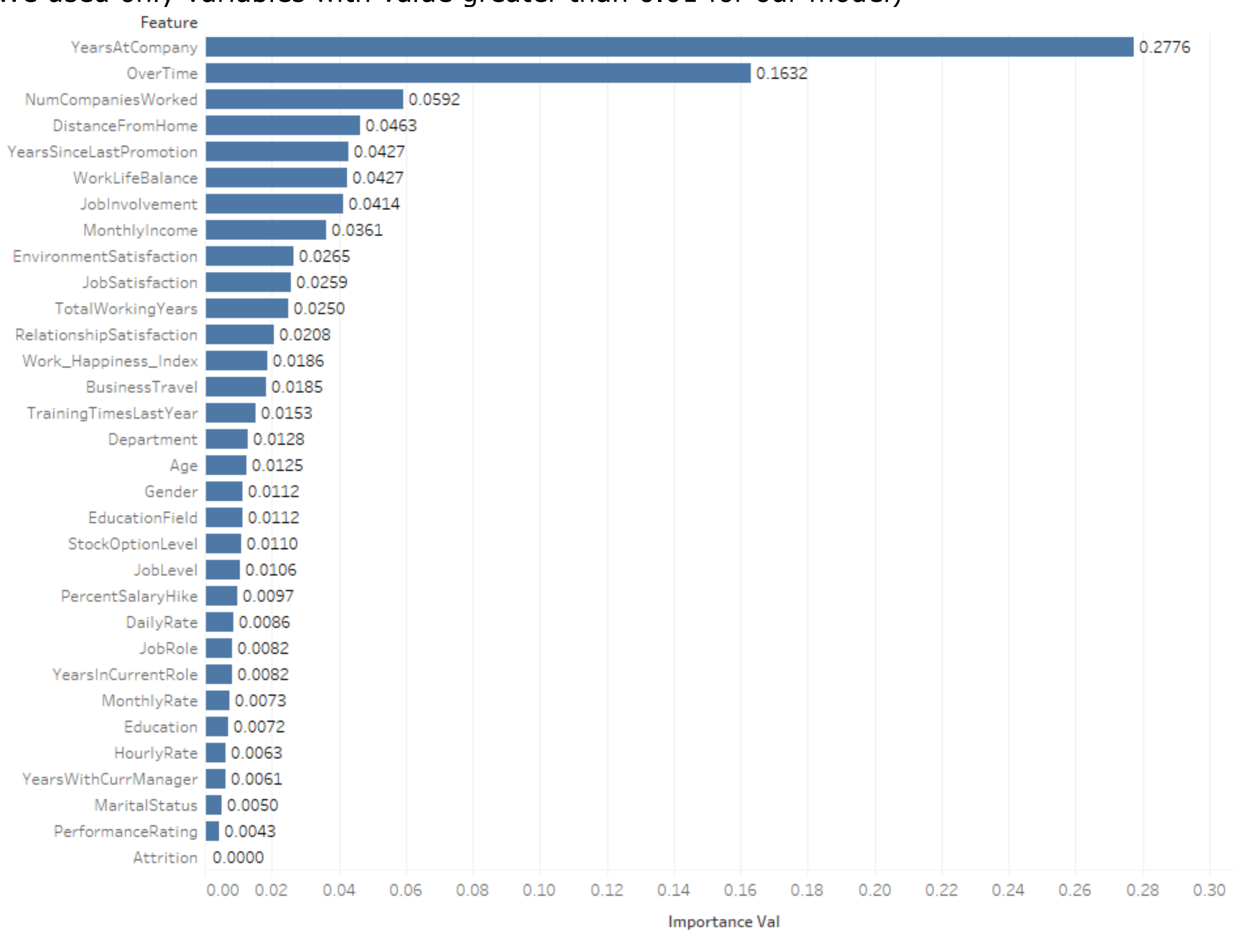
Important features as per Random forest model (acc vs Gini):

We used only variables with value greater than 0 for MeanDecreaseAccuracy and Variables with value greater than 9 for MeanDecreaseGini for our models.



Important features as per SVM model:

We used only variables with value greater than 0.01 for our model)



MODEL CREATION USING IMPORTANT VARIABLES

Below are the Models created selecting only important features:

Model#	**Algorithm used**	**Variable Count**	**Threshold Value**
MODEL 1	LOGISTIC REGRESSION	18	Any prediction value > **0.53,**"Yes","No"
MODEL 2	LOGISTIC REGRESSION – chisq	20	Any prediction value > **0.48,**"Yes","No"
MODEL 3	DECISION TREE WITH INFORMATION GAIN	16	tuneLength = 12
MODEL 4	DECISION TREE WITH GINI INDEX	19	tuneLength = 9
MODEL 5	RANDOM FOREST (Gini > 9)	14	ntree = 55
MODEL 6	SVM	21	cost = 20, gamma=0.005

Confusion and Accuracy: AFTER Data manipulation including only Important features

LOGISTIC REGRESSION WITH ANOVA OUTPUT

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  365  49
       Yes   7  20

               Accuracy : 0.873
                 95% CI : (0.8383, 0.9026)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.04779

                  Kappa : 0.3604
 Mcnemar's Test P-Value : 4.281e-08

            Sensitivity : 0.28986
            Specificity : 0.98118
         Pos Pred Value : 0.74074
         Neg Pred Value : 0.88164
             Prevalence : 0.15646
         Detection Rate : 0.04535
   Detection Prevalence : 0.06122
      Balanced Accuracy : 0.63552

       'Positive' Class : Yes
```

LOGISTIC REGRESSION WITH CHISQ OUTPUT

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  363  42
       Yes   9  27

               Accuracy : 0.8844
                 95% CI : (0.8508, 0.9127)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.00896

                  Kappa : 0.4559
 Mcnemar's Test P-Value : 7.433e-06

            Sensitivity : 0.39130
            Specificity : 0.97581
         Pos Pred Value : 0.75000
         Neg Pred Value : 0.89630
             Prevalence : 0.15646
         Detection Rate : 0.06122
   Detection Prevalence : 0.08163
      Balanced Accuracy : 0.68356

       'Positive' Class : Yes
```

DECISION TREE WITH INFO GAIN OUTPUT

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  357  52
       Yes  15  17

               Accuracy : 0.8481
                 95% CI : (0.8111, 0.8803)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.4278

                  Kappa : 0.2636
 Mcnemar's Test P-Value : 1.092e-05

            Sensitivity : 0.24638
            Specificity : 0.95968
         Pos Pred Value : 0.53125
         Neg Pred Value : 0.87286
             Prevalence : 0.15646
         Detection Rate : 0.03855
   Detection Prevalence : 0.07256
      Balanced Accuracy : 0.60303

       'Positive' Class : Yes
```

DECISION TREE WITH GINI INDEX OUTPUT

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  356  48
       Yes  16  21

               Accuracy : 0.8549
                 95% CI : (0.8185, 0.8864)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.2809805

                  Kappa : 0.3222
 Mcnemar's Test P-Value : 0.0001066

            Sensitivity : 0.30435
            Specificity : 0.95699
         Pos Pred Value : 0.56757
         Neg Pred Value : 0.88119
             Prevalence : 0.15646
         Detection Rate : 0.04762
   Detection Prevalence : 0.08390
      Balanced Accuracy : 0.63067

       'Positive' Class : Yes
```

RANDOM FOREST Gini > 9

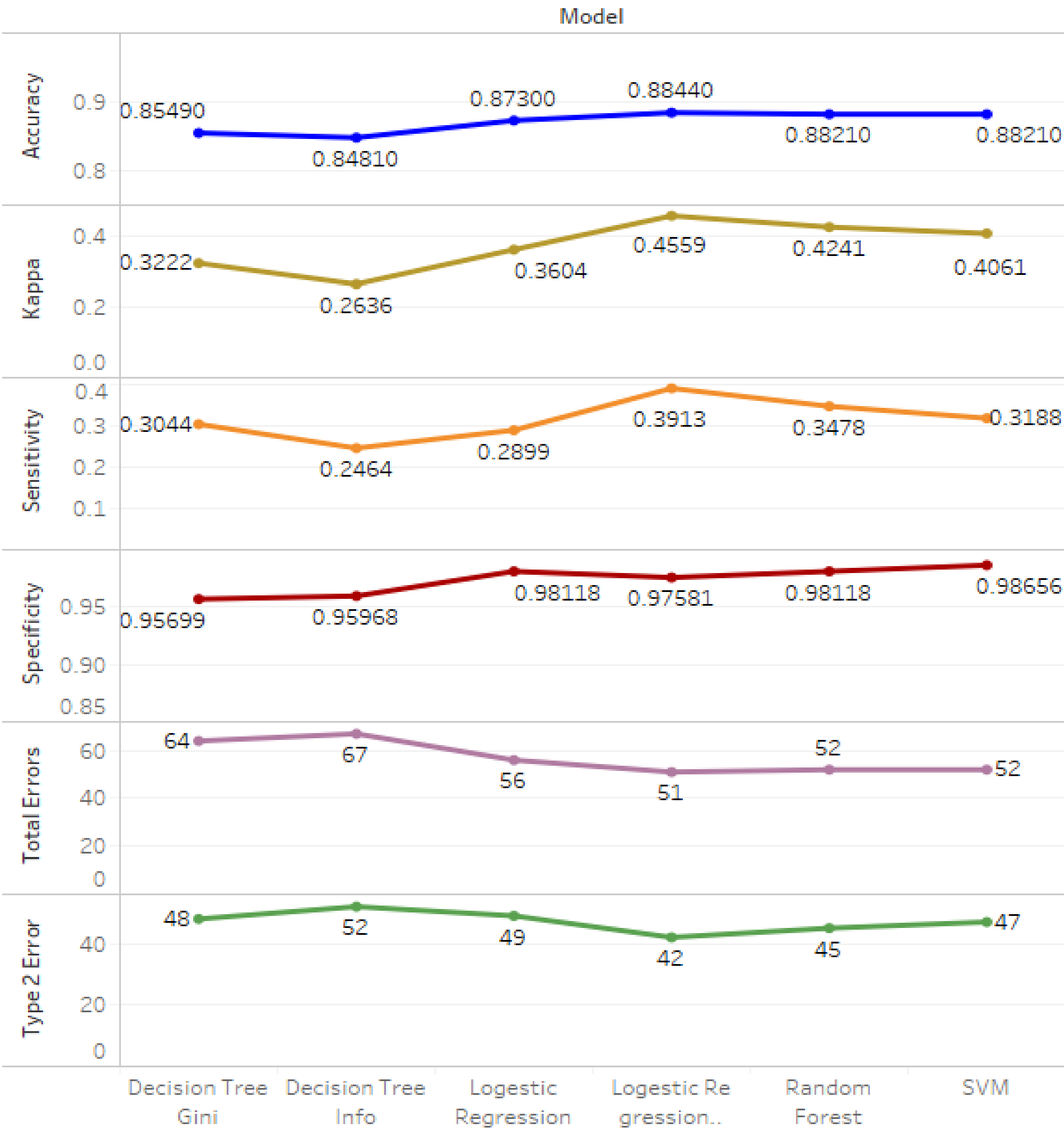
```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  365  45
       Yes   7  24

               Accuracy : 0.8821
                 95% CI : (0.8483, 0.9107)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.01302

                  Kappa : 0.4241
 Mcnemar's Test P-Value : 2.882e-07

            Sensitivity : 0.34783
            Specificity : 0.98118
         Pos Pred Value : 0.77419
         Neg Pred Value : 0.89024
             Prevalence : 0.15646
         Detection Rate : 0.05442
   Detection Prevalence : 0.07029
      Balanced Accuracy : 0.66450

       'Positive' Class : Yes
```

SVM

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  367  47
       Yes   5  22

               Accuracy : 0.8821
                 95% CI : (0.8483, 0.9107)
    No Information Rate : 0.8435
    P-Value [Acc > NIR] : 0.01302

                  Kappa : 0.4061
 Mcnemar's Test P-Value : 1.303e-08

            Sensitivity : 0.31884
            Specificity : 0.98656
         Pos Pred Value : 0.81481
         Neg Pred Value : 0.88647
             Prevalence : 0.15646
         Detection Rate : 0.04989
   Detection Prevalence : 0.06122
      Balanced Accuracy : 0.65270

       'Positive' Class : Yes
```

Model Accuracy Graph:

AFTER Data manipulation including only important features



Conclusion:

Top 2 Best Models as per Technique 3

Technique#	Best Model	Threshold	Accuracy	Kappa	Sensitivity	Specificity
Technique 3	Logistic Regression (var=20)	0.48	0.8844	0.45591	0.39130	0.97581
Technique 3	Random Forest (gini>9) (var=14)	ntree=55	0.8821	0.4241	0.34783	0.98118

After, implementing Technique 3, now when we compare all the models we can conclude that the Logistic Regression model is giving the best **accuracy of 0.8844** among all.

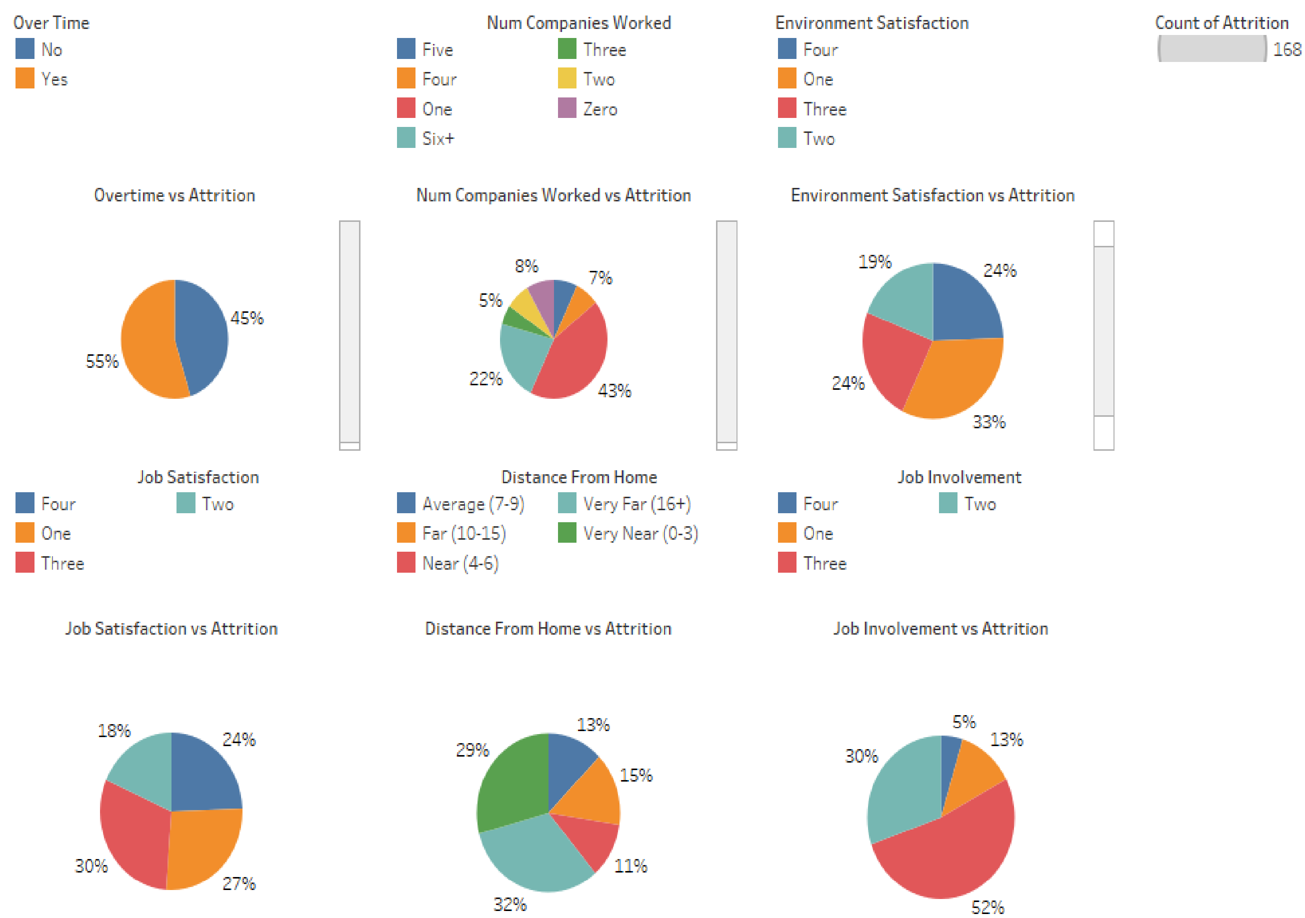
Final Conclusion:

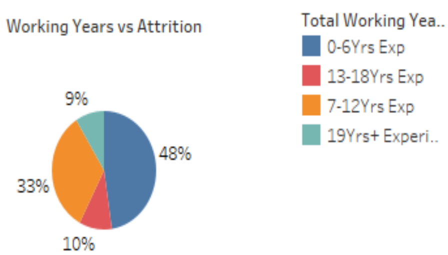
Objective 1: Top 10 important features for attrition:

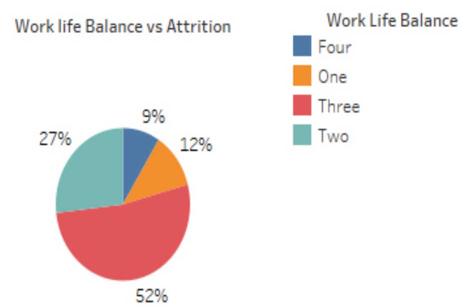
Below are the top 10 important features from the output using the VarImp function on the various models:

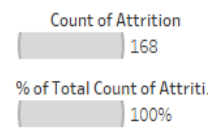
1. Overtime
2. Number of Companies worked
3. Environmental Satisfaction
4. Distance from Home
5. Job Satisfaction
6. Job Involvement
7. Total Working Years
8. Work Life Balance
9. Business Travel
10. Age

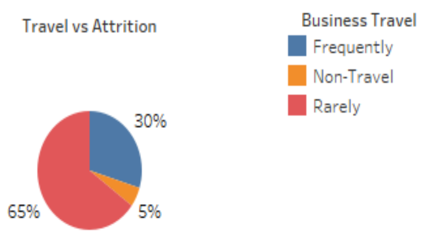
Find below the charts showing the relationship between the Top 10 features and Attrition:

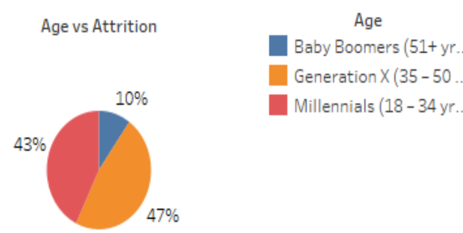



Working Years vs Attrition
Total Working Yea..
0-6Yrs Exp
13-18Yrs Exp
7-12Yrs Exp
19Yrs+ Experi..
9%
48%
33%
10%


Work life Balance vs Attrition
Work Life Balance
Four
One
Three
Two
9%
27%
12%
52%


Count of Attrition
168
% of Total Count of Attriti..
100%


Travel vs Attrition
Business Travel
Frequently
Non-Travel
Rarely
30%
65%
5%


Age vs Attrition
Age
Baby Boomers (51+ yr..
Generation X (35 – 50 ..
Millennials (18 – 34 yr..
10%
43%
47%

OBJECTIVE 2:

Creating the best predictive model:

Below is the list of all the Best models based on Techniques 1 to 3

Technique#	Best Model	Threshold	Accuracy	Kappa	Sensitivity	Specificity
Technique 1	Logistic Regression (var=32)	0.51	0.8912	0.5196	0.47826	0.96774
Technique 1	SVM (var=32)	c=9, g=0.015	0.8912	0.5196	0.47826	0.96774
Technique 2	Logistic Regression (var=32)	0.56	0.9002	0.5200	0.42029	0.98924
Technique 3	Logistic Regression (var=20)	0.48	0.8844	0.45591	0.39130	0.97581

If you have a look at the above table then you can see that we were able to improve the accuracy as well as the Kappa Value when we created Logistic models comparing all features and after imputing the outliers as well as scaling (Technique 2) when compared to Technique 1

Best Model after all 3 Techniques is:

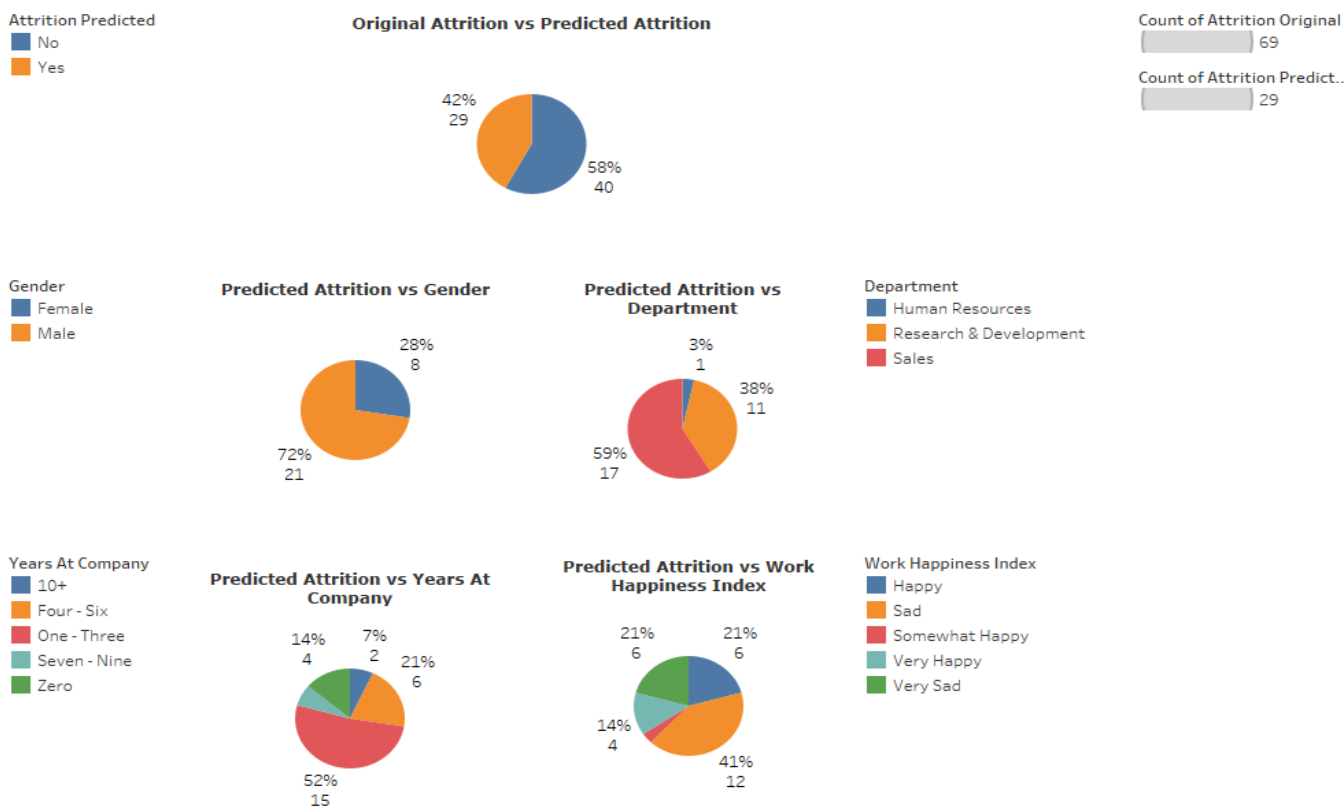
Technique#	Best Model	Threshold	Accuracy	Kappa	Sensitivity	Specificity
Technique 2	Logistic Regression (var=32)	0.56	0.9002	0.5200	0.42029	0.98924

Significant findings from Best Logistic Regression Model:

Based on the Logistic Model Predicting we see that we were able to make 29 correct predictions which is 42% when compared to expected prediction. Summary of the findings from predictions is mentioned below:

1. Possibility of Attrition is high in **Males** than Females.
2. **Sales Department** has the highest Attrition rate.
3. Possibility of Attrition is high in **One-Three Year** old employees.
4. **Sad** employees (who rated 1 or 2 for Job Satisfaction along with 3 or 4 for either one or both Environment Satisfaction and Relationship Satisfaction) are more likely to leave the organization.

Significance Attrition findings from Logistic Predictive Model



A BIG THANK YOU TO ALL THE TEACHERS AND SUPPORT STAFF OF IMARTICUS LEARNING