```
#Business Analytics with R - BUAN6356.001



#Group Project - Human Resources analytics for
employee attrition



#Instructor: Professor Ling Ge


#Group: 2
  #Ha Minh Huy Truong
  #Sameera Epari
  #Shubham Kothari
  #Nikhil Shah
  #Naveen Kumar Palaniappan
  #Shwana Ali
```

#Table of content

#1.Executive summary ---------- 1

#2.Project motivation ---------- 1

#3.Data description ---------- 1

#4.Exploratory data analysis ---------- 4
  #4.1.Data preparation ---------- 4
    #4.1.1.Data importation ---------- 4
    #4.1.2.Missing value check ---------- 4
    #4.1.3.Unwanted variables removal ---------- 4
    #4.1.4.Data type transformation ---------- 5
  #4.2.Data exploration ---------- 5
    #4.2.1.Attrition variable distribution ---------- 5
    #4.2.2.Relationship between Attrition and Remuneration ---------- 6
    #4.2.3.Relationship between Attrition and Job ---------- 8
    #4.2.4.Relationship between Attrition and Work Load ---------- 12
    #4.2.5.Relationship between Attrition and Location ---------- 13
    #4.2.6.Relationship between Attrition and Privilege ---------- 14

#5.Model and analysis ---------- 15
  #5.1.Data split ---------- 15
  #5.2.Variable selection ---------- 18
  #5.3.Modeling ---------- 19
  #5.4.Imbalance issue ---------- 22
  #5.5.Modeling after handling imbalance issue ---------- 23
  #5.6.Model selection ---------- 26

#6.Findings and managerial implications ---------- 28

#7.Conclusions ---------- 28

```
#1.Executive summary

#The aim of this project is to perform data exploration and build models to p
redict whether an employee will stay or leave the company. We have IBM attrit
ion data set to perform this task. Before building the model, we check the nu
ll values, transform data types and select variables. After that, we build mo
dels and decide the best one based on accuracy and sensitivity since our focu
s is to check which employee is likely to leave the company. Additionally, we
use ROC plot to confirm our decision on model selection. At the end of this p
roject, we get to know what are the important variables impacting the attriti
on so that the company can take actions to reduce the attrition rate.



#2.Project motivation

#A recruiter’s role has evolved over the years. Earlier, they were only respo
nsible for filling up vacancies arising out of a new role or requirement, or
an employee moving out. In the current context, a recruiter has a bigger role
to play. Organizations now understand that their biggest asset is their workf
orce, which explains why there is an increasing emphasis on finding and attra
cting the best talent.  This growing competition for talent is also a result
of rising candidate awareness. High-performance employees understand they are
a valuable asset and therefore perform comprehensive analysis before committi
ng to a particular organization.

#Job search is no longer a tedious process. In this smart age, job seekers ar
e constantly checking out newer and better opportunities on the go, or even,
while at work. Thus, the role of a recruiter, hiring manager, or talent acqui
sition specialist has grown manifold. From filling up vacant positions to fin
ding and hiring top talent, to retaining trained workforce, it’s a tough task
at hand. And HR analytics can help businesses make smarter decisions.



#3.Data description

#Uncover the factors that lead to employee attrition and explore important qu
estions such as ‘show me a breakdown of distance from home by job role and at
trition’ or ‘compare average monthly income by education and attrition’. This
is a fictional data set created by IBM data scientists.
```

Variable	Description	Values
Age	Age of the Employee	
Attrition	If a person has left the organization or not	“yes” or “no”

Business Travel	Travel is undertaken for work or business purposes	”Non-Travel” or “Travel-Frequently” or “Travel-Rarely”
Daily Rate	Rate/cost of an employee for an entire workday	
Department	The departments in which the employees work	“Sales”, “Research”, development” and “Human Resources”
Distance	The number of miles an employee has to travel to get to work	
Education	Level of study	1 'Below College' 2 'College' 3 'Bachelor' 4 'Master' 5 'Doctor'
Education Field	The field of study	
Employee Count	The employee count is the number of employees	
Employee Number	It is the ID of the employee	
Environment Satisfaction	The employee satisfaction level about the work environment in the organization	1: 'Low', 2: 'Medium', 3: 'High', 4: 'Very High'
Gender	Gender of the Employee	“Male” or “Female”
Hourly Rate	Amount of money that is paid or earned for every hour worked	
Job Involvement	It refers to the psychological and emotional extent to which employee participates in his/her work, profession, and company	'Low', 2: 'Medium', 3: 'High', 4: 'Very High'
Job Level	Level of the job containing ordinal values	1: 'Low', 2: 'Medium', 3: 'High', 4: 'Highest’
Job Role	Type of the job role of the employee	
Job Satisfaction	Level of Satisfaction of the employee	1: 'Low', 2: 'Medium', 3: 'High', 4: 'Very High'

Marital Status	Marital Status of the employee	“Single” or “Married” or “Divorced”
Monthly Income	The monthly wage of an employee	
Monthly Rate	Amount of money that is paid or earned for every month worked by the employee	
Num Companies Worked	The number of companies the employee has worked prior to joining the organization	
Over 18	If the employee is above 18 or not	“Y” or “N”
Over Time	If the employee works overtime or not	“Yes” or “No”
Percentage Salary Hike	Percentage of the previous hike of the employee.	
Performance Rating	Rating of employee’s performance	1 'Low' 2 'Good' 3 'Excellent' 4 'Outstanding'
Relationship Satisfaction	Employee satisfaction for the professional relationships within the organization	1: 'Low', 2: 'Medium', 3: 'High', 4: 'Very High'
Standard Hours	Working hours of the employee	
Stock Options Level	Level of the stock option provided by the organization to the employee containing ordinal values	1: 'Low', 2: 'Medium', 3: 'High', 4: 'Highest’
Total Working Years	Overall working experience of the employee	
Training Times Last Year	Number of times training has been provided to the employee in the previous year	
Work-Life Balance	Describes the balance that a working individual needs between time	1: 'Low', 2: 'Medium', 3: 'High', 4: 'Very High'
Years At Company	Number of years the employee has been at the company	

Years in Current Role	*Number of years employee has worked in the current role*	
Years Since Promotion	*Number of years since employee received last promotion*	
Years with Curr Manager	*Number of years working with the manager*	

```
#4.Exploratory Data Analysis

#4.1.Data preparation
#4.1.1.Data importation
library(tidyverse)
```

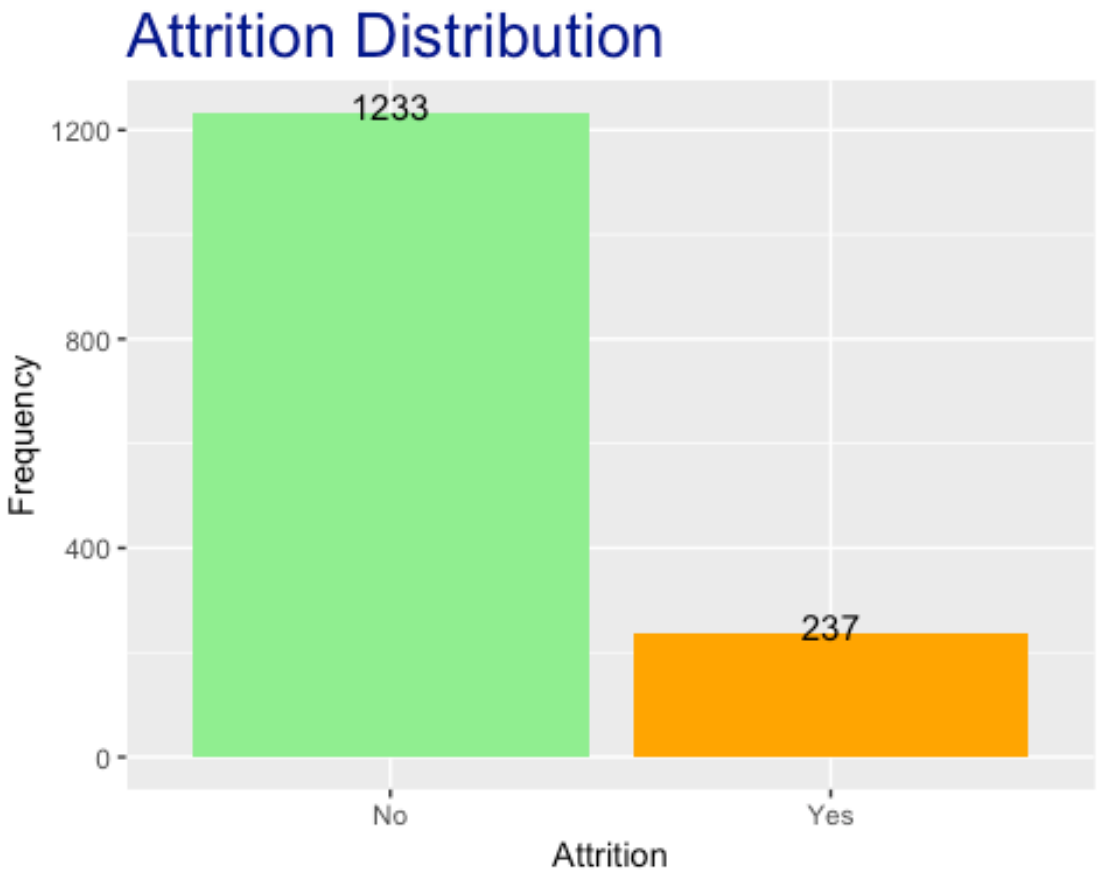
```
## ── Attaching packages ─────────────────────────────────────
──────────── tidyverse 1.2.1 ──

## ✓ ggplot2 3.1.0     ✓ purrr   0.2.5
## ✓ tibble  2.0.1     ✓ dplyr   0.7.8
## ✓ tidyr   0.8.2     ✓ stringr 1.3.1
## ✓ readr   1.3.1     ✓ forcats 0.3.0

## ── Conflicts ────────────────────────────────────────────────
──── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library(dplyr)
library(ggplot2)
myfile <- read.csv('WA_Fn-UseC_-HR-Employee-Attrition.csv')

#4.1.2.Missing value check
#Dataset is clean, there is no missing value.
colnames(myfile)[colSums(is.na(myfile))>0]
```

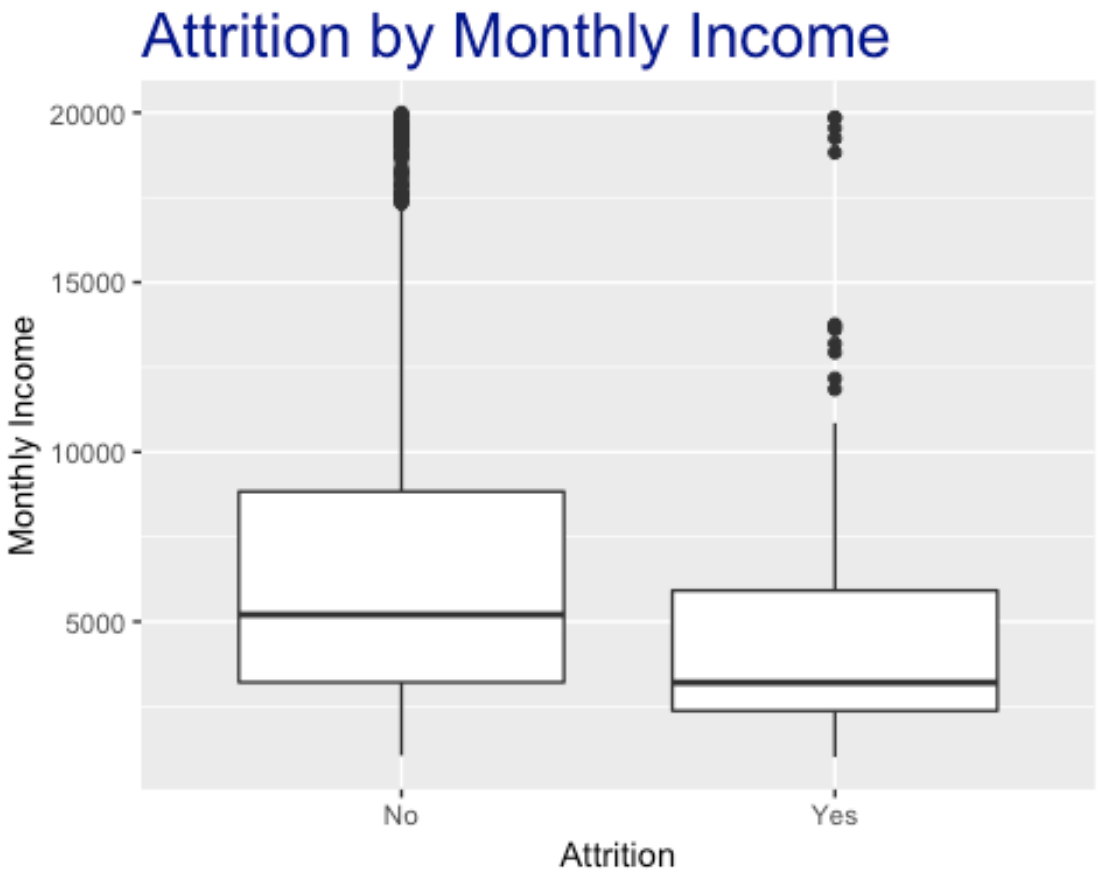
```
## character(0)
```

```
#4.1.3.Unwanted variables removal
#We found that there is EmployeeNumber variable which is in numeric form and
which is not necessary for us to run model because it is actually employee ID
s.
#In addition, there are three variables (EmployeeCount, StandardHours, and Ov
er18) in which there is only one same value for across observations. For exam
ple, all observations have the same value which is "1" at EmployeeCount varia
ble.
#Since those variables are not meaningful for our model, we remove them from
dataset.
```

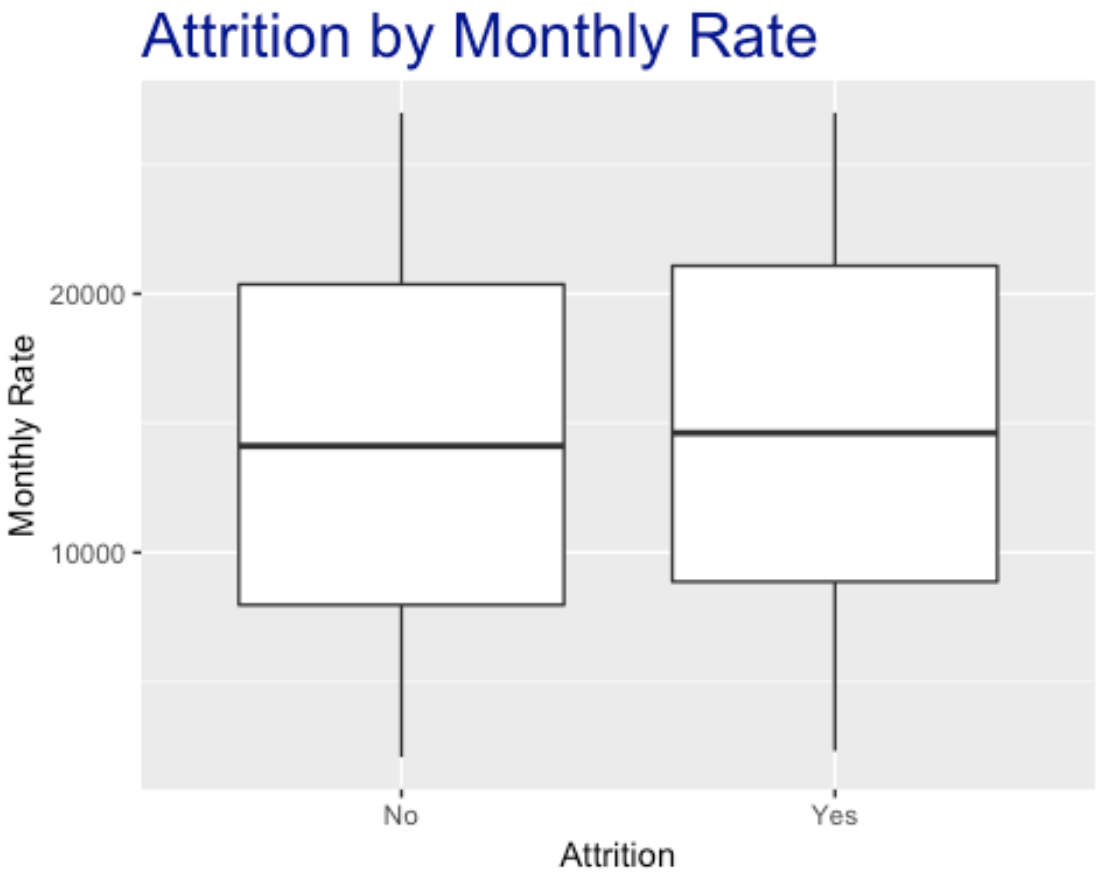
```r
myfile$EmployeeNumber <- NULL
myfile$EmployeeCount <- NULL
myfile$StandardHours <- NULL
myfile$Over18 <- NULL

#4.1.4.Data type transformation
#There are many categorical variables which are in numeric type. For example, Education variable describes education levels of an employee, such as "College", "Bachelor", "Master", etc. Therefore, this variable should be categorical, not numeric one.
#In total, there are such ten variables: Education, EnvinromentSatisfaction, JobInvolvement, JobLevel, JobStatisfaction, PerformanceRating, RelationshipSatisfaction, StockOptionLevel, TrainingTimesLastYear, and WorkLifeBalance.
#Therefore, we transform those variables to categorical ones by using as.factor()
myfile$Education <- as.factor(myfile$Education)
myfile$EnvironmentSatisfaction <- as.factor(myfile$EnvironmentSatisfaction)
myfile$JobInvolvement <- as.factor(myfile$JobInvolvement)
myfile$JobLevel <- as.factor(myfile$JobLevel)
myfile$JobSatisfaction <- as.factor(myfile$JobSatisfaction)
myfile$PerformanceRating <- as.factor(myfile$PerformanceRating)
myfile$RelationshipSatisfaction <- as.factor(myfile$RelationshipSatisfaction)
myfile$StockOptionLevel <- as.factor(myfile$StockOptionLevel)
myfile$TrainingTimesLastYear <- as.factor(myfile$TrainingTimesLastYear)
myfile$WorkLifeBalance <- as.factor(myfile$WorkLifeBalance)

#4.2.Data exploration
#4.2.1.Attrition variable distribution
#Reponse data (Attribution) is imbalanced between two classes: 1233 "No" vs. 237 "Yes". Therefore, this may bias our prediction. In other words, our prediction result will have more "No" than "Yes".
#We may consider SMOTE to balance it.
ggplot(data=myfile, aes(x=Attrition))+
  geom_bar(fill=c('LightGreen','Orange'))+
  geom_text(aes(label=..count..), stat='count', vjust=0.2, size=4)+
  ggtitle('Attrition Distribution')+
  scale_y_continuous(name='Frequency')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```



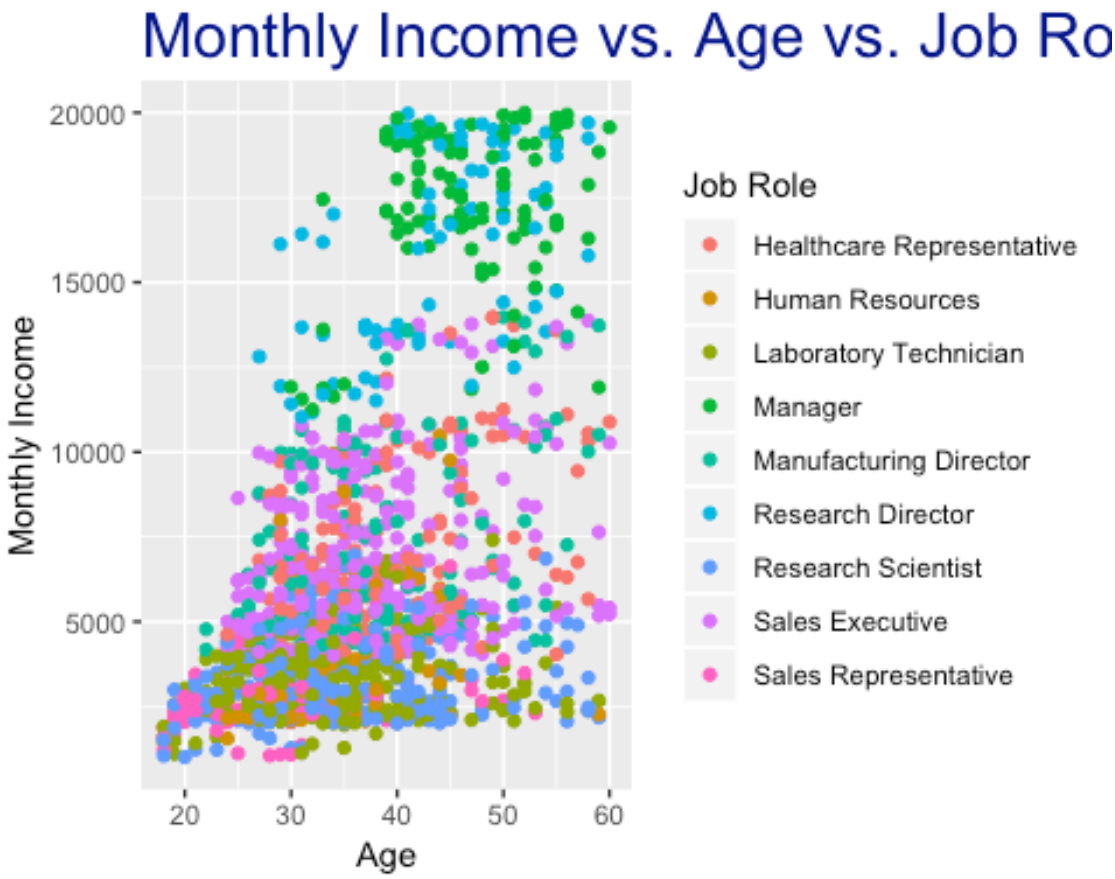

```
#4.2.2.Relationship between Attrition and Remuneration (MonthlyIncome and Mon
thlyRate)
#Employees with low monthly income have a higher attrition rate, compared to
those with high monthly income.
#However, regarding monthly rate, there is no difference in attrition rate be
tween two groups of employees who resigned and who stay with the company.
#Therefore, we can see that, no matter what pay rate employees can earn, thei
r concern is actual income in each month. This remark needs to be taken in co
nsideration when we want to make any change regarding remuneration.
ggplot(data=myfile)+
  geom_boxplot(aes(x=Attrition, y=MonthlyIncome))+
  ggtitle('Attrition by Monthly Income')+
  scale_y_continuous(name='Monthly Income')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```




```
ggplot(data=myfile)+
  geom_boxplot(aes(x=Attrition, y=MonthlyRate))+
  ggtitle('Attrition by Monthly Rate')+
  scale_y_continuous(name='Monthly Rate')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```




```
#4.2.3.Relationship between Attrition and Job (JobRole and JobLevel)
#Senior level such as Directors and Managers has higher age and also has high
er monthly income. This group of employees tend to stay in the company. Compa
red to senior level, junior level such as Laboratory Technicians, Sales Repre
sentatives, Hunan Resources is younger but has lower monthly income. The latt
er group actually contributes to high attrition rate of the company.
#When we split attrition rate among job level, the result is consistent with
our findings from previous part. Employees with low job levels (level 1, 2, 3
) have a higher attrition rate, compared to those with higher job levels (lev
el 4 and 5). Again, low job levels are actually junior level, while high job
levels are senior level.
ggplot(data=myfile)+
  geom_point(aes(x=Age, y=MonthlyIncome, color=JobRole))+
  ggtitle('Monthly Income vs. Age vs. Job Role')+
  scale_y_continuous(name='Monthly Income')+
  scale_color_discrete(name='Job Role')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```

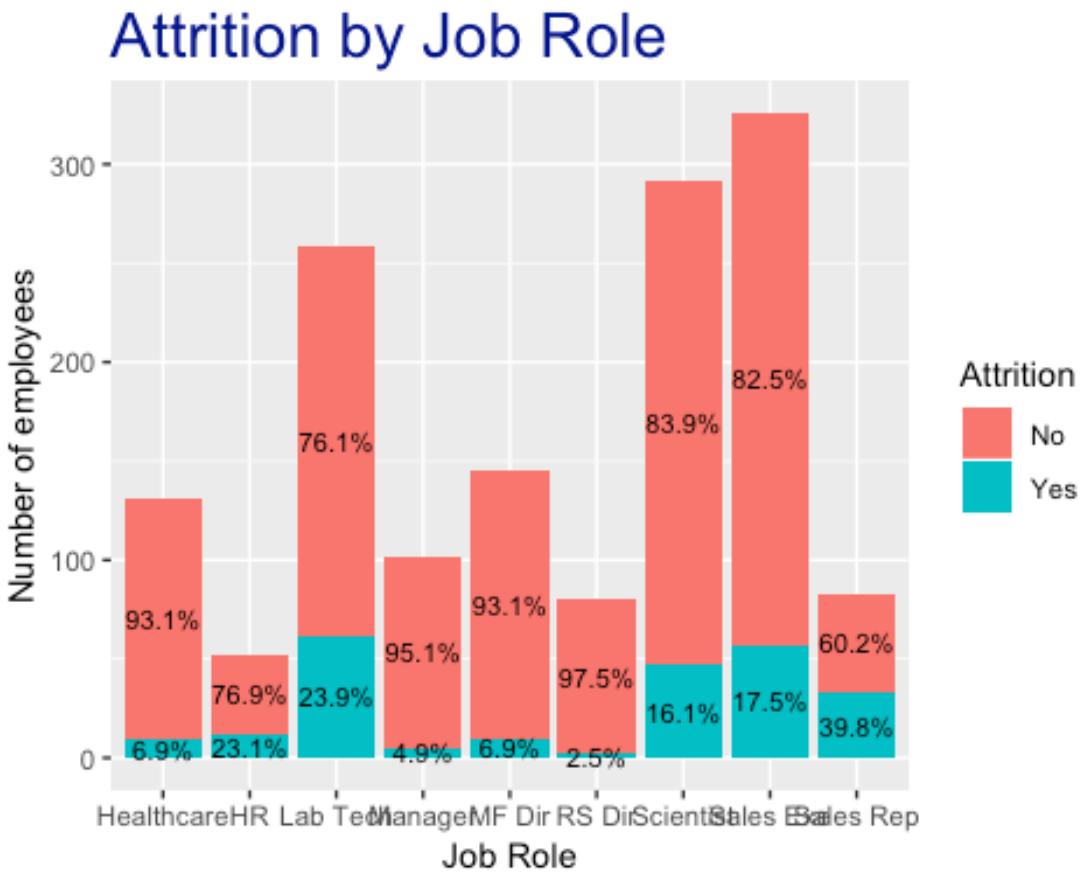



```
jobrole <- myfile%>%
  group_by(JobRole)%>%
  count(Attrition)%>%
  mutate(AttritionRate=scales::percent(n/sum(n)))
jobrole
```

```
## # A tibble: 18 x 4
## # Groups:   JobRole [9]
##    JobRole                   Attrition     n AttritionRate
##    <fct>                     <fct>     <int> <chr>
##  1 Healthcare Representative No          122 93.1%
##  2 Healthcare Representative Yes           9 6.9%
##  3 Human Resources           No           40 76.9%
##  4 Human Resources           Yes          12 23.1%
##  5 Laboratory Technician     No          197 76.1%
##  6 Laboratory Technician     Yes          62 23.9%
##  7 Manager                   No           97 95.1%
##  8 Manager                   Yes           5 4.9%
##  9 Manufacturing Director    No          135 93.1%
## 10 Manufacturing Director    Yes          10 6.9%
## 11 Research Director         No           78 97.5%
## 12 Research Director         Yes           2 2.5%
## 13 Research Scientist        No          245 83.9%
```

```
## 14 Research Scientist        Yes          47 16.1%
## 15 Sales Executive           No          269 82.5%
## 16 Sales Executive           Yes          57 17.5%
## 17 Sales Representative      No           50 60.2%
## 18 Sales Representative      Yes          33 39.8%
```

```
ggplot(data=myfile, aes(x=JobRole, fill=Attrition))+
  geom_bar()+
  geom_text(data=jobrole, aes(y=n,label=AttritionRate), position=position_stack(vjust=0.5), size=3)+
  ggtitle('Attrition by Job Role')+
  scale_x_discrete(name='Job Role', label=c('Healthcare','HR','Lab Tech','Manager','MF Dir','RS Dir','Scientist','Sales Exe','Sales Rep'))+
  scale_y_continuous(name='Number of employees')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```

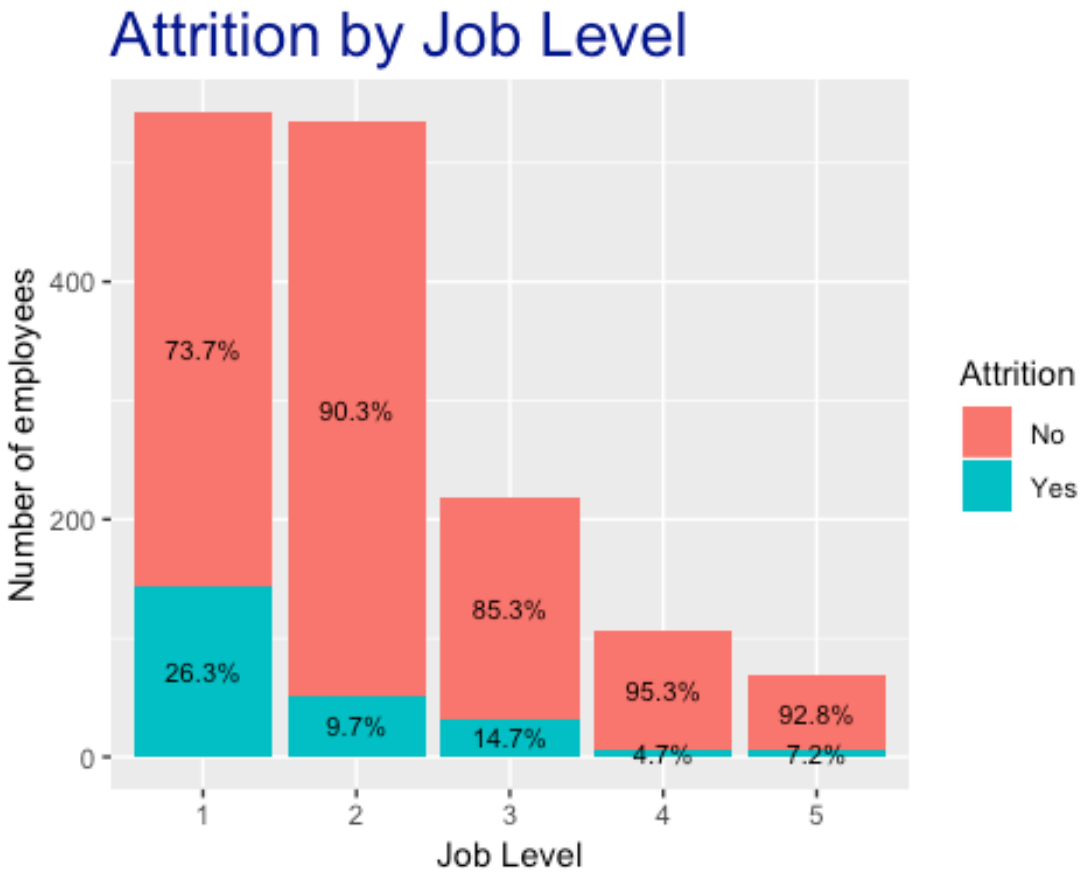



```
level <- myfile%>%
  group_by(JobLevel)%>%
  count(Attrition)%>%
  mutate(AttritionRate=scales::percent(n/sum(n)))
level
```

```
## # A tibble: 10 x 4
## # Groups:   JobLevel [5]
```

```
##    JobLevel Attrition     n AttritionRate
##    <fct>    <fct>     <int> <chr>
##  1 1        No          400 73.7%
##  2 1        Yes         143 26.3%
##  3 2        No          482 90.3%
##  4 2        Yes          52 9.7%
##  5 3        No          186 85.3%
##  6 3        Yes          32 14.7%
##  7 4        No          101 95.3%
##  8 4        Yes           5 4.7%
##  9 5        No           64 92.8%
## 10 5        Yes           5 7.2%
```

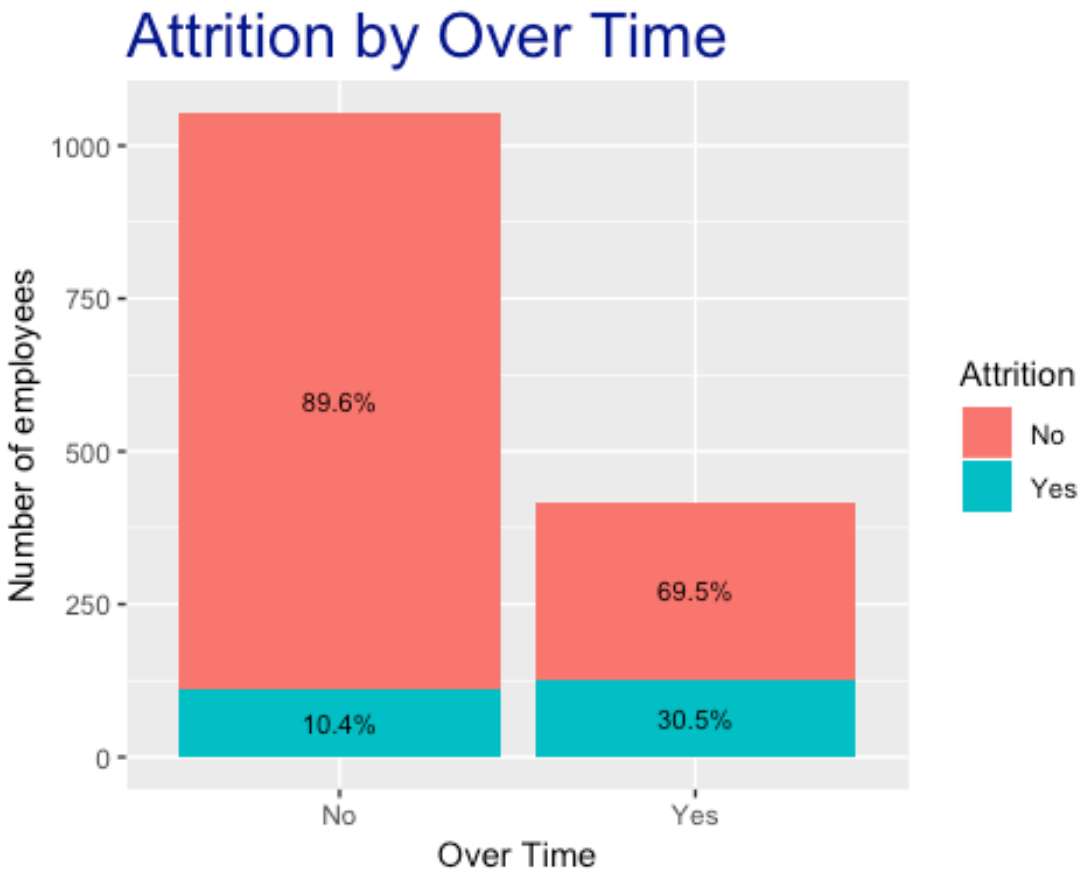
```
ggplot(data=myfile, aes(x=JobLevel, fill=Attrition))+
  geom_bar()+
  geom_text(data=level, aes(y=n,label=AttritionRate), position=position_stack
(vjust=0.5), size=3)+
  ggtitle('Attrition by Job Level')+
  scale_x_discrete(name='Job Level')+
  scale_y_continuous(name='Number of employees')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```



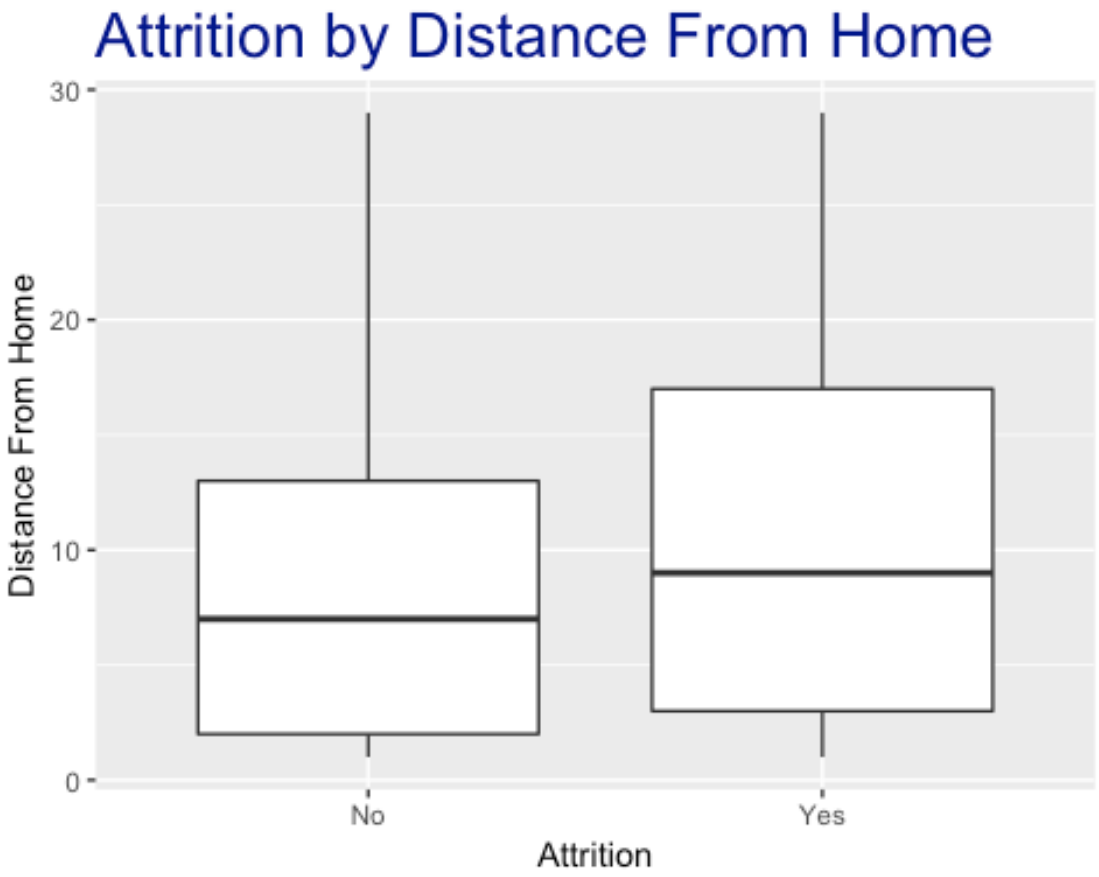
```r
#4.2.4.Relationship between Attrition and Work Load (OverTime)
#It makes sense that employees who have high work load which requires them to work overtime tend to have higher attrition rate. Meanwhile, for those who do not need to do overtime, they tend to stay with the company.
ot <- myfile%>%
  group_by(OverTime)%>%
  count(Attrition)%>%
  mutate(AttritionRate=scales::percent(n/sum(n)))
ot
```

```
## # A tibble: 4 x 4
## # Groups:   OverTime [2]
##   OverTime Attrition     n AttritionRate
##   <fct>    <fct>     <int> <chr>
## 1 No       No          944 89.6%
## 2 No       Yes         110 10.4%
## 3 Yes      No          289 69.5%
## 4 Yes      Yes         127 30.5%
```

```r
ggplot(data=myfile, aes(x=OverTime, fill=Attrition))+
  geom_bar()+
  geom_text(data=ot, aes(y=n,label=AttritionRate), position=position_stack(vjust=0.5), size=3)+
  ggtitle('Attrition by Over Time')+
  scale_x_discrete(name='Over Time')+
  scale_y_continuous(name='Number of employees')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```



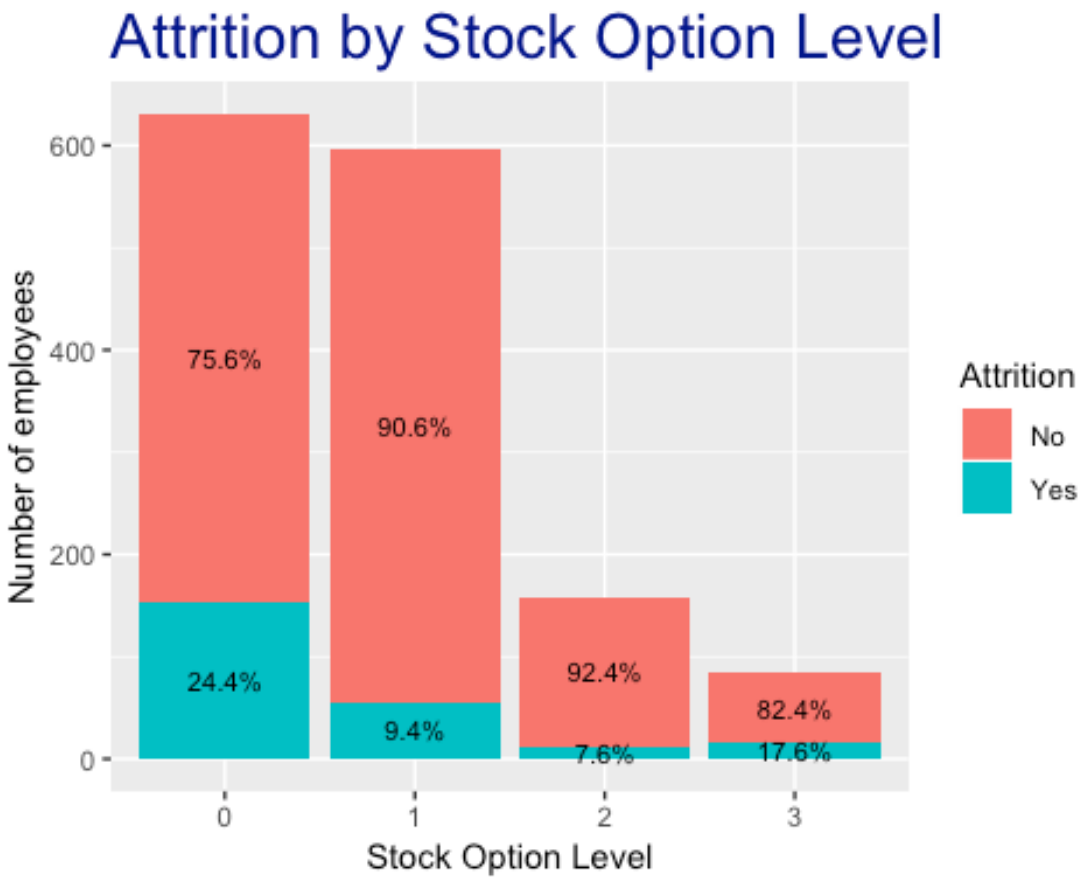

```
#4.2.5.Relationship between Attrition and Location (DistanceFromHome)
#Employees who live far from office tend to have a higher attrition rate, com
pared to those who live nearby office area. It turns out that office location
ggplot(data=myfile)+
  geom_boxplot(aes(x=Attrition, y=DistanceFromHome))+
  ggtitle('Attrition by Distance From Home')+
  scale_y_continuous(name='Distance From Home')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```




```
#4.2.6.Relationship between Attrition and Privilege (StockOptionLevel)
#It is signaled that stock level is one of important factors which impact att
rition rate. Employees who have higher stock level tend to be more engaged an
d to stay with the company. In contrast, employees with lower stock level ten
d to leave the company.
stock <- myfile%>%
  group_by(StockOptionLevel)%>%
  count(Attrition)%>%
  mutate(AttritionRate=scales::percent(n/sum(n)))
stock
```

```
## # A tibble: 8 x 4
## # Groups:   StockOptionLevel [4]
##   StockOptionLevel Attrition     n AttritionRate
##   <fct>            <fct>     <int> <chr>
## 1 0                No          477 75.6%
## 2 0                Yes         154 24.4%
## 3 1                No          540 90.6%
## 4 1                Yes          56 9.4%
## 5 2                No          146 92.4%
## 6 2                Yes          12 7.6%
## 7 3                No           70 82.4%
## 8 3                Yes          15 17.6%
```

```
ggplot(data=myfile, aes(x=StockOptionLevel, fill=Attrition))+
  geom_bar()+
  geom_text(data=stock, aes(y=n,label=AttritionRate), position=position_stack
(vjust=0.5), size=3)+
  ggtitle('Attrition by Stock Option Level')+
  scale_x_discrete(name='Stock Option Level')+
  scale_y_continuous(name='Number of employees')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```




```
#5.Models and analysis

#5.1.Data split
require(DMwR)

## Loading required package: DMwR

## Loading required package: lattice

## Loading required package: grid

library(e1071)
require(ROCR)

## Loading required package: ROCR
```

```
## Loading required package: gplots

##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##     lowess

library(gbm)

## Loaded gbm 2.1.5

library(glmnet)

## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following object is masked from 'package:tidyr':
##
##     expand

## Loading required package: foreach

##
## Attaching package: 'foreach'

## The following objects are masked from 'package:purrr':
##
##     accumulate, when

## Loaded glmnet 2.0-16

require(boot)

## Loading required package: boot

##
## Attaching package: 'boot'

## The following object is masked from 'package:lattice':
##
##     melanoma

library(caret)

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following object is masked from 'package:glmnet':
##
##     auc
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(modelr)
```

```
##
## Attaching package: 'modelr'
```

```
## The following object is masked from 'package:DMwR':
##
##     bootstrap
```

```
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:boot':
##
##     logit
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

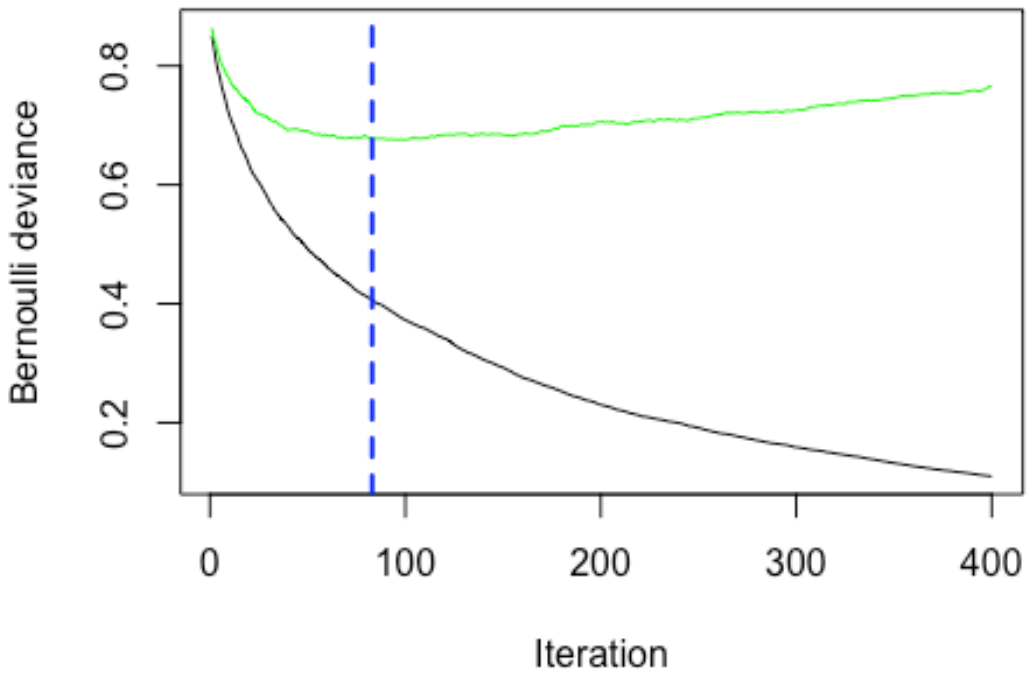
```
library(class)
library(randomForest)
```

```
## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':
##
##     combine

## The following object is masked from 'package:ggplot2':
##
##     margin

library(caTools)
#First, we divide data set into training and test test with a ratio of 9:1.
set.seed(0)
subset <- sample.split(myfile$Attrition, SplitRatio = 0.9)
trainO <- subset(myfile, subset==TRUE)
test <- subset(myfile, subset==FALSE)

#5.2.Variable selection
trainO$Attrition <- ifelse(trainO$Attrition=="Yes",1,0)
x <- model.matrix(Attrition~., trainO)[,-1]
set.seed(1)
cv.logfit <- cv.glmnet(x, trainO$Attrition, family = "binomial", alpha = 1)
logfit <- glmnet(x,trainO$Attrition, family = "binomial", alpha = 1, lambda =
(cv.logfit$lambda.min))
trainO$Attrition <- ifelse(trainO$Attrition==1, "Yes","No")
trainO$Attrition <- as.factor(trainO$Attrition)
lasso.df <- as.data.frame(as.matrix(coef(logfit)))%>%
  mutate(feature = row.names(.))
names(lasso.df)[names(lasso.df)=="s0"] <- "coefficient"
ggplot(lasso.df, aes(x = reorder(feature, coefficient),
                     y = coefficient)) +
  geom_bar(stat='identity') +
  coord_flip() +
  theme_classic() +
  labs(
    x     = "Feature",
    y     = "Coefficient",
    title = "Lasso- Model Selection"
  )
```


```
#The above graph shows the importance of each coefficient in the data, using
lasso. All the coefficients which have no impact on Attrition have been penal
ized by lasso and have been brought down to zero. Hence, using lasso, we deci
de to remove MonthlyRate and PercentSalaryHike, as they seem to have no impac
t on Attrition.
a <- (Attrition~.-MonthlyRate-PercentSalaryHike)

#5.3.Modeling
#Logistic
glm.fit<-glm(a, trainO, family = "binomial")
glm.predict<- predict(glm.fit, test, type = "response")
table(predicted=glm.predict>0.5, actual=test$Attrition)

##          actual
## predicted  No Yes
##     FALSE 119  10
##     TRUE    4  14

#Naive Bayes
nb.fit<- naiveBayes(a, trainO)
nb.predict<- predict(nb.fit,test)
table(nb.predict, test$Attrition)
```

```
##
## nb.predict  No Yes
##        No  107   8
##        Yes  16  16

mean(nb.predict==test$Attrition)

## [1] 0.8367347

#LDA
ldfit<- lda(a, trainO)
ldpredict<- predict(ldfit, test)
mean(ldpredict$class==test$Attrition)

## [1] 0.9047619

table(predicted=ldpredict$class,actual=test$Attrition)

##          actual
## predicted  No Yes
##       No  121  12
##       Yes   2  12

#SVM
svm.fit<-svm(a, trainO, cost= 12,kernel= "linear", scale = TRUE, decision.val
ues=TRUE, probability=T)
svm.predict<-predict(svm.fit, test, prabability=T)
table(predicted=svm.predict, actual=test$Attrition)

##          actual
## predicted  No Yes
##       No  120  12
##       Yes   3  12

mean(svm.predict==test$Attrition)

## [1] 0.8979592

#Random Forrest
set.seed(3)
rf.fit<- randomForest(a, data=trainO, mtry = 6,importance = TRUE, n.tree=600
)
rf.predict<- predict(rf.fit, test)
table(predicted=rf.predict, actual=test$Attrition)

##          actual
## predicted  No Yes
##       No  123  20
##       Yes   0   4

mean(rf.predict==test$Attrition)

## [1] 0.8639456
```

```r
#GBM
trainO$Attrition<- ifelse(trainO$Attrition=="Yes",1,0)
test$Attrition<- ifelse(test$Attrition=="Yes", 1,0)
set.seed(44)
boost.fit=gbm(Attrition~.,data=trainO,distribution="bernoulli", n.minobsinnod
e = 10,n.trees=400,shrinkage = 0.1, interaction.depth=4, cv.folds = 5, n.core
s = 2)
best.boost<- gbm.perf(boost.fit, method = "cv")
```

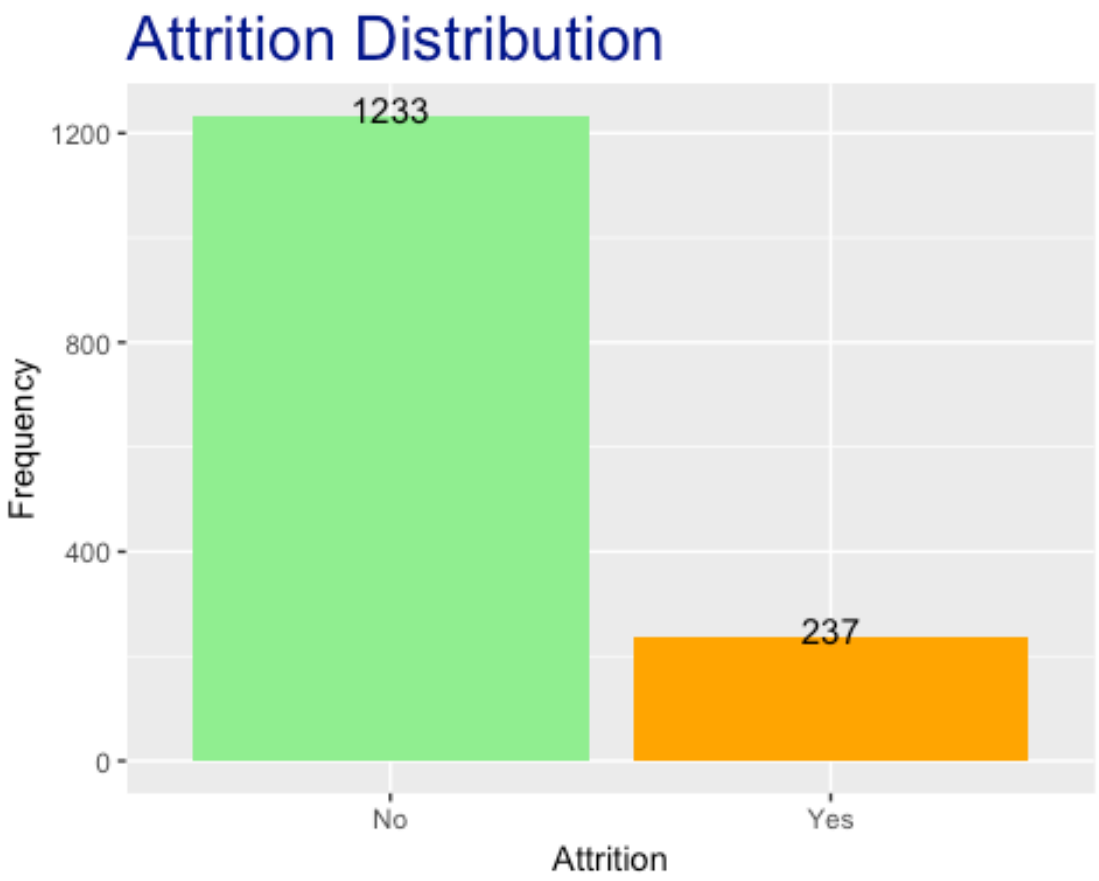



```r
boost.predict<- predict(boost.fit,test, n.trees = best.boost, type = "respons
e")
table(predicted=boost.predict>0.5,actual= test$Attrition)
```

```
##          actual
## predicted   0   1
##     FALSE 123  15
##     TRUE    0   9
```

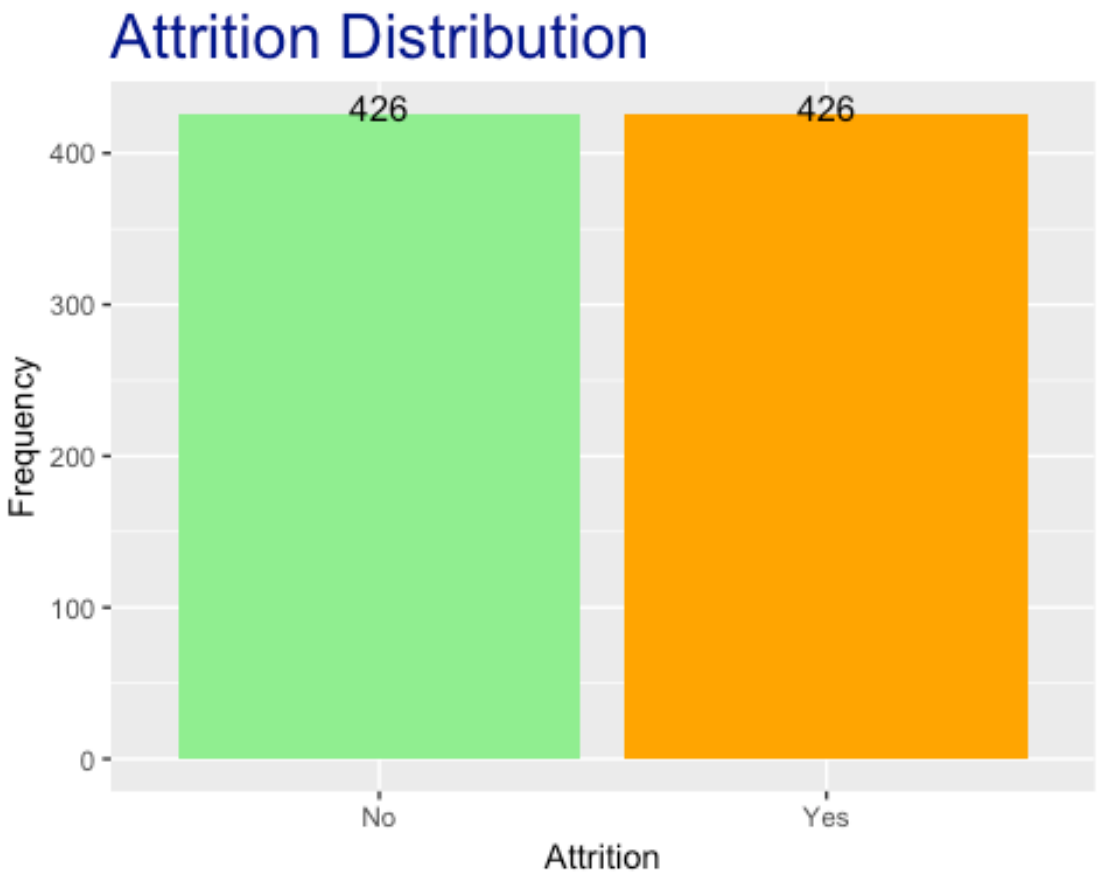
```r
trainO$Attrition<- as.factor(ifelse(trainO$Attrition==1, "Yes","No"))
test$Attrition<- as.factor(ifelse(test$Attrition==1, "Yes","No"))
#As we can observe that the accuracy of all the models is really good, almost
all the models cross the accuracy of 90%. However, most of them are lagging o
n the sensitivity part and in this project our aim is to accurately predict t
he employee attrition (whether an employee will leave the company or not), wh
```

```
ich our models are not able to serve well.

#5.4.Imbalance issue
#As we find out from Exploration part that there is a huge imbalance in our d
ependent variable which is probably causing the sensitivity issue. The sensit
ivity is calculated as the number of correct positive predictions of employee
attrition divided by the total number of positive employee attrition.
ggplot(data=myfile, aes(x=Attrition))+
  geom_bar(fill=c('LightGreen','Orange'))+
  geom_text(aes(label=..count..), stat='count', vjust=0.2, size=4)+
  ggtitle('Attrition Distribution')+
  scale_y_continuous(name='Frequency')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```



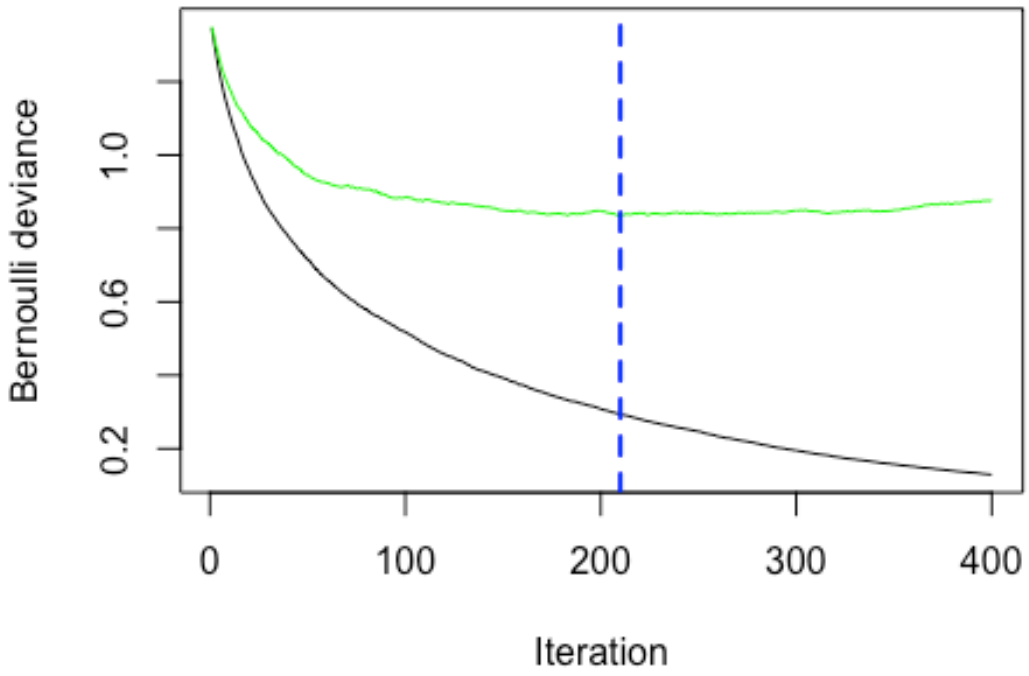

```
#Now, in order to take care of the imbalance issue, we need to balance the da
ta either by under-sampling the majority class which is "No", or over-samplin
g the minority class which is "Yes". By under-sampling, we will lose lot of o
ur data and we cannot afford to lose data as we do not have much data. But we
know the function called SMOTE which is kind of in between under-sample as we
ll as over-sample at the same time. Hence, we use SMOTE to get rid of the imb
alance issue. Moreover, SMOTE should only be applied on training data and lea
ve untouched on test data.
train <- SMOTE(Attrition~.,trainO,perc.over =100)
```

```
#After running SMOTE we see our dependent variable in training set is balance
d, each class now has 426 observations. So we will re-run the models on train
ing data and validate the fit on test data.
ggplot(data=train, aes(x=Attrition))+
  geom_bar(fill=c('LightGreen','Orange'))+
  geom_text(aes(label=..count..), stat='count', vjust=0.2, size=4)+
  ggtitle('Attrition Distribution')+
  scale_y_continuous(name='Frequency')+
  theme(plot.title=element_text(size=20, color='DarkBlue'))
```




```
#5.5.Modeling after handling imbalance issue
#Random Forest
set.seed(3)
rf.fit1<- randomForest(a, data=train, mtry = 6,importance = TRUE, n.tree=600
)
rf.predict1<- predict(rf.fit1, test)
table(predicted=rf.predict1, actual=test$Attrition)

##           actual
## predicted  No Yes
##       No  108   6
##       Yes  15  18

mean(rf.predict1==test$Attrition)
```

```
## [1] 0.8571429

#Logistic
glm.fit1<-glm(a, trainO, family = "binomial")
glm.predict1<- predict(glm.fit1, test, type = "response")
table(predicted=glm.predict1>0.5, actual=test$Attrition)

##          actual
## predicted  No Yes
##     FALSE 119  10
##     TRUE    4  14

#Naive bayes
nb.fit1<- naiveBayes(a, train)
nb.predict1<- predict(nb.fit1,test)
table(nb.predict1, test$Attrition)

##
## nb.predict1 No Yes
##         No  78   5
##         Yes 45  19

mean(nb.predict1==test$Attrition)

## [1] 0.6598639

#LDA
ldfit1<- lda(a, train)
ldpredict1<- predict(ldfit1, test)
mean(ldpredict1$class==test$Attrition)

## [1] 0.8095238

table(predicted=ldpredict1$class,actual=test$Attrition)

##          actual
## predicted  No Yes
##       No  101   6
##       Yes  22  18

#GBM
train$Attrition<- ifelse(train$Attrition=="Yes",1,0)
test$Attrition<- ifelse(test$Attrition=="Yes", 1,0)
set.seed(44)
boost.fit1=gbm(Attrition~.,data=train,distribution="bernoulli", n.minobsinnod
e = 10,n.trees=400,shrinkage = 0.1, interaction.depth=4, cv.folds = 5, n.core
s = 2)
best.boost1<- gbm.perf(boost.fit1, method = "cv")
```




```
boost.predict1<- predict(boost.fit1,test, n.trees = best.boost, type = "respo
nse")
table(predicted=boost.predict1>0.5,actual= test$Attrition)
```

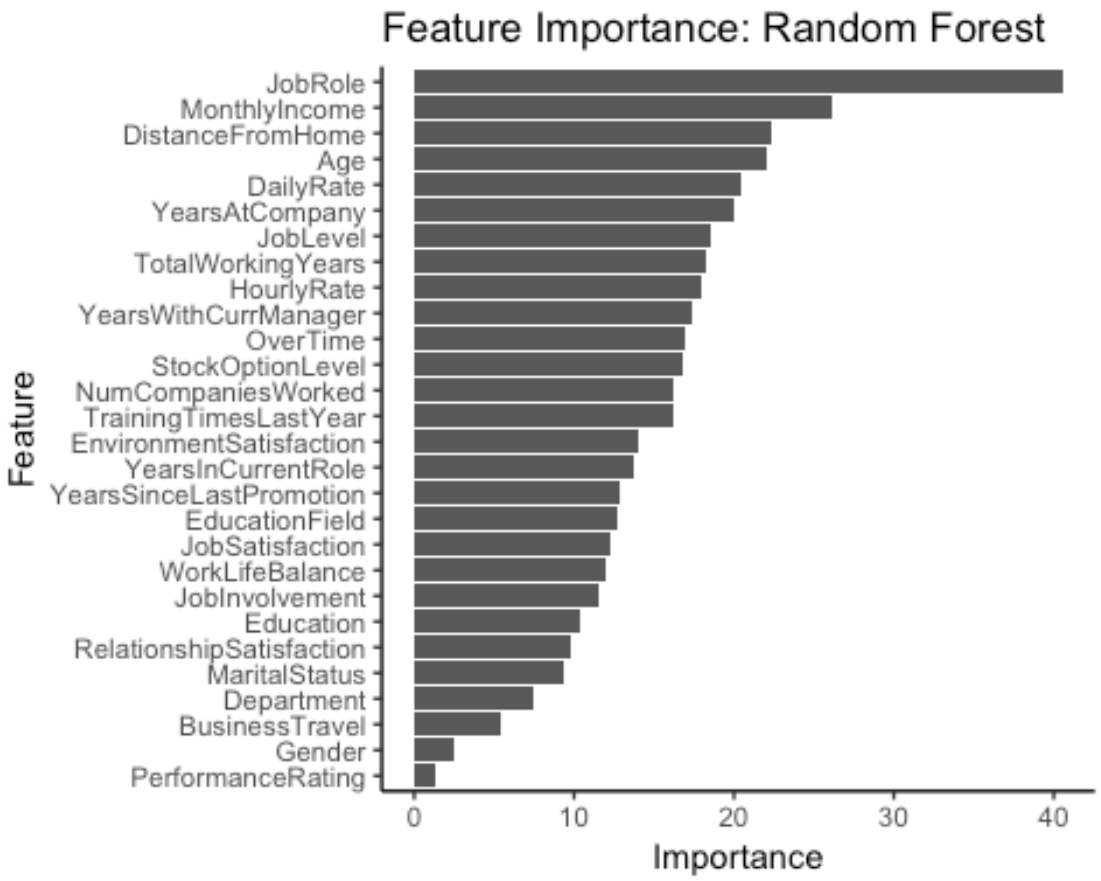
```
##          actual
## predicted  0  1
##     FALSE 98  6
##     TRUE  25 18
```

```
train$Attrition<- as.factor(ifelse(train$Attrition==1, "Yes","No"))
test$Attrition<- as.factor(ifelse(test$Attrition==1, "Yes","No"))
#SVM
svm.fit1<-svm(a, train, cost= 12,kernel= "linear", scale = TRUE, decision.val
ues=TRUE, probability=T)
svm.predict1<-predict(svm.fit1, test, prabability=T)
table(predicted=svm.predict1, actual=test$Attrition)
```

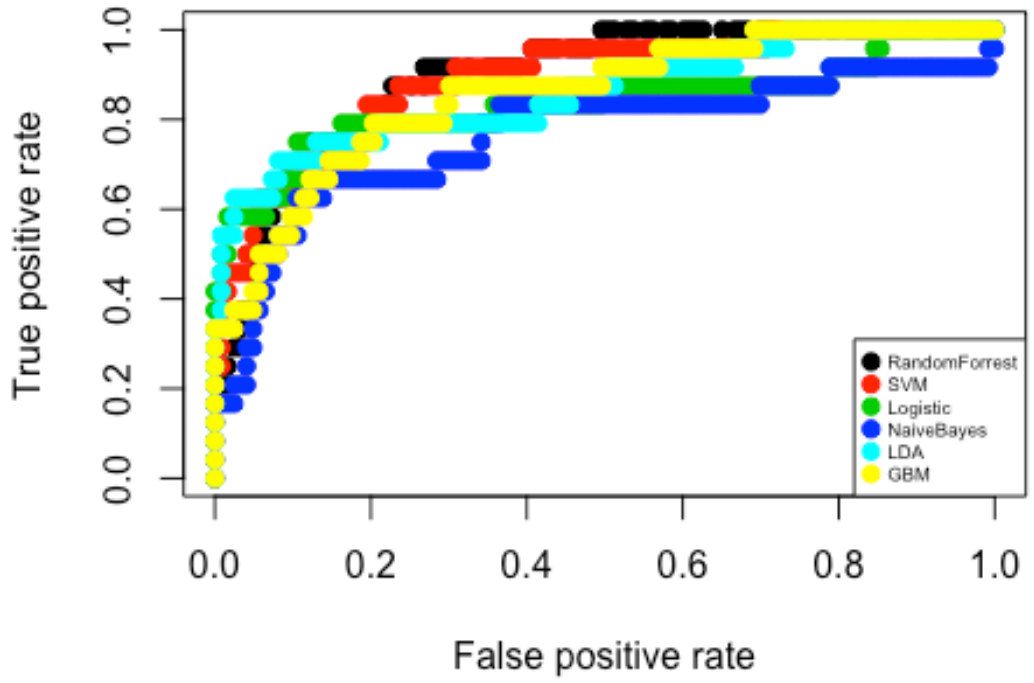
```
##          actual
## predicted No Yes
##       No  97   4
##       Yes 26  20
```

```
mean(svm.predict1==test$Attrition)
```

```
## [1] 0.7959184

#After taking care of the imbalance issue, although accuracy has been impacte
d, our models perform far better in terms of sensitivity. The graph gives us
fair idea about model and it is performance on both aspects i.e, accuracy as
well as sensitivity.

#5.6.Model selection
#Importance Plot
feat_imp_df <- importance(rf.fit1) %>%
  data.frame() %>%
  mutate(feature = row.names(.))
#Plot dataframe
ggplot(feat_imp_df, aes(x = reorder(feature, MeanDecreaseGini),
                        y = MeanDecreaseGini)) +
  geom_bar(stat='identity') +
  coord_flip() +
  theme_classic() +
  labs(
    x     = "Feature",
    y     = "Importance",
    title = "Feature Importance: Random Forest"
  )
```



```
#ROC
rocplot=function(pred, truth, ...){
  predob = prediction (pred, truth)
  perf = performance (predob , "tpr", "fpr")
  plot(perf ,...)
}
rocplot(predict(rf.fit1, test,type = "prob")[,2],test$Attrition,type="b",lty=
1, col= 1, pch=19)
rocplot(attributes(predict(svm.fit1, test, decision.values = TRUE, probabilit
y = T))$probabilities[,2],test$Attrition,add=T, col= 2, type="b", pch=19)
rocplot(predict(glm.fit1, test, type = "response"), test$Attrition, add=T, co
l= 3, type="b", pch=19)
rocplot(predict(nb.fit1,test, type = "raw")[,2],test$Attrition, add=T, col=4,
type="b", pch=19)
rocplot(ldpredict1$posterior[,2], test$Attrition, add=T, col=5, type="b", pch
=19)
rocplot(boost.predict1, test$Attrition, add=T, col= 7, type="b", pch=19)
legend("bottomright",legend = c("RandomForrest", "SVM","Logistic", "NaiveBaye
s","LDA","GBM"), col = c(1,2,3,4,5,7),pt.cex = 1, cex = 0.5,pch=19)
```




```
#To confirm our decision on model selection, we plot ROC curve which also sho
ws the performance of our models and LDA clearly is touching more to the corn
er.
```

```
#Additionally, we have the plot of variables ranked on the base of its import
ance, derived from Random Forrest Model. From this plot, we know which variab
les are important to consider action plan for the company.



#6.Findings and managerial implications

#Remuneration: employees who have low monthly income tend to leave the compan
y. Therefore, we may consider to increase their remuneration. However, since
monthly rate is not a significant factor to attrition, we need to ensure that
their total earnings per month increases. By that, we may increase their basi
c salary or offer additional monthly allowance. Furthermore, we may update mo
nthly bonus scheme, especially for Sales Representatives, to motivate them as
well as to give them an opportunity to earn extra.

#Job: we notice that employees with higher job level tend to stay with the co
mpany. In addition, job level is one of the significant factors which impact
attrition rate. As such, we may consider to promote outstanding employees who
deliver continously great results. In fact, instead of looking for external c
andidates to fill senior positions, we may give the chance to our internal ca
ndidates.

#Work Load: employees tend to leave the company if they have too much overtim
e, and the reason for overtime is workload. To overcome this challenge, we ma
y consider to hire interns or part-time employees to cover partial work of fu
ll-time employees. Additionally, we may invest in technology so that machines
can also cover partial work of employees.

#Location: employees who live far from office tend to have a higher attrition
rate. For those whose job is not required to physical presence at office, suc
h as Sales Executives, we may offer them work-from-home option. For those who
se job is required to present at office, such as Laboratory Technicians, we m
ay offer transporation allowance or company shuttle bus or we even consider t
o move office to strategic locations where are accessible to all employees.

#Privilege: stock option level is one of the important factors that can help
to retain employees. In fact, offering stock option to employees can enhace e
ngagement from employees as we are giving ownership of the company to them.



#7.Conclusions

#HR Managers of IBM Analytics can use our analysis to accurately predict the
attrition using our LDA model as it best predicts the proportion of employees
leaving the company. That is, if we are provided with all the employee detail
s and when we put it in our model it will rightly predict the probability of
the employee attrition.
```