Predicting Whether or Not an Employee Will Quit

Introduction

Many companies often lose some of their best employees due to low satisfactory levels or unsatisfactory working conditions. Often when employees are unhappy, they will jump ship and move on to the next job. Some employees quit without any indication, while others it was a long time coming.

These types of shifts in employee numbers can cause a decrease in overall productivity along with company success. Identifying and catching possible identifiers in whether or not an employee will quit can often save the company from low productivity and losing profit.

Problem

For many companies, losing employees is a costly problem, especially if the employee is highly valued handling top projects. Each time an employee quits, another one must be hired and trained, if the newly trained employee highly productive great, if not they have to repeat the hiring process which is a strain on productivity. The company would like to know why they are losing some of their valued employees, and if there is a way to retain them before they decide to quit.

Our goal in this analysis is to predict whether employees will stay or quit. Companies can then decide on how to retain some of their valued employees. This type of analysis can help companies protect their best employees from quitting.

Data Set

The data set for this analysis focuses on the statistics gathered by human resources on employees that have quit and current employees. In this data set there are 14,999 data entries and 10 variables. The origial data set is simulated data to present a possible problem a company may be faced with. The data is located at https://www.kaggle.com/ludobenistant/hr-analytics.

Employee action is to quit or stay. Left (0 = stay, 1 = quit)

Here are the factors included by the HR stats.

- Satisfaction, employee's satisfaction level at work, ranging between 0 and 1.
- Evaluation, company's last evaluation of an employee, ranging between 0 and 1.
- NumberProjects, the number of projects handled by the employee.
- AvgMonthlyHours, the average montly hours worked by the employee.
- YearsWithCompany, number of years the employee has worked for the company.
- WorkAccident, whether or not the employee expereience a workplace accident.
- Promotion, whether the employee has been promoted in 5 years (0 = no, 1 = yes)
- Department, 10 levels of different jobs offered by the company.
- Salary, 3 levels of salary, low, medium and high.

Data Limitations

Instead of including the exact amount of salary, the data set only includes a factor with 3 levels. If the exact salary were provided, the company could have a more accurate analysis. Also, by including salary amount can help the company while negotiating new contracts. Instead of a range of between "low and medium" they could have an exact amount predicted to offer their employee for them to stay.

The data set is very straight forward and could include other factors that affect the workplace. For example, employee altercations or commute to work distance. These other factors could help provide a better analysis of whether or not an employee will leave their job.

Data Wrangling

The data did not contain any missing values. We adjusted the column names to better reflect the data represented and to clean up the names for presentation. Also, three of the independent variables needed to be adjusted to be factors, so that they correctly reflect the variables represented.

```
#Add data set to workspace and name it hr_stat. Check data.
hr_stat <- read.csv("HR_comma_sep.csv")
summary(hr_stat)
```

```
##  satisfaction_level last_evaluation  number_project  average_montly_hours
##  Min.   :0.0900     Min.   :0.3600   Min.   :2.000   Min.   : 96.0
##  1st Qu.:0.4400     1st Qu.:0.5600   1st Qu.:3.000   1st Qu.:156.0
##  Median :0.6400     Median :0.7200   Median :4.000   Median :200.0
##  Mean   :0.6128     Mean   :0.7161   Mean   :3.803   Mean   :201.1
##  3rd Qu.:0.8200     3rd Qu.:0.8700   3rd Qu.:5.000   3rd Qu.:245.0
##  Max.   :1.0000     Max.   :1.0000   Max.   :7.000   Max.   :310.0
##
##  time_spend_company Work_accident         left
##  Min.   : 2.000     Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 3.000     1st Qu.:0.0000   1st Qu.:0.0000
##  Median : 3.000     Median :0.0000   Median :0.0000
##  Mean   : 3.498     Mean   :0.1446   Mean   :0.2381
##  3rd Qu.: 4.000     3rd Qu.:0.0000   3rd Qu.:0.0000
##  Max.   :10.000     Max.   :1.0000   Max.   :1.0000
##
##  promotion_last_5years         sales         salary
##  Min.   :0.00000       sales      :4140   high  :1237
##  1st Qu.:0.00000       technical  :2720   low   :7316
##  Median :0.00000       support    :2229   medium:6446
##  Mean   :0.02127       IT         :1227
##  3rd Qu.:0.00000       product_mng: 902
##  Max.   :1.00000       marketing  : 858
##                        (Other)    :2923
```

```
#Check for missing values.
summary(is.na(hr_stat))
```

```
##  satisfaction_level last_evaluation number_project  average_montly_hours
##  Mode :logical      Mode :logical   Mode :logical   Mode :logical
##  FALSE:14999        FALSE:14999     FALSE:14999     FALSE:14999
##  time_spend_company Work_accident      left         promotion_last_5years
##  Mode :logical      Mode :logical   Mode :logical   Mode :logical
##  FALSE:14999        FALSE:14999     FALSE:14999     FALSE:14999
##    sales           salary
##  Mode :logical   Mode :logical
##  FALSE:14999     FALSE:14999
```

```
#Rename variable names to be clean and clear.
hr_stat <- hr_stat %>%
  rename(Satisfaction = satisfaction_level) %>%
```

```r
  rename(Evaluation = last_evaluation) %>%
  rename(NumberProjects = number_project) %>%
  rename(AvgMonthlyHours = average_montly_hours) %>%
  rename(YearsWithCompany = time_spend_company) %>%
  rename(WorkAccident = Work_accident) %>%
  rename(Quit = left) %>%
  rename(Promotion = promotion_last_5years) %>%
  rename(Department = sales) %>%
  rename(Salary = salary)

#Change "Quit", "WorkAccident" and "Promotion" to a factor of 0 and 1, 1 being Yes 0 being No.
hr_stat$Quit <- factor(hr_stat$Quit)
hr_stat$Promotion <- factor(hr_stat$Promotion)
hr_stat$WorkAccident <- factor(hr_stat$WorkAccident)

#Change salary to ordered()
hr_stat$Salary <- ordered(hr_stat$Salary, c("low","medium","high"))

#Check data set for final tweaks.
str(hr_stat)
```

```
## 'data.frame':    14999 obs. of  10 variables:
##  $ Satisfaction    : num  0.38 0.8 0.11 0.72 0.37 0.41 0.1 0.92 0.89 0.42 ...
##  $ Evaluation      : num  0.53 0.86 0.88 0.87 0.52 0.5 0.77 0.85 1 0.53 ...
##  $ NumberProjects  : int  2 5 7 5 2 2 6 5 5 2 ...
##  $ AvgMonthlyHours : int  157 262 272 223 159 153 247 259 224 142 ...
##  $ YearsWithCompany: int  3 6 4 5 3 3 4 5 5 3 ...
##  $ WorkAccident    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Quit            : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Promotion       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Department      : Factor w/ 10 levels "accounting","hr",..: 8 8 8 8 8 8 8 8 8 8 ...
##  $ Salary          : Ord.factor w/ 3 levels "low"<"medium"<..: 1 2 2 1 1 1 1 1 1 1 ...
```

Preliminary Analysis

In the premliminary analysis we want to explore each of the independent variables and their relationship to those who left and who stayed.

- The average employee satisfaction rating is at 61.28% satisfaction.
- Those who left the company had an average satifaction rating was 44%.
- The company has a 23.8% employee quitting percentage.

```r
mean(hr_stat$Satisfaction)
```

```
## [1] 0.6128335
```

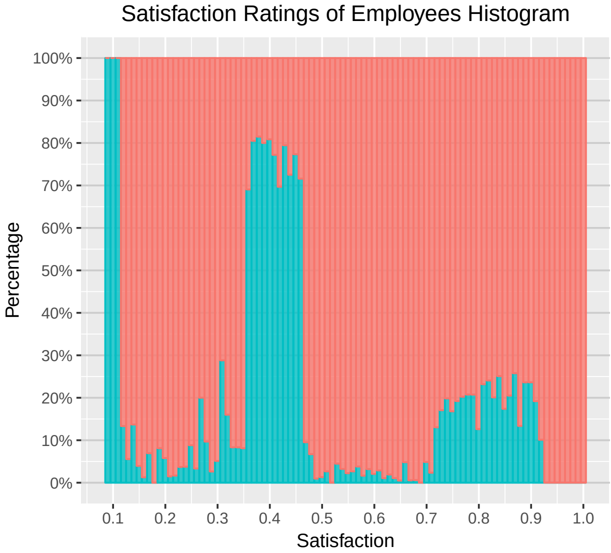
```r
avgsatleft <- hr_stat %>%
  filter(Quit == 1)
mean(avgsatleft$Satisfaction)
```

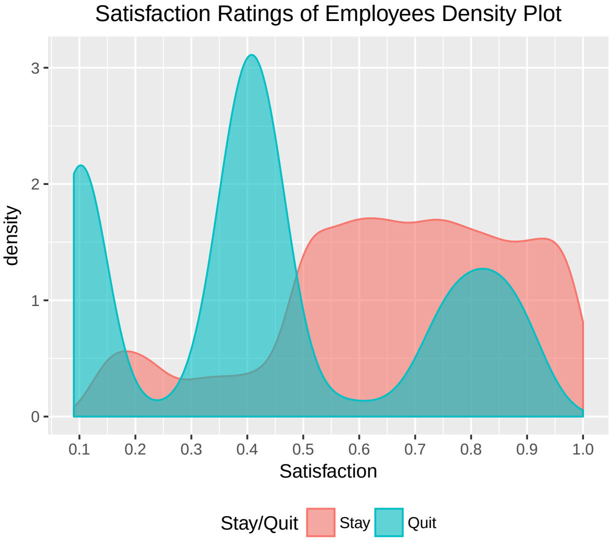
```
## [1] 0.440098
```

```r
nrow(avgsatleft)/nrow(hr_stat)
```

```
## [1] 0.2380825
```

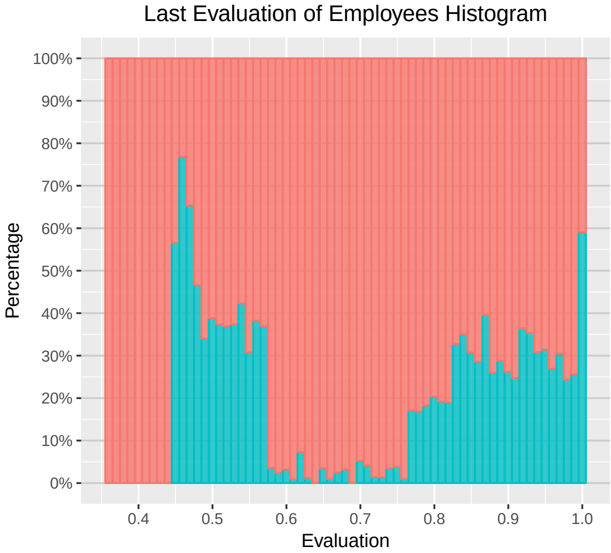
Satisfaction Level

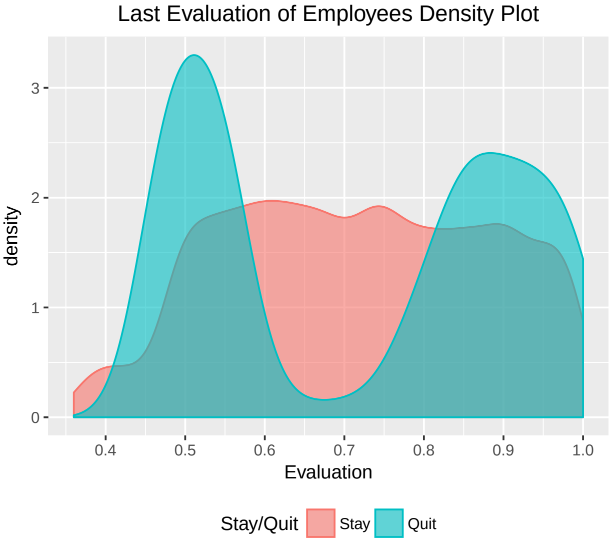




- The plot indicates that most employees who left have a low satisfaction level between 0.37 - 0.50.
- There is a tri-modal effect. Satisfaction levels of (< 15), (0.35 - 0.50), (0.7-0.9) left the company more.
- From the individuals who stayed, we can see a general trend of having 50% or higher satisfaction.

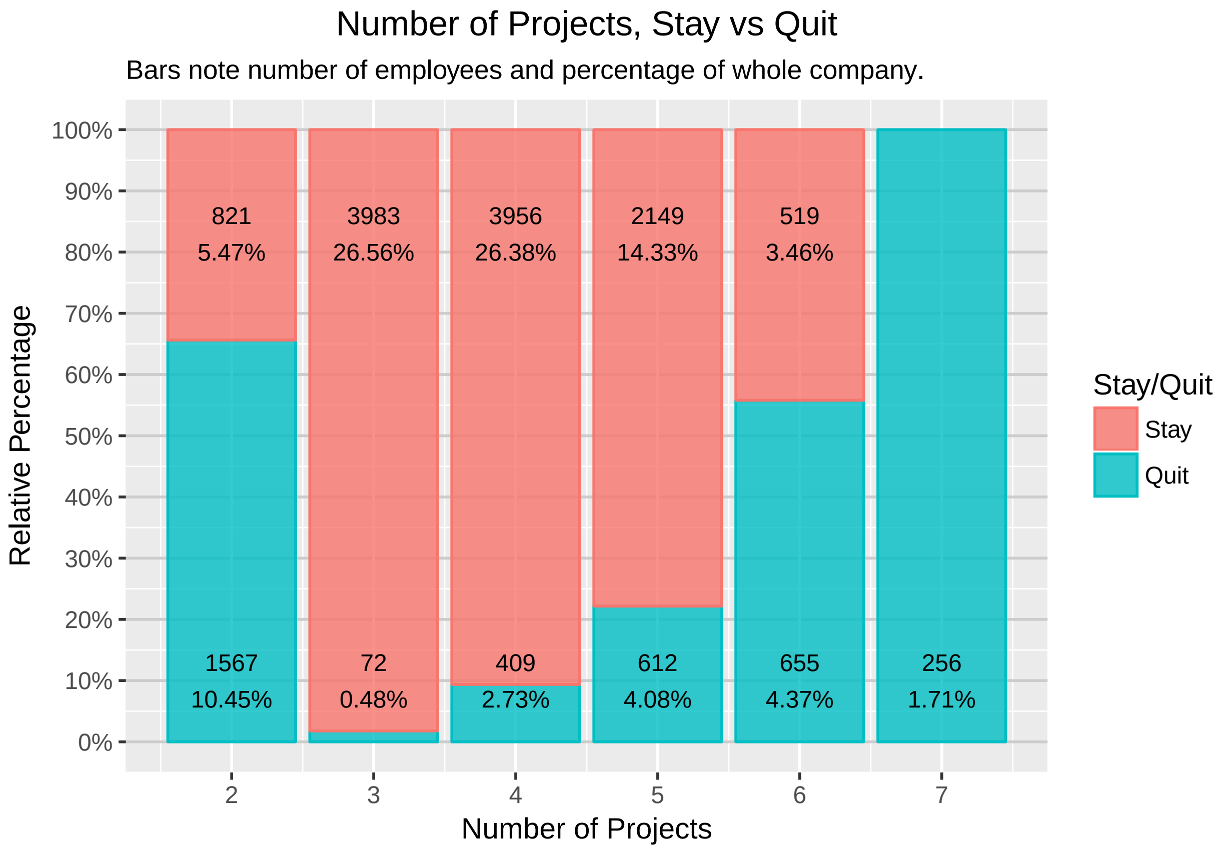
Last Evaluation





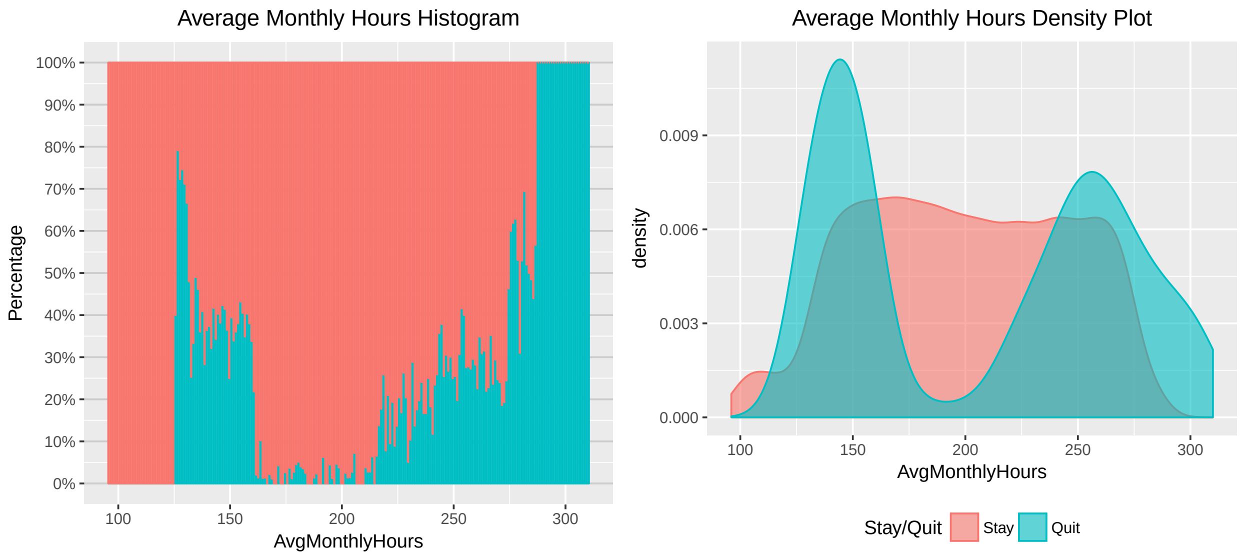
- Bi-modal relationship between quitting and company evaluation.
- The company is losing many of their top evaluated performers.
- Individuals who are staying have an evaluation of above 40%.

Number of Projects



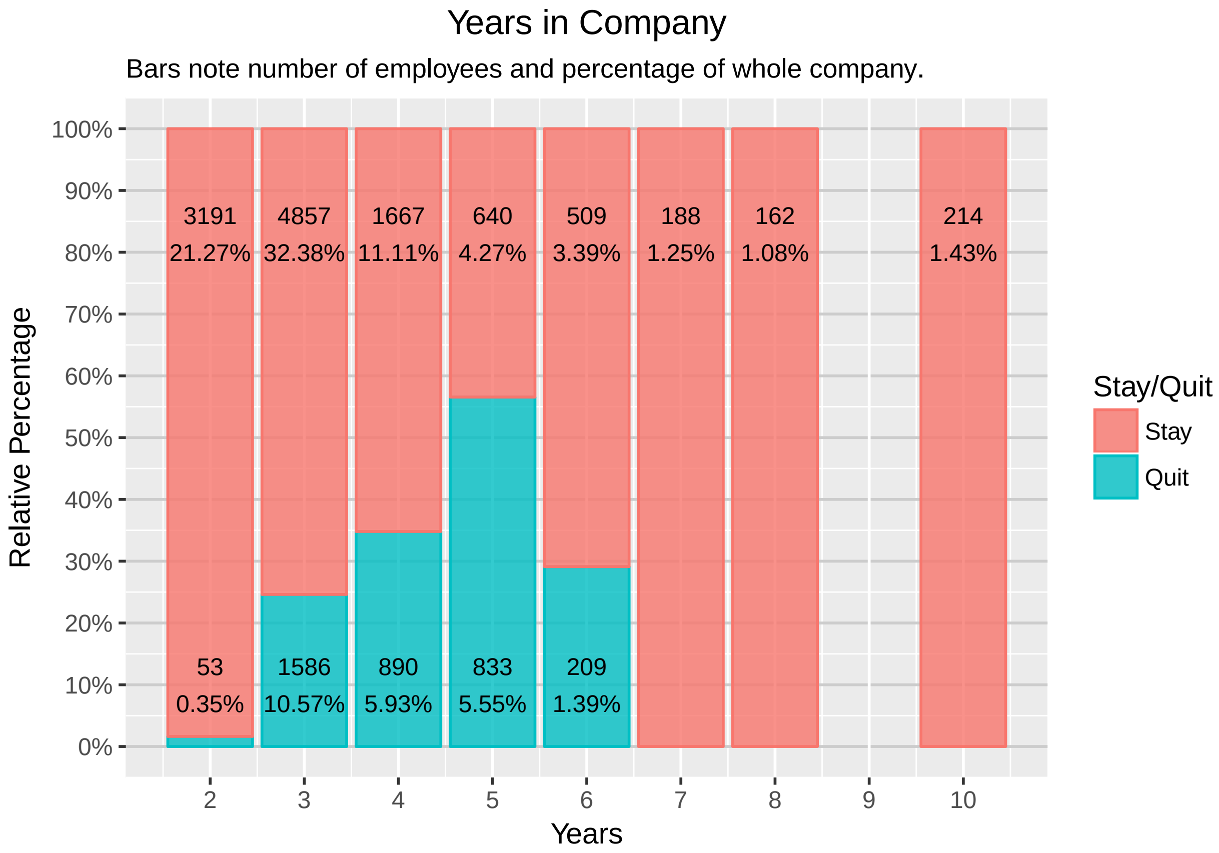
- Individuals with 2, (4-6+) projects are more likely to quit.
- 65% of individuals with 2 and 6+ projects have quit, 16.53% of the whole company.
- Most employees with 3-4 projects have stayed with the company, 52.94% of employees.
- Employees with 2 projects, 65% have quit, which is about half of total employee who quit.

Average Montly Hours Worked



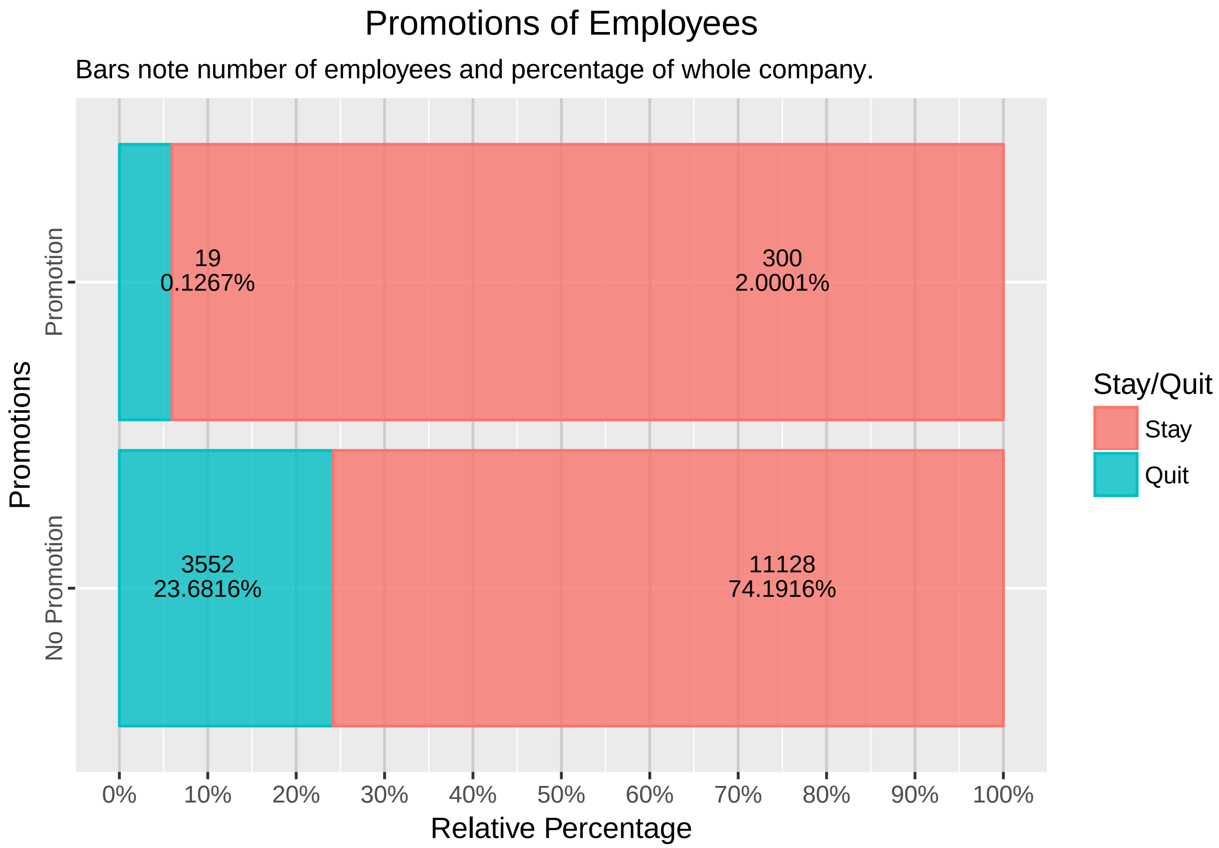
- Bi-modal relationship, many employees who left either worked under 175 hours or above 225.
- There is a higher percentage of employees quitting with 250+ hours.
- Employees who are underworked and overworked are quitting.

Time spent with company



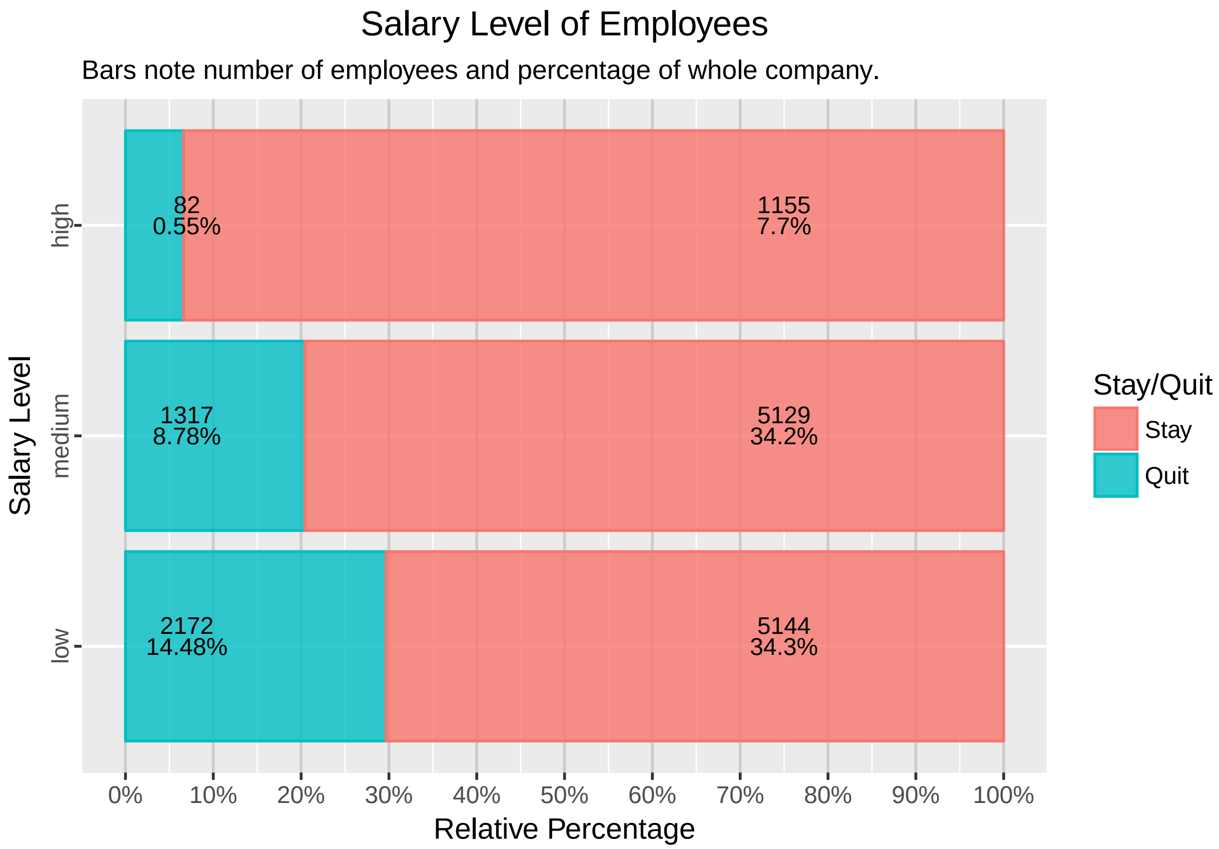
- Most employees left between working 3-6 years with the company, 23.44% of the company.
- Of the employees who have been with the company for 5 years, 50% have quit.

Promotions



- From this table we see that most individuals who left were not offered a promotion in the last 5 years.
- 25% of employees who were not offered a promotion have quit.
- 5% of promoted employees have quit.

Salary

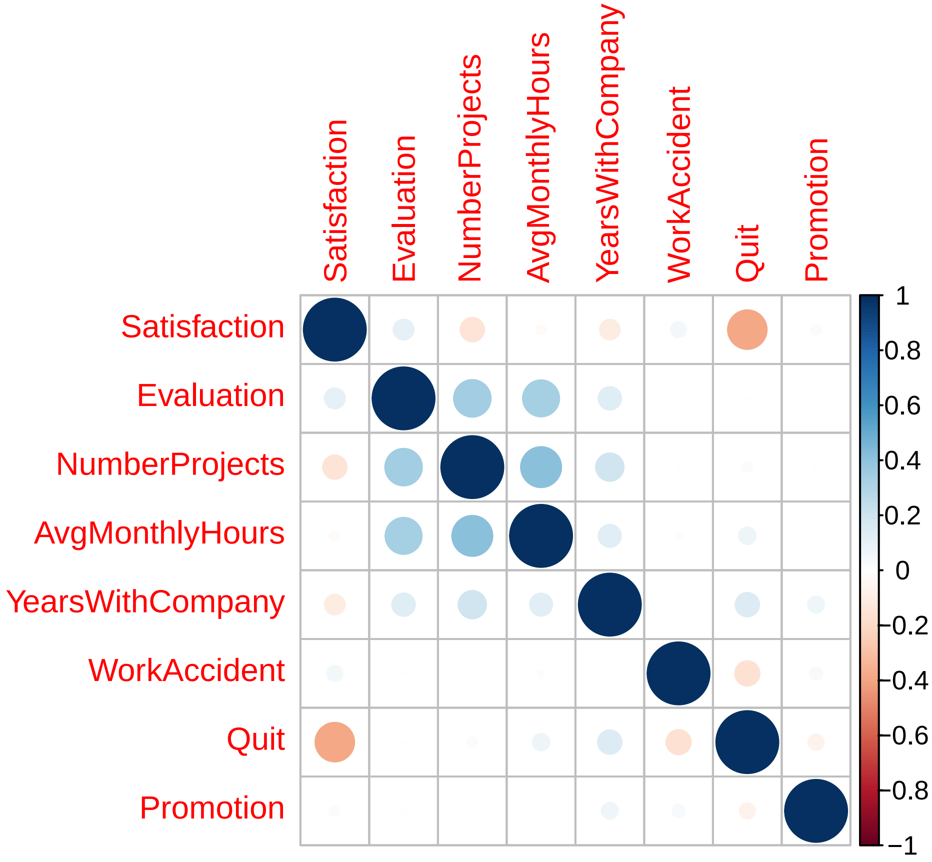


- Most of the employees who quit were in the low and medium bracket of salary.
- About 6.6% of high paid employees have quit.
- 20% of medium paid employees have quit.
- 30% of low paid employees have quit.

Correlation of Variables

Checking the correlation of variables can help avoid colinearity in our analysis. Therefore, running a correlation check can help make sure our coefficients show an accurate relationship.

```
##                   Satisfaction   Evaluation NumberProjects AvgMonthlyHours
## Satisfaction        1.00000000  0.105021214   -0.142969586    -0.020048113
## Evaluation          0.10502121  1.000000000    0.349332589     0.339741800
## NumberProjects     -0.14296959  0.349332589    1.000000000     0.417210634
## AvgMonthlyHours    -0.02004811  0.339741800    0.417210634     1.000000000
## YearsWithCompany   -0.10086607  0.131590722    0.196785891     0.127754910
## WorkAccident        0.05869724 -0.007104289   -0.004740548    -0.010142888
## Quit               -0.38837498  0.006567120    0.023787185     0.071287179
## Promotion           0.02560519 -0.008683768   -0.006063958    -0.003544414
##                   YearsWithCompany WorkAccident        Quit    Promotion
## Satisfaction          -0.100866073  0.058697241 -0.38837498  0.025605186
## Evaluation             0.131590722 -0.007104289  0.00656712 -0.008683768
## NumberProjects         0.196785891 -0.004740548  0.02378719 -0.006063958
## AvgMonthlyHours        0.127754910 -0.010142888  0.07128718 -0.003544414
## YearsWithCompany       1.000000000  0.002120418  0.14482217  0.067432925
## WorkAccident           0.002120418  1.000000000 -0.15462163  0.039245435
## Quit                   0.144822175 -0.154621634  1.00000000 -0.061788107
## Promotion              0.067432925  0.039245435 -0.06178811  1.000000000
```



- Negative correlation (-0.3883) between quitting and satisfaction rating.
- Positive correlation between evaluation, average montly hours(0.3397), and number of projects(0.3493). * Positive correlation between average monthly hours, evaluation (0.3397), and number of projects(0.4172).

Machine Learning

Now that we have a general view of the variables in relation to employees who have stayed and quit. We will use different machine learning methods to build models to predict whether or not and indivudal will quit or stay at the company. We will use three different models, classification and regression tree, logistic regression model, and random forest model.

Splitting the Data Into Training and Testing Subsets

```
#Split data into training and testing.
set.seed(1234)
divide = sample.split(hr_stat, SplitRatio = 0.75)
hr_stat_training = subset(hr_stat, divide == TRUE)
hr_stat_test = subset(hr_stat, divide == FALSE)

#Check the split of data for percentage. Should be approximately 75%
nrow(hr_stat_training)
```

```
## [1] 10499
```

```
nrow(hr_stat_training)/nrow(hr_stat)
```

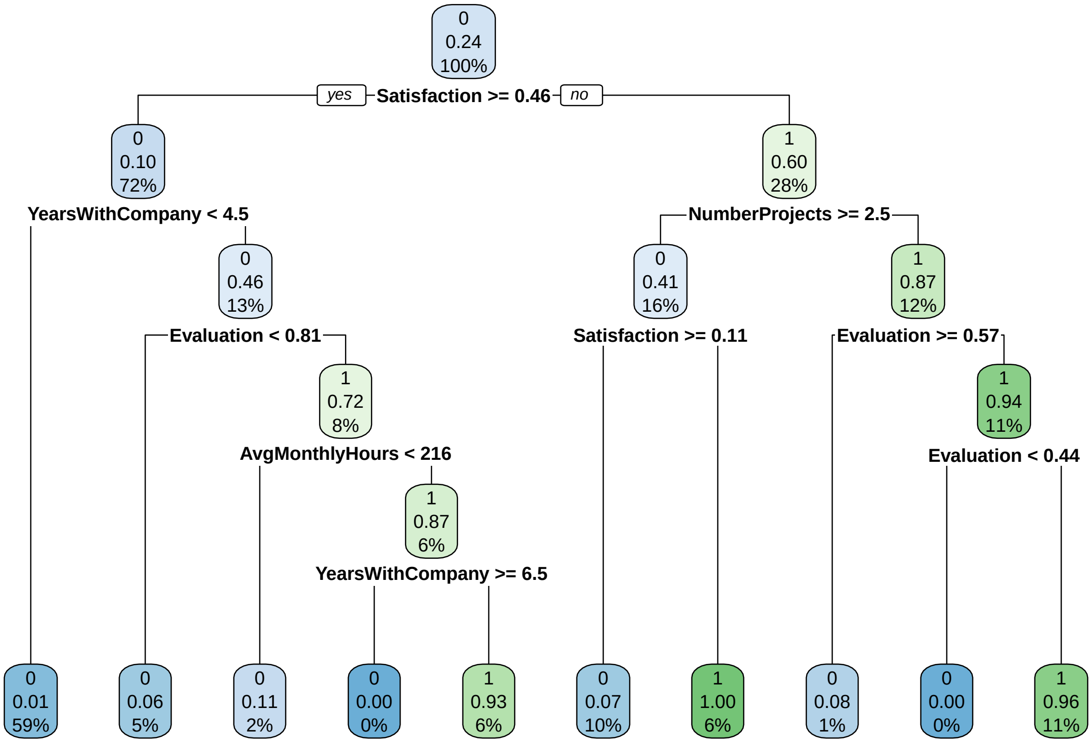
```
## [1] 0.69998
```

Instead of a 75/25 split, we have about a 70/30 split. Since our data set is large it should not be an issue.

Classification Tree

Running the classification tree will help indicate which variables are most important to our model. By seeing which variables siginificant we can then make a better logistic regression model.

```
# Create classification tree using training set
hr_stat_CART = rpart(Quit ~ ., data = hr_stat_training, method = "class",
                     control = rpart.control(minibucket = 25))
rpart.plot(hr_stat_CART)
```



- From the tree the most important factors are satisfaction, years with company, number projects, evaluation, and average monthly hours.
- Now that we have our classification tree, lets see how accurate our model is by using the test subset to predict the whether or not employees will stay or quit.

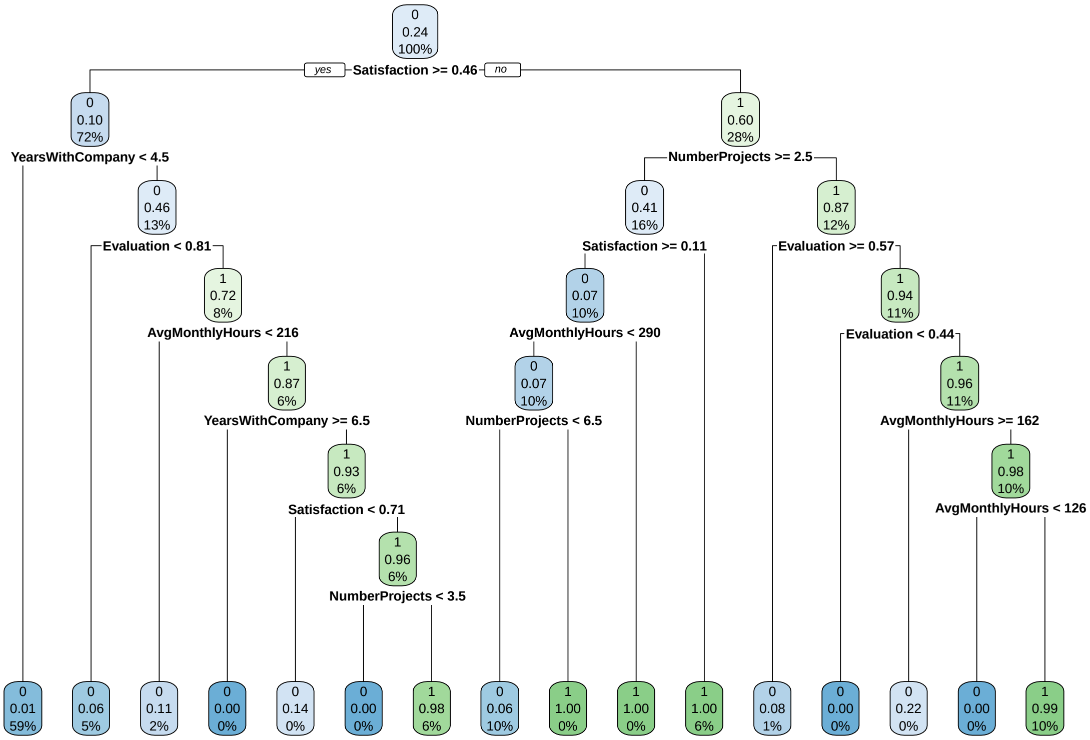
```
PredictCART1 <- predict(hr_stat_CART, newdata = hr_stat_test, type = "class")

confusionMatrix(PredictCART1, hr_stat_test$Quit)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 3391   93
##          1   38  978
##
##                Accuracy : 0.9709
##                  95% CI : (0.9655, 0.9756)
##     No Information Rate : 0.762
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9183
##  Mcnemar's Test P-Value : 2.382e-06
##
##             Sensitivity : 0.9889
##             Specificity : 0.9132
##          Pos Pred Value : 0.9733
##          Neg Pred Value : 0.9626
##              Prevalence : 0.7620
##          Detection Rate : 0.7536
##    Detection Prevalence : 0.7742
##       Balanced Accuracy : 0.9510
##
##        'Positive' Class : 0
##
```

- The classification tree had a 97% accuracy in predicting the test subset.

Lets see if adding more nodes will help strengthen our model.

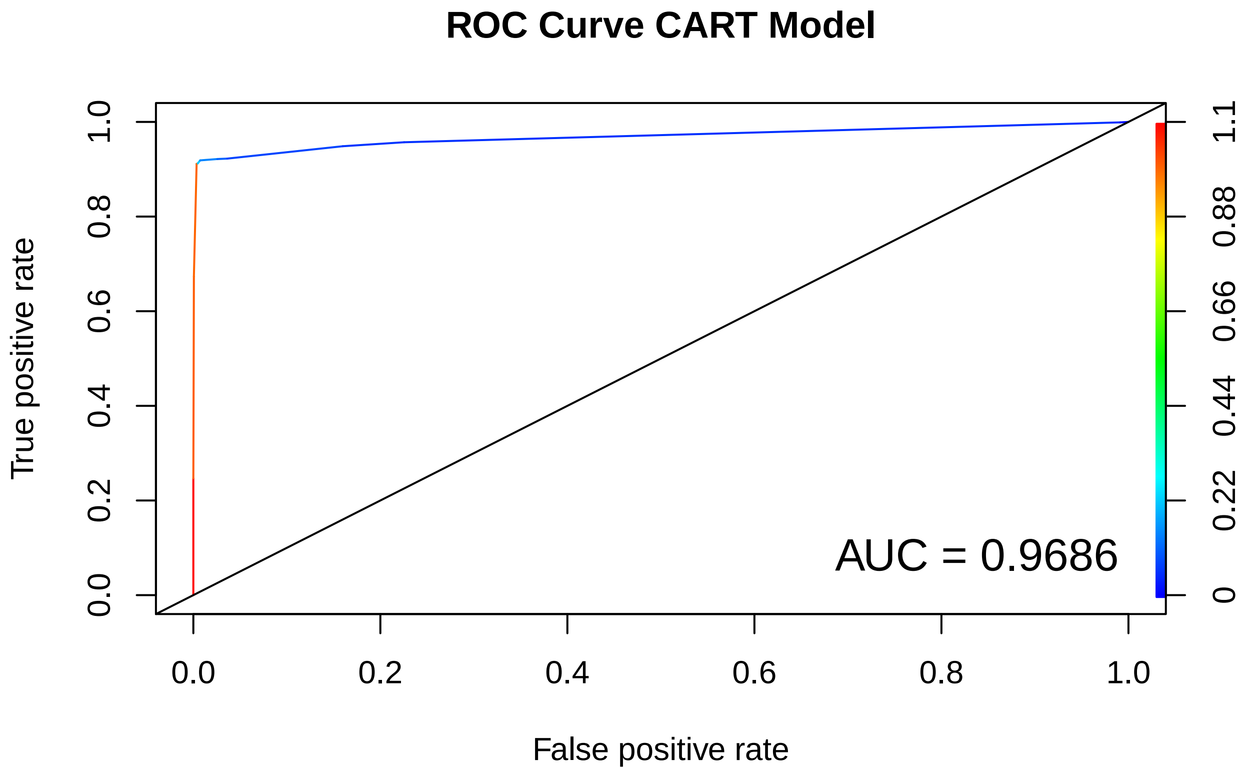
```r
hr_stat_CART2 = rpart(Quit ~ ., data = hr_stat_training, method = "class",
                       control = rpart.control(minibucket = 25, cp = .002))
rpart.plot(hr_stat_CART2)
```

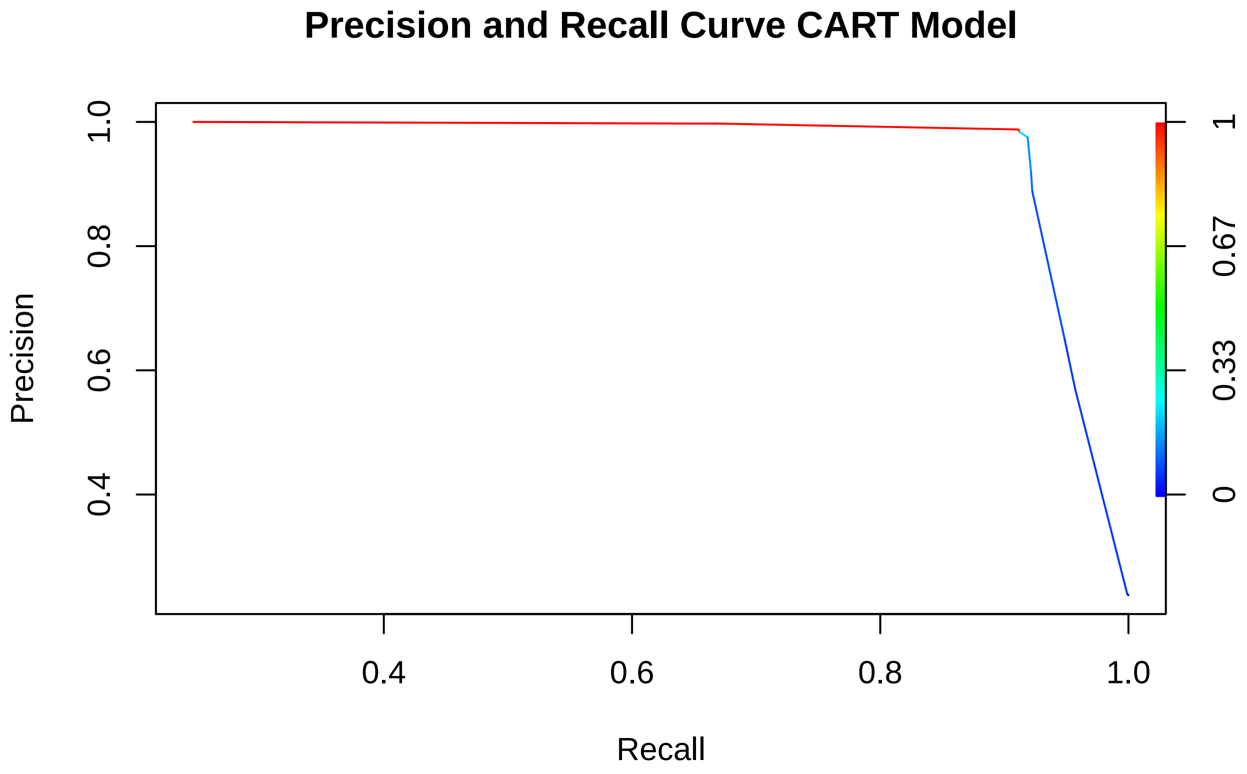


```r
PredictCART2 <- predict(hr_stat_CART2, newdata = hr_stat_test, type = "class")
confusionMatrix(PredictCART2, hr_stat_test$Quit)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 3417   95
##          1   12  976
##
##                Accuracy : 0.9762
##                  95% CI : (0.9713, 0.9805)
##     No Information Rate : 0.762
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9326
##  Mcnemar's Test P-Value : 2.241e-15
##
##             Sensitivity : 0.9965
##             Specificity : 0.9113
##          Pos Pred Value : 0.9729
##          Neg Pred Value : 0.9879
##              Prevalence : 0.7620
##          Detection Rate : 0.7593
##    Detection Prevalence : 0.7804
##       Balanced Accuracy : 0.9539
##
##        'Positive' Class : 0
##
```

- This model has an accuracy of 97.6% in predicting our test subset.
- Adding more nodes improvd the accuracy of our model by 0.5% which is a small improvment.


ROC Curve CART Model
True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0
0
0.22
0.44
0.66
0.88
1.1
AUC = 0.9686



- The ROC curve has an AUC of 0.9686, which means the CART model is an excellent model for predicting whether or not an employee will quit their job.
- There is a high true positive rate without any false positive hits making this a strong model.
- Precision Recall plot also shows high performance of our model, precision does not fall till about 0.9 recall.

Logistic Regression Model

With the CART model, we were able to achieve 97.6% accuracy, now lets use logistic regression to build a model. First we will build the model, then remove any of the insignificant variables. After we have our significant model, we will run a check on the variables using the variable inflation factor. Since some of our variables are correlated, we want to check to make sure multicollinearity does not occur. Multicollinearity can affect our coefficients and not show the actual relationship of our independent and dependent variables.

```
#First make a logistic regression model using all variables.
model1 <- glm(Quit ~ ., family = binomial, data = hr_stat_training)
summary(model1)
```

```
##
## Call:
## glm(formula = Quit ~ ., family = binomial, data = hr_stat_training)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1861  -0.6631  -0.4048  -0.1216   3.0655
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -0.3923846  0.1844040  -2.128 0.033349 *
## Satisfaction          -4.1399026  0.1171923 -35.326  < 2e-16 ***
## Evaluation             0.8066603  0.1792679   4.500 6.80e-06 ***
## NumberProjects        -0.3218998  0.0256281 -12.560  < 2e-16 ***
## AvgMonthlyHours        0.0045140  0.0006164   7.323 2.43e-13 ***
## YearsWithCompany       0.2607978  0.0184872  14.107  < 2e-16 ***
## WorkAccident1         -1.4813980  0.1059685 -13.980  < 2e-16 ***
## Promotion1            -1.4376200  0.3096515  -4.643 3.44e-06 ***
## Departmenthr           0.2345076  0.1579529   1.485 0.137632
## DepartmentIT          -0.1004455  0.1470809  -0.683 0.494653
## Departmentmanagement  -0.3027015  0.1902336  -1.591 0.111562
## Departmentmarketing    0.1054093  0.1582669   0.666 0.505397
## Departmentproduct_mng -0.0791575  0.1564702  -0.506 0.612930
## DepartmentRandD       -0.6394020  0.1760604  -3.632 0.000282 ***
## Departmentsales       -0.0161270  0.1235454  -0.131 0.896143
## Departmentsupport      0.1267998  0.1315192   0.964 0.334988
## Departmenttechnical    0.1196931  0.1286373   0.930 0.352128
## Salary.L              -1.2928204  0.1073446 -12.044  < 2e-16 ***
## Salary.Q              -0.3085470  0.0702494  -4.392 1.12e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 11526  on 10498  degrees of freedom
## Residual deviance:  9015  on 10480  degrees of freedom
## AIC: 9053
##
## Number of Fisher Scoring iterations: 5
```

All variables are significant, none will be removed unless there is multicollinearity.

```r
#Run VIF to find if there is a variable inflation.
vif(model1)
```

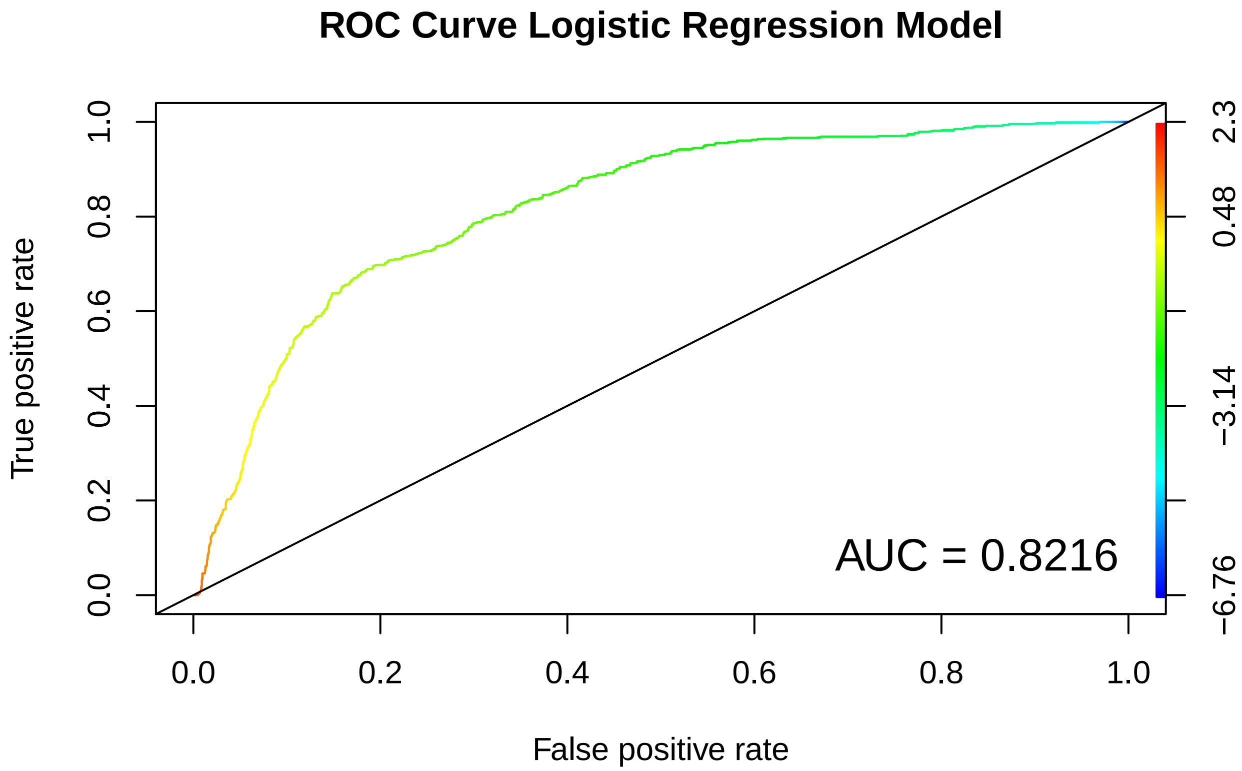
```
##                      GVIF Df GVIF^(1/(2*Df))
## Satisfaction     1.169278  1        1.081332
## Evaluation       1.464610  1        1.210211
## NumberProjects   1.810086  1        1.345394
## AvgMonthlyHours  1.515744  1        1.231156
## YearsWithCompany 1.113695  1        1.055318
## WorkAccident     1.010342  1        1.005157
## Promotion        1.014848  1        1.007397
## Department       1.052682  9        1.002856
## Salary           1.043847  2        1.010786
```

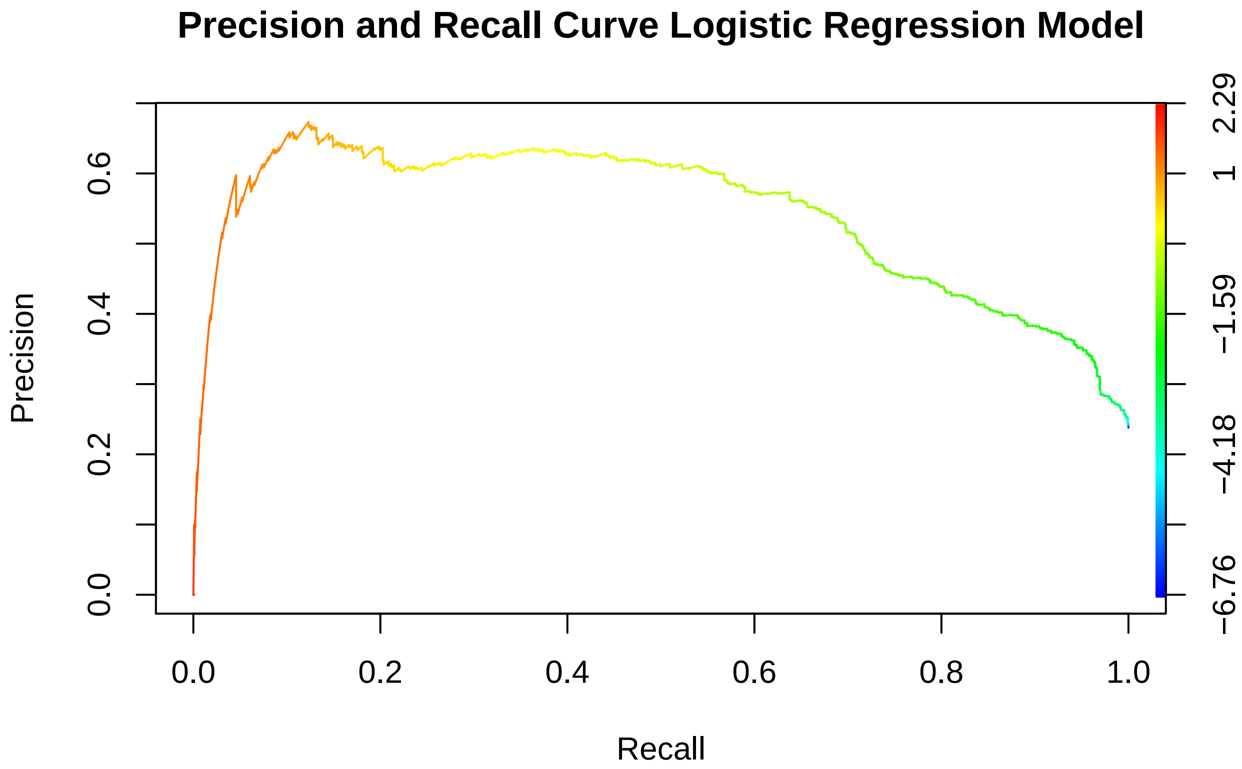
Since the GVIF is not above 5, we do not have multicollinearity and can continue on with our analysis. Now we will see how accurate our model is on the testing data set.

```r
Predmodel1 <- predict(model1, hr_stat_test, type = "response" )
confusionMatrix(as.numeric(Predmodel1 > 0.5), hr_stat_test$Quit)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 3213  698
##          1  216  373
##
##                Accuracy : 0.7969
##                  95% CI : (0.7848, 0.8086)
##     No Information Rate : 0.762
##     P-Value [Acc > NIR] : 1.243e-08
##
##                   Kappa : 0.3375
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9370
##             Specificity : 0.3483
##          Pos Pred Value : 0.8215
##          Neg Pred Value : 0.6333
##              Prevalence : 0.7620
##          Detection Rate : 0.7140
##    Detection Prevalence : 0.8691
##       Balanced Accuracy : 0.6426
##
##        'Positive' Class : 0
##
```

- The logistic regression model was 79.69% accurate.


ROC Curve Logistic Regression Model
True positive rate
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
AUC = 0.8216
2.3
0.48
−3.14
−6.76



- Logistic regression model is not as accurate as our CART model, but still has high performance with a AUC of 0.8216.
- The precision and recall plot also shows that our logistic regression model is high performing, but not as good as our CART model.
- We can try to improve the model by adding interactions between variables.

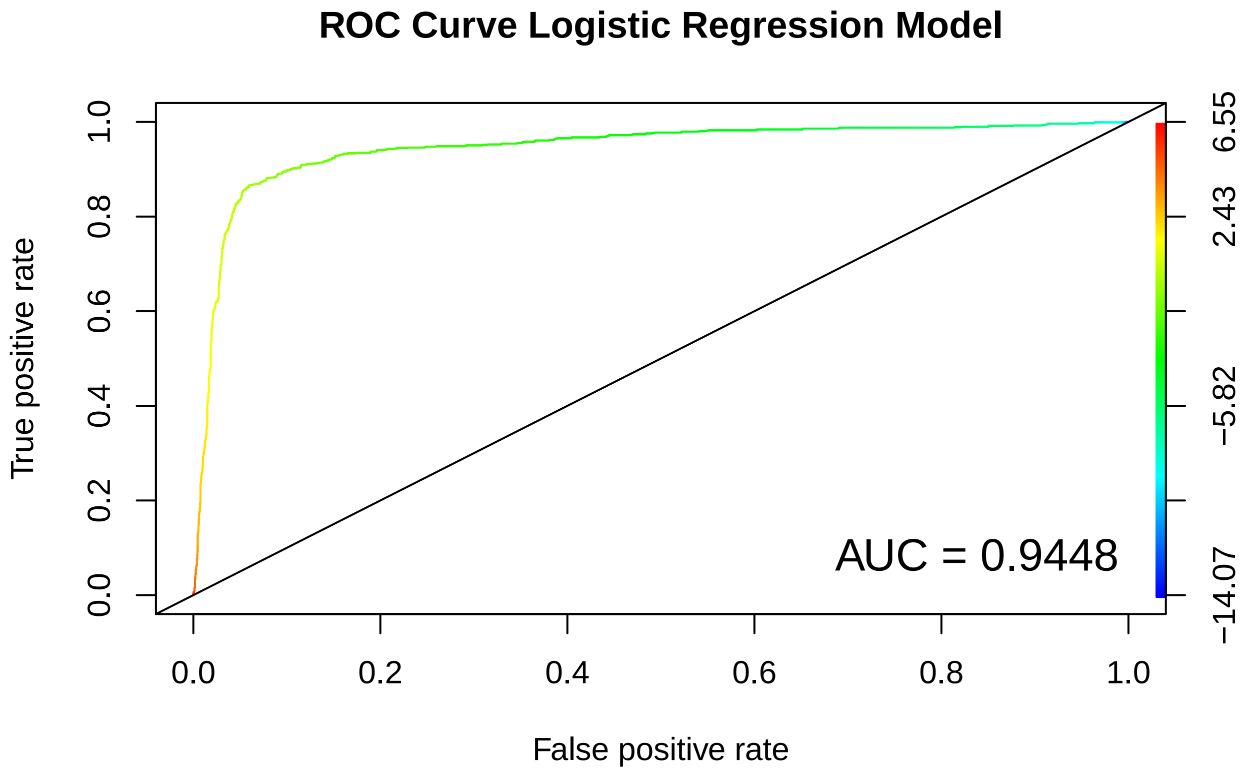
```
modelinteraction2 <- glm(Quit ~ . -Department -WorkAccident -Promotion + Satisfaction*Evaluation + Sati
                           Satisfaction*YearsWithCompany + Evaluation*NumberProjects + Evaluation*AvgMon
                           Evaluation*YearsWithCompany , family = binomial, data = hr_stat_training)
summary(modelinteraction2)
```

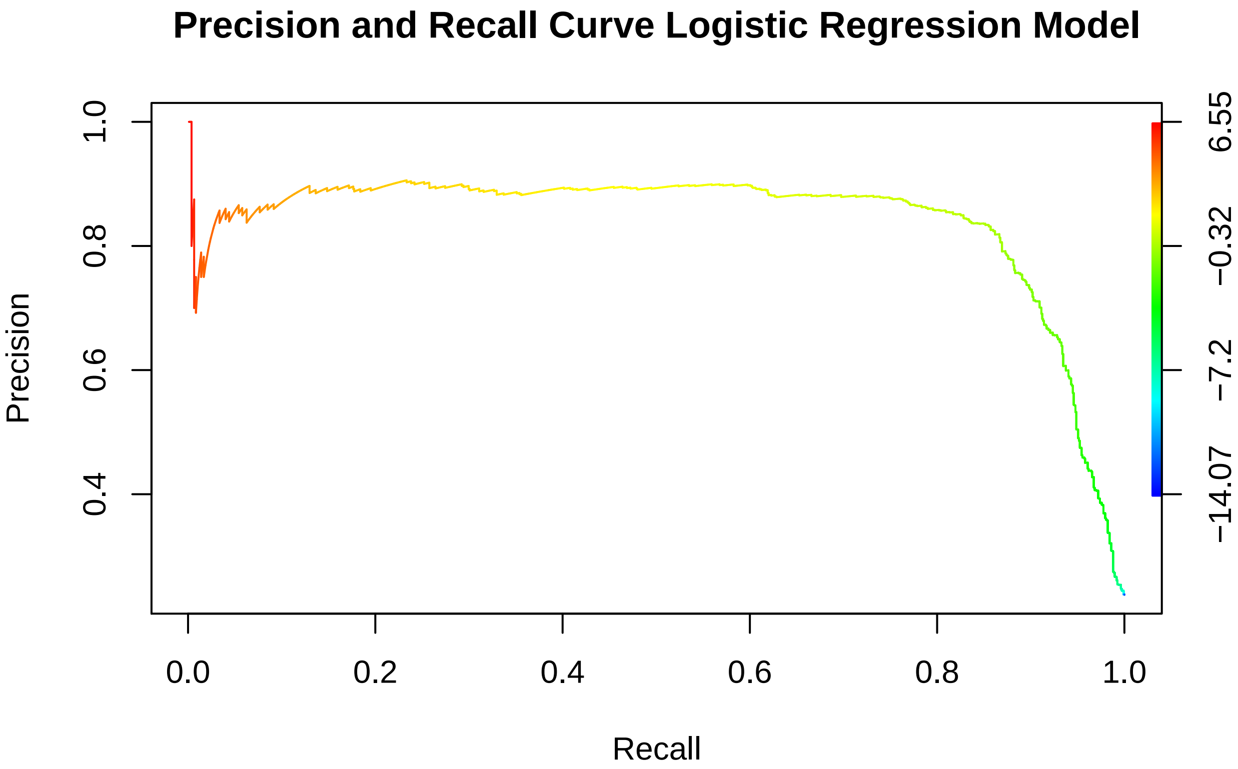
```
##
## Call:
## glm(formula = Quit ~ . - Department - WorkAccident - Promotion +
##     Satisfaction * Evaluation + Satisfaction * NumberProjects +
##     Satisfaction * YearsWithCompany + Evaluation * NumberProjects +
##     Evaluation * AvgMonthlyHours + Evaluation * YearsWithCompany,
##     family = binomial, data = hr_stat_training)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.0908  -0.3270  -0.1476  -0.0208   4.5588
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   42.282395   0.981266  43.090  < 2e-16 ***
## Satisfaction                 -21.669347   1.072517 -20.204  < 2e-16 ***
## Evaluation                    -60.343972   1.533671 -39.346  < 2e-16 ***
## NumberProjects                 -4.527113   0.168207 -26.914  < 2e-16 ***
## AvgMonthlyHours                -0.063748   0.003894 -16.371  < 2e-16 ***
## YearsWithCompany               -1.893472   0.134888 -14.037  < 2e-16 ***
## Salary.L                       -1.311418   0.136354  -9.618  < 2e-16 ***
## Salary.Q                       -0.311733   0.090975  -3.427 0.000611 ***
## Satisfaction:Evaluation        19.388175   1.316508  14.727  < 2e-16 ***
## Satisfaction:NumberProjects    -0.741789   0.161048  -4.606  4.1e-06 ***
## Satisfaction:YearsWithCompany   1.700687   0.121943  13.947  < 2e-16 ***
## Evaluation:NumberProjects       6.617394   0.230296  28.734  < 2e-16 ***
## Evaluation:AvgMonthlyHours      0.103599   0.005361  19.323  < 2e-16 ***
## Evaluation:YearsWithCompany     1.454478   0.157140   9.256  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 11525.8  on 10498  degrees of freedom
## Residual deviance:  5100.2  on 10485  degrees of freedom
## AIC: 5128.2
##
## Number of Fisher Scoring iterations: 7
```

```
Predmodel100 <- predict(modelinteraction2, hr_stat_test, type = "response" )
confusionMatrix(as.numeric(Predmodel100 > 0.5), hr_stat_test$Quit)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 3272  186
##          1  157  885
##
##                Accuracy : 0.9238
##                  95% CI : (0.9156, 0.9314)
##     No Information Rate : 0.762
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.7879
##  Mcnemar's Test P-Value : 0.1306
##
##             Sensitivity : 0.9542
##             Specificity : 0.8263
##          Pos Pred Value : 0.9462
##          Neg Pred Value : 0.8493
##              Prevalence : 0.7620
##          Detection Rate : 0.7271
##    Detection Prevalence : 0.7684
##       Balanced Accuracy : 0.8903
##
##        'Positive' Class : 0
##
```

- Adding significant interactions has improved our model greatly by 12.69%, our new model has a accuracy of 92.38%.


ROC Curve Logistic Regression Model
True positive rate
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
6.55
2.43
−5.82
−14.07
AUC = 0.9448



- We were able to improve the model by adding interactions between variables. The AUC is now 0.9448.
- The new model shows that the interactions between variables are significant within the model.

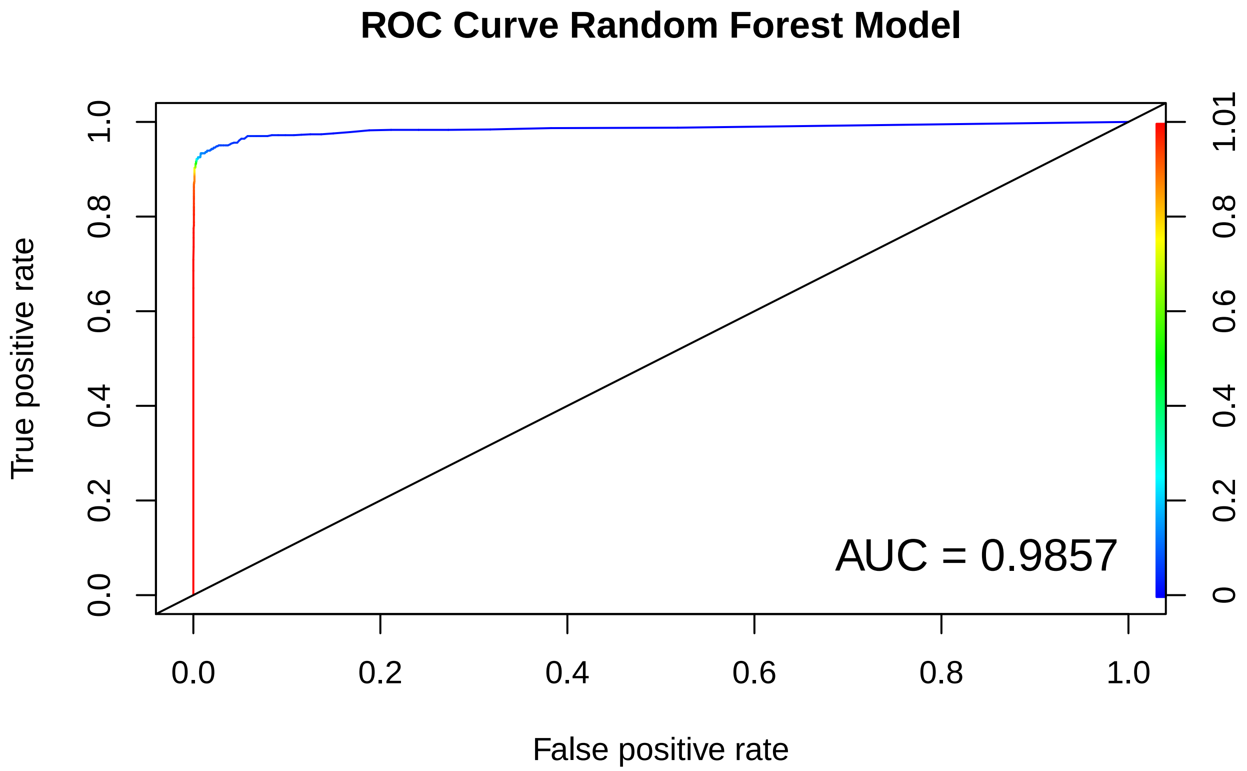
Random Forest Model

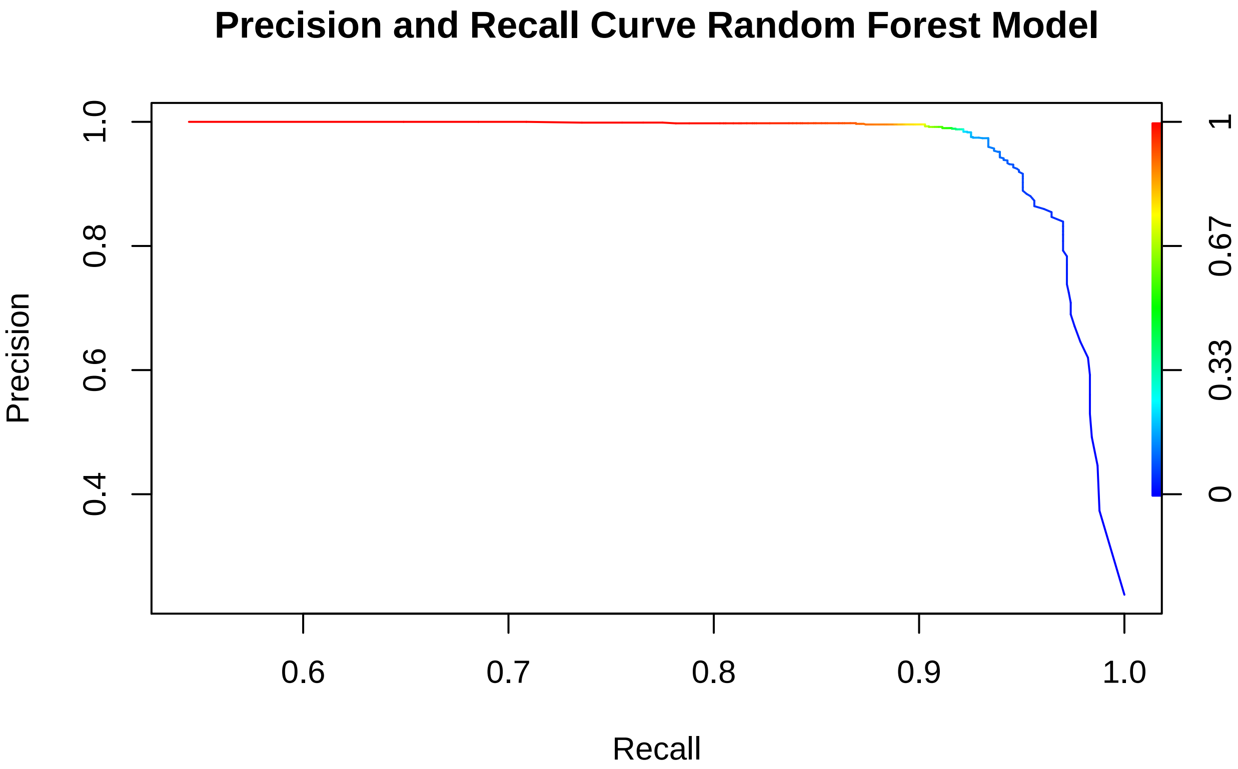
The last model we will build is using the random forest model. The random forest model produces multiple models on the training data set and averages them to create a stronger model than the basic decision tree. By averaging multiple trees, the random forest model reduces the variance in the average decision tree model.

We will use the same training and testing data set from the previous models.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 3419   91
##          1   10  980
##
##                Accuracy : 0.9776
##                  95% CI : (0.9728, 0.9817)
##     No Information Rate : 0.762
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9365
##  Mcnemar's Test P-Value : 1.716e-15
##
##             Sensitivity : 0.9971
##             Specificity : 0.9150
##          Pos Pred Value : 0.9741
##          Neg Pred Value : 0.9899
##              Prevalence : 0.7620
##          Detection Rate : 0.7598
##    Detection Prevalence : 0.7800
##       Balanced Accuracy : 0.9561
##
##        'Positive' Class : 0
##
```

This is a 0.14% accuracy improvment on our CART model with 97.76% accuracy.


ROC Curve Random Forest Model
True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0
AUC = 0.9857
0
0.2
0.4
0.6
0.8
1.01



- The random forest model is our best model. It has the highest percentage accuracy in predictions of our testing subset, also the highest ROC and the highest precision and recall plots.

Application of Logistic Regression Model

Hypothetically, if the company is curious about certain employees and would like to see if what the probability is that they will quit, they enter the employee's HR data and see what are they chances they will quit. Similarly, they can see what to offer those employees to decrease the chances of them quitting their job.

First we enter two employees who work for the company. They each have high evaluations and perform well.

```
##   Satisfaction Evaluation NumberProjects AvgMonthlyHours YearsWithCompany
## 1         0.11       0.93              7             308                4
## 2         0.90       0.98              4             264                6
##   WorkAccident Promotion  Department Salary
## 1            0         0          IT medium
## 2            0         0 product_mng medium
```

Then we insert the values into our logistic model to see what is the probability of the employee quitting.

```
##   Satisfaction Evaluation NumberProjects AvgMonthlyHours YearsWithCompany
## 1         0.11       0.93              7             308                4
## 2         0.90       0.98              4             264                6
##   WorkAccident Promotion  Department Salary       fit       se.fit
## 1            0         0          IT medium 0.9957675 0.0008916159
## 2            0         0 product_mng medium 0.9247015 0.0103458417
##   residual.scale
## 1              1
## 2              1
```

These individuals are 99.6% and 92.4% chance of quitting. Lets try to change their HR data to decrease their chances of quitting. If we decrease their number of projects, decrease their work hours, or increase their salary can we decrease their probability of quiting?

```
##   Satisfaction Evaluation NumberProjects AvgMonthlyHours YearsWithCompany
## 1         0.11       0.93              4             275                4
## 2         0.90       0.98              4             250                6
##   WorkAccident Promotion  Department Salary       fit     se.fit
## 1            0         0          IT medium 0.4374672 0.04828453
## 2            0         0 product_mng   high 0.6615020 0.05057592
##   residual.scale
## 1              1
## 2              1
```

After changing the factors, employee 1 is now 43.7% likely to quit and employee 2 is 66%. Knowing this information, employers can be better prepped for the evaluations and negotiations with their employees.

Conclusions

1. Each of our models were strong in predicting whether or not an employee will quit their job. Our strongest model was the random forest model, then the CART model and lastly the logistic regression model.
2. The employer can predict the actions of their employees with high accuracy and confidence. They can use the models to help alert whether or not an employee will quit their position.
3. Employee satifaction, evaluation, number of projects, years with company and average monthly hours are high indicators on if an employee will quit or not. Based on our preliminary analysis the employer can determine what is optimal for each of the indicators.

Recommendations

1. There are many reasons why an individual quits a job, variables that were not included in this data set. I would recommend adding other variables such as commute time or employee altercations to help train a more resilient model and to rule out extraneous uncontrollable factors like health or personal reasons.
2. Instead of using factor levels to describe salary, it would be better to use an actual number or range. This way we can predict how much salary is needed to keep an employee from quitting. By predicting the amount needed, the company will know the best salary offer the employee without overshooting and costing the company resources.
3. Lastly, I would recommend the employer to run the model on their currently employees and see which individuals are flagged as potential quitters. Then depending on if the employee is expendable or not the company should take further action to protect their assets.