Attrition Capstone

Avery Clark

January 1, 2020

Executive Summary

In this analysis, I used machine learning methods to build prediction models designed to predict what whether an employee will stay with the company (IBM) or will leave.

In this section I'll describe the dataset and summarize the goal of the project and key steps that were performed.

The data was provided by IBM and can be found on Kaggle here: https://www.kaggle.com/pavansubhasht/ibm-hr-analytics-attrition-dataset

My goal was to build a prediction model with a prediction accuracy 88%. I surpassed that goal.

I split the data into a training set (90% of data) to train the prediction models and a testing set (10% of data) to test the accuracy of the prediction model.

After running three prediction models, the highest accuracy obtained was 0.8911565 or 89.11565%. Surpassing my goal of 88% prediction accuracy.

The most effective prediction model was "Generalized Linear Model".

This report contains four sections: Executive Summary, Analysis, Results, and Conclusion.

Executive Summary describes the dataset and summarizes the goal of the project and key steps that were performed.

Analysis explains the process and techniques used, such as data cleaning, data exploration and visualization, any insights gained, and the modeling approach.

Results presents the modeling results and discusses the model performance.

Conclusion gives a brief summary of the report, its limitations and future work.

Thank you for taking the time to look at this report. I hope that you will run this code by stepping through (by pressing Ctrl + Enter) as I'm explaining it.

Analysis

In this section, I'll explain the process and techniques used, such as data cleaning, data exploration and visualization, any insights gained, and the modeling approach. You'll see these models in action in the Results section.

90% of the data was designated for training the prediction model and 10% of the data was reserved for testing the accuracy of that model's predictions.

A simple way of thinking about this is that the model (or algorithm) will learn about the data by taking in different factors and will make a prediction of which employees will stay and which will leave. Different approaches will have the model/algorithm using the factors given to it in different ways to make predictions.

The model/algorithm decides to predict a review rating “Y” based on factors “A”, “B”, and “C” (or more). Then the model/algorithm is exposed to the testing dataset to see if what it predicts as the review rating “Y” (based on the factors in the new dataset “A”, “B”, and “C”) is actually that accurate or not.

I hope that you will step through the code with me as I explain it.

You can run all of the code by clicking Run. You can run it line by line by pressing Ctrl + Enter on your keyboard. You can also highlight a section of code and run just that by clicking Run or pressing Ctrl + Enter on your keyboard.

Let’s dig in!

These next lines will install what is needed to run the code and will skip what your system already has installed.

Note: This could take a few minutes.

```r
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: caret

## Loading required package: lattice

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.6.2
```

```r
if(!require(data.table)) install.packages("data.table", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: data.table
```

```r
if(!require(dotwhisker)) install.packages("dotwhisker", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: dotwhisker

## Warning: package 'dotwhisker' was built under R version 3.6.2
```

```r
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: tidyverse

## -- Attaching packages ----------------------------------------------------------------------------- ti

## v tibble  2.1.3     v purrr   0.3.3
## v tidyr   1.0.0     v dplyr   0.8.3
## v readr   1.3.1     v stringr 1.4.0
## v tibble  2.1.3     v forcats 0.4.0

## -- Conflicts ------------------------------------------------------------------------------- tidyvers
## x dplyr::between()   masks data.table::between()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x purrr::lift()      masks caret::lift()
## x purrr::transpose() masks data.table::transpose()
```

```r
if(!require(rmarkdown)) install.packages("rmarkdown", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: rmarkdown

## Warning: package 'rmarkdown' was built under R version 3.6.2
```

```r
if(!require(readr)) install.packages("readr", repos = "http://cran.us.r-project.org")
if(!require(rpart)) install.packages("rpart", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: rpart
```

```r
if(!require(pROC)) install.packages("pROC", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: pROC

## Warning: package 'pROC' was built under R version 3.6.2

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

##
## Attaching package: 'pROC'

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

```r
if(!require(rpart.plot)) install.packages("rpart.plot", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: rpart.plot

## Warning: package 'rpart.plot' was built under R version 3.6.2
```

```r
library(caret)
library(data.table)
library(dotwhisker)
library(tidyverse)
library(rmarkdown)
library(readr)
library(rpart)
library(pROC)
library(rpart.plot)


wd <- getwd()

# Uncomment and run the next
# line to see your working directory:
# wd

setwd(wd)

# You can change this by editing the file path instead
# of using "wd".




# Now we'll download our data.

downloadedFile <- "https://raw.githubusercontent.com/AveryClark/Harvard-Attrition-Capstone/master/HR-Em

CSV_HR_Attrition <- read_csv(url(downloadedFile))
```

```
## Parsed with column specification:
## cols(
```

```
##   .default = col_double(),
##   Attrition = col_character(),
##   BusinessTravel = col_character(),
##   Department = col_character(),
##   EducationField = col_character(),
##   Gender = col_character(),
##   JobRole = col_character(),
##   MaritalStatus = col_character(),
##   Over18 = col_character(),
##   OverTime = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```r
# Let's probe the data and see what we learn.
head(CSV_HR_Attrition)
```

```
## # A tibble: 6 x 35
##     Age Attrition BusinessTravel DailyRate Department DistanceFromHome
##   <dbl> <chr>     <chr>              <dbl> <chr>                 <dbl>
## 1    41 Yes       Travel_Rarely       1102 Sales                     1
## 2    49 No        Travel_Freque~       279 Research ~                8
## 3    37 Yes       Travel_Rarely       1373 Research ~                2
## 4    33 No        Travel_Freque~      1392 Research ~                3
## 5    27 No        Travel_Rarely        591 Research ~                2
## 6    32 No        Travel_Freque~      1005 Research ~                2
## # ... with 29 more variables: Education <dbl>, EducationField <chr>,
## #   EmployeeCount <dbl>, EmployeeNumber <dbl>,
## #   EnvironmentSatisfaction <dbl>, Gender <chr>, HourlyRate <dbl>,
## #   JobInvolvement <dbl>, JobLevel <dbl>, JobRole <chr>,
## #   JobSatisfaction <dbl>, MaritalStatus <chr>, MonthlyIncome <dbl>,
## #   MonthlyRate <dbl>, NumCompaniesWorked <dbl>, Over18 <chr>,
## #   OverTime <chr>, PercentSalaryHike <dbl>, PerformanceRating <dbl>,
## #   RelationshipSatisfaction <dbl>, StandardHours <dbl>,
## #   StockOptionLevel <dbl>, TotalWorkingYears <dbl>,
## #   TrainingTimesLastYear <dbl>, WorkLifeBalance <dbl>,
## #   YearsAtCompany <dbl>, YearsInCurrentRole <dbl>,
## #   YearsSinceLastPromotion <dbl>, YearsWithCurrManager <dbl>
```

```r
tibble(CSV_HR_Attrition)
```

```
## # A tibble: 1,470 x 1
##    CSV_HR_Attritio~ $Attrition $BusinessTravel $DailyRate $Department
##               <dbl> <chr>      <chr>                <dbl> <chr>
##  1               41 Yes        Travel_Rarely         1102 Sales
##  2               49 No         Travel_Frequen~        279 Research &~
##  3               37 Yes        Travel_Rarely         1373 Research &~
##  4               33 No         Travel_Frequen~       1392 Research &~
##  5               27 No         Travel_Rarely          591 Research &~
##  6               32 No         Travel_Frequen~       1005 Research &~
##  7               59 No         Travel_Rarely         1324 Research &~
##  8               30 No         Travel_Rarely         1358 Research &~
##  9               38 No         Travel_Frequen~        216 Research &~
## 10               36 No         Travel_Rarely         1299 Research &~
## # ... with 1,460 more rows, and 30 more variables:
## #   $DistanceFromHome <dbl>, $Education <dbl>, $EducationField <chr>,
```

```
## #   $EmployeeCount <dbl>, $EmployeeNumber <dbl>,
## #   $EnvironmentSatisfaction <dbl>, $Gender <chr>, $HourlyRate <dbl>,
## #   $JobInvolvement <dbl>, $JobLevel <dbl>, $JobRole <chr>,
## #   $JobSatisfaction <dbl>, $MaritalStatus <chr>, $MonthlyIncome <dbl>,
## #   $MonthlyRate <dbl>, $NumCompaniesWorked <dbl>, $Over18 <chr>,
## #   $OverTime <chr>, $PercentSalaryHike <dbl>, $PerformanceRating <dbl>,
## #   $RelationshipSatisfaction <dbl>, $StandardHours <dbl>,
## #   $StockOptionLevel <dbl>, $TotalWorkingYears <dbl>,
## #   $TrainingTimesLastYear <dbl>, $WorkLifeBalance <dbl>,
## #   $YearsAtCompany <dbl>, $YearsInCurrentRole <dbl>,
## #   $YearsSinceLastPromotion <dbl>, $YearsWithCurrManager <dbl>
```

```r
str(CSV_HR_Attrition)
```

```
## Classes 'spec_tbl_df', 'tbl_df', 'tbl' and 'data.frame': 1470 obs. of  35 variables:
##  $ Age                     : num  41 49 37 33 27 32 59 30 38 36 ...
##  $ Attrition               : chr  "Yes" "No" "Yes" "No" ...
##  $ BusinessTravel          : chr  "Travel_Rarely" "Travel_Frequently" "Travel_Rarely" "Travel_Frequen
##  $ DailyRate               : num  1102 279 1373 1392 591 ...
##  $ Department              : chr  "Sales" "Research & Development" "Research & Development" "Research
##  $ DistanceFromHome        : num  1 8 2 3 2 2 3 24 23 27 ...
##  $ Education               : num  2 1 2 4 1 2 3 1 3 3 ...
##  $ EducationField          : chr  "Life Sciences" "Life Sciences" "Other" "Life Sciences" ...
##  $ EmployeeCount           : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ EmployeeNumber          : num  1 2 4 5 7 8 10 11 12 13 ...
##  $ EnvironmentSatisfaction : num  2 3 4 4 1 4 3 4 4 3 ...
##  $ Gender                  : chr  "Female" "Male" "Male" "Female" ...
##  $ HourlyRate              : num  94 61 92 56 40 79 81 67 44 94 ...
##  $ JobInvolvement          : num  3 2 2 3 3 3 4 3 2 3 ...
##  $ JobLevel                : num  2 2 1 1 1 1 1 1 3 2 ...
##  $ JobRole                 : chr  "Sales Executive" "Research Scientist" "Laboratory Technician" "Res
##  $ JobSatisfaction         : num  4 2 3 3 2 4 1 3 3 3 ...
##  $ MaritalStatus           : chr  "Single" "Married" "Single" "Married" ...
##  $ MonthlyIncome           : num  5993 5130 2090 2909 3468 ...
##  $ MonthlyRate             : num  19479 24907 2396 23159 16632 ...
##  $ NumCompaniesWorked      : num  8 1 6 1 9 0 4 1 0 6 ...
##  $ Over18                  : chr  "Y" "Y" "Y" "Y" ...
##  $ OverTime                : chr  "Yes" "No" "Yes" "Yes" ...
##  $ PercentSalaryHike       : num  11 23 15 11 12 13 20 22 21 13 ...
##  $ PerformanceRating       : num  3 4 3 3 3 3 4 4 4 3 ...
##  $ RelationshipSatisfaction: num  1 4 2 3 4 3 1 2 2 2 ...
##  $ StandardHours           : num  80 80 80 80 80 80 80 80 80 80 ...
##  $ StockOptionLevel        : num  0 1 0 0 1 0 3 1 0 2 ...
##  $ TotalWorkingYears       : num  8 10 7 8 6 8 12 1 10 17 ...
##  $ TrainingTimesLastYear   : num  0 3 3 3 3 2 3 2 2 3 ...
##  $ WorkLifeBalance         : num  1 3 3 3 3 2 2 3 3 2 ...
##  $ YearsAtCompany          : num  6 10 0 8 2 7 1 1 9 7 ...
##  $ YearsInCurrentRole      : num  4 7 0 7 2 7 0 0 7 7 ...
##  $ YearsSinceLastPromotion : num  0 1 0 3 2 3 0 0 1 7 ...
##  $ YearsWithCurrManager    : num  5 7 0 0 2 6 0 0 8 7 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Age = col_double(),
##   ..   Attrition = col_character(),
##   ..   BusinessTravel = col_character(),
```

```
##   ..   DailyRate = col_double(),
##   ..   Department = col_character(),
##   ..   DistanceFromHome = col_double(),
##   ..   Education = col_double(),
##   ..   EducationField = col_character(),
##   ..   EmployeeCount = col_double(),
##   ..   EmployeeNumber = col_double(),
##   ..   EnvironmentSatisfaction = col_double(),
##   ..   Gender = col_character(),
##   ..   HourlyRate = col_double(),
##   ..   JobInvolvement = col_double(),
##   ..   JobLevel = col_double(),
##   ..   JobRole = col_character(),
##   ..   JobSatisfaction = col_double(),
##   ..   MaritalStatus = col_character(),
##   ..   MonthlyIncome = col_double(),
##   ..   MonthlyRate = col_double(),
##   ..   NumCompaniesWorked = col_double(),
##   ..   Over18 = col_character(),
##   ..   OverTime = col_character(),
##   ..   PercentSalaryHike = col_double(),
##   ..   PerformanceRating = col_double(),
##   ..   RelationshipSatisfaction = col_double(),
##   ..   StandardHours = col_double(),
##   ..   StockOptionLevel = col_double(),
##   ..   TotalWorkingYears = col_double(),
##   ..   TrainingTimesLastYear = col_double(),
##   ..   WorkLifeBalance = col_double(),
##   ..   YearsAtCompany = col_double(),
##   ..   YearsInCurrentRole = col_double(),
##   ..   YearsSinceLastPromotion = col_double(),
##   ..   YearsWithCurrManager = col_double()
##   .. )
```

```
table(CSV_HR_Attrition$Attrition)
```

```
##
##    No  Yes
## 1233  237
```

```
head(CSV_HR_Attrition$Over18)
```

```
## [1] "Y" "Y" "Y" "Y" "Y" "Y"
```

```
levels(as.factor(CSV_HR_Attrition$Over18))
```

```
## [1] "Y"
```

```
levels(as.factor(CSV_HR_Attrition$EmployeeCount))
```

```
## [1] "1"
```

```
levels(as.factor(CSV_HR_Attrition$StandardHours))
```

```
## [1] "80"
```

```
# I'll remove the "Over18," "EmployeeCount," and "StandardHours" columns since
# all the values are the same in each. You can see this by looking at each column's
```

```
# values as factors. These three have only one factor each.

dropColumns <- c("Over18", "EmployeeCount", "StandardHours")
CSV_HR_Attrition <- CSV_HR_Attrition[ , !(names(CSV_HR_Attrition) %in% dropColumns)]

tibble(CSV_HR_Attrition)
```

```
## # A tibble: 1,470 x 1
##    CSV_HR_Attritio~ $Attrition $BusinessTravel $DailyRate $Department
##               <dbl> <chr>      <chr>                <dbl> <chr>
##  1               41 Yes        Travel_Rarely         1102 Sales
##  2               49 No         Travel_Frequen~        279 Research &~
##  3               37 Yes        Travel_Rarely         1373 Research &~
##  4               33 No         Travel_Frequen~       1392 Research &~
##  5               27 No         Travel_Rarely          591 Research &~
##  6               32 No         Travel_Frequen~       1005 Research &~
##  7               59 No         Travel_Rarely         1324 Research &~
##  8               30 No         Travel_Rarely         1358 Research &~
##  9               38 No         Travel_Frequen~        216 Research &~
## 10               36 No         Travel_Rarely         1299 Research &~
## # ... with 1,460 more rows, and 27 more variables:
## #   $DistanceFromHome <dbl>, $Education <dbl>, $EducationField <chr>,
## #   $EmployeeNumber <dbl>, $EnvironmentSatisfaction <dbl>, $Gender <chr>,
## #   $HourlyRate <dbl>, $JobInvolvement <dbl>, $JobLevel <dbl>,
## #   $JobRole <chr>, $JobSatisfaction <dbl>, $MaritalStatus <chr>,
## #   $MonthlyIncome <dbl>, $MonthlyRate <dbl>, $NumCompaniesWorked <dbl>,
## #   $OverTime <chr>, $PercentSalaryHike <dbl>, $PerformanceRating <dbl>,
## #   $RelationshipSatisfaction <dbl>, $StockOptionLevel <dbl>,
## #   $TotalWorkingYears <dbl>, $TrainingTimesLastYear <dbl>,
## #   $WorkLifeBalance <dbl>, $YearsAtCompany <dbl>,
## #   $YearsInCurrentRole <dbl>, $YearsSinceLastPromotion <dbl>,
## #   $YearsWithCurrManager <dbl>
```

Now I'll run a multiple regression analysis on all the data to see which variables make the biggest difference.

Factors are not allowed in the variable you're trying to predict for in multiple regression analysis, so I'll need to convert the Attrition variable into numeric form first.

```
CSV_HR_Attrition$Attrition <- as.factor(CSV_HR_Attrition$Attrition)

CSV_HR_Attrition$Attrition <- ifelse(CSV_HR_Attrition$Attrition=="Yes", 0, 1)[CSV_HR_Attrition$Attritio



allCovariatesEffectsMR <- lm(Attrition ~ Age + BusinessTravel + DailyRate + Department + DistanceFromHo
                              + Education + EducationField + EmployeeNumber + EnvironmentSatisfaction
                              + Gender + HourlyRate + JobInvolvement + JobLevel
                              + JobRole + JobSatisfaction + MaritalStatus + MonthlyIncome + MonthlyRate
                              + NumCompaniesWorked + OverTime + PercentSalaryHike + PerformanceRating
                              + RelationshipSatisfaction + StockOptionLevel + TotalWorkingYears
                              + TrainingTimesLastYear + WorkLifeBalance + YearsAtCompany + YearsInCurren
                              + YearsSinceLastPromotion + YearsWithCurrManager, data=CSV_HR_Attrition)



summary(allCovariatesEffectsMR)
```

```
##
## Call:
## lm(formula = Attrition ~ Age + BusinessTravel + DailyRate + Department +
##     DistanceFromHome + Education + EducationField + EmployeeNumber +
##     EnvironmentSatisfaction + Gender + HourlyRate + JobInvolvement +
##     JobLevel + JobRole + JobSatisfaction + MaritalStatus + MonthlyIncome +
##     MonthlyRate + NumCompaniesWorked + OverTime + PercentSalaryHike +
##     PerformanceRating + RelationshipSatisfaction + StockOptionLevel +
##     TotalWorkingYears + TrainingTimesLastYear + WorkLifeBalance +
##     YearsAtCompany + YearsInCurrentRole + YearsSinceLastPromotion +
##     YearsWithCurrManager, data = CSV_HR_Attrition)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.55266 -0.20551 -0.08396  0.08281  1.14588
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       5.626e-01  1.779e-01   3.163 0.001596
## Age                              -3.504e-03  1.327e-03  -2.640 0.008370
## BusinessTravelTravel_Frequently   1.523e-01  3.305e-02   4.609 4.41e-06
## BusinessTravelTravel_Rarely       6.561e-02  2.853e-02   2.300 0.021586
## DailyRate                        -2.698e-05  2.120e-05  -1.272 0.203414
## DepartmentResearch & Development  1.293e-01  1.171e-01   1.104 0.269643
## DepartmentSales                   1.053e-01  1.211e-01   0.869 0.384814
## DistanceFromHome                  3.624e-03  1.048e-03   3.457 0.000562
## Education                         1.909e-03  8.543e-03   0.223 0.823252
## EducationFieldLife Sciences      -1.225e-01  8.376e-02  -1.462 0.143969
## EducationFieldMarketing          -8.209e-02  8.923e-02  -0.920 0.357706
## EducationFieldMedical            -1.344e-01  8.409e-02  -1.598 0.110168
## EducationFieldOther              -1.443e-01  8.995e-02  -1.604 0.108977
## EducationFieldTechnical Degree   -2.674e-02  8.748e-02  -0.306 0.759905
## EmployeeNumber                   -7.553e-06  1.420e-05  -0.532 0.594843
## EnvironmentSatisfaction          -4.040e-02  7.800e-03  -5.179 2.55e-07
## GenderMale                        3.527e-02  1.742e-02   2.025 0.043058
## HourlyRate                       -1.688e-04  4.188e-04  -0.403 0.686901
## JobInvolvement                   -5.800e-02  1.199e-02  -4.836 1.47e-06
## JobLevel                         -5.416e-03  2.855e-02  -0.190 0.849544
## JobRoleHuman Resources            2.163e-01  1.224e-01   1.767 0.077495
## JobRoleLaboratory Technician      1.369e-01  4.001e-02   3.421 0.000642
## JobRoleManager                    5.061e-02  6.793e-02   0.745 0.456363
## JobRoleManufacturing Director     1.466e-02  3.921e-02   0.374 0.708604
## JobRoleResearch Director         -3.382e-03  6.056e-02  -0.056 0.955470
## JobRoleResearch Scientist         3.858e-02  3.960e-02   0.974 0.330155
## JobRoleSales Executive            1.017e-01  7.748e-02   1.313 0.189440
## JobRoleSales Representative       2.553e-01  8.608e-02   2.965 0.003073
## JobSatisfaction                  -3.735e-02  7.718e-03  -4.839 1.45e-06
## MaritalStatusMarried              1.323e-02  2.299e-02   0.575 0.565056
## MaritalStatusSingle               1.102e-01  3.145e-02   3.503 0.000475
## MonthlyIncome                     1.460e-06  7.600e-06   0.192 0.847726
## MonthlyRate                       4.697e-07  1.193e-06   0.394 0.693790
## NumCompaniesWorked                1.720e-02  3.807e-03   4.519 6.72e-06
## OverTimeYes                       2.105e-01  1.896e-02  11.102  < 2e-16
## PercentSalaryHike                -2.181e-03  3.675e-03  -0.594 0.552852
```

```
## PerformanceRating                  1.826e-02  3.717e-02   0.491 0.623347
## RelationshipSatisfaction          -2.330e-02  7.892e-03  -2.953 0.003202
## StockOptionLevel                  -1.654e-02  1.367e-02  -1.210 0.226380
## TotalWorkingYears                 -3.715e-03  2.417e-03  -1.537 0.124436
## TrainingTimesLastYear             -1.341e-02  6.635e-03  -2.021 0.043491
## WorkLifeBalance                   -3.137e-02  1.206e-02  -2.601 0.009384
## YearsAtCompany                     5.499e-03  2.989e-03   1.840 0.065995
## YearsInCurrentRole                -9.218e-03  3.876e-03  -2.378 0.017517
## YearsSinceLastPromotion            1.081e-02  3.416e-03   3.164 0.001588
## YearsWithCurrManager              -9.565e-03  3.971e-03  -2.408 0.016150
##
## (Intercept)                       **
## Age                               **
## BusinessTravelTravel_Frequently   ***
## BusinessTravelTravel_Rarely       *
## DailyRate
## DepartmentResearch & Development
## DepartmentSales
## DistanceFromHome                  ***
## Education
## EducationFieldLife Sciences
## EducationFieldMarketing
## EducationFieldMedical
## EducationFieldOther
## EducationFieldTechnical Degree
## EmployeeNumber
## EnvironmentSatisfaction           ***
## GenderMale                        *
## HourlyRate
## JobInvolvement                    ***
## JobLevel
## JobRoleHuman Resources            .
## JobRoleLaboratory Technician      ***
## JobRoleManager
## JobRoleManufacturing Director
## JobRoleResearch Director
## JobRoleResearch Scientist
## JobRoleSales Executive
## JobRoleSales Representative       **
## JobSatisfaction                   ***
## MaritalStatusMarried
## MaritalStatusSingle               ***
## MonthlyIncome
## MonthlyRate
## NumCompaniesWorked                ***
## OverTimeYes                       ***
## PercentSalaryHike
## PerformanceRating
## RelationshipSatisfaction          **
## StockOptionLevel
## TotalWorkingYears
## TrainingTimesLastYear             *
## WorkLifeBalance                   **
## YearsAtCompany                    .
```

```
## YearsInCurrentRole              *
## YearsSinceLastPromotion         **
## YearsWithCurrManager            *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3219 on 1424 degrees of freedom
## Multiple R-squared:  0.2578, Adjusted R-squared:  0.2343
## F-statistic: 10.99 on 45 and 1424 DF,  p-value: < 2.2e-16
```

```r
modcoef <- summary(allCovariatesEffectsMR)[["coefficients"]]
modcoef[order(modcoef[ , 4]), ]
```

```
##                                       Estimate   Std. Error     t value
## OverTimeYes                       2.105109e-01 1.896146e-02 11.10203745
## EnvironmentSatisfaction          -4.039838e-02 7.800256e-03 -5.17911016
## JobSatisfaction                  -3.734573e-02 7.717576e-03 -4.83904922
## JobInvolvement                   -5.799974e-02 1.199305e-02 -4.83611308
## BusinessTravelTravel_Frequently   1.523356e-01 3.305102e-02  4.60910532
## NumCompaniesWorked                1.720494e-02 3.807065e-03  4.51921397
## MaritalStatusSingle               1.101726e-01 3.145363e-02  3.50269960
## DistanceFromHome                  3.623923e-03 1.048184e-03  3.45733326
## JobRoleLaboratory Technician      1.368703e-01 4.000868e-02  3.42101500
## YearsSinceLastPromotion           1.080870e-02 3.415859e-03  3.16426884
## (Intercept)                       5.625943e-01 1.778818e-01  3.16274327
## JobRoleSales Representative       2.552823e-01 8.608494e-02  2.96547038
## RelationshipSatisfaction         -2.330324e-02 7.892294e-03 -2.95265763
## Age                              -3.503724e-03 1.326940e-03 -2.64045451
## WorkLifeBalance                  -3.137426e-02 1.206103e-02 -2.60129253
## YearsWithCurrManager             -9.564876e-03 3.971491e-03 -2.40838427
## YearsInCurrentRole               -9.218075e-03 3.875674e-03 -2.37844474
## BusinessTravelTravel_Rarely       6.561128e-02 2.852533e-02  2.30010596
## GenderMale                        3.526610e-02 1.741569e-02  2.02496145
## TrainingTimesLastYear            -1.340756e-02 6.634887e-03 -2.02076656
## YearsAtCompany                    5.498919e-03 2.988749e-03  1.83987321
## JobRoleHuman Resources            2.162787e-01 1.224204e-01  1.76668796
## EducationFieldOther              -1.442552e-01 8.994517e-02 -1.60381277
## EducationFieldMedical            -1.344146e-01 8.409132e-02 -1.59843611
## TotalWorkingYears                -3.715170e-03 2.416649e-03 -1.53732316
## EducationFieldLife Sciences      -1.224587e-01 8.376255e-02 -1.46197385
## JobRoleSales Executive            1.017194e-01 7.747902e-02  1.31286393
## DailyRate                        -2.698256e-05 2.120486e-05 -1.27247028
## StockOptionLevel                 -1.653885e-02 1.366554e-02 -1.21025970
## DepartmentResearch & Development  1.293380e-01 1.171204e-01  1.10431620
## JobRoleResearch Scientist         3.857533e-02 3.959955e-02  0.97413555
## EducationFieldMarketing          -8.209259e-02 8.922692e-02 -0.92004287
## DepartmentSales                   1.052571e-01 1.210785e-01  0.86932895
## JobRoleManager                    5.060928e-02 6.792715e-02  0.74505233
## PercentSalaryHike                -2.181405e-03 3.674667e-03 -0.59363344
## MaritalStatusMarried              1.322947e-02 2.298850e-02  0.57548241
## EmployeeNumber                   -7.552936e-06 1.419857e-05 -0.53195029
## PerformanceRating                 1.826019e-02 3.717322e-02  0.49121891
## HourlyRate                       -1.688342e-04 4.187907e-04 -0.40314702
## MonthlyRate                       4.696845e-07 1.192707e-06  0.39379710
## JobRoleManufacturing Director     1.465729e-02 3.921099e-02  0.37380581
```

```
## EducationFieldTechnical Degree   -2.674023e-02 8.748217e-02 -0.30566487
## Education                         1.908573e-03 8.543067e-03  0.22340602
## MonthlyIncome                     1.459656e-06 7.600158e-06  0.19205599
## JobLevel                         -5.416375e-03 2.854708e-02 -0.18973481
## JobRoleResearch Director         -3.382003e-03 6.055672e-02 -0.05584851
##                                      Pr(>|t|)
## OverTimeYes                      1.592330e-27
## EnvironmentSatisfaction          2.549019e-07
## JobSatisfaction                  1.446516e-06
## JobInvolvement                   1.467684e-06
## BusinessTravelTravel_Frequently  4.406043e-06
## NumCompaniesWorked               6.720770e-06
## MaritalStatusSingle              4.748139e-04
## DistanceFromHome                 5.616142e-04
## JobRoleLaboratory Technician     6.415342e-04
## YearsSinceLastPromotion          1.587610e-03
## (Intercept)                      1.595894e-03
## JobRoleSales Representative      3.072521e-03
## RelationshipSatisfaction         3.202139e-03
## Age                              8.369998e-03
## WorkLifeBalance                  9.383562e-03
## YearsWithCurrManager             1.614969e-02
## YearsInCurrentRole               1.751709e-02
## BusinessTravelTravel_Rarely      2.158624e-02
## GenderMale                       4.305760e-02
## TrainingTimesLastYear            4.349078e-02
## YearsAtCompany                   6.599488e-02
## JobRoleHuman Resources           7.749469e-02
## EducationFieldOther              1.089771e-01
## EducationFieldMedical            1.101678e-01
## TotalWorkingYears                1.244363e-01
## EducationFieldLife Sciences      1.439690e-01
## JobRoleSales Executive           1.894403e-01
## DailyRate                        2.034138e-01
## StockOptionLevel                 2.263801e-01
## DepartmentResearch & Development 2.696426e-01
## JobRoleResearch Scientist        3.301547e-01
## EducationFieldMarketing          3.577062e-01
## DepartmentSales                  3.848137e-01
## JobRoleManager                   4.563630e-01
## PercentSalaryHike                5.528516e-01
## MaritalStatusMarried             5.650560e-01
## EmployeeNumber                   5.948434e-01
## PerformanceRating                6.233473e-01
## HourlyRate                       6.869006e-01
## MonthlyRate                      6.937898e-01
## JobRoleManufacturing Director    7.086044e-01
## EducationFieldTechnical Degree   7.599045e-01
## Education                        8.232516e-01
## MonthlyIncome                    8.477257e-01
## JobLevel                         8.495440e-01
## JobRoleResearch Director         9.554703e-01
```

```r
topFactors <- modcoef[order(modcoef[ , 4]), ]
topFactors[1:10,4]
```

```
##                    OverTimeYes        EnvironmentSatisfaction
##                   1.592330e-27                   2.549019e-07
##                JobSatisfaction                 JobInvolvement
##                   1.446516e-06                   1.467684e-06
## BusinessTravelTravel_Frequently             NumCompaniesWorked
##                   4.406043e-06                   6.720770e-06
##            MaritalStatusSingle                DistanceFromHome
##                   4.748139e-04                   5.616142e-04
##   JobRoleLaboratory Technician        YearsSinceLastPromotion
##                   6.415342e-04                   1.587610e-03
```

```r
topFactors[1:10,0]
```

```
##
## OverTimeYes
## EnvironmentSatisfaction
## JobSatisfaction
## JobInvolvement
## BusinessTravelTravel_Frequently
## NumCompaniesWorked
## MaritalStatusSingle
## DistanceFromHome
## JobRoleLaboratory Technician
## YearsSinceLastPromotion
```

By sorting by p-value, we can see that according to our multiple reggression analysis, the factors with the greatest significance on attrition (in order) are: OverTime, EnvironmentSatisfaction, JobSatisfaction, JobInvolvement, BusinessTravel, NumCompaniesWorked, MaritalStatus, DistanceFromHome, and JobRole.

Note: When I tried to reach a higher accuracy level by using only some columns that had proven to be significant in this test, my accuracy actually decreased. So I let each type of analysis decide for itself which predictors to include from the entire list.

Now that we've seen what the most important factors for predicting attrition are according to our multiple regression analysis, let's see what they are according to a RPART (Recursive Partitioning And Regression Trees) analysis.

The RPART analysis works by splitting the data into groups like a big decision tree. It then makes its predictions per entry (or in our case, per employee) based upon where the predictors fall in its decision tree path.

Before we begin the RPATH analysis, I'd like to clean up the JobRole variable by shortening the titles to make it nicer to present in our graphs.

This will also give us separate datasets for each job role, which we'll test for differences later.

```r
head(CSV_HR_Attrition$JobRole)
```

```
## [1] "Sales Executive"       "Research Scientist"    "Laboratory Technician"
## [4] "Research Scientist"    "Laboratory Technician" "Laboratory Technician"
```

```r
jobRoleLevels <- as.factor(CSV_HR_Attrition$JobRole)
levels(jobRoleLevels)
```

```
## [1] "Healthcare Representative" "Human Resources"
## [3] "Laboratory Technician"     "Manager"
```

```
## [5] "Manufacturing Director"    "Research Director"
## [7] "Research Scientist"        "Sales Executive"
## [9] "Sales Representative"
```

```
for (i in seq(1, length(CSV_HR_Attrition$JobRole), by=1)) {

  if(CSV_HR_Attrition$JobRole[i] == "Healthcare Representative") {
    CSV_HR_Attrition$JobRole[i] <- "HC R"
    if(isFALSE(exists("dbSplit_HCRData"))) {
      dbSplit_HCRData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_HCRData <- rbind(dbSplit_HCRData, CSV_HR_Attrition[i,])
    }
  }

  else if(CSV_HR_Attrition$JobRole[i] == "Human Resources") {
    CSV_HR_Attrition$JobRole[i] <- "HR"
    if(isFALSE(exists("dbSplit_HRData"))) {
      dbSplit_HRData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_HRData <- rbind(dbSplit_HRData, CSV_HR_Attrition[i,])
    }
  }

  else if(CSV_HR_Attrition$JobRole[i] == "Laboratory Technician") {
    CSV_HR_Attrition$JobRole[i] <- "LT"
    if(isFALSE(exists("dbSplit_LTData"))) {
      dbSplit_LTData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_LTData <- rbind(dbSplit_LTData, CSV_HR_Attrition[i,])
    }
  }

  else if(CSV_HR_Attrition$JobRole[i] == "Manager") {
    CSV_HR_Attrition$JobRole[i] <- "Mgr"
    if(isFALSE(exists("dbSplit_MgrData"))) {
      dbSplit_MgrData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_MgrData <- rbind(dbSplit_MgrData, CSV_HR_Attrition[i,])
    }
  }

  else if(CSV_HR_Attrition$JobRole[i] == "Manufacturing Director") {
    CSV_HR_Attrition$JobRole[i] <- "MD"
    if(isFALSE(exists("dbSplit_MDData"))) {
      dbSplit_MDData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_MDData <- rbind(dbSplit_MDData, CSV_HR_Attrition[i,])
    }
```

```r
  }

  else if(CSV_HR_Attrition$JobRole[i] == "Research Director") {
    CSV_HR_Attrition$JobRole[i] <- "RD"
    if(isFALSE(exists("dbSplit_RDData"))) {
      dbSplit_RDData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_RDData <- rbind(dbSplit_RDData, CSV_HR_Attrition[i,])
    }
  }

  else if(CSV_HR_Attrition$JobRole[i] == "Research Scientist") {
    CSV_HR_Attrition$JobRole[i] <- "R Sci"
    if(isFALSE(exists("dbSplit_RSciData"))) {
      dbSplit_RSciData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_RSciData <- rbind(dbSplit_RSciData, CSV_HR_Attrition[i,])
    }
  }

  else if(CSV_HR_Attrition$JobRole[i] == "Sales Executive") {
    CSV_HR_Attrition$JobRole[i] <- "Sal Ex"
    if(isFALSE(exists("dbSplit_SalExData"))) {
      dbSplit_SalExData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_SalExData <- rbind(dbSplit_SalExData, CSV_HR_Attrition[i,])
    }
  }

  else if(CSV_HR_Attrition$JobRole[i] == "Sales Representative") {
    CSV_HR_Attrition$JobRole[i] <- "Sal R"
    if(isFALSE(exists("dbSplit_SalRData"))) {
      dbSplit_SalRData <- CSV_HR_Attrition[i,]
    }
    else {
      dbSplit_SalRData <- rbind(dbSplit_SalRData, CSV_HR_Attrition[i,])
    }
  }

}

jobRoleLevelsAfter <- as.factor(CSV_HR_Attrition$JobRole)
levels(jobRoleLevelsAfter)
```

```
## [1] "HC R"   "HR"     "LT"     "MD"     "Mgr"    "R Sci"  "RD"     "Sal Ex"
## [9] "Sal R"
```

```r
head(jobRoleLevelsAfter)
```

```
## [1] Sal Ex R Sci  LT     R Sci  LT     LT
## Levels: HC R HR LT MD Mgr R Sci RD Sal Ex Sal R
```

Now on to the RPATH analysis.

```
CSV_HR_Attrition$Attrition <- as.factor(CSV_HR_Attrition$Attrition)



set.seed(1, sample.kind="Rounding")
```

```
## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding'
## sampler used
```

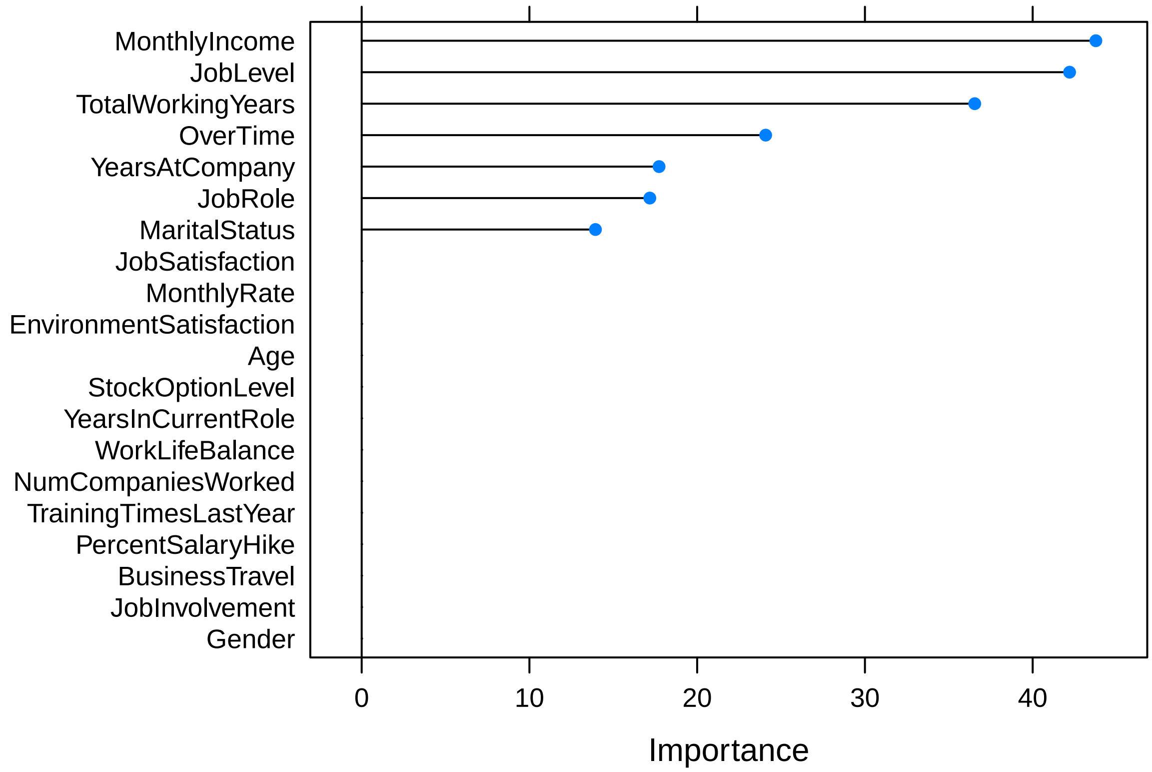
```
tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "cv", number = 2)

CSV_HR_Attrition.train.rpart <- train(
  y = CSV_HR_Attrition$Attrition,
  x = subset(CSV_HR_Attrition, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)
```

```
## Warning: Setting row names on a tibble is deprecated.
```

```
## Warning: Setting row names on a tibble is deprecated.
```

```
## Warning: Setting row names on a tibble is deprecated.
```

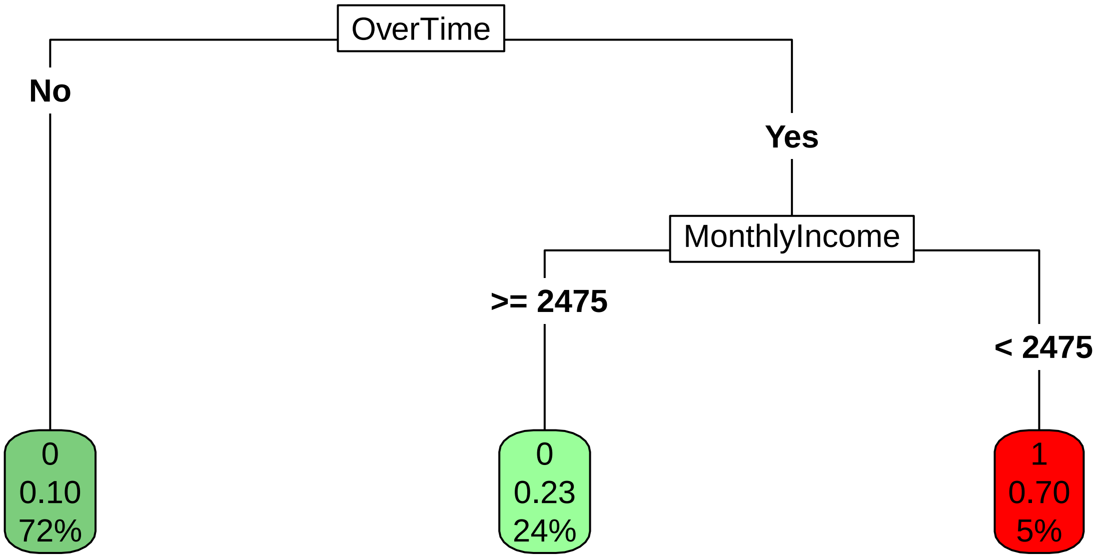
```
plot(varImp(CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



According to our RPART analysis, the most important factors in predicting attrition are:

MonthlyIncome, JobLevel, TotalWorkingYears, OverTime, YearsAtCompany, JobRole, and MaritalStatus.

```r
rpart.plot(CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "red"),
           tweak = 1)
```



According to our RPART Analysis:

If an employee does NOT work overtime, the probability they will leave the company is 10%. This group accounts for around 72% of our dataset.

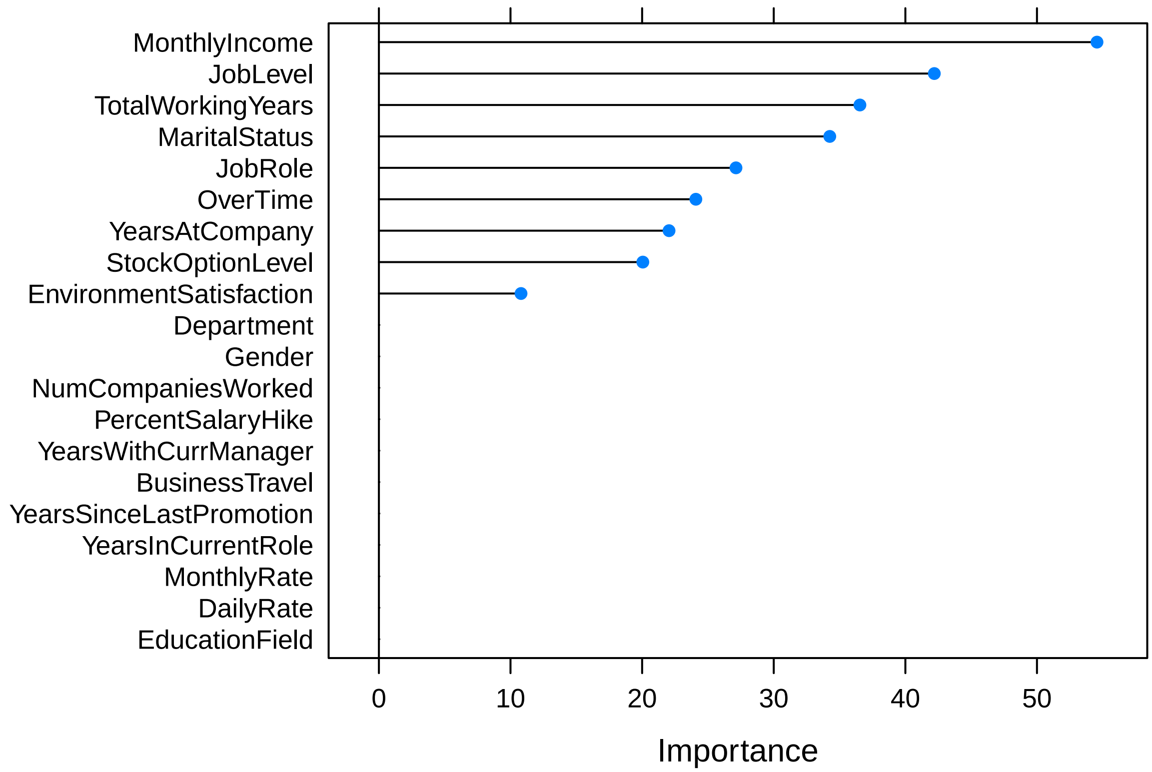
If an employee DOES work overtime and also makes $2475 or more per month, the probability they will leave the company is 23%. This group accounts for around 24% of our dataset.

If an employee DOES work overtime and also makes LESS THAN $2475 per month, the probability they will leave the company is 70%. This group accounts for around 5% of our dataset.

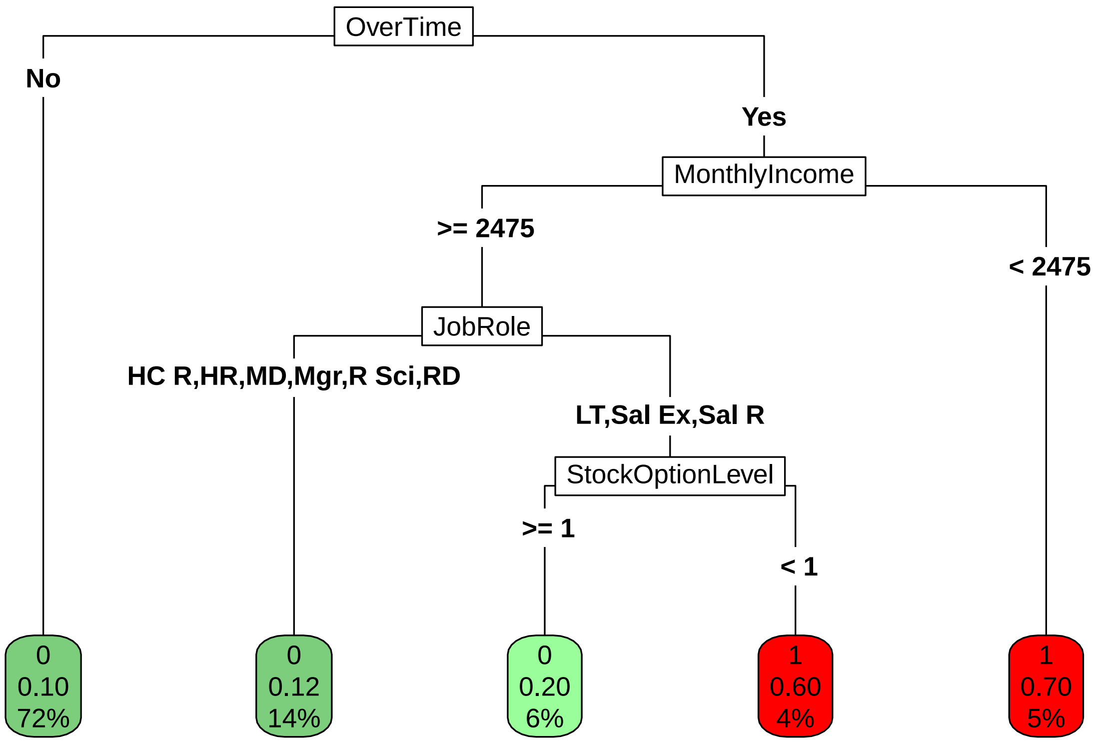
Now let's repeat the RPART analysis, but with more tests to get better detail and accuracy.

```
set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "cv", number = 6)

CSV_HR_Attrition.train.rpart <- train(
  y = CSV_HR_Attrition$Attrition,
  x = subset(CSV_HR_Attrition, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)
```

```r
plot(varImp(CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```




```r
rpart.plot(CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "red"),
           tweak = 0.95)
```

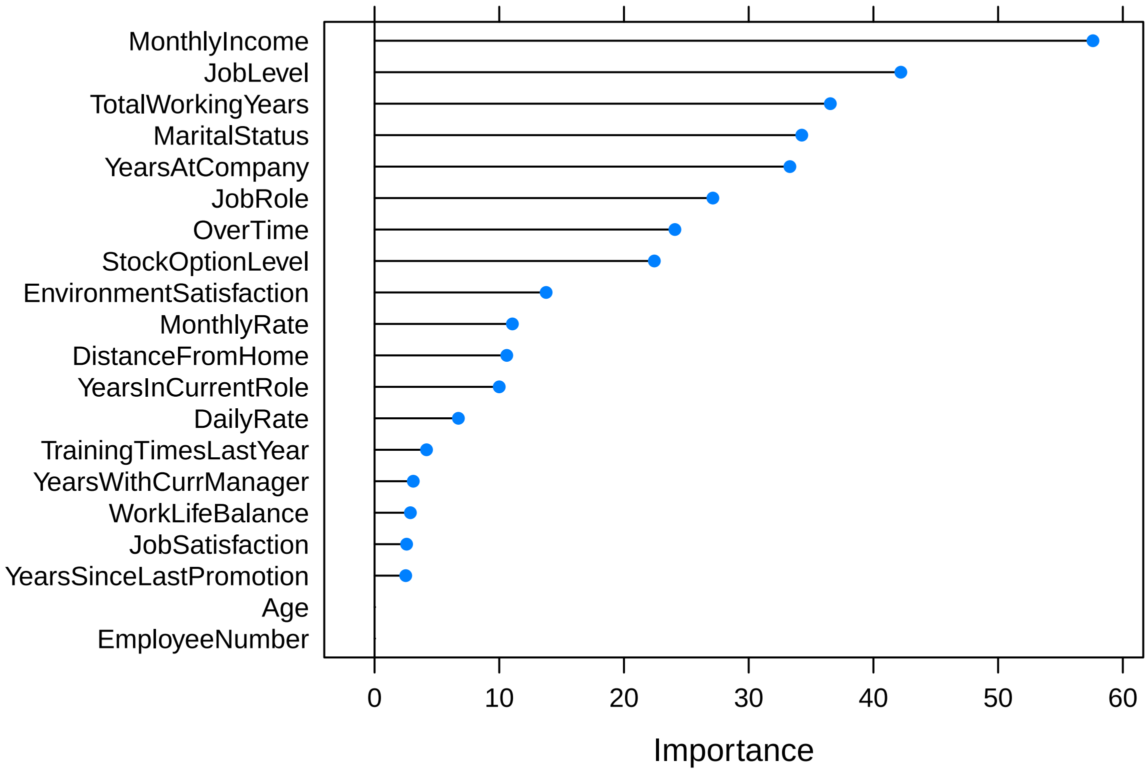


Now we can see that with more tests, our RPART analysis has similar conclusions but more detail and more accuracy.

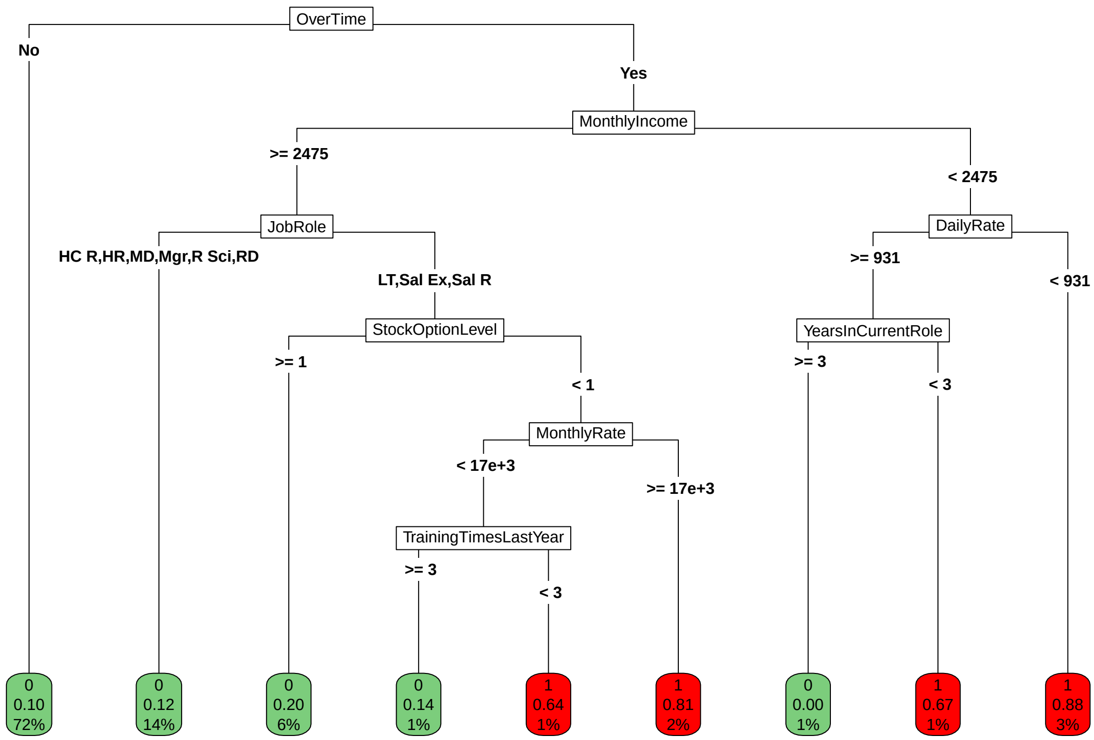
If you're confused about how to interpret this, look at the explanation of the first RPART plot above.

Just for good measure, let's see what happens when we have lots of tests.

```
set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

CSV_HR_Attrition.train.rpart <- train(
  y = CSV_HR_Attrition$Attrition,
  x = subset(CSV_HR_Attrition, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)


plot(varImp(CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```




```r
rpart.plot(CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "red"),
           tweak = 0.95)
```

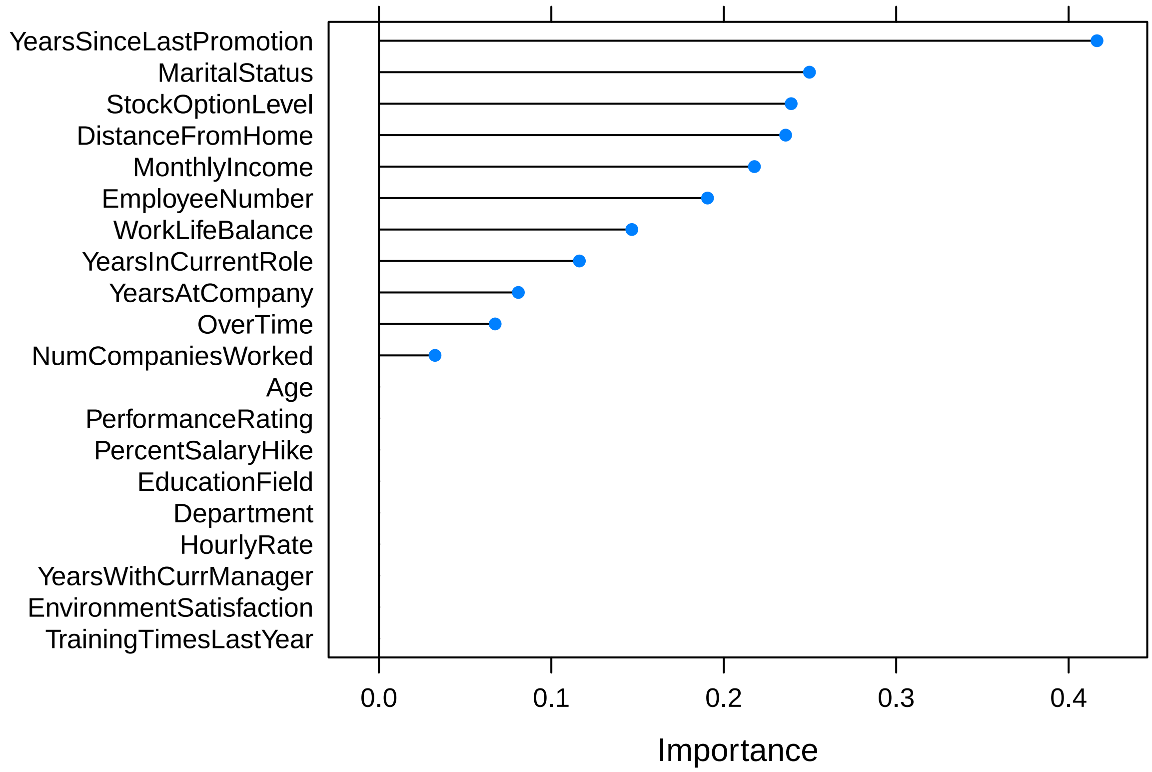


Now we can reach conclusions that have even more detail and accuracy.

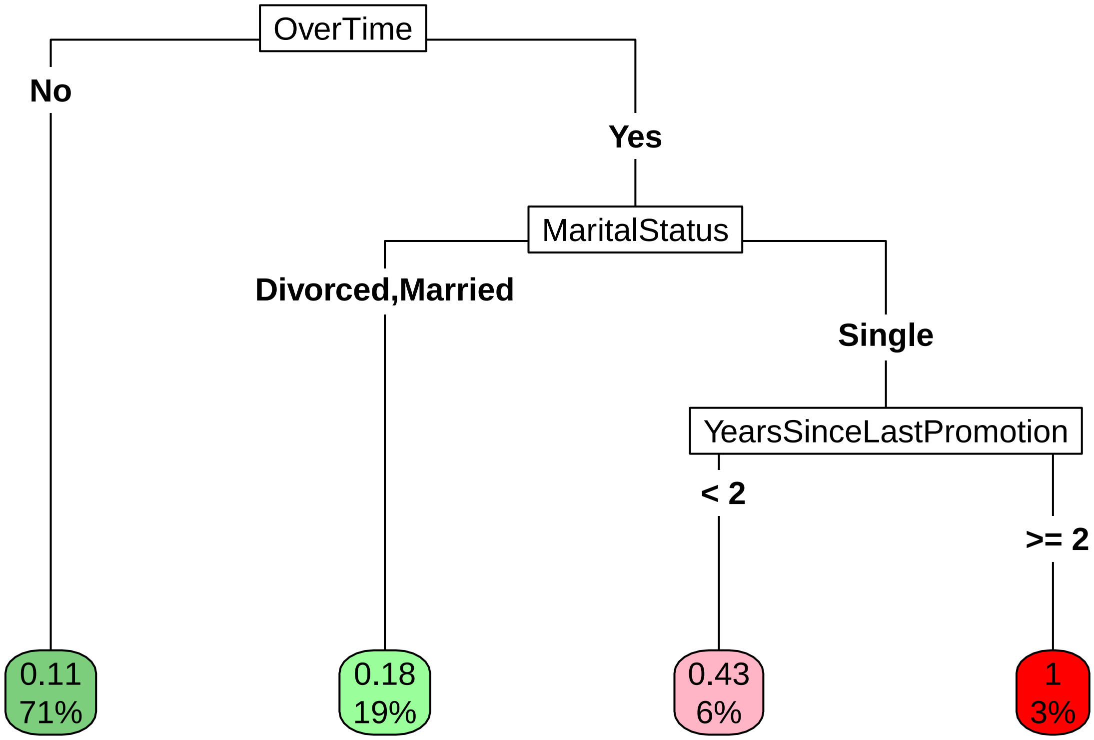
If you're confused about how to interpret this, look at the explanation of the first RPART plot above.

Now let's run the same analysis but looking at job roles separately, instead of all job roles together.

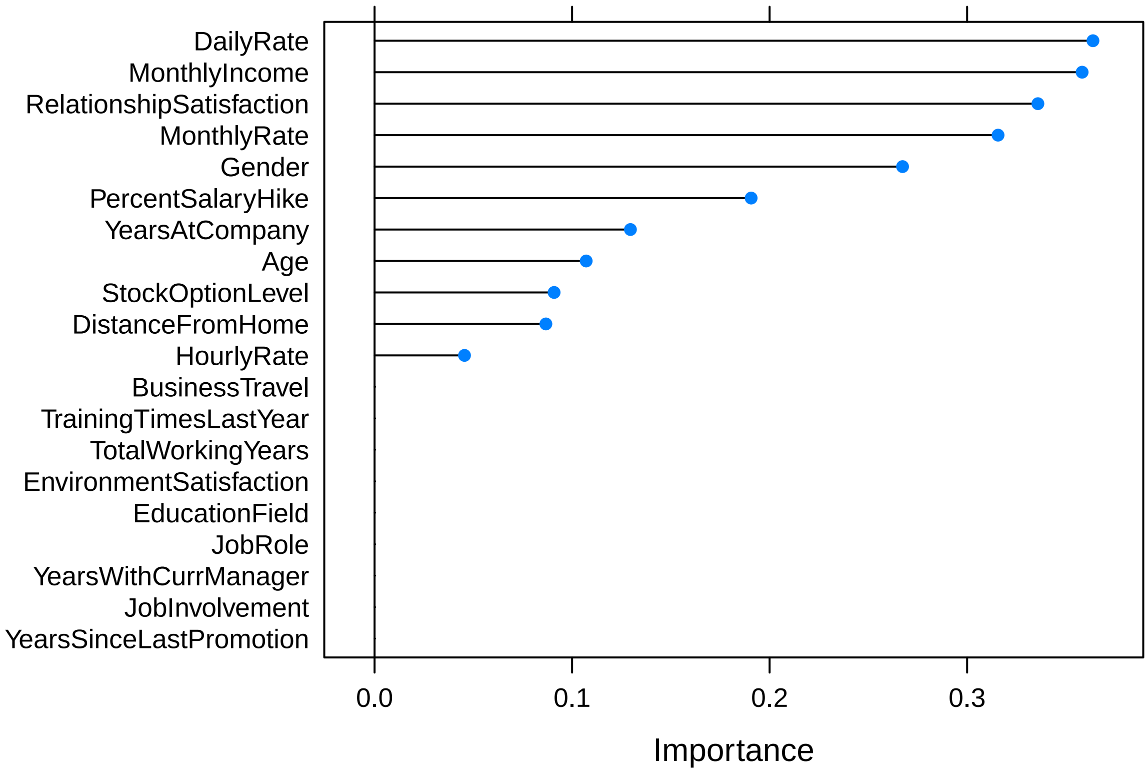
```
# Sales Executive:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_SalExData$Attrition,
  x = subset(dbSplit_SalExData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Sales Executive:

plot(varImp(jobRoleSplit_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



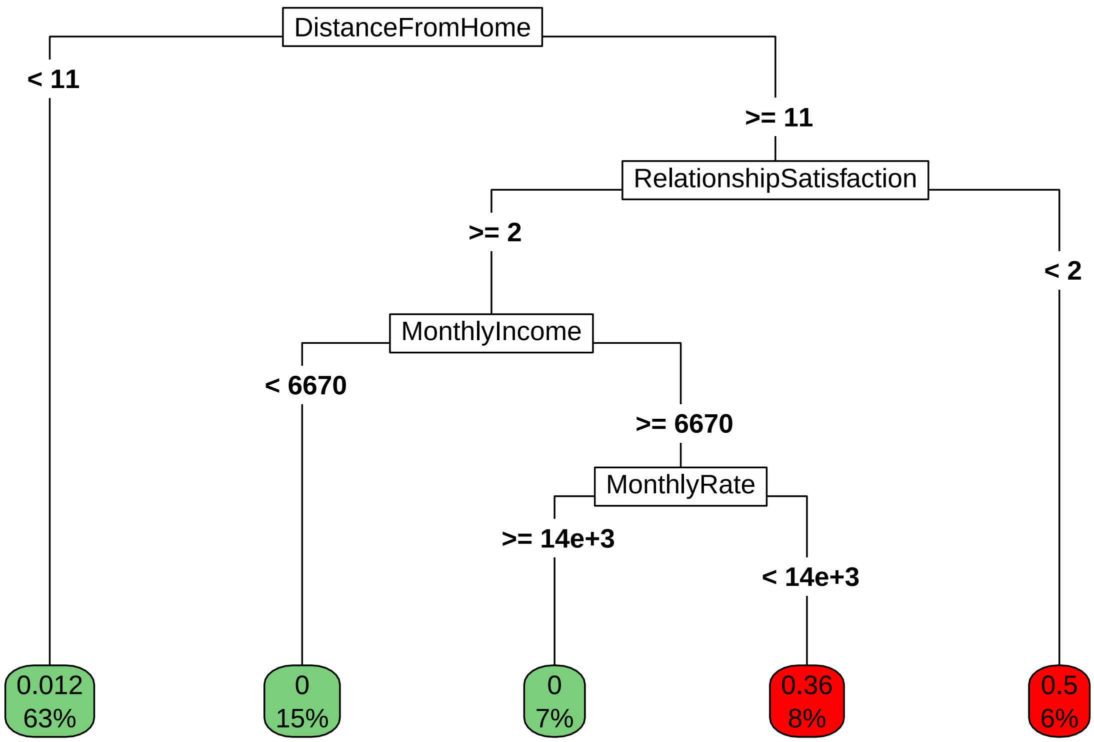

```r
# Sales Executive:

rpart.plot(jobRoleSplit_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```



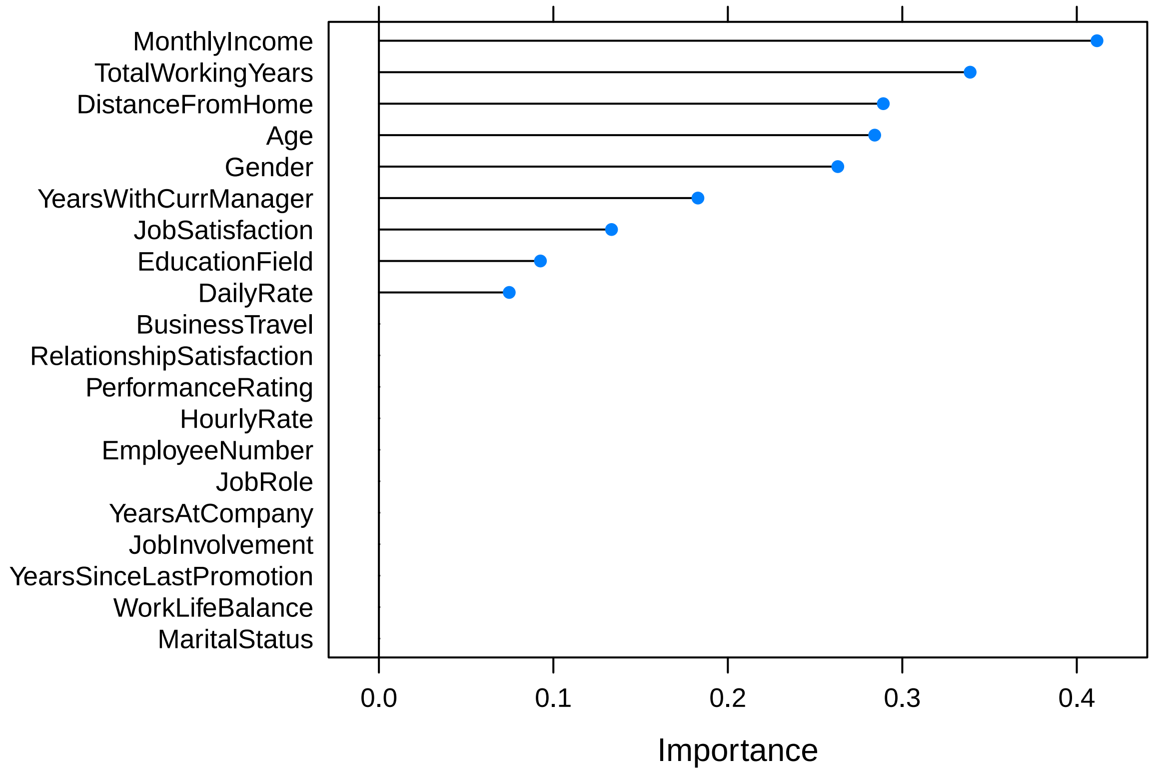

```r
# Now let's look at the Job Role of
# Healthcare Representative:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_HCRData$Attrition,
  x = subset(dbSplit_HCRData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Healthcare Representative:

plot(varImp(jobRoleSplit_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



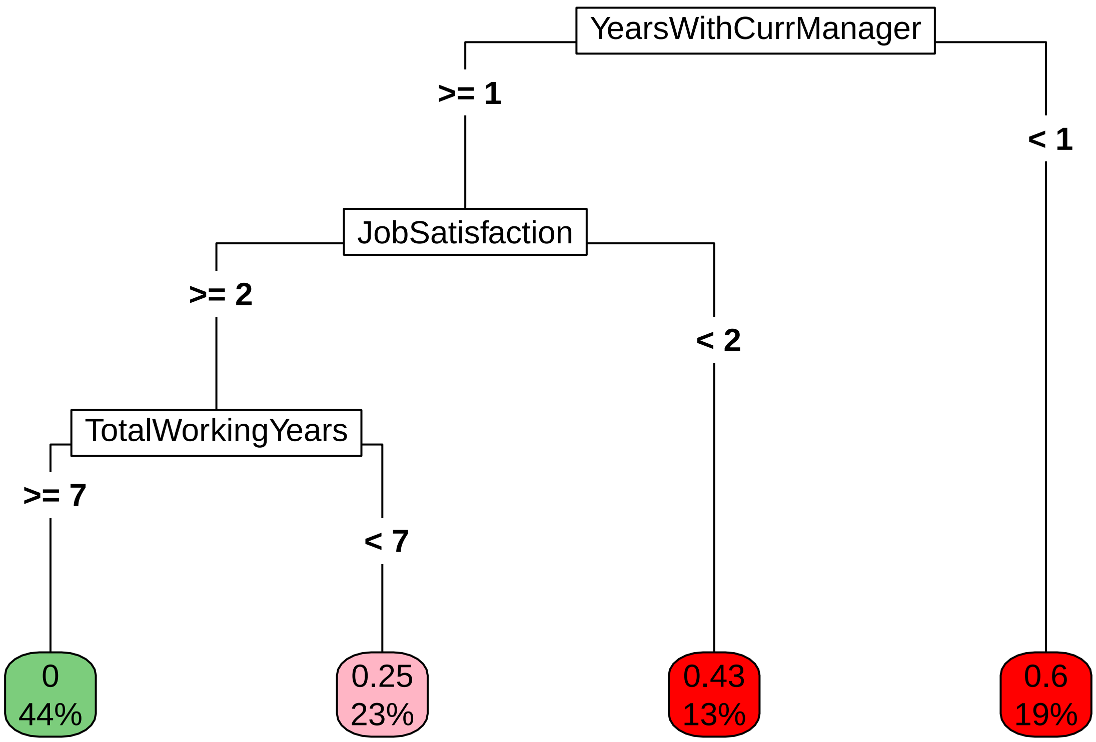

```r
# Healthcare Representative:

rpart.plot(jobRoleSplit_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```



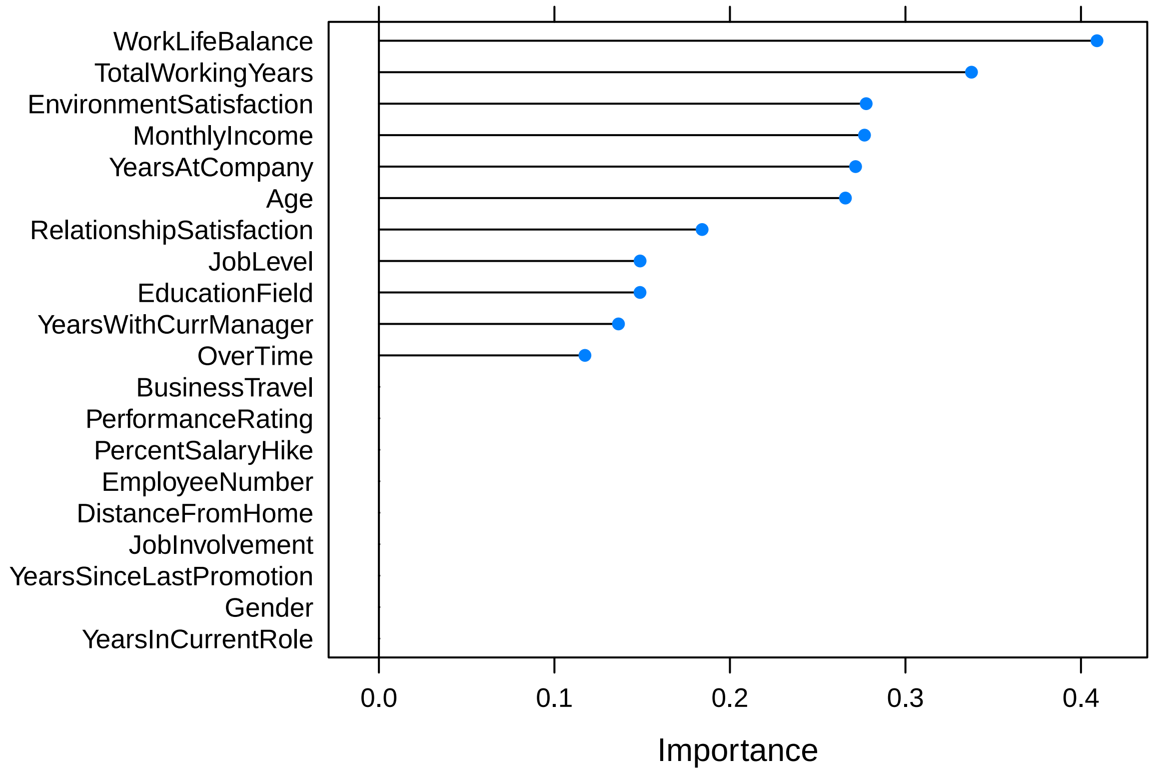

```r
# Now let's look at the Job Role of
# Human Resources:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_HRData$Attrition,
  x = subset(dbSplit_HRData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Human Resources:

plot(varImp(jobRoleSplit_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



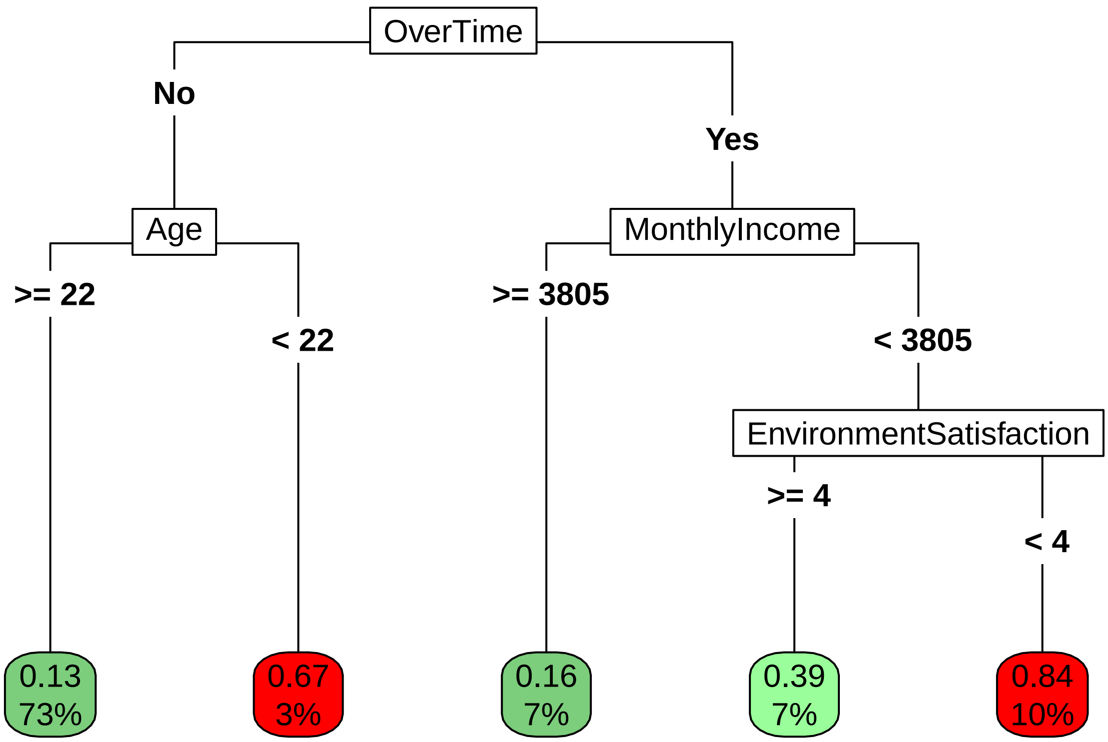

```r
# Human Resources:

rpart.plot(jobRoleSplit_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```



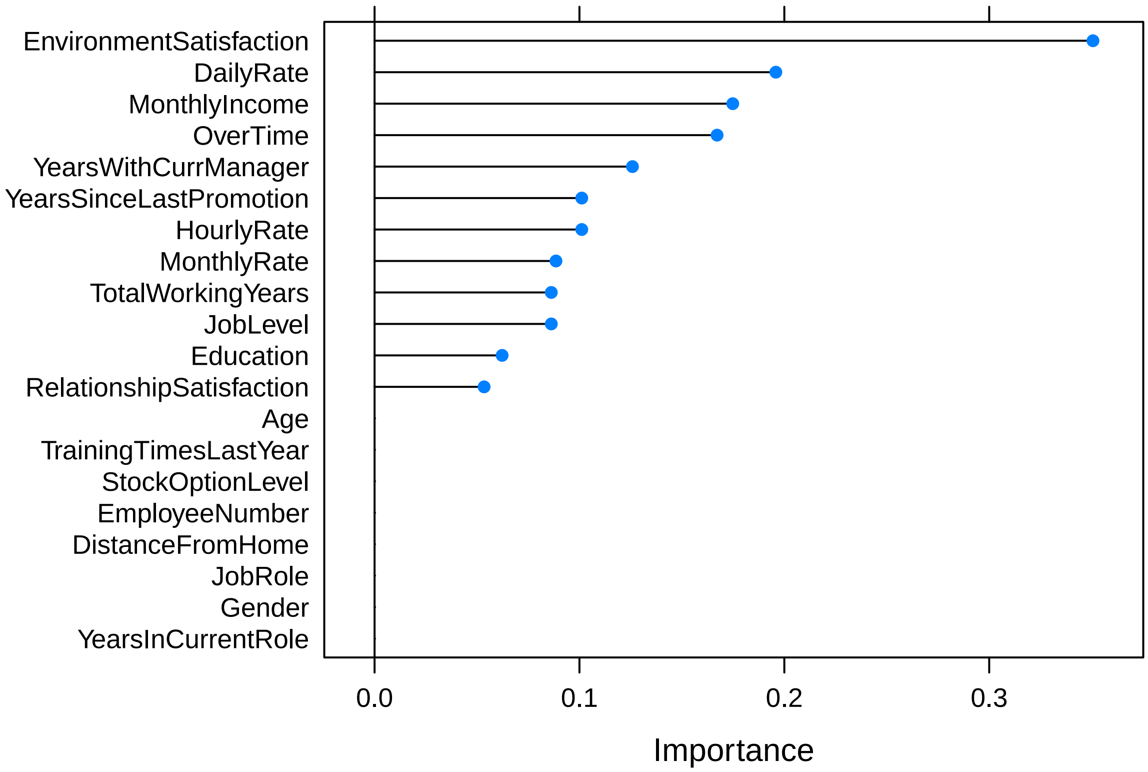

```r
# Now let's look at the Job Role of
# Laboratory Technician:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_LTData$Attrition,
  x = subset(dbSplit_LTData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Laboratory Technician:

plot(varImp(jobRoleSplit_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



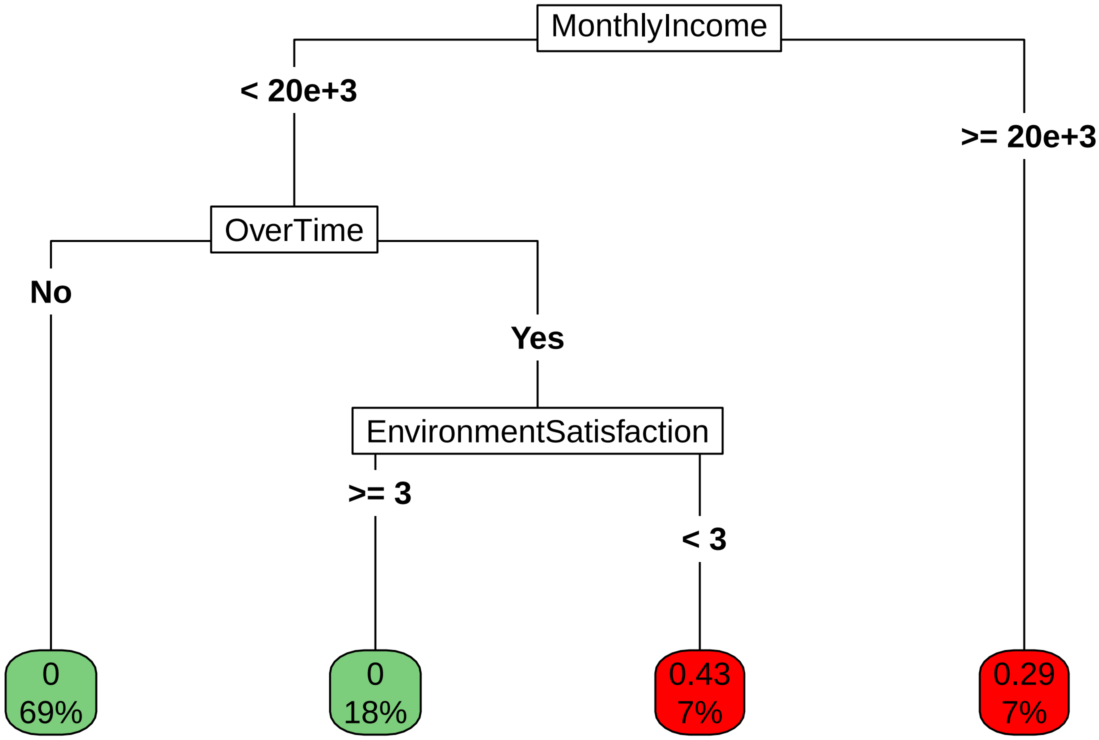

```r
# Laboratory Technician:

rpart.plot(jobRoleSplit_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```



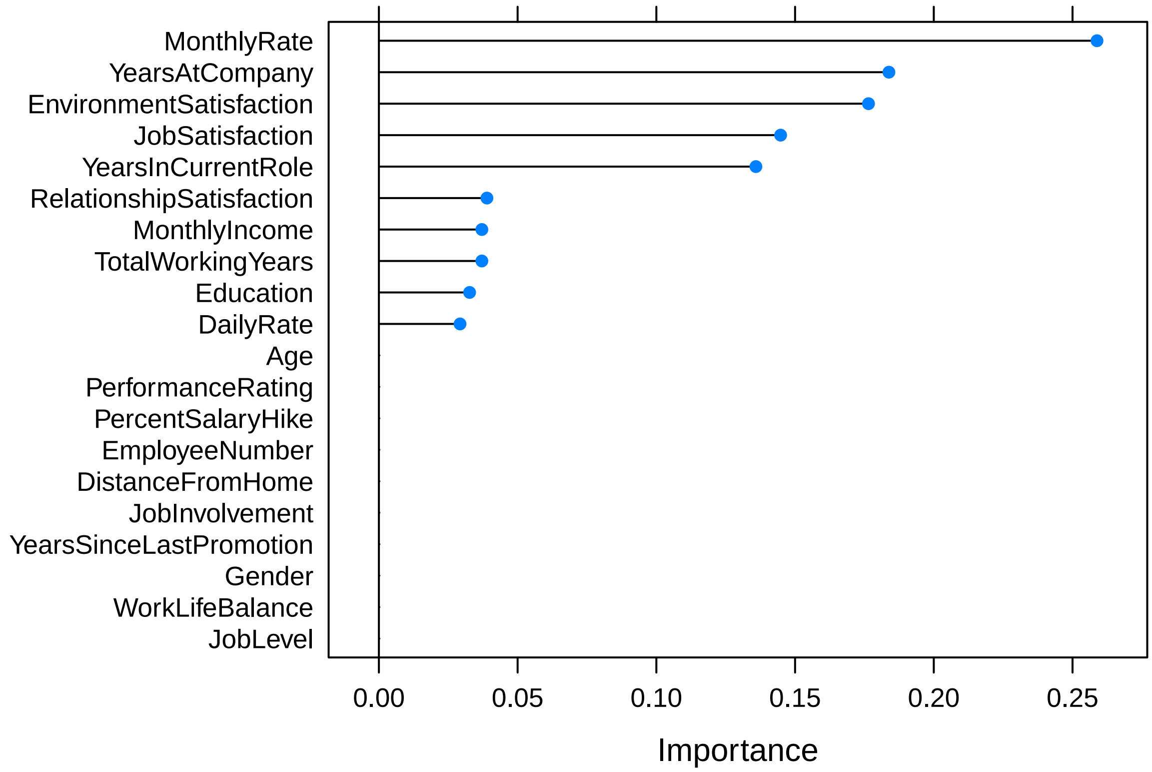

```r
# Now let's look at the Job Role of
# Manager:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_MgrData$Attrition,
  x = subset(dbSplit_MgrData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Manager:

plot(varImp(jobRoleSplit_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



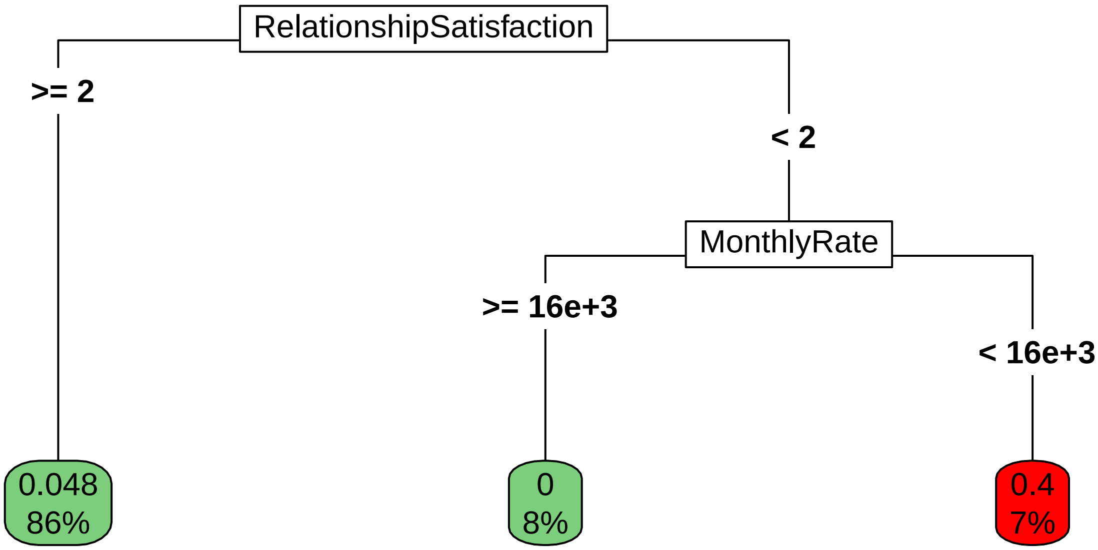

```r
# Manager:

rpart.plot(jobRoleSplit_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```



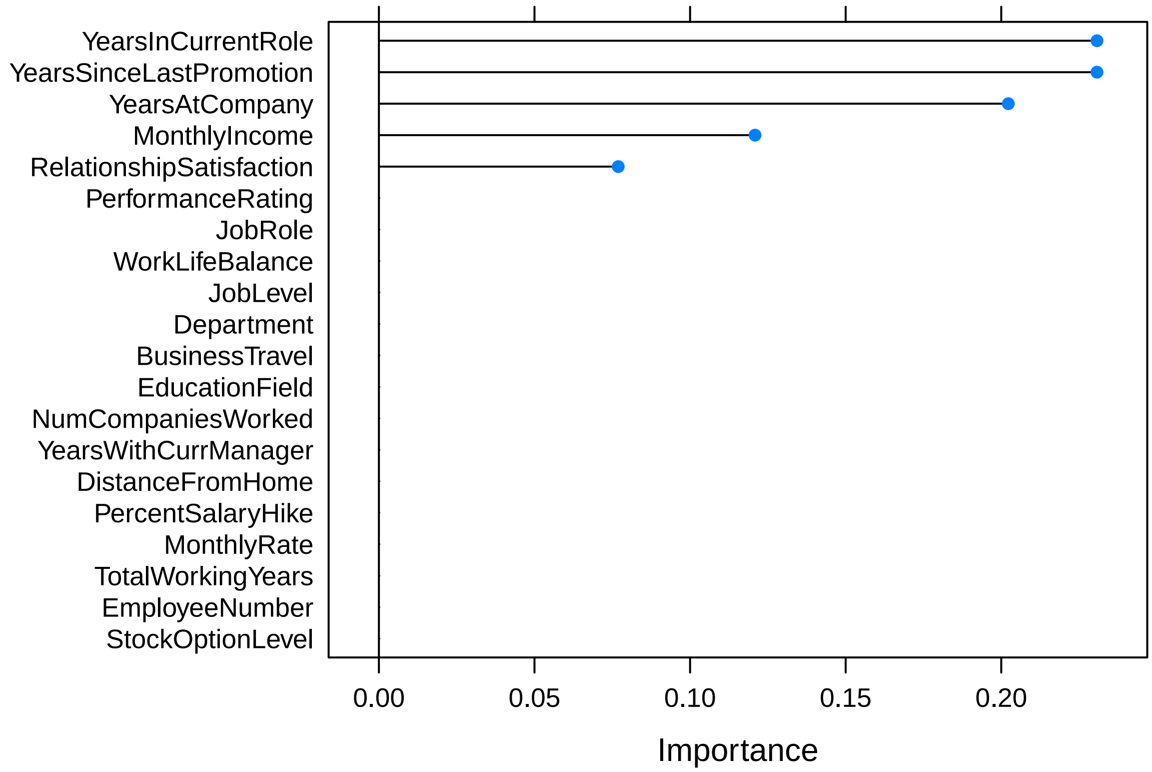

```r
# Now let's look at the Job Role of
# Manufacturing Director:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_MDData$Attrition,
  x = subset(dbSplit_MDData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Manufacturing Director:

plot(varImp(jobRoleSplit_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



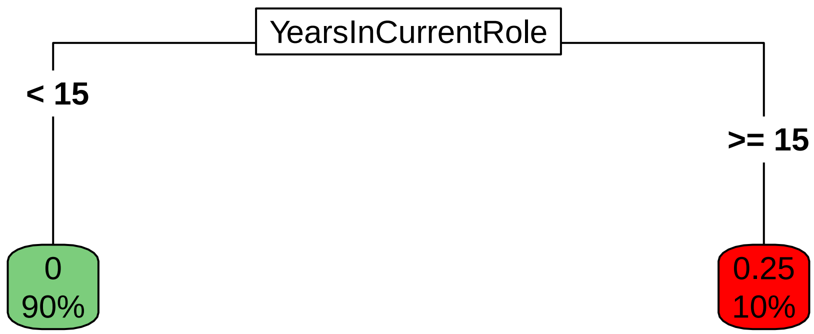

```r
# Manufacturing Director:

rpart.plot(jobRoleSplit_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```



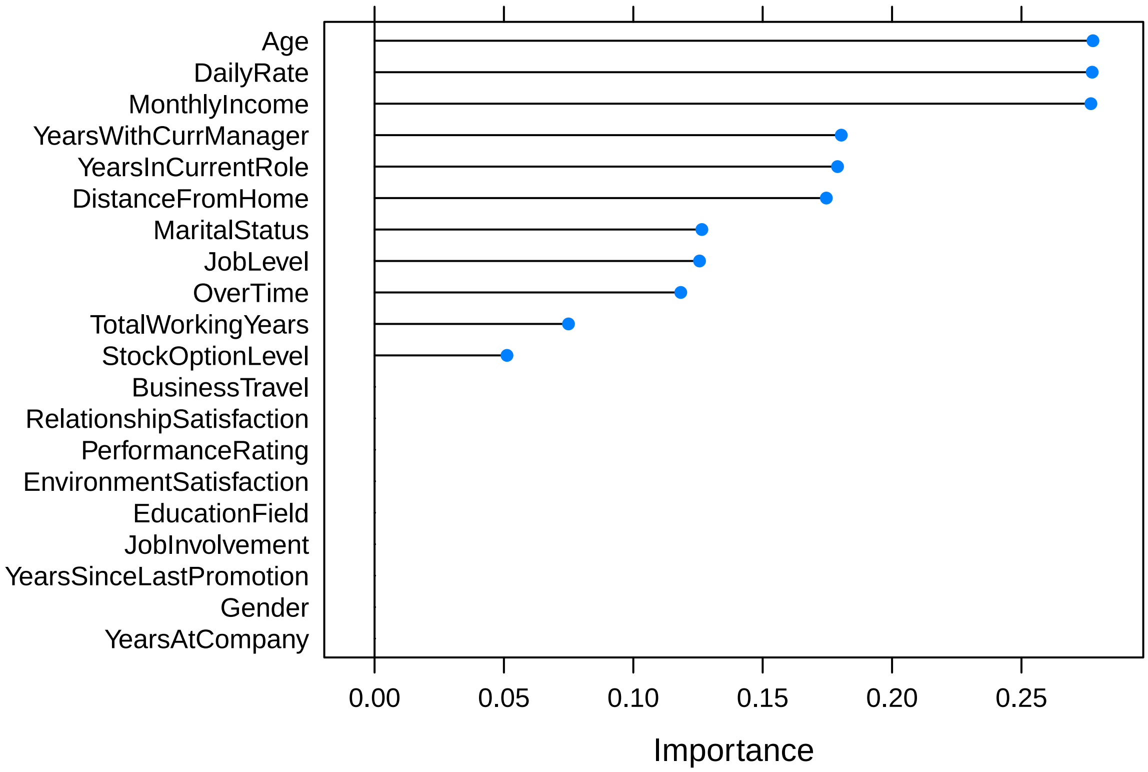

```
# Now let's look at the Job Role of
# Research Director:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit7_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_RDData$Attrition,
  x = subset(dbSplit_RDData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Research Director:

plot(varImp(jobRoleSplit7_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



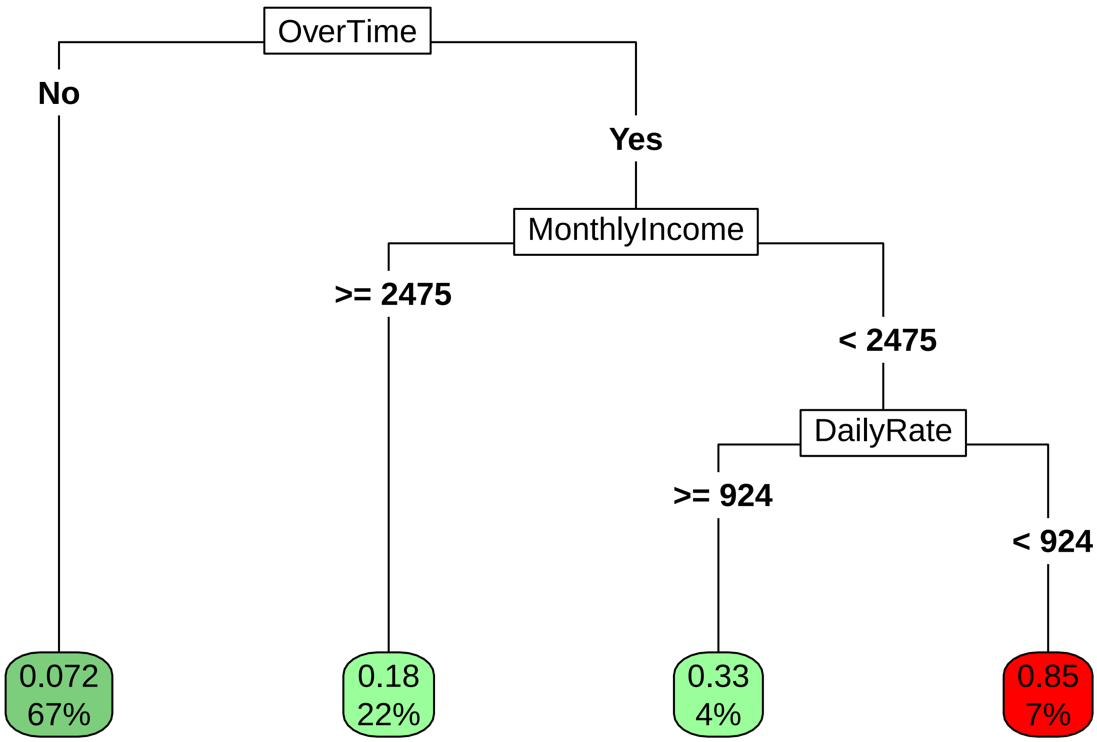

```
# Research Director:

rpart.plot(jobRoleSplit7_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```



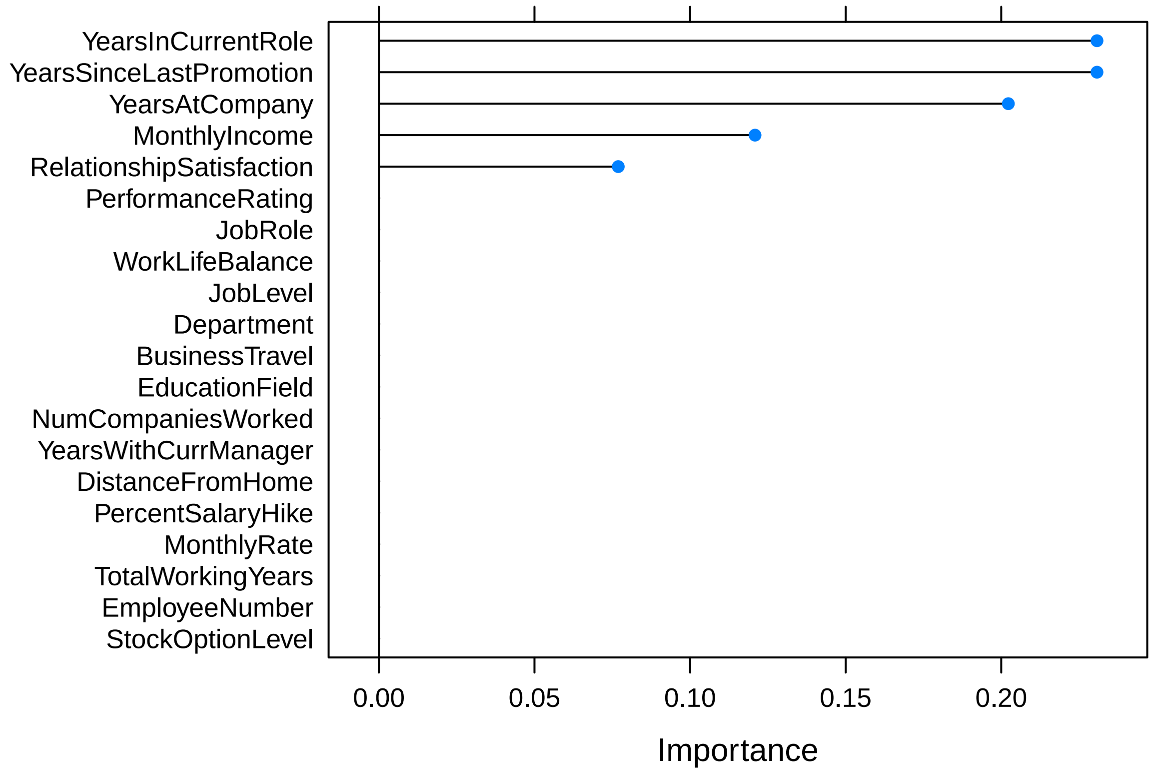

```r
# Now let's look at the Job Role of
# Research Scientist:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit8_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_RSciData$Attrition,
  x = subset(dbSplit_RSciData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Research Scientist:

plot(varImp(jobRoleSplit8_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



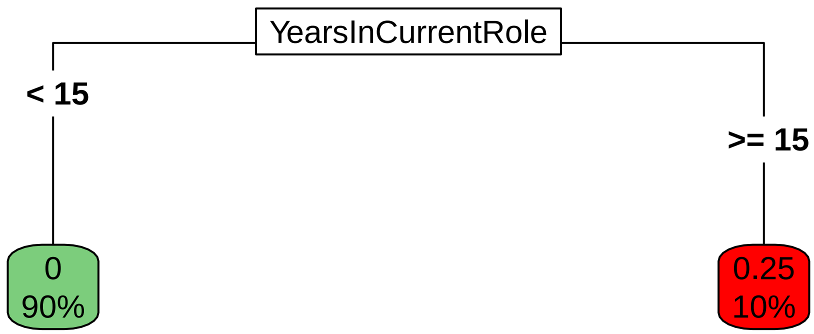

```
# Research Scientist:

rpart.plot(jobRoleSplit8_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "red1", "red"),
           tweak = 1)
```



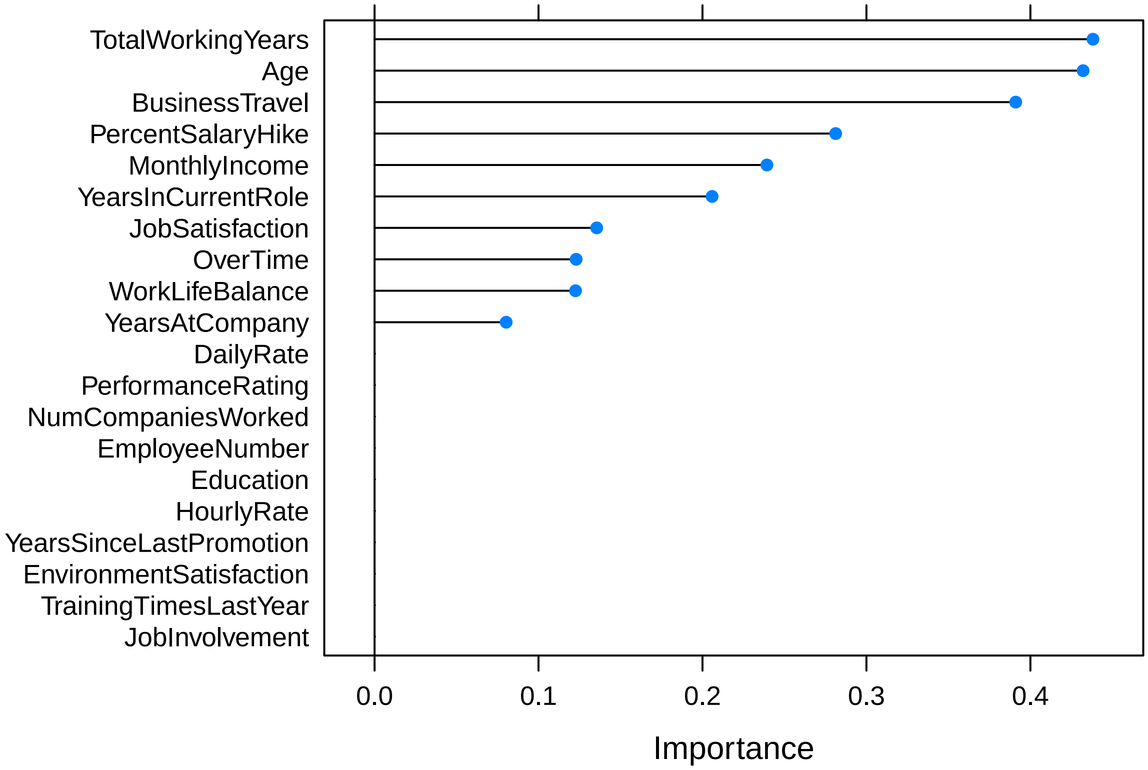

```
# Now let's look at the Job Role of
# Sales Executive:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit9_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_RDData$Attrition,
  x = subset(dbSplit_RDData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Sales Executive:

plot(varImp(jobRoleSplit9_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```



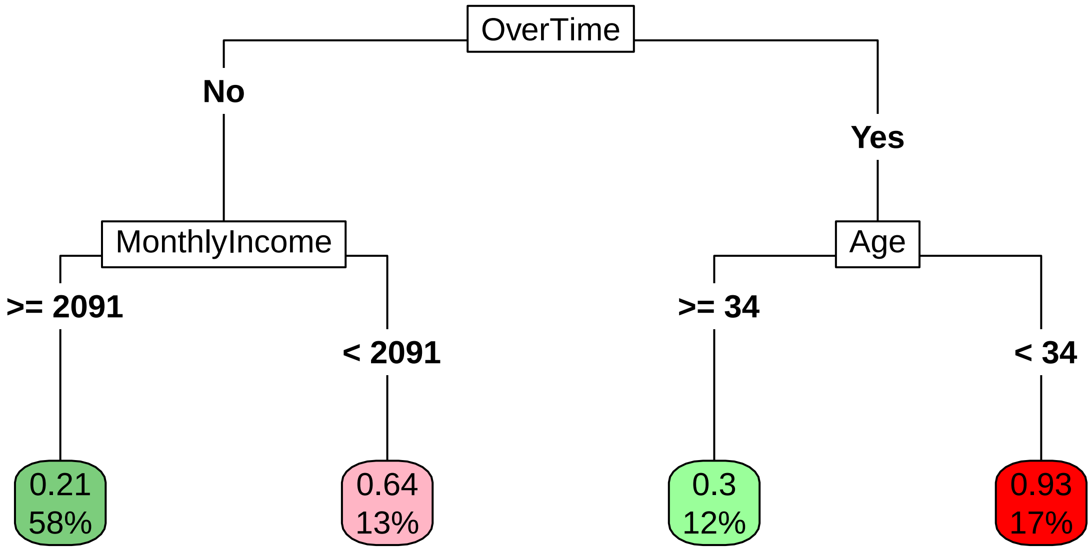

```r
# Sales Executive:

rpart.plot(jobRoleSplit9_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```




```
# Now let's look at the Job Role of
# Sales Representative:

set.seed(1, sample.kind="Rounding")

tuneGrid.rpart <- expand.grid(
  cp = seq(.01, .05, by = .005)
)

ctrl <- trainControl(method = "repeatedcv", number = 20, repeats = 5)

jobRoleSplit10_CSV_HR_Attrition.train.rpart <- train(
  y = dbSplit_SalRData$Attrition,
  x = subset(dbSplit_SalRData, select = -Attrition),
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tuneGrid.rpart,
  na.action = na.pass)

# Sales Representative:

plot(varImp(jobRoleSplit10_CSV_HR_Attrition.train.rpart, scale = FALSE), 20)
```




```r
# Sales Representative:

rpart.plot(jobRoleSplit10_CSV_HR_Attrition.train.rpart$finalModel, type = 5,
           box.palette = c("palegreen3", "palegreen1", "pink1", "red"),
           tweak = 1)
```



Now we'll split our data into a training dataset and a validation dataset.

The testing set will be 10% of the data.

```
CSV_HR_Attrition$Attrition <- ifelse(CSV_HR_Attrition$Attrition==1, 0, 1)[CSV_HR_Attrition$Attrition]
# The next line sets a random seed
# so that anyone else running this
# code can replicate the same results.
set.seed(1, sample.kind="Rounding")
# if using R 3.5 or earlier, use `set.seed(1)` instead
test_index <- createDataPartition(y = CSV_HR_Attrition$Attrition, times = 1, p = 0.1, list = FALSE)
trainingSet <- CSV_HR_Attrition[-test_index,]
testingSet <- CSV_HR_Attrition[test_index,]

head(trainingSet)
```

```
## # A tibble: 6 x 32
##     Age Attrition BusinessTravel DailyRate Department DistanceFromHome
##   <dbl>     <dbl> <chr>              <dbl> <chr>                 <dbl>
## 1    41         1 Travel_Rarely       1102 Sales                     1
## 2    49         0 Travel_Freque~       279 Research ~                8
## 3    37         1 Travel_Rarely       1373 Research ~                2
## 4    33         0 Travel_Freque~      1392 Research ~                3
## 5    27         0 Travel_Rarely        591 Research ~                2
## 6    32         0 Travel_Freque~      1005 Research ~                2
## # ... with 26 more variables: Education <dbl>, EducationField <chr>,
## #   EmployeeNumber <dbl>, EnvironmentSatisfaction <dbl>, Gender <chr>,
```

```
## #   HourlyRate <dbl>, JobInvolvement <dbl>, JobLevel <dbl>, JobRole <chr>,
## #   JobSatisfaction <dbl>, MaritalStatus <chr>, MonthlyIncome <dbl>,
## #   MonthlyRate <dbl>, NumCompaniesWorked <dbl>, OverTime <chr>,
## #   PercentSalaryHike <dbl>, PerformanceRating <dbl>,
## #   RelationshipSatisfaction <dbl>, StockOptionLevel <dbl>,
## #   TotalWorkingYears <dbl>, TrainingTimesLastYear <dbl>,
## #   WorkLifeBalance <dbl>, YearsAtCompany <dbl>, YearsInCurrentRole <dbl>,
## #   YearsSinceLastPromotion <dbl>, YearsWithCurrManager <dbl>
```

```r
tibble(trainingSet)
```

```
## # A tibble: 1,323 x 1
##    trainingSet$Age $Attrition $BusinessTravel $DailyRate $Department
##              <dbl>      <dbl> <chr>                <dbl> <chr>
##  1              41          1 Travel_Rarely         1102 Sales
##  2              49          0 Travel_Frequen~        279 Research &~
##  3              37          1 Travel_Rarely         1373 Research &~
##  4              33          0 Travel_Frequen~       1392 Research &~
##  5              27          0 Travel_Rarely          591 Research &~
##  6              32          0 Travel_Frequen~       1005 Research &~
##  7              59          0 Travel_Rarely         1324 Research &~
##  8              30          0 Travel_Rarely         1358 Research &~
##  9              38          0 Travel_Frequen~        216 Research &~
## 10              36          0 Travel_Rarely         1299 Research &~
## # ... with 1,313 more rows, and 27 more variables:
## #   $DistanceFromHome <dbl>, $Education <dbl>, $EducationField <chr>,
## #   $EmployeeNumber <dbl>, $EnvironmentSatisfaction <dbl>, $Gender <chr>,
## #   $HourlyRate <dbl>, $JobInvolvement <dbl>, $JobLevel <dbl>,
## #   $JobRole <chr>, $JobSatisfaction <dbl>, $MaritalStatus <chr>,
## #   $MonthlyIncome <dbl>, $MonthlyRate <dbl>, $NumCompaniesWorked <dbl>,
## #   $OverTime <chr>, $PercentSalaryHike <dbl>, $PerformanceRating <dbl>,
## #   $RelationshipSatisfaction <dbl>, $StockOptionLevel <dbl>,
## #   $TotalWorkingYears <dbl>, $TrainingTimesLastYear <dbl>,
## #   $WorkLifeBalance <dbl>, $YearsAtCompany <dbl>,
## #   $YearsInCurrentRole <dbl>, $YearsSinceLastPromotion <dbl>,
## #   $YearsWithCurrManager <dbl>
```

```r
str(trainingSet)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    1323 obs. of  32 variables:
##  $ Age                     : num  41 49 37 33 27 32 59 30 38 36 ...
##  $ Attrition               : num  1 0 1 0 0 0 0 0 0 0 ...
##  $ BusinessTravel          : chr  "Travel_Rarely" "Travel_Frequently" "Travel_Rarely" "Travel_Frequen
##  $ DailyRate               : num  1102 279 1373 1392 591 ...
##  $ Department              : chr  "Sales" "Research & Development" "Research & Development" "Research
##  $ DistanceFromHome        : num  1 8 2 3 2 2 3 24 23 27 ...
##  $ Education               : num  2 1 2 4 1 2 3 1 3 3 ...
##  $ EducationField          : chr  "Life Sciences" "Life Sciences" "Other" "Life Sciences" ...
##  $ EmployeeNumber          : num  1 2 4 5 7 8 10 11 12 13 ...
##  $ EnvironmentSatisfaction : num  2 3 4 4 1 4 3 4 4 3 ...
##  $ Gender                  : chr  "Female" "Male" "Male" "Female" ...
##  $ HourlyRate              : num  94 61 92 56 40 79 81 67 44 94 ...
##  $ JobInvolvement          : num  3 2 2 3 3 3 4 3 2 3 ...
##  $ JobLevel                : num  2 2 1 1 1 1 1 1 3 2 ...
##  $ JobRole                 : chr  "Sal Ex" "R Sci" "LT" "R Sci" ...
```

```
## $ JobSatisfaction         : num  4 2 3 3 2 4 1 3 3 3 ...
## $ MaritalStatus           : chr  "Single" "Married" "Single" "Married" ...
## $ MonthlyIncome           : num  5993 5130 2090 2909 3468 ...
## $ MonthlyRate             : num  19479 24907 2396 23159 16632 ...
## $ NumCompaniesWorked      : num  8 1 6 1 9 0 4 1 0 6 ...
## $ OverTime                : chr  "Yes" "No" "Yes" "Yes" ...
## $ PercentSalaryHike       : num  11 23 15 11 12 13 20 22 21 13 ...
## $ PerformanceRating       : num  3 4 3 3 3 3 4 4 4 3 ...
## $ RelationshipSatisfaction: num  1 4 2 3 4 3 1 2 2 2 ...
## $ StockOptionLevel        : num  0 1 0 0 1 0 3 1 0 2 ...
## $ TotalWorkingYears       : num  8 10 7 8 6 8 12 1 10 17 ...
## $ TrainingTimesLastYear   : num  0 3 3 3 3 2 3 2 2 3 ...
## $ WorkLifeBalance         : num  1 3 3 3 3 2 2 3 3 2 ...
## $ YearsAtCompany          : num  6 10 0 8 2 7 1 1 9 7 ...
## $ YearsInCurrentRole      : num  4 7 0 7 2 7 0 0 7 7 ...
## $ YearsSinceLastPromotion : num  0 1 0 3 2 3 0 0 1 7 ...
## $ YearsWithCurrManager    : num  5 7 0 0 2 6 0 0 8 7 ...
```

```r
head(testingSet)
```

```
## # A tibble: 6 x 32
##     Age Attrition BusinessTravel DailyRate Department DistanceFromHome
##   <dbl>     <dbl> <chr>              <dbl> <chr>                 <dbl>
## 1    22         0 Non-Travel          1123 Research ~               16
## 2    38         0 Travel_Rarely        371 Research ~                2
## 3    39         1 Travel_Rarely        895 Sales                     5
## 4    37         0 Travel_Rarely        408 Research ~               19
## 5    35         0 Travel_Rarely       1214 Research ~                1
## 6    40         0 Travel_Freque~       530 Research ~                1
## # ... with 26 more variables: Education <dbl>, EducationField <chr>,
## #   EmployeeNumber <dbl>, EnvironmentSatisfaction <dbl>, Gender <chr>,
## #   HourlyRate <dbl>, JobInvolvement <dbl>, JobLevel <dbl>, JobRole <chr>,
## #   JobSatisfaction <dbl>, MaritalStatus <chr>, MonthlyIncome <dbl>,
## #   MonthlyRate <dbl>, NumCompaniesWorked <dbl>, OverTime <chr>,
## #   PercentSalaryHike <dbl>, PerformanceRating <dbl>,
## #   RelationshipSatisfaction <dbl>, StockOptionLevel <dbl>,
## #   TotalWorkingYears <dbl>, TrainingTimesLastYear <dbl>,
## #   WorkLifeBalance <dbl>, YearsAtCompany <dbl>, YearsInCurrentRole <dbl>,
## #   YearsSinceLastPromotion <dbl>, YearsWithCurrManager <dbl>
```

```r
tibble(testingSet)
```

```
## # A tibble: 147 x 1
##    testingSet$Age $Attrition $BusinessTravel $DailyRate $Department
##             <dbl>      <dbl> <chr>                <dbl> <chr>
##  1             22          0 Non-Travel            1123 Research &~
##  2             38          0 Travel_Rarely          371 Research &~
##  3             39          1 Travel_Rarely          895 Sales
##  4             37          0 Travel_Rarely          408 Research &~
##  5             35          0 Travel_Rarely         1214 Research &~
##  6             40          0 Travel_Frequen~        530 Research &~
##  7             37          1 Travel_Rarely          807 Human Reso~
##  8             34          0 Travel_Rarely          665 Research &~
##  9             36          0 Travel_Rarely          922 Research &~
## 10             30          0 Travel_Rarely         1240 Human Reso~
```

```
## # ... with 137 more rows, and 27 more variables: $DistanceFromHome <dbl>,
## #   $Education <dbl>, $EducationField <chr>, $EmployeeNumber <dbl>,
## #   $EnvironmentSatisfaction <dbl>, $Gender <chr>, $HourlyRate <dbl>,
## #   $JobInvolvement <dbl>, $JobLevel <dbl>, $JobRole <chr>,
## #   $JobSatisfaction <dbl>, $MaritalStatus <chr>, $MonthlyIncome <dbl>,
## #   $MonthlyRate <dbl>, $NumCompaniesWorked <dbl>, $OverTime <chr>,
## #   $PercentSalaryHike <dbl>, $PerformanceRating <dbl>,
## #   $RelationshipSatisfaction <dbl>, $StockOptionLevel <dbl>,
## #   $TotalWorkingYears <dbl>, $TrainingTimesLastYear <dbl>,
## #   $WorkLifeBalance <dbl>, $YearsAtCompany <dbl>,
## #   $YearsInCurrentRole <dbl>, $YearsSinceLastPromotion <dbl>,
## #   $YearsWithCurrManager <dbl>
```

```r
str(testingSet)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    147 obs. of  32 variables:
##  $ Age                     : num  22 38 39 37 35 40 37 34 36 30 ...
##  $ Attrition               : num  0 0 1 0 0 0 1 0 0 0 ...
##  $ BusinessTravel          : chr  "Non-Travel" "Travel_Rarely" "Travel_Rarely" "Travel_Rarely" ...
##  $ DailyRate               : num  1123 371 895 408 1214 ...
##  $ Department              : chr  "Research & Development" "Research & Development" "Sales" "Researc
##  $ DistanceFromHome        : num  16 2 5 19 1 1 6 6 3 9 ...
##  $ Education               : num  2 3 3 2 3 4 4 4 2 3 ...
##  $ EducationField          : chr  "Medical" "Life Sciences" "Technical Degree" "Life Sciences" ...
##  $ EmployeeNumber          : num  22 24 42 61 105 119 133 138 155 184 ...
##  $ EnvironmentSatisfaction : num  4 4 4 2 2 3 3 1 1 3 ...
##  $ Gender                  : chr  "Male" "Male" "Male" "Male" ...
##  $ HourlyRate              : num  96 45 56 73 30 78 63 41 39 48 ...
##  $ JobInvolvement          : num  4 3 3 3 2 2 3 3 3 3 ...
##  $ JobLevel                : num  1 1 2 1 1 4 1 2 1 2 ...
##  $ JobRole                 : chr  "LT" "R Sci" "Sal R" "R Sci" ...
##  $ JobSatisfaction         : num  4 4 4 2 3 2 1 3 4 4 ...
##  $ MaritalStatus           : chr  "Divorced" "Single" "Married" "Married" ...
##  $ MonthlyIncome           : num  2935 3944 2086 3022 2859 ...
##  $ MonthlyRate             : num  7324 4306 3335 10227 26278 ...
##  $ NumCompaniesWorked      : num  1 5 3 4 1 1 4 1 5 0 ...
##  $ OverTime                : chr  "Yes" "Yes" "No" "No" ...
##  $ PercentSalaryHike       : num  13 11 14 21 18 22 22 14 22 19 ...
##  $ PerformanceRating       : num  3 3 3 4 3 4 4 3 4 3 ...
##  $ RelationshipSatisfaction: num  2 3 3 1 1 4 4 3 1 4 ...
##  $ StockOptionLevel        : num  2 0 1 0 0 1 0 0 1 0 ...
##  $ TotalWorkingYears       : num  1 6 19 8 6 22 7 16 7 12 ...
##  $ TrainingTimesLastYear   : num  2 3 6 1 3 3 3 3 2 2 ...
##  $ WorkLifeBalance         : num  2 3 4 3 3 2 3 3 3 1 ...
##  $ YearsAtCompany          : num  1 3 1 1 6 22 3 16 1 11 ...
##  $ YearsInCurrentRole      : num  0 2 0 0 4 3 2 13 0 9 ...
##  $ YearsSinceLastPromotion : num  0 1 0 0 0 11 0 2 0 4 ...
##  $ YearsWithCurrManager    : num  0 2 0 0 4 11 2 10 0 7 ...
```

Now let's build some prediction models and look at their accuracy.

Results

Now we'll go over the models and the final results.

Note: When I tried to reach a higher accuracy level by using only some columns that had proven to be significant, my accuracy actually decreased. So I've let each type of analysis decide for itself which predictors to include.

Now we'll build two functions that will help us see the accuracy of our prediction models.

This function will round our decimals up or down to 1 or 0.

```
roundBinary = function(x) {
  posneg = sign(x)
  z = abs(x)*10^0
  z = z + 0.5
  z = trunc(z)
  z = z/10^0
  z*posneg
}

# This function will insert our model into a confusion matrix
# to test model accuracy against the test set.
accuracy <- function(model_testing) {
  u <- union(model_testing, testingSet$Attrition)
  t <- table(factor(model_testing, u), factor(testingSet$Attrition, u))
  confusionMatrix(t)
}


# For our first prediction model, we'll start with a very simple approach.
# Let's see what the majority of people did and predict that outcome for
# every employee.
mu_hat <- mean(trainingSet$Attrition)
mu_hat
```

```
## [1] 0.1632653
```

```
percentLeft <- mean(trainingSet$Attrition)
percentLeft
```

```
## [1] 0.1632653
```

```
# 16.32653% of the employees in the training set left the company.

percentStayed <- (1 - percentLeft)
percentStayed
```

```
## [1] 0.8367347
```

83.67347% of the employees in the training set stayed with the company.

So for our first model, we're going to predict the most common outcome (FALSE or 0, which means the employee stayed) as our prediction for everyone in the company to establish as our baseline accuracy level. Then we will hopefully improve accuracy in subsequent models. Let's see how accurate this approach is.

```
length(testingSet$Attrition)
```

```
## [1] 147
```

```
# There are 147 employees in the testing set.

sum(testingSet$Attrition)
```

```
## [1] 21
```

```r
# Only 21 left the company.

length(testingSet$Attrition) - sum(testingSet$Attrition)
```

```
## [1] 126
```

```r
# 126 stayed with the company.

model01 <- rep(0, length(testingSet$Attrition))
model01
```

```
##   [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [36] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [71] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [106] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [141] 0 0 0 0 0 0 0
```

```r
model01 <- roundBinary(model01)
model01
```

```
##   [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [36] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [71] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [106] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [141] 0 0 0 0 0 0 0
```

```r
matrixModel01 <- accuracy(model01)
matrixModel01
```

```
## Confusion Matrix and Statistics
##
##
##       0   1
##   0 126  21
##   1   0   0
##
##                Accuracy : 0.8571
##                  95% CI : (0.79, 0.9093)
##     No Information Rate : 0.8571
##     P-Value [Acc > NIR] : 0.5579
##
##                   Kappa : 0
##
##  Mcnemar's Test P-Value : 1.275e-05
##
##             Sensitivity : 1.0000
##             Specificity : 0.0000
##          Pos Pred Value : 0.8571
##          Neg Pred Value :    NaN
##              Prevalence : 0.8571
##          Detection Rate : 0.8571
##    Detection Prevalence : 1.0000
##       Balanced Accuracy : 0.5000
##
##        'Positive' Class : 0
##
```

```r
# The confusion matrix will show us the model's prediction accuracy.
matrixModel01$overall[1]
```

```
##  Accuracy
## 0.8571429
```

```r
model01_Acc <- matrixModel01$overall[1]

# 85.71429% stayed with the company which means our first model's
# prediction (that everyone stayed) has 85.71429% accuracy.

cat(paste0("The first model has ", model01_Acc*100, "% accuracy."))
```

```
## The first model has 85.7142857142857% accuracy.
```

```r
# Let's put this model into a list and start off our list of attempts:
accuracyTestResultsList <- tibble(method = "Most Common Outcome/Naive Approach Model", Accuracy = model0
accuracyTestResultsList %>% knitr::kable()
```

method	Accuracy
Most Common Outcome/Naive Approach Model	0.8571429

Now we'll carry out the same steps as we did in model 1 except we'll run a RPART (Recursive Partitioning And Regression Trees) analysis.

The RPART analysis works by splitting the data into groups like a big decision tree. It then makes its predictions per entry (or in our case, per employee) based upon where the predictors fall in its decision tree path.

Notice I'm allowing the model to pull from all the predictors available. When I tried to limit the model to only the most significant predictors, it returned a lower accuracy level.

```r
model02 <- rpart(Attrition~.,data=trainingSet)
model02
```

```
## n= 1323
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 1323 180.7347000 0.16326530
##    2) OverTime=No 943  87.8154800 0.10392360
##      4) TotalWorkingYears>=1.5 887  70.3156700 0.08680947 *
##      5) TotalWorkingYears< 1.5 56  13.1250000 0.37500000
##       10) BusinessTravel=Non-Travel,Travel_Rarely 48   9.9166670 0.29166670
##         20) DailyRate>=344.5 39   5.7435900 0.17948720 *
##         21) DailyRate< 344.5 9   1.5555560 0.77777780 *
##       11) BusinessTravel=Travel_Frequently 8   0.8750000 0.87500000 *
##    3) OverTime=Yes 380  81.3578900 0.31052630
##      6) MonthlyIncome>=3751.5 251  38.1992000 0.18725100
##       12) JobRole=HC R,LT,MD,Mgr,R Sci,RD,Sal R 161  14.4099400 0.09937888 *
##       13) JobRole=HR,Sal Ex 90  20.3222200 0.34444440
##         26) DistanceFromHome< 11 59   8.9491530 0.18644070 *
##         27) DistanceFromHome>=11 31   7.0967740 0.64516130 *
##      7) MonthlyIncome< 3751.5 129  31.9224800 0.55038760
##       14) Age>=30.5 69  16.4347800 0.39130430
```

```
##         28) EnvironmentSatisfaction>=1.5 59  12.8813600 0.32203390
##           56) DailyRate>=1133.5 22   1.8181820 0.09090909 *
##           57) DailyRate< 1133.5 37   9.1891890 0.45945950 *
##         29) EnvironmentSatisfaction< 1.5 10   1.6000000 0.80000000 *
##       15) Age< 30.5 60  11.7333300 0.73333330
##         30) YearsWithCurrManager>=0.5 37   8.9189190 0.59459460
##           60) EmployeeNumber>=1118.5 14   2.8571430 0.28571430 *
##           61) EmployeeNumber< 1118.5 23   3.9130430 0.78260870 *
##         31) YearsWithCurrManager< 0.5 23   0.9565217 0.95652170 *
```

```r
model02 <- predict(model02,testingSet,type = "matrix")
model02
```

```
##          1          2          3          4          5          6
## 0.95652174 0.09937888 0.08680947 0.08680947 0.08680947 0.08680947
##          7          8          9         10         11         12
## 0.45945946 0.08680947 0.08680947 0.18644068 0.08680947 0.08680947
##         13         14         15         16         17         18
## 0.08680947 0.08680947 0.08680947 0.08680947 0.08680947 0.08680947
##         19         20         21         22         23         24
## 0.08680947 0.08680947 0.08680947 0.08680947 0.09937888 0.08680947
##         25         26         27         28         29         30
## 0.08680947 0.08680947 0.08680947 0.08680947 0.08680947 0.08680947
##         31         32         33         34         35         36
## 0.08680947 0.09937888 0.09937888 0.08680947 0.45945946 0.08680947
##         37         38         39         40         41         42
## 0.08680947 0.08680947 0.08680947 0.09937888 0.08680947 0.08680947
##         43         44         45         46         47         48
## 0.78260870 0.08680947 0.08680947 0.08680947 0.08680947 0.17948718
##         49         50         51         52         53         54
## 0.08680947 0.08680947 0.08680947 0.45945946 0.08680947 0.08680947
##         55         56         57         58         59         60
## 0.95652174 0.08680947 0.08680947 0.77777778 0.08680947 0.09937888
##         61         62         63         64         65         66
## 0.18644068 0.18644068 0.08680947 0.18644068 0.64516129 0.09937888
##         67         68         69         70         71         72
## 0.08680947 0.17948718 0.08680947 0.08680947 0.08680947 0.08680947
##         73         74         75         76         77         78
## 0.08680947 0.08680947 0.45945946 0.08680947 0.08680947 0.08680947
##         79         80         81         82         83         84
## 0.08680947 0.08680947 0.08680947 0.80000000 0.08680947 0.08680947
##         85         86         87         88         89         90
## 0.18644068 0.08680947 0.28571429 0.08680947 0.08680947 0.08680947
##         91         92         93         94         95         96
## 0.08680947 0.08680947 0.18644068 0.08680947 0.08680947 0.45945946
##         97         98         99        100        101        102
## 0.08680947 0.08680947 0.08680947 0.09937888 0.08680947 0.08680947
##        103        104        105        106        107        108
## 0.08680947 0.09090909 0.09937888 0.08680947 0.08680947 0.45945946
##        109        110        111        112        113        114
## 0.08680947 0.08680947 0.08680947 0.08680947 0.08680947 0.08680947
##        115        116        117        118        119        120
## 0.09937888 0.08680947 0.17948718 0.08680947 0.08680947 0.09937888
##        121        122        123        124        125        126
## 0.09937888 0.08680947 0.08680947 0.18644068 0.45945946 0.08680947
```

```
##        127        128        129        130        131        132
## 0.77777778 0.08680947 0.09937888 0.08680947 0.08680947 0.17948718
##        133        134        135        136        137        138
## 0.08680947 0.08680947 0.08680947 0.08680947 0.95652174 0.08680947
##        139        140        141        142        143        144
## 0.17948718 0.18644068 0.08680947 0.08680947 0.08680947 0.08680947
##        145        146        147
## 0.08680947 0.08680947 0.08680947
```

```r
model02 <- as.vector(model02)
tibble(model02)
```

```
## # A tibble: 147 x 1
##    model02
##      <dbl>
##  1  0.957
##  2  0.0994
##  3  0.0868
##  4  0.0868
##  5  0.0868
##  6  0.0868
##  7  0.459
##  8  0.0868
##  9  0.0868
## 10  0.186
## # ... with 137 more rows
```

```r
model02 <- roundBinary(model02)
model02
```

```
##   [1] 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [36] 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0
##  [71] 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [106] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0
## [141] 0 0 0 0 0 0 0
```

```r
table(testingSet$Attrition,model02)
```

```
##    model02
##       0   1
##   0 122   4
##   1  17   4
```

```r
confusionMatrix(table(testingSet$Attrition,model02))
```

```
## Confusion Matrix and Statistics
##
##    model02
##       0   1
##   0 122   4
##   1  17   4
##
##                Accuracy : 0.8571
##                  95% CI : (0.79, 0.9093)
##     No Information Rate : 0.9456
##     P-Value [Acc > NIR] : 0.999983
##
```

```
##                   Kappa : 0.2139
##
##  Mcnemar's Test P-Value : 0.008829
##
##             Sensitivity : 0.8777
##             Specificity : 0.5000
##          Pos Pred Value : 0.9683
##          Neg Pred Value : 0.1905
##              Prevalence : 0.9456
##          Detection Rate : 0.8299
##    Detection Prevalence : 0.8571
##       Balanced Accuracy : 0.6888
##
##        'Positive' Class : 0
##
```

```
matrixModel02 <- accuracy(model02)
matrixModel02
```

```
## Confusion Matrix and Statistics
##
##
##       1   0
##   1   4   4
##   0  17 122
##
##                Accuracy : 0.8571
##                  95% CI : (0.79, 0.9093)
##     No Information Rate : 0.8571
##     P-Value [Acc > NIR] : 0.557858
##
##                   Kappa : 0.2139
##
##  Mcnemar's Test P-Value : 0.008829
##
##             Sensitivity : 0.19048
##             Specificity : 0.96825
##          Pos Pred Value : 0.50000
##          Neg Pred Value : 0.87770
##              Prevalence : 0.14286
##          Detection Rate : 0.02721
##    Detection Prevalence : 0.05442
##       Balanced Accuracy : 0.57937
##
##        'Positive' Class : 1
##
```

```
matrixModel02$overall[1]
```

```
##  Accuracy
## 0.8571429
```

```
model02_Acc <- matrixModel02$overall[1]
```

Even though the RPART model took a different approach and predicted true for some employees leaving (unlike the first model), it also has an accuracy level of 85.71429%.

```r
cat(paste0("The second model also has ", model02_Acc*100, "% accuracy despite using a different approach
```

```
## The second model also has 85.7142857142857% accuracy despite using a different approach.
```

```r
# Let's put this model into a list and start off our list of attempts:
accuracyTestResultsList <- bind_rows(accuracyTestResultsList,
                                     tibble(method = "RPART Model", Accuracy = model02_Acc))
accuracyTestResultsList %>% knitr::kable()
```

method	Accuracy
Most Common Outcome/Naive Approach Model	0.8571429
RPART Model	0.8571429

Now we'll carry out the same steps as we did in model 2 except we'll run a Generalized Linear Model analysis. This will run a logistic regression, analyzing the relationships between our predictors and what we are trying to predict in order to build an accurate model.

```r
model03 <- glm(Attrition~.,data=trainingSet)
model03
```

```
##
## Call:  glm(formula = Attrition ~ ., data = trainingSet)
##
## Coefficients:
##                     (Intercept)                                Age
##                       5.981e-01                         -3.776e-03
##   BusinessTravelTravel_Frequently       BusinessTravelTravel_Rarely
##                       1.610e-01                          7.686e-02
##                       DailyRate  DepartmentResearch & Development
##                      -2.361e-05                          8.739e-02
##                 DepartmentSales                   DistanceFromHome
##                       3.874e-02                          3.910e-03
##                       Education       EducationFieldLife Sciences
##                       5.421e-04                         -6.868e-02
##         EducationFieldMarketing             EducationFieldMedical
##                      -2.289e-02                         -9.643e-02
##             EducationFieldOther    EducationFieldTechnical Degree
##                      -9.139e-02                          2.768e-02
##                  EmployeeNumber            EnvironmentSatisfaction
##                      -1.114e-05                         -4.379e-02
##                      GenderMale                         HourlyRate
##                       3.419e-02                         -4.019e-04
##                  JobInvolvement                           JobLevel
##                      -5.861e-02                         -5.706e-03
##                       JobRoleHR                          JobRoleLT
##                       1.457e-01                          1.350e-01
##                       JobRoleMD                         JobRoleMgr
##                       3.266e-03                          5.222e-02
##                    JobRoleR Sci                          JobRoleRD
##                       3.904e-02                         -9.302e-03
##                   JobRoleSal Ex                        JobRoleSal R
##                       1.264e-01                          2.543e-01
##                 JobSatisfaction               MaritalStatusMarried
##                      -3.427e-02                          1.467e-02
```

```
##              MaritalStatusSingle                       MonthlyIncome
##                        1.151e-01                           2.212e-06
##                      MonthlyRate                  NumCompaniesWorked
##                        5.147e-07                           1.752e-02
##                      OverTimeYes                   PercentSalaryHike
##                        2.141e-01                          -1.246e-03
##                PerformanceRating            RelationshipSatisfaction
##                        2.679e-03                          -2.013e-02
##                 StockOptionLevel                   TotalWorkingYears
##                       -1.552e-02                          -4.716e-03
##            TrainingTimesLastYear                     WorkLifeBalance
##                       -1.376e-02                          -2.966e-02
##                   YearsAtCompany                  YearsInCurrentRole
##                        6.547e-03                          -9.538e-03
##          YearsSinceLastPromotion                YearsWithCurrManager
##                        1.008e-02                          -8.746e-03
##
## Degrees of Freedom: 1322 Total (i.e. Null);  1277 Residual
## Null Deviance:       180.7
## Residual Deviance: 133.3     AIC: 812.5
```

```r
model03 <- predict(model03,testingSet,type = "response")
model03
```

```
##            1            2            3            4            5
##  0.198485119  0.308230447  0.064135841  0.252449091  0.182833979
##            6            7            8            9           10
##  0.164265664  0.371249779  0.027281074  0.203840207  0.277400981
##           11           12           13           14           15
##  0.396051226  0.216642713  0.175334585  0.083762245  0.089659570
##           16           17           18           19           20
## -0.179385915  0.389920106 -0.058995350 -0.312516692 -0.164243286
##           21           22           23           24           25
## -0.095104828  0.050112768 -0.023025577  0.344358533  0.241803184
##           26           27           28           29           30
##  0.010137487  0.029495000  0.128663843  0.120845221  0.138429326
##           31           32           33           34           35
##  0.105065255  0.176625261  0.327422633  0.329980767  0.403648686
##           36           37           38           39           40
##  0.091233279  0.041216749 -0.043369211  0.198720641  0.140666194
##           41           42           43           44           45
##  0.053990890  0.007443332  0.210668894  0.376580894 -0.096157293
##           46           47           48           49           50
##  0.162238747  0.317806324  0.271973918  0.195093311  0.199273493
##           51           52           53           54           55
## -0.171687842  0.321884826  0.163403073  0.022822017  0.355104143
##           56           57           58           59           60
## -0.220487589  0.204749786  0.127935336  0.052806761  0.234394816
##           61           62           63           64           65
##  0.135228975  0.265336410  0.053110553  0.202253452  0.379332943
##           66           67           68           69           70
##  0.122817342  0.035198543  0.207333792  0.334066123 -0.006797459
##           71           72           73           74           75
## -0.010139070  0.050345950  0.124893618 -0.063375800  0.443619009
##           76           77           78           79           80
```

```
## -0.034793693  0.361695452  0.450549657 -0.235973429 -0.144859751
##           81           82           83           84           85
##  0.186636305  0.655794245  0.026978265  0.091157128  0.558965704
##           86           87           88           89           90
##  0.156663368  0.390734254  0.114060805  0.279074249  0.222416966
##           91           92           93           94           95
##  0.146275969  0.129162312  0.037361455  0.572810713 -0.112864598
##           96           97           98           99          100
##  0.188572913  0.101421215  0.079583094 -0.004349394  0.164754806
##          101          102          103          104          105
##  0.122923338  0.172025092  0.286833444  0.256748446  0.094887513
##          106          107          108          109          110
##  0.231996928  0.070933994  0.542159456  0.083118121 -0.117171333
##          111          112          113          114          115
##  0.169592199  0.160833299  0.060719115  0.386133331  0.178168517
##          116          117          118          119          120
## -0.007368554  0.071857183  0.173234114 -0.087781784  0.442066267
##          121          122          123          124          125
##  0.282816279 -0.002450331 -0.221876836  0.406924466  0.229927401
##          126          127          128          129          130
## -0.025383507  0.305739663  0.329576591  0.038443053  0.210326930
##          131          132          133          134          135
##  0.027719366  0.162956364  0.019021851  0.108343831  0.040240033
##          136          137          138          139          140
## -0.095814928  0.551858806  0.075759267  0.319990908  0.333352237
##          141          142          143          144          145
##  0.270179382  0.065044495  0.064252262 -0.025989441 -0.084712660
##          146          147
## -0.289272199 -0.025451798
```

```r
tibble(model03)
```

```
## # A tibble: 147 x 1
##    model03
##      <dbl>
##  1  0.198
##  2  0.308
##  3  0.0641
##  4  0.252
##  5  0.183
##  6  0.164
##  7  0.371
##  8  0.0273
##  9  0.204
## 10  0.277
## # ... with 137 more rows
```

```r
model03 <- as.vector(model03)

model03 <- roundBinary(model03)
model03
```

```
##   [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [36] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [71] 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
## [106] 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
```

```
## [141] 0 0 0 0 0 0 0
```

```r
table(testingSet$Attrition,model03)
```

```
##    model03
##       0   1
##   0 126   0
##   1  16   5
```

```r
confusionMatrix(table(testingSet$Attrition,model03))
```

```
## Confusion Matrix and Statistics
##
##    model03
##       0   1
##   0 126   0
##   1  16   5
##
##                Accuracy : 0.8912
##                  95% CI : (0.8293, 0.9365)
##     No Information Rate : 0.966
##     P-Value [Acc > NIR] : 0.9999879
##
##                   Kappa : 0.3488
##
##  Mcnemar's Test P-Value : 0.0001768
##
##             Sensitivity : 0.8873
##             Specificity : 1.0000
##          Pos Pred Value : 1.0000
##          Neg Pred Value : 0.2381
##              Prevalence : 0.9660
##          Detection Rate : 0.8571
##    Detection Prevalence : 0.8571
##       Balanced Accuracy : 0.9437
##
##        'Positive' Class : 0
##
```

```r
matrixmodel03 <- accuracy(model03)
matrixmodel03
```

```
## Confusion Matrix and Statistics
##
##
##       0   1
##   0 126  16
##   1   0   5
##
##                Accuracy : 0.8912
##                  95% CI : (0.8293, 0.9365)
##     No Information Rate : 0.8571
##     P-Value [Acc > NIR] : 0.1432608
##
##                   Kappa : 0.3488
##
```

```
##  Mcnemar's Test P-Value : 0.0001768
##
##             Sensitivity : 1.0000
##             Specificity : 0.2381
##          Pos Pred Value : 0.8873
##          Neg Pred Value : 1.0000
##              Prevalence : 0.8571
##          Detection Rate : 0.8571
##    Detection Prevalence : 0.9660
##       Balanced Accuracy : 0.6190
##
##        'Positive' Class : 0
##
```

```
matrixmodel03$overall[1]
```

```
##  Accuracy
## 0.8911565
```

```
model03_Acc <- matrixmodel03$overall[1]

# Our Generalized Linear Model reached 89.11565% accuracy, which is
# higher than the previous models.

cat(paste0("The third model has ", model03_Acc*100, "% accuracy."))
```

```
## The third model has 89.1156462585034% accuracy.
```

```
# Let's put this model into a list and start off our list of attempts:
accuracyTestResultsList <- bind_rows(accuracyTestResultsList,
                                     tibble(method = "Generalized Linear Model", Accuracy = model03_Acc)


# Let's see our final results:
accuracyTestResultsList %>% knitr::kable()
```

method	Accuracy
Most Common Outcome/Naive Approach Model	0.8571429
RPART Model	0.8571429
Generalized Linear Model	0.8911565

```
# The Generalized Linear Model has the highest prediction accuracy
# with 89.11565% accuracy.

cat("The Generalized Linear Model has the highest prediction accuracy of all the models,
    with 89.11565% accuracy.")
```

```
## The Generalized Linear Model has the highest prediction accuracy of all the models,
##     with 89.11565% accuracy.
```

Conclusion

In this section I'll give a brief summary of the report, its limitations and future work.

I split the data into a training set (90% of data) to train the prediction models and a testing set (10% of data) to test the accuracy of the prediction model.

When I tried to reach a higher accuracy level by using only some columns that had proven to be significant in early tests, my accuracy actually decreased. So I let each type of analysis decide for itself which predictors to include from the entire list.

After running three prediction models, the highest accuracy obtained was 0.8911565 or 89.11565%. Surpassing my goal of 88% prediction accuracy.

The most effective prediction model was "Generalized Linear Model".

I feel as though my report has some limitations. I could have taken more modeling approaches to potentially reach a higher prediction accuracy.

I would like to improve this analysis in the future by finding some prediction model approaches that will give me a prediction accuracy of greater than 93%.

Thank you for reading my report. I hope you enjoyed it.

- Avery Clark