Employee Atrrition

Logistic Regression Classification

Alexander Turner

Required Packages

```r
library(tidyverse)
library(rlang)
library(GGally)
library(ggmosaic)
library(car)
library(glmulti)
```

Introduction & Aims

The data being used for this analysis is a fictional dataset created by **IBM** data scientists. It has been created to facilitate the development of statistical and machine learning classification models and is one of many published by IBM for this purpose. This particular dataset contains a binary dependent variable and various independent variables covering a range of data types (numeric, nominal and ordinal).

As the analysis title suggests, this dataset is concerned with employee attrition and consequently the binary dependent variable is whether or not a particular employee left the company. In addition to this there a number of independent variables (listed below) related to each employee which may have an impact on whether they decided to leave or not.

The aim of this analysis is to develop an understanding of what impacts employee attrition and if it can be accurately predicted. This initial phase will consist of data pre-processing and exploratory data visualisations. The latter phase will involve fitting a logistic regression model and performing appropriate diagnostics. The logistic regression will provide a probability of an employee leaving based on a given set of characteristics.

Data Source & Overview

The data can be found **here**. The data has also been posted on **Kaggle** as a reviewed dataset. A number of variables in the dataset have been dropped due to ambiguity of what they are actually are / mean.

Data source:

- Title: SAMPLE DATA Employee Attrition and Performance
- URL: https://www.ibm.com/communities/analytics/watson-analytics-blog/hr-employee-attrition/

Variables:

Dependent variable:

- **Attrition:** whether an employee left the company or not - nominal binary

Independent variables:

- **Age:** employee age - numeric
- **Business Travel:** amount of travel employee does as part of job - ordinal
- **Education Level:** employee education level - ordinal
- **Gender:** employee gender - nominal binary
- **Job Role:** employee job role - nominal
- **Job Level:** seniority of employee - numeric
- **Job Satisfaction:** employee job satisfaction - ordinal
- **Overtime:** whether or not the employee works overtime - nominal binary
- **Performance Rating:** employee performance rating - ordinal
- **Work-Life Balance:** employee work-life balance - ordinal
- **Years in Current Role:** employee years in current role - numeric
- **Years Since Last Promotion:** employee years since last promotion - numeric

Information regarding factor levels and level frequencies are shown in the data pre-processing to follow.

Data Pre-Processing

```r
# import data
attrition <- read_csv("./employee_attrition.csv",
                      col_types = c(Age = "i",
                                    JobLevel = "i",
                                    YearsInCurrentRole = "i",
                                    YearsSinceLastPromotion = "i"))

# data
attrition %>% glimpse()
```

```
## Observations: 1,470
## Variables: 13
## $ Attrition               <chr> "Yes", "No", "Yes", "No", "No", "No", ...
## $ Age                     <int> 41, 49, 37, 33, 27, 32, 59, 30, 38, 36...
## $ BusinessTravel          <chr> "Travel_Rarely", "Travel_Frequently", ...
## $ Education               <dbl> 2, 1, 2, 4, 1, 2, 3, 1, 3, 3, 3, 2, 1,...
## $ Gender                  <chr> "Female", "Male", "Male", "Female", "M...
## $ JobRole                 <chr> "Sales Executive", "Research Scientist...
## $ JobLevel                <int> 2, 2, 1, 1, 1, 1, 1, 1, 3, 2, 1, 2, 1,...
## $ JobSatisfaction         <dbl> 4, 2, 3, 3, 2, 4, 1, 3, 3, 3, 2, 3, 3,...
## $ OverTime                <chr> "Yes", "No", "Yes", "Yes", "No", "No",...
## $ PerformanceRating       <dbl> 3, 4, 3, 3, 3, 3, 4, 4, 4, 3, 3, 3, 3,...
## $ WorkLifeBalance         <dbl> 1, 3, 3, 3, 3, 2, 2, 3, 3, 2, 3, 3, 2,...
## $ YearsInCurrentRole      <int> 4, 7, 0, 7, 2, 7, 0, 0, 7, 7, 4, 5, 2,...
## $ YearsSinceLastPromotion <int> 0, 1, 0, 3, 2, 3, 0, 0, 1, 7, 0, 0, 4,...
```

```r
# convert column names to lower case
names(attrition) <- names(attrition) %>%
  tolower()

# check for missing values
attrition %>% map_dbl(~ is.na(.) %>% sum())
```

```
##               attrition                     age          businesstravel
##                       0                       0                       0
##               education                  gender                 jobrole
##                       0                       0                       0
##                joblevel         jobsatisfaction                overtime
##                       0                       0                       0
##       performancerating         worklifebalance      yearsincurrentrole
##                       0                       0                       0
## yearssincelastpromotion
##                       0
```

```r
# convert character columns to factors
attrition <- attrition %>%
  mutate_if(is.character, factor)

# re-label business travel factor & order
attrition$businesstravel <- attrition$businesstravel %>%
  str_replace("[-_]", " ") %>%
  factor(levels = c("Non Travel", "Travel Rarely", "Travel Frequently"),
         ordered = T)
```

```r
# convert appropriate integer columns to factors
attrition$education <- attrition$education %>%
  factor(levels = 1:5,
         labels = c("Below College", "College", "Bachelor", "Master", "Doctor"),
         ordered = T)

attrition$jobsatisfaction <- attrition$jobsatisfaction %>%
  factor(levels = 1:4,
         labels = c("Low", "Medium", "High", "Very High"),
         ordered = T)

attrition$performancerating <- attrition$performancerating %>%
  factor(levels = 1:4,
         labels = c("Low", "Good", "Excellent", "Outstanding"),
         ordered = T) %>%
  fct_drop() # drop "low" & "good" levels that are empty

attrition$worklifebalance <- attrition$worklifebalance %>%
  factor(levels = 1:4,
         labels = c("Bad", "Good", "Better", "Best"),
         ordered = T)
```

Data Summaries

```r
# numeric variables summary
attrition %>%
  select_if(is.integer) %>%
  map(summary)
```

```
## $age
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   18.00   30.00   36.00   36.92   43.00   60.00 
## 
## $joblevel
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   1.000   2.000   2.064   3.000   5.000 
## 
## $yearsincurrentrole
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   0.000   2.000   3.000   4.229   7.000  18.000 
## 
## $yearssincelastpromotion
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   0.000   0.000   1.000   2.188   3.000  15.000
```

```r
# factor variables summary (including dependent variable)
attrition %>%
  select_if(is.factor) %>%
  map(table) %>%
  map(~ as_tibble(., .name_repair = make.names))
```

```
## $attrition
## # A tibble: 2 x 2
##   X         n
##   <chr> <int>
## 1 No     1233
## 2 Yes     237
## 
## $businesstravel
## # A tibble: 3 x 2
##   X                     n
##   <chr>             <int>
## 1 Non Travel          150
## 2 Travel Rarely      1043
## 3 Travel Frequently   277
## 
## $education
## # A tibble: 5 x 2
##   X                 n
##   <chr>         <int>
## 1 Below College   170
## 2 College         282
## 3 Bachelor        572
## 4 Master          398
## 5 Doctor           48
## 
```

```
## $gender
## # A tibble: 2 x 2
##   X          n
##   <chr>  <int>
## 1 Female   588
## 2 Male     882
##
## $jobrole
## # A tibble: 9 x 2
##   X                             n
##   <chr>                     <int>
## 1 Healthcare Representative   131
## 2 Human Resources              52
## 3 Laboratory Technician       259
## 4 Manager                     102
## 5 Manufacturing Director      145
## 6 Research Director            80
## 7 Research Scientist          292
## 8 Sales Executive             326
## 9 Sales Representative         83
##
## $jobsatisfaction
## # A tibble: 4 x 2
##   X             n
##   <chr>     <int>
## 1 Low         289
## 2 Medium      280
## 3 High        442
## 4 Very High   459
##
## $overtime
## # A tibble: 2 x 2
##   X         n
##   <chr> <int>
## 1 No     1054
## 2 Yes     416
##
## $performancerating
## # A tibble: 2 x 2
##   X               n
##   <chr>       <int>
## 1 Excellent    1244
## 2 Outstanding   226
##
## $worklifebalance
## # A tibble: 4 x 2
##   X          n
##   <chr>  <int>
## 1 Bad       80
## 2 Good     344
## 3 Better   893
## 4 Best     153
```

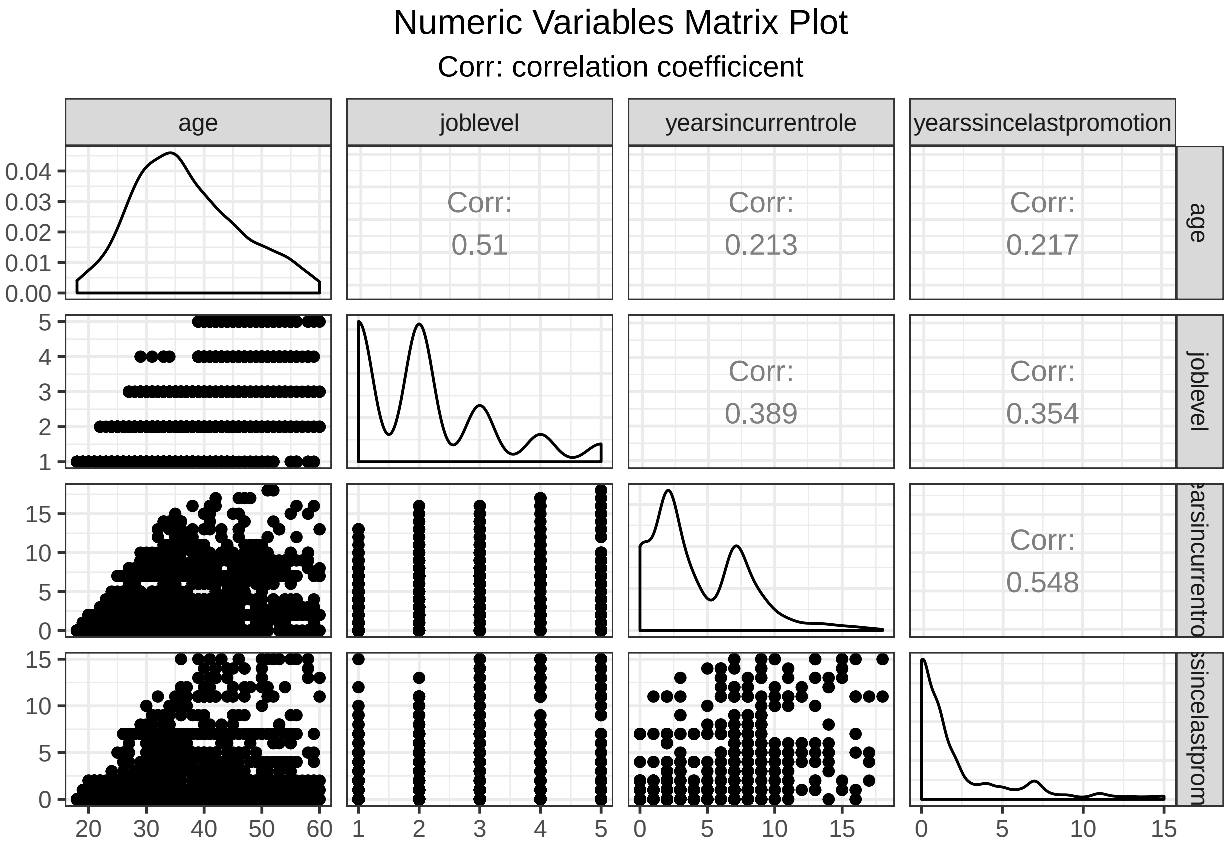
Data Visualisations - Functions & Data

The following code is to define some data and user-defined functions for plotting.

```r
means <- attrition %>%
  group_by(attrition) %>%
  summarise_if(is.integer, mean) %>%
  gather(variable, mean, -1)

hist_fun <- function(df, variable) {
  quo_variable <- enquo(variable)

  df %>%
    ggplot(aes(!!quo_variable, fill = attrition)) +
    geom_histogram(binwidth = 1) +
    facet_grid(attrition ~ ., scales = "free_y") +
    geom_vline(data = filter(means, variable == quo_text(quo_variable)),
               aes(xintercept = mean, linetype = ""),
               size = 1) +
    scale_fill_brewer(palette = "Set2") +
    scale_linetype_manual("Mean", values = "solid") +
    theme(plot.title = element_text(hjust = 0.5)) +
    guides(fill = F) +
    labs(y = "Count")
}

mosaic_fun <- function(df, variable) {
  quo_variable <- enquo(variable)
  sym_variable <- ensym(variable)

  df %>%
    group_by(attrition, !!quo_variable) %>%
    summarise(count = n()) %>%
    ggplot() +
    geom_mosaic(aes(x = product(!!sym_variable), weight = count, fill = attrition)) +
    scale_fill_brewer(palette = "Set2") +
    theme(plot.title = element_text(hjust = 0.5)) +
    labs(y = "Proportion",
         fill = "Attrition")
}

xtab_fun <- function(df, variable) {
  quo_variable <- enquo(variable)

  df %>%
    select(!!quo_variable, attrition) %>%
    table()
}
```

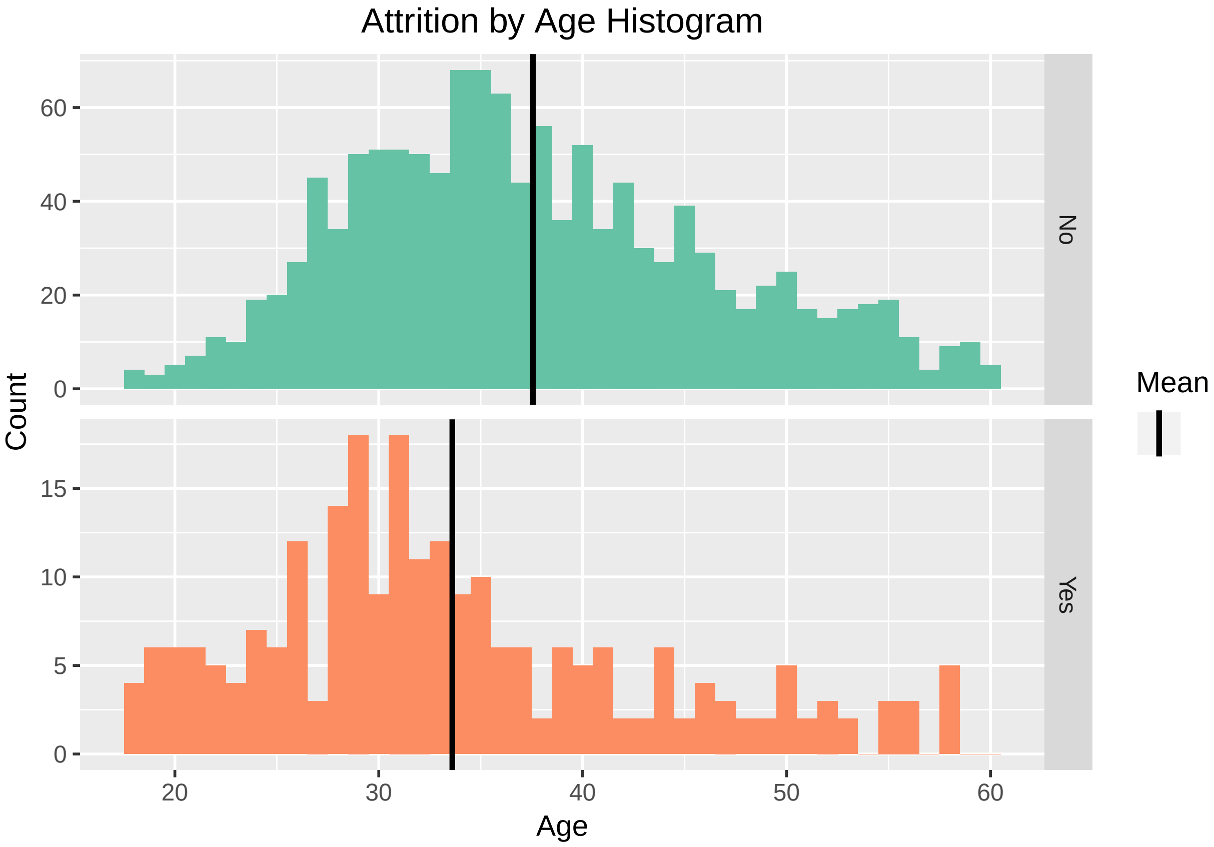
Data Visualisations - Plots

```
attrition %>%
  select_if(is.numeric) %>%
  ggpairs() +
  labs(title = "Numeric Variables Matrix Plot",
       subtitle = "Corr: correlation coefficicent") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```



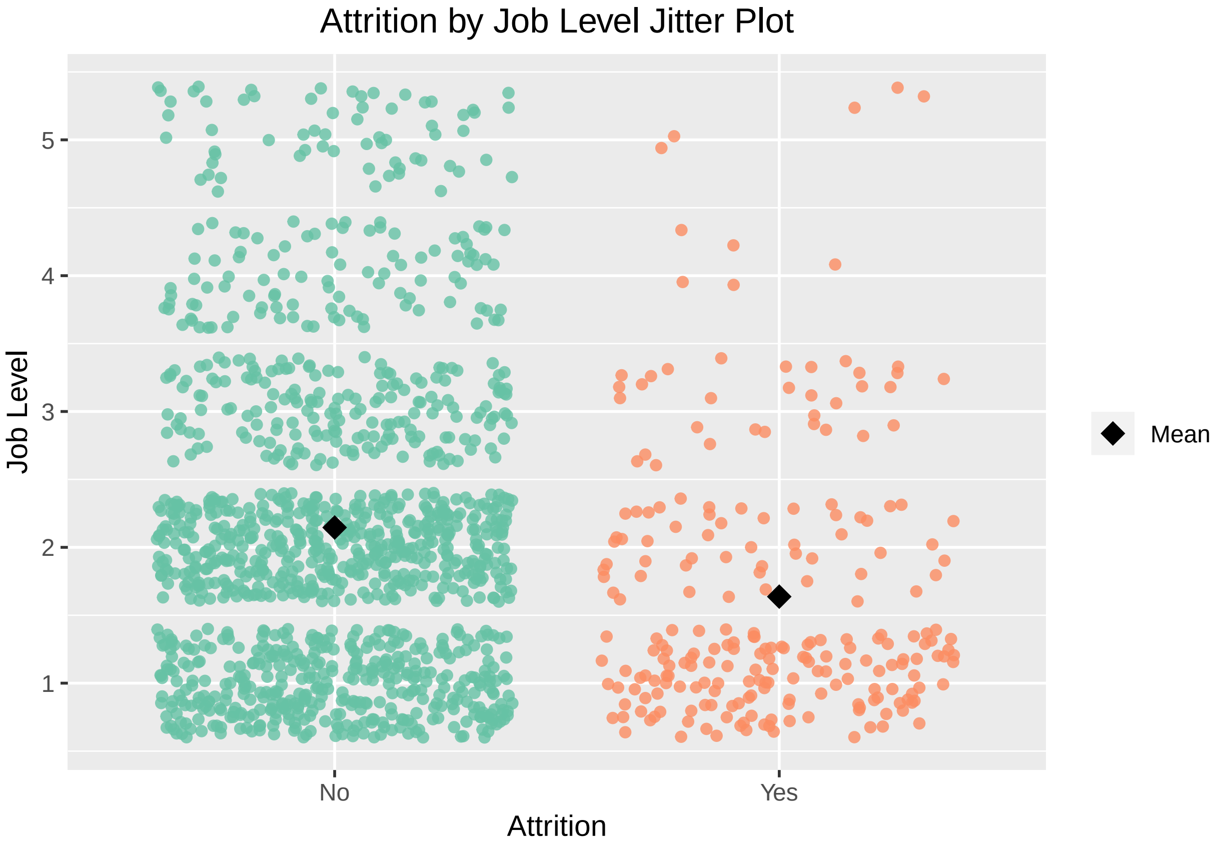
There are no particularly strong correlations among the numeric independent variables which is ideal in a regression model as multicollinearity among predictors can be problematic. Years in current role and years since last promotion have a moderately strong postie correlation and so does age and job level. The job level variable also has a slightly weaker postie correlation with both years in current role and years since last promotion. All of these relationships intuitively make sense.

```r
hist_fun(attrition, age) +
  xlab("Age") +
  ggtitle("Attrition by Age Histogram")
```



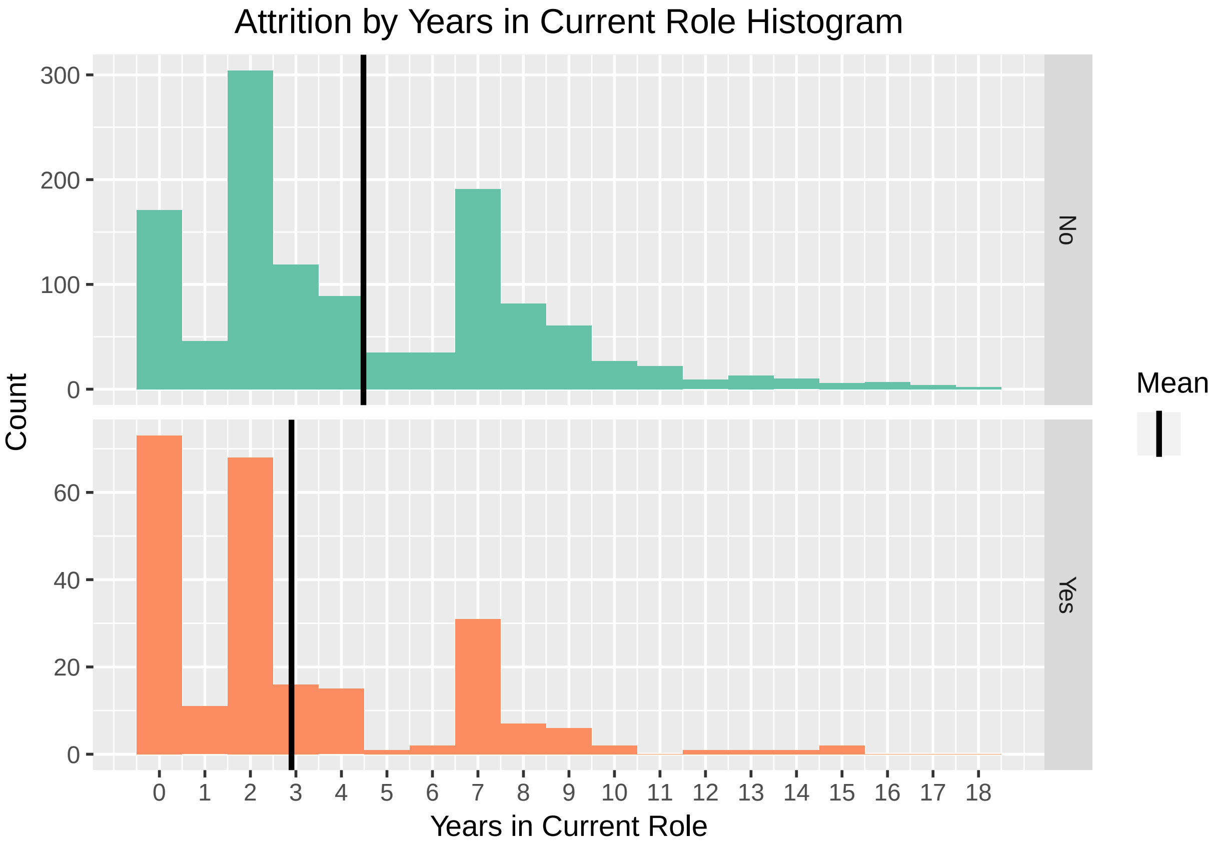
In general, it is younger employees departing the company with an average age of 33.6 for employees that left and 37.6 for those that have stayed. The histogram shows a relatively large number of employees aged 25 and below have departed.

```
attrition %>%
  ggplot(aes(attrition, joblevel, colour = attrition)) +
  geom_jitter(size = 1.5, alpha = 0.8) +
  stat_summary(fun.y = "mean",
               aes(shape = "Mean"),
               geom = "point",
               colour = "black",
               size = 4) +
  scale_colour_brewer(palette = "Set2") +
  scale_shape_manual("", values = 18) +
  scale_y_continuous(breaks = seq(0, 5, by = 1)) +
  labs(title = "Attrition by Job Level Jitter Plot",
       x = "Attrition",
       y = "Job Level") +
  theme(plot.title = element_text(hjust = 0.5)) +
  guides(colour = F)
```



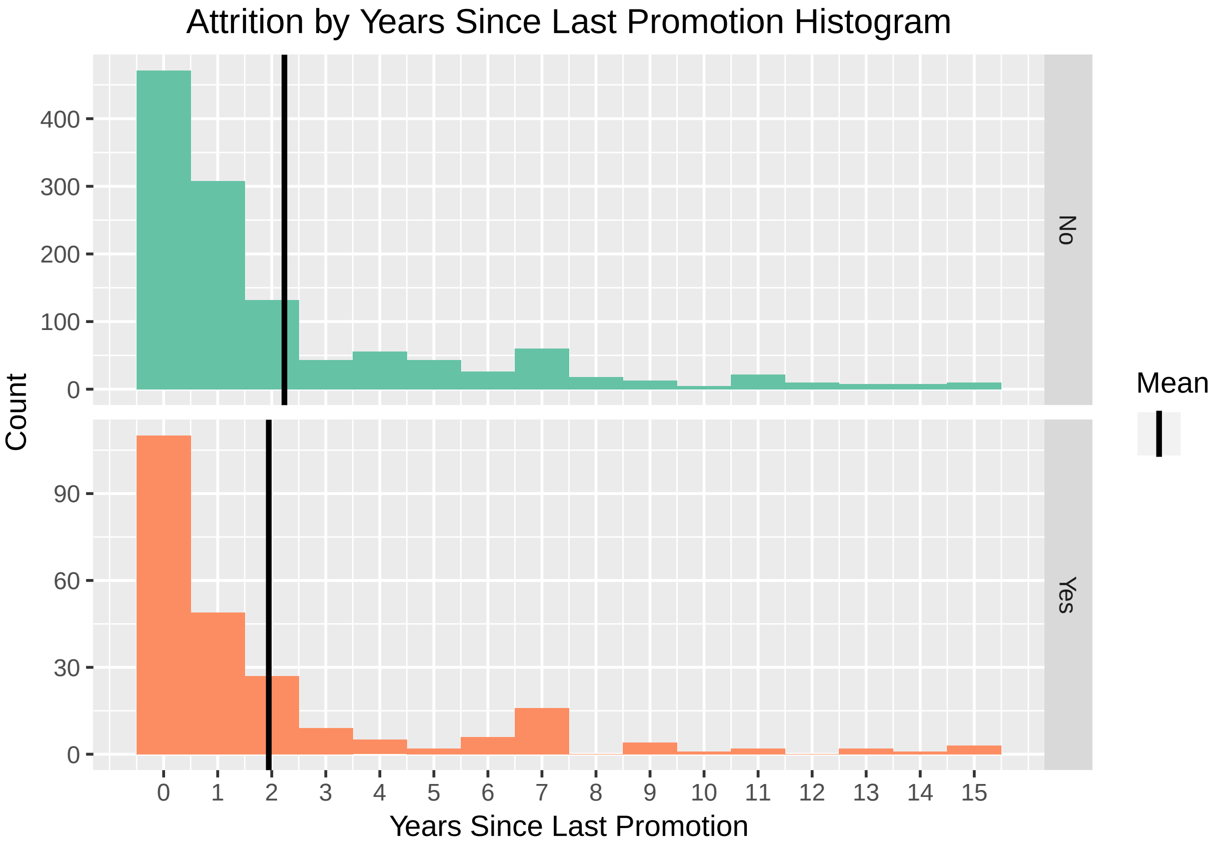
A jitter plot has been used for this numeric variable as there are only five values job level can take and a histogram would be ill-suited. For the job level variable it is evident that lower level employees are tending to leave at higher rates than their higher-level colleagues. For the employees that left the average job level is 1.6, compared to 2.2 for those that chose to stay. This ties in with the previous insight regarding age as the two variables are positively correlated. The jitter plot shows very few employees at job level 4 and above have left.

```
hist_fun(attrition, yearsincurrentrole) +
  xlab("Years in Current Role") +
  ggtitle("Attrition by Years in Current Role Histogram") +
  scale_x_continuous(breaks = seq(min(attrition$yearsincurrentrole),
                                  max(attrition$yearsincurrentrole),
                                  by = 1))
```



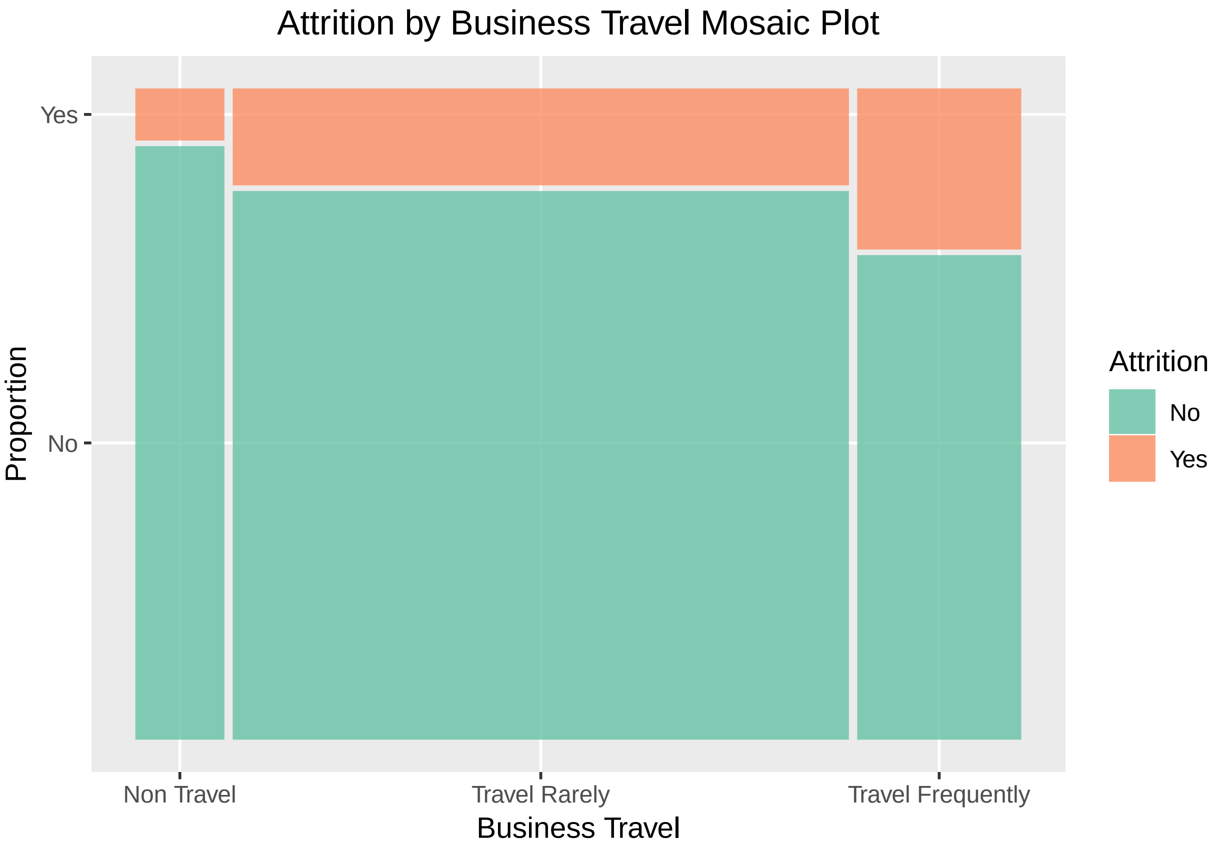
Employees that have left tend to have been in their roles for shorter times. Of those that left the average time in their current role was 2.9 years but for employees that haven't left it is 4.5 years. It seems that lower-level employees that have been in their roles for less time tend to be the ones opting to leave.

```r
hist_fun(attrition, yearssincelastpromotion) +
  xlab("Years Since Last Promotion") +
  ggtitle("Attrition by Years Since Last Promotion Histogram") +
  scale_x_continuous(breaks = seq(min(attrition$yearssincelastpromotion),
                                  max(attrition$yearssincelastpromotion),
                                  by = 1))
```



In terms of years since last promotion, there is little difference in the average (1.9 years for employees that left vs. 2.2 years for those that didn't) and the histogram shows the distribution for the groups is very much the same as well. One might have expected the average to be higher for those that left as not receiving a promotion would typically be seen as a reason for an employee potentially becoming disgruntled and leaving.

```r
mosaic_fun(attrition, businesstravel) +
  xlab("Business Travel") +
  ggtitle("Attrition by Business Travel Mosaic Plot")
```

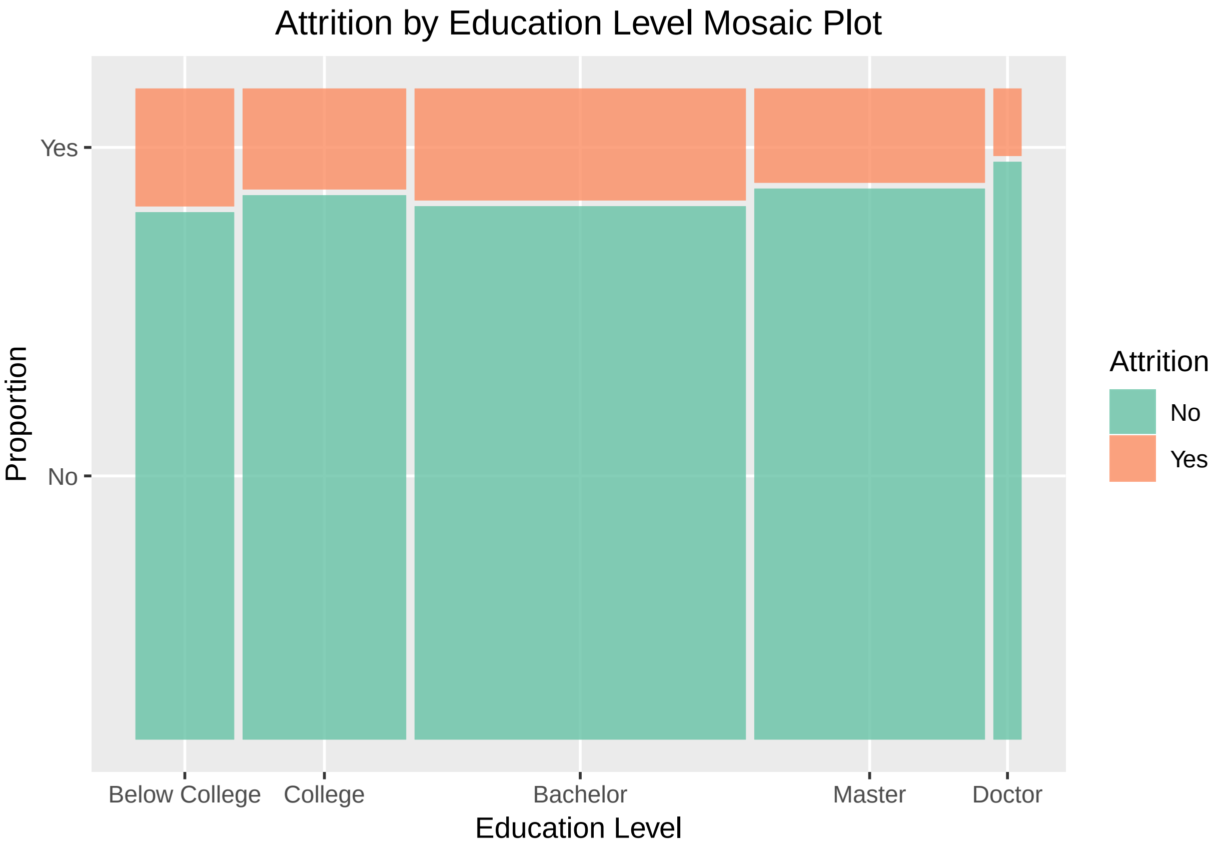


```r
xtab_fun(attrition, businesstravel)
```

```
##                      attrition
## businesstravel        No Yes
##   Non Travel         138  12
##   Travel Rarely      887 156
##   Travel Frequently 208  69
```

Most employees travel rarely, with some travelling frequently and relatively few not at all. It is evident that increased travel appears to be associated with being more likely to leave the company. Approximately 25% of those who travel frequently departed compared to less than 10% for those that weren't required to travel.

```
mosaic_fun(attrition, education) +
  xlab("Education Level") +
  ggtitle("Attrition by Education Level Mosaic Plot")
```

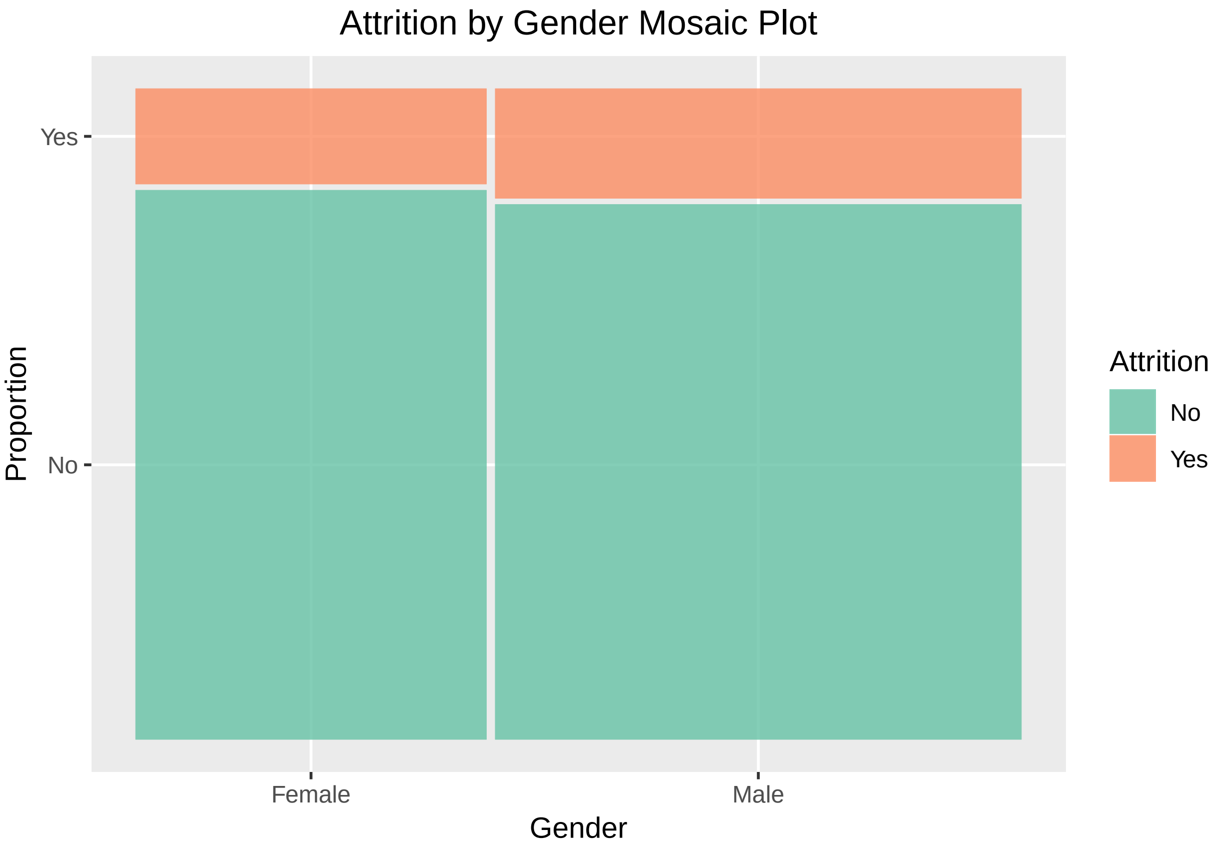


```
xtab_fun(attrition, education)
```

```
##                attrition
## education        No Yes
##   Below College 139  31
##   College       238  44
##   Bachelor      473  99
##   Master        340  58
##   Doctor         43   5
```

Level of education presents fairly even results with the only exception being the low rate of departure for those with a doctorate level degree, but there are very small number of employees at this level. Most employees have a bachelor or master and the number of employees without at least some college education is quite small.

```
mosaic_fun(attrition, gender) +
  xlab("Gender") +
  ggtitle("Attrition by Gender Mosaic Plot")
```

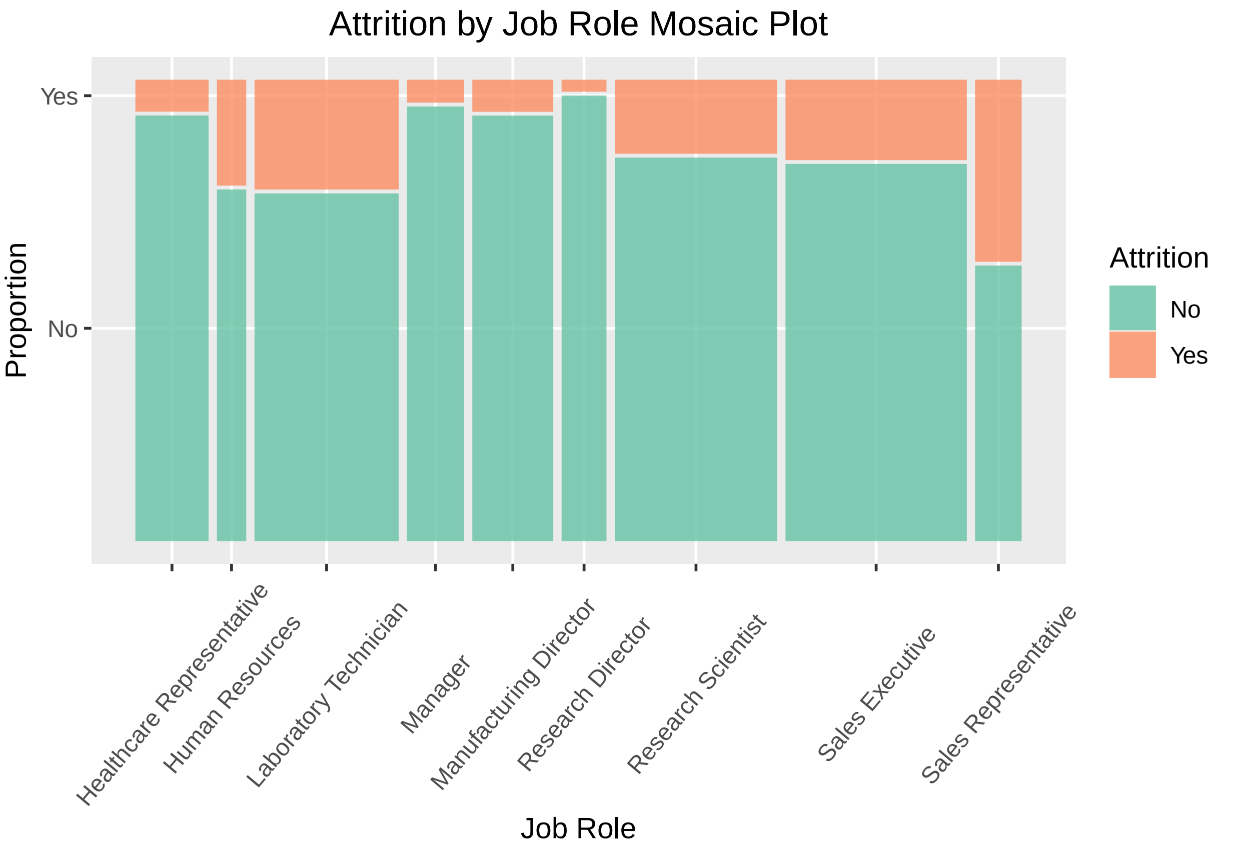


```
xtab_fun(attrition, gender)
```

```
##         attrition
## gender    No Yes
##   Female 501  87
##   Male   732 150
```

The company is well over 50% of male and this group is slightly more likely to leave compared to female employees.

```r
mosaic_fun(attrition, jobrole) +
  xlab("Job Role") +
  ggtitle("Attrition by Job Role Mosaic Plot") +
  theme(axis.text.x = element_text(angle = 50, vjust = 0.5))
```

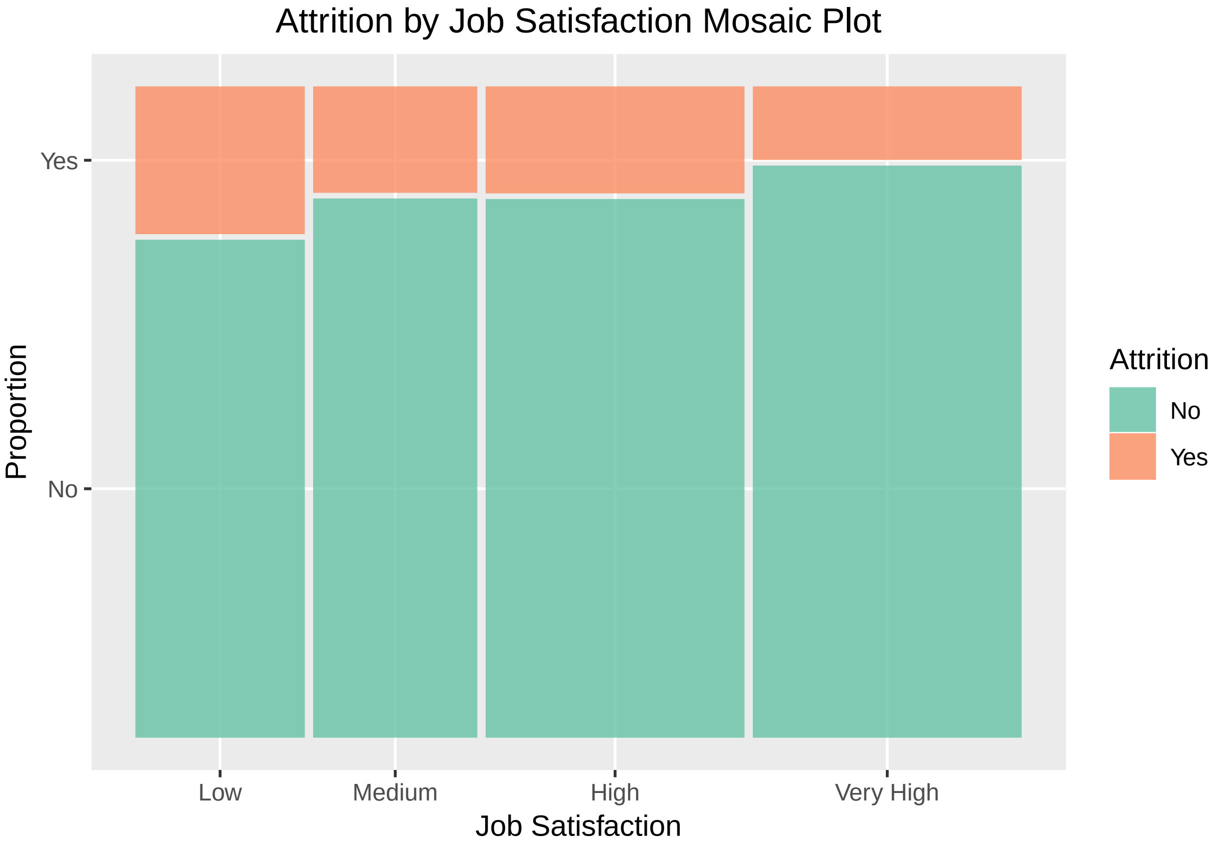


```r
xtab_fun(attrition, jobrole)
```

```
##                            attrition
## jobrole                      No Yes
##   Healthcare Representative 122   9
##   Human Resources            40  12
##   Laboratory Technician     197  62
##   Manager                    97   5
##   Manufacturing Director    135  10
##   Research Director          78   2
##   Research Scientist        245  47
##   Sales Executive           269  57
##   Sales Representative       50  33
```

Across the nine job roles there is a lot of variation in terms of proportion of employees that left. Sales representatives are the most likely to have departed, with ~40% departing. The job roles with relatively few employees (with the exception of human resources) are the ones where more employees stay.

```
mosaic_fun(attrition, jobsatisfaction) +
  xlab("Job Satisfaction") +
  ggtitle("Attrition by Job Satisfaction Mosaic Plot")
```

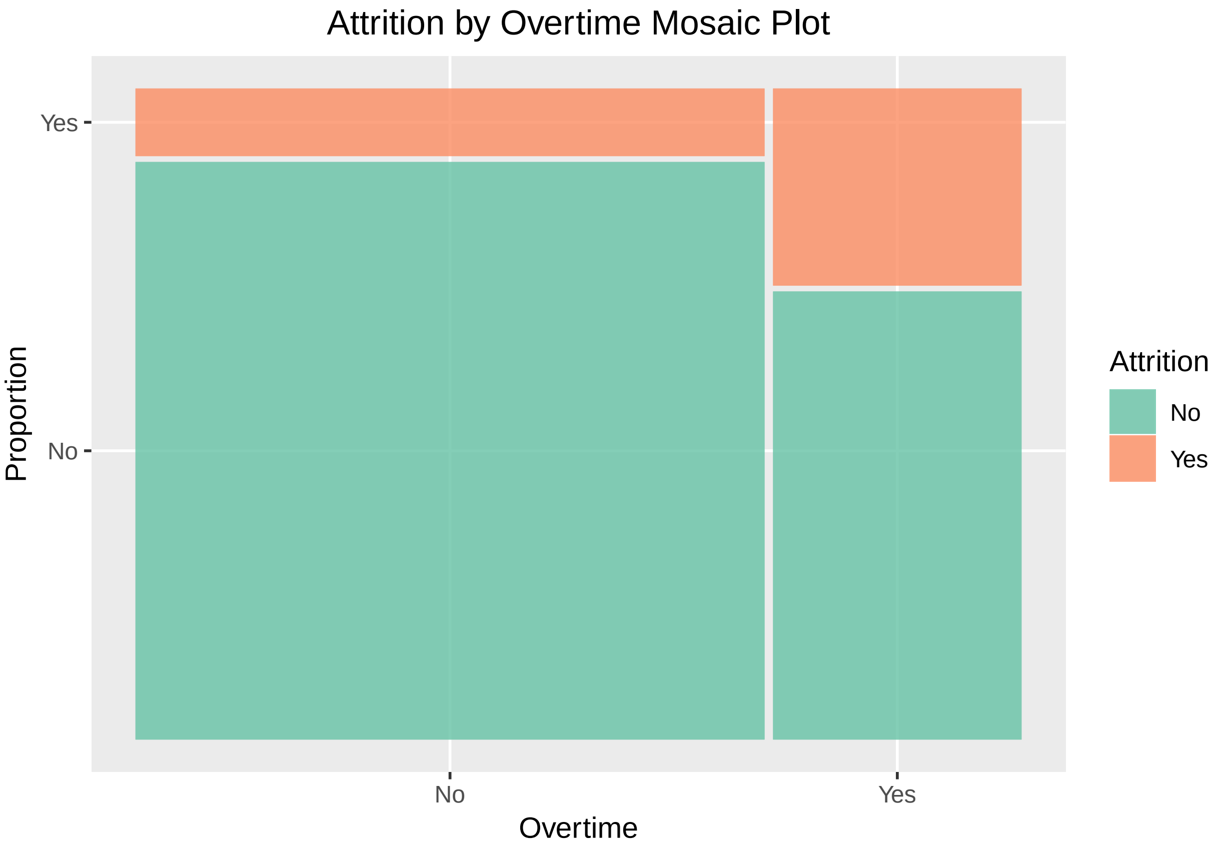


```
xtab_fun(attrition, jobsatisfaction)
```

```
##                attrition
## jobsatisfaction  No Yes
##       Low       223  66
##       Medium    234  46
##       High      369  73
##       Very High 407  52
```

The job satisfaction variable provides an expected insight by revealing employees with low job satisfaction are most likely to leave and those with very high job satisfaction are most likely to stay. Most employees job satisfaction is either High or Very High.

```
mosaic_fun(attrition, overtime) +
  xlab("Overtime") +
  ggtitle("Attrition by Overtime Mosaic Plot")
```

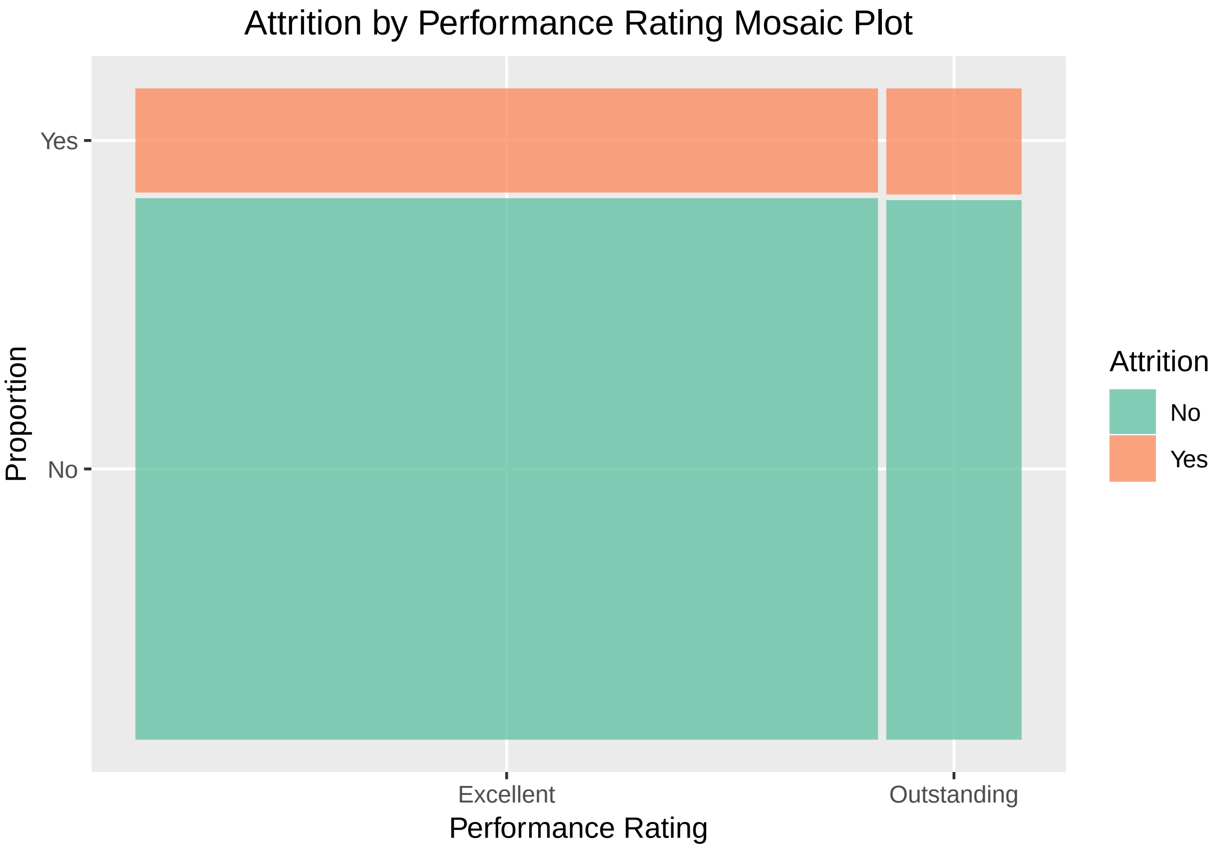


```
xtab_fun(attrition, overtime)
```

```
##          attrition
## overtime   No  Yes
##      No   944  110
##      Yes  289  127
```

Another perhaps expected result with employees doing overtime work being more likely to leave than those who don't. Most employees are not required (or maybe choose not) to do overtime.

```r
mosaic_fun(attrition, performancerating) +
  xlab("Performance Rating") +
  ggtitle("Attrition by Performance Rating Mosaic Plot")
```

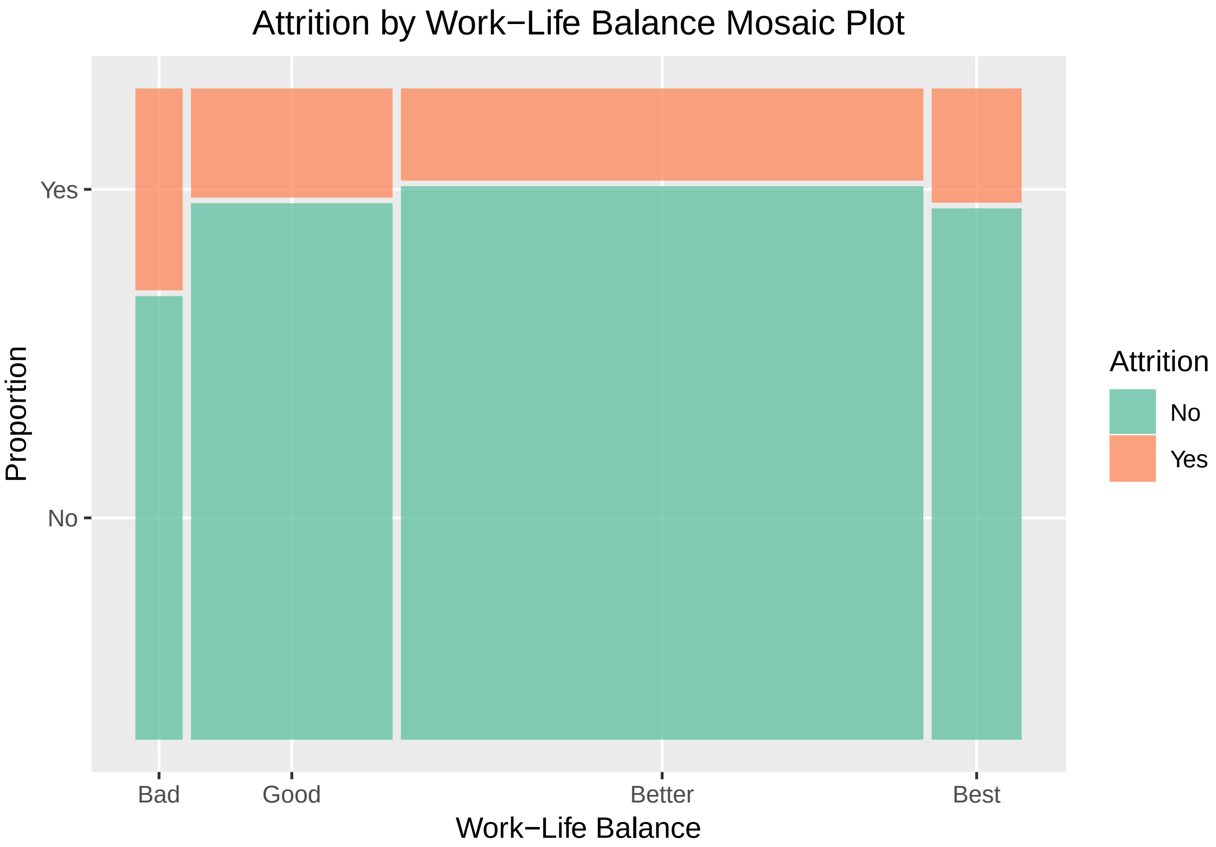


```r
xtab_fun(attrition, performancerating)
```

```
##                  attrition
## performancerating   No  Yes
##       Excellent   1044  200
##       Outstanding  189   37
```

Despite there being four possible performance ratings, all 1,470 employees were rated either Excellent or Outstanding. The overwhelming majority were in the Excellent group and attrition levels were almost identical across groups.

```
mosaic_fun(attrition, worklifebalance) +
  xlab("Work-Life Balance") +
  ggtitle("Attrition by Work-Life Balance Mosaic Plot")
```



```
xtab_fun(attrition, worklifebalance)
```

```
##                 attrition
## worklifebalance  No Yes
##          Bad     55  25
##          Good   286  58
##          Better 766 127
##          Best   126  27
```

Finally, work-life balance shows most employees are described as having Good and Better balance between their work and personal lives. Unsurprisingly, those with Bad work-life balance had the highest proportion of leavers. Across all the others groups, levels were fairly consistent with Best even having a higher attrition rate than Better.

Conclusion - Exploratory Analysis

In conclusion, it appears a number of the independent variables do have an impact on employee on departures. It will be up to the logistic regression model to do the heavy lifting and determine which effects are statistically significant. The model can also help determine which variables impact attrition rates given the presence of other explanatory variables. Interaction effects and higher order terms may also be explored. The model stage will also involve determining how to treat all of the independent variables and whether some should be dropped altogether.

Logistic Regression - Methodology

Logistic regression is a statistical model used to make predictions when there is a binary dependent variable. It does so by assigning a probability (denoted by π) to each of the 2 mutually exclusive outcomes. Like other regression models, logistic regression can handle nominal and numerical independent variables and makes no assumptions about the distributions of these variables.

In the case of a dichotomous dependent variable like this the regression the link function is equal to the natural log of the **odds**, rather than the actual value of the predicted variable itself (the probability of an employee leaving). This is known as the **logit transformation** and the **logistic function** is inverse of the logit. The logistic function is used because its domain is $(-\infty, \infty)$ and its range is naturally bound between 0 and 1, making it ideal for generating probabilities based on a wide range of possible input values. Binary logistic regression like this is a classification problem and the model prediction is whichever binary outcome it assigns the higher probability. For example, if Employee 2 has a probability of 0.54 then the model has predicted they will depart.

Below is the mathematics behind what was previously explained.

Logit function:

$$logit(\pi) = log(\frac{\pi}{1-\pi}) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$$

Logistic function:

$$logistic(x) = \frac{e^x}{1+e^x}$$

Where:

$$x = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$$

Treatment of Independent Variables

All the numeric and nominal independent variables will be fed into the model as they are. For the nominal variables, coding them as factors earlier means R knows they are categorical when creating the model. This will not be the case for the ordinal variables, though. The way they will be input into the model is listed below. Where it seems like it can be more safely assumed that the ordinal variable levels will have a linear effect on the dependent variable, they are coded as numeric. This would imply equal spacing between the variable levels and mean level 4 would be seen as twice level 2, for example. Obviously, this is not entirely true for ordinal variables as they are not ratio data. When it seems like this assumption cannot be met, they are coded as nominal and all information pertaining to their order is lost. Both approaches are not perfect, but this is statistics after all so nothing is ever 100% certain.

Variable:

- **Business Travel:** will be converted to nominal as the difference between Travel Rarely and Travel Frequently is not clear. Non Travel will become the base level in the model
- **Education Level:** will be converted to nominal as it is hard to say how much of difference there is for Master (level 4) vs Bachelor (level 3) and Bachelor vs College (level 2). Below College will become the base level in the model
- **Job Satisfaction:** will be converted to numeric as this variable appears to have equal spacing between factor levels
- **Performance Rating:** will be converted to numeric numeric as this variable appears to have equal spacing between factor levels
- **Work-Life Balance:** will be converted to numeric numeric as this variable appears to have equal spacing between factor levels

Data Transformation

The code below transforms the variables as outlined above.

```r
# transform variables to nominal
attrition_2 <- attrition %>%
  mutate(businesstravel = businesstravel %>% factor(ordered = F),
         education = education %>% factor(ordered = F))

# transform variables to numeric
attrition_2 <- attrition_2 %>%
  mutate(jobsatisfaction = jobsatisfaction %>% as.numeric(),
         performancerating = performancerating %>% as.numeric(),
         worklifebalance = worklifebalance %>% as.numeric())
```

Logistic Regression Model

It was touched on in the methodology section, but to be explicitly clear the base level in this logistic regression model will be No for the dependent attrition variable. This means that any output values above 0.5 will be predicting an employee to leave.

```
# logistic regression model
model_1 <- glm(
  attrition ~ ., # include all variables
  family = binomial(link = logit),
  data = attrition_2
)

# model summary
model_1 %>% summary()
```

```
##
## Call:
## glm(formula = attrition ~ ., family = binomial(link = logit),
##     data = attrition_2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7939  -0.5719  -0.3565  -0.1683   3.5002
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)                      -1.066779   0.848669  -1.257 0.208753
## age                              -0.037851   0.011006  -3.439 0.000584 ***
## businesstravelTravel Rarely       0.665683   0.334775   1.988 0.046762 *
## businesstravelTravel Frequently   1.390391   0.358813   3.875 0.000107 ***
## educationCollege                  0.109535   0.293536   0.373 0.709032
## educationBachelor                 0.238813   0.261186   0.914 0.360538
## educationMaster                   0.190439   0.282861   0.673 0.500781
## educationDoctor                  -0.117879   0.567479  -0.208 0.835445
## genderMale                        0.281318   0.167078   1.684 0.092230 .
## jobroleHuman Resources            1.503540   0.543460   2.767 0.005664 **
## jobroleLaboratory Technician      1.507402   0.452536   3.331 0.000865 ***
## jobroleManager                   -0.013823   0.668890  -0.021 0.983512
## jobroleManufacturing Director     0.192504   0.504933   0.381 0.703021
## jobroleResearch Director         -0.855208   0.867082  -0.986 0.323983
## jobroleResearch Scientist         0.735410   0.458653   1.603 0.108844
## jobroleSales Executive            1.270531   0.403460   3.149 0.001638 **
## jobroleSales Representative       2.075562   0.500280   4.149 3.34e-05 ***
## joblevel                          0.009176   0.179441   0.051 0.959218
## jobsatisfaction                  -0.339890   0.071990  -4.721 2.34e-06 ***
## overtimeYes                       1.614243   0.168026   9.607  < 2e-16 ***
## performancerating                 0.003948   0.223339   0.018 0.985898
## worklifebalance                  -0.285641   0.112917  -2.530 0.011418 *
## yearsincurrentrole               -0.169925   0.033552  -5.065 4.09e-07 ***
## yearssincelastpromotion           0.134198   0.034162   3.928 8.55e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 1298.6  on 1469  degrees of freedom
## Residual deviance: 1023.2  on 1446  degrees of freedom
## AIC: 1071.2
##
## Number of Fisher Scoring iterations: 6
```

The model summary shows the estimated regression coefficients in the Estimate column, along with the standard errors (Std. Error), z values (z value) and corresponding p-values (Pr(>|z|)).

Variable Significance Test

The p values from the `summary` function can be used to test independent variable significance, but in general **likelihood ratio tests** are preferred. The `Anova` function from the `car` package does this.

```r
# likelihood ratio test
model_1 %>% Anova()
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: attrition
##                         LR Chisq Df Pr(>Chisq)
## age                       12.492  1  0.0004087 ***
## businesstravel            21.942  2  1.719e-05 ***
## education                  1.320  4  0.8579389
## gender                     2.875  1  0.0899601 .
## jobrole                   46.103  8  2.272e-07 ***
## joblevel                   0.003  1  0.9592306
## jobsatisfaction           22.657  1  1.937e-06 ***
## overtime                  96.606  1  < 2.2e-16 ***
## performancerating          0.000  1  0.9859010
## worklifebalance            6.357  1  0.0116906 *
## yearsincurrentrole        28.488  1  9.428e-08 ***
## yearssincelastpromotion   15.046  1  0.0001049 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The likelihood ratio tests shows some variables are not significant, given all other variables are in the model.

Deviance & AIC

In the output from the `summary` call are three key elements in assessing the model: null deviance, residual deviance and **AIC**.

Deviance refers to how much one model deviates from another as measured by $-2log(\Lambda)$ where Λ is the likelihood under H_0 divided by the likelihood under H_A (the likelihood ratio).

- **Null deviance:** denotes how much the 'silly' model deviates with one parameter deviates from using a saturated model
- **Residual deviance:** denotes how much the model of interest deviates from the saturated model

Ideally, residual deviance is as low as possible but, just like R^2 in linear regression, this number will only go down if we keep adding more and more parameters. This is where AIC comes in as it helps with model comparison and, ultimately, model selection. AIC is $-2log(\Lambda) + 2r$ where r is the total number of model parameters. It isn't helpful by itself but is when comparing models, with lower AIC indicating a better model. AIC effectively punishes models with a large number of parameters that have little impact individually (but may collectively result in a low residual deviance).

As not all variables are significant in the original model, an algorithm will be used to to sort through the 2^{12} (which is manageable) total number of main effect independent variable combinations possible and find the best in terms of minimising the AIC value. The `glmulti` function from the `glmulti` package will be used to do this.

Model Selection Algorithm

The below code implements the exhaustive search of main effects aimed at minimising the AIC.

```
model_test <- glmulti(
  attrition ~ ., # test all variables
  family = binomial(link = logit),
  data = attrition_2,
  fitfunction = "glm",
  level = 1, # test only main effects
  method = "h", # perform an exhastive search
  crit = "aic" # minimise AIC
)
```

Best model (in terms of main effects) found:

```
model_test %>%
  weightable() %>%
  slice(1) %>%
  select(model) %>%
  pull() %>%
  str_replace_all(" ", "") %>%
  str_split("[+]") %>%
  unlist() %>%
  .[-1]
```

```
## [1] "businesstravel"          "gender"
## [3] "jobrole"                 "overtime"
## [5] "age"                     "jobsatisfaction"
## [7] "worklifebalance"         "yearsincurrentrole"
## [9] "yearssincelastpromotion"
```

AIC value:

```
model_test %>%
  weightable() %>%
  slice(1) %>%
  select(aic)
```

```
##       aic
## 1 1060.51
```

The AIC is lower than the initial model that simply contained all main effects.

Model - Version 2

```r
# optimal model (for AIC reduction)
model_2 <- glm(
  attrition ~ age +
    businesstravel +
    gender +
    jobrole +
    overtime +
    jobsatisfaction +
    worklifebalance +
    yearsincurrentrole +
    yearssincelastpromotion,
  family = binomial(link = logit),
  data = attrition_2
)

# model summary
model_2 %>% summary()
```

```
##
## Call:
## glm(formula = attrition ~ age + businesstravel + gender + jobrole +
##     overtime + jobsatisfaction + worklifebalance + yearsincurrentrole +
##     yearssincelastpromotion, family = binomial(link = logit),
##     data = attrition_2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8531  -0.5714  -0.3557  -0.1702   3.4494
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -0.891218   0.730675  -1.220 0.222572
## age                             -0.036843   0.010195  -3.614 0.000302 ***
## businesstravelTravel Rarely      0.668217   0.333197   2.005 0.044913 *
## businesstravelTravel Frequently  1.397338   0.357272   3.911 9.19e-05 ***
## genderMale                       0.274857   0.166408   1.652 0.098594 .
## jobroleHuman Resources           1.494381   0.521904   2.863 0.004192 **
## jobroleLaboratory Technician     1.500264   0.409505   3.664 0.000249 ***
## jobroleManager                   0.005583   0.608075   0.009 0.992675
## jobroleManufacturing Director    0.195420   0.504425   0.387 0.698451
## jobroleResearch Director        -0.872526   0.826098  -1.056 0.290876
## jobroleResearch Scientist        0.736313   0.412572   1.785 0.074311 .
## jobroleSales Executive           1.276697   0.403261   3.166 0.001546 **
## jobroleSales Representative      2.063739   0.458011   4.506 6.61e-06 ***
## overtimeYes                      1.602308   0.167047   9.592  < 2e-16 ***
## jobsatisfaction                 -0.342335   0.071764  -4.770 1.84e-06 ***
## worklifebalance                 -0.292840   0.112511  -2.603 0.009248 **
## yearsincurrentrole              -0.169904   0.033343  -5.096 3.47e-07 ***
## yearssincelastpromotion          0.135557   0.033654   4.028 5.63e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1298.6  on 1469  degrees of freedom
## Residual deviance: 1024.5  on 1452  degrees of freedom
## AIC: 1060.5
##
## Number of Fisher Scoring iterations: 6
```

Variable Significance Test - Version 2

```
# likelihood ratio test
model_2 %>% Anova()
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: attrition
##                          LR Chisq Df Pr(>Chisq)
## age                        13.736  1  0.0002104 ***
## businesstravel             22.360  2  1.395e-05 ***
## gender                      2.765  1  0.0963223 .
## jobrole                    56.932  8  1.857e-09 ***
## overtime                   96.054  1  < 2.2e-16 ***
## jobsatisfaction            23.139  1  1.507e-06 ***
## worklifebalance             6.733  1  0.0094655 **
## yearsincurrentrole         29.041  1  7.086e-08 ***
## yearssincelastpromotion    15.734  1  7.291e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

All variables are significant except Gender, which just misses the 0.05 cut.

Goodness of Fit

After using `glmulti` to provide the optimal model (in terms of lowest AIC) a goodness of fit test can be used to evaluate the model. A simple but effective measure for this is calculating deviance to degrees of freedom ratio. Ideally this value will not be greater than 1.

```r
# deviance to degrees of freedom ratio
(model_2$deviance / model_2$df.residual) %>% round(2)
```

```
## [1] 0.71
```

The value is well below 1, indicating a good model.

Estimated Model

As is the case with linear regression, all coefficients can be interpreted as the effect of a 1 unit change in the given variable holding all other variables constant. The base levels for all nominal variables become part of the intercept parameter.

$$\begin{gathered}
logit(\hat{\pi}) = -0.8912 - 0.0368 age + 0.6682 businesstravel.rarely + 1.3973 businesstravel.frequently \\
+ 0.2749 gender.male + 1.4944 jobrole.HR + 1.5003 jobrole.labtech + 0.0056 jobrole.manager \\
+ 0.1954 jobrole.manufacturingdirector - 0.8725 jobrole.researchdirector + 0.7363 jobrole.researchscientist \\
+ 1.2767 jobrole.salesexec + 2.0637 jobrole.salesrep + 1.6023 overtime.yes - 0.3423 jobsatisfaction \\
- 0.2928 worklifebalance - 0.1699 yearsincurrentrole + 0.1356 yearssincelastpromotion
\end{gathered}$$

Variable Effects

The regression coefficient estimates show how each of the variables impact attrition probability:

- **Age:** the older an employee is the more probable it is that they leave
- **Business Travel:**
  - Travel Rarely: increases probability of leaving relative to base level Non Travel
  - Travel Frequently: increases probability of leaving relative to base level Non Travel
- **Gender:** Male increases the probability relative to base level Female
- **Job Role:**
  - Human Resources: increases probability of leaving relative to base level Healthcare Representative
  - Laboratory Technician: increases probability of leaving relative to base level Healthcare Representative
  - Manager: decreases probability of leaving relative to base level Healthcare Representative
  - Manufacturing Director: increases probability of leaving relative to base level Healthcare Representative
  - Research Director: decreases probability of leaving relative to base level Healthcare Representative
  - Research Scientist: increases probability of leaving relative to base level Healthcare Representative
  - Sales Executive: increases probability of leaving relative to base level Healthcare Representative
- **Job Satisfaction:** the higher an employee's job satisfaction is (the greater the value of the numeric variable) the less probable it is that they leave
- **Overtime:** Yes increases probability of leaving relative to base level No
- **Work-Life Balance:** the better an employee's work-life balance is (the greater the value of the numeric variable) the less probable it is that they leave
- **Years in Current Role:** the longer an employee has been in their current role the the less probable it is that they leave
- **Years Since Last Promotion:** the longer since an employee's last promotion the more probable it is that they leave

Odds Ratios

One of the main benefits of using logistic regression when working with probabilities is the ability to use **odds ratios** to interpret the effect of changing individual variables.

This comes from the fact that:

$$log(\frac{\pi}{1-\pi}) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$$

Becomes:

$$\frac{\pi}{1-\pi} = e^{\beta_0 + \beta_1 x_1 + ... + \beta_p x_p}$$

Which, of course, is the odds:

$$Odds = e^{\beta_0 + \beta_1 x_1 + ... + \beta_p x_p}$$

So, if the x corresponding to β_1 increases by 1 unit then:

$$OddsRatio = \frac{e^{\beta_0 + \beta_1 (x_1 + 1)}}{e^{\beta_0 + \beta_1 x_1}} = \frac{e^{\beta_0 + \beta_1 x_1 + \beta_1}}{e^{\beta_0 + \beta_1 x_1}} = e^{\beta_1}$$

For every 1 unit (the unit change can be of any magnitude and for any value of x) change in x the odds are e^{β_1} as large. This is true for numeric and nominal variables, except for the nominal variables with more than 1 level. In that case, the above is true when comparing any level to the base (which becomes part of the intercept parameter typically) but it is different when comparing other levels, for example β_2 to β_1.

This is shown below:

$$OddsRatio = \frac{e^{\beta_0 + \beta_2}}{e^{\beta_0 + \beta_1}} = e^{\beta_2 - \beta_1}$$

It is worth noting things are not so simple when considering interactions, quadratic terms etc.

To show how this works in practice some variables will be used for demonstration. Age will be used as the numeric variable and Job Role as the nominal variable.

For every 10 unit increase in Age (employee gets 10 years older) the odds are:

```r
# odds ratio
exp(model_2$coefficients[2] * 10) %>% round(2) %>% unname()
```

```
## [1] 0.69
```

times as large. In other words, the odds are lower. It could also be said that the odds are:

```r
# invert odds ratio
(1 / exp(model_2$coefficients[2] * 10)) %>% round(2) %>% unname()
```

```
## [1] 1.45
```

as large for an employee staying for every 10 years younger they are. Odds ratios can always be inverted when considering the decrease in a variable like this.

The odds of a Laboratory Technician leaving compared to a Healthcare Representative are:

```
# odds ratio
exp(model_2$coefficients[7]) %>% round(2) %>% unname()
```

```
## [1] 4.48
```

times as large. In other words, the odds are higher that Laboratory Technicians leave compared to Healthcare Representatives.

The odds of a Laboratory Technician leaving compared to a Research Scientist are:

```
# odds ratio
exp(model_2$coefficients[7] - model_2$coefficients[11]) %>% round(2) %>% unname()
```

```
## [1] 2.15
```

times as large. In other words, the odds are higher that Laboratory Technicians leave compared to Research Scientists.

Conclusion - Model

In conclusion, it has been demonstrated that with the independent variables in this particular dataset a useful logistic regression model can be obtained for predicting attrition. With a relatively small number of variables, an exhaustive search of the main features was possible resulting in a model that had the lowest AIC possible. Of course, this analysis did not consider interactions between variables or higher order terms. Adding these in could potentially reduce the AIC and result in better overall prediction values but there is also a risk of overfitting. Finally, the goodness of fit test result showed a value well below 1, indicating a good model.