IBM Attrition

Mulisana Gerveshi

09/05/2020

Employees are the backbone of any organization. Its performance is heavily based on the quality of the employees and retaining them. Employee attrition can make an organization lose money, valuable employees and can impact it's productivity too.

This dataset is fictional data created by IBM data scientists. The aim of it is to identify the factors that cause employee attrition and give other suggestions. To achive that, in the first part of this project I will be doing some exploratory data analysis (EDA) of the important variables to understand the impact of each of them into employee attrition. In the next step I will be putting some of the most important variables in a correlation matrix. The matrix will determine if a relationship exists between the variables and how strong it is. In the second part I will work decision tree as a predictive model for Attrition.

Summary: 1) Summary of data 2) Attrition Level 3) Income Analysis 4) Gender Analysis 5) Environment analysis 6) Other factors 7) Correlation matrix 8) Decision tree

```
#Load the libraries that we will be using
library(readr)
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(tidyselect)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v tibble  2.1.3     v stringr 1.4.0
## v purrr   0.3.3     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(skimr)
library(highcharter)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## Highcharts (www.highcharts.com) is a Highsoft software product which is
```

```
## not free for commercial and Governmental use
```

```
library(highr)
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(highcharter)
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

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

```
library(rpart.plot)
```

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

```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(RColorBrewer)
library(ggcorrplot)
library(scales)
```

```
##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor
```

```
library(forcats)
library(cowplot)
```

```
##
## ************************************************************

## Note: As of version 1.0.0, cowplot does not change the

##   default ggplot2 theme anymore. To recover the previous

##   behavior, execute:
##   theme_set(theme_cowplot())

## ************************************************************
```

```
library(ggthemes)
```

```
##
## Attaching package: 'ggthemes'

## The following object is masked from 'package:cowplot':
##
##     theme_map
```

```
library(devtools)
```

```
## Loading required package: usethis
```

```
library(plyr)
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##
## Attaching package: 'plyr'

## The following objects are masked from 'package:plotly':
##
##     arrange, mutate, rename, summarise

## The following object is masked from 'package:purrr':
##
##     compact

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
#Import the dataset
df <- read_csv("~/WA_Fn-UseC_-HR-Employee-Attrition.csv")
```

```
## 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.
```

1) Summary of Data Before starting with the actual analysis, we need to take a look and understand the data.

```
glimpse(df)
```

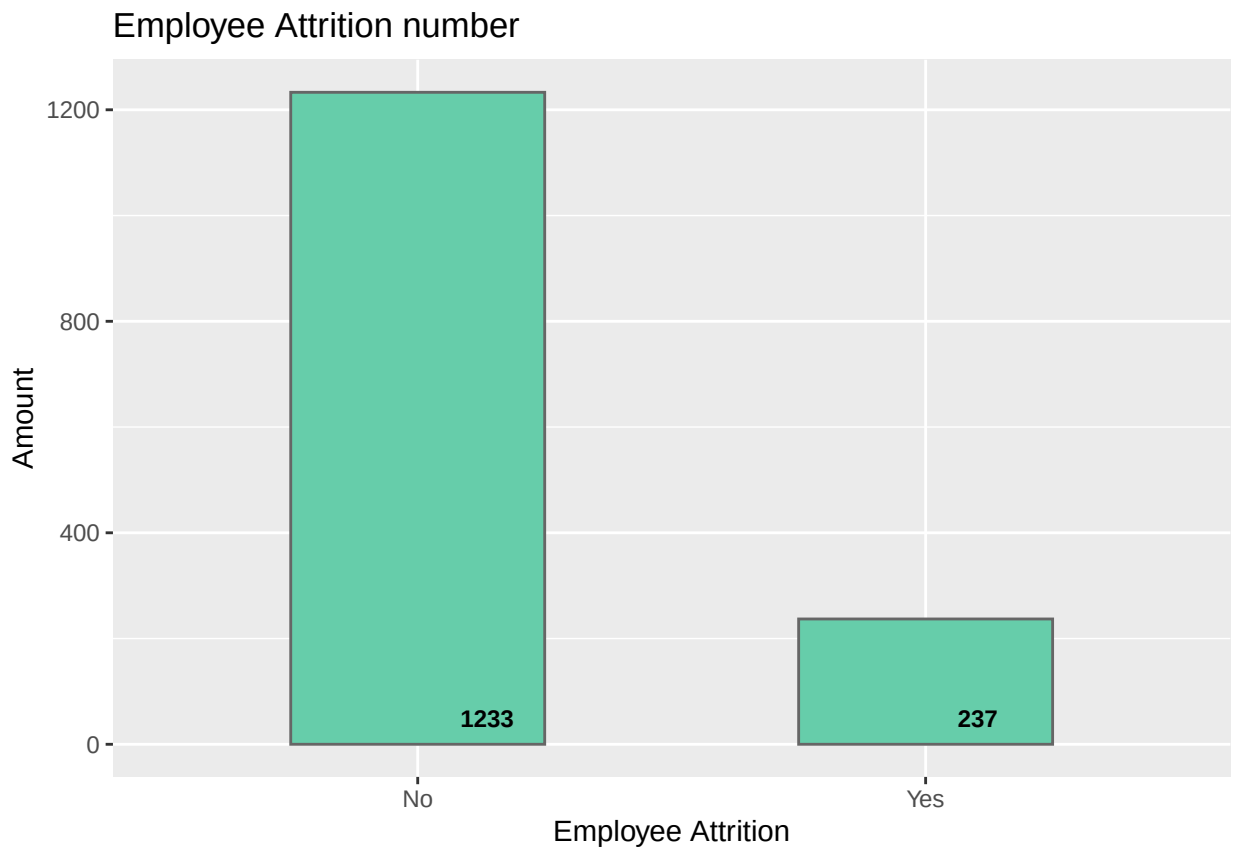
```
## Observations: 1,470
## Variables: 35
## $ Age                      <dbl> 41, 49, 37, 33, 27, 32, 59, 30, 38, 36, 35...
## $ Attrition                <chr> "Yes", "No", "Yes", "No", "No", "No", "No"...
## $ BusinessTravel           <chr> "Travel_Rarely", "Travel_Frequently", "Tra...
```

```
## $ DailyRate                <dbl> 1102, 279, 1373, 1392, 591, 1005, 1324, 13...
## $ Department               <chr> "Sales", "Research & Development", "Resear...
## $ DistanceFromHome         <dbl> 1, 8, 2, 3, 2, 2, 3, 24, 23, 27, 16, 15, 2...
## $ Education                <dbl> 2, 1, 2, 4, 1, 2, 3, 1, 3, 3, 3, 2, 1, 2, ...
## $ EducationField           <chr> "Life Sciences", "Life Sciences", "Other",...
## $ EmployeeCount            <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ EmployeeNumber           <dbl> 1, 2, 4, 5, 7, 8, 10, 11, 12, 13, 14, 15, ...
## $ EnvironmentSatisfaction  <dbl> 2, 3, 4, 4, 1, 4, 3, 4, 4, 3, 1, 4, 1, 2, ...
## $ Gender                   <chr> "Female", "Male", "Male", "Female", "Male"...
## $ HourlyRate               <dbl> 94, 61, 92, 56, 40, 79, 81, 67, 44, 94, 84...
## $ JobInvolvement           <dbl> 3, 2, 2, 3, 3, 3, 4, 3, 2, 3, 4, 2, 3, 3, ...
## $ JobLevel                 <dbl> 2, 2, 1, 1, 1, 1, 1, 1, 3, 2, 1, 2, 1, 1, ...
## $ JobRole                  <chr> "Sales Executive", "Research Scientist", "...
## $ JobSatisfaction          <dbl> 4, 2, 3, 3, 2, 4, 1, 3, 3, 3, 2, 3, 3, 4, ...
## $ MaritalStatus            <chr> "Single", "Married", "Single", "Married", ...
## $ MonthlyIncome            <dbl> 5993, 5130, 2090, 2909, 3468, 3068, 2670, ...
## $ MonthlyRate              <dbl> 19479, 24907, 2396, 23159, 16632, 11864, 9...
## $ NumCompaniesWorked       <dbl> 8, 1, 6, 1, 9, 0, 4, 1, 0, 6, 0, 0, 1, 0, ...
## $ Over18                   <chr> "Y", "Y", "Y", "Y", "Y", "Y", "Y", "Y", "Y...
## $ OverTime                 <chr> "Yes", "No", "Yes", "Yes", "No", "No", "Ye...
## $ PercentSalaryHike        <dbl> 11, 23, 15, 11, 12, 13, 20, 22, 21, 13, 13...
## $ PerformanceRating        <dbl> 3, 4, 3, 3, 3, 3, 4, 4, 4, 3, 3, 3, 3, 3, ...
## $ RelationshipSatisfaction <dbl> 1, 4, 2, 3, 4, 3, 1, 2, 2, 2, 3, 4, 4, 3, ...
## $ StandardHours            <dbl> 80, 80, 80, 80, 80, 80, 80, 80, 80, 80, 80...
## $ StockOptionLevel         <dbl> 0, 1, 0, 0, 1, 0, 3, 1, 0, 2, 1, 0, 1, 1, ...
## $ TotalWorkingYears        <dbl> 8, 10, 7, 8, 6, 8, 12, 1, 10, 17, 6, 10, 5...
## $ TrainingTimesLastYear    <dbl> 0, 3, 3, 3, 3, 2, 3, 2, 2, 3, 5, 3, 1, 2, ...
## $ WorkLifeBalance          <dbl> 1, 3, 3, 3, 3, 2, 2, 3, 3, 2, 3, 3, 2, 3, ...
## $ YearsAtCompany           <dbl> 6, 10, 0, 8, 2, 7, 1, 1, 9, 7, 5, 9, 5, 2,...
## $ YearsInCurrentRole       <dbl> 4, 7, 0, 7, 2, 7, 0, 0, 7, 7, 4, 5, 2, 2, ...
## $ YearsSinceLastPromotion  <dbl> 0, 1, 0, 3, 2, 3, 0, 0, 1, 7, 0, 0, 4, 1, ...
## $ YearsWithCurrManager     <dbl> 5, 7, 0, 0, 2, 6, 0, 0, 8, 7, 3, 8, 3, 2, ...
```

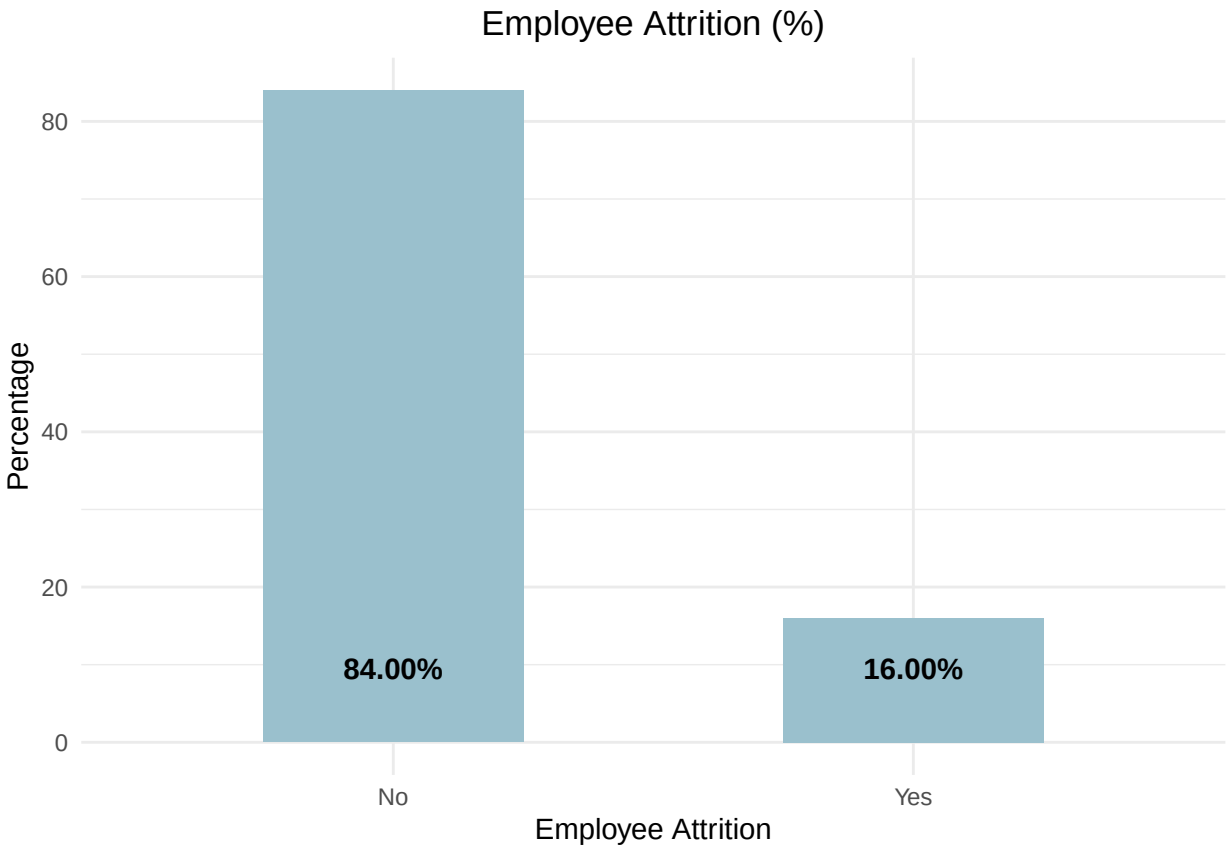
There are 1,470 observations and 35 variables. Most of the variables are integer and some of them are factors. There are no missing values.

2) Attrition Level In order to try and find the reasons for attrition we will be having a look at its level first (number and percentage).

```
attrition_number <- df %>%  dplyr::group_by(Attrition) %>%
   dplyr::summarise(Count=n()) %>%
ggplot(aes(x=Attrition, y=Count)) + geom_bar(stat="identity", fill="mediumaquamarine", color="grey40", w
geom_text(aes(x=Attrition, y=0.01, label= Count),
            hjust=-0.8, vjust=-1, size=3,
            colour="black", fontface="bold",
         angle=360) + labs(title="Employee Attrition number",
                           x="Employee Attrition",y="Amount")
attrition_number
```



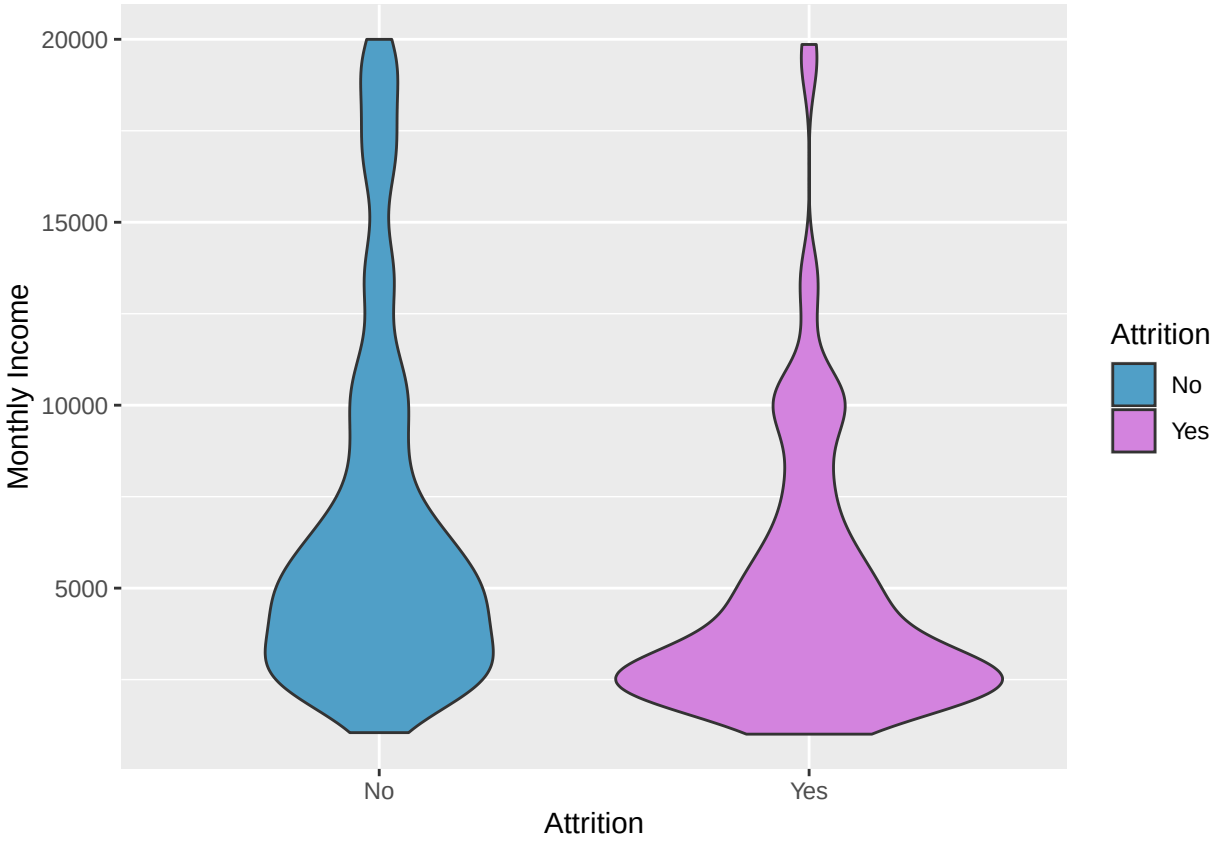

```
attrition_percentage <- df %>% group_by(Attrition) %>%
   dplyr::summarise(Count=n()) %>%
mutate(pct=round(prop.table(Count),2) * 100) %>%
ggplot(aes(x=Attrition, y=pct)) +
  geom_bar(stat="identity", fill = "lightblue3", width = 0.5) +
geom_text(aes(x=Attrition, y=0.01,
              label= sprintf("%.2f%%", pct)),
              hjust=0.5, vjust=-3, size=4,
              colour="black", fontface="bold") +
  theme_minimal() +
  labs(x="Employee Attrition", y="Percentage")  +
labs(title="Employee Attrition (%)") + theme(plot.title=element_text(hjust=0.5))
attrition_percentage
```



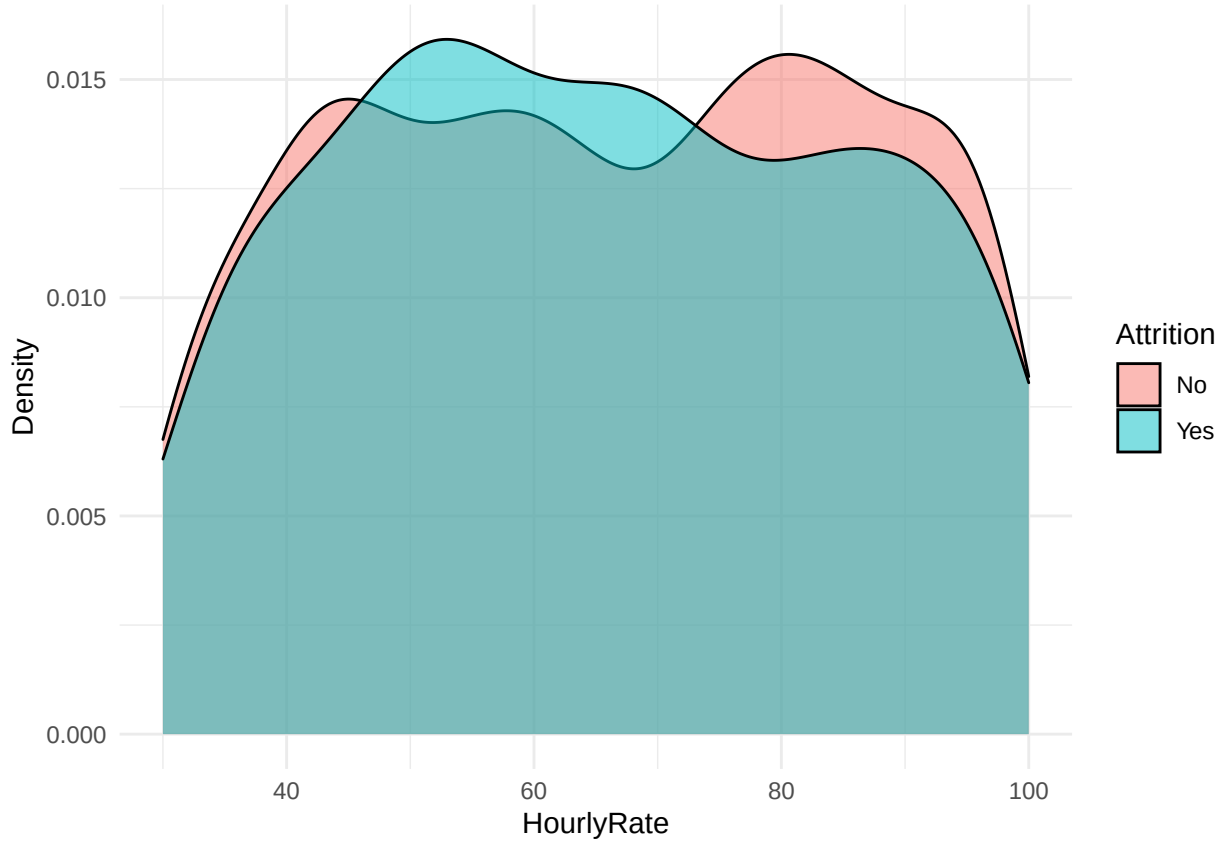
The attrition percentage in the organization is 16%.

3) Income Analysis: on the third part we will try to understand the impact that pay has on attrition.

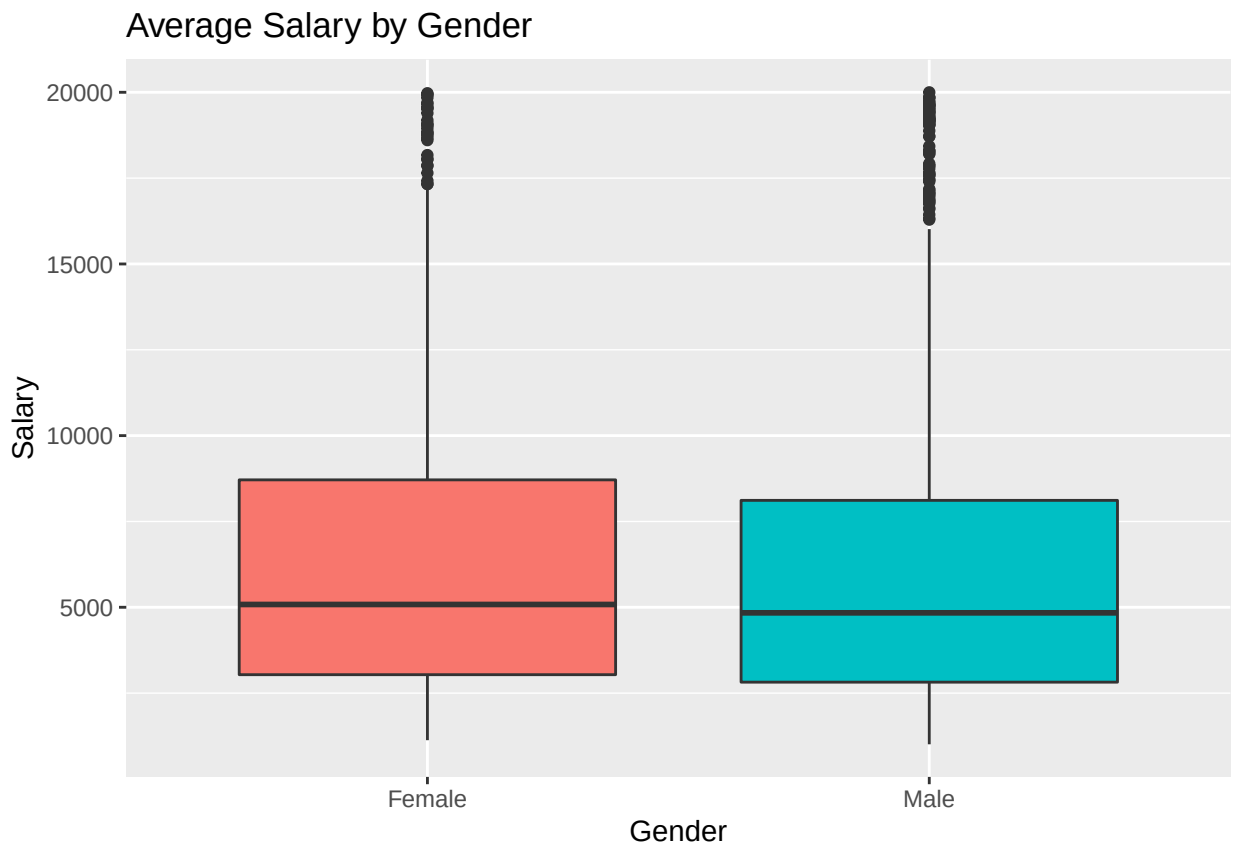
```
att_by_monthlyIncome <- df %>%
ggplot(aes(x = Attrition, y = MonthlyIncome, fill = Attrition)) + geom_violin() +
  scale_fill_manual(values = c("#509FC7", "#D383DE")) +
  ylab("Monthly Income")
att_by_monthlyIncome
```




```
att_by_hourlyrate <- df %>%
  group_by(HourlyRate, Attrition) %>%
  ggplot(aes(x = HourlyRate, fill = Attrition)) +
  geom_density(alpha = 0.5) + theme_minimal() +
  ylab("Density")
att_by_hourlyrate
```



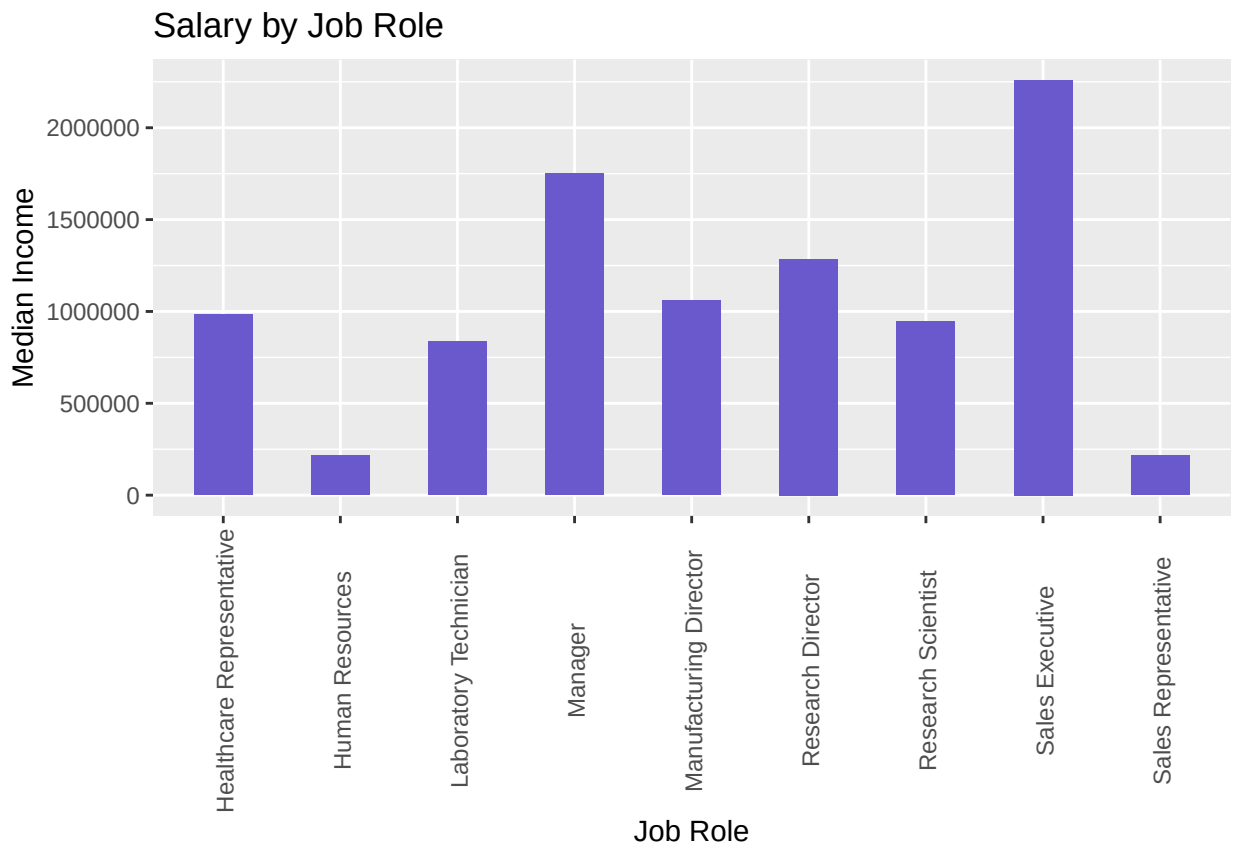

```
avg_salar_gender <- df %>%
  group_by(MonthlyIncome, Gender) %>%
  ggplot(aes(x = Gender, y = MonthlyIncome, fill = Gender)) +
  geom_boxplot() +
  ggtitle("Average Salary by Gender") + ylab("Salary") +
  theme(legend.position = "none")
avg_salar_gender
```




```
att_by_percentsalaryhike <- df %>%
  select(Attrition, PercentSalaryHike, MonthlyIncome) %>%
ggplot(aes(x=PercentSalaryHike, y=MonthlyIncome)) + geom_jitter(aes(col=Attrition), alpha=0.7) +
theme(panel.background = element_rect(fill = "darkgrey"), legend.position="bottom") +
  scale_color_manual(values=c("#a358e8", "#f2fa11")) +
labs(title="The Impact of Income on Attrition") +
  ylab("Monthly Income") + xlab("Percent Salary Hike")
att_by_percentsalaryhike
```


```
salary_jobrole <- df %>%
  ggplot(aes(x=JobRole, y=MonthlyIncome)) +
  geom_bar(stat="identity", width=.5, fill="slateblue3") +
geom_smooth() +
  labs(title="Salary by Job Role",
       x="Job Role",
       y="Median Income") +
  theme(axis.text.x = element_text(angle=90, vjust=0.6))
salary_jobrole
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



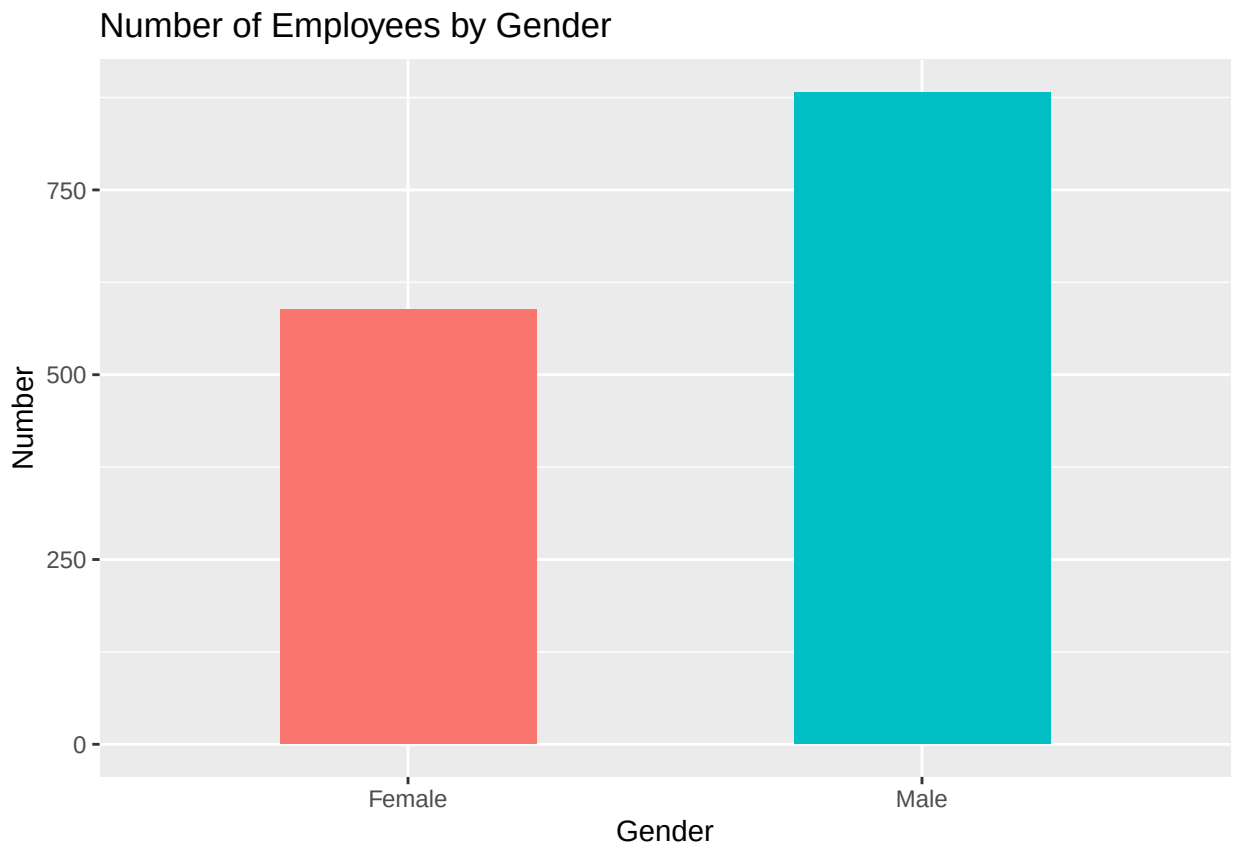

```
perf_by_inc <- df %>%
  select(PerformanceRating, MonthlyIncome, Attrition) %>% group_by(factor(PerformanceRating), Attrition)
ggplot(aes(x=factor(PerformanceRating),
           y=MonthlyIncome, fill=Attrition)) +
  geom_violin()  + facet_wrap(~Attrition) +
scale_fill_manual(values=c("#f6acc8", "#584153")) +
  theme_minimal() +
theme(legend.position="bottom", strip.background = element_blank(), strip.text.x = element_blank(),
     plot.title=element_text(hjust=0.5, color="white"), plot.background=element_rect(fill="gray89"),
                                                        axis.text.x=element_text(colour="white"), axis.
                                                        axis.title=element_text(colour="white"),

     legend.text=element_text(color="white")) +
labs(x="Performance Rating",y="Monthly Income")
perf_by_inc
```

Summary for Income Analysis:

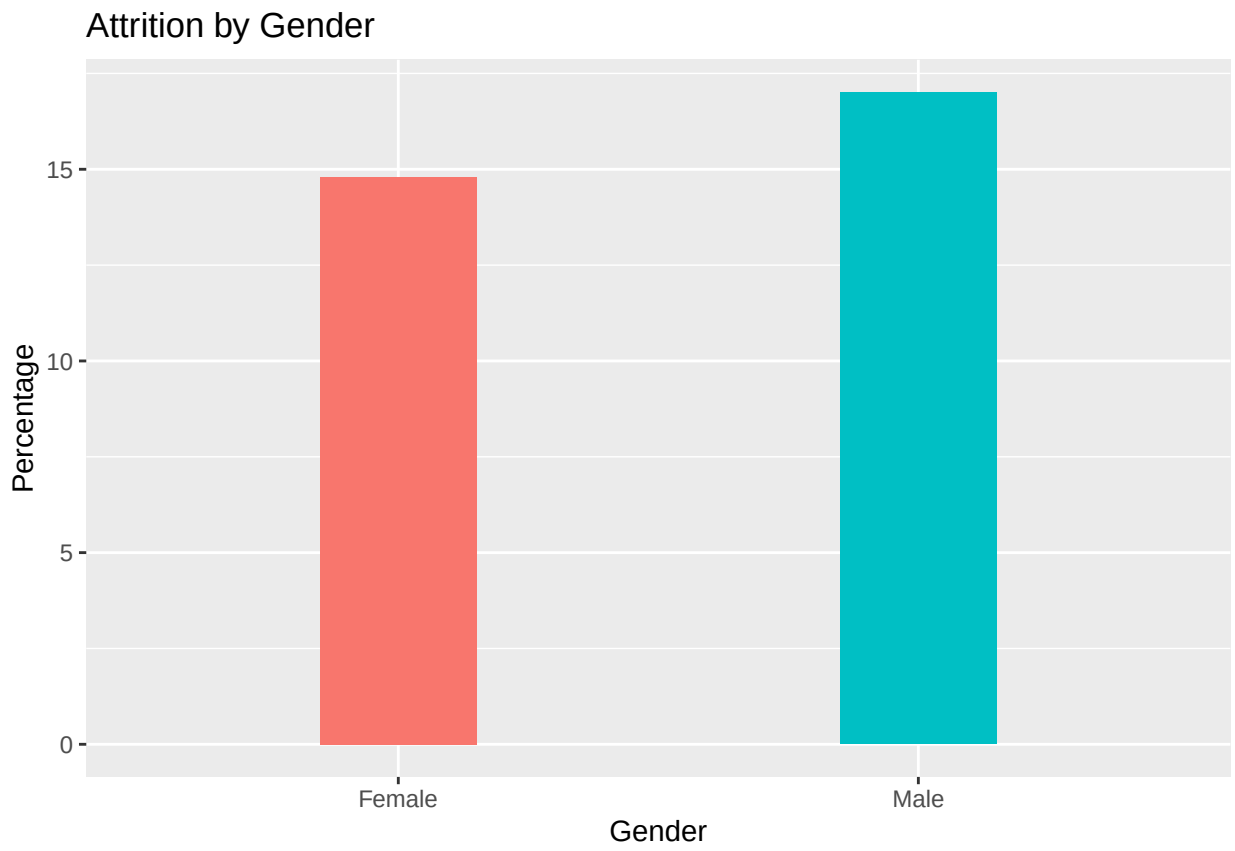
As expected, the attrition by monthly income graph shows that attrition level is higher when the monthly income is relatively low. The Hourly Rate – Attrition plot suggests that the highest level of attrition happens between the hourly rate 45 and 72. We cannot notice any significant difference in payment between females and males. When taking into consideration performance rating and monthly income, attrition is high when employees have a performance rating 4 (high) and low monthly income.

4) The impact of Gender:

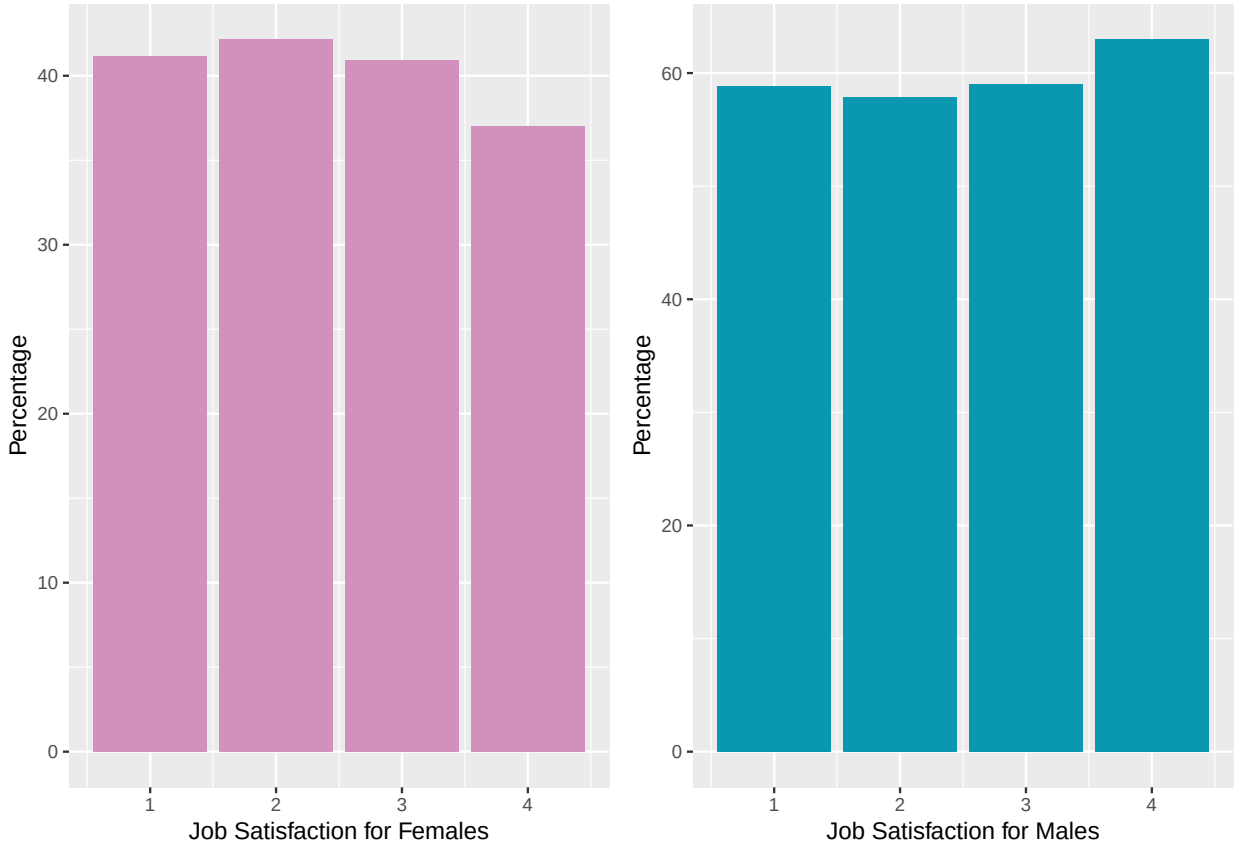
```
empl_by_gender <- df %>%
  ggplot(aes(x = Gender, fill = Gender)) +
  geom_bar(width = 0.5) + ylab("Number") +
  ggtitle("Number of Employees by Gender") +
  theme(legend.position = "none")
empl_by_gender
```



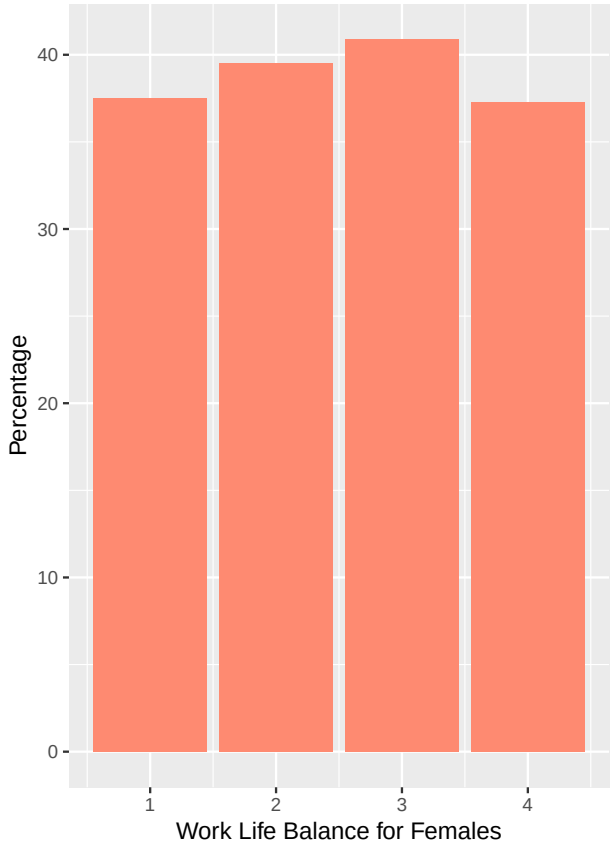

```
att_by_gender <- df %>% group_by(Gender) %>%
dplyr::summarise(Attrition_perc = sum (Attrition == "Yes")/n()*100) %>%
  ggplot(aes(x = Gender, y = Attrition_perc, fill = Gender)) +
  geom_col(width = 0.3) + theme(legend.position = "none") +
  ylab("Percentage") + ggtitle("Attrition by Gender")
att_by_gender
```

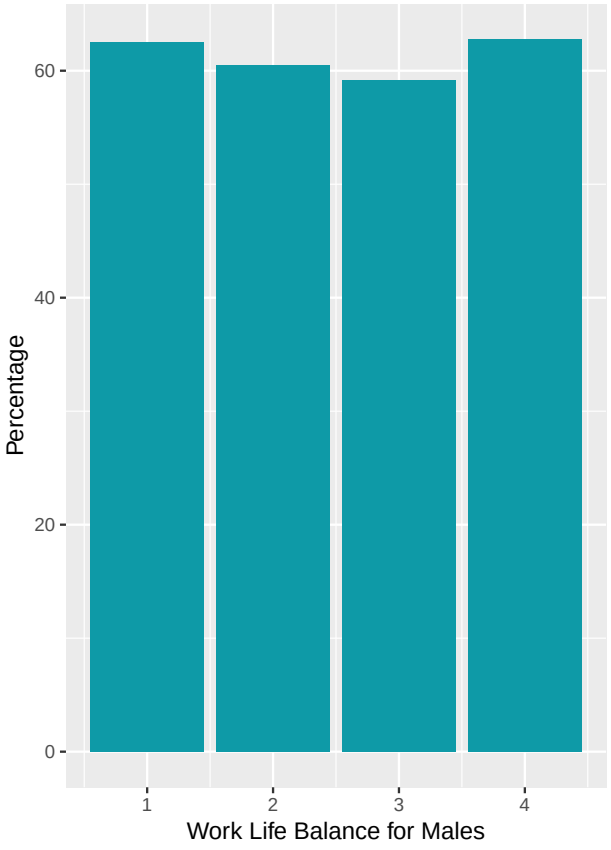



```
jobsat_female <- df %>% group_by(JobSatisfaction) %>%
  dplyr:: summarise(perc_f = sum (Gender == "Female")/n()*100) %>%
  ggplot() +
  geom_bar(aes(x = JobSatisfaction, y = perc_f),
           stat = "identity", fill = "#d291bc") +
  theme(legend.position = "none") + ylab("Percentage") +
  xlab("Job Satisfaction for Females") +
  theme_grey(base_size = 9)

jobsat_male <- df %>% group_by(JobSatisfaction) %>%
  dplyr::summarise(perc_m = sum (Gender == "Male")/n()*100) %>%
  ggplot() +
  geom_bar(aes(x = JobSatisfaction, y = perc_m),
           stat = "identity", fill = "#0a97b0") +
  theme(legend.position = "none") + ylab("Percentage") +
  xlab("Job Satisfaction for Males") +
  theme_grey(base_size = 9)
plott <- plot_grid(jobsat_female, jobsat_male)
plott
```




```
wl_balance_female <- df %>% group_by(WorkLifeBalance) %>%
  dplyr::summarise(perc_f = sum (Gender == "Female")/n()*100) %>%
  ggplot() +
  geom_bar(aes(x = WorkLifeBalance, y = perc_f),
           stat = "identity", fill = "#fe8a71") +
  theme(legend.position = "none") + ylab("Percentage") +
  xlab("Work Life Balance for Females") +
  theme_grey(base_size = 9)

wl_balance_male <- df %>% group_by(WorkLifeBalance) %>%
  dplyr::summarise(perc_m = sum (Gender == "Male")/n()*100) %>%
  ggplot() +
  geom_bar(aes(x = WorkLifeBalance, y = perc_m),
           stat = "identity", fill = "#0e9aa7") +
  theme(legend.position = "none") + ylab("Percentage") +
  xlab("Work Life Balance for Males") +
  theme_grey(base_size = 9)
plott <- plot_grid(wl_balance_female, wl_balance_male)
plott
```






```
wl_mean_f <- df %>% select(WorkLifeBalance, Gender) %>%
  filter(Gender == "Female") %>%
  summarise(ymean = mean(WorkLifeBalance))
 wl_mean_f #Work life balance mean rate for females
```

```
##      ymean
## 1 2.763605
```

```
 wl_mean_m <- df %>%
  select(WorkLifeBalance, Gender) %>%
  filter(Gender == "Male") %>%
  summarise(mean = mean(WorkLifeBalance))
 wl_mean_m #Work life balance mean rate for males
```

```
##       mean
## 1 2.759637
```

```
mean_salary_m <- df %>%
  select(Gender, MonthlyIncome) %>%
  filter(Gender == "Male") %>%
  summarize(mean=mean(MonthlyIncome))
mean_salary_m #Average monthly salary for men
```

```
##       mean
## 1 6380.508
```

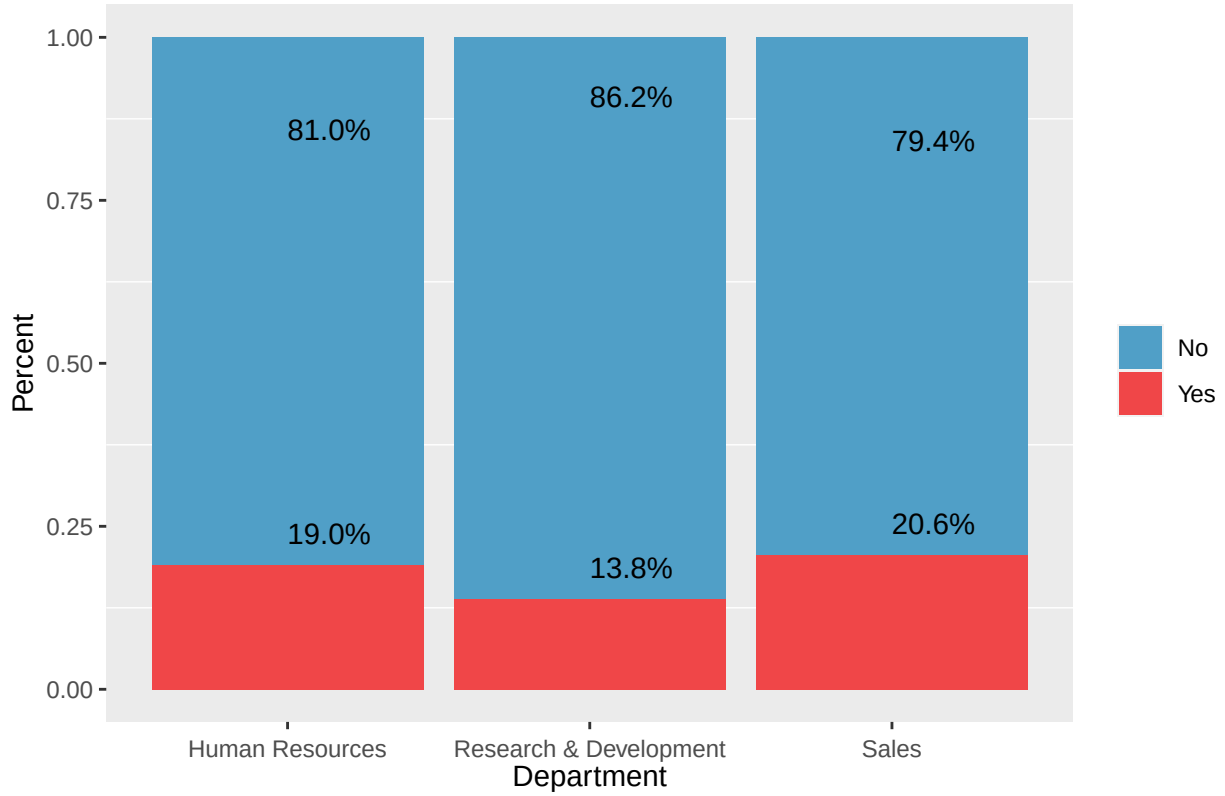
```r
mean_salary_m <- df %>% select(Gender, MonthlyIncome) %>%
  filter(Gender == "Female") %>% summarize(mean=mean(MonthlyIncome))
mean_salary_m #Average monthly salary for women
```

```
##       mean
## 1 6686.566
```

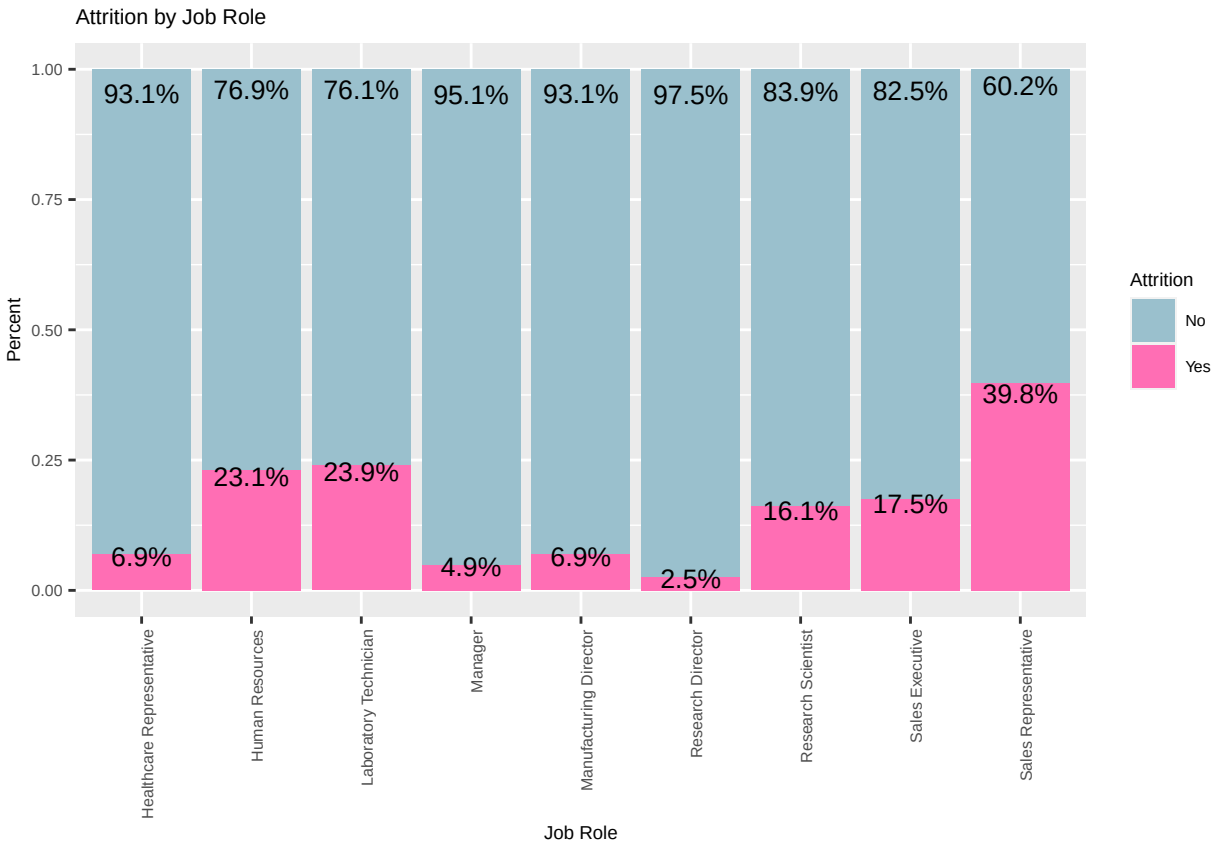
Summary for gender analysis: The attrition is higher for males. The average salaries are nearly the same for both genders. "Job satisfaction by gender" graph suggests that more males have a rate 4 of job satisfaction and more females have job a rate 1 of job satisfaction. The work life balance mean for males i s 2.759 and for females is 2.763605 and the average monthly salary for women is 6686.5 while the average for men is 6380.508.

5)Job role, Job position and satisfaction Analysis:

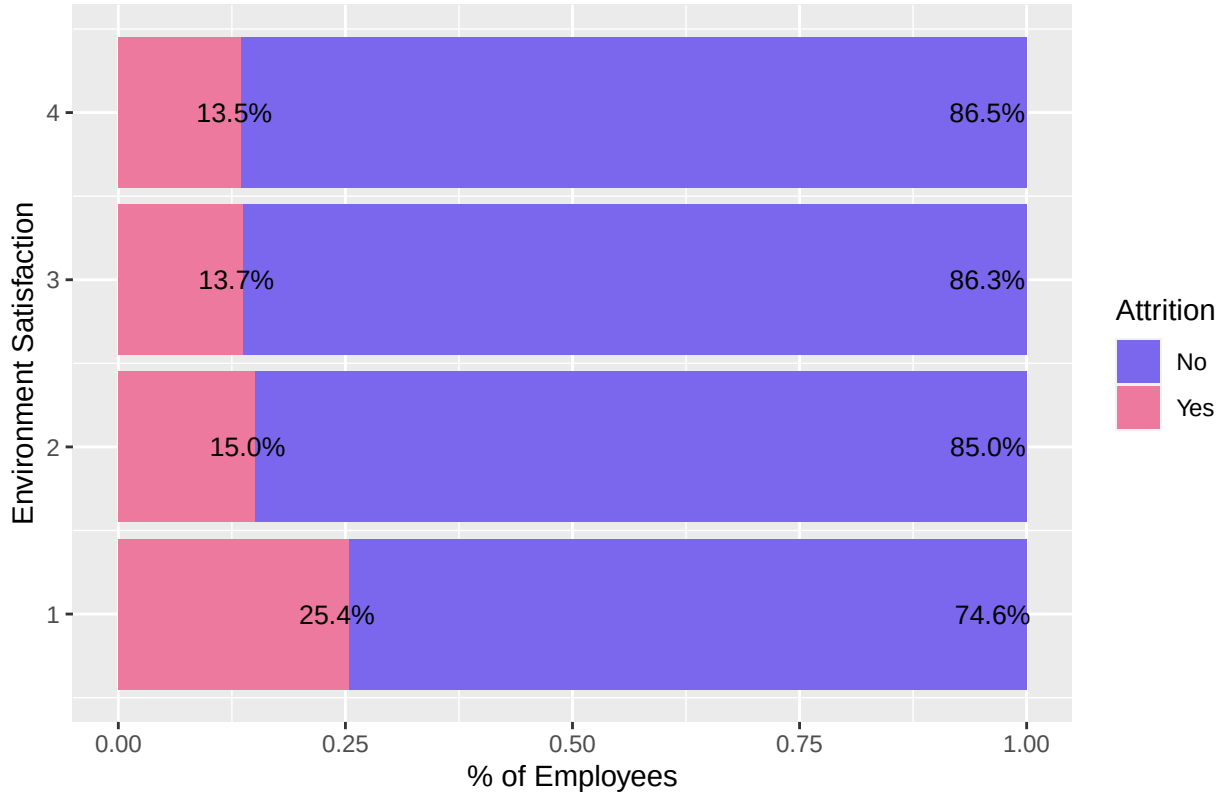
```r
att_by_dep <- df %>% group_by(Department) %>%
  dplyr::count(Attrition) %>% dplyr::mutate(pct=n/sum(n)) %>%
  ggplot(aes(x = Department, y = pct,
             fill = Attrition, order = Attrition)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label=paste0(sprintf("%1.1f", pct*100),"%"),
                hjust=0, vjust=-1)) +
  labs(title = "", y = "Percent", x = "Department") +
theme(legend.title = element_blank(),
panel.grid.major = element_blank(),
axis.text.x=element_text(hjust=0.5,vjust=0.05)) +
   scale_fill_manual(values = c("#509FC7", "#F04648"))
  att_by_dep
```



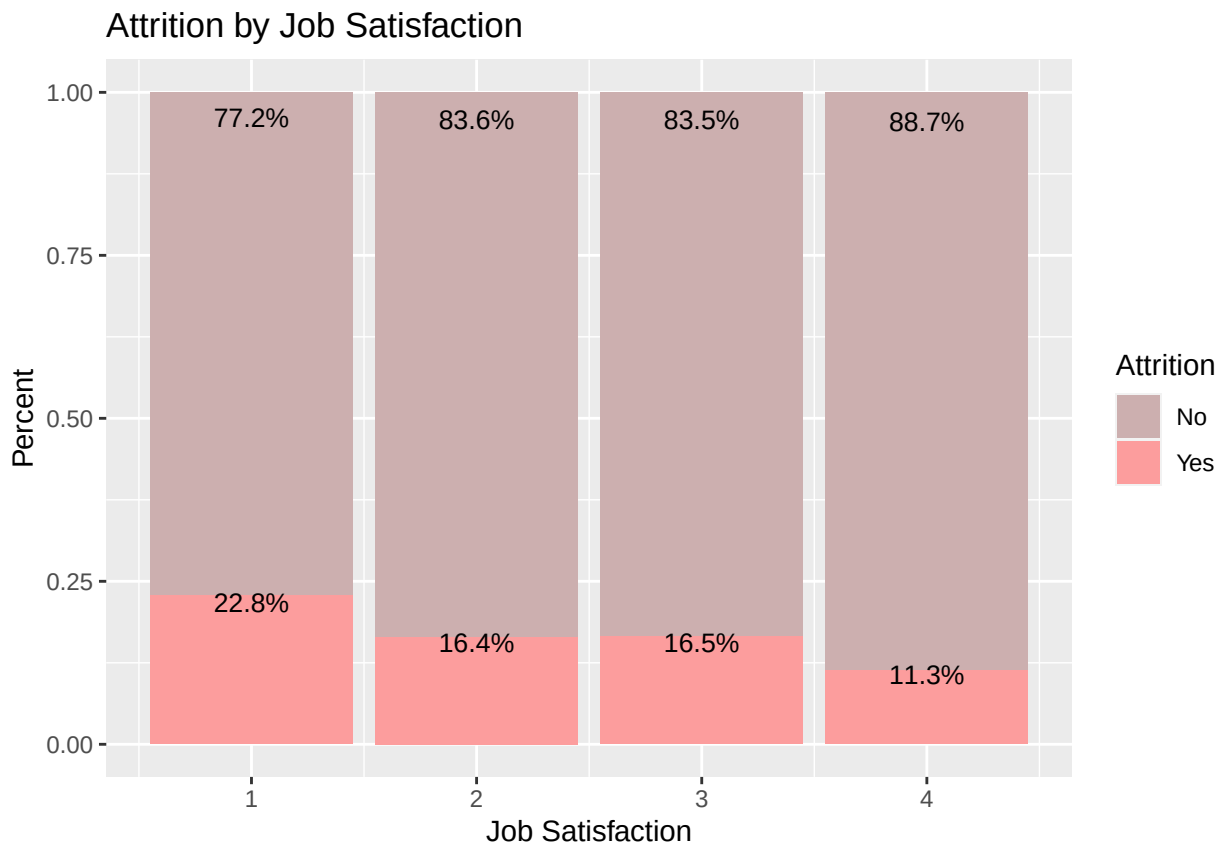

```
att_by_jobrole <- df %>% group_by(JobRole) %>%
  dplyr::count(Attrition) %>% dplyr::mutate(pct=n/sum(n)) %>%
  ggplot(aes(x = JobRole, y = pct,
             fill = Attrition, order = Attrition)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label=paste0(sprintf("%1.1f", pct*100),"%")), position=position_stack(0.95), size = 3.6
  labs(title = "", y = "Percent", x = "Job Role") +
  scale_fill_manual(values = c("lightblue3", "hotpink1")) +
  theme(text = element_text(size=7),
        axis.text.x = element_text(angle=90, hjust=1)) +
  ggtitle("Attrition by Job Role")
att_by_jobrole
```



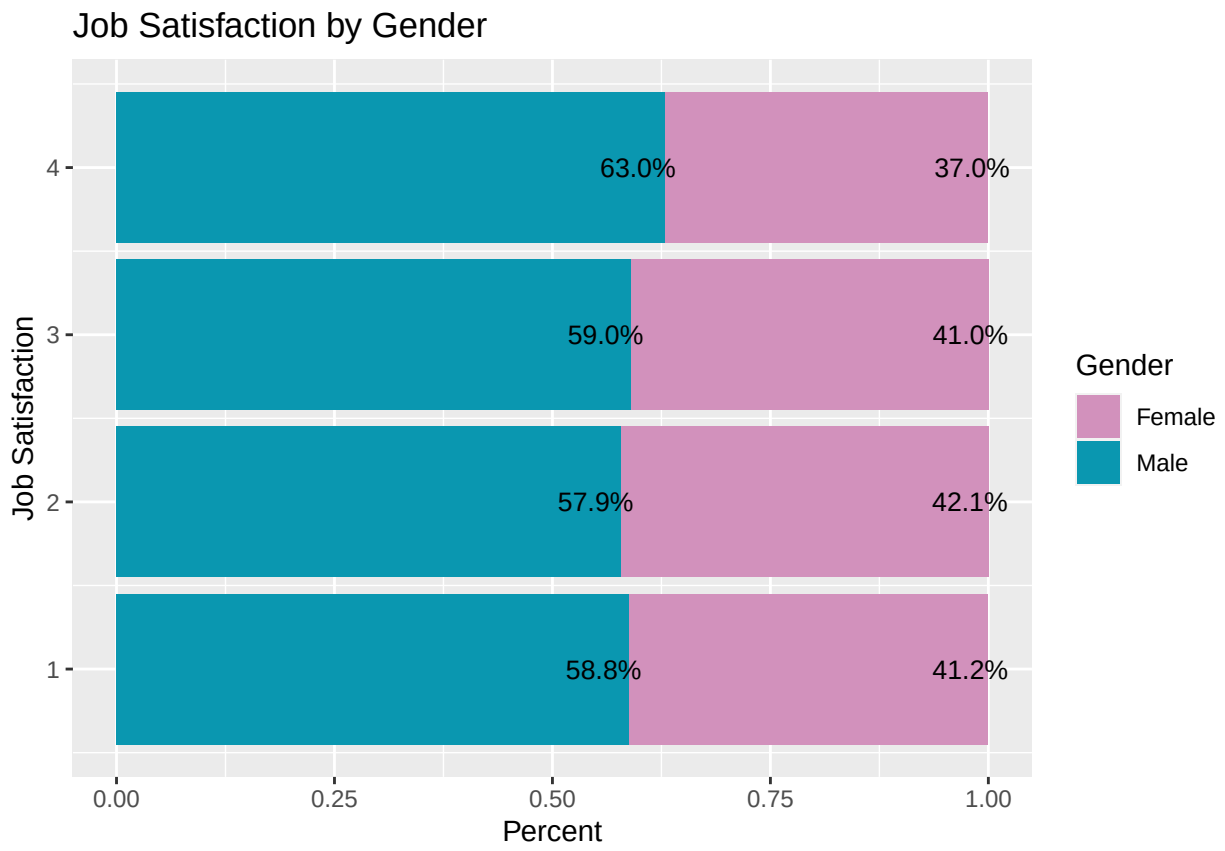

```
env_sat <- df %>% group_by(EnvironmentSatisfaction) %>%
  dplyr::count(Attrition) %>% dplyr::mutate(pct=n/sum(n)) %>%
  ggplot(aes(x = EnvironmentSatisfaction, y = pct,
             fill = Attrition, order = Attrition)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label=paste0(sprintf("%1.1f", pct*100),"%")), position=position_stack(0.95), size = 3.6
  labs(title = "", y = "% of Employees",
       x = "Environment Satisfaction") +
scale_fill_manual(values = c("slateblue2", "palevioletred2")) +
  coord_flip()

env_sat
```




```
job_sat <- df %>% group_by(JobSatisfaction) %>%
  dplyr::count(Attrition) %>% dplyr::mutate(pct=n/sum(n)) %>%
  ggplot(aes(x = JobSatisfaction, y = pct,
             fill = Attrition, order = Attrition)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label=paste0(sprintf("%1.1f", pct*100),"%")), position=position_stack(0.95), size = 3.6
  labs(title = "", y = "Percent", x = "Job Satisfaction") + scale_fill_manual(values = c("#ccafaf", "#f
  ggtitle("Attrition by Job Satisfaction")
job_sat
```

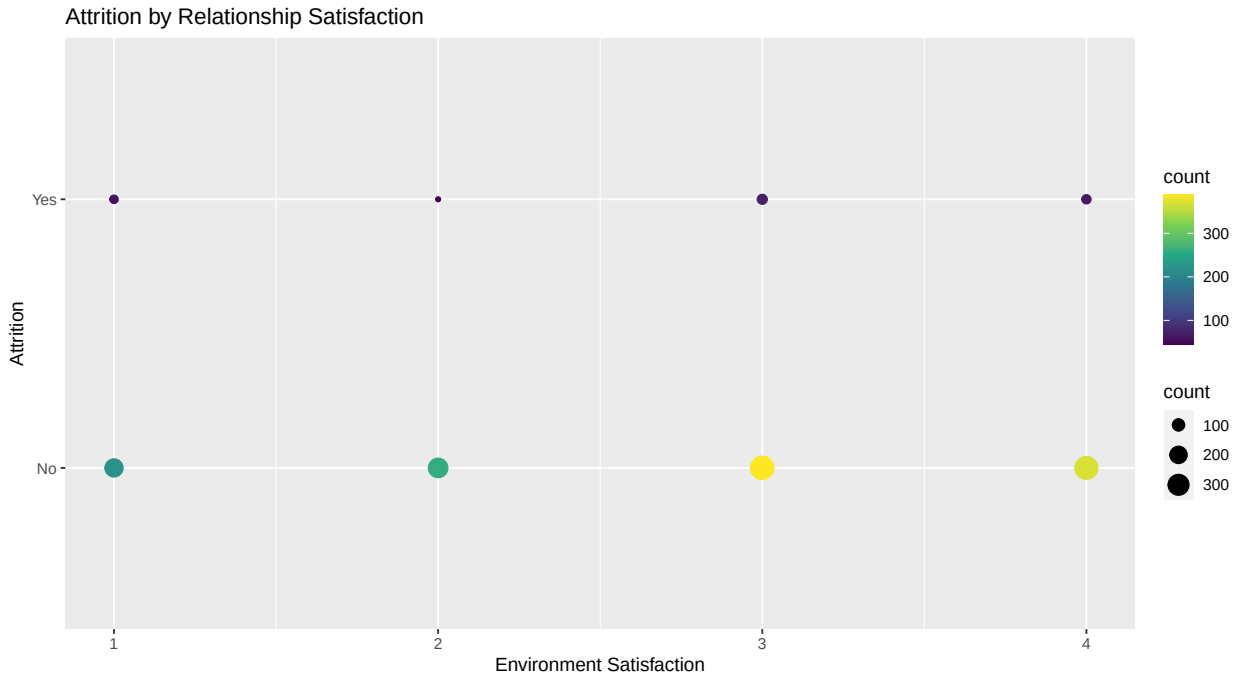


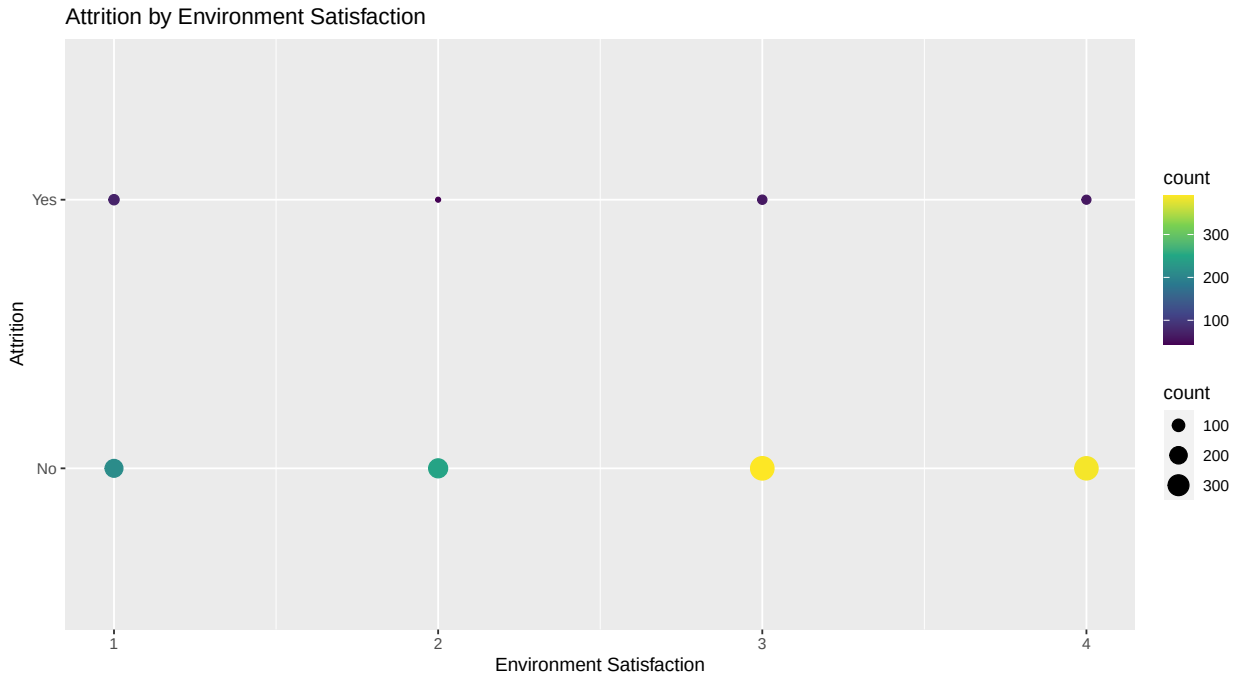

```
job_sat_gender <- df %>% group_by(JobSatisfaction) %>%
  dplyr::count(Gender) %>% dplyr::mutate(pct=n/sum(n)) %>%
  ggplot(aes(x = JobSatisfaction, y = pct,
             fill = Gender, order = Gender)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label=paste0(sprintf("%1.1f", pct*100),"%")), position=position_stack(0.95), size = 3.6
  labs(title = "", y = "Percent", x = "Job Satisfaction") + scale_fill_manual(values = c("#d291bc", "#0
  ggtitle("Job Satisfaction by Gender") +  coord_flip()
job_sat_gender
```




```
df %>% ggplot(aes(x = YearsWithCurrManager,
                  y = JobSatisfaction, color = Attrition)) +
  geom_jitter(alpha = 0.5) +  theme(legend.position = "bottom") +
  ylab("Job Satisfaction") + xlab("Years with Current Manager") +
  scale_color_manual(values = c("#d291bc", "#0a97b0"))
```

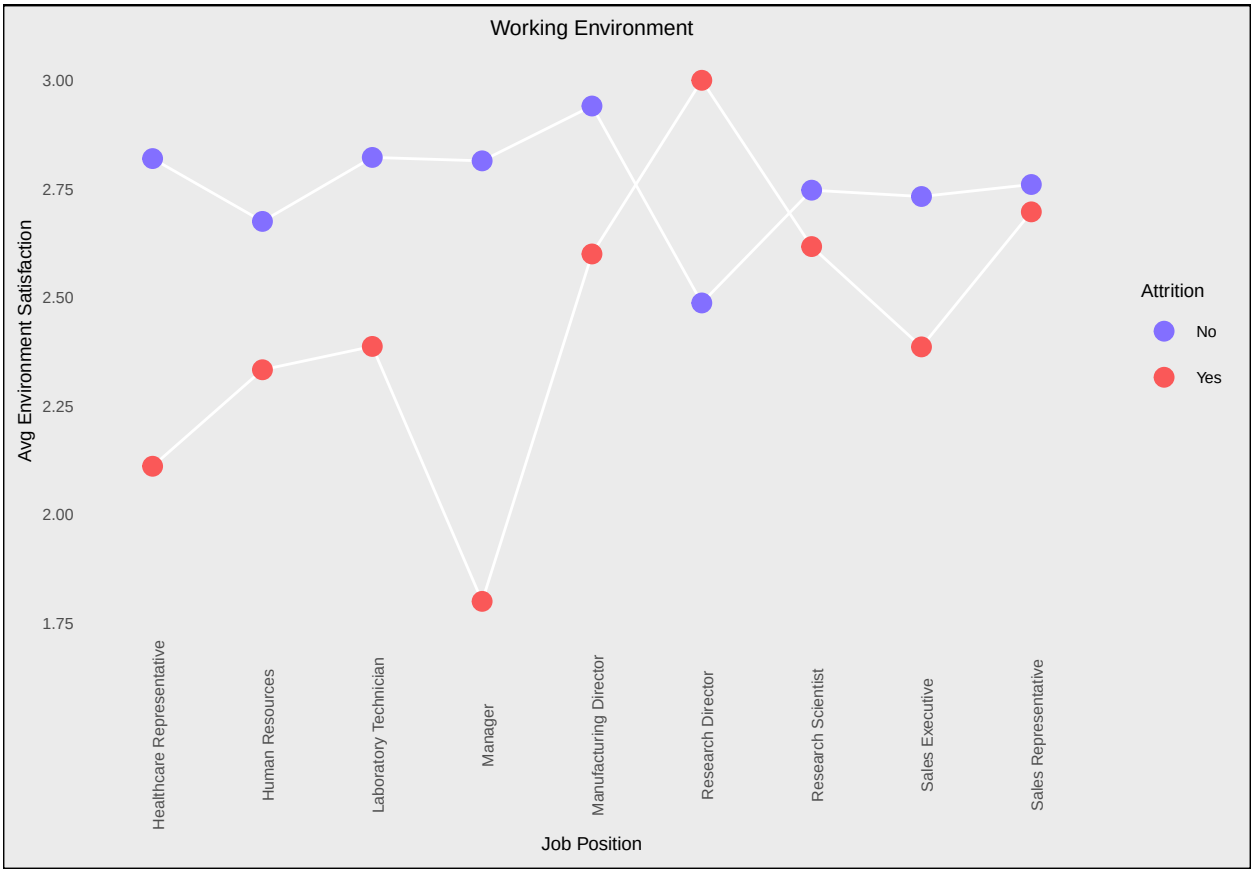

```
att_relat_sat <- df %>%
  group_by(RelationshipSatisfaction, Attrition) %>%
  dplyr::summarise(count=n()) %>%
  ggplot(aes(x = RelationshipSatisfaction,
             y = Attrition)) +
  geom_point(aes(color = count, size = count)) +
  ggtitle("Attrition by Relationship Satisfaction") +
  xlab("Environment Satisfaction") + scale_color_viridis_c()



att_env_sat<- df %>%
  group_by(EnvironmentSatisfaction, Attrition) %>%
  dplyr::summarise(count=n()) %>%
  ggplot(aes(x = EnvironmentSatisfaction,
             y = Attrition)) +
  geom_point(aes(color = count, size = count)) +
  ggtitle("Attrition by Environment Satisfaction") +
  xlab("Environment Satisfaction") + scale_color_viridis_c()



plott <- plot_grid(att_relat_sat, att_env_sat, ncol = 1)
plott
```






```
attrition_by_job_level_role <- df %>%
  ggplot(aes(x = JobRole, y = JobLevel)) +
  geom_jitter(aes(color = Attrition), alpha = 0.6) +
  labs(title = "Attrition by Job Role and Job Level",
       y = "Job Level", x = "Job Role") +
  theme(legend.position="bottom",
        text = element_text(size=9)) + coord_flip()
attrition_by_job_level_role
```


```
df$EnvironmentSatisfaction <- as.numeric(df$EnvironmentSatisfaction)
job_position_env_satisf <- df %>%
  select(EnvironmentSatisfaction, JobRole, Attrition) %>%
  group_by(JobRole, Attrition) %>%
dplyr::summarize(avg_env=mean(EnvironmentSatisfaction)) %>%
ggplot(aes(x=JobRole, y=avg_env)) +
  geom_line(aes(group=Attrition),
            color="white") +
geom_point(aes(color=Attrition), size=3) +
  theme_minimal() + theme(plot.title=element_text(hjust=0.5), axis.text.x=element_text(angle=90),
                                                                plot.background=element_rect(fill=
                          text = element_text(size=7)) +
labs(title="Working Environment",
     y="Avg Environment Satisfaction", x="Job Position") + scale_color_manual(values=c("slateblue1", "#
job_position_env_satisf
```

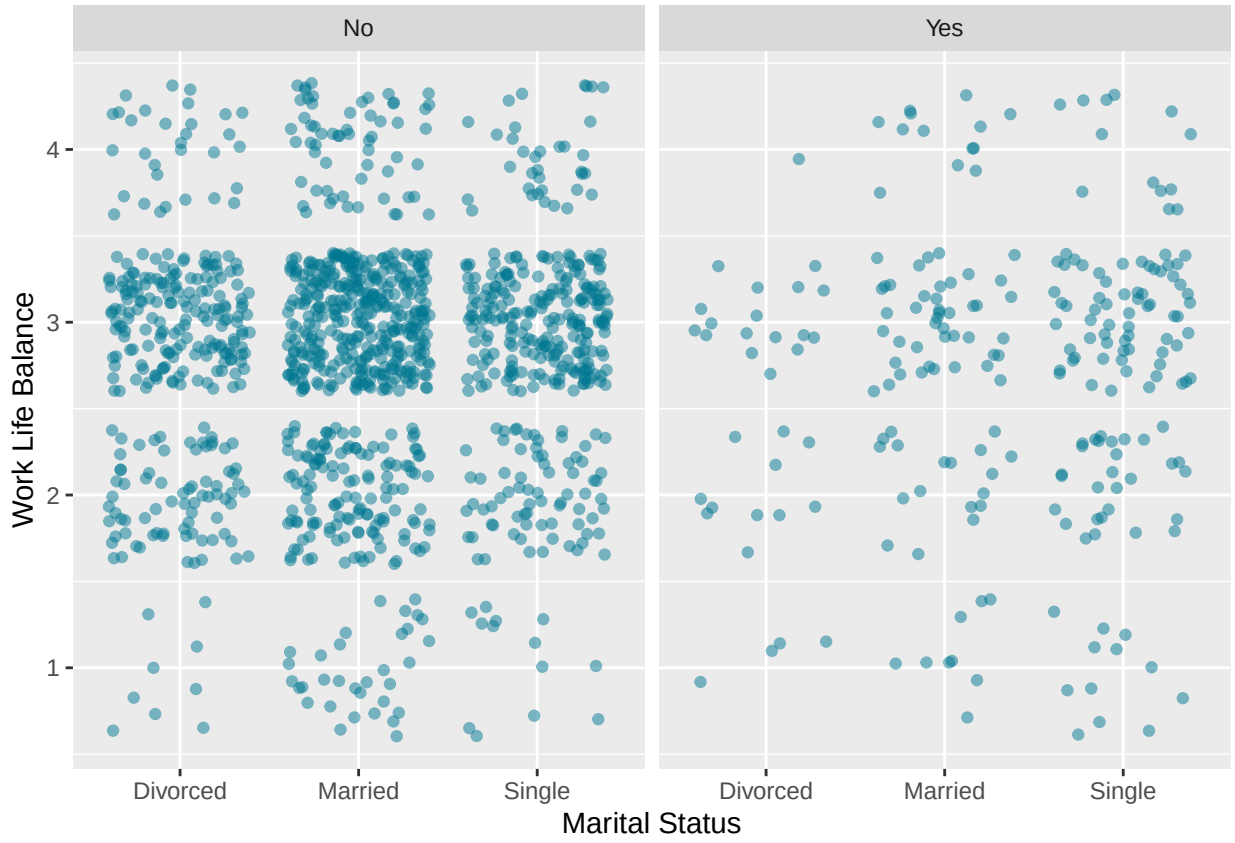


Summary for job role, position and satisfaction Analysis:

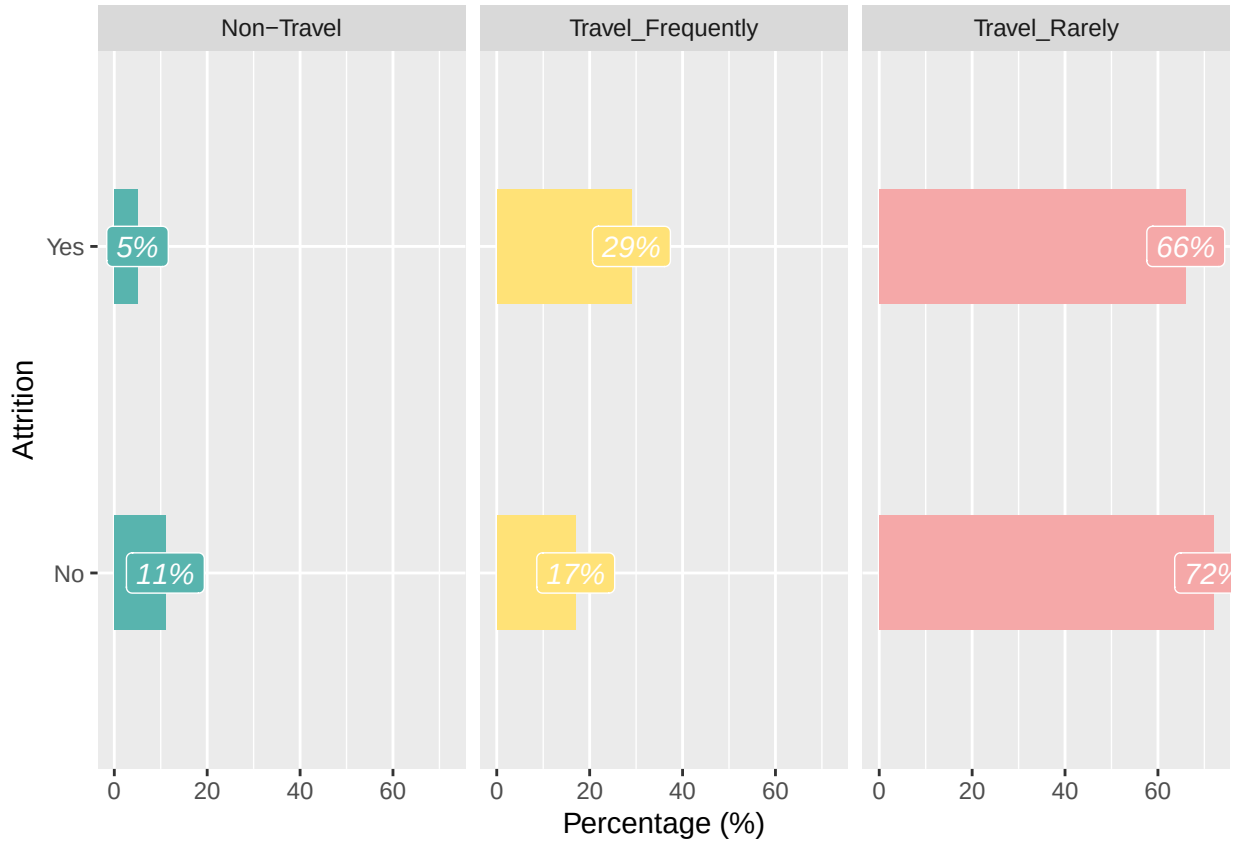
-The Sales Department have the highest percentage of attrition by 20.6%. -Three job roles that have more employees leaving are Sales Representative (39.8%), Laboratory Technician (23.9%) and Human Resources (23.1%). -Employees who have the lowest rate of Environment Satisfaction are the ones with highest level of attrition. -The Job Satisfaction by Gender graph shows that more males are satisfied from their job than females.
-Employees who work with new managers have a lower satisfaction score and higher attrition than employees who work with managers that have been there for a longer time. -We cannot also see any clear correlation between Job Satisfaction and Years Current Manager. -Environment Satisfaction and Relationship Satisfaction appear to have quite similar impact on Attrition, and it is slightly higher when the satisfaction level is 1 and 3. -Managers and healthcare representatives that have left the company have lower average environment satisfaction. From the employees that have not left the organization, the research directors have the lowest average environment satisfaction

6) Other factors:

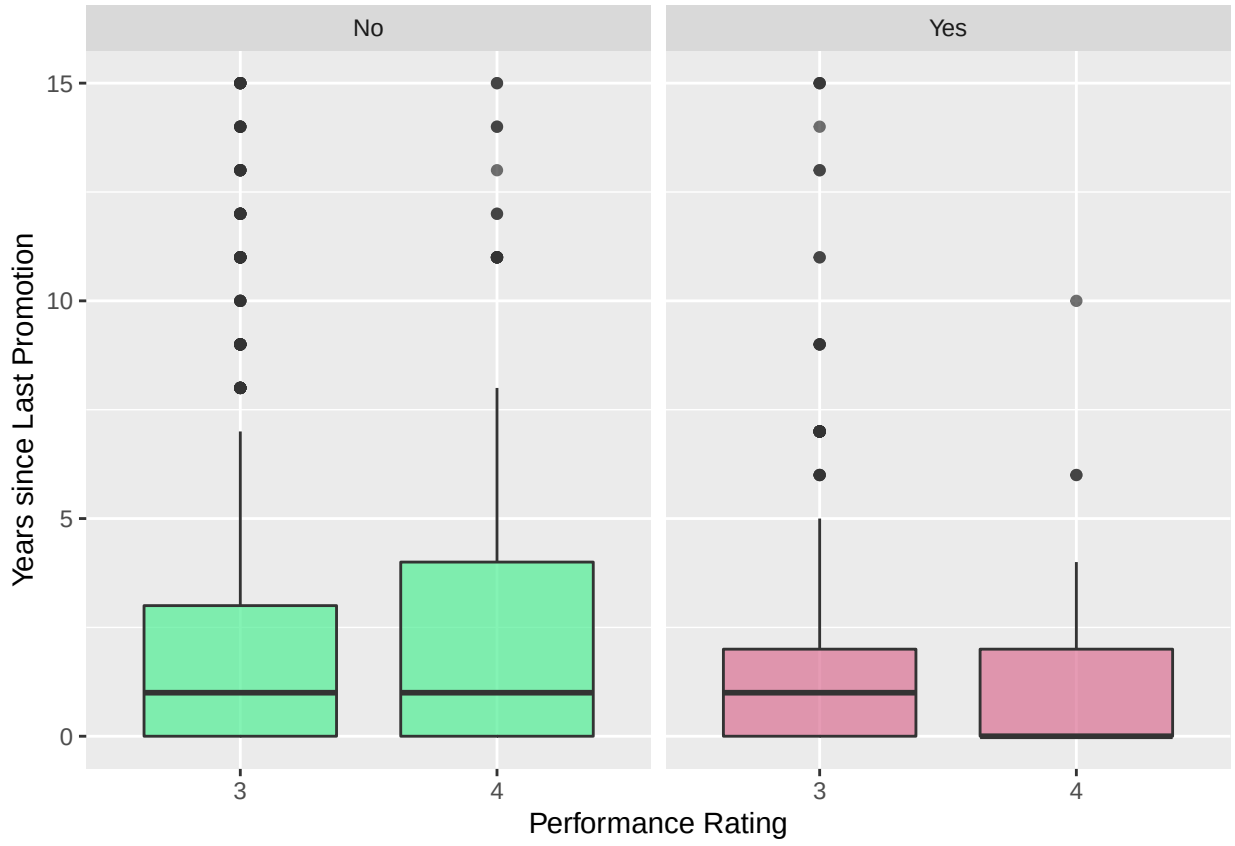
```
attr_wl_maritalst <- df %>%
  ggplot(aes(x = MaritalStatus,
             y = WorkLifeBalance, color = "#4863c7")) +
  geom_jitter(stat = "identity", alpha = 0.5) +
 # scale_fill_manual(values=c(, "#e35146"))  +
  facet_wrap(~Attrition) +
  theme(legend.position = "none")  +
  ylab("Work Life Balance") + xlab("Marital Status") + scale_color_manual(values=c("#007892", "#ff427f"

attr_wl_maritalst
```



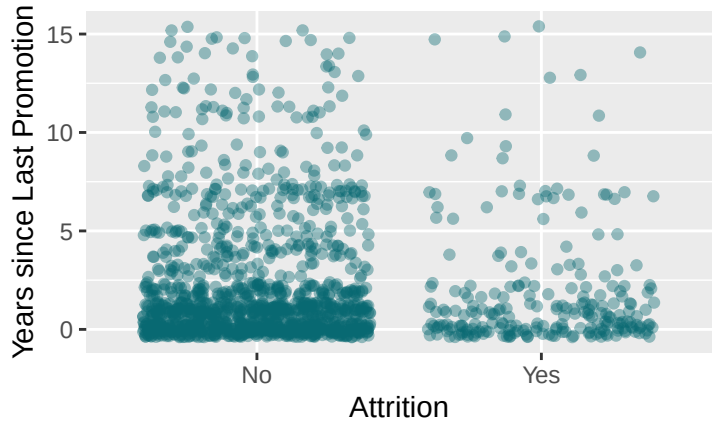

```
travel_pct <- df %>% group_by(Attrition, BusinessTravel) %>%
dplyr::summarize(count=n()) %>%
  dplyr::mutate(pct=round(prop.table(count),2) * 100) %>%
ggplot(aes(x=Attrition, y=pct, fill=BusinessTravel)) + geom_bar(stat='identity', width = 0.35) +
  facet_wrap(~BusinessTravel)  +
theme(legend.position="none") +
geom_label(aes(label=paste0(pct, "%"), fill = BusinessTravel),
           colour = "white", fontface = "italic")  +
scale_fill_manual(values=c("#58b4ae", "#ffe277", "#f5a8a8")) +
scale_color_manual(values = c("snow4", "snow4", "snow4" )) + labs(x="Attrition", y="Percentage (%)") + c
travel_pct
```




```
Performance.Rating <- as.factor(df$PerformanceRating)
plot5 <- df %>%
  ggplot(aes(x= Performance.Rating,
             y=YearsSinceLastPromotion,
             group = Performance.Rating, fill = Attrition)) +
        geom_boxplot(alpha=0.7) +
        theme(legend.position="none") +
        facet_wrap(~ Attrition) +
  xlab("Performance Rating") +
  ylab("Years since Last Promotion") +
  scale_fill_manual(values = c("seagreen2", "palevioletred"))
plot5
```

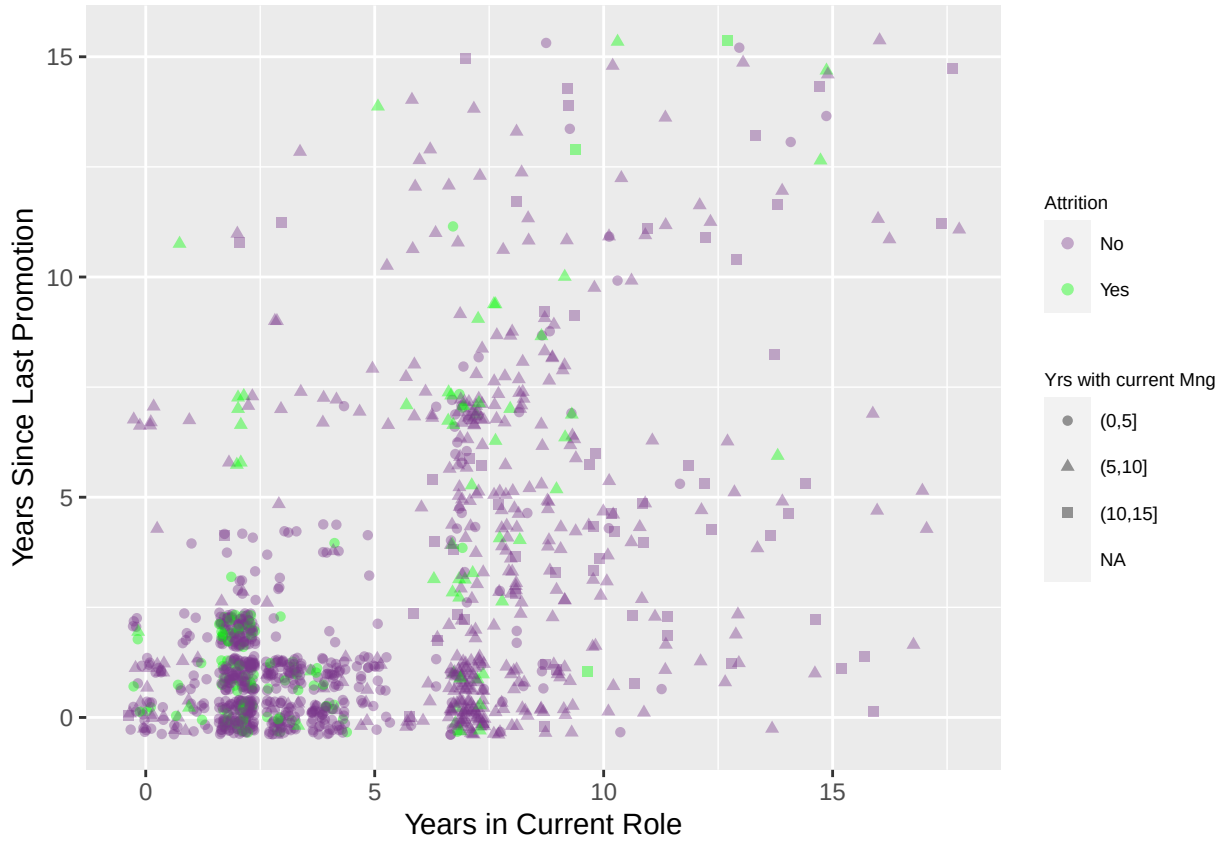



```
df %>% ggplot(aes(x = Attrition,
                  y = YearsSinceLastPromotion)) +
  geom_jitter(alpha = 0.4, color = "#086972") +
  ylab("Years since Last Promotion")
```

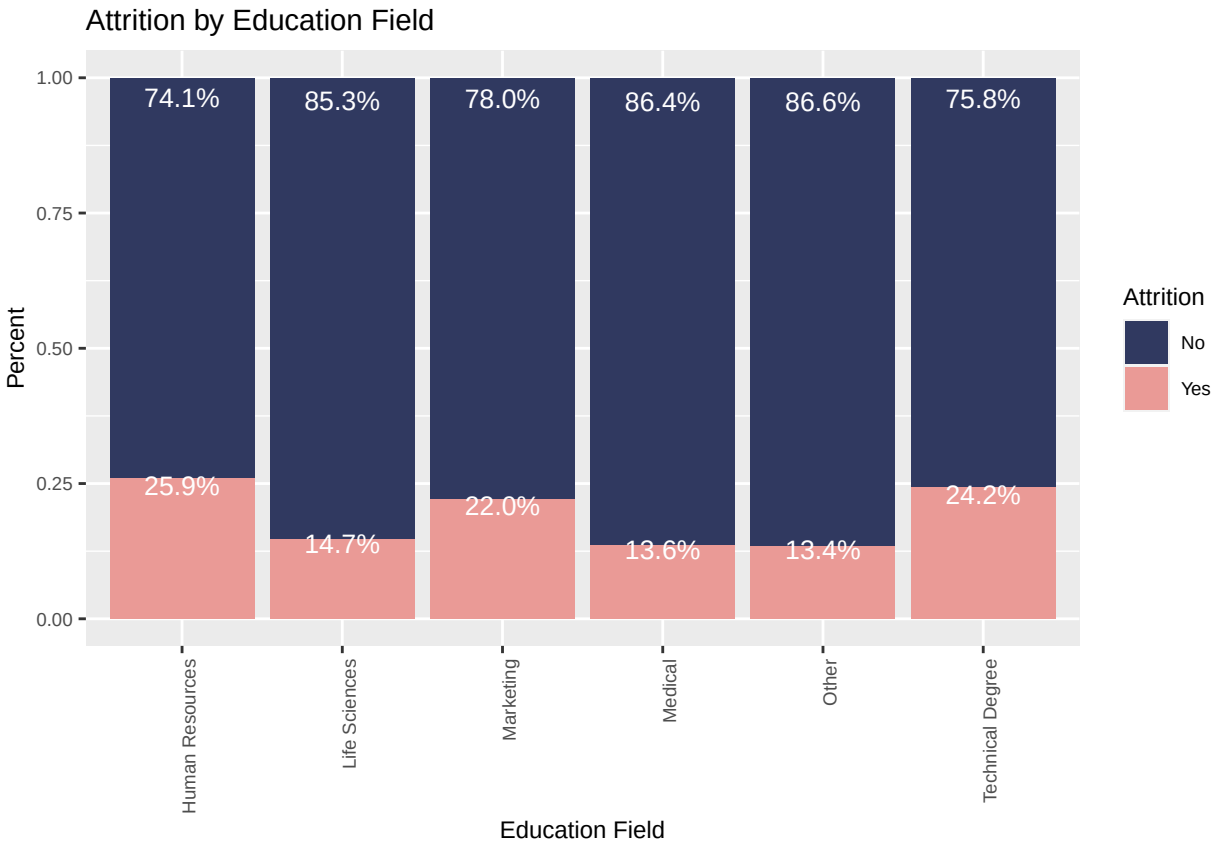



```
YwM_split <- cut(df$YearsWithCurrManager, seq(0, 17, 5))
df %>%
  ggplot(aes(x = YearsInCurrentRole,
             y = YearsSinceLastPromotion,
             color = Attrition)) +
```

```
  geom_jitter(aes(shape = YwM_split),
              alpha = 0.4, size = 1.5) + scale_color_manual(values=c("Yes"="green",
                                                                     "No"="mediumorchid4")) +
  labs(shape = "Yrs with current Mng") +  theme(legend.title=element_text(size=7),
                                               legend.text=element_text(size=7)) +
  xlab("Years in Current Role") +
  ylab("Years Since Last Promotion")
```




```
att_by_educ <- df %>% group_by(EducationField) %>%
  dplyr::count(Attrition) %>% dplyr::mutate(pct=n/sum(n)) %>%
  ggplot(aes(x = EducationField, y = pct,
             fill = Attrition, order = Attrition)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label=paste0(sprintf("%1.1f", pct*100),"%")), position=position_stack(0.95), size = 3.6
  labs(title = "", y = "Percent", x = "Education Field") + scale_fill_manual(values = c("#303960", "#ea
  theme(text = element_text(size=9),
        axis.text.x = element_text(angle=90, hjust=1)) +
  ggtitle("Attrition by Education Field")
att_by_educ
```



Analysis for Other Factors: -The employees with Marital Status - Single that left the organization had a lower Work Life Balance than other former employees. - Business Travel plot shows us that employees that travel frequently have the highest percentage of Attrition. - Attrition is higher when Performance Rating is 3. When comparing it with Years since Last Promotion, most of the Attrition is positioned around years 0 – 3 of Years since Last Promotion. - The Attrition by Education Field graph suggests that Human Resource education field have the highest percentage of Attrition.

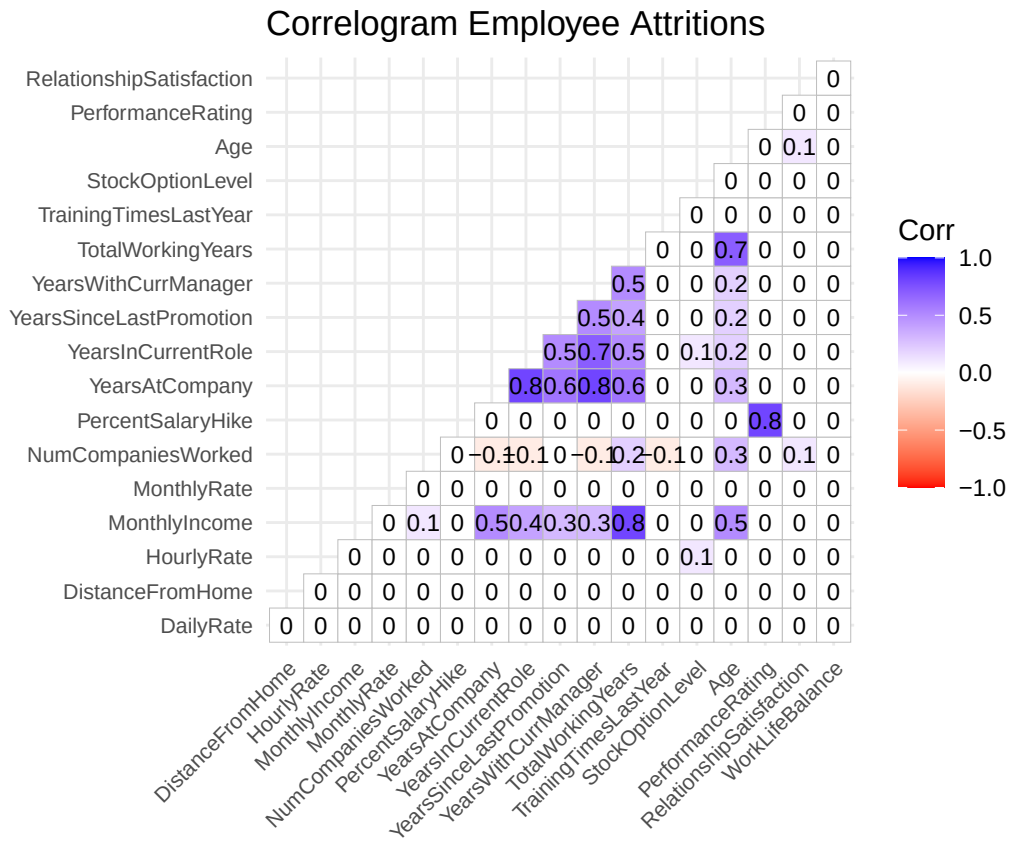
In the next section we will build the correlation matrix:

```
df <- read_csv("~/WA_Fn-UseC_-HR-Employee-Attrition.csv")
```

```
## 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.
```

```
numeric <- df %>% dplyr::select(DailyRate,DistanceFromHome,HourlyRate,
MonthlyIncome,MonthlyRate,NumCompaniesWorked, PercentSalaryHike, YearsAtCompany,YearsInCurrentRole, Yea

matr <- dplyr::select_if(numeric, is.numeric)
correl <- round(cor(matr), 1)


matrix1 <- ggcorrplot(correl,
           lab = TRUE,
           lab_size = 3,
           type = "lower",
           method="square",
           colors = c("red", "white", "blue"),
           title="Correlogram Employee Attritions",
           ggtheme=theme_minimal()) +
  theme(axis.text.x=element_text(size = 8),
        axis.text.y=element_text(size = 8))
matrix1
```



Monthly Income has a strong positive correlation with Total Working Years and a positive correlation with Years at Company and Age. Number of Companies Worked have a weak positive correlation with Age. The higher Years at Company, the higher Monthlye Income. The higher Years in Current Role, the higher Years with Current Manager.

Now let us plot the decision tree, an algorithm that is used to solve both Regression and Classification problems:

```
df <- read_csv("~/WA_Fn-UseC_-HR-Employee-Attrition.csv")
```

```
## 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.
```

```
#coerce these columns into factors
cols <- c("Education", "EnvironmentSatisfaction",
          "JobInvolvement", "JobLevel",
         "JobSatisfaction", "PerformanceRating",
         "RelationshipSatisfaction",
         "StockOptionLevel", "TrainingTimesLastYear", "WorkLifeBalance")
df[cols] <- lapply(df[cols], factor)

shuffle_index <- sample(1:nrow(df))
#head(shuffle_index)


df2 <- df
#create test train
create_train_test <- function(df2, size = 0.8, train = TRUE) {
  n_row = nrow(df2)
  total_row = size * n_row
  train_sample <- 1: total_row
  if (train == TRUE) {
    return (df2[train_sample, ])
  } else {
    return (df2[-train_sample, ])
  }
}
```

```
#test the function and check the dimension
data_train <- create_train_test(df2, 0.8, train = TRUE)
data_test <- create_train_test(df2, 0.8, train = FALSE)
dim(data_train)
```

```
## [1] 1176   35
```

```r
dim(data_test)
```

```
## [1] 294  35
```

The model correctly predicted 1176 NO-Attrition cases and classified 35 No-Attrition as Attrition. It also misclassified 294 employees as negative-Attrition employees but it turned out they had left.

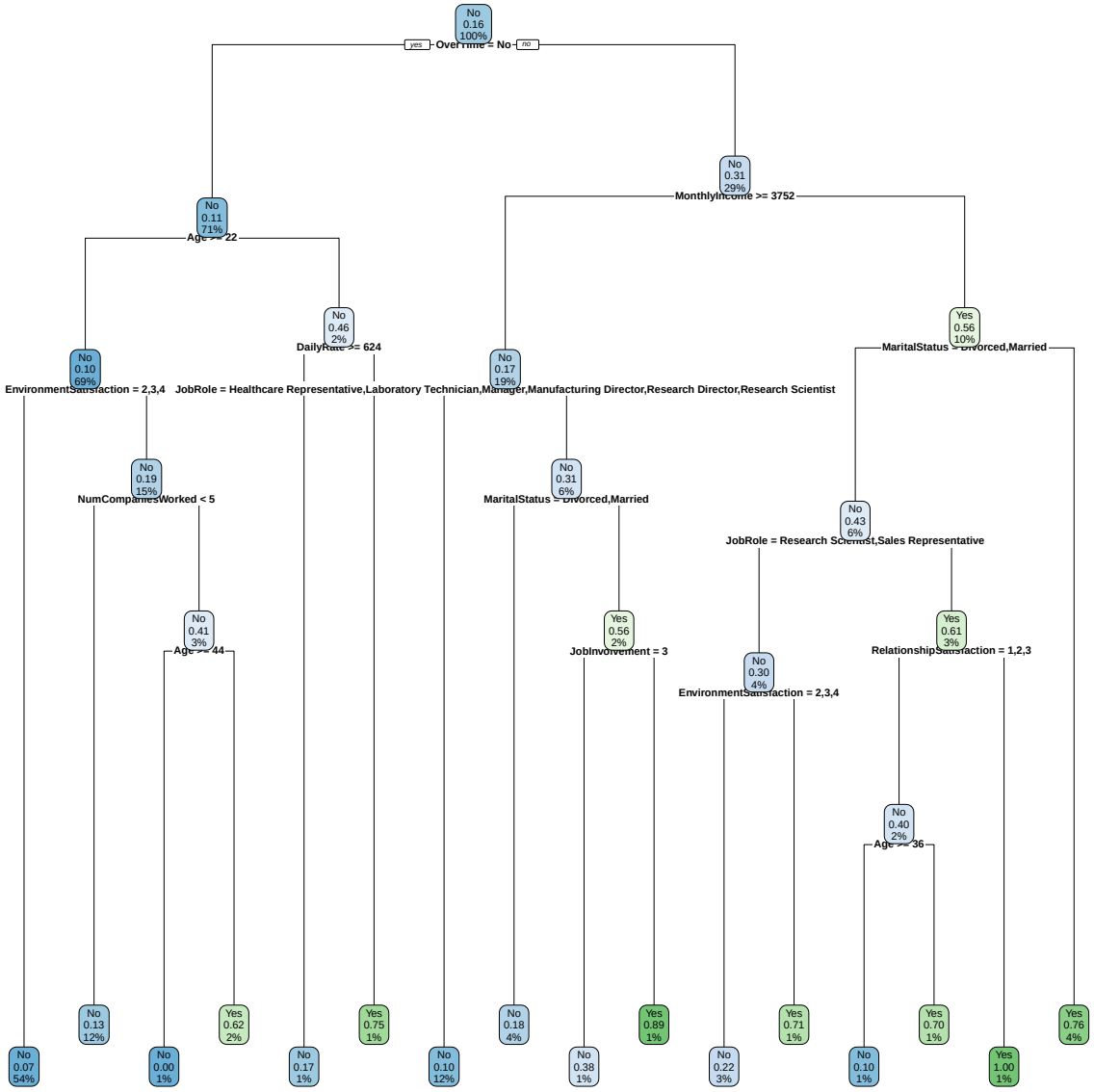
```r
#use the function prop.table() combined with table() to verify if the randomization process is correct.
prop.table(table(data_train$Attrition))
```

```
##
##       No      Yes
## 0.835034 0.164966
```

```r
prop.table(table(data_test$Attrition))
```

```
##
##        No       Yes
## 0.8537415 0.1462585
```

```r
#building the model
fit <- rpart(Attrition~., data = data_train,
             method = 'class')
rpart.plot(fit, extra = 106, cex = 0.55)
```




```
#predict the test dataset
predict_unseen <-predict(fit, data_test, type = 'class')
#testing the employees who left the company and those who did not
table_mat <- table(data_test$Attrition, predict_unseen)
table_mat
```

```
##      predict_unseen
##        No Yes
##   No  236  15
##   Yes  34   9
```

```
#measure the performance with a confusion matrix
accuracy_Test <- sum(diag(table_mat)) / sum(table_mat)
print(paste('Accuracy for test', accuracy_Test))
```

```
## [1] "Accuracy for test 0.833333333333333"
```

```
#we have a score of 83.3 %, which is a relatively high accuracy so we
#will leave it as it is (for the moment)
```