University of Exeter Business School

BEM2031 Introduction to Business Analytics

Assignment

Academic Year 2023 / 2024

Student ID: 710032265

Contents

1 Understanding the Business

1.1 The Goals of the Project

The overarching goal of this project entails conducting an extensive evaluation of employee statistics, particularly focusing on instances of employees leaving the company. This involves the meticulous examination of data to discern patterns and draw conclusive insights. Then, to provide the employers a detailed breakdown of steps to help reduce the number of employees leaving. This includes what steps to take, how to implement them and which departments to focus on first .

1.2 The Analysis

The data analysis is thorough yet incomplete and inaccurate. The report identifies trends that mostly use satisfaction level, which is not the most useful variable and is the only purely subjective result that is skewed for bad employees. Further, and more importantly, the report has no breakdown of department, it simply searches for trends in the data across the whole company. Then still offers no actionable solutions to the tailored to the wider company or specific departments.

The costs of this analysis of employee retention primarily include the resources and time invested in collecting, organising, and analysing the data. Additionally, there would inevitably be costs associated with implementing any recommendations derived from the analysis, had there been any recommendations. These could include training programs or policy changes. On the other hand, the benefits of this analysis could potentially be significant. By identifying patterns and trends in employee turnover, the company can take proactive measures to address underlying issues, ultimately improving employee retention. This can lead to higher employee morale, increased productivity, and ultimately, cost savings associated with recruiting and training. However, the benefits are to be limited as the analysis fails to provide any actionable insights tailored to the company or the specific departments.

From this specific report it is unlikely that anyone would benefit or be potentially harmed as the report writer makes no actionable recommendations. However, an employer may take their own insights from the report. For example as the Random Forest implies that ‘Satisfaction Level’ and ‘Time spent in the Company’ are the strongest variables. The employer may then be more inclined to let go employees they deem to be unhappy, or that spend what they perceive to be too much time at work.

Furthermore, it is suggested that ‘Salary’ is not an important factor to employee retention, this statistic would only drive employers to give fewer raises, and offer lower starting salaries, increasing the harm to employees. Moreover, given deeper analysis, this data trend may not be accurate for all departments and could actually lead to an increase in employee turnover.

Overall, this report is unlikely to produce any benefits to anyone as it has not shown any actionable solutions to be implemented.

2 Understanding the Data

2.1 The Data

The data is somewhat appropriate, however it lacks some important additions and useful terminologies. Firstly, there is no information on when any of the data was collected. This is important as had this data been available, trends over time would have been able to be analysed. This would help contribute towards making actionable changes to potential managers and staff. Further, the term "work accident" may not fully encompass all work-related incidents, such as cases involving abuse or assault. This suggests potential gaps in the coverage of workplace safety concerns. Finally, while the inclusion of salary and department data is valuable, converting these string variables into numerical values could enhance the analytical capabilities robustness of the analysis.

There is a multitude of information that is unknown to the reader of the report. From the source of the data to the year it was collected there are some serious gaps in information.
Understanding whether the data was collected from a manager, employer or HR significantly impacts the data and biases that lie within it. As, if it were collected from HR, this would invalidate the data for the HR department as it would have been collected by their superiors. which makes for poor data reliability. Furthermore, regarding the time frame during which the data was collected. Understanding when these results were compiled and whether there are discernible trends over time could offer valuable context for interpreting the findings.

Some data that should have been included could include more specific performances metrics, such as date of departure from the company or even location. This additional data could show potential geographical trends amongst different branches of the company, or similar trends over time. For example, showing 'an increase in sales employees leaving in the London branch over the last 2 years'. This sort of data is actionable and can lead to clear recommendations to help reduce turnover.

2.2 The Visualisations in the Report

The visualisations throughout the report are dubious at best. For example Figure(1) shows the large distribution of data amongst the different departments. Yet the report goes on to view the average salary of all the data, which now knowing the department distribution, holds little relevance. What more, after this point the report never returns to to concept of viewing data based on department, despite knowing the lack of even representation.

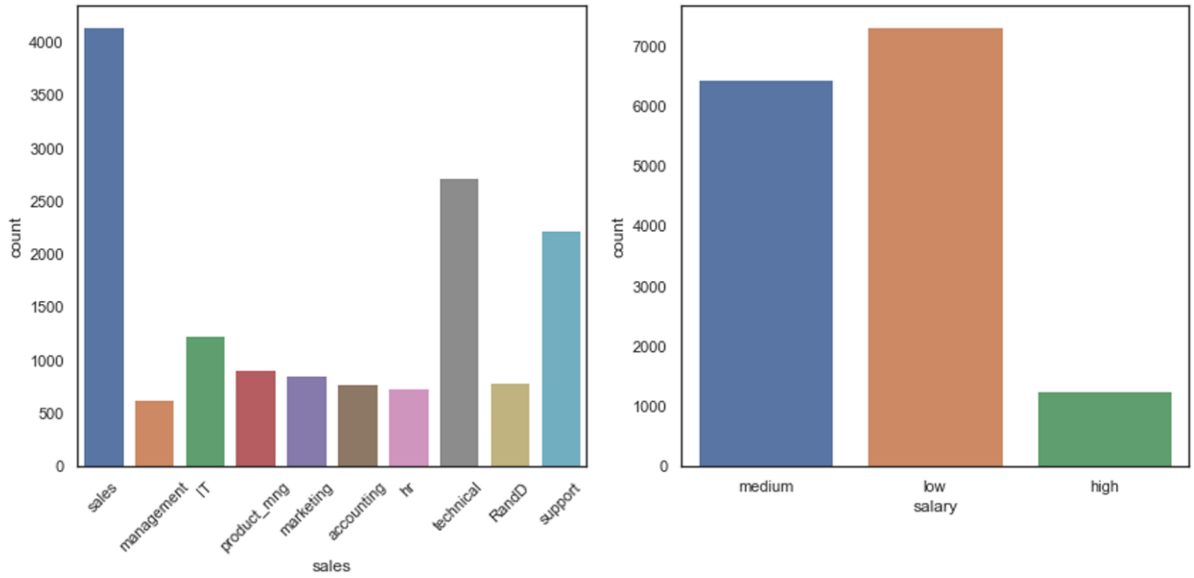

Figure 1: The First Visualisation from the Report

This represents how poor the visualisations are at conveying a meaningful narrative, fundamentally the report falls at the first hurdle in this sense.

2.3 Improvements to the Visualisations

To improve the effectiveness of the visualisations the ‘Salary’ data should have been further refined. Searching for inter-departmental trends would lead to accurate data analysis that can lead to actionable, effective suggestions. For example ‘Sales’ is the largest department, and ‘Management’ is the smallest, so one could assume there would be a discrepancy between the two. Hence why below I have done precisely this and will continue in the ‘Improved Analysis’ Section.

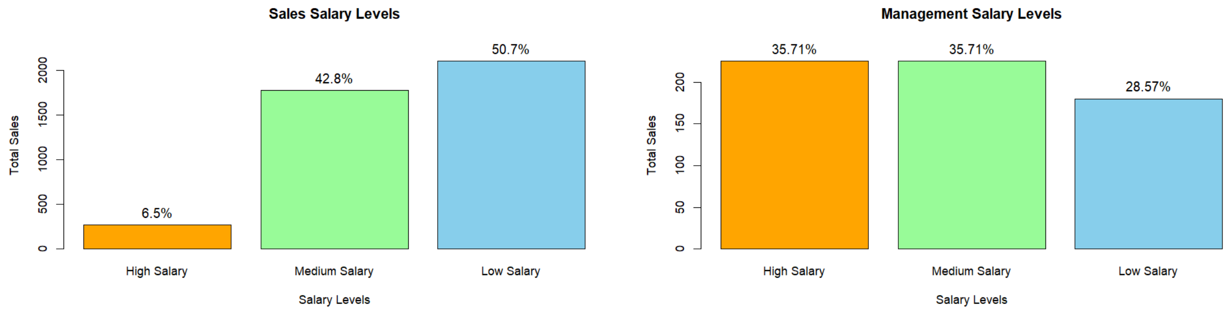

Figure 2: Sales and Management Salary Breakdown

Figure(2) shows precisely why the original report should have broken down the data by department. It is evident that the ‘Salary’ distributions are very different when comparing ‘Sales’ and ‘Management’.

3 Data Preparation

3.1 Data Corruption

The report adequately addressed the presence of missing values within the dataset, demonstrating a proactive approach to data analysis. However, the report failed to examine other possible forms of data corruption. For example no attempt was made to check for duplicate or incorrectly formatted data entries. Failing to verify this makes the report vulnerable to producing incorrect visualisations and undermining the accuracy of the findings.

Further, the report could have delved into detecting and managing outliers within the dataset. Looking at the Interquartile Range (IQR) and the z score would have helped the report identify the potential anomalous data. Additionally, given the lack of numeration in the 'Salary' and 'Sales' columns, the data the should have been reshaped to have new 'Dummy Columns' containing Boolean values (1s and 0s).

3.2 Trust in the Data

As to weather we can trust the data, there are many different questions to be asked:

- What branch/location was the data was collected from?
- What is the source of the information?
- What year is the data from?
- Who collected the data?

This information is crucial to having reliable trustworthy data. For example the data would be much more credible had it been collected by a third party as apposed to an internal figure. Or even if there were dates attached to each entry so as to be able to truly verify if there were any duplicates.

4 Modelling

4.1 Analytical Choices

The choice to use a Random Forest and a Decision Tree were sensible options as they both offer different types of information in varying different formats. However, the main issue arises as to how they were implemented. The report used both predictive models to evaluate the company as a whole, this did find some general patterns, but never evaluates if these are true inter-departmentally. The report should have created new data frames for specific departments to evaluate the differing trends, so as to allow them to make actionable suggestions based on informed data. Failing to do this forces the report to make large generalisations not appropriate for a professional company.

4.2 Reliability of the Models

Looking at the Model evaluation we see that for the Random Forest the report identified a Variable Importance plot was and found that it's mean accuracy was calculated as 98.99% accurate. However this was only evaluated on the 'training data' and no 'test data' was used to verify the fit of the model, hence we cannot know whether the model was good at generalising unseen data.
As for the Decision Tree, it was plotted to view the outcomes and similarly to the Random Forest had it's mean accuracy calculated as 95%. This, like the previous model suggest a good fit however, given no further analysis using unseen data was performed, so we cannot know truly how it's accuracy.

Overfitting occurs when a model learns noise or random fluctuations in the training data instead of the underlying pattern. There were no explicit signs of overfitting mentioned in the analysis. However, the analysis required to capture such outcomes was not performed. For example, the report could have utilised ROC curves to identify the AUC of the Random Forest. Comparing the AUC for the 'training data' and the 'test data' would inform us whether the model was over or underfitting the data. On top of this, inspecting the confusion matrix of the Random Forest to evaluate is it was prone to 'False Positives' or 'False Negatives' could hugely add to the report.

4.3 What do the Models tell us

In the report a Variable Importance plot was introduced to show how much each variable affects the outcome of employee retention. This showed that 'Satisfaction Level' by far was the most important variable to someone leaving the company, with next being 'Time Spent' in the company and the 'Number of Projects'. However, it is important to remember that this is generalised for the whole company and may not be correct for specific departments.
Although not explicitly expressed in the report, from inspection on the decision tree we can deduce that a higher satisfaction level correlates to a higher likelihood of staying and a higher time spent in the company correlates to a higher likelihood of staying. Nevertheless, this is again a generalised estimate. It would be much more reliable and clear if a Logistic Regression Model was introduced to evaluate *how* these variable affect the outcome, as shown in an upcoming section.

4.4 Improved Analysis

N.B - (The visualisations in the following section were produced in R-Studio as improvements and additions to the original report, the code can be found in the appendix.)

When producing these visualisations, I took the original dataset, then randomly separated it into 'test' (30%) and 'training' (70%) dataset. These will be use throughout to evaluate the predictive models.

4.4.1 Salary Breakdown

After identifying the 'Salary' distribution in Figure(2), it is suitable to then break this down to two columns 'Left' and 'Stayed' to further evaluated any underlying trends.

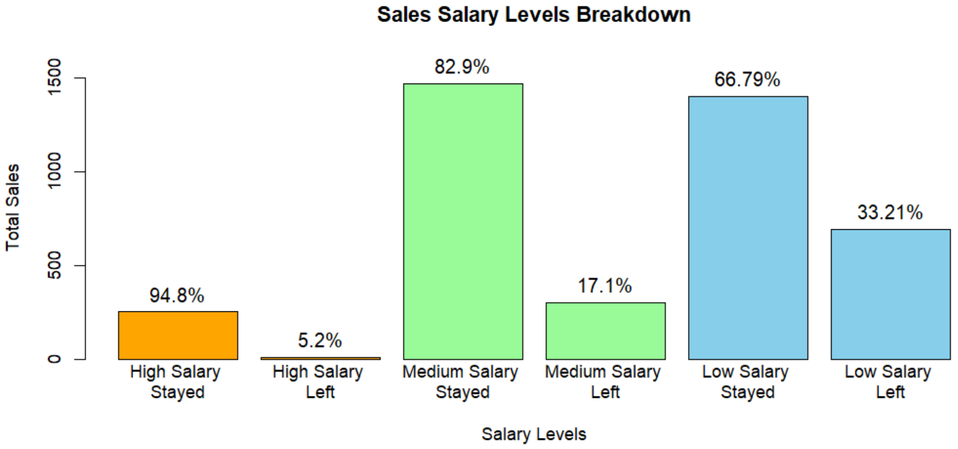

Figure 3: Sales Salary Breakdown

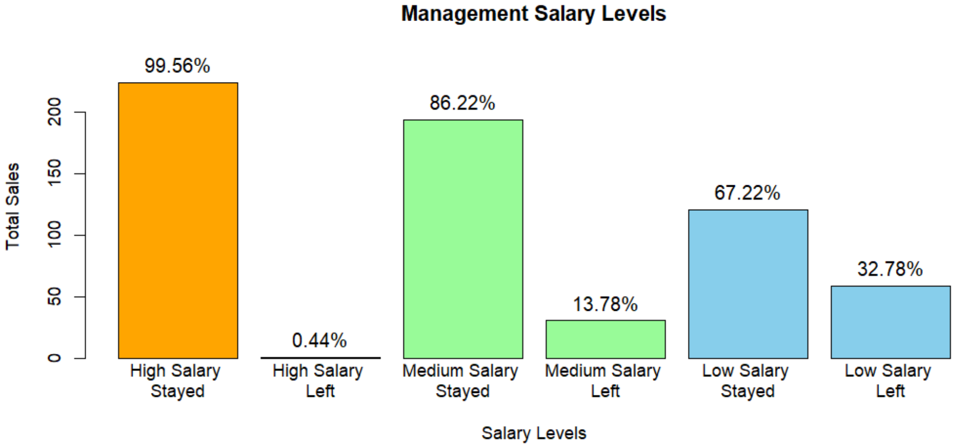

Figure 4: Management Salary Breakdown

Figure(4) shows us a more detailed breakdown of whether employees decided to stay or leave, based on their pay. We see that 'High Salary' Managers were more likely to stay than 'High Salary' Sales workers, and that 'Low Salary' workers in both departments are roughly equally likely to stay. 'High Salary' Managers are likely more probable to stay due to the nature of the position as it often require years of promotion, hence we could assume that the 0.44% of them who are leaving is likely due to retirement.

4.4.2 Random Forest

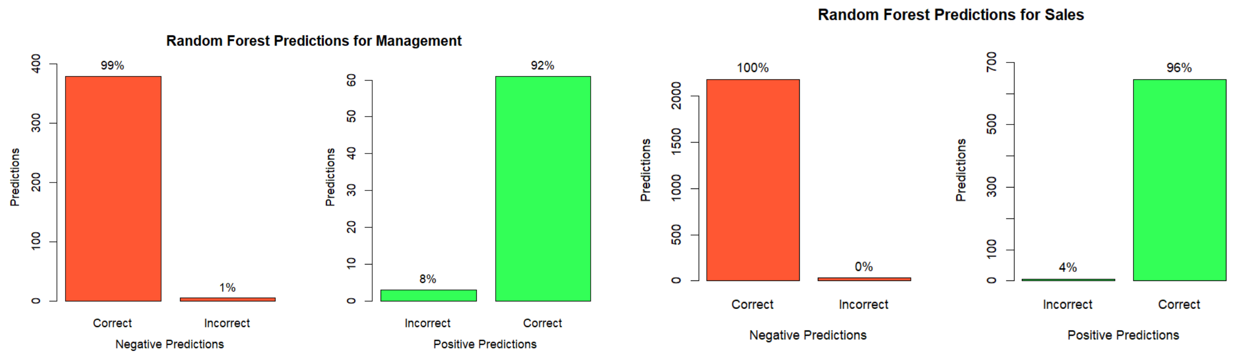

Figure 5: Random Forests for both Departments

Looking at the Random Forest breakdown in Figure(5)we see that for both 'Sales' and 'Management' it favours 'False Positives'. Although, this becomes twice as likely for the 'Management' data compared to the 'Sales' data. This allow us to understand potential inconsistencies with the predictions and then value providing more data from 'Sales' to the Random Forest to improve its accuracy.

4.4.3 ROC of the Random Forest

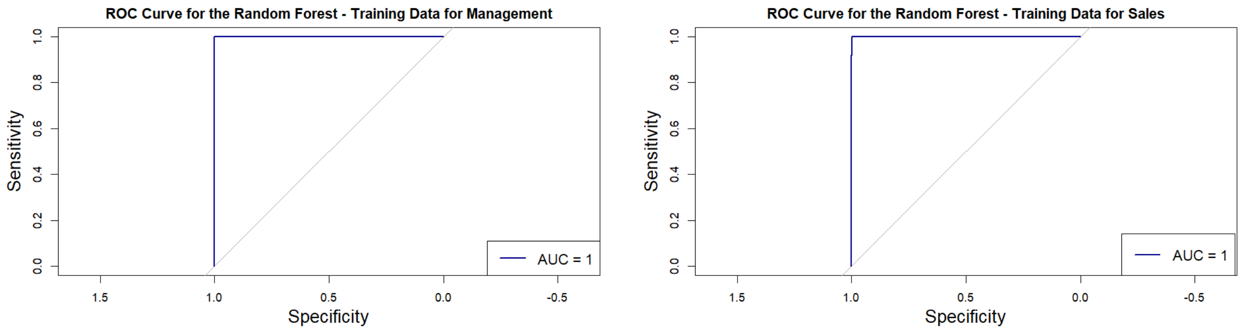

Figure 6: ROC Plot for training data

Looking at the ROC curves, we can identify that the Random Forest is an extremely good fit for the training data for both 'Sales' and 'Management', having achieved an AUC = 1 for both.

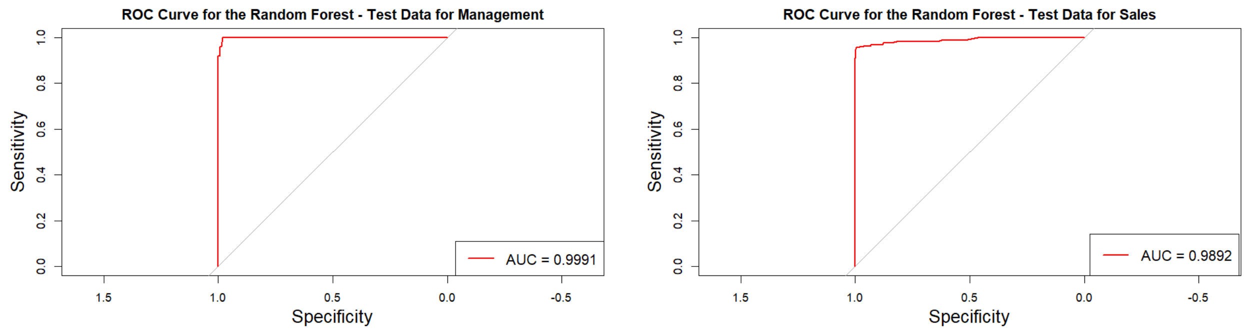

Figure 7: ROC Plot for test data

Then, comparing Figure(6) to the ROC of the 'test data', we see the AUC has slightly dropped to 0.9991 and 0.9892. However, these are so close to 1 we can confidently say that the Random Forest shows no signs of over-fitting or under-fitting and would be suitable to use on any new unseen data.

4.4.4 Variable Importance of Random Forest

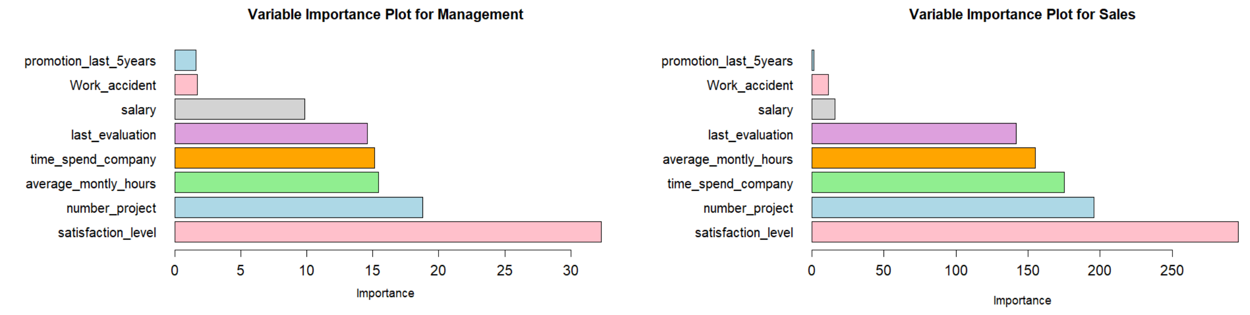

Figure 8: Variable Importance Plots

Looking at this visualisation we can see that the order of variable importance doesn't change inter-departmentally. However, the Importance of them does. For example looking at 'Salary' we identify that it is of much more importance for employees from 'Management' than from 'Sales'. This is likely due to the nature of the pay scales of both departments. But, ultimately this still doesn't tell us *how* they impact employee retention.

4.4.5 Variable Effects from the Logistic Regression Model

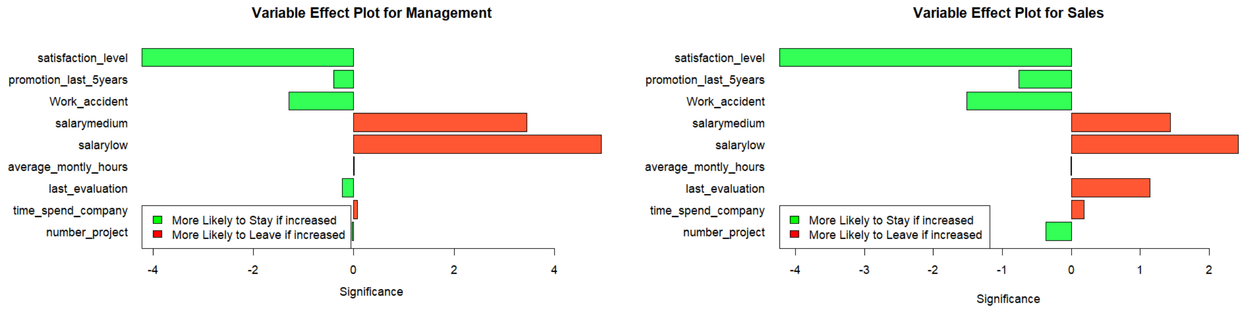

Figure 9: Variable Importance Effect Plot

To understand, *how* the different variables impact if an employee leaves, we have to perform a Logistic Regression on the data. This now show significance on a Positive and Negative scale, which allows us to understand the implications of different variables. As above the red bars represent variables which, if increased would increase the chance of an employee leaving. Along with green bars representing variables which if increased would increase the chance of an employee staying. The biggest discrepancy being 'last_evaluation' which shows as red for 'Sales' and green for 'Management'.

4.4.6 ROC of the Logistic Regression Model

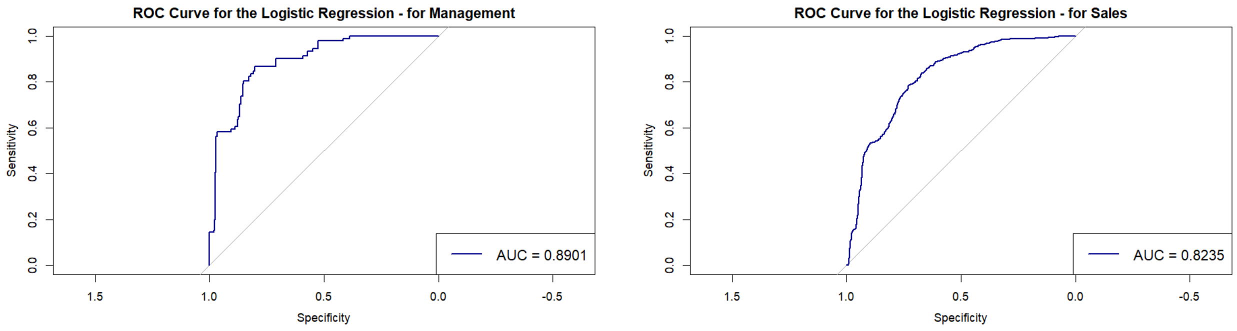

Figure 10: ROC Plot for the Logistic Regression

To check that the Logistic Regression in Figure(9) is accurate, it is good practice to verify this using the ROC curve. This produced AUC of 0.8901 and 0.8235 for the training data and test data respectively. Both these scores suggest the model is a good fit for the data and would be suitable on new, unseen data. We can confidently say that it shows no signs of over or underfitting.

5 Evaluation

5.1 The Process, Data and Visualisations

The analytical process was fairly poor and did not endeavour to discover underlying trends within the company, as it did not even simply separate the data amongst departments. Further, the data was an inadequate choice, as contains no time related data, meaning no trends over time could be assessed. Not to mention that the data contained many duplicates, likely from the 'Management' column as this is not a department, but simply a section from every other department. Finally, the data contained simple spelling errors, for example 'average_montly_hours' should have been 'Average_Monthly_Hours'.

Next the analytical choices and visualisations. They did not convey any form of narrative towards identifying or solving a problem. The visualisations did identify patterns in the wider data, however, this offered no help in terms of suggestions for the company to take action.
So to sum up, no this project should not move forward to deployment as it contains no clear actionable suggestions, and if an employer were to act on some of the visualisations, it is likely that many people would be harmed and the company would not succeed as a result.

5.2 Changes

To continue this analysis into the future I have broken down the variable importance from the Random Forest into different departments. This allows us to identify more specific trends and find out what variables the different departments value more. For example 'Management' value their salary quite highly, whereas for 'Sales' it is barely a significant factor. This is likely due to 'Sales' employees working off of commission, hence making their base salary less important.

Further to identify *how* these factor affect the probability of an employee leaving the company, I have performed a Logistic Regression Model as shown in Figure(9). Doing this for separate departments is vital, as we can understand which areas to target specifically, and then make informed decisions.

5.3 Additions

Some additions to the report could include, some analysis on the Random Forest to verify it's accuracy. This could include ROC plots as I used in Figure(6) or perhaps PRC. The addition of these would make the results more reliable, adding to the robustness of the report allowing for trusted recommendations.

6 Deployment

6.1 Actions

From the Report:
I would recommend that the company aims to improve the general satisfaction of all the employees. However, as the report has no actionable suggestions on how to implement this, I could make no recommendations to elaborate on this.

From my Own Analysis:
I would recommend that the company aims to increase the rate at which it promotes its 'Sales' employees so as to improve their retention. This is because they value promotions significantly. This could be due to promotions being rarer in 'Sales' or some other underlying factors. To understand this further, deeper analysis would nee to be carried out. It is feasible to increase the promotion rate of 'Sales' employees as there would likely be other departments that do not uphold promotions as highly. Hence the resources could be allocated differently.

Further the company could aim to slightly reduce the total time employees spend at the company for both 'Management' and 'Sales' employees as this increases their likelihood of staying as shown in Figure(9) . The task of reducing 'Sales' employee's time spent at the company is fairly straight forward. Although, it is understandable that employers may be sceptical about this reduction and it's effect on profits/turnover. This model does not explicitly tell us if the reduction in work hours would directly translate to increased profits, but it does inform us that it would lead to a higher employee retention.

A final suggestion could be for the company to aim to reduce the rate at which it gives 'Sales' employees raises, as it has a lesser effect on their retention. Although, we have seen that promotions have a strong impact, from Figure(9) we see that without the title, a pay raise for 'Sales' employees does little for their retention. This money could be distributed to other departments to improve their retention depending on how they value raises. However, this needs to be implemented carefully so as not to harm the 'Sales' employees. Obviously, reducing the raises for 'Sales' employees is not a difficult task for employers. But, the difficulty comes in moderating this, so as not to disproportionately target them and create further pay imbalances.

Overall, I would suggest that the company invests in collecting more detailed data. More particularly some data matching to the date at which the current data was collected. Then also I would suggest having more detailed data, such as when an employee last asked for a raise and if it was successful, as well as whether they received compensation for their work place accident. This last one is particularly important as more deeper analysis is needed to look into 'work_accident' as in Figure(9) some employees are more likely to stay given they had a work accident. I believe this is a good area for the company to investigate further. Further, it would be useful to have the employees previous evaluations, this way we could identify if 'Hard Working' employees are being adequately compensated. I believe this data analysis would be the most impactful on driving company profits.

6.2 Deploying an Ongoing Process

To effectively deploy this strategy, I would suggest that the company implements the previous recommendations, including driving my analysis further by examining each department individually and assessing the variables they prioritise the most through additional Variable Importance plots. Furthermore, enhancing the analysis by incorporating more data to track ongoing trends over time would be hugely beneficial. In addition to these steps, fostering a culture of data-driven decision-making across all levels of the organisation is essential. This involves providing training and resources to employees to enable them to understand and utilise the data effectively in their day-to-day operations. Regular communication and feedback loops should be established to ensure that insights from data analysis are effectively integrated into decision-making processes. Ultimately, the company should consistently evaluate their actions and their impact to ensure continuous improvement in company turnover.

6.3 Different types of Deployment

A **Descriptive Deployment** requires identifying important metrics and trends in the data. For example recognising the importance of ‘satisfaction_level’ in the data and understanding how it affects employee retention.

A **Predictive Deployment** requires a consistent approach to the model, ensuring it is regularly updated with new information to improve the accuracy for new predictions. In the case if the current data, this could look like providing the Random Forest with updated information on existing employees, such as updating their salary status or even new information such as their ‘Last_Request_for_Raise’ or ‘Date_of_Last_Evaluation’.

A **Prescriptive Deployment** requires suggesting feasible recommendations, to help improve the valued outcomes. Such as the ones previously suggested, like reducing ‘Sales’ employees work hours to improve employee retention.

To implement an ongoing prescriptive dashboard system I would need to develop systems to automatically input the newly incoming data into the Random Forest after being cleaned. Alongside this, new algorithms to verify the accuracy and generate feasible recommendations would be put in place. These would help to create a clear and simple system for employers and partners to interact with, and then make quality informed decisions.

6.4 Consequences

From this analysis, if my suggestions were implemented correctly I would expect employee turnover to decrease dramatically in all departments, as well as employee satisfaction to increase or at least remain constant. In turn this would drive up company profits, improving the quality of life for the employees and the profit margins for the employers.

7 Code Appendix

```
rm(list = ls())
library(tidyverse)
library(ggplot2)
library(outliers)
library(fastDummies)
library(factoextra)
library(FactoMineR)
library(survival)
library(survminer)
library(randomForest)
library(fmsb)
library(rpart)
library(rpart.plot)


# Import the data
hr <- read.csv('kaggle_hr_analytics.csv')

#summary(hr)
glimpse(hr)

# dummy columns
hr <- dummy_cols(hr, c('salary', 'sales'))

# rename columns
hr <- hr %>%
  rename(department =  sales) %>%
  mutate(department = fct_infreq(department))

# add an ID column
hr$ID <- as.character(1:nrow(hr))
glimpse(hr)





##sales salary breakdown

# select the data
hr_sales_low =  sum(hr$sales_sales[hr$salary_low == 1])
hr_sales_med =  sum(hr$sales_sales[hr$salary_medium == 1])
hr_sales_high =  sum(hr$sales_sales[hr$salary_high == 1])


# Create a vector containing the sums
sales_sums <- c(hr_sales_high, hr_sales_med, hr_sales_low)
total_sales <- sum(sales_sums)
# Calculate percentages
percentages <- round((sales_sums / total_sales) * 100, 2)
```

```
# Assigning colors
bar_colors <- c("orange", "palegreen","skyblue")

# Create bar plot with specified colors
barplot(sales_sums,
        beside = TRUE,
        names.arg = c( "High Salary", "Medium Salary","Low Salary"),
        main = "Sales Salary Levels",
        xlab = "Salary Levels",
        ylab = "Total Sales",
        col = bar_colors,
        ylim = c(0, max(sales_sums) * 1.1))

# Add percentages on the bars
text(x = barplot(sales_sums, beside = TRUE, plot = FALSE),
     y = sales_sums + 0.5,
     labels = paste0(percentages, "%"),
     pos = 3,
     cex = 1.1,
     col = "black")








## managment salary breakdown

# select new data
hr_managment_low =  sum(hr$sales_management[hr$salary_low == 1])
hr_managment_med =  sum(hr$sales_management[hr$salary_medium == 1])
hr_managment_high =  sum(hr$sales_management[hr$salary_high == 1])


# Create a vector containing the sums
hr_sums <- c(hr_managment_high,  hr_managment_med, hr_managment_low)

# Calculate total hr
total_hr <- sum(hr_sums)

# Calculate percentages
percentages <- round((hr_sums / total_hr) * 100, 2)

# Assigning colors
bar_colors <- c("#FF5733", "#33FF57", "#3366FF")
bar_colors <- c("orange", "palegreen","skyblue")

# Create bar plot with specified colors
barplot(hr_sums,
        beside = TRUE,
        names.arg = c( "High Salary", "Medium Salary","Low Salary"),
        main = "Management Salary Levels",
        xlab = "Salary Levels",
        ylab = "Total Sales",
        col = bar_colors,
        ylim = c(0, max(hr_sums) * 1.1))


# Add percentages on the bars
text(x = barplot(hr_sums, beside = TRUE, plot = FALSE),
     y = hr_sums + 0.5,
     labels = paste0(percentages, "%"),
     pos = 3,
     cex = 1.1,
     col = "black")






## sales salary left breakdown
hr_sales_low_stay =  sum(hr$sales_sales[hr$salary_low == 1 & hr$left == 0])
hr_sales_low_left =  sum(hr$sales_sales[hr$salary_low == 1 & hr$left == 1])

hr_sales_med_stay =  sum(hr$sales_sales[hr$salary_medium == 1 & hr$left == 0])
hr_sales_med_left =  sum(hr$sales_sales[hr$salary_medium == 1 & hr$left == 1])

hr_sales_high_stay =  sum(hr$sales_sales[hr$salary_high == 1 & hr$left == 0])
hr_sales_high_left =  sum(hr$sales_sales[hr$salary_high == 1 & hr$left == 1])


# Create a vector containing the sums
sales_sums <- c(  hr_sales_high_stay,hr_sales_high_left,
                  hr_sales_med_stay, hr_sales_med_left,
                  hr_sales_low_stay,hr_sales_low_left)


total_sales <- sum(sales_sums)



sum1 = sum(sales_sums[1] , sales_sums[2])
sum2 = sum(sales_sums[3] , sales_sums[4])
sum3 = sum(sales_sums[5] , sales_sums[6])

percentages = c(round((sales_sums[1] / sum1) * 100, 2), round((sales_sums[2] / sum1) *
    100, 2),
                round((sales_sums[3] / sum2) * 100, 2), round((sales_sums[4] / sum2) *
    100, 2),
                round((sales_sums[5] / sum3) * 100, 2), round((sales_sums[6] / sum3) *
    100, 2))




# Assigning colors
bar_colors <- c("orange", "orange", "palegreen","palegreen", "skyblue", "skyblue")

# Create bar plot with specified colors
barplot(sales_sums,
        beside = TRUE,
        names.arg = c("High Salary \nStayed", "High Salary \nLeft",
                      "Medium Salary \nStayed", "Medium Salary \nLeft",
                      "Low Salary \nStayed", "Low Salary \nLeft"),

        main = "Sales Salary Levels Breakdown",
        xlab = "Salary Levels",
        ylab = "Total Sales",
        col = bar_colors,
        ylim = c(0, max(sales_sums) * 1.1))


# Add percentages on the bars
text(x = barplot(sales_sums, beside = TRUE, plot = FALSE),
     y = sales_sums,
     labels = paste0(percentages, "%"),
     pos = 3,
     cex = 1.1,
     col = "black")



## Management salary left breakdown

hr_sales_low_stay =  sum(hr$sales_management[hr$salary_low == 1 & hr$left == 0])
hr_sales_low_left =  sum(hr$sales_management[hr$salary_low == 1 & hr$left == 1])

hr_sales_med_stay =  sum(hr$sales_management[hr$salary_medium == 1 & hr$left == 0])
hr_sales_med_left =  sum(hr$sales_management[hr$salary_medium == 1 & hr$left == 1])

hr_sales_high_stay =  sum(hr$sales_management[hr$salary_high == 1 & hr$left == 0])
hr_sales_high_left =  sum(hr$sales_management[hr$salary_high == 1 & hr$left == 1])



# Create a vector containing the sums
sales_sums <- c(  hr_sales_high_stay,hr_sales_high_left,
                  hr_sales_med_stay, hr_sales_med_left,
                  hr_sales_low_stay,hr_sales_low_left)


total_sales <- sum(sales_sums)
# Calculate percentages
percentages <- round((sales_sums / total_sales) * 100, 2)


sum1 = sum(sales_sums[1] , sales_sums[2])
sum2 = sum(sales_sums[3] , sales_sums[4])
sum3 = sum(sales_sums[5] , sales_sums[6])

percentages = c(round((sales_sums[1] / sum1) * 100, 2), round((sales_sums[2] / sum1) * 100, 2),
                round((sales_sums[3] / sum2) * 100, 2), round((sales_sums[4] / sum2) * 100, 2),
                round((sales_sums[5] / sum3) * 100, 2), round((sales_sums[6] / sum3) * 100, 2))
```

```


# Assigning colors
bar_colors <- c("orange", "orange", "palegreen","palegreen", "skyblue", "skyblue")

# Create bar plot with specified colors
barplot(sales_sums,
        beside = TRUE,
        names.arg = c("High Salary \nStayed", "High Salary \nLeft",
                      "Medium Salary \nStayed", "Medium Salary \nLeft",
                      "Low Salary \nStayed", "Low Salary \nLeft"),

        main = "Management Salary Levels",
        xlab = "Salary Levels",
        ylab = "Total Sales",
        col = bar_colors,
        ylim = c(0, max(sales_sums) * 1.1))


# Add percentages on the bars
text(x = barplot(sales_sums, beside = TRUE, plot = FALSE),
     y = sales_sums,
     labels = paste0(percentages, "%"),
     pos = 3,
     cex = 1.1,
     col = "black")









## Random Forest
library(dplyr)
set.seed(42)

# CHANGE HERE!!!
variable = "Management"
hr_sales = hr[hr$sales_management == 1, ]




hr_sales$left <- factor(hr_sales$left)



#use 70% of dataset as training set and 30% as test set
sample <- sample(c(TRUE, FALSE), nrow(hr_sales), replace=TRUE, prob=c(0.7,0.3))
hr_train  <- hr_sales[sample, ]
hr_test   <- hr_sales[!sample, ]


```

```


set.seed(42)

# Fit the random forest model
rf <- randomForest(left ~ number_project +
                     time_spend_company +
                     last_evaluation +
                     average_montly_hours +
                     salary +
                     Work_accident +
                     promotion_last_5years +
                     satisfaction_level
                   ,

                   data = hr_train,
                   type = "classification")



# print confusion matrix
print(rf$confusion)


## look at the rf



# Extract confusion matrix data with percentages
conf_data <- rf$confusion[, 1:3]

# Separate negative and positive predictions
conf_neg <- conf_data[, 1]
conf_pos <- conf_data[, 2]

# Set up the plotting area
par(mfrow = c(1, 2))

# Create bar plot for negative predictions
neg_bar <- barplot(conf_neg,
                   xlab = "Negative Predictions",
                   ylab = "Predictions",
                   col = "#FF5733",
                   names.arg = c("Correct", "Incorrect"),
                   ylim = c(0, max(conf_neg) * 1.1))


# Add percentage labels
text(x = neg_bar[2],
     y = conf_neg[2],
     label = paste0(round(conf_data[1, 3] * 100), "%"),
     pos = 3,
     col = "black")

text(x = neg_bar[1],
     y = conf_neg[1],
     label = paste0(round((1 - conf_data[1, 3]) * 100), "%"),
     pos = 3,
     col = "black")



# Create bar plot for positive predictions
pos_bar <- barplot(conf_pos,
                   xlab = "Positive Predictions",
                   ylab = "Predictions",
                   col = "#33FF57",
                   names.arg = c("Incorrect", "Correct"),
                   ylim = c(0, max(conf_pos) * 1.1))


# Add percentage labels
text(x = pos_bar[2],
     y = conf_pos[2],
     label = paste0(round((1 - conf_data[2, 3]) * 100), "%"),
     pos = 3,
     col = "black")

text(x = pos_bar[1],
     y = conf_pos[1],
     label = paste0(round(conf_data[2, 3] * 100), "%"),
     pos = 3,
     col = "black")


# Add a title
title(main = paste0("Random Forest Predictions for ", variable),
      outer = TRUE,
      line = -1.5)







## Look at the accuracy


library(pROC)


# plot size
par(mfrow = c(1, 1))

# Predictions for training data
pred_train <- predict(rf,
                      newdata = hr_train,
                      type = "prob")[, 2]

# ROC curve for training data
mailing_train_roc <- roc(hr_train$left, pred_train)
```

```r

# Plot ROC curve for training data
plot(mailing_train_roc,
     main = paste0("ROC Curve for the Random Forest - Training Data for ",  variable),
     col = "darkblue",
     lwd = 2,
     cex.lab = 1.5)

legend("bottomright",
       legend = paste("AUC =", round(auc(mailing_train_roc), 4)),
       col = "darkblue",
       lwd = 2,
       cex = 1.2)





# plot size
par(mfrow = c(1, 1))

# Predictions for test data
pred_test <- predict(rf, newdata = hr_test, type = "prob")[, 2]

# ROC curve for test data
mailing_test_roc <- roc(hr_test$left, pred_test)


# Plot ROC curve for test data
plot(mailing_test_roc,
     main = paste0("ROC Curve for the Random Forest - Test Data for ",  variable),
     col = "red",
     lwd = 2,
     cex.lab = 1.5)

# display a legend
legend("bottomright",
       legend = paste("AUC =", round(auc(mailing_test_roc), 4)),
       col = "red",
       lwd = 2,
       cex = 1.2)







## Variable Importance

# plot dimensions
par(mar = c(5, 12, 4, 2), mfrow = c(1,1))

# Get variable importance data from the random forest model
var_imp <- data.frame(Variables = row.names(rf$importance), Importance = rf$importance[,1])
```

```r

# put the data in decreasing order
var_imp <- var_imp[order(var_imp$Importance, decreasing = TRUE), ]


# define the colours be used
col <- c("pink",
         "lightblue",
         "lightgreen",
         "orange",
         "plum",
         "lightgrey")



# Create a bar plot with sorted values
barplot(var_imp$Importance,
        names.arg = var_imp$Variables,
        xlab = "Importance",
        main = paste0("Variable Importance Plot for ", variable),
        col = col,
        cex.names = 1.1,
        cex.axis = 1.2,
        las = 1,
        horiz = TRUE)  # <- this line makes the bars horizontal








## Logistic Regression

# Fit logistic regression model
logit_model <- glm(left ~ number_project +
                     time_spend_company +
                     last_evaluation +
                     average_montly_hours +
                     salary +
                     Work_accident +
                     promotion_last_5years +
                     satisfaction_level,

                   data = hr_sales,
                   family = binomial)

# Summary of the model
log_sum = summary(logit_model)

# Interpret coefficients and their significance
log_sum$coefficients



```

```
## plot it

# Extract coefficients from the summary
coefficients <- log_sum$coefficients[2:10, 0:1]



# Plot
par(mar = c(5, 10, 4, 2), mfrow = c(1,1))  # Adjust margin to accommodate longer variable names

barplot(coefficients,
        horiz = TRUE,
        col = ifelse(coefficients < 0, "#33FF57", "#FF5733"),
        las = 1,
        xlab = "Significance",
        main = paste0("Variable Effect Plot for ", variable),
)

legend("bottomleft",
       legend = c("More Likely to Stay if increased", "More Likely to Leave if increased"),
       fill = c("green", "red"))




## check the accuracy:


library(pROC)

# Compute predicted probabilities
predicted_probs <- predict(logit_model, type = "response")

# Compute ROC curve
roc_curve <- roc(hr_sales$left, predicted_probs)

# Plot ROC curve
plot(roc_curve,
     col = "darkblue",
     main = paste0("ROC Curve for the Logistic Regression - for ",  variable))


legend("bottomright",
       legend = paste("AUC =", round(auc(roc_curve), 4)),
       col = "darkblue",
       lwd = 2,
       cex = 1.2)
```