HR ANALYTICS

Exploratory & Predictive Analytics

Submitted By: Submitted To:

Kamal Preetham Ravipati (kxr220022) Prof. Thomas Lavastida

Shivali Mavi (sxm220124)

Vishesh Walia (vxw220000)

# Table of Contents

# 

[Table of Contents 2](#_Toc134650209)

[Abstract 3](#_Toc134650210)

[Introduction 3](#_Toc134650211)

[Dataset Description 4](#_Toc134650212)

[Data Preprocessing 5](#_Toc134650213)

[Exploratory Data Analysis 7](#_Toc134650214)

[Models 9](#_Toc134650215)

[Logistic Regression: 10](#_Toc134650216)

[Gaussian Naive Bayes 12](#_Toc134650217)

[Result 13](#_Toc134650218)

[References 14](#_Toc134650219)

# 

# Abstract

The goal of this project is to develop a predictive model to determine which factors are contributing to employee attrition in a company. The team will perform exploratory data analysis to identify key observations and use regression models such as logistic regression or random forest to predict employee attrition. The project will showcase the team's skillset in data wrangling, data visualization, and machine learning implementation. The outcome of this project could help organizations to identify attrition risks and retain key employees.

# Introduction

The aim of this project is to predict employee attrition using regression analysis. We are motivated to explore this real-world problem as it allows us to apply our classroom learning of machine learning and classification modeling techniques to a practical scenario. The dataset we used contains 14,999 records with 10 features, including both numerical and categorical variables. We conducted exploratory data analysis, data wrangling, and data visualization to understand the impact of work accidents, satisfaction levels, and salary on the attrition rate of employees. We also performed hypothesis testing techniques such as F-regression and chi-square testing to identify key observations. We used logistic regression and other classification algorithms such as random forest to solve the binary classification problem.

One of the most significant takeaways from this project is that the satisfaction level of employees plays a vital role in predicting employee attrition. Another takeaway is that the average monthly working hours and the number of projects are other essential factors that contribute to employee attrition.

The exploratory analysis answers the following questions:

* How many records of people leaving the company exist in the dataset?
* What is the percentage of churn by salary bucket?
* How many people, who had work accidents, actually left the company?
* How work accidents have impacted the satisfaction level of the employees?
* How satisfaction levels influence whether to stay or leave the company?
* Average satisfaction levels for people who leave and stay back in the company
* Does lower satisfaction levels lead to people leaving the company?
* How last evaluation scores influencing whether to stay or leave the company?
* How time spent in the company influences attrition?

# Dataset Description

This project will use an actual dataset from Kaggle that contains information about a company's employees in a csv file format. The goal of the project is to use this dataset to build a predictive model that can determine whether or not an employee is likely to leave the company based on the given features. Some of the characteristics of the features are as follows:

1. Satisfaction level (column: satisfaction\_level): This feature is a continuous numerical variable with a range of values from 0 to 1. It measures employee’s level of job satisfaction, with 1 being highly satisfied and 0 being highly dissatisfied.

2. Last Evaluation (column: last\_evaluation): This feature measures the employee's most recent performance evaluation using a numerical variable with values between 0 and 1, with 1 being the highest score and 0 being the lowest.

3. Number of Projects (column: number\_project): This represents the number of projects the employee has worked on during their time in the company. It is a numerical variable.

4. Average Monthly Hours (column: average\_montly\_hours): This measures the typical number of hours the employee works each month.

5. Time Spent in Company (column: time\_spend\_company): The number of years the employee has spent in the company represented by a numerical variable.

6. Work Accident (column: Work\_accident): This is a binary variable which indicates whether or not employee has experienced an accident at work.

7. Left (column: left): This is our target variable which we want to predict. This is a binary variable that represents whether or not the employee has left the company. The value 1 indicates the employee has left and 0 indicates the employee is still a part of the company.

8. Promotion Last 5 years (column: promotion\_last\_5years): This feature is also a binary variable that indicates whether or not the employee has received the promotion in the last 5 years.

9. Department (column: department): This feature is a categorical variable that identifies the employee’s department of employment. It has 10 distinct values for this feature, corresponding to 10 different departments

10. Salary (column: salary): This last feature is also a categorical variable that demonstrates the salary level of the employee as low, medium or high.

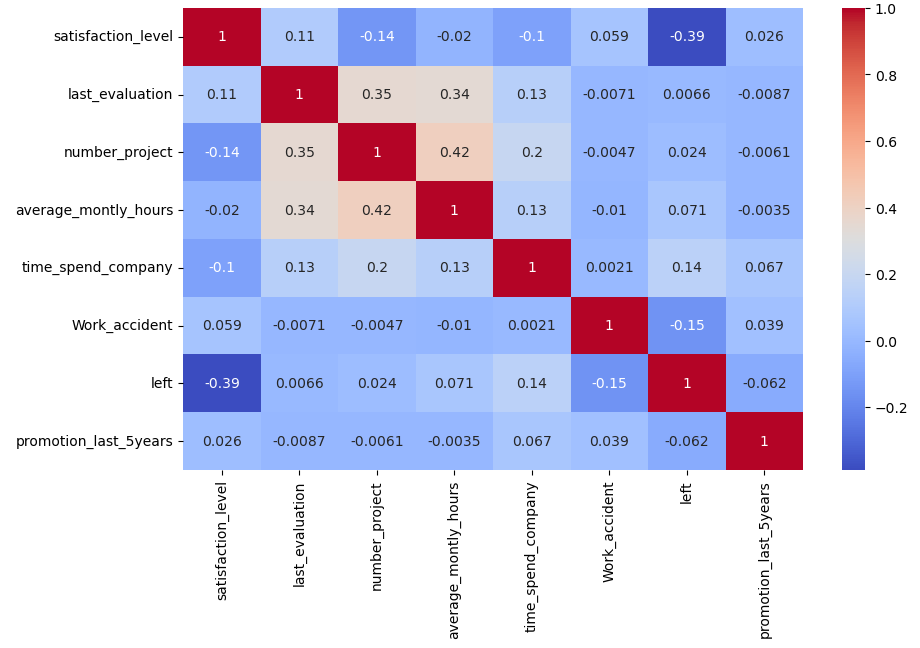
The dataset consists of both categorical and numerical variables. The target variable is binary variable, with a value of 1 representing an employee who has left the company and 0 representing an employee who has stayed. It has 14,999 records with 10 features which is suitable for developing a binary classification model to predict employee attrition.

# Data Preprocessing

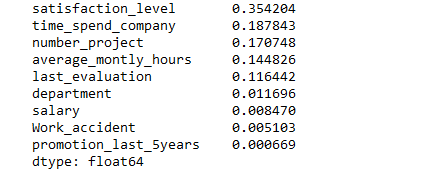
In order to prepare the data for modeling, several steps were taken. Firstly, the null values were treated appropriately. Next, categorical variables were encoded and converted into dummy variables to facilitate the logistic regression model's understanding of the variables. Finally, the dataset was split into training and testing datasets with an 80-20 split, respectively.

These steps were taken to ensure that the data was ready for the modeling process and to ensure that the model could accurately interpret and analyze the data. The process of data preparation is essential in any analysis or modeling task, as it sets the foundation for subsequent steps in the analytical process. The use of appropriate statistical tests to filter out insignificant columns and split the dataset into training and testing subsets are important steps in ensuring the accuracy and reliability of the model's outputs.

As part of the preprocessing process, we also engineer our features to figure out those features that are important for us to determine and predict the attrition rates with the data we have. We start off with drawing a correlation matrix among all the features.



Looking at the matrix, we observe a fairly strong correlation between ‘satisfaction\_level’ and ‘left’. To further get a list of features in order of their importance, we run the data through a Random Forest Classifier and observe the results. These “important features” were later used to fine tune our models and improve their performance.

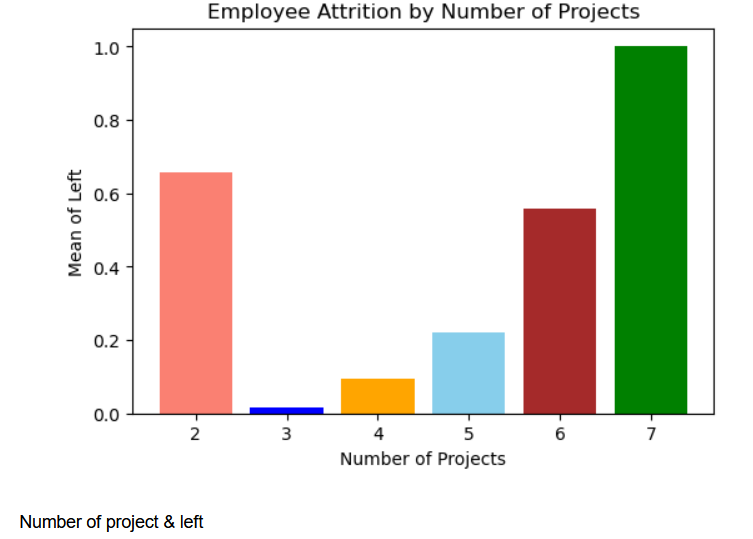


These are the features other than our target variable in the descending order of their importance towards our target variable.

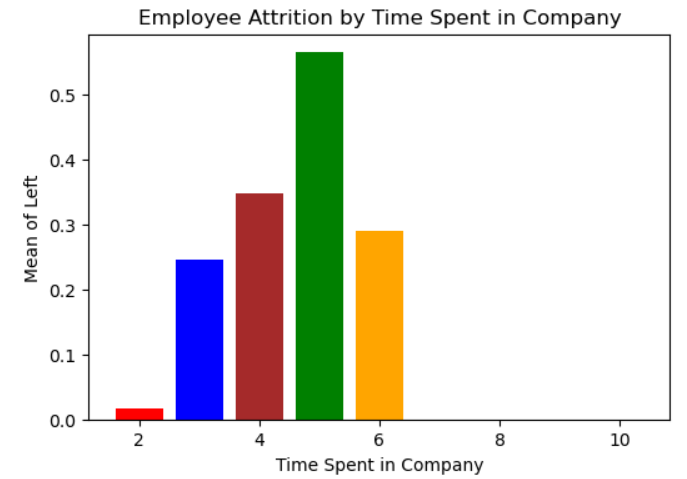
# Exploratory Data Analysis

Visualizations involve plots that represent various factors which are the causes of attrition risks. These will be helpful for us to easily identify the key factors that are leading to churn, so that organizations can design the strategies to avoid the risks. Here we can see some of the visualizations that will help us identify the factors that contribute to employee churn.

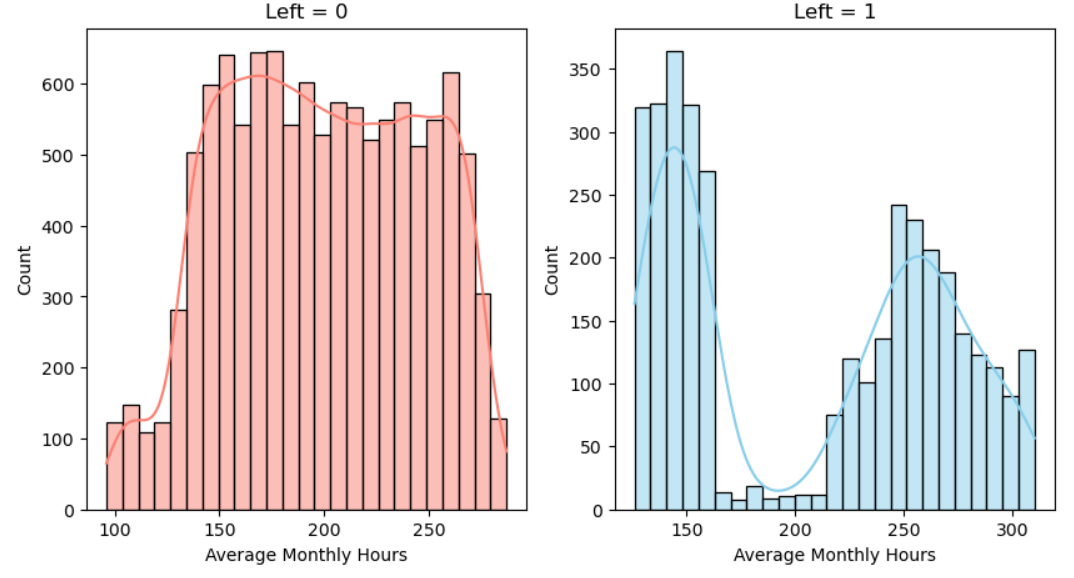




Overworked employees are more likely to leave. Interesting observation is that not enough work is also causing employees to look for other opportunities.



Employees completing their 5-year term at the company are likely to leave.



Attrition rates are very low for employees working around 200 hours a month, overworked employees have higher attrition rates and employees attending less than 150 hours a month have the highest attrition rates.

We can observe from the above visualizations that our target variable ‘left’ is influenced by the variable ‘salary’, ‘number of projects’, ‘time spent in company’, ‘average monthly hours’.

# Models

The employee dataset can be used to develop predictive models for identifying employees who are most likely to leave the organisation based on various features. The initial stage is to preprocess and clean the data which involves handling missing values, dealing with outliers, encoding categorical variables, and scaling numerical variables. After the data has been preprocessed, it is split between training and testing sets. The training set is used to train the model and the testing set is used to evaluate the model’s performance.

Following that, various machine learning methodologies can be employed to build the predictive models. Some of the models included are logistic regression, random forest, KNN models, support vector machines, and more. These algorithms aim to analyze the correlations between independent variables and dependent variables. After training, the model's performance is measured using measures including accuracy, precision, recall, F1-score, and ROC-AUC. The model can then be fine-tuned by adjusting hyperparameters or feature selection.

Once the model is selected, it can be used to make predict which employees are most likely to leave the company based on their input features. Below listed are some of the models we used in our project.

## Logistic Regression:

Logistic regression is a statistical method that estimates the probability of an output variable. In the case of the HR employee dataset, we determine whether or not an employee has left the organization based on input data, such as satisfaction level, last evaluation, number of projects and so on.

The algorithm works by fitting a logistic function to the training data to generate a decision boundary that differentiates between employees who left and those who did not. The logistic regression model generates a probability score for each employee, and a threshold can be specified to categorize each person as likely to quit or unlikely to leave.

Random Forest:

Random Forest Classifier, on the other hand, is a form of ensemble learning algorithm that makes predictions using several decision trees. It operates by training a large number of decision trees on a part of the input features and a subset of the training data. When making a prediction, the algorithm combines the findings from all of the trees to reach a final choice.

The Random Forest Classifier (RFC) method can be used to identify the most important input features that contribute to an employee leaving the organization in the case of the HR employee dataset. The technique works by training a large number of decision trees on the input characteristics and evaluating the importance of each feature in predicting outcomes. Based on the input features, the Random Forest Classifier algorithm can then be used to predict which employees are most likely to leave the organization.

K - Nearest Neighbor:

K-Nearest Neighbors (KNN) is non-parametric algorithm which bases its predictions on how closely the input data resembles the labeled samples in the training dataset. The most important step in this analysis is selecting K, a lower number of K can make the model more sensitive to noise, whereas a larger value of K makes the model more resilient to outliers but can smooth decision boundaries.

By measuring the separations between the input data points and their K nearest neighbors, the KNN model is fitted to the training dataset. The input data point will be given the class label that the vast majority of the K closest neighbors have. After the KNN model has been trained, it may be used to create predictions based on fresh, unused data points. Determine the distances from each test data point to its K closest neighbors in the training dataset, and then designate the most prevalent class label among the neighbors as the test point's projected class.

Experiment with various K values and assess their effects on the performance of the model. To increase the model's accuracy even further, you might investigate different strategies like feature engineering or hyperparameter tuning.

## Gaussian Naive Bayes

Gaussian Naïve Bayes method is a probabilistic machine learning approach used for classification tasks. It is based on Bayes' theorem and the notion of feature independence. The algorithm is well-known for its simplicity and agility while working with high-dimensional datasets.

In the case of HR employee dataset, Gaussian Naive Bayes can be used to predict which employees are likely to leave the organization based on input features such as satisfaction level, last evaluation, number of projects, and so on. Given the input features, the algorithm estimates the probability of an employee leaving or not leaving the organization. Gaussian Naive Bayes assumes that the input parameters are independent of one another and that each feature's distribution is Gaussian (normal). This means that each feature is treated as a normal distribution with a mean and standard deviation, and the probability density function of the input features is used to calculate the likelihood of an employee leaving or staying with the organization. Using the training set, this approach estimates the mean and standard deviation of each input for each class (left or not left). The algorithm then applies Bayes' theorem to compute the probability of an employee leaving or not leaving the organization given the input features.

Support Vector Machine

On the dataset of HR employees, classification tasks can be performed using the supervised machine learning method known as the Support Vector Machine (SVM). Pre-processing the dataset first include resolving missing values, deleting unnecessary columns, encoding categorical variables, and dividing the dataset into training and testing sets. Second, for the SVM algorithm to perform better, feature scaling methods like StandardScaler or MinMaxScaler must be used on the input features. The SVM model is then trained on the training set, looking for the optimum hyperplane that divides the two groups, former workers and current employees, by the greatest margin.  
The margin, which is maximized during training, is the separation between the hyperplane and the nearest data points for each class. Hyperparameters like the kernel function, regularization parameter (C), and gamma parameter should be modified using methods like GridSearchCV in order to improve the performance of the SVM model. Finally, measures like as accuracy, precision, recall, F1-score, and confusion matrix may be used to assess the performance of the SVM model on the testing set. SVM is a strong method that can handle high-dimensional datasets and nonlinear decision boundaries, but it uses more processing resources and is sensitive to the selection of hyperparameters, thus it's crucial to fine-tune the hyperparameters to get the best results.

We use the above-mentioned models in phases while introducing important features, cross validation, grid search in each. We assess the models’ performance based on their r-squared value on the test sets.

Table

Description automatically generated

We see the Random Forest Classifier performing the best with a 99.2% accuracy in predictions.

With Important Features

We make use of the important features we shortlisted through feature engineering and exploratory analysis. We use the top 5 important features and get the data for only those features and re-run the models and get the scores to compare.

Table

Description automatically generated

We see an improved performance from all the models, but Random Forest still outperforms the other models.

Cross Validation

In the next phase we introduce 10-fold cross validation onto the data. We compare the model performances again. In this approach, the data is divided into 10 equal-sized subsets or folds. The model is trained on 9 folds and evaluated on the remaining fold. This process is repeated 10 times, each time using a different fold as the validation set. The performance metrics obtained from each iteration are then averaged to produce a more reliable estimate of the model's performance.

Table

Description automatically generated

Slight improvement in performance for each of them, but considering the computational complexity cross validation brings, it is not a good trade off to run cross validation.

Grid Search

In the final phase, we also introduce Grid search for tuning the hyperparameters for each model. Hyperparameters are those parameters that are set before the model is run and are used to set the limitations on the scope of the model. The hyperparameters selected cover all the use cases of the model and the model performance of the model with the best parameters is selected for comparison.

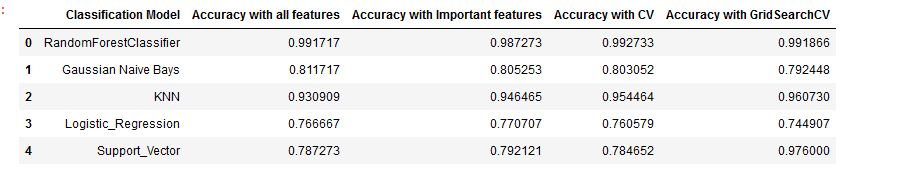
Table

Description automatically generated

Again, there are very slight changes in the performances of all the models except for SVM, we see a marked improvement of 20% in SVM accuracy. This explains the sensitivity of SVM with respect to hyperparameters.

# Result

In this study, the evaluation metric selected was the accuracy score, which measures the proportion of correctly classified instances in relation to the total number of instances. The models were evaluated on all features, important features, and grid cv variations of data. The Random Forest Classifier, an ensembled machine learning model, and K-nearest neighbour achieved the highest accuracy scores, both surpassing 94% in all data variations. The Gaussian Naive Bayes model also performed well with an accuracy score of over 80% for all data variations. However, the logistic regression model underperformed compared to other models, achieving an average score of around 76% in each of the data variations.



It is important to acknowledge some limitations in this study. One limitation is the potential bias in the target variable, which could have affected the accuracy of the models. In addition, it would have been more appropriate to select evaluation metrics such as AUC ROC score, sensitivity, specificity, and f-1 score to better capture the performance of the models in terms of their ability to correctly identify the positive cases.

# 

# References

1) HR Analytics Dataset – Kaggle -  
 <https://www.kaggle.com/datasets/homayoonkhadivi/human-resource-datasets>

2) Predict Attrition Rate – Medium -  
<https://mandynicolehong.medium.com/hr-analytics-predicting-employee-attrition-with-machine-learning-9ce2f05fce65>

3) Employee Retention Plan – Towards  
https://towardsdatascience.com/building-an-employee-churn-model-in-python-to-develop-a-strategic-retention-plan-57d5bd882c2d

4) Chen, H., & Zhao, Y. (2020). Employee retention prediction in talent management using machine learning. Journal of Intelligent & Fuzzy Systems, 39(1), 1203-1214.

5) Hameed, S., & Waheed, A. (2019). Employee retention prediction using data mining techniques: A case study of a large organization. International Journal of Computer Applications, 181(38), 35-41.