Machine Learning II: Employee Attrition Prediction Model

Group B

I. Executive Summary

The goal of this project is to predict whether an employee will voluntarily leave the company, also called attrition.

As a base in our predictive model we used a dataset which contains information about employees' demographics, job level, department, pay, etc. Through these features we will be able to build a model that will predict whether an employee will leave the company or not.

We implemented feature engineering to clean the data and optimize the algorithm's predictions. The techniques we implemented were:

- Categorical variables encoding using Label Encoding
- Creating new variables, from existing ones, that contain higher quality information for the model, thus, removing correlated features from our model.

We implemented different models, parameters, and techniques for each to get the best prediction capabilities. For the final model, we implemented a combination of Random Forest and XGBoost, which had an AUC score of .82.

With our predictions, **the company can utilize them with strategies for employee retention** such as:

- Sending monthly or quarterly surveys to gather feedback regarding employee dissatisfaction.
- Focus groups to get more in depth information and ideas to improve the employee experience.
- Providing benefits or salary increases to top performers who are in risk of leaving.

II. Data Description and Preparation

The primary aim of our machine learning model is to **accurately classify employees as either attrited or non-attrited**.

Our dataset comprises 1677 employee records with 35 features. The initial exploration of the data is an imperative step, as the quality of the model is affected by the quality of the data that is used to train it. After analyzing the data, we discovered some valuable information to help us develop our predictive model:

- There were no missing values in the data.
- 3 features contained a single value and provided no useful information :
  - `'Over18'`

    - 'EmployeeCount'
    - 'StandardHours'
- Our target variable 'Attrition' exhibits an imbalanced distribution, with a ratio of approximately 1:10 between attrited and non-attrited records (Figure 1). As expected, there is a lower percentage of employees that leave a company compared to those that stay.

III. Feature Engineering

The Feature Engineering part is the most important pre-processing step for creating any predictive algorithm, since the data has to be in a format that can help optimize the model to get the most accurate results.

Step 1: Deleting unnecessary features

Firstly, we dropped the 3 features mentioned above ('Over18', 'EmployeeCount', and 'StandardHours') with single values because these do not provide useful information for the model.

Step 2: Creating New Variables

After checking the correlation between variables, our group determined to modify certain original variables to better suit the implementation of our machine learning model, with the goal of improving its overall performance. Here are the newly created variables:

- **'Age_group'**: categorizing 'Age' into 5 different groups ('18-30', '31-40', '41-50', '51-60', '61-70')

- **'Distance_group'**: categorizing 'DistanceFromHome' into 3 different groups ('Short', 'Medium', 'Long')

- **'Satisfaction_level'**: summing 'EnvironmentSatisfaction' and 'RelationshipSatisfaction'

- **'Job_Title'**: 'JobRole'+' - '+'Department'

- **'Experience_Factor'**: 'TotalWorkingYears' * 'JobLevel'

Step 3: Numerical variables

For numerical columns, we created box plots and examined the distribution of the variables. During this process, we observed the presence of outliers in certain columns and we have chosen to retain these outliers in our dataset, as they may contain valuable information for predicting and explaining the attrition rate.

For our final random forest classifier model, we did not scale the numerical features. We took his decision because decision trees methods are not sensitive to the variance in the data. More information here.

Step 4: Encoding categorical variables

To facilitate further data analysis, we applied Label Encoding for all categorical features (Figure 2).

The main reason for not using OHE (One Hot Encoding) on our model is that this kind of technique can really tank the performance of tree-based models. *"By one-hot encoding a categorical variable, we are inducing sparsity into the dataset which is undesirable."*More information about this here.

Step 5: Dropping Used and Highly Correlated Variables

After creating new variables, we dropped some of the original ones that have been aggregated (`‘EnvironmentSatisfaction’, ’RelationshipSatisfaction’, ‘DistanceFromHome’, ‘Age’, ‘JobRole’, ‘Department’, ‘JobLevel’, ‘PercentSalaryHike’, ‘TotalWorkingYears’`). In addition, after creating `‘Experience_Factor`’, it became highly correlated with `‘MonthlyIncome`’, which is why we decided to drop the latter one as well.

Also, there are a few highly correlated variables in our data, including `‘StockOptionLevel’` & `‘MaritalStatus’` and `‘YearsAtCompany’` & `‘YearsInCurrentRole’` & `‘YearsWithCurrentManager’` & `‘YearsSinceLastPromotion’.` Therefore:

- We dropped `‘StockOptionLevel’, ’YearsInCurrentRole’, ‘YearsWithCurrManager’, ‘YearsSinceLastPromotion’.`
- We only kept `‘MaritalStatus’` and `‘YearsAtCompany’.`

As a result of the feature engineering steps described above, we now have 9 categorical variables and 12 numerical variables, which will be useful for further analysis. It’s important to note that during our data cleaning and feature engineering experiments, we had to make significant assumptions as to what the specific values meant for certain variables such as Performance Rating or Job Level.

IV. Model Development and Deployment

As a trial, we ran 3 different models:

- Basic Random Forest
- XGboost Classifier
- Logistic Regression

However, the results weren’t ideal, with AUC only around 0.54 for the 3 models (Figure 3). The reason may be the unbalanced dataset we have, as the ratio between attrited and non-attrited records is approximately 1:10.

In addition, for the Logistic Regression, we used Label Encoding, which is not ideal for this specific algorithm. However, we made this decision based on the assumption that we were going to need a more powerful algorithm, such as a Random Forest. In addition, we scaled the data only to try this specific algorithm, but scaling was not implemented again when we decided that Random Forest was the best algorithm for our needs.

We considered applying undersampling to handle the unbalanced dataset. However, we realized that the disadvantage of undersampling is that it may result in the loss of important information because some data from the majority class will be removed.

Therefore, we decided to go with another method, which is **iterative sampling**. This method has some similarities with *Bagging*, but instead of being run in parallel, its run iteratively, storing all predictions made instead of discarding the worst ones, the benefit with this is that it generates synthetic data points that are similar to the minority class, reducing bias and improving the model's ability to generalize. To be more specific, our dataset contains a total of 200 instances of attrition (represented as 1s) and 1477 instances of non-attrition (represented as 0s). In order to ensure balanced training data, we conducted seven iterations of model training, each time utilizing the same 200 instances of attrition but with a different set of 200 instances of non-attrition. This approach ensured that the model was trained on an equal number of attrition and non-attrition instances in each iteration.

After completing the seven iterations, we still had 77 instances of non-attrition that were not used in the training process. These remaining instances will be utilized in our validation set, which will provide an additional measure of model performance on unseen data.

V. Final Model

After conducting the trials, we observed that both the **XGBoost** and **Random Forest** models possess unique strengths, prompting us to combine the two approaches to extract the maximum benefits from each. Below are some advantages of each algorithm and the combination of the two:

Benefits	XGBoost	Random Forest	Combination
Better prediction accuracy	✔️	✔️	Produces a more accurate and robust model.
Improved generalization	✔️	✔️	Reduces the risk of overfitting and improves generalization.
Enhanced feature importance	✔️	✔️	Provides a more comprehensive understanding of which features are most important.
Faster training time	✔️	✔️	Decreased training time, without sacrificing model performance
Better interpretability	❌	✔️	Increased interpretability than a XGBoost model alone.
Robustness to outliers	❌	✔️	Produces a more robust model that is less affected by outliers.

Our ultimate prediction outcome will involve computing the mean of the predictions obtained from XGBoost and Random Forest models, followed by rounding the result to 1 or 0.

For measuring the prediction power of our random forest classifier, we decided to not use the accuracy metric, because the target variable is imbalanced, as we previously explained. Therefore, we decided to use the area under the curve (AUC) metric,which is better representative of our model's prediction power. The final AUC score for our model was .82 for Public Kaggle and .76 for the Private Kaggle.

This model is a good starting point for predicting attrition rates in companies and taking preventative measures to retain talent to avoid the loss of expertise, relationships, productivity, etc. However, for this model to be as efficient as possible it needs to be implemented along with specific strategies to act upon the predictions and use tactics to retain employees, specifically those that are the top performers at the company.

VI. Insights and Business Strategy for Employee Retention

There are many reasons for high attrition rates in companies including poor leadership, lack of recognition, unsupportive work environment, etc. According to the U.S. Bureau of Labor, the average employee turnover rate in 2021 was 47.2%. This is detrimental as companies that have a high turnover rate tend to perform worse than competitors due to low productivity and employee engagement.

Therefore, being able to predict what employees are likely to leave a company will prove itself to be vital to keep employee satisfaction high and to drive the success of the company in general. **Some of the benefits of our models predictions are:**

- Less energy and time spent by the human resources team in the recruitment process, as they will not need to constantly replace lost talent. This will lead to financial savings and will allow more resources to be used for employee growth and career development initiatives.

- Consequently, new employees will not be constantly hired, which means that onboarding and training costs will decrease. This will not only allow human resources to focus on employee development, but will also leave more time for managers and senior employees for other projects and initiatives. Additionally, keeping the same talent for a significant period of time allows for different teams and departments to gain expertise in their specific industry, which can only be acquired through experience.

- Managers can take preventative measures with employees who are predicted to be at risk of leaving. These can include meetings to talk about job satisfaction, goal setting, and general feedback for how the employee experience can be improved.

- With a better understanding of employee behavior and the factors that contribute to retention, the company can make informed decisions that lead to enhanced organizational performance and competitive advantage.

By utilizing our model, the company can have a data-driven approach to handle employee retention and can implement a variety of strategies to keep employee satisfaction and engagement. **Some of the employee retention strategies include:**

- Sending monthly or quarterly surveys to all employees, adding emphasis to those that are predicted as leaving, to gather feedback as to what aspects of the company are causing their dissatisfaction. The answers from their surveys will be analyzed by the management team in order to create new strategies for the future and re-define the strategy of the company if needed.

- Sending invitations for focus groups, led by the human resources department, to those who responded to the survey to get more in depth information about their dissatisfaction and also ideas for how to improve the employee experience. The focus groups will be divided by department as the organizational structure and duties are different for each.

- Providing benefits or salary increases to those employees who are predicted as leaving, but also have the highest performance rating or those employees who have not been promoted in 4 years or more. With these financial incentives, employees are less likely to leave to a competitor based only on a higher compensation.

- Implementing career development programs, such as mentorships or training, for entry-level employees who have been hired less than a year ago and are predicted as leaving. These programs will be implemented in order to encourage employees to keep engaged with the company and increase their skills to improve productivity.

- Creating a positive company culture that emphasizes teamwork, collaboration, and respect can create a sense of community and belonging among employees, making them more likely to stay with the company long-term.

VII. Conclusion

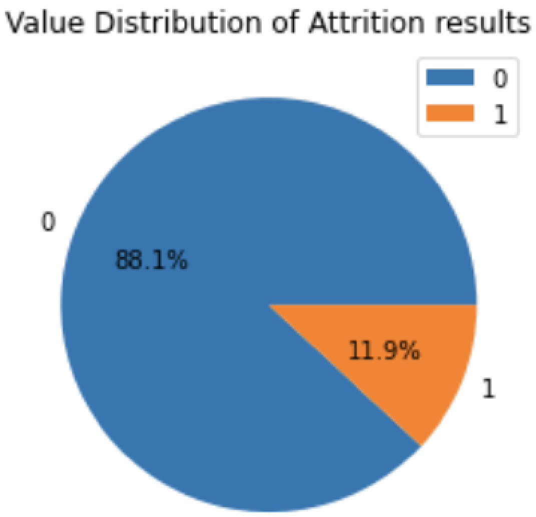
Upon completing the project, our group gained valuable insights for future endeavors, such as:

- Combination of XGBoost and Random Forest can produce a more robust model.
- Tree-based algorithms require minimal data cleaning in terms of scaling and encoding.
- Iterative sampling can be a superior alternative to undersampling to prevent loss of crucial information.

As a recommendation for future improvement, we suggest for the company to gather more information about the employee’s satisfaction and engagement such as: satisfaction changes over time, how much autonomy employees feel, number of projects they are involved in, etc. This is because there are ways to clean data and improve algorithms, but having relevant data is what helps the most to get more accurate predictions.

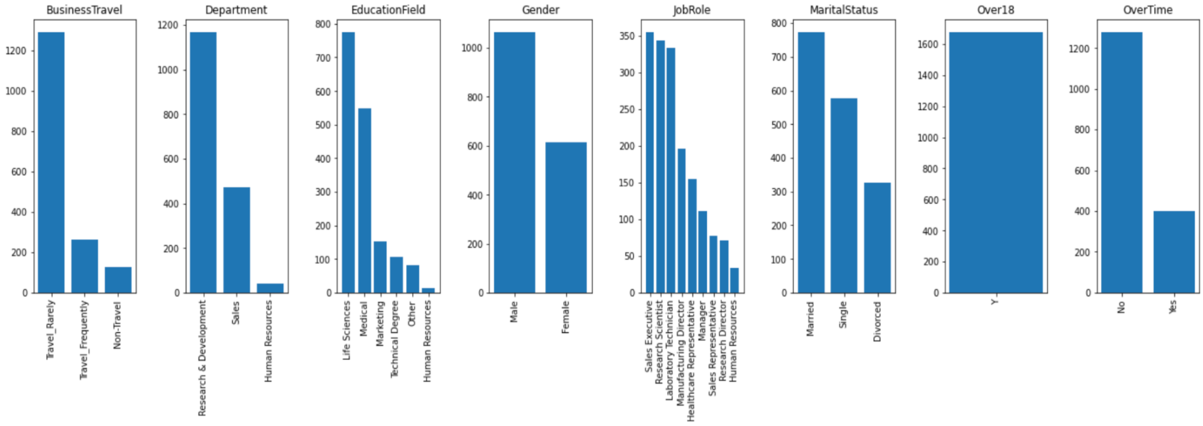
VIII. Appendix

Figure 1: Balance of Target Variable “Attrition”



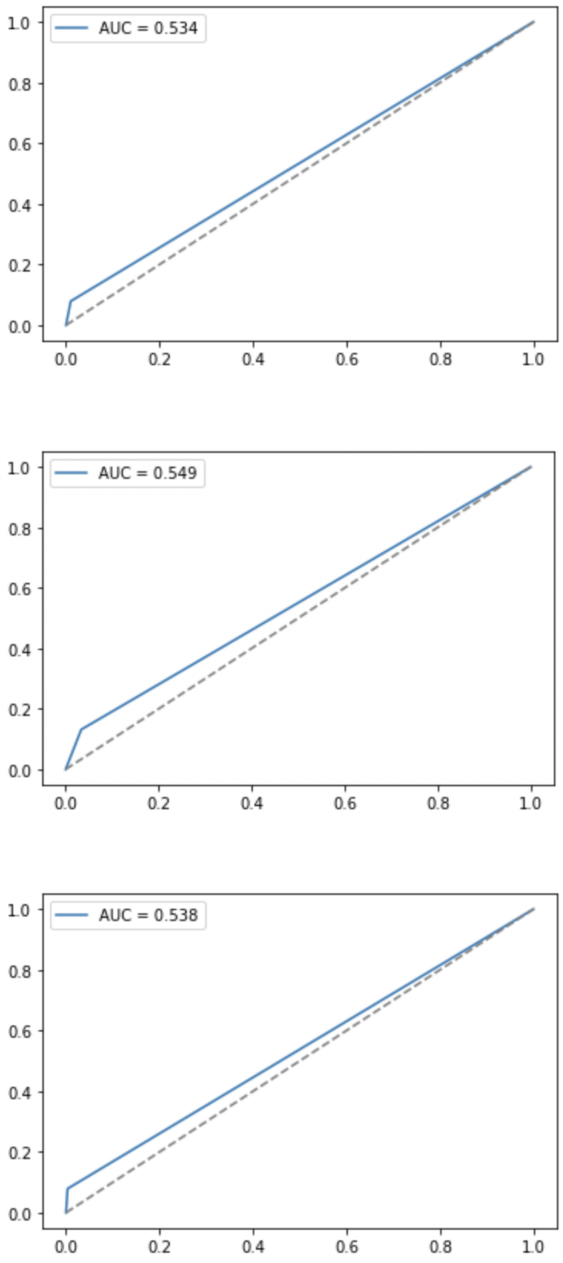
Our target variable 'Attrition' exhibits an imbalanced distribution, with a ratio of approximately 1:10 between attrited and non-attrited records. This affects the way we will train our model as well as the metrics that we will use to check how effective it is in predicting the attrition rate.

Figure 2: Distribution of Categorical Variables



The distribution of the categorical variables helped us figure out what changes we could do to clean the data, which included using Label Encoding and creating new variables that contained better information to train the prediction model, such as Job Title.

Figure 3: AUC score for Basic Random Forest, XGBoost Classifier, Logistic Regression models respectively



As we can see, a score of less than .6 is not ideal for a predictive model. The reason for the low scores may be the unbalanced dataset we have, as the ratio between attrited and non-attrited records is approximately 1:10.