# Attrition Rates - Predicting Future Employee Resignations

*Mohammad Ayan Raheel, Marella Ma, Peter Ye, Tina Tong*

## Table of Contents

## Introduction

High Employee Attrition Rates affect companies negatively as organizations want to retain talent – there are costs associated with losing employees. Whether an organization is losing its employees to their competition or losing the money they invest to onboard an employee, a high attrition rate gives an organization a negative reputation.
To calculate the Attrition Rates, we must check the total percentage of employees that quit over the average number of employees. Predominantly, attrition rates are considered one of the most important metrics for the Human Resources (HR) department; monitoring these rates are valuable in gauging whether the HR teams to improve workforce planning and people management.
To balance the attrition rate is to signal a healthy workforce, whereas ineffectiveness in balancing the attrition rates can indicate underlying issues that need to be addressed by the relevant departments of the organization. Hence, being provided a company's data on their employees, we are trying to build a best-fitting machine learning algorithm to accurately predict which employees will end up leaving the organization in the future. From this analysis, we will primarily focus on the predictions for the future, however, in the process will also inevitably check which features are the most important in predicting the organization's attrition rate.

## Objectives

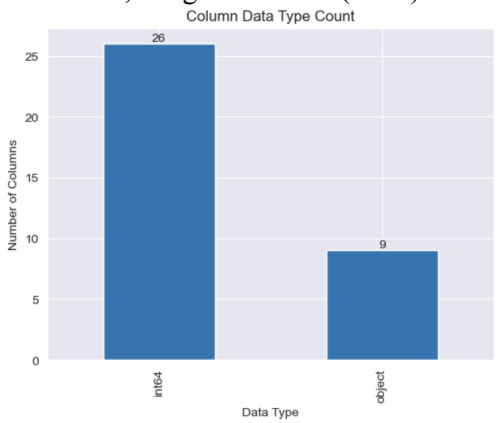
Our solution proposed is to create machine learning algorithms to accurately predict future attrition for the organization. Working with the dataset, our solution will use supervised learning methodologies in order to create the best performing model that accurately categorizes an employee into Class 0 or Class 1. The model's classification in two distinct categories will accurately be used to show the susceptibility of an employee to leave the organization - much like Customer Churn analysis where the goal is to accurately predict whether a specific, particular customer will end up churning. Other examples with similar characteristics include image classification and medical diagnosis classification within the healthcare sector. Such problems share key characteristics and structure in their predictions based on supervised learning techniques.
Hence, we categorize this problem as a supervised learning problem. This is because in supervised learning, the algorithm that we are developing is trained on the labeled dataset, where the input data is paired with the corresponding correct output. This means that we have a train and test set, where the independent (x) variables are utilized to predict the dependent (y) variable. While the independent variables in our dataset are defined below (every variable that is not attrition), the Class 0 (attrition = no) and Class 1 (attrition = yes) is our final prediction we are trying to get at. Similarly, this problem is also framed as an offline problem as the model we are utilizing is trained on a fixed dataset (which will be discussed below). However, if there are any changes to the dataset in the future, we would like to be informed in order to incorporate a constant learning approach.
To evaluate the performance of our model, we will be checking the Accuracy, Precision, Recall and F1 Scores. These metrics are used to check how our individual as well as collective models are performing in a classification setting. To check the model performance further, we will also discuss the Confusion Matrix and the RP-ROC scores. This makes the performance measure in line with the business objective to predict future attrition – since our overall objective is to focus on identifying employees at risk. Harmonizing the Recall and Precision, our primary focus then becomes the F1 Score.
To reach the business objective, therefore, the minimum model performance needed would entail defining the success criteria in predicting attrition. Defining recall as the risk of misclassifying an employee who is to leave in the future and precision as risk of misclassifying an employee who does not leave, a target level of 0.5 F1 Score is needed. At the same time, we will also try to minimize precision which entails that an employee will leave the organization even though they do not end up leaving. Accuracy may not be the best metric to use as our dataset is imbalanced between the classes (attrition = yes or no). Generally, when there is a class imbalance, accuracy does not provide a valid measure. Lastly, as all great data science teams, we will work collaboratively within the team and with the organization's human expertise to solve the problem. To manually solve the problem, we will present an understanding of the problem, define the methodology used, utilize and clean the dataset we are provided and present our findings.
Our assumption for the project includes that the dataset is complete and is being provided by the HR department based on employees that are currently working at the organization. Similarly, if there are to be any changes in the structure of the organization, we shall be notified immediately so that we can clean the dataset from our end. The data we have received is all encompassing, and we assume alignment between our expectations and the expectations of the stakeholders of the organization. This can all be verified through open communication with the organization, which has been taking place.

## Problem Statement

A debt recovery organization is struggling with a high attrition rate of 14%. We are tasked with helping the organization address the issue. Based on the organizational expectations, we are answering the following questions –
**Which employees are at risk of leaving the organization – predicting future attrition? (Primary Concern)**
**After modeling, which features are the most important in leading to high attrition rates? (Secondary Consideration)**

## Data

The data provided comprises a total of 35 columns – 34 of which form our independent variables. The 'Attrition' column is our dependent variable and takes on a Boolean value of either 'Yes' or 'No.' Out of the 34 columns that are in our independent set, 26 columns are numerical, integer variables (int64) and 9 are categorical variables.




Additionally, checking for NULL values, we observed that the dataset provided was rather clean. Had this not been the case, we would have had to clean the dataset in the data preprocessing stage – cleaning null values can be done in one of three ways. These ways include either dropping the NULL/NA values as a whole, importing mean, median or mode for the NULL columns and rows depending on the nature and structure of the data (skewness and distribution) or adapting the k-nearest neighbors' data. In our analysis, we found "EmployeeNumber" is a unique identifier to identify which particular employee we are referring to within the organization (based on the badge number of the employee) and will hence be dropped before we can start our analysis. "StandardHours" only represents the total number of minimum hours an employee is expected to work at the organization and should be dropped from our analysis as well. Given that Attrition is a categorical variable, converting it into a binary column will enhance our ability to conduct numerical feature analysis. This transformation will simplify the process of integrating attrition data into quantitative analyses, allowing for more straightforward interpretation and evaluation of its impact. From there, we convert our dependent variable "Attrition" to a binary column to conduct numerical feature analysis.

## EDA and Graph Analysis

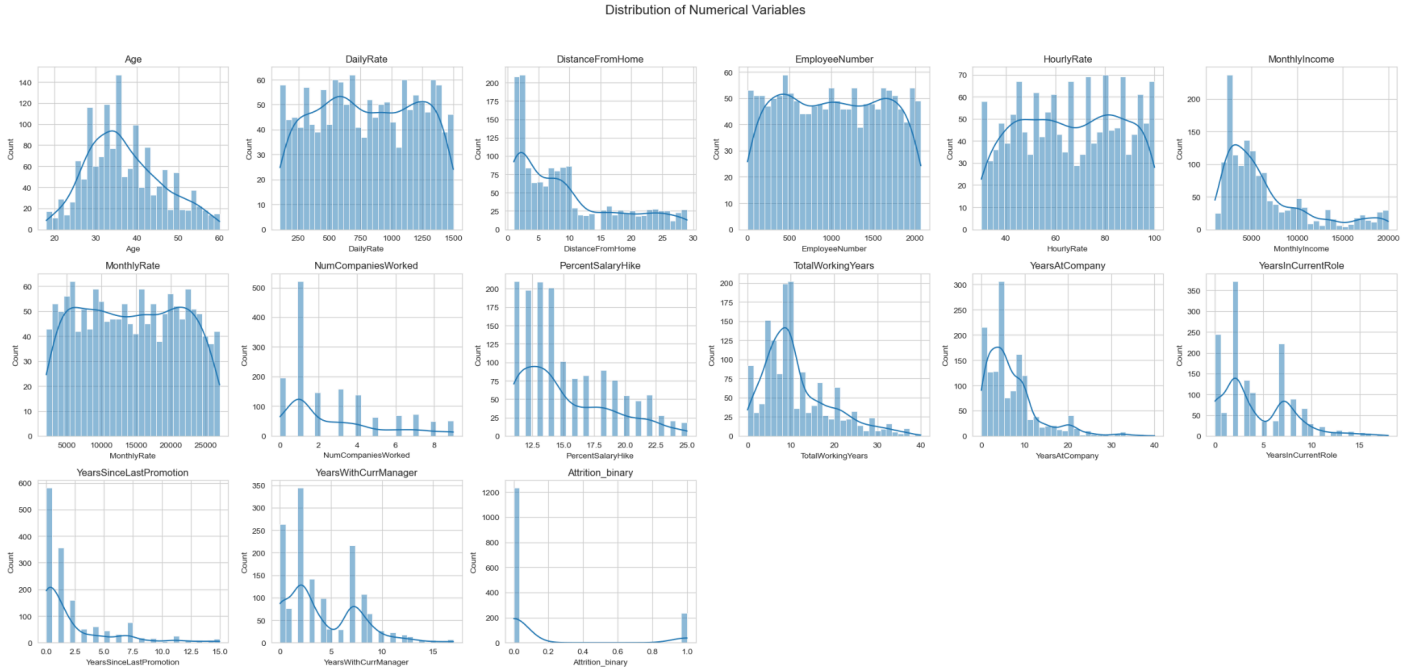
First, we plotted the distributions for all the 26 numerical variables to check if any outlier data points and noise is present within our dataset. This is important to check as noise can disrupt our analysis by skewing the data, making it hard to derive meaningful insights. Provided in the appendix is a quick summary of the columns in our dataset with data descriptions.
Further segmenting our understanding of the dataset, we conducted VIF (Variance Inflation Factor) analysis to check for multicollinearity. This entails examining the correlation between the independent variables in our dataset to assess how they relate to each other. The reason for addressing multicollinearity is that presence of it in the data can undermine the statistical significance of our independent variables and can affect Accuracy Scores.

| Variable | VIF | Variable | VIF |
|---|---|---|---|
| Age | 2.005141 | NumCompaniesWorked | 1.24991 |
| DailyRate | 1.015706 | PercentSalaryHike | 1.006563 |
| DistanceFromHome | 1.014292 | TotalWorkingYears | 4.644095 |
| EmployeeNumber | 1.005897 | YearsAtCompany | 4.574045 |
| HourlyRate | 1.005781 | YearsInCurrentRole | 2.703721 |
| MonthlyIncome | 2.536239 | YearsSinceLastPromotion | 1.681179 |
| MonthlyRate | 1.008155 | YearsWithCurrManager | 2.762414 |
| Attrition_binary | 1.082243 | | |

As can be seen from the table above, every numerical variable in our dataset has a VIF reading of below 10. This points to low levels of multicollinearity and that our independent variables are sufficiently independent for the purpose of our analysis. Why this is important is because for Logistic Regression, we operate under the IID assumptions.

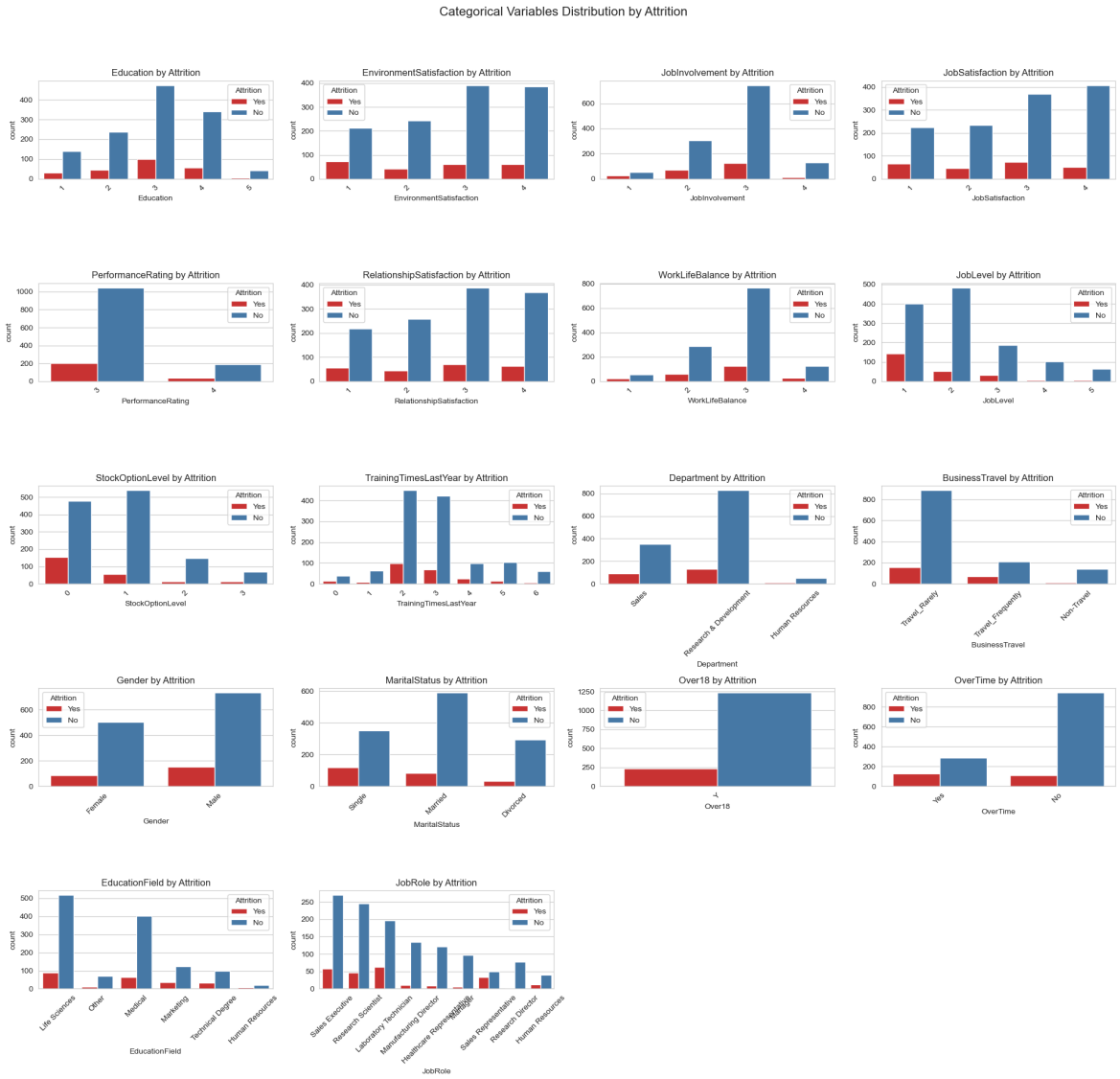
### Distribution of Numerical Variable

Most variables display a variety of distribution shapes, including normal-like distributions (e.g., Age), skewed distributions (e.g., TotalWorkingYears, YearsAtCompany), and distributions with clear peaks (e.g., Education). Several variables exhibit potential outliers. For instance, TotalWorkingYears, YearsAtCompany, YearsInCurrentRole, YearsSinceLastPromotion, and YearsWithCurrManager show long tails, suggesting the presence of employees with unusually high values compared to the rest.
There is some stochastic noise that shows the variability seen in variables like DailyRate, HourlyRate, and MonthlyRate appears stochastic, reflecting natural fluctuations in compensation rates. In terms of rounding errors, there is no clear indication of rounding errors from these plots; most continuous variables show smooth distributions without unnatural spikes that would suggest rounding. These variables are encoded as integers but represent ordered categories, making them ordinal categorical variables. Their discrete and limited nature suggests that analyses should treat them as categories, which can provide insights into group differences or trends that might not be apparent when treating them purely as numeric.

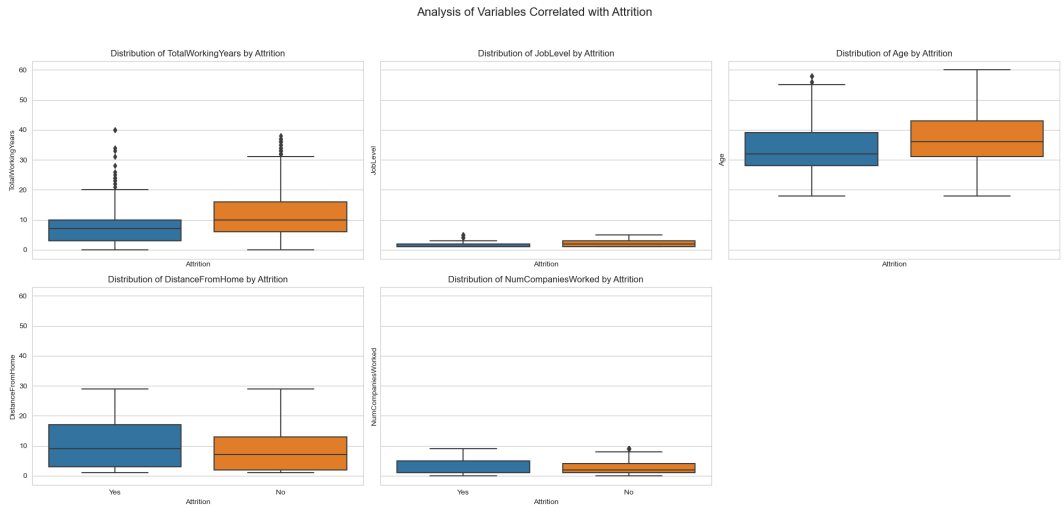



### Categorical variables distribution by attrition

These bar graph visualizations illustrate the distribution of both transformed and original categorical variables by attrition status, providing insights into how different categories may influence an employee's decision to leave. Education, EnvironmentSatisfaction, JobInvolvement, JobSatisfaction, etc.: These transformed categorical variables show varying distributions across their categories when split by attrition status. Certain levels of satisfaction (e.g., lower environment satisfaction) and involvement might be more associated with higher attrition rates. Department: The distribution suggests that attrition rates may vary across different departments, with some departments potentially experiencing higher turnover. BusinessTravel: Employees who travel frequently appear to have a higher attrition rate compared to those who travel less or not at all, indicating that travel demands may impact employee retention. Gender: The attrition distribution between genders offers insights into whether one gender may be more likely to leave than the other, although differences may not be pronounced. MaritalStatus: The attrition rates differ among single, married, and divorced employees, with single employees potentially showing higher attrition, suggesting marital status may play a role in an employee's likelihood to leave. Over 18: Only has one two-sided bar, indicating that every employee is over 18. We should drop it before building a model.

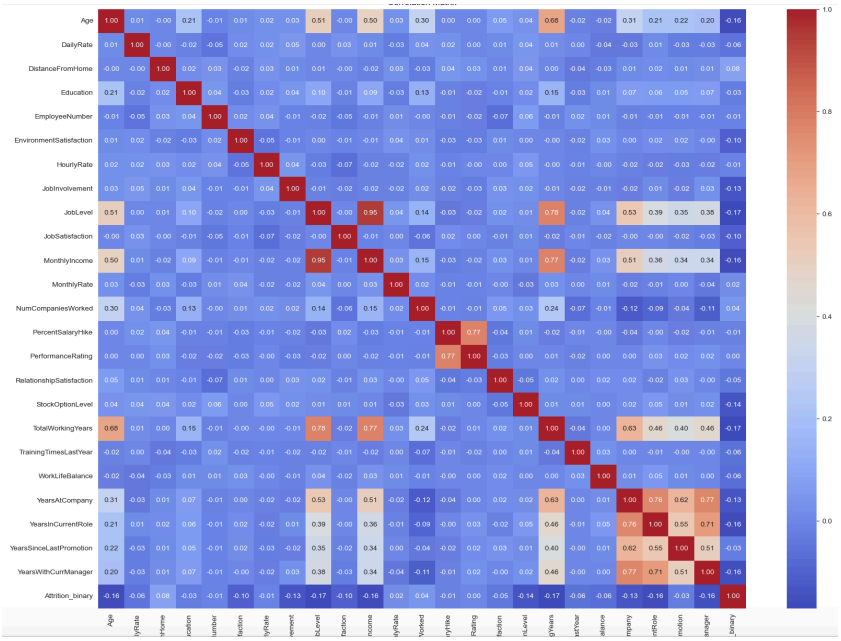



## Analysis of variables correlated with attrition



The boxplot visualizations provide a detailed view of how the selected variables correlate with attrition: TotalWorkingYears: Employees who have left the company tend to have fewer total working years compared to those who stayed, indicating that individuals with more experience or tenure are less likely to leave. JobLevel: Similarly, employees at higher job levels are less likely to have left the company. This suggests that higher positions, which likely come with better compensation and job satisfaction, are associated with lower attrition rates. Age: The age distribution shows that younger employees are more likely to leave than older ones. This could be due to various factors, including career exploration, seeking better opportunities, or less attachment to the organization. DistanceFromHome: Employees who live further from work are more likely to leave the company. The commute might be a significant factor in their decision to leave, possibly due to the stress and time associated with long commutes. NumCompaniesWorked: There's a noticeable trend where employees who have worked at more companies are also more likely to leave. This could indicate a pattern of job-hopping or a preference for change, suggesting these employees may be less inclined to long-term commitments to a single employer.

**Correlation Matrix**




The heatmap above displays the correlation matrix for all numerical variables in the dataset. Each cell in the heatmap shows the correlation coefficient between two variables, ranging from -1 to 1. A coefficient close to 1 indicates a strong positive correlation, meaning that as one variable increases, the other tends to increase as well. Conversely, a coefficient close to -1 indicates a strong negative correlation, where an increase in one variable is associated with a decrease in the other. Coefficients around 0 suggest little to no linear relationship. Variables like TotalWorkingYears, JobLevel, and Age have notable negative correlations with attrition, reinforcing the idea that more experienced and older employees are less likely to leave. DistanceFromHome and NumCompaniesWorked show positive correlations with attrition, suggesting factors such as commute distance and a history of working for multiple companies may contribute to an employee's likelihood of leaving.

A strong positive correlation indicates that, as expected, older employees tend to have more working years. This relationship is intuitive and highlights career progression. This relationship is shown through the variables of age and total working years. There's a significant positive correlation between job level and monthly income, suggesting that higher job levels are associated with higher monthly incomes. This reflects the typical organizational structure where positions at higher levels command greater compensation. Years at the company shows strong positive correlations with variables like YearsInCurrentRole, YearsWithCurrManager, and YearsSinceLastPromotion. These correlations suggest that employees who have been at the company longer tend to have spent more time in their current role, with their current manager, and it has been longer since their last promotion. This could indicate career stability but might also suggest potential areas for improving career development pathways to prevent stagnation. The positive correlation between total working years and job level indicates that employees with more total working years tend to have higher job levels, suggesting a progression in their careers over time. Monthly income shows strong positive correlations with TotalWorkingYears, JobLevel, and Age. These relationships underscore the impact of experience, position, and age on

compensation. A slight positive correlation suggests that, on average, older employees have worked for more companies. This could reflect career exploration or changes over an individual's working life. This relationship is depicted through the variables of NumCompaniesWorked and age.

## Data Preprocessing and One-Hot Encoding/MinMaxScaler

We initially applied one hot encoding to mitigate any bias towards higher numerical values. This approach ensures that the models do not mistakenly infer larger values hold greater importance.
MinMaxScaler

| | Age | DailyRate | DistanceFromHome | Education | EnvironmentSatisfaction | HourlyRate | JobInvolvement | JobLevel | JobSatisfaction | MonthlyIncome |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0.547619 | 0.715820 | 0.000000 | 0.25 | 0.333333 | 0.914286 | 0.666667 | 0.25 | 1.000000 | 0.262454 |
| 1 | 0.738095 | 0.126700 | 0.250000 | 0.00 | 0.666667 | 0.442857 | 0.333333 | 0.25 | 0.333333 | 0.217009 |
| 2 | 0.452381 | 0.909807 | 0.035714 | 0.25 | 1.000000 | 0.885714 | 0.333333 | 0.00 | 0.666667 | 0.056925 |
| 3 | 0.357143 | 0.923407 | 0.071429 | 0.75 | 1.000000 | 0.371429 | 0.666667 | 0.00 | 0.666667 | 0.100053 |
| 4 | 0.214286 | 0.350036 | 0.035714 | 0.00 | 0.000000 | 0.142857 | 0.666667 | 0.00 | 0.333333 | 0.129489 |

Next, we used MinMaxScaler to normalize features within 0-1. Variables such as Age, DailyRate, Education, HourlyRate are converted to 0-1 values.

## Numerical Features

When first analyzing the data, we ran a statistical value analysis to see the distributions of each variable. Above is a snippet of the output for the first few variables, but the code was run for every variable. After running this analysis, we found that the variables EmployeeCount and StandardHours consistently hold the values of 1 and 80, respectively. This consistency suggests these variables do not vary and, therefore, are unlikely to provide meaningful insights for future research. As such, it is recommended that they be excluded from further analysis.
Given that Attrition is a categorical variable, converting it into a binary column will enhance our ability to conduct numerical feature analysis. This transformation will simplify the process of integrating Attrition data into quantitative analyses, allowing for more straightforward interpretation and evaluation of its impact.

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| Age | 1470.0 | 36.923810 | 9.135373 | 18.0 | 30.00 | 36.0 | 43.00 | 60.0 |
| DailyRate | 1470.0 | 802.485714 | 403.509100 | 102.0 | 465.00 | 802.0 | 1157.00 | 1499.0 |
| DistanceFromHome | 1470.0 | 9.192517 | 8.106864 | 1.0 | 2.00 | 7.0 | 14.00 | 29.0 |
| Education | 1470.0 | 2.912925 | 1.024165 | 1.0 | 2.00 | 3.0 | 4.00 | 5.0 |
| EmployeeCount | 1470.0 | 1.000000 | 0.000000 | 1.0 | 1.00 | 1.0 | 1.00 | 1.0 |
| EmployeeNumber | 1470.0 | 1024.865306 | 602.024335 | 1.0 | 491.25 | 1020.5 | 1555.75 | 2068.0 |
| EnvironmentSatisfaction | 1470.0 | 2.721769 | 1.093082 | 1.0 | 2.00 | 3.0 | 4.00 | 4.0 |
| HourlyRate | 1470.0 | 65.891156 | 20.329428 | 30.0 | 48.00 | 66.0 | 83.75 | 100.0 |
| JobInvolvement | 1470.0 | 2.729932 | 0.711561 | 1.0 | 2.00 | 3.0 | 3.00 | 4.0 |

## Models Used

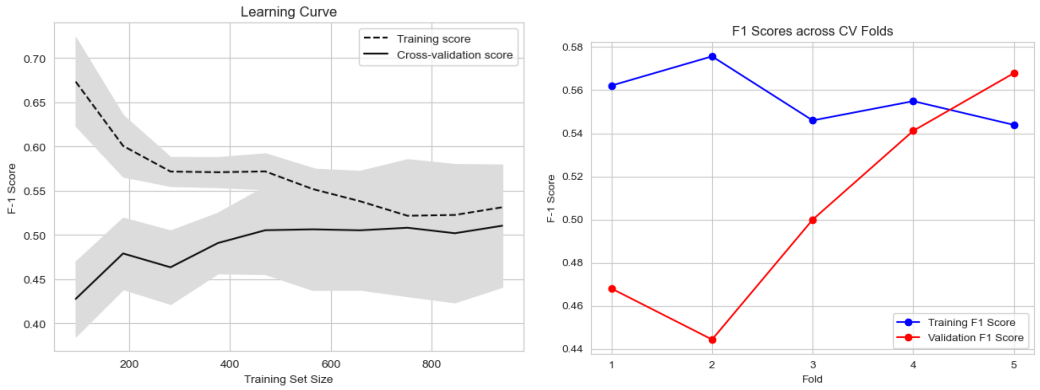
### Logistic Regression

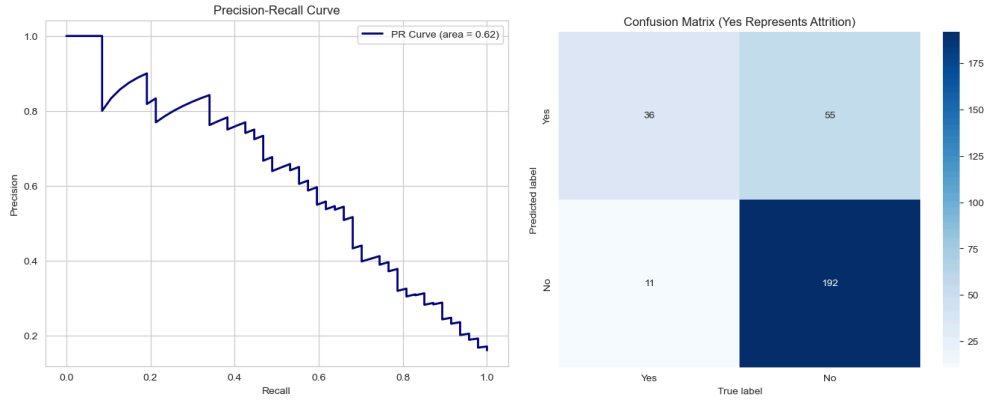
Logistic regression was selected as the initial model due to its appropriateness for binary classification problems, which in our context, is predicting employee attrition (yes/no). It's a foundational statistical method that estimates the probability of a binary outcome based on one or more independent variables. Its simplicity, interpretability, and direct approach to estimating probabilities make it an excellent starting point for understanding factors influencing attrition. Our results of this model offer a solid foundation for strategic HR interventions.
To maintain the original class distribution within both training and testing datasets, we utilized a stratified train-test split. This approach guaranteed that the proportion of each class in our splits accurately reflected the unbalanced nature of our dataset, preventing the model from being biased towards the majority class.
We adopted a StratifiedKFold cross-validation method to ensure that each fold was representative of the overall class distribution, thereby mitigating the risk of model evaluation bias. In tandem with this, a comprehensive grid search was conducted to optimize the logistic regression model's hyperparameters, specifically focusing on regularization strength (C), the solver used (solver), and the strategy for handling class imbalance (class_weight). The inclusion of the

class_weight parameter was pivotal, allowing us to directly address the challenge posed by our unbalanced dataset by adjusting the weight of classes inversely proportional to their frequency. The optimal model configuration, identified through grid search, featured a regularization strength (C) of 0.1, utilized the liblinear solver, and applied a balanced class weight.

| Parameters | Value | Definition |
|---|---|---|
| C | 0.1 | Strong regularization helps to prevent model from overfitting |
| solver | liblinear | Optimization for small data |
| class_weight | balanced | Handle imbalanced data |

The learning curve for the logistic regression model shows that the F1 score decreases on the training set while slightly increasing on the cross-validation set as more data is used, indicating that the model may be starting to generalize better but could potentially benefit from further data or model complexity optimization. The graph of F-1 Scores Across CV folds demonstrates that the validation F1 score has a notable dip in the third fold, which might suggest that the particular split of data in this fold was more challenging for the model to predict. This could be due to a more pronounced class imbalance or a set of harder-to-classify examples in that fold. The training F1 scores are consistently higher than the validation scores, reinforcing the indication of overfitting seen in the learning curve.




From the Precision-Recall Curve, we can observe that the area under the PR curve indicates that the model, with its current configuration, is better at predicting non-attrition than attrition, which could be due to the imbalance in the dataset. The confusion matrix indicates that the logistic regression has a high recall (0.95) for the negative class (class 0), indicating that it is very effective at identifying the majority class, which is likely the non-attrition group. However, precision is low (0.40) for the positive class (class 1), suggesting that when the model predicts attrition, it is correct only 40% of the time. The F1 score for the positive class is 0.52, indicating a moderate balance between precision and recall for the minority class, but this also highlights that there is significant room for improvement, especially in precision. The overall accuracy of the model on the test data is 0.78, which might appear high, but given the class imbalance, this metric may be less informative. The macro-average F1 score is 0.69, reflecting the average

performance across both classes, without taking class imbalance into account. The weighted average F1 score is 0.80, which accounts for class imbalance by weighting the F1 score of each class by its support.

**Support Vector Machines (SVM)**

As with our logistic regression model, we began by splitting our dataset into training and testing sets using a stratified approach. This was to ensure that the class distribution in our unbalanced dataset was consistent across both sets. A stratified train-test split preserves the class distribution, which is critical when modeling datasets with a significant imbalance between classes.

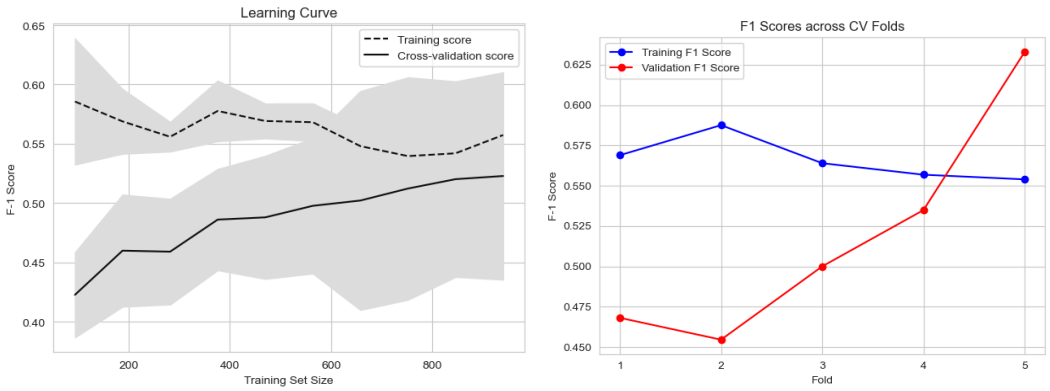
Our data may have high-dimensional spaces and the classes are not linearly separable. Therefore, we chose SVM to capture more complex relationships due to its ability to create non-linear boundaries. Unlike logistic regression, which is based on probability estimates, SVM focuses on maximizing the margin between classes, which can lead to better generalization on unseen data.

We conducted a grid search to explore a range of hyperparameters for the SVM. The parameters included the regularization parameter (C), the kernel type (linear, RBF, poly), the kernel coefficient (gamma), and the class weighting (class_weight). The inclusion of class_weight is crucial for unbalanced datasets, as it allows the model to pay more attention to the minority class by assigning a higher penalty to misclassifications of that class.

The optimal hyperparameters identified were the result of an exhaustive search aimed at maximizing the F1 score, which harmonizes precision and recall and is particularly suitable for evaluating performance on unbalanced datasets.

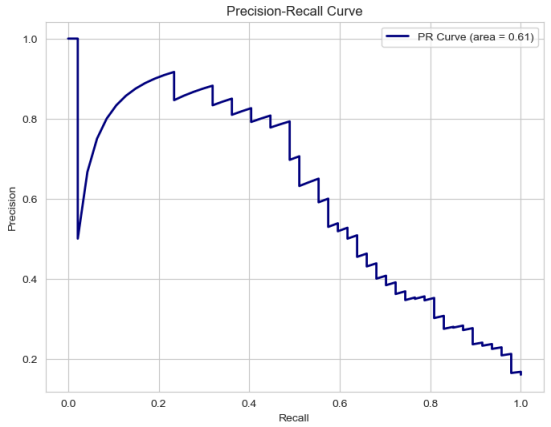
| Parameters | Value | Definition |
|---|---|---|
| C | 1.1 | Trade-off between achieving a low error on the training data and minimizing the norm of the weights |
| kernel | rbf | Handle non-linear boundaries between classes |
| gamma | auto | Set to 1/ number_of_features |
| class_weights | balanced | Handle imbalanced data |

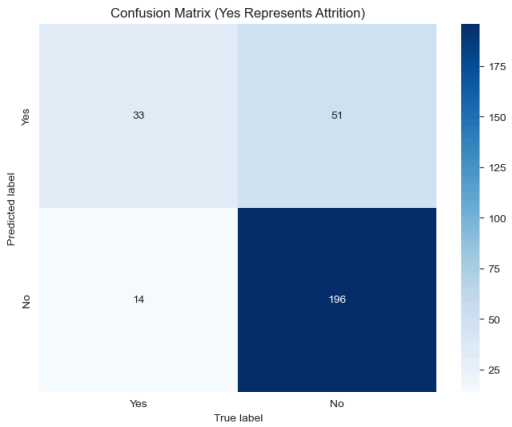
The graph of the Learning Curve demonstrates that the training score starts high and decreases slightly as more data is added, which is typical as the model has more difficulty fitting a larger dataset. The cross-validation score increases with more data, suggesting that the model is benefiting from more data and is improving its ability to generalize. The convergence of the training and cross-validation scores is not yet apparent, suggesting that the model may continue to benefit from more data, or alternatively, the model might need more complex features to improve its performance further. The graph of F-1 Scores Across CV folds demonstrates that there is some variability in the F1 scores between folds, particularly in the validation scores. This could suggest that the model's performance is sensitive to the specific makeup of the folds, which is not uncommon in practice, especially with unbalanced datasets. The general upward trend in the validation F1 score with successive folds indicates that certain folds of the data are more conducive to the model, or that the model parameters are becoming better tuned to the data distribution across folds.




The Precision-Recall Curve demonstrates that the AUC of 0.61 suggests that the SVM model has a reasonable separation between the classes. A perfect model would have an AUC of 1.0. The PR curve starts with high precision at

low recall levels, which means that when the model predicts an employee is likely to leave, it does so with high confidence. However, as the model tries to capture more true positives (increasing recall), the precision decreases, suggesting that it starts to make more false positive errors. The model shows a balance between precision and recall, and a F1 score of 58% for Attrition Class, but there is room for improvement. Optimizing the threshold could potentially increase the area under the curve, leading to better model performance.




**Random Forests**

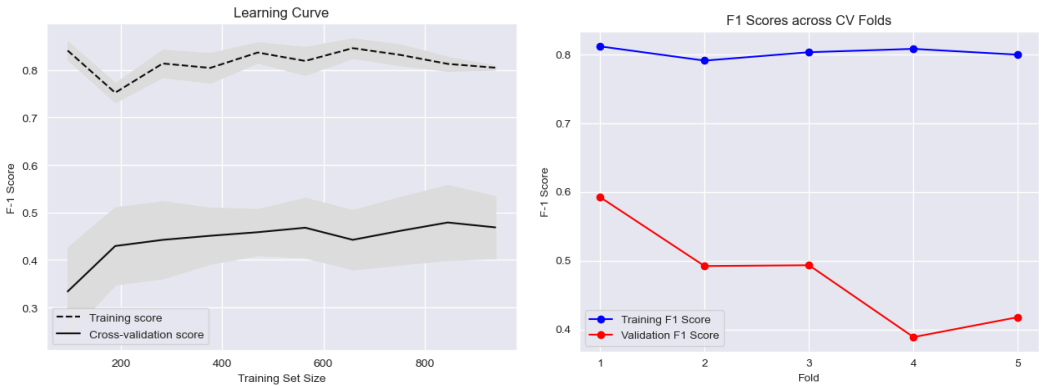
Random forest, a tree-based ensemble method, is introduced after SVM because of its ability to handle non-linearity with even greater finesse through the use of multiple decision trees. This model can capture complex interaction structures in the data that neither the logistic regression nor the SVM can easily uncover. It's also robust to overfitting and can handle unbalanced datasets well, providing importance scores for each feature based on how much they contribute to reducing the variance in the model.

Similar to the process of fine-tuning SVM, we conducted a grid search to explore a range of hyperparameters for Random Forests. The parameters included the max_depth, the min_samples_split, min_samples_leaf, max_leaf_nodes, and class_weight, which help to identify which settings improve model performance the most, taking into account the trade-off between bias and variance (underfitting vs. overfitting).

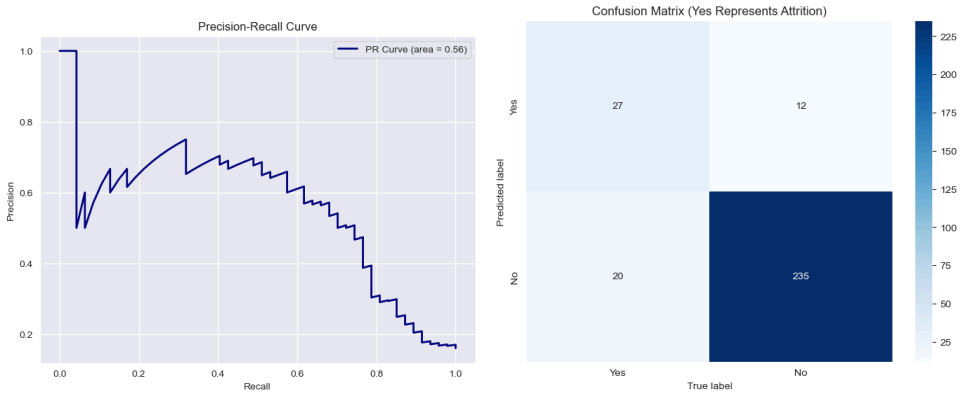
The optimal hyperparameters identified were:

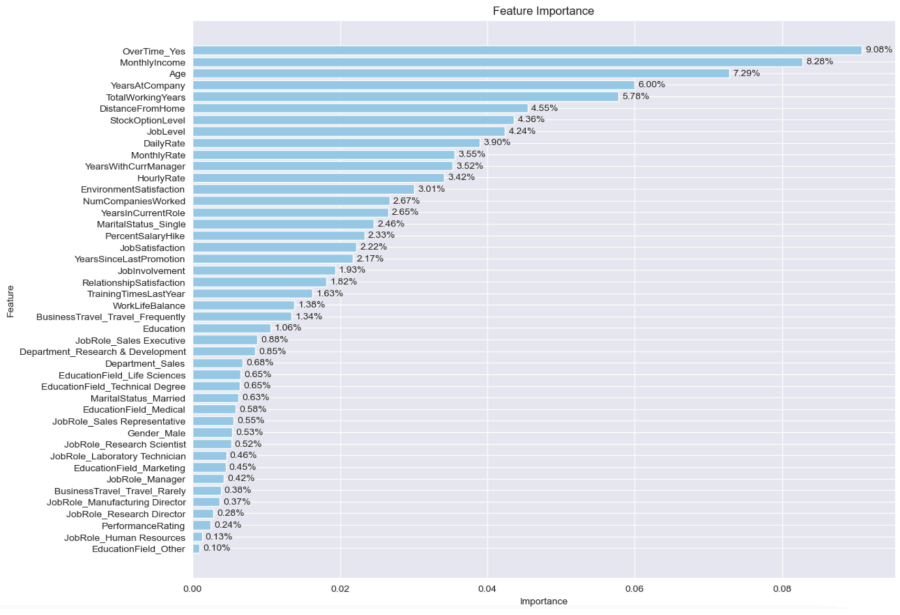
| Parameters | Value | Definition |
|---|---|---|
| max_depth | 27 | A range from 5 to 30 is provided, allowing trees to have a varied depth. |
| min_samples_split | 30 | Ranging from 2 to 99, this hyperparameter defines the minimum number of samples that are required in a node to consider it for splitting. |
| min_samples_leaf | 4 | This hyperparameter sets the minimum number of samples that must be present in a leaf node after splitting a node, ranging from 2 to 99. |
| max_leaf_nodes | 93 | This defines the maximum number of leaf nodes a tree can have. A range from 2 to 100 is given, allowing the trees to vary from being very simple (few leaf nodes) to more complex (many leaf nodes). |
| class_weights | balanced | This parameter is used to weigh the classes either as `"balanced"` or `None`. |

The learning curve indicates the random forest model is improving in its ability to generalize as more data is provided, though there is a slight overfitting since the training score is consistently higher than the validation score. The second graph shows high variability in the model's performance across different cross-validation folds, with a notably poor performance in fold 4, which could indicate issues with the model's generalization to all subsets of data or potential data-specific issues in that fold.




The PR graph shows a model with an area under the curve (AUC) of 0.57. The curve starts with high precision at low recall levels, which means the model is very precise when it predicts a positive class but only for a small fraction of all positive instances. As the recall increases (meaning the model tries to capture more positive instances), the precision drops significantly, which is typical as it becomes more challenging to maintain high precision with higher recall.The confusion matrix demonstrates the model has a precision of 0.69, indicating it correctly predicts attrition about 69% of the time when it does predict it. The recall for this class is 0.57, meaning the model identifies 57% of actual attrition cases. The F1-score is 0.63, showing a moderate balance between precision and recall for the Attrition Class. The model is better at predicting non-attrition than attrition.



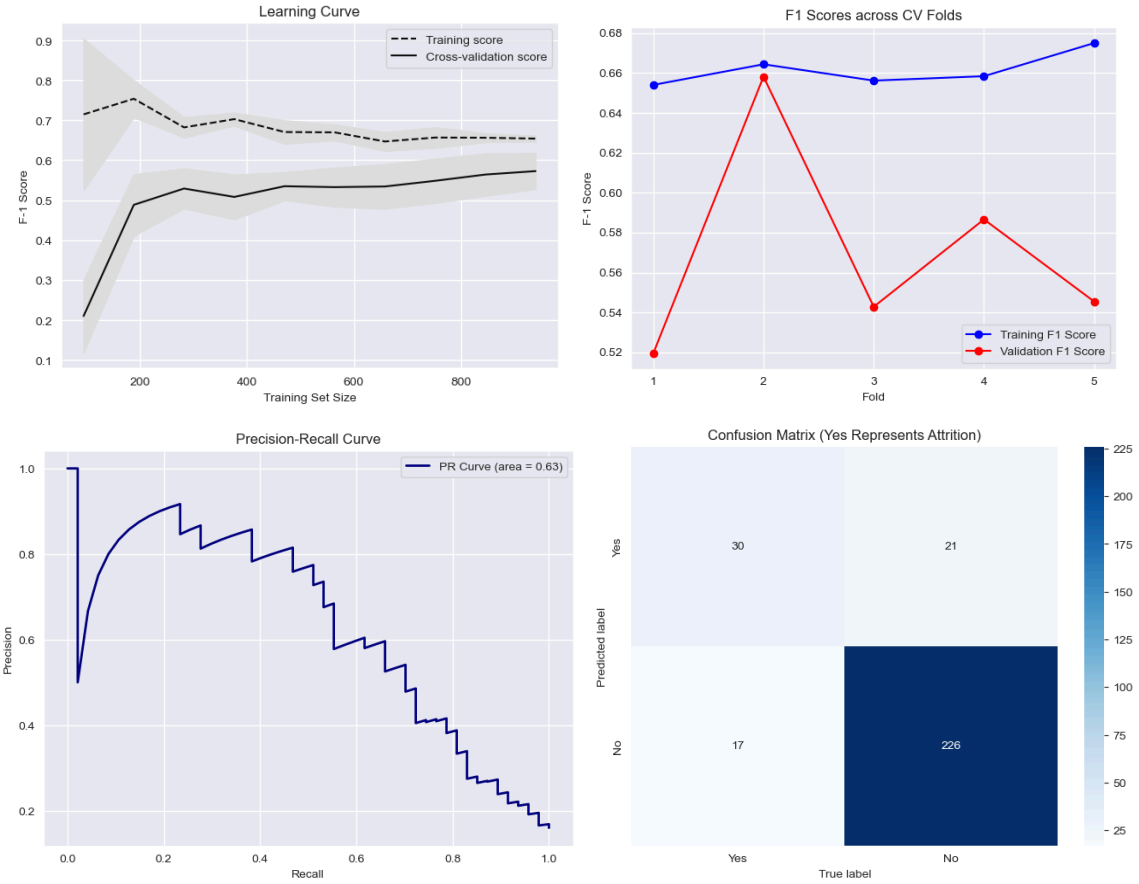

Feature importance can provide domain insights by highlighting which factors are most influential in the outcome, which can be valuable for business stakeholders to make informed decisions. The features OverTime_Yes,

MonthlyIncome, Age, YearsAtCompany, and TotalWorkingYears are the most influential (over 5% importance) in predicting employee attrition. This suggests that factors like working overtime, income, age, tenure, and overall working experience are critical in understanding why employees might leave the company. OverTime_Yes has an importance of around 9.08%, making it the most influential feature in this model, suggesting that employees who frequently work overtime are at a higher risk of leaving. Features like DistanceFromHome, StockOptionLevel, and JobLevel have moderate importance, suggesting they do have an impact on attrition, but not as strongly as the top features. Features at the bottom of the plot, such as EducationField_Other and JobRole_HumanResources, have very low importance scores, implying that they have minimal impact on the model's predictions.

**Ensemble Model**

Finally, an ensemble model, which combines predictions from multiple models, is considered the most sophisticated approach in our progression. An ensemble can leverage the strengths of all the previous models while compensating for their individual weaknesses. The idea behind the Ensemble Model is that the collective knowledge of the crowd will yield better results than the individual models. While Ensemble Models can be trained on samples of the same dataset with replacement, for this analysis, we combined the various models above and utilized them to train our Ensemble Model.

| Weight of Logistic Regression | Weight of SVM | Weight of Random Forest |
|---|---|---|
| 0.33 | 0.33 | 0.34 |
| 0.25 | 0.25 | 0.5 |
| 0.5 | 0.25 | 0.25 |
| 0.25 | 0.5 | 0.25 |




The provided graphs show a model whose performance stabilizes with increased training data but does not significantly improve after a certain size, suggesting a plateau in learning. Cross-validation scores indicate good

generalization, although a variance in F1 scores across folds points to data-specific sensitivity. The precision-recall curve with an AUC of 0.63 demonstrates a moderate balance between precision and recall. The ensemble model outperforms other models like logistic regression, SVM, and Random Forest in F1 score for predicting attrition, with a higher F1 score of 0.61 on the test set implies effective generalization and robustness, making the model a reliable predictive tool for attrition.

## Results and Conclusions

The table presents a comparison of four different machine learning approaches—Logistic Regression, Support Vector Machine (SVM), Random Forest, and an Ensemble Method—used for predicting employee attrition. The evaluation metrics used for comparison are Test Accuracy, Precision, Recall, F-1 Score, and Precision-Recall Area Under Curve (PR-AUC). The Random Forest model exhibits the highest Test Accuracy and F-1 score at 0.89, but its Precision and Recall are not the highest among the models. The Ensemble Method, while having a slightly lower Test Accuracy of 0.87, presents a better balance with an F-1 Score of 0.61 and a Recall score of 0.57, and the highest PR-AUC at 0.63. The F-1 Score and PR-AUC is particularly relevant for this application due to the imbalanced nature of the dataset.

| Approach | Test Accuracy | Precision | Recall | F-1 Score | PR_AUC |
|---|---|---|---|---|---|
| Logistic Regression | 0.78 | 0.40 | 0.77 | 0.52 | 0.62 |
| SVM | 0.78 | 0.39 | 0.70 | 0.50 | 0.61 |
| Random Forest | 0.89 | 0.69 | 0.57 | 0.63 | 0.57 |
| Ensemble Method | 0.87 | 0.59 | 0.64 | 0.61 | 0.63 |

In our pursuit to tackle the substantial 15% attrition rate, the Ensemble Method has been identified as the most effective predictive model due to its optimal balance between precision and recall, reflected by its F-1 Score and PR-AUC. This balance is of paramount importance as it allows us to accurately identify employees who are likely to leave (recall) while keeping false positives to a minimum (precision). The latter is particularly critical as it ensures that the human resources team's efforts are concentrated and that company resources are not misused on incorrect predictions. With the Ensemble Method's recall rate of 0.64, we have the potential to identify and thus retain 15% of our workforce that would otherwise have been lost to attrition, effectively reducing our overall attrition rate from 14% to 5.04%.

To complement the predictive prowess of the Ensemble Method, insights from the Random Forest model’s feature importance analysis are instrumental. It highlights OverTime_Yes, MonthlyIncome, Age, YearsAtCompany, and TotalWorkingYears as significant contributors to attrition, each with more than 5% importance. By instituting targeted interventions that address these areas—such as managing overtime demands, offering competitive compensation, and providing tailored career development opportunities—we can proactively mitigate the risk factors leading to employee turnover. Strategically focusing on these elements will not only cut down the current attrition rate but also cultivate a work environment that supports employee well-being and loyalty, further stabilizing our workforce.

**References**

Attrition Rate Dataset -
https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset/data

Arresting and Reducing Sky-High Attrition Rates in Tech -
https://draup.com/talent/blog/arresting-and-reducing-sky-high-attrition-rates-in-tech/

Employee Attrition Rates Calculation: A Quick Guide for 2024 -
https://technologyadvice.com/blog/human-resources/attrition-rate/#:~:text=A%20high%20attrition%20rate%20indicates,as%20a%20high%20attrition%20rate.

Five Hidden Costs of Employee Attrition -
https://www.forbes.com/sites/forbeseq/2023/03/21/five-hidden-costs-of-employee-attrition/?sh=5639858b62f4

Accuracy, Precision, Recall or F1?
https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9

Handling Missing Data: Mean, Median, Mode
https://www.shiksha.com/online-courses/articles/handling-missing-data-mean-median-mode/

Learning Curve
https://www.scikit-yb.org/en/latest/api/model_selection/learning_curve.html

Best Practices for Feature Importance
https://towardsdatascience.com/best-practice-to-calculate-and-interpret-model-feature-importance-14f0e11ee660

Attrition Rates: How to Calculate and Analyze the Key HR Metric -
https://www.aihr.com/blog/attrition-rate/

# Appendix

| Column Name | Data Type | Sample Value | Column Description |
|---|---|---|---|
| Age | Numerical | 41 | Age of employee at time = current |
| Attrition | Binary | Yes | Whether Attrition or not |
| BusinessTravel | Categorical | Travel_Rarely | Travel frequency for the job |
| DailyRate | Numbercial | 1102 | |
| Department | Categorical | Sales | Department the employee is working at |
| DistanceFromHome | Numerical | 24 | Distance from home to work |
| Education | Categorical | 1 | Education level of the employee |
| EducationField | Categorical | Medical | Education field of the employee |
| EmployeeCount | Numerical | 1 | 1 for each |
| EmployeeNumber | Numerical | 1 | Employee ID (unique identifier) |
| EnvironmentSatisfaction | Numerical | 2 | Environment Satisfaction Level (ordinal) |
| Gender | Binary | Female | Gender of the employee |
| HourlyRate | Numerical | 82 | |
| JobInvolvement | Categorical | 3 | 1-4 (ordinal variable - higher = more involvement) |
| JobLevel | Categorical | 2 | 1-5 (ordinal variable - higher = more level) |
| JobRole | Categorical | Sales Executive | Job position of the employee |
| JobSatisfaction | Categorical | 3 | 1-4 (ordinal variable - higher the better) |
| MaritalStatus | Categorical | Single | Marital Status of the employee |
| MonthlyIncome | Numerical | 2670 | Income per month |
| MonthlyRate | Numerical | 12847 | |
| NumCompaniesWorked | Numerical | 4 | Number of companies have worked before |
| Over18 | Binary | Y | Over 18 years old (binary yes or no) |
| OverTime | Binary | Yes | Overtime compensation available (binary) |
| PercentageSalaryHike | Numerical | 12 | Salary hike for the employee in % |
| PerformanceRating | Categorical | 3 | Score in scale of 3-4 |
| RelationSatifaction | Categorical | 4 | Score in scale of 1-4 |
| StandardHours | Numerical | 80 | Standard working hours per week |
| StockOptionLevel | Categorical | 1 | Option in scale of 0-3 |
| TotalWorkingYears | Numerical | 2 | Year In scale of 0-6 |
| TraningTimesLastYear | Numerical | 5 | Times in scale of 1-4 |
| WorkLifeBalance | Numerical | 3 | 1-4 (ordinal - higher = more balance) |
| YearsAtCompany | Numerical | 1 | How many years has the employee been working at the company? |
| YearsInCurrentRole | Numerical | 1 | How many years has the employee been working in this position? |
| YearsSinceLastPromotion | Numerical | 0 | How many years has the employee been working since last promotion? |
| YearsWithCurrManager | Numerical | 0 | How many years has the employee been working with the current manager? |

Figure 1: Data Description Columns.