Contents

[1.0. Abstract 4](#_Toc202217381)

[2.0. Introduction 4](#_Toc202217382)

[2.2. Literature Review 5](#_Toc202217383)

[2.2.1 Machine Learning Techniques and Performance in Attrition Prediction 5](#_Toc202217384)

[2.3. Exploratory Data Analysis 7](#_Toc202217385)

[2.3.1. Univariate Analysis 7](#_Toc202217386)

[2.3.2. Bivariate Analysis 9](#_Toc202217387)

[2.3.2.3. Numerical to Numerical Features Correlation 12](#_Toc202217388)

[2.3. Research Objective 15](#_Toc202217389)

[3.0 Data Mining Algorithm 15](#_Toc202217390)

[3.1 Data Preprocessing 15](#_Toc202217391)

[3.2 Data Mining Model Selection 18](#_Toc202217392)

[3.2.1 Related Work 19](#_Toc202217393)

[3.2.2 Selected Model for Data Mining 21](#_Toc202217394)

[3.3 Data Mining Modeling 21](#_Toc202217395)

[3.3.1 Model Validation Technique 21](#_Toc202217396)

[3.3.2 Handling Class Imbalance 22](#_Toc202217397)

[3.3.3. Classification Models and Parameter Tuning 24](#_Toc202217398)

[3.3.4 Data Mining Modeling and Benchmarking 27](#_Toc202217399)

[4.0 Results 28](#_Toc202217400)

[5.0 Conclusion 34](#_Toc202217401)

[6.0. REFERENCES 35](#_Toc202217402)

Predicting Employee Attrition Using Data Mining Models in WEKA

# Abstract

This study explores the use of data mining techniques to predict employee attrition using a IBM HR dataset. This is a crucial task for organizations seeking to reduce employee turnover and retain the best top talents among their employees. The dataset, sourced from Kaggle, contains 35 attributes that represent employee demographics, satisfaction levels, job roles etc. After data cleaning, ensuring no outliers and duplicates and feature engineering, a total of 13 high-impact predictors were selected.

Nine classification models were trained and evaluated using WEKA: Logistic Regression, Decision Tree (J48), Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Neural Network (MLP), Naïve Bayes, Bagging, and AdaBoostM1. Hyperparameter tuning was performed for each model to optimize performance.

The models were then evaluated using 10-fold cross-validation on a balanced dataset. After evaluating based on accuracy, RMSE, ROC area and training speed, the top three models identified were Logistic Regression (accuracy 73.5%), Support Vector Machine (73.8%), and AdaBoostM1 with SVM as the base classifier (73.8%). These findings demonstrate that machine learning methods, particularly ensemble methods, offer a robust performance in employee attrition prediction. These results provide valuable insights for HR departments to manage workforce stability proactively.

# 2.0. Introduction

Employee attrition, defined as the progressive drop in the number of employees without replacement, is a significant concern for organizations today. It leads to a loss of talent, poor research, resources spent on training and can drain profitability. High attrition rates cause substantial costs, impacting overall efficiency, market value, and disrupting workflow, potentially leading to employee burnout and demoralizing remaining staff. Replacing trained and experienced employees is a complex and costly task, estimated at approximately 1.5 times the annual salary of the employee who left. Consequently, organizations are increasingly turning to machine learning (ML) and data mining techniques to predict attrition and understand its underlying causes, allowing for proactive human resource (HR) management and retention strategies. HR departments generate enormous amounts of data daily, including leaves, social conflicts, annual evaluations, wages, benefits, recruitments, and career evaluations. Hence, This study explores the use of data mining techniques to predict employee attrition using a IBM HR dataset.

## Literature Review

### 2.2.1 Machine Learning Techniques and Performance in Attrition Prediction

Various machine learning algorithms have been applied to predict employee attrition, with differing levels of success across studies and datasets.

#### Multi-Layer Perceptron (MLP) / Neural Networks

R. Shiva Shankar et al. (2021) found that MLP outperformed other models (Random Forest Classifier, TabNet, Gradient Boosting Classifier) in their study, achieving an accuracy of 0.88. Lok Sundar Ganthi et al. (2021) also utilized Neural Networks, including Multi-Layer Perceptron, achieving 88% accuracy with good precision and recall. Norsuhada Mansor et al. (2021) initially found ANN (MLP) to have the highest accuracy (86.76%) and best RMSE (0.3359) among DT, SVM, and ANN models. De Rovere (2023) also found Neural Networks to excel in sensitivity.

#### Support Vector Machines (SVM)

Norsuhada Mansor et al. (2021) conducted a comparative study using the IBM Human Resource Analytic Employee Attrition and Performance dataset. Their optimized SVM model achieved the highest accuracy of 88.87%, surpassing Artificial Neural Networks (ANN) and Decision Trees (DT) after parameter tuning and regularization. This performance was achieved with the Pearson VII kernel (PUK) set at a regularization parameter (C) of 10. Similarly, Fatma Ozdemir et al. (2020) indicated that SVM was a recommended technique for assessing employee attrition. Tanya Attri (2018) also utilized SVM with Simulated Annealing (SA) feature selection, achieving an overall accuracy of 53.17% but a significant sensitivity (true positive accuracy) of 80.43%, emphasizing its capability to identify employees highly likely to leave. SVM was also mentioned as a classification method applied to HR data for attrition prediction. Liang and Kang (2024) introduced a Test-Time Layer to enhance SVM classifiers.

#### Ensemble Methods (Boosting, Bagging)

Divyang Jain (2017) explored a hybrid model of ensemble methods (stacking, bagging, and boosting). Their Adaptive Boosting (ADA) model achieved 88.8% accuracy, along with 85.93% sensitivity and 90.68% specificity, outperforming other ensemble techniques like Random Forest and GBM, and traditional classification models like SVM, GLM, Decision Trees, and KNN. Roopak Krishna and Neeta Singh (2023) applied Bagging Classifier and Random Forest, among others, to the IBM HR dataset, with Bagging Classifier achieving 86.62% accuracy. Ensemble models like Random Forest and XG Boost have also shown consistent performance across diverse datasets, with recommendations for using multiple models and performance metrics to improve reliability. De Rovere (2023) found XG Boost produced the highest accuracy, while Random Forest excelled in precision.

#### Logistic Regression

While Roopak Krishna and Neeta Singh (2023) identified Logistic Regression as their best performer, achieving 88.44% accuracy, Sanidhya Barara and Umang Soni (2023) noted that Logistic Regression was unable to predict attrition very satisfactorily in their study. Benson and Rajeev (2024) applied Logistic Regression to IBM HR Data, achieving 87% accuracy, emphasizing its interpretability and relevance in HR environments. Logistic Regression was also applied in a study on HR data for attrition prediction. Zarmina Jaffar et al. (2023) found Logistic Regression accuracy to be 79.23% with 10-fold cross-validation.

#### J48 (Decision Tree)

Zarmina Jaffar et al. (2023) reported that J48 performed exceptionally well, achieving an accuracy of 98.84% (training set) with a True Positive rate of 0.984%, making it the best among classifiers tested in their study. Karpagam (2019) also observed J48 showing better accuracy (around 82.4-82.7%) compared to Naïve Bayes but taking more time. Norsuhada Mansor et al. (2021) also included Decision Tree (J48) in their comparative study. J48 was applied in a study on HR data for attrition prediction.

## **2.3. Exploratory Data Analysis**

The dataset utilized in this EDA analysis consisted of a mixture of numerical and categorical variables, which together will represent various aspects of employee demographics, job satisfaction, and organizational characteristics. The charts and table below are a breakdown of the features involved in the analysis of both Univariate and Bivariate Analysis.

### **2.3.1. Univariate Analysis**

#### 2.3.1.2. Histograms of the Numerical Variables

This figure 0-1 presents a comprehensive set of histograms illustrating the distribution of various numerical variables within the dataset. Most variables show non-normal distributions, with several being skewed either to the left or right, such as MonthlyIncome, YearsSinceLastPromotion, and DistanceFromHome. Categorical numeric features like JobInvolvement, Education, and EnvironmentSatisfaction display distinct bars indicating discrete levels. A few variables such as DailyRate and HourlyRate appear relatively uniformly distributed, while others like PerformanceRating and StockOptionLevel are heavily concentrated in one or two categories. These visualizations provide valuable insights into data variability, common value ranges, and potential outliers—crucial for informing subsequent analysis steps like normalization or feature engineering.

A collage of graphs

AI-generated content may be incorrect.

Figure ‑ Histograms of Numerical Features

#### 2.3.1.1. Table showing Descriptive Statistics of Numerical Features

Table below presents the key descriptive statistics for the numerical variables in the dataset. It includes the sum, mean, minimum, maximum, median, and standard deviation for each feature. These metrics help to understand the central tendency, dispersion, and overall distribution of values across employee records.

A screenshot of a data

AI-generated content may be incorrect.

Table 1: Table of Numerical Features Univariate Analysis

### **2.3.2. Bivariate Analysis**

#### 2.3.2.1. Table of Numerical Feature F score and P Value (Against Attrition)

This table summarizes the statistical relationship between each numerical feature and employee attrition using F-score and p-value metrics. Higher F-scores indicate greater variance between attrition groups for a given feature, suggesting stronger discriminatory power. P-values assess the statistical significance of these differences; features with p-values below 0.05 are considered significantly associated with attrition. This analysis helps identify which numerical attributes may contribute to attrition prediction and should be prioritized in modeling efforts.

|  |  |  |  |
| --- | --- | --- | --- |
| **Rank** | **Feature** | **F Score** | **P-Value** |
| 1 | AverageYearsPerCompany | 59.571218 | 2.49E-14 |
| 2 | TotalWorkingYears | 42.982532 | 8.22E-11 |
| 3 | YearsInCurrentRole | 36.302996 | 2.25E-09 |
| 4 | YearsWithCurrManager | 33.428836 | 9.45E-09 |
| 5 | YearsAtCompany | 33.257373 | 1.03E-08 |
| 6 | JobLevel | 32.950945 | 1.20E-08 |
| 7 | Age | 32.946505 | 1.20E-08 |
| 8 | WorkSatisfactionAvg | 30.253155 | 4.64E-08 |
| 9 | MonthlyIncome | 29.131169 | 8.16E-08 |
| 10 | JobInvolvement | 20.607273 | 6.21E-06 |
| 11 | StockOptionLevel | 16.543765 | 5.07E-05 |
| 12 | EnvironmentSatisfaction | 15.807592 | 7.44E-05 |
| 13 | MonthlyIncomePerLevel | 14.018546 | 1.90E-04 |
| 14 | JobSatisfaction | 8.050683 | 4.63E-03 |
| 15 | WorkLifeBalance | 6.597056 | 1.03E-02 |
| 16 | DistanceFromHome | 4.896084 | 2.71E-02 |
| 17 | RelationshipSatisfaction | 3.070167 | 8.00E-02 |
| 18 | DailyRate | 3.021635 | 8.24E-02 |
| 19 | NumCompaniesWorked | 2.958283 | 8.57E-02 |
| 20 | TrainingTimesLastYear | 2.442962 | 1.18E-01 |
| 21 | Education | 1.994485 | 1.58E-01 |
| 22 | PercentSalaryHike | 1.350215 | 2.45E-01 |
| 23 | YearsSinceLastPromotion | 0.975353 | 3.24E-01 |
| 24 | MonthlyRate | 0.51463 | 4.73E-01 |
| 25 | HourlyRate | 0.009367 | 9.23E-01 |
| 26 | PerformanceRating | NaN | NaN |

Table : Numerical Feature F score and P Value (Against Attrition)

#### 2.3.2.2.Table of Categorical Feature Chi Score and P Value (Against Attrition)

|  |  |  |  |
| --- | --- | --- | --- |
| **Rank** | **Feature** | **Chi² Score** | **P-Value** |
|  |  |  |  |
| 1 | OverTime | 51.820487 | 6.08E-13 |
| 2 | MaritalStatus | 10.951433 | 9.35E-04 |
| 3 | JobRole | 2.969333 | 8.49E-02 |
| 4 | Department | 1.399218 | 2.37E-01 |
| 5 | Gender | 1.00141 | 3.17E-01 |
| 6 | EducationField | 0.030552 | 8.61E-01 |
| 7 | BusinessTravel | 0.022307 | 8.81E-01 |

Table : Categorical Feature Chi Score and P Value (Against Attrition)

#### **2.3.2.3. Numerical to Numerical Features Correlation**

This heatmap illustrates the Pearson correlation coefficients between all numerical variables in the dataset. Strong positive correlations are observed between JobLevel and MonthlyIncome (0.94), and between TotalWorkingYears and both Age (0.66) and MonthlyIncome (0.75), indicating potential multicollinearity. Most other variable pairs exhibit weak correlations, suggesting limited linear relationships. The symmetric matrix helps identify redundant features and informs feature selection or dimensionality reduction during modeling.

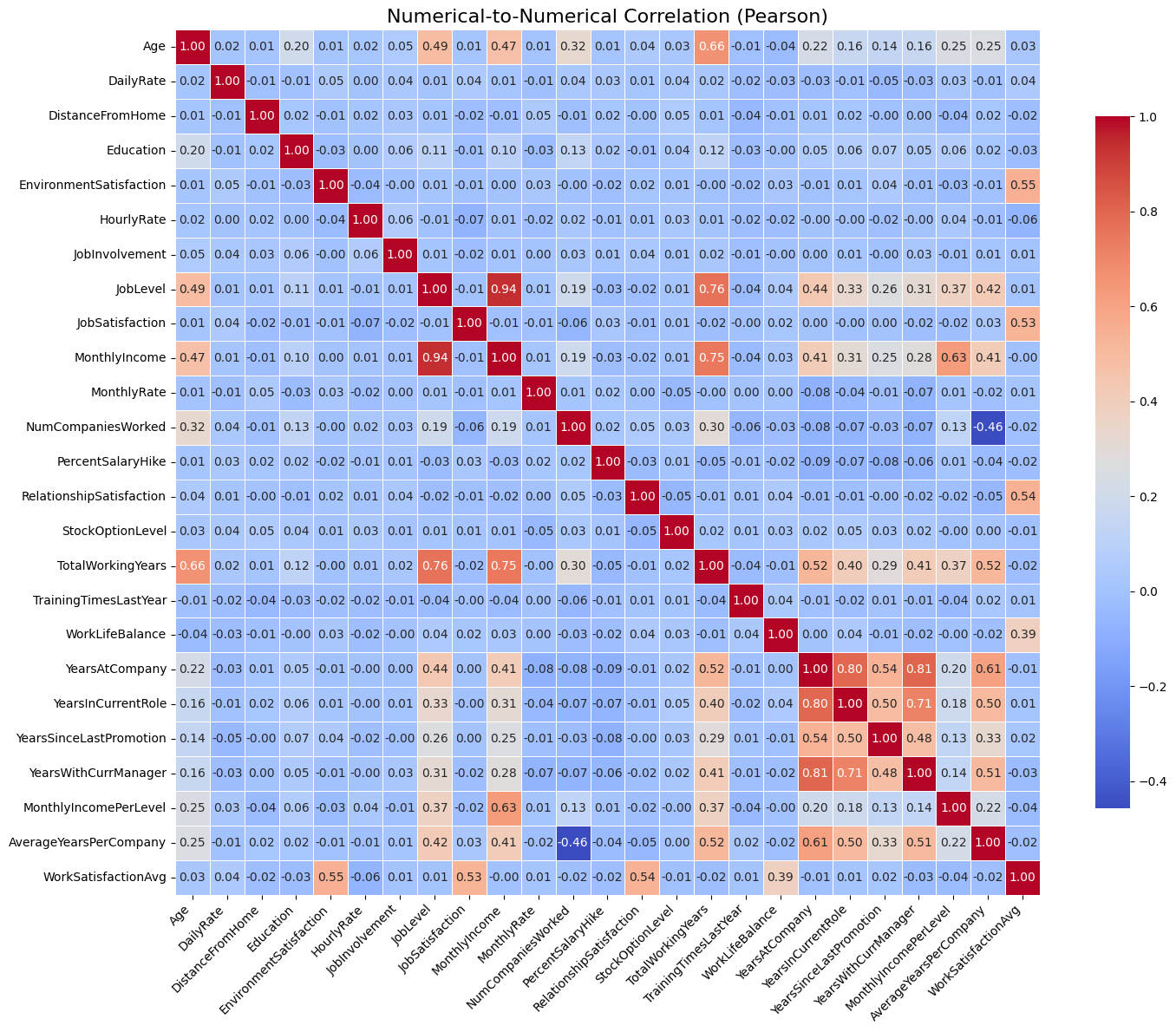


Figure ‑ Numerical – To- Numerical Corelation

#### **2.3.2.4. Categorical to Categorical Features Correlation**

This heatmap visualizes the Cramér’s V correlation between categorical variables, indicating the strength of association between each pair. Most categorical features show weak or no association (values close to 0), suggesting minimal redundancy. Notably, strong correlations exist between Department and JobRole (0.95), and between Department and EducationField (0.59), implying overlap in category distributions. These insights help in identifying potential multicollinearity and guide feature selection for modeling categorical data.

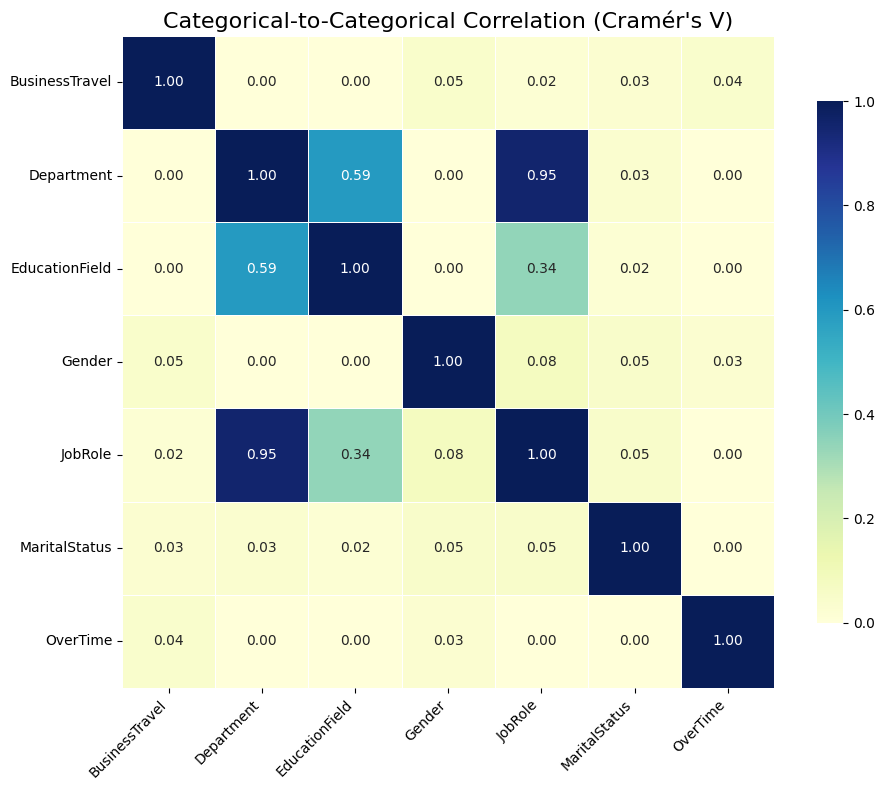


Figure ‑: Categorical to Categorical Corelation

#### 2.3.2.5. Correlation, Gain Ratio, Sym. Uncertainty Ranking

This table presents a comparative ranking of features based on three different importance metrics: Pearson correlation, Gain Ratio, and Symmetrical Uncertainty, all in relation to the target variable Attrition. AverageYearsPerCompany, TotalWorkingYears, and YearsInCurrentRole rank highest by correlation, indicating strong linear relationships. In contrast, MonthlyRate and MonthlyIncome rank highest under Gain Ratio and Symmetrical Uncertainty, highlighting their informativeness and relevance to classification tasks. The varying rankings across methods underscore the importance of using multiple metrics to capture different aspects of feature significance : linear, information-based, and symmetrical dependencies.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Correlation Rank | Correlation Attribute | Correlation Score | Gain Ratio Rank | Gain Ratio Attribute | Gain Ratio Score | Sym. Uncertainty Rank | Sym. Uncertainty Attribute | Sym. Uncertainty Score |
| 1 | AverageYearsPerCompany | 0.2188 | 1 | MonthlyRate | 0.0418 | 1 | MonthlyRate | 0.0786 |
| 2 | TotalWorkingYears | 0.1871 | 2 | MonthlyIncome | 0.0413 | 2 | MonthlyIncome | 0.0776 |
| 3 | YearsInCurrentRole | 0.1724 | 3 | MonthlyIncomePerLevel | 0.0399 | 3 | MonthlyIncomePerLevel | 0.0751 |
| 4 | YearsWithCurrManager | 0.1656 | 4 | DailyRate | 0.033 | 4 | DailyRate | 0.0618 |
| 5 | YearsAtCompany | 0.1652 | 5 | OverTime | 0.0323 | 5 | OverTime | 0.037 |
| 6 | JobLevel | 0.1645 | 6 | AverageYearsPerCompany | 0.0164 | 6 | AverageYearsPerCompany | 0.0296 |
| 7 | Age | 0.1645 | 7 | JobLevel | 0.012 | 7 | JobRole | 0.018 |
| 8 | WorkSatisfactionAvg | 0.1578 | 8 | JobRole | 0.011 | 8 | JobLevel | 0.0179 |
| 9 | MonthlyIncome | 0.1549 | 9 | StockOptionLevel | 0.0099 | 9 | TotalWorkingYears | 0.0151 |
| 10 | JobInvolvement | 0.1307 | 10 | TotalWorkingYears | 0.0086 | 10 | Age | 0.0149 |
| 11 | StockOptionLevel | 0.1173 | 11 | Age | 0.0084 | 11 | StockOptionLevel | 0.0142 |
| 12 | EnvironmentSatisfaction | 0.1147 | 12 | BusinessTravel | 0.0083 | 12 | YearsInCurrentRole | 0.0133 |
| 13 | MonthlyIncomePerLevel | 0.1081 | 13 | YearsInCurrentRole | 0.008 | 13 | YearsAtCompany | 0.0132 |
| 14 | JobSatisfaction | 0.0821 | 14 | YearsWithCurrManager | 0.0079 | 14 | YearsWithCurrManager | 0.0132 |
| 15 | WorkLifeBalance | 0.0744 | 15 | YearsAtCompany | 0.0077 | 15 | HourlyRate | 0.0126 |

Table : Correlation, Gain Ratio, Sym. Uncertainty Ranking

## Research Objective

The Objective of this study is to explore the use of data mining techniques to predict employee attrition using an IBM HR dataset. The study aims to:

* Find the best three models to be used for predicting employee attrition, determining whether they will leave ("Yes") or stay ("No").
* Address this prediction as a binary classification problem, where the output consists of two distinct classes.

The overarching goal is to assist organizations in proactively managing human resources, reducing employee turnover, and retaining top talent.

## 3.0 Data Mining Algorithm

## 3.1 Data Preprocessing

Data preprocessing is a crucial step to be taken before applying any data mining modeling techniques to the dataset. Mansor et al. (2021) stated that this process greatly improves the quality of the dataset by ensuring accuracy, completeness, and consistency. The three key steps in data preprocessing are data cleaning, data transformation, and data reduction.

1. Data Cleaning

In this data cleaning step, a few actions were taken to clean the dataset:

* Handling missing value: All missing values in the dataset will be replaced by a mean value. However, there are no missing values identified from the dataset, indicating the dataset was free from missing values.
* Removing duplicate rows and columns: All duplicated instances and attributes will be removed to avoid redundancy in the model. However, there are no duplicated instances and attributes identified from the dataset, indicating the dataset was free from duplicated instances and attributes.
* Removing useless columns: Non-informative attributes in the dataset will be removed.
* Removing columns with constant values: All attributes with no variance will be removed from the dataset as they do not contribute to prediction.
* Handling inconsistent formatting: All inconsistent text labels such as case sensitivity or abbreviations will be standardized across the dataset. However, no inconsistent labels were found, indicating all values in attributes are well standardized.
* Outlier detection and removal: The Interquartile Range filter in Weka was used to remove extreme outliers from the dataset.

Data cleaning of the dataset was conducted using both Google Colab and Weka. Tasks such as handling missing value, removing duplicate rows and columns, removing useless columns, removing columns with constant values, and handling inconsistent formatting were performed in Google colab. On the other hand, outlier detection and removal were performed in Weka. As a result of the cleaning process, four attributes were removed: one non-informative attribute (EmployeeNumber) and three constant-value attributes (EmployeeCount, Over18, and StandardHours). Additionally, 283 extreme outlier instances were removed from the dataset. These actions reduced the dataset size from (1470, 35) down to (1187, 31).

1. Data Transformation

Data transformation was applied to convert data into a more meaningful format that is more suitable for data mining and modeling. Two actions that are used to transform data are feature construction and normalization.

* Feature Construction:

Vouk et al. (2023) suggest that feature construction is one of the most important action in data transformation to boost classification accuracy and model comprehensibility. Three new attributes were constructed from existing attributes, with the calculations shown in Table 5 below. The first, MonthlyIncomePerLevel, was used to gain insight on pay imbalance in different employee’s job levels. Secondly, AverageYearsPerCompany, was used to provide insight into employee loyalty or job-hopping behavior. Lastly, WorkSatisfactionAvg, was used to reflect overall workplace satisfaction. Exploratory Data Analysis (EDA) proved that these newly created features showed strong correlations with the target variable, indicating effective feature construction.

|  |  |
| --- | --- |
| **Attribute** | **Calculation** |
| 1. MonthlyIncomePerLevel |  |
| 1. AverageYearsPerCompany |  |
| 1. WorkSatisfactionAvg | Where:  ES - EnvironmentSatisfaction  JS - JobSatisfaction  RS - RelationshipSatisfaction  WLB - WorkLifeBalance |

Table 5: New Features Created

* Normalization:

Normalization was performed after data reduction to only normalize attributes that are being considered to be used for data mining and modeling.

All numerical features were normalized to a common scale of 0 to 1 to improve comparability and enhance model convergence. Ordinal features such as JobInvolvement, JobLevel, StockOptionLevel, and WorkSatisfactionAvg were ordinal encoded to reflect their ranked nature, followed by normalization. Categorical variables like OverTime and Attrition were preserved in their nominal form for compatibility with classifiers.

1. Data Reduction

Yahia et al. (2021) suggest that data reduction was used to minimize the dimension of the dataset and retain only highly correlated and informative attributes, improving computational efficiency and model performance. In this study, data reduction was driven by both exploratory data analysis (EDA) and correlation, gain Ratio, symmetrical uncertainty ranking.

Attribute selected as high predictive feature must meet the following criteria:

* For numerical attribute, ANOVA F-test scores greater than 10
* For categorical attribute, Chi-square test scores greater than 10
* Top 10 attributes from correlation, gain ratio, and symmetrical uncertainty rankings

After evaluating all available attributes, 13 attributes were selected for modeling shown in Table 6 below.

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Attribute** | **ANOVA F-test scores** | **Chi-square test scores** |
| 1 | AverageYearsPerCompany | 59.571218 | - |
| 2 | TotalWorkingYears | 42.982532 | - |
| 3 | YearsInCurrentRole | 36.302996 | - |
| 4 | YearsWithCurrManager | 33.428836 | - |
| 5 | YearsAtCompany | 33.257373 | - |
| 6 | JobLevel | 32.950945 | - |
| 7 | Age | 32.946505 | - |
| 8 | WorkSatisfactionAvg | 30.253155 | - |
| 9 | MonthlyIncome | 29.131169 | - |
| 10 | JobInvolvement | 20.607273 | - |
| 11 | StockOptionLevel | 16.543765 | - |
| 12 | MonthlyIncomePerLevel | 14.018546 | - |
| 13 | OverTime | - | 51.820487 |

Table ; High Predictive Attributes Selected for Modeling

## 3.2 Data Mining Model Selection

In this study, employee-related attributes were chosen with the objective of predicting employee attrition status, whether an employee will leave ("Yes") or stay ("No"). This prediction task falls under the category of binary classification, where the output of the prediction consists of only two distinct classes. As a result, data mining models selected must be well-suited for solving binary classification problems, ensuring accurate differentiation between employees who stay and those who are likely to leave.

### 3.2.1 Related Work

Table 7 below shows all the literature review findings related to a comparative study on employee attrition using machine learning classification algorithms.

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Author** | **Objective of Study** | **Classification Technique Studied** |
| 1 | Ozdemir et al. (2020) | This paper uses various data mining classification algorithms to predict employee attrition. | Support Vector Machine (SVM), Random Forest, J48, LogitBoost, Multilayer Perceptron (MLP), K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA), Naive Bayes, Bagging, AdaBoost, Logistic Regression |
| 2 | Barara and Soni (2023) | This paper uses various machine learning models to predict employee attrition and the time frame in which an employee is likely to leave. | Logistic Regression, Linear Regression, Decision Tree, Random Forest, KNN, Radius Nearest Neighbors, Naïve Bayes Classifier, Bayesian Ridge |
| 3 | Attri, T. (2018) | Investigate the extent to which ML techniques can help in predicting employee attrition. | Random Forest, Logistic Regression, SVM, Gradient Boosting Machine |
| 4 | Repaso et al. (2022) | To develop a predictive model to identify whether an applicant in the BPO Company is a possible stayer or a hopper type of employee using Naïve Bayes Algorithm. | Naïve Bayes Algorithm |
| 5 | Jaffar, Z., Noor, W., & Kanwal, Z. (2019) | Use data mining to predict employees who are expected to quit/leave an organization. | J48, Naive Bayes, Logistic Regression, Bayesian Network, OneR |
| 6 | PM, U., & Balaji, N. (2019) | To use data mining techniques to understand the factors influencing employee attrition using Weka. | J48, Naive Bayes, K-means, Expectation Maximization (EM) |
| 7 | Shankar et al. (2018) | To prevent employee attrition using classification methods and support economic growth by reducing human resource costs | Decision tree, Logistic Regression, SVM, KNN, Random Forest, Naive Bayes |
| 8 | Yadav et al. (2018) | The aim of this paper is to provide a framework for predicting employee churn by analyzing the employee’s precise behaviors and attributes using classification techniques. | Logistic Regression, SVM, Random Forest, Decision Tree, AdaBoost |
| 9 | Shankar et al. (2021) | To use advanced Machine Learning techniques to solve the attrition problem. | Random Forest, TabNet, Gradient Boosting Classifier, Multi-Layer Perceptron (MLP) |
| 10 | Jain, D. (2017) | Contribute to HR predictive analytics by predicting employee attrition using a hybrid model of ensemble methods. | Stacking, Bagging, Boosting |
| 11 | Ganthi et al. (2021) | Use machine learning techniques to predict employees who are planning to leave the company. | Decision tree, Random Forest, K-Nearest Neighbors, Neural Networks, extreme gradient boosting and Ada-Boosting |
| 12 | Singh (2023) | To compare machine learning models for accurate prediction of employee attrition. | Bagging Classifier, Random Forest, Logistic Regression, K Nearest Neighbor |

Table 7: Related Work on Employee Attrition

### 3.2.2 Selected Model for Data Mining

To ensure the reliability of this study, model selection was guided by their popularity in past research. Specifically, models were considered eligible if they appeared in at least three separate classification studies reviewed in the literature. Based on this criterion, a total of nine classification models that met this requirement were selected for data mining and modeling in this study as shown in Table 8.

|  |  |  |
| --- | --- | --- |
| **No.** | **Selected Classification Model** | **Number of Literature Mentions** |
| 1 | Logistic Regression | 7 |
| 2 | Decision Tree (J48) | 4 |
| 3 | Random Forest | 8 |
| 4 | Support Vector Machine (SVM) | 4 |
| 5 | K-Nearest Neighbors (KNN) | 5 |
| 6 | Neural Networks (MLP) | 3 |
| 7 | Naïve Bayes | 6 |
| 8 | Ensemble Model: Bagging | 3 |
| 9 | Ensemble Model: Boosting (AdaBoostM1) | 6 |

Table 8: Nine Data Mining Models

## 3.3 Data Mining Modeling

Before performing data mining modeling with the selected classification models, several preparatory steps were required, including model validation, handling of class imbalance, and model hyperparameter tuning. These procedures were performed to enhance the validity, reliability, and overall performance of classification models.

### 3.3.1 Model Validation Technique

|  |  |
| --- | --- |
| **Dataset** | **Number of Instances** |
| Training | 1068 |
| Testing | 119 |
| Total | 1187 |

Table 9: Data Splitting with 10-fold Cross-validation

A 10-fold cross-validation is applied to the training dataset to assess the generalization ability of each classifier, preventing overfitting. Overfitting is a serious issue when a data mining model cannot generalize or predict unseen data well. There are different ways to resolve overfitting, K-fold cross-validation is an effective way to prevent it. This was proven by Hoang Lan Vuet al. (2022) who stated that K-fold cross-validation minimizes the risk of overfitting by ensuring that every data point is used for both training and validation. This approach was done by randomly partitioning the dataset into ten equal sized folds. In each training iteration, one fold is used as the test set, while the remaining nine folds are used for training. This will be iterated 10 times, and the results are averaged to produce a more general estimate of model accuracy.

### 3.3.2 Handling Class Imbalance

Class data imbalance is another serious issue in classification problems. As discussed by F. Thabtahet al. (2020), class imbalance will cause classifier bias where data mining algorithms tend to favor the majority class, often misclassifying the minority class. The target variable, Attrition, has an imbalance of class distribution with 192 instances predicted as “Yes” and 995 instances predicted as “No”. This class imbalance in our dataset will reduce the performance of the classification model by poorly predicting the minority class. To mitigate this issue, the Weka ClassBalancer filter was applied to our dataset. This filter resolves the issue by reweighting the instances rather than duplicating or removing data. As a result, this filter improves classifier accuracy to the minority class without changing the distribution in our dataset.

|  |  |
| --- | --- |
| **ClassBalancer filter** | **Output** |
| Before applying |  |
| After applying |  |

Table 10: Class Balancing Using Weka ClassBalancer Filter

### 3.3.3. Classification Models and Parameter Tuning

The next step after model validation and class balancing is to perform hyperparameter tuning for each selected classification model. Hyperparameter tuning is the process of optimizing the settings of a model to improve its performance. Ning Dinget al. (2023) highlights that fine tuning of model parameters significantly reduces computational costs while maintaining or even enhancing model accuracy. Each model was tuned using different parameters, as shown in Table 11 below. For each model, the parameter value that results in highest accuracy will be selected to be used in the next data mining modeling phase.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Model** | **Parameter to Tune** | **Parameter Value** | **Accuracy** | **Recommended Parameter Value** |
| Logistic Regression | Ridge | 1.0E-8 | 0.735 | 1.0E-8 |
| Decision Tree(J48) | Confidence Factor | **0.15** | **0.691** | 0.15 |
| 0.25 | 0.680 |
| 0.35 | 0.674 |
| 0.45 | 0.673 |
| Random Forest | NumIterations- The number of trees in the random forest. | 50 | 0.714 | 150 |
| 100 | 0.724 |
| **150** | **0.732** |
| 200 | 0.726 |
| Support Vector Machine (SVM) | Kernal | **PloyKernel** | **0.738** | PloyKernel |
| RBFKernal | 0.702 |
| Puk | 0.726 |
| K-Nearest Neighbors (KNN) | Number of Neighbors (K) | 1 | 0.637 | 101 |
| 11 | 0.661 |
| **101** | **0.729** |
| 201 | 0.718 |
| Neural Networks (MLP) | Learning Rate | 0.05 | 0.718 | 0.1 |
| **0.1** | **0.721** |
| 0.2 | 0.715 |
| 0.3 | 0.699 |
| Naïve Bayes | Use Kernal Estimator | False | 0.647 | True |
| **True** | **0.687** |
| Bagging | Base Model | REPTree | 0.709 | Random Forest |
| **Random Forest** | **0.742** |
| Logistic Regression | 0.728 |
| SMV | 0.732 |
| AdaBoostM1 | Base Model | Decision Stump | 0.701 | SVM |
| Random Forest | 0.730 |
| Logistic Regression | 0.735 |
| **SMV** | **0.738** |

Table :Hyperparameter Tuning for Each Selected Model

1. Logistic Regression

Based on literature, logistic regression is one of the most commonly used models for employee attrition prediction. Logistic regression is a statistical linear model that is suitable to be used for binary classification problems in this study. It predicts the probability of employee attrition using the sigmoid function with a linear combination of attributes as input.

The hyperparameter to tune in logistic regression is the Ridge value, which is the strength of L2 regularization. However, adjusting the default Ridge value of 1.0E-8 did not result in any improvement of model accuracy, which remained at 0.735. Therefore, the default Ridge value of 1.0E-8 was selected as the optimal value.

1. Decision Tree(J48)

J48 has been selected as the decision tree algorithm used for employee attrition prediction. It is a tree-based model that splits the dataset based on feature values to form decision paths leading to class labels.

The hyperparameter to tune in J48 is the confidence factor, which controls pruning to prevent overfitting. After fine tuning this parameter, a value of 0.15 gave the best accuracy of 0.691 and was selected as the optimal value.

1. Random Forest

Random forest is an ensemble of decision trees that aggregates the predictions of multiple trees to improve model accuracy. Based on literature, this is the most common model used for employee attrition prediction.

The hyperparameter to tune in random forest is the number of trees (NumIteration). A number of 150 trees were proven to be the best performance at 0.732 accuracy and was selected as the optimal value.

1. Support Vector Machine (SVM)

The SVM is a maximum-margin classifier that finds the optimal hyperplane separating attrition classes in high-dimensional space. It transforms data using kernel functions to maximize class separation.

Thus, the hyperparameter to tune in SVM is the Kernal type used. The options provided in Weka for Kernal type used in SVM are PolyKernel, RBF and PUK. PolyKernel was selected as the optimal Kernal type as it yielded the highest accuracy at 0.738, outperforming RBF and PUK kernels.

1. K-Nearest Neighbors (KNN)

KNN is a lazy learning algorithm that classifies new data based on the majority class among its k nearest neighbors in the training set.

The hyperparameter to tune in KNN is the number of neighbors (K) to use. The optimal value selected was 101 neighbors, as it achieved the highest accuracy at 0.729.

1. Neural Networks

For neural networks, the Multilayer Perceptron (MLP) was selected as the algorithm used for employee attrition prediction. It is a feed-forward artificial neural network capable of modeling non-linear relationships between employee-related features and the target variable attrition.

For MLP, the hyperparameter to be tuned is the learning rate used for weight update of the model. The optimized learning rate was obtained at 0.1, with an accuracy of 0.721.

1. Naïve Bayes

The naive bayes is a probabilistic classifier based on Bayes' Theorem. It calculates the posterior probability for each class and chooses the class with the highest probability.

The hyperparameter to be tune for naive bayes is the kernal estimator option. When enabled, it will use a kernel estimator for numeric attributes rather than a normal distribution. Setting it to true increased accuracy from 0.687 to 0.647. As a result, ‘true’ was selected as the optimal configuration.

1. Bagging

Bagging is an ensemble method that reduces variance by averaging predictions from multiple base learners trained on bootstrapped subsets. It works by training multiple classifiers with different samples of data and making final predictions by voting for employee attrition status classification problem.

The hyperparameter for tuning in bagging is the choice of base model used as a weak learner in the ensemble. After testing several base models, Random Forest was selected as the optimal base model with the highest accuracy of 0.742.

1. AdaBoostM1

This is a boosting technique that focuses on misclassified instances in each round, adjusting weights to improve overall performance.

The hyperparameter for tuning in AdaBoostM1 is the same as bagging, which is the base model. After testing several base models, SVM was selected as the optimal base model for this boosting technique with the highest accuracy of 0.738.

### 3.3.4 Data Mining Modeling and Benchmarking

All nine classification models were then trained and evaluated using their respective optimal parameter values in the Weka explorer. Table 12 below shows the performance metrics of each classification model using 10-fold cross-validation as test option.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Model** | **Accuracy (%)** | **RMSE** | **ROC Area** | **Speed (Sec)** |
| Logistic Regression | 73.5 | 0.4252 | 0.804 | 0.01 |
| Decision Tree(J48) | 69.1 | 0.5429 | 0.680 | 0.06 |
| Random Forest | 73.2 | 0.4670 | 0.780 | 0.34 |
| Support Vector Machine (SVM) | 73.8 | 0.5125 | 0.737 | 0.02 |
| K-Nearest Neighbors (KNN) | 72.9 | 0.4401 | 0.774 | 0 |
| Neural Networks (MLP) | 72.1 | 0.4840 | 0.748 | 0.74 |
| Naïve Bayes | 68.7 | 0.4878 | 0.753 | 0.01 |
| Bagging | 74.2 | 0.4502 | 0.786 | 1.6 |
| AdaBoostM1 | 73.8 | 0.4376 | 0.777 | 0.07 |

Table :Classification Model Performance Metrics

To determine the top three best performing data mining models, all nine classification models were benchmarked with the Experimenter interface in Weka. The Experimenter interface in Weka enabled consistent model evaluation by applying identical validation method and output metrics across classification models.

Figure 0-4 shows the configuration of the experiment in Weka, where all nine models were added into the “Algorithms” box with their optimal parameters. Logistic Regression was chosen as the target model to be compared by other models. The settings for the experiment type are cross-validation with 10 folds and 10 repetitions for iteration control were applied to ensure statistical reliability.

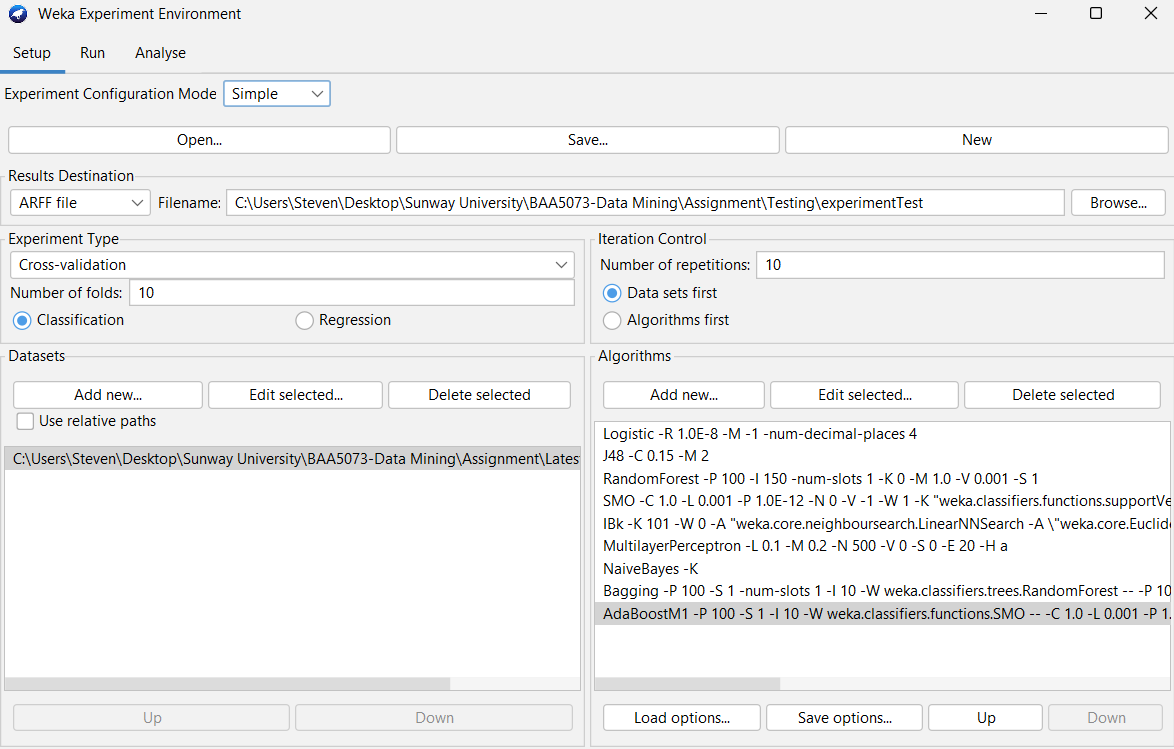


Figure ‑: Configuration of the Experiment in Weka

# 4.0 Results

This part represents the experimental findings from applying the 9 classification models to predict employee attrition using the WEKA platform. The evaluation used 10-fold cross-validation on a balanced dataset.

Models were evaluated on the basis of:

* Accuracy: Correct predictions over total instances
* RMSE (Root Mean Square Error): Average prediction error magnitude
* ROC Area: AUC score representing classifier’s ability to distinguish classes
* Build Time: Efficiency of model training
* Confusion Matrix: Details of model performance on individual class predictions
* Average model performance in 10 iterations

#### 4.1. Optimized Algorithm Ranking and Parameter Tuning

The models were ranked based on mean accuracy, error rate, and ROC after running 10 iterations with 10-fold cross-validation experiments with hyperparameter optimization in WEKA’s Experimenter module. The experiment results from Weka show that only SVM and AdaBoostM1 are performing equally well as compared to the Logistic Regression.

Figure 0-1 below shows the summary of model performance ranking results.

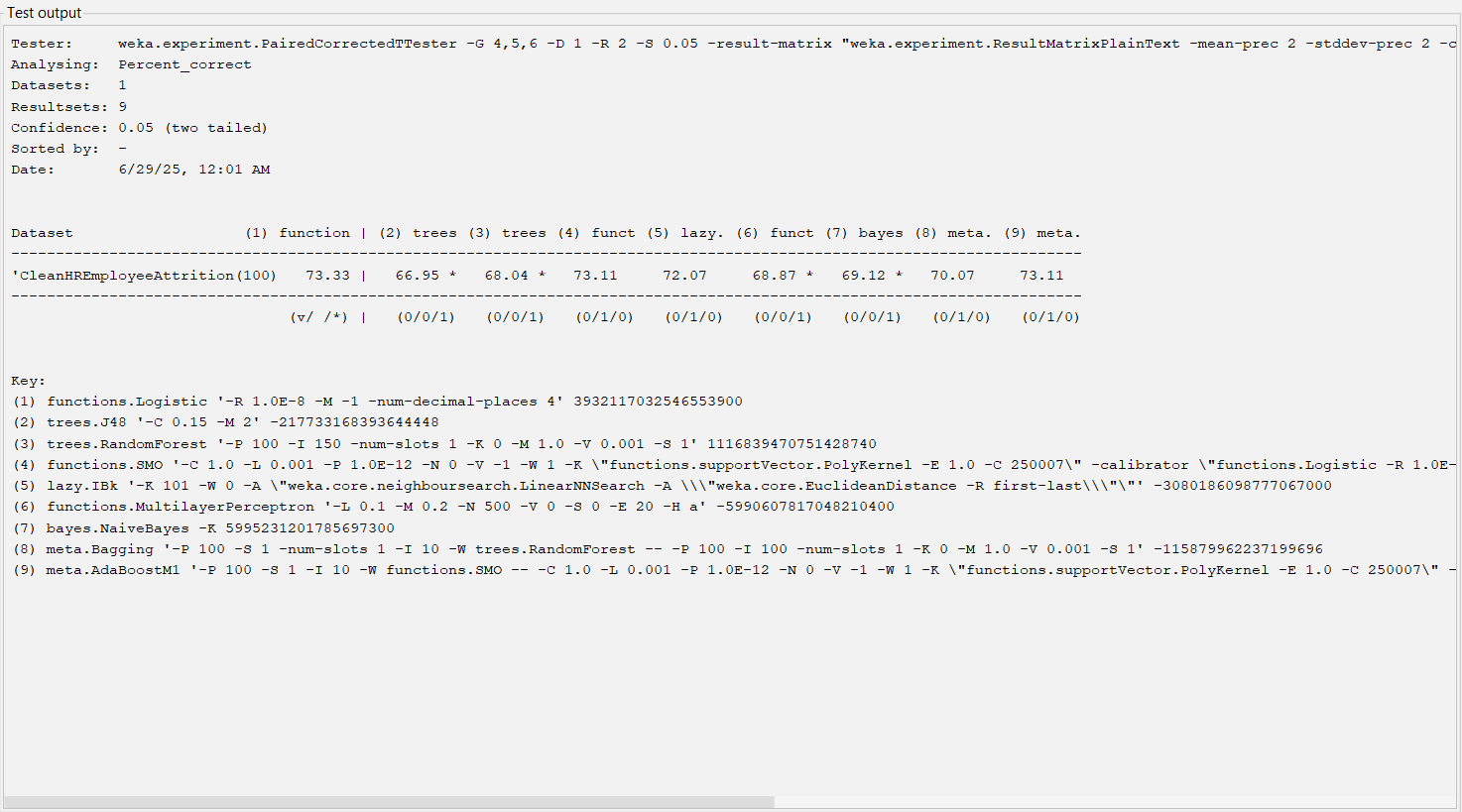


Figure 0‑1: Test Output of Benchmarking 9 Data Mining Models

The key optimized parameters:

|  |  |
| --- | --- |
| **Algorithm** | **Optimized Parameters** |
| Logistic Regression | Ridge = 1.0E-8 |
| SVM | Kernel = PolyKernel |
| AdaBoostM1 | Base = SVM |

Table : Key Optimized Parameters

#### 4.2. The Top Three Classification Models

These are the top three models selected for deeper evaluation based on balanced performance across all metrics and business interpretability.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Model** | **Accuracy (%)** | **RMSE** | **ROC Area** | **Speed (Sec)** |
| Logistic Regression | 73.5 | 0.4252 | 0.804 | 0.01 |
| SVM | 73.8 | 0.5125 | 0.737 | 0.02 |
| AdaBoostM1 with SVM as Base Model | 73.8 | 0.4376 | 0.777 | 0.07 |

Table : Top 3 Classification Models

#### 4.3. Confusion Matrix Analysis

The confusion Metrix Analysis shows how well each model classified the two classes:

* “Yes” (employee will leave)
* “No” (employee will stay)

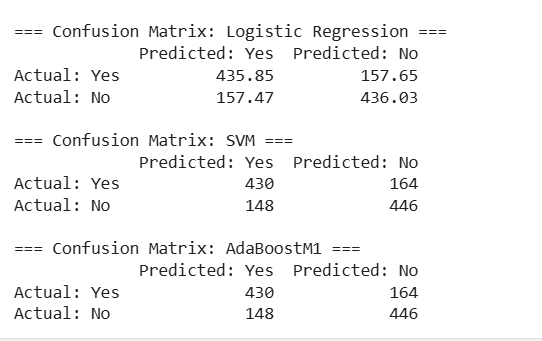


Figure ‑: Confusion Matrices\

#### 4.4. Comparative Interpretation

1. Logistic Regression

* Highest ROC (0.804) and lowest RMSE (0.4252)
* Performs consistently across metrics and is highly interpretable, aligning with Benson and Rajeev (2024), who achieved 87% accuracy using logistic regression on a similar IBM dataset.
* Fastest to train (0.01s) and easiest to integrate in real-time HR dashboards
* Best suited when explainability and speed are critical.

1. Support Vector Machine

* Achieved highest accuracy (73.8%), showing strong boundary learning
* Precision improved but with slightly higher RMSE, indicating less stable probability estimates
* Effective prediction accuracy is prioritized over interpretability, as echoed by Liang & Kang (2024), who enhanced SVM using adaptive training methods.

1. AdaBoostM1 (with SVM)

* Matched SVM in accuracy but showed lower RMSE (0.4376) and improved ROC (0.777)
* Demonstrates benefit of ensemble learning through boosted generalization
* Slightly longer build time (0.07s), but manageable for offline HR systems
* Supports findings by Alqahtani et al. (2024), who emphasize ensemble models' superiority in balancing multiple performance metrics

#### 4.4. Visual Summary of Model Performance

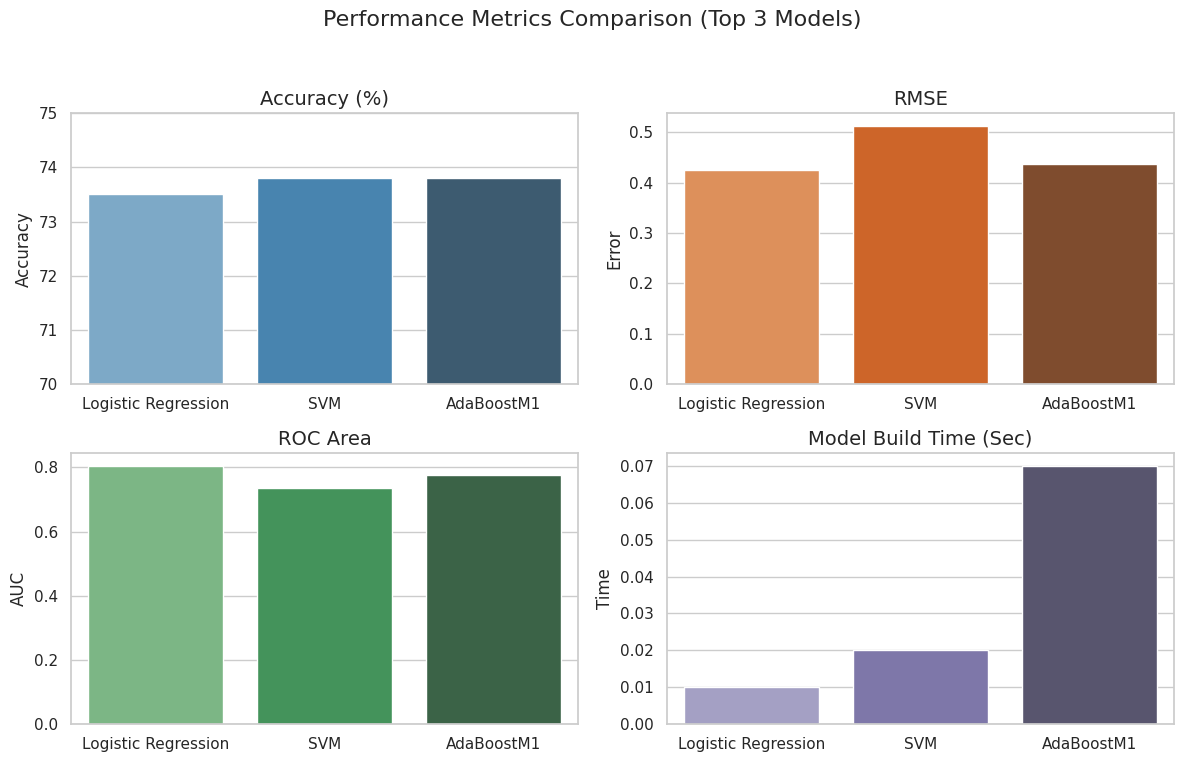


Figure ‑: Performance Matrices

This visualization Figure 0-3 illustrates:

* Accuracy was highest for SVM and AdaBoostM1 (73.8%)
* Logistic Regression achieved the lowest RMSE (0.4252) and highest ROC (0.804)
* Models build time was lowest for Logistic Regression (0.01s)

#### 4.5. Business Context Implication

Predicting employee attrition has significant implications for business operations, work planning, and financial performance. The 3 top-performing models offer distinct advantages that can be mapped to different organizational contexts.

##### 4.5.1. Logistic Regression

This makes it an excellent choice for transparent decision making in HR systems. The model generates coefficients that clearly impact the risk of attrition from an employee. This clarity enables HR teams to explain and defend their decisions, which is particularly important in regulated sectors like healthcare, education, or even the government. The minimal training makes it highly practial for use in real-time dashboards and workforce analytics reporting.

##### 4.5.2. SVM

This is excellent in classification accuracy, which is highly valuable in high-volume and high-turnover industries like retail, customer service and call centers. Despite it being less interpretable like the Logistic Regression, it is suitable for alert systems here decisions do not need to be manually scrutinized. Where organizations value prediction accuracy over understandability, SVM can be used to flag at-risk employees and incentives to retain or train employees can be done.

##### 4.5.3. AdaBoost M1

It performed comparable to SVM but with improved stability and slightly better ROC and RMSE. This model is ideal where data is noisy pr the class distributions vary over time. The ensemble method’s resilience makes it suitable for HR departments that are undergoing digital transformation, where accuracy and generalizability are equally important.

Summary Table:

|  |  |  |
| --- | --- | --- |
| **Model** | **Strengths** | **Best Use Case** |
| Logistic Regression | Interpretable, low error, fast to deploy | HR dashboards and explainable predictions |
| SVM | High accuracy, handles complex patterns | Bulk prediction and forecasting systems |
| AdaBoostM1 | Robust, stable, resilient to noise/class imbalance | HR automation tools with ensemble model support |

Table : Summary Table

Since businesses need to balance how well a model predicts with how easy it is to use in real life and choosing the best model should look at more than just accuracy. It should also so consider how easy it is to understand and what the business needs.   
  
Here, the model’s performances help in making a smart choice when using prediction tool for your requirements or set goals.

# 5.0 Conclusion

This study applied data mining techniques to predict employee attrition using a structured HR dataset. After data preprocessing, nine machine learning models were trained and evaluated using WEKA. The top-performing models were Support Vector Machine (73.8%), AdaBoostM1 with SVM (73.8%), and Logistic Regression (73.5%). While all three models performed well, Logistic Regression stood out in ROC area and interpretability, making it suitable for HR applications.

#### Limitations

The dataset is historical and does not account for real-time employee feedback.

Although class balancing was applied, natural class imbalance still affects precision for minority classes.

The analysis was limited to structured features. There was no inclusion of textual or sentiment data.

#### Recommendations

Future research should explore explainable AI methods (e.g., SHAP, LIME) to improve model interpretability.

Integrating real-time HR feedback or exit interview text data can enhance prediction accuracy.

Deployment of models should consider organizational goals, whether interpretability, speed, or accuracy is the top priority.

# 6.0. REFERENCES

D V, B. A., & Rajeev, H. (2024). Employee Attrition Prediction. *Indian Journal of Data Mining (IJDM)*. <https://doi.org/10.54105/ijdm.a1636.04010524>

Kumar, A., & Ananthanarayanan, Dr. N. R. (2024). Employee Attrition Prediction Using DJANGO Web Application. *International Journal of Research Publication and Reviews*, *5*(5), 12692–12697. <https://doi.org/10.55248/gengpi.5.0524.1443>

Sari, S. F., & Lhaksmana, K. M. (2022). Employee Attrition Prediction Using Feature Selection with Information Gain and Random Forest Classification. *Journal of Computer System and Informatics*, *3*(4), 410–419. <https://doi.org/10.47065/josyc.v3i4.2099>

Barara, S., & Soni, U. (2023). *Employee Attrition Prediction using Machine Learning*. 1–9. <https://doi.org/10.1109/iceccme57830.2023.10252571>

Liang, P., & Kang, D. (2024). *Classifier Optimization with a Test-Time Training Layer for Employee Attrition Prediction*. 801–804. <https://doi.org/10.1109/eecsi63442.2024.10776222>

Rovere, F. (2023). A Study on Predicting Employee Attrition Using Machine Learning. *Studies in Computational Intelligence*, 55–69. <https://doi.org/10.1007/978-3-031-19608-9_5>

Al-Alawi, A. I., & Ghanem, Y. A. (2024). *Predicting Employee Attrition Using Machine Learning: A Systematic Literature Review*. 526–530. <https://doi.org/10.1109/icetsis61505.2024.10459451>

Ozdemir, F., Coskun, M., Gezer, C., & Gungor, V. C. (2020). Assessing Employee Attrition Using Classifications Algorithms. ICISDM ’20: Proceedings of the 2020 the 4th International Conference on Information System and Data Mining, 118–122. <https://doi.org/10.1145/3404663.3404681>

Barara, S., & Soni, U. (2023). Employee Attrition Prediction using Machine Learning. 2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), 1–9. <https://doi.org/10.1109/iceccme57830.2023.10252571>

Attri, T. (2018). Why an employee leaves: Predicting using data mining techniques (Doctoral dissertation, Dublin, National College of Ireland).

4Repaso, J. a. A., Capariño, E. T., Hermogenes, M. G. G., & Perez, J. G. (2022). Determining factors resulting to employee attrition using data mining techniques. International Journal of Education and Management Engineering, 12(3), 22–29. <https://doi.org/10.5815/ijeme.2022.03.03>

Jaffar, Z., Noor, W., & Kanwal, Z. (2019). Predictive human resource analytics using data mining classification techniques. Int. J. Comput, 32(1), 9-20.

PM, U., & Balaji, N. (2019). Analysing Employee attrition using machine learning. Karpagam J. Comput. Sci, 13, 277-282.

Shankar, R. S., Rajanikanth, J., Sivaramaraju, V., & Murthy, K. (2018). PREDICTION OF EMPLOYEE ATTRITION USING DATAMINING. 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN). <https://doi.org/10.1109/icscan.2018.8541242>

Yadav, S., Jain, A., & Singh, D. (2018). Early Prediction of Employee Attrition using Data Mining Techniques. 2018 IEEE 8th International Advance Computing Conference (IACC). <https://doi.org/10.1109/iadcc.2018.8692137>

Shankar, R. S., Priyadarshini, V., Neelima, P., & Raminaidu, C. (2021). Analyzing Attrition and Performance of an Employee using Machine Learning Techniques. 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), 1601–1608. <https://doi.org/10.1109/iceca52323.2021.9676102>

Jain, D. (2017). Evaluation of employee attrition by effective feature selection using hybrid model of ensemble methods (Doctoral dissertation, Dublin, National College of Ireland).

Ganthi, L. S., Nallapaneni, Y., Perumalsamy, D., & Mahalingam, K. (2021). Employee attrition prediction using machine learning algorithms. In Lecture notes in networks and systems (pp. 577–596). <https://doi.org/10.1007/978-981-16-5120-5_44>

Singh, R. K. N. (2023). Impact Analysis on Employee Attrition using Machine Learning Techniques. International Journal of Science and Research (IJSR), 12(6), 2762–2766. <https://doi.org/10.21275/sr23627120419>

Mansor, N., Sani, N. S., & Aliff, M. (2021). Machine learning for predicting employee attrition. International Journal of Advanced Computer Science and Applications (IJACSA), 12(11).

Vouk, B., Guid, M., & Robnik-Sikonja, M. (2023). Feature construction using explanations of individual predictions. Eng. Appl. Artif. Intell., 120, 105823. <https://doi.org/10.1016/j.engappai.2023.105823>

Yahia, N., Hlel, J., & Colomo-Palacios, R. (2021). From Big Data to Deep Data to Support People Analytics for Employee Attrition Prediction. IEEE Access, 9, 60447-60458. <https://doi.org/10.1109/ACCESS.2021.3074559>

Vu, H., Ng, K., Richter, A., & An, C. (2022). Analysis of input set characteristics and variances on k-fold cross validation for a Recurrent Neural Network model on waste disposal rate estimation.. Journal of environmental management, 311, 114869 . <https://doi.org/10.1016/j.jenvman.2022.114869>

Thabtah, F., Hammoud, S., Kamalov, F., & Gonsalves, A. (2020). Data imbalance in classification: Experimental evaluation. Inf. Sci., 513, 429-441. <https://doi.org/10.1016/j.ins.2019.11.004>

Ding, N., Qin, Y., Yang, G., Wei, F., Yang, Z., Su, Y., Hu, S., Chen, Y., Chan, C., Chen, W., Yi, J., Zhao, W., Wang, X., Liu, Z., Zheng, H., Chen, J., Liu, Y., Tang, J., Li, J., & Sun, M. (2023). Parameter-efficient fine-tuning of large-scale pre-trained language models. Nature Machine Intelligence, 5, 220-235. <https://doi.org/10.1038/s42256-023-00626-4>

