MACHINE LEARNING - R

EMPLOYEE ATTRITION ANALYSIS

Identifying the best ML model for predicting employee attrition using R

EFFORTS BY: MRIDUL BHALLA

Introduction

This project aims to develop an optimal machine learning model in R for predicting employee attrition at ABC Company. Employee attrition, defined as the voluntary departure of employees, poses challenges such as increased recruitment costs and loss of expertise. By leveraging different ML models, we seek to determine the most effective approach for predicting and mitigating attrition.

Problem Statement

ABC Company is grappling with high employee attrition, recognizing the pivotal role employees play in meeting deadlines, driving sales, and enhancing brand reputation through positive customer interactions. To address this, the company's Human Resources department has gathered data on various variables and aims to develop a precise ML model to predict employee resignations effectively.

Understanding the Dataset Variables

The dataset for analysis comprises 17 variables, each capturing distinct aspects concerning employee demographics, employment history, and work-related factors. Here's a concise summary of the variables provided.

S. NO.	VARIABLE NAME	DESCRIPTION
1	Age	The age of the employee
2	Attrition	Indicates whether an employee has left the company (Yes) or is still employed (No)
3	BusinessTravel	Frequency of business travel among employees (Travel_Rarely, Travel_Frequently, Non-Travel)
4	Department	The department in which the employee works (Sales, Human Resources, Research & Development)
5	DistanceFromHome	The distance of the employee's home from the workplace.
6	EducationField	The education background of the employee. (Human Resources, Life Sciences, Marketing, Medical, Technical Degree, Other)
7	Employee ID	Unique identifier for each employee
8	Gender	Gender of the employee. (Male, Female)
9	JobLevel	Level of the employee's job within the organizational hierarchy (1, 2, 3, 4, 5)
10	MaritalStatus	Marital status of the employee (Married, Single, Divorced)
11	MonthlyIncome	The monthly income of the employee
12	NumCompaniesWorked	Number of companies the employee has worked for previously.
13	PercentSalaryHike	Percentage increase in salary during the most recent salary hike.
14	TotalWorkingYears	Total number of years the employee has been working.
15	YearsAtCompany	Number of years the employee has been with the company.
16	YearsSinceLastPromotion	Number of years since the employee's last promotion.
17	YearsWithCurrManager	Number of years the employee has been working under the current manager.

Exploratory Data Analysis

Data Cleaning

STEP 1. Converting the Data Set into a Data Frame

Initially, we converted the raw data file "Employee Attrition" stored in the variable EA into a data frame. We utilized the **as.data.frame** function for this conversion.

STEP 2. Correcting Data Types

Here we convert categorical fields into factors and also address missing values in the **NumCompaniesWorked** and **TotalWorkingYears** columns, where they're represented as "NA" characters by using the **as.numeric function.**

STEP 3. Removing Unnecessary Columns

Here, we eliminate unnecessary columns to ensure the accuracy of our predictive machine-learning models. In this case, the only irrelevant column was the **EmployeeID** column.

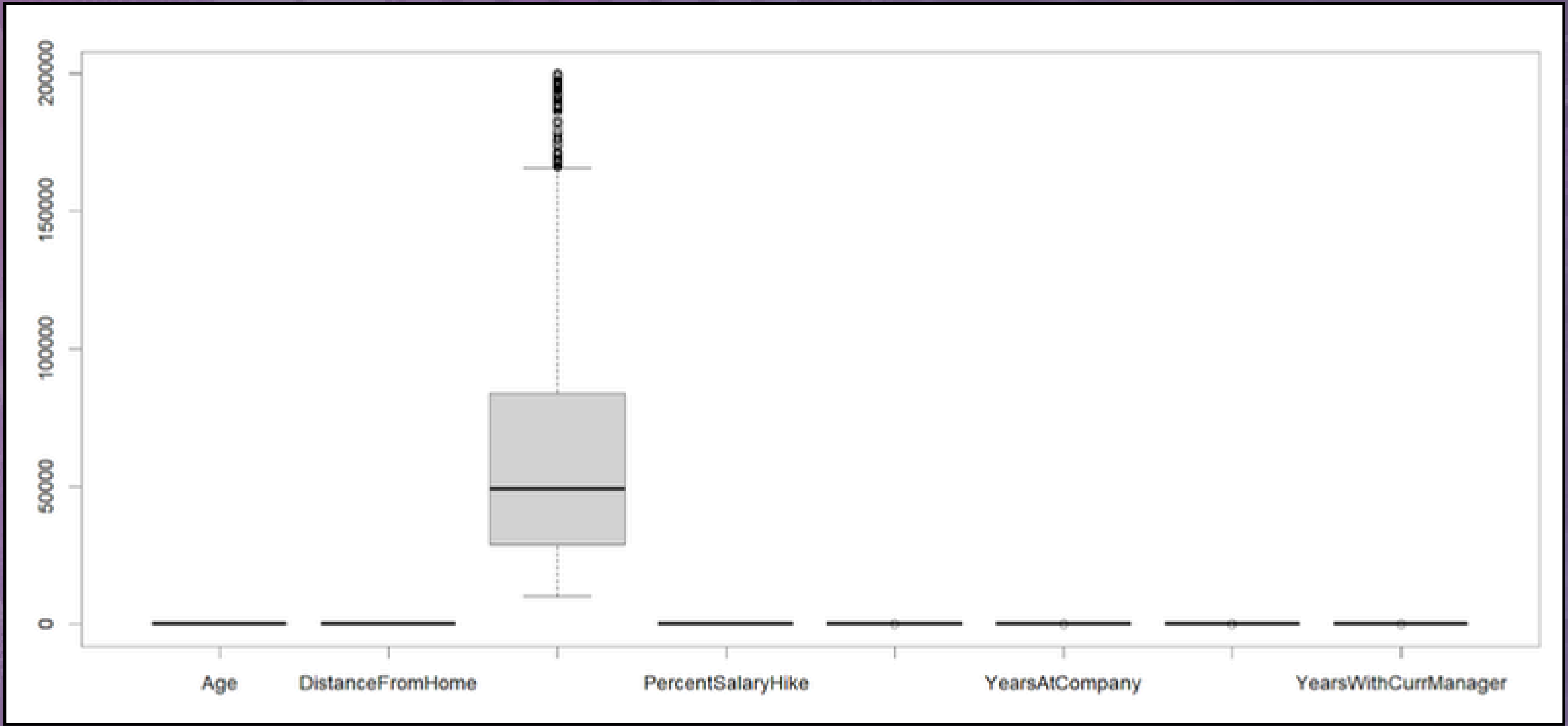
STEP 4. Dealing with Missing Values

We addressed the missing values in two columns: **NumCompaniesWorked** and **TotalWorkingYears.** We opted for median-based imputation over deletion due to missing values falling below the 70-75% threshold for a column.

STEP 5. Checking and Treating Normality and Outliers

We assess data normality and identify outliers using skewness and kurtosis criteria (-1 to +1 for skewness, -4 to +4 for kurtosis) and boxplots, and we found **MonthlyIncome**, **TotalWorkingYears**, **YearsSinceLastPromotion**, and **YearsWithCurrManager** columns to be problematic.

```
> skewness(EA[c(1,5,10,12,13,14,15,16)])
                    Age        DistanceFromHome           MonthlyIncome
              0.4128645               0.9571400               1.3684185
      PercentSalaryHike       TotalWorkingYears          YearsAtCompany
              0.8202899               1.1184985               1.7627284
YearsSinceLastPromotion    YearsWithCurrManager
              1.9822646               0.8326003
> kurtosis(EA[c(1,5,10,12,13,14,15,16)])
                    Age        DistanceFromHome           MonthlyIncome
               2.593149                2.771852                3.997738
      PercentSalaryHike       TotalWorkingYears          YearsAtCompany
               2.696344                3.919605                6.918057
YearsSinceLastPromotion    YearsWithCurrManager
               6.596318                3.166398
```



STEP 6. Storing the outliers in vectors

Before treating our outliers, we have to store the outliers in vectors, as it is important for the methods of treating outliers where we remove outliers and the method where we impute outliers.

Exploratory Data Analysis

Treatment of Normality & Outliers

STEP 1. Creating Data Copies

To preserve the original data, we create three copies labeled EA1, EA2, and EA3, allowing for the independent application of three methods to treat normality and outliers.

STEP 2. Applying Method 1 (Removing Outliers in EA1)

Outlier removal in data copy EA1, targeting outliers stored in vectors out10, 13, 14, 15, and 16, fails to normalize the data or resolve the outlier issue in previous columns, with additional outliers observed in the **Age** column.

STEP 3. Applying Method 2 (Imputing Outliers in EA2)

In the 2nd method, using data copy EA2, we replace outlier values with NA and impute them using the median since the data is numeric. However, despite these efforts, the data remains non-normal, with persistent outliers.

STEP 4. Applying Method 3 (Transformation of Outliers in EA3)

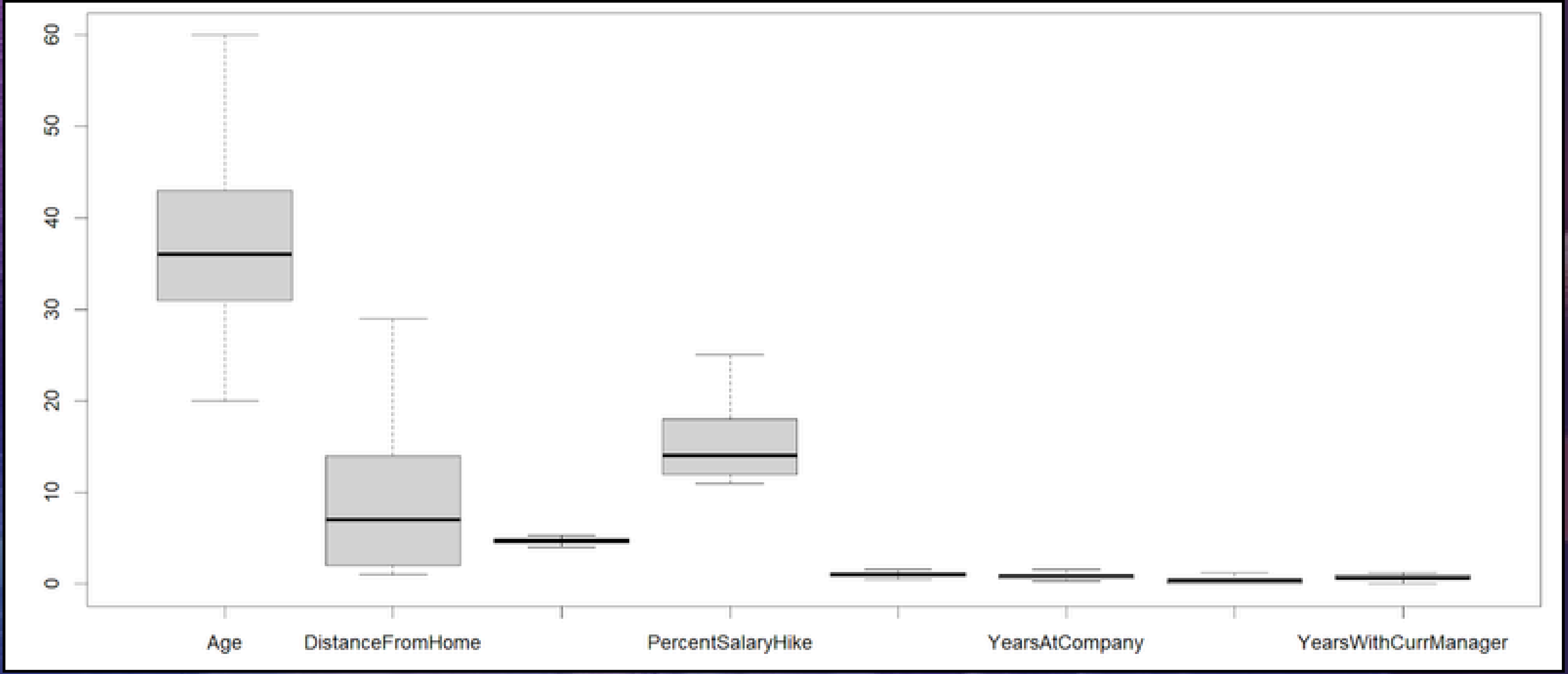
In the third method, applied to data copy EA3, we focus on variable transformation. This includes square root transformation for EA3a and log transformation for EA3b. While square root transformation resolves normality issues, outliers persist in two columns. After applying the log transformation to data copy EA3B, using the formula log10(x+1) for values equal to 0, normality concerns are resolved, yet outliers persist.

Treatment of Normality and Outliers

STEP 5. Performing Log Transformation & then Removing Outliers

In this step, we combined two methods for treating normality and outliers by creating data copy EA4 from the log-transformed dataset EA3b. Outliers in columns **(TotalWorkingYears & YearsAtCompany)** were stored in vectors out 13a and out 14a, then removed from EA4. This integration yielded a final dataset that is normally distributed with no outliers, serving as the basis for all machine learning models.

```
> skewness(EA4[c(1,5,10,12,13,14,15,16)])
                Age     DistanceFromHome          MonthlyIncome     PercentSalaryHike
         0.46487194           0.92276118             0.28791248            0.82791983
  TotalWorkingYears       YearsAtCompany YearsSinceLastPromotion YearsWithCurrManager
         0.01013163           0.04576743             0.60871682           -0.47079453
> kurtosis(EA4[c(1,5,10,12,13,14,15,16)])
                Age     DistanceFromHome          MonthlyIncome     PercentSalaryHike
           2.549793             2.679714               2.307532              2.702307
  TotalWorkingYears       YearsAtCompany YearsSinceLastPromotion YearsWithCurrManager
           2.594497             2.587616               2.275235              2.493079
```



Splitting the Data for Machine Learning

Data splitting is a critical step in machine learning where the dataset is divided into training and testing subsets to evaluate model performance objectively.

1. Setting Seed

Utilizing `set.seed(100)` in R ensures reproducibility by initializing the random number generator with a specific seed value.

2. Data Partitioning

Employing `createDataPartition()` from the caret package divides the dataset into training and testing sets. The proportion allocated to training is 80%.

3. Training and Testing Sets

The training set (80% of the data) is used to train machine learning models, while the testing set (20%) is used for evaluation.

Examining the testing set reveals an imbalance: 699 instances belong to the negative class (No), while only 110 belong to the positive class (Yes). This imbalance can bias model performance, prioritizing accuracy on the majority class. Hence, balanced accuracy and precision are also considered for model comparison.

Making Various Machine Learning Models

MODEL	ALGORITHM USED
Model 1	Based on Binomial logistic regression using all the independent variables
Model 2	Based on Binomial logistic regression using only the significant independent variables
Model 3	Based on Naive Bayes Algorithm
Model 4	Based on decision tree using gini index as the attribute selection measure
Model 5	Based on decision tree using information gain as the attribute selection measure
Model 6	Based on random forest
Model 7	Made on Non-transformed data using decision tree and Gini index as the ASM
Model 8	Made on Non-transformed data using decision tree and information gain as the ASM
Model 9	Made on non transformed data using random forest

Comparison Table of Machine Learning Models

MODEL	ACCUARCY	SENSITIVITY	SPECIFICITY	PRECISION	BALANCED ACCURACY
Model 1	86.4%	3.64%	99.43%	50.00%	51.42%
Model 2	86.65%	3.64%	99.71%	66.67%	51.68%
Model 3	86.4%	0.00%	100.00%	N/A	50.00%
Model 4	87.52%	13.64%	99.14%	71.43%	56.39%
Model 5	87.27%	12.73%	98.99%	66.67%	55.86%
Model 6	99.63%	97.27%	100.00%	100.00%	98.64%
Model 7	83.8%	0.00%	100.00%	N/A	50.00%
Model 8	83.8%	0.00%	100.00%	N/A	50.00%
Model 9	84.34%	4.17%	99.83%	82.85%	52.00%

Evaluation

METRICS	OBSERVATION
Accuracy	Model 6 (Random Forest) achieved the highest accuracy of 99.63%, indicating the proportion of correct predictions made by the model across all classes.
Sensitivity	Model 6 (Random Forest) achieved the highest sensitivity of 97.27%, indicating its ability to correctly identify employees who left the company (attrition cases) among all actual attrition cases.
Specificity	Model 6 (Random Forest) achieved a perfect specificity of 100.00%, indicating its ability to correctly identify employees who did not leave the company (non-attrition cases) among all actual non-attrition cases.
Precision	Model 6 (Random Forest) achieved a perfect precision of 100.00%, indicating that when it predicts attrition, it is always correct.
Balanced Accuracy	Model 6 (Random Forest) achieved the highest balanced accuracy of 98.64%, the average of sensitivity and specificity, providing a balanced assessment of the model's performance across both classes.

Model 6, employing the Random Forest algorithm, demonstrates superior predictive performance for attrition prediction at ABC Company. With an accuracy of 99.63% and excellent sensitivity and specificity, it outperforms all other models. The Random Forest algorithm's ability to handle complex data relationships and prevent overfitting makes it the optimal choice. Thus, Model 6 is recommended for informing HR decision-making processes regarding employee attrition at ABC Company.


THANK
YOU