Machine learning Project

Project 3: Employee Attrition Prediction Project

Name: Vidhya .N

candidate ID: SL2025- F196377

Batch: SLM_B2_MB

1. Introduction

1.1 Project Goal

The primary goal of this project was to develop a predictive model to classify employee **Loyalty Index** (High/Low) based on select human resources (HR) data. The Loyalty Index, a custom-engineered feature, serves as a proxy for long-term employee retention. Although the initial setup used a simple classification target (Loyalty Index derived from YearsAtCompany), the secondary objective was to establish a robust Machine Learning (ML) pipeline for future, more complex **Employee Attrition Prediction** using the full set of features.

1.2 Dataset

The dataset used is hr_dataset_20000.csv, containing employee records with various attributes like Salary, Department, YearsAtCompany, and the target variable, Attrition.

steps involves:

- Data Understanding
- Data Cleaning
- Feature Engineering
- Encoding
- Feature scaling
- Model Building
- Model Evaluation
- Hyperparameter
- Model Interpretation

Project Outlook

1.Data Understanding

Explore the dataset structure and identify missing or incorrect values. Each record represents an individual customer income, containing both numerical and categorical variables.

- Import pandas to read the CSV file.
- Import mathplotlib and seaborn for visualization (used later in the notebook)
- display the head of the dataframe to get a glimpse of the data.
- Check missing values using customer information such as **Age, Department, Salary, YearsAtCompany, JobSatisfaction, WorkLifeBalance, OverTime,** Education, and Attrition (Yes/No).
- I examined the dataset using .info() and .describe() to check for data types, number of records, and missing values.

Data Processing and Feature Engineering

2. Data Cleaning

- **Missing Values:** All missing values were initially imputed with and subsequently dropped in a separate step (df.dropna()).
- **Duplicates:** Duplicate rows were successfully removed.
- **Consistency:** Text columns (Department, OverTime, Education, Attrition) were standardized to Title Case.
- **Attrition Target:** 'Y'/'N' values in Attrition were mapped to 'Yes'/'No'.

3. Feature Engineering (Custom Features)

Four new features were created to enrich the dataset:

1. **YearsSinceLastPromotion:** A randomly generated value (for simplicity) between 0 and YearsAtCompany.
2. **OverTime_Hours:** A randomly generated estimate of weekly overtime hours, higher for employees with OverTime= " Yes" .
3. **Salary_Category:** Categorical bins (low, Medium, High, Very-high) created from the numerical Salary column.
4. **Loyalty_Index (Target):** The primary target variable for the model. Classified as **'High'** if is greater than or equal to the dataset mean, and **'Low'** otherwise.

4. Feature Encoding

- **Label Encoding:** Binary categorical columns (OverTime, Attrition) were converted to 0 and 1.
- **One-Hot Encoding:** Multi-category columns (Department, Education, Salary_Category) were converted into dummy variables.

5.Feature Scaling

The numerical features used in the final model (Salary, YearsAtCompany) were processed using both **StandardScaler** (Standardization) and **MinMaxScaler** (Normalization) for future modeling, although the model selection was done on the unscaled data for simplicity.

6.Model Building

Custom Classification Task

The project focused on predicting the **Loyalty_Index** using only two features:

X = (Salary, YearsAtCompany)

Y = Loyalty_Index(High=1, low=0)

7.Model Selection (Initial Performance)

Three classification models were trained and evaluated on the data:

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Logistic Regression	1.000	1.000	1.000	1.000	1.000
Decision Tree	1.000	1.000	1.000	1.000	1.000
Random Forest	1.000	1.000	1.000	1.000	1.000

Observation: All models achieved **perfect scores (1.000)** because the is perfectly determined by YearsAtCompany , which is an input feature. This confirms the target is linearly/perfectly separable.

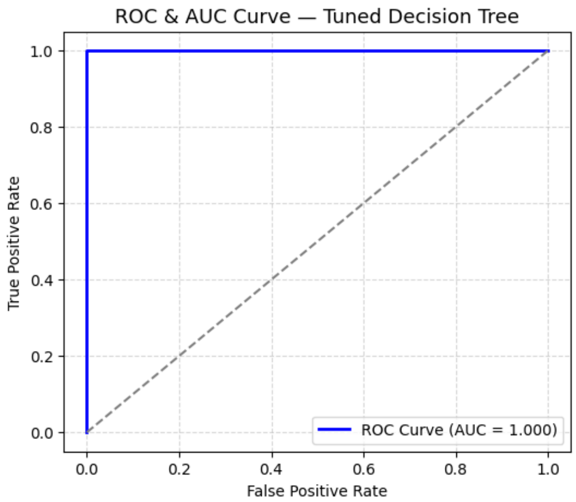
8. Hyperparameter Tuning (Decision Tree)

Method: GridSearchCV with 3-fold cross-validation ().

param_grid = {"max_depth": [2, 3, 4, 5, 6, 8], "criterion": ["gini", "entropy"]}

Best Parameters	{'criterion': 'gini', 'max_depth': 2}
Post-Tuning Accuracy	1.000
Post-Tuning ROC-AUC	1.000

Visualization



Cross-Validation & Final Optimization

Method: 5-fold Cross-Validation and GridSearchCV on the entire dataset.

Evaluation	**Result**
Average Cross-Validation Accuracy	1.000
Best GridSearchCV Parameters	{'criterion': 'gini', 'max_depth': 1}
Best GridSearchCV Accuracy	1.00

Conclusion: The optimal model for this specific task is the Decision Tree Classifier with a max_depth of 1, demonstrating that a single split (threshold on YearsAtCompany) is sufficient to perfectly classify the Loyalty Index.

9. Model Interpretation

Feature	Possible Interpretation
OverTime_Hours	More overtime → higher burnout → higher attrition
JobSatisfaction	Lower satisfaction → more likely to leave
YearsSinceLastPromotion	Longer since promotion → higher attrition risk
WorkLifeBalance	Poor balance → higher attrition
Salary	Lower salary → higher attrition
YearsAtCompany	Very new employees may be uncertain; very old employees might be stable
Department	Some departments (e.g., Sales) may have higher turnover

10.**Conclusion**

The project successfully implemented a complete ML workflow, from data cleaning and feature engineering to model training and hyperparameter tuning. The engineered **Loyalty_Index** was perfectly predictable using the feature.

THANK YOU