HR Employee Attrition Prediction – Data Science Assignment

**1. Introduction**

**Objective:**

The goal of this project is to analyze HR data and predict employee attrition. By leveraging machine learning techniques, we aim to identify key factors influencing employee retention and turnover.

**Dataset Description:**

* The dataset contains demographic, job satisfaction, performance, and compensation-related information.
* Target Variable: "Attrition" (Yes/No).

**2. Exploratory Data Analysis (EDA)**

**2.1 Data Quality Check**

**Missing Values Analysis**

* No missing values were found in the dataset.

**Outlier Detection**

* Box plots identified **outliers in Monthly Income**.
* Decision: Retained outliers since salary variation is natural.

**2.2 Statistical Tests & Hypothesis Testing**

To validate relationships between features and attrition, we conducted the following tests:

**1. Independent T-Test (Attrition vs. Monthly Income)**

* **p-value < 0.05**, so we reject the null hypothesis.
* **Conclusion:** Employees with lower salaries have a higher chance of leaving.

**2. ANOVA Test (Attrition vs. Job Roles)**

* Job role significantly impacts attrition rates.

**3. Mann-Whitney U Test (Attrition vs. Performance Rating)**

* No significant difference between performance ratings of those who stayed vs. left.

**2.3 Summary Insights from EDA**

* Employees with **low job satisfaction and lower salaries** are more likely to leave.
* **Job roles significantly impact attrition rates.**

**3. Feature Engineering**

**3.1 Encoding Categorical Variables**

* Applied **One-Hot Encoding** for categorical features (e.g., Job Role, Business Travel).

**3.2 Feature Selection**

* Removed highly correlated variables.
* **Final key features:** Age, Monthly Income, Job Satisfaction, Work-Life Balance, Job Role.

**4. Model Building & Performance**

**4.1 Machine Learning Models Used**

* **Logistic Regression** (Baseline)
* **Random Forest Classifier**
* **XGBoost Classifier** (Final Model)

**4.2 Confusion Matrix Before Handling Class Imbalance**

Since attrition cases ("Yes") were much fewer than non-attrition cases ("No"), initial models performed poorly on minority class prediction.

**Initial Confusion Matrix (Without Class Balancing)**

|  |  |  |
| --- | --- | --- |
|  | **Predicted No** | **Predicted Yes** |
| **Actual No** | 2900 | 100 |
| **Actual Yes** | 350 | 50 |

* **Issue:**
  + The model **misclassified 350 employees who left** as staying.
  + **Very low recall (50/400 = 12.5%)** for predicting "Yes" cases.
  + The model learned to predict "No" most of the time due to class imbalance.

**5. Handling Class Imbalance & Model Improvement**

**5.1 Why Class Imbalance Needed to Be Addressed?**

* The dataset was **highly imbalanced**, with far fewer employees leaving compared to staying.
* If not handled, the model **fails to identify employees likely to leave**, making it useless for HR decisions.

**5.2 Solution: SMOTE (Synthetic Minority Oversampling Technique)**

* Applied **SMOTE** to generate synthetic data for the minority class ("Yes").
* This helped **balance the dataset** and improve recall for attrition cases.

**6. Model Performance After Hyperparameter Tuning**

**6.1 Hyperparameter Tuning Process**

Used **GridSearchCV** to optimize the XGBoost model:

* n\_estimators: 100 → 200
* max\_depth: 3 → 6
* learning\_rate: 0.1 → 0.05
* colsample\_bytree: 0.8 → 0.9

**6.2 Confusion Matrix After Class Balancing & Hyperparameter Tuning**

|  |  |  |
| --- | --- | --- |
|  | **Predicted No** | **Predicted Yes** |
| **Actual No** | 2800 | 200 |
| **Actual Yes** | 150 | 250 |

* **Improvements:**
  + Recall for attrition cases improved from **12.5% → 62.5% (250/400)**.
  + The model now correctly predicts more employees who are likely to leave.

**7. Challenges Faced During Model Building**

**1. Class Imbalance**

* **Problem:** The dataset had far more "No" cases than "Yes."
* **Solution:** Applied **SMOTE** to balance the classes and improve minority class predictions.

**2. Outliers in Monthly Income**

* **Problem:** Salaries had extreme values, which could mislead the model.
* **Solution:** Outliers were retained since salary variation is natural.

**3. Choosing the Best Model**

* **Problem:** Logistic Regression and Random Forest failed to capture attrition trends well.
* **Solution:** **XGBoost performed best after hyperparameter tuning and class balancing.**

**8. Conclusion & Key Takeaways**

**8.1 Key Findings**

* **Low salary and job dissatisfaction** drive attrition.
* **Job roles significantly impact attrition rates.**
* **Handling class imbalance was critical for improving model recall.**

**8.2 Final Model Performance**

* **Best Model:** **XGBoost (After Hyperparameter Tuning & SMOTE)**
* **Final Recall for Attrition Cases:** **62.5% (250/400)**

**8.3 Future Scope**

* Deploy model into an **HR dashboard** for real-time predictions.
* Experiment with **Deep Learning** (e.g., Neural Networks) for further improvement.