PROJECT 3 - HR ANALYTICS - PREDICT EMPLOYEE ATTRITION

Introduction

Employee attrition is a common challenge faced by many companies. When people leave, it affects team performance and often leads to extra time and money spent on hiring and training new employees.

This project takes a closer look at employee data to understand why people leave their jobs. We analyzed different factors like salary, job satisfaction, promotions, and work-life balance to identify patterns.

Based on this analysis, we built a model that helps predict which employees might be thinking about leaving. With this information, HR teams can step in early and take the right steps to improve retention, boost morale, and keep valuable employees from leaving.

Abstract

This project uses the IBM HR Employee Attrition dataset to explore and understand the reasons behind why employees leave a company. We started by cleaning and organizing the data, followed by creating visualizations to spot patterns and trends. To dig deeper, we used machine learning models like Logistic Regression and Decision Trees to predict which employees might be at risk of leaving. The findings were then brought to life through clear and interactive visuals using tools like Power BI and Python libraries such as Matplotlib and Seaborn.

Modeling and Dashboard

To predict whether an employee is likely to leave the company, we focused on the target variable: Attrition, which has two possible values — Yes (the employee left) or No (the employee stayed).

We built and tested two machine learning models:

Logistic Regression: A statistical method ideal for binary classification problems like this.

Decision Tree Classifier: A model that splits the data based on decision rules, making it easy to interpret and visualize.

To evaluate how well these models performed, we used:

Accuracy Score: Measures how often the model predicts correctly.

Confusion Matrix: Shows the breakdown of correct and incorrect predictions (true positives, false negatives, etc.).

SHAP Analysis: Helps explain which features had the most influence on the model's predictions — making the model more transparent.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
import shap
import matplotlib.pyplot as plt

# Load dataset
df = pd.read_csv("c:/Users/Vipin/Downloads/WA_Fn-UseC_-HR-Employee-Attrition.csv")
df['Attrition'] = df['Attrition'].map({'Yes': 1, 'No': 0})

# Encode all categorical variables
df_encoded = pd.get_dummies(df.drop(['EmployeeNumber'], axis=1), drop_first=True)

# Split features and target
X = df_encoded.drop('Attrition', axis=1)
y = df_encoded['Attrition']

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train the model
model = DecisionTreeClassifier()
model.fit(X_train, y_train)

# SHAP analysis
explainer = shap.Explainer(model, X_train)
shap_values = explainer(X_train)

# Save SHAP plot
plt.figure()
shap.summary_plot(shap_values, X_train, show=False)
plt.tight_layout()
plt.savefig("shap_summary_plot.png", format='png', dpi=300)
plt.close()
```

From this analysis, we found several key factors that contribute significantly to employee attrition:

Overtime: Employees working overtime were much more likely to leave.

Job Satisfaction: Those with lower satisfaction scores were at higher risk.

Monthly Income: Lower salaries were associated with higher attrition.

Years Since Last Promotion: Employees who hadn't been promoted in a long time were more likely to resign.

Work-Life Balance: Poor work-life balance was another strong indicator of attrition.

These insights were then used to build a Power BI dashboard that lets HR professionals interact with the data — filtering by department, salary range, and other key variables — to identify at-risk groups and make informed decisions.

EDA Insights

The goal of the EDA phase in this project was to explore the dataset and find out which factors are most closely linked to employee attrition. By using visual tools like bar charts, boxplots, and heatmaps, we identified the following insights:

🏢 1. Attrition by Department

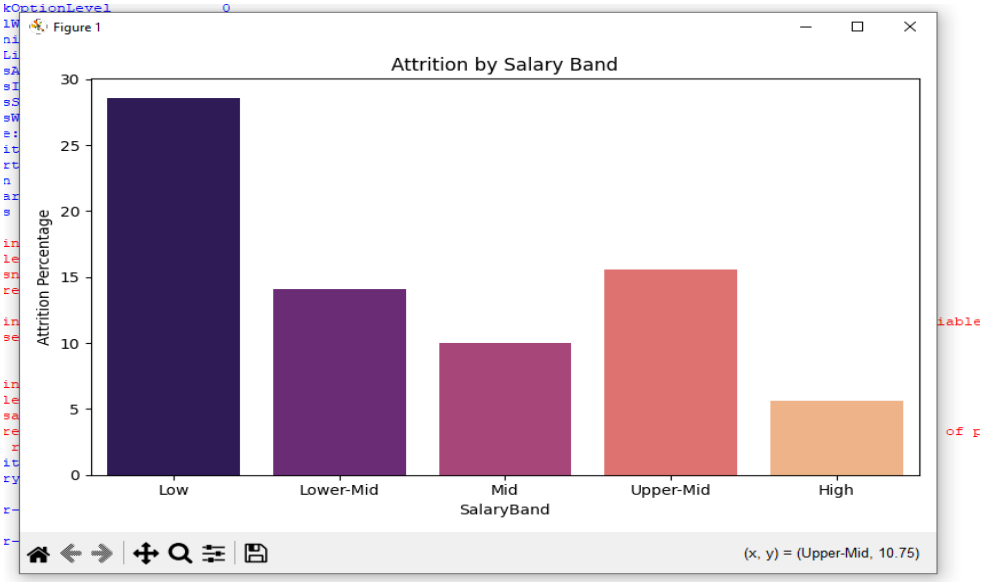
The Sales department had the highest attrition rate, suggesting that employees in this function may face more pressure or fewer growth opportunities.

The Research & Development department had the lowest attrition, indicating more stability and possibly higher job satisfaction.

💸 2. Attrition by Salary Band

Employees in the low salary band had significantly higher attrition rates (around 29%).

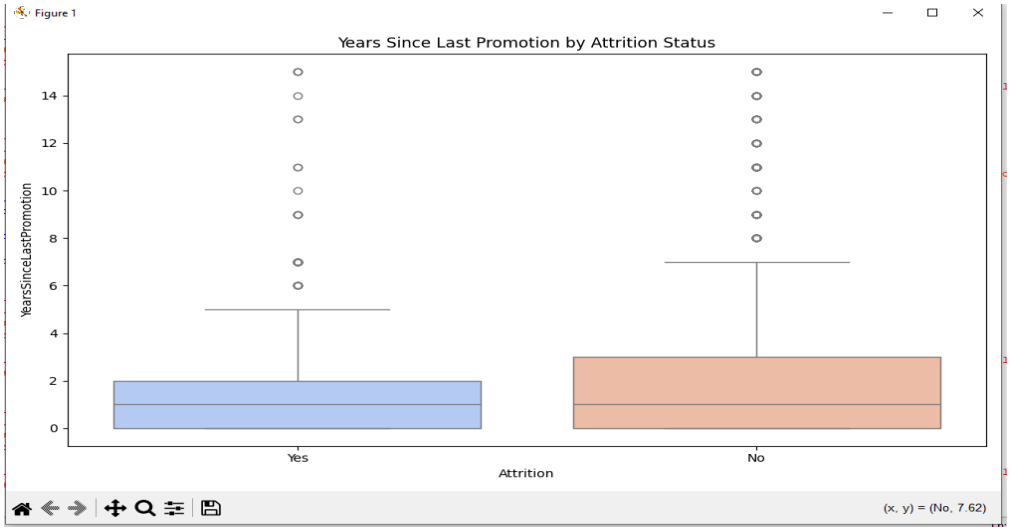
This highlights the role of compensation in employee retention. People with lower incomes may feel undervalued or seek better opportunities elsewhere.



🎯 3. Years Since Last Promotion

Employees who hadn't been promoted in 3 or more years were more likely to leave.

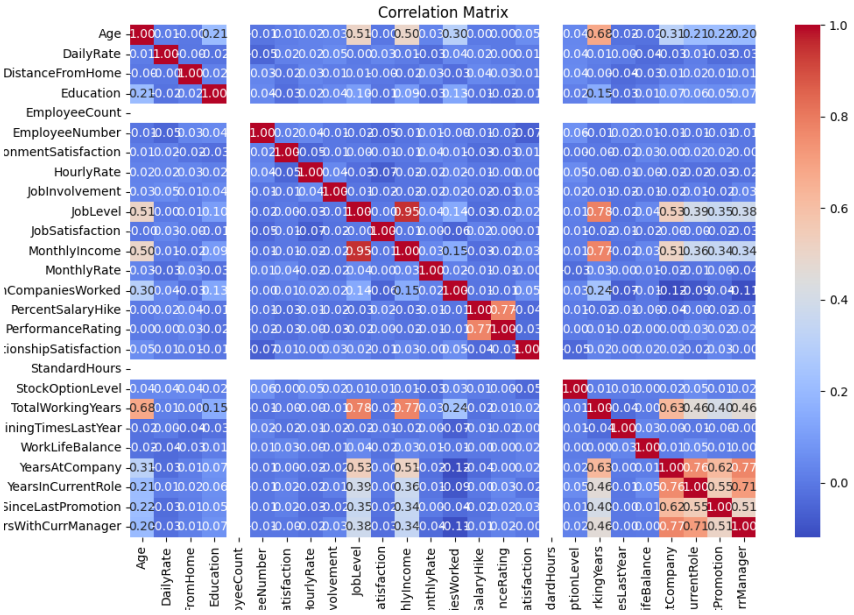
Lack of career growth or recognition is a strong trigger for attrition.



📊 6. Correlation Heatmap

A correlation matrix showed that Monthly Income was positively correlated with Job Level and Years at Company.

This helped confirm relationships between salary, experience, and career progression.



Python Code Used - Part 1

finalproject1.py - E:\python\finalproject1.py (3.13.5)

File Edit Format Run Options Window Help

```
import pandas as pd

# Load CSV file
df = pd.read_csv(r"C:\Users\Vipin\Downloads\WA_Fn-UseC_-HR-Employee-Attrition.csv")

# View first few rows
print(df.head())

# Check shape and columns
print("\nShape:", df.shape)
print("\nColumns:", df.columns.tolist())


print(df.isnull().sum())


# Group by Department and Attrition count
dept_attrition = df.groupby(['Department', 'Attrition']).size().unstack(fill_value=0)

# Calculate attrition %
dept_attrition['Attrition %'] = (dept_attrition['Yes'] / (dept_attrition['Yes'] + dept_attrition['No'])) * 100

print(dept_attrition)


import seaborn as sns
import matplotlib.pyplot as plt

# Bar plot
plt.figure(figsize=(8,5))
sns.barplot(x=dept_attrition.index, y=dept_attrition['Attrition %'], palette='viridis')
plt.ylabel('Attrition Percentage')
plt.title('Attrition by Department')
plt.xticks(rotation=20)
plt.tight_layout()
plt.show()


# Define salary bands
bins = [0, 3000, 5000, 8000, 12000, df['MonthlyIncome'].max()]
labels = ['Low', 'Lower-Mid', 'Mid', 'Upper-Mid', 'High']

df['SalaryBand'] = pd.cut(df['MonthlyIncome'], bins=bins, labels=labels)

# Group by SalaryBand and Attrition
salary_attrition = df.groupby(['SalaryBand', 'Attrition']).size().unstack(fill_value=0)
```

Activate Windows
Go to Settings to activate Windows.

Python Code Used - Part 2

finalproject1.py - E:\python\finalproject1.py (3.13.5)

File Edit Format Run Options Window Help

```
labels = ['Low', 'Lower-Mid', 'Mid', 'Upper-Mid', 'High']

df['SalaryBand'] = pd.cut(df['MonthlyIncome'], bins=bins, labels=labels)

# Group by SalaryBand and Attrition
salary_attrition = df.groupby(['SalaryBand', 'Attrition']).size().unstack(fill_value=0)

# Calculate %
salary_attrition['Attrition %'] = (salary_attrition['Yes'] / (salary_attrition['Yes'] + salary_attrition['No'])) * 100

print(salary_attrition)


plt.figure(figsize=(8,5))
sns.barplot(x=salary_attrition.index, y=salary_attrition['Attrition %'], palette='magma')
plt.ylabel('Attrition Percentage')
plt.title('Attrition by Salary Band')
plt.tight_layout()
plt.show()




# Attrition vs YearsSinceLastPromotion
plt.figure(figsize=(10,6))
sns.boxplot(x='Attrition', y='YearsSinceLastPromotion', data=df, palette='coolwarm')
plt.title('Years Since Last Promotion by Attrition Status')
plt.tight_layout()
plt.show()



import seaborn as sns
import matplotlib.pyplot as plt

# Keep only numeric columns
numeric_df = df.select_dtypes(include=['int64', 'float64'])

# Correlation matrix
corr = numeric_df.corr()

# Heatmap
plt.figure(figsize=(12,10))
sns.heatmap(corr, annot=True, fmt=".2f", cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```

Activate Windows
Go to Settings to activate Windows.

Tools and Methodologies

Tools Used:
- Python: For data analysis, visualization, and model building (Pandas, Seaborn, Sklearn, SHAP)
- Power BI: For interactive visual dashboards
- Jupyter Notebook: Development environment

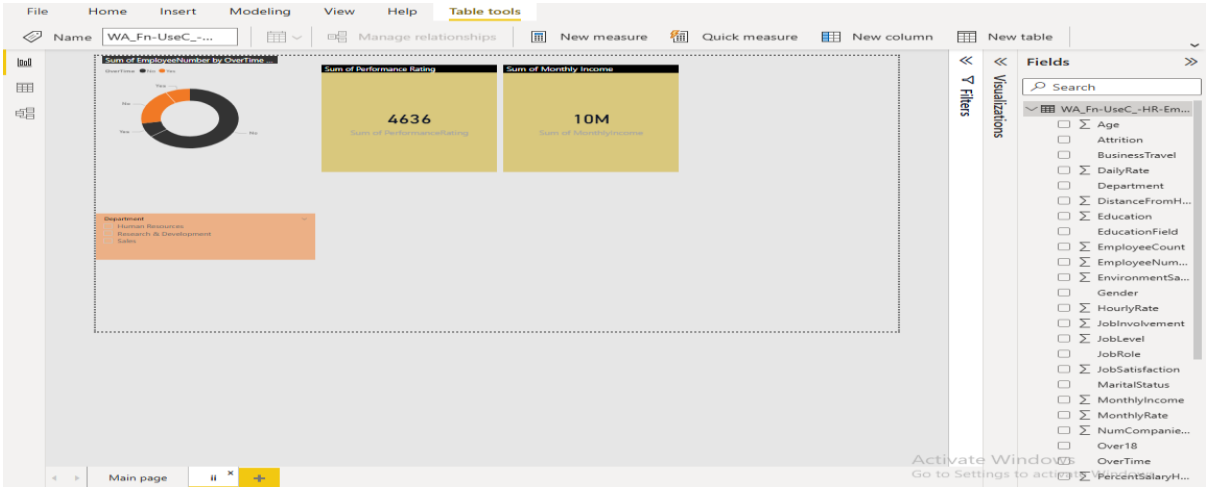
Methods Applied:
- Exploratory Data Analysis (EDA)
- Correlation Heatmaps
- Classification Models: Logistic Regression and Decision Tree
- SHAP Analysis for Feature Importance
- Evaluation: Accuracy Score, Confusion Matrix

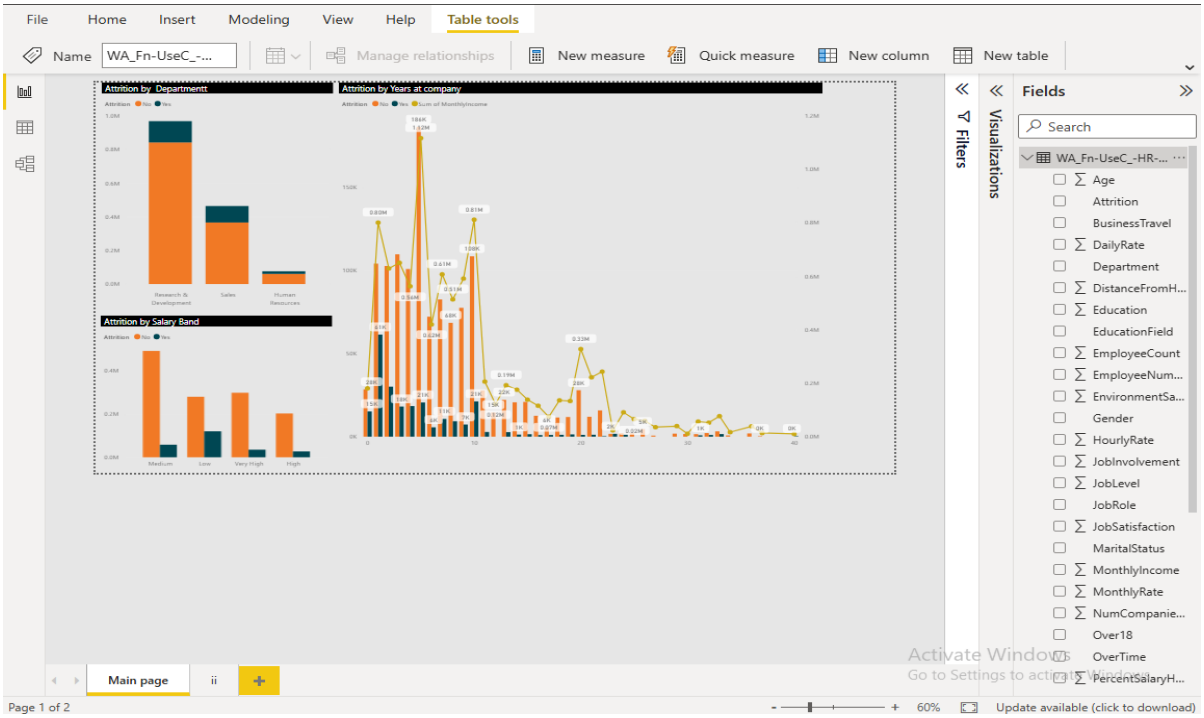
Exploratory Data Analysis (EDA)

- Attrition by Department: Highlights that some departments experience higher attrition.
- Attrition by Salary Band: Indicates low salary ranges contribute to higher turnover.
- Years Since Last Promotion: Shows employees without recent promotions are more likely to leave.
- Correlation Heatmap: Reveals strong relationships between attrition and variables like Overtime, Income, and Job Satisfaction.

Power BI Dashboard

The Power BI dashboard presents critical HR metrics such as overtime distribution, department-level filtering, performance scores, and overall salary statistics. This visualization empowers HR decision-makers to interactively explore attrition patterns and take timely actions.





Modeling Results

Target Variable: Attrition (Yes/No)
Models Used:
- Logistic Regression
- Decision Tree Classifier

Model Evaluation Metrics:
- Accuracy Score
- Confusion Matrix
- SHAP Analysis for interpretability

Top Predictors Identified:
- Overtime
- Monthly Income
- Job Satisfaction
- Years Since Last Promotion
- Work-Life Balance

Conclusion and Recommendations

This HR analytics project effectively identifies key drivers of employee attrition using statistical methods and predictive models. The integration of SHAP analysis made the model's predictions interpretable, supporting actionable HR insights.

Key Findings:
- Overtime and lack of promotion were top contributors to attrition.
- Employees with low monthly income and poor job satisfaction showed higher resignation rates.
- Power BI dashboards allow continuous tracking of HR KPIs for proactive planning.

Recommendations:
1. Reduce Overtime: Encourage balanced work hours and promote mental well-being.
2. Improve Compensation: Revise salary bands and offer performance bonuses.
3. Promote Career Growth: Create transparent promotion policies and learning programs.
4. Monitor Trends: Use dashboards regularly to identify early signs of disengagement.

With these interventions, organizations can build a more satisfied, loyal, and productive workforce.