www.kiet.edu
Delhi-NCR, Ghaziabad

KIET
GROUP OF INSTITUTIONS
Connecting Life with Learning

Assessment Report

on

“Employee Attrition Prediction”

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AIML)

By

Name : Abhay Gupta

- Ayush Patel
- Aditi Sharma
- Aryan Tomar
- Ankit Jaiswal

Roll Number : 202401100400004

202401100400065

2024011004000037

202401100400013

202401100400055

Section: A

Under the supervision of

“Bikki Kumar Gupta”

KIET Group of Institutions, Ghaziabad

May, 2025

1. Introduction

Employee attrition refers to the gradual loss of employees over time. Predicting whether an employee is likely to leave the company helps businesses reduce turnover and plan effective retention strategies. In this project, we use IBM HR Analytics data and machine learning to build a classification model that predicts attrition and highlights important contributing factors.

2. Problem Statement

To predict if an employee is likely to leave the company using IBM HR Analytics data. Focus on classification techniques and visualize feature importance.

3. Objectives

- Preprocess the dataset for training a machine learning model.
- Train a Logistic Regression model to classify loan defaults.
- Evaluate model performance using standard classification metrics.

- Visualize the confusion matrix using a heatmap for interpretability.

---

4. Methodology

Data Collection:

- Dataset used: IBM HR Analytics Employee Attrition & Performance
- Source: Kaggle (Link)

Data Preprocessing:

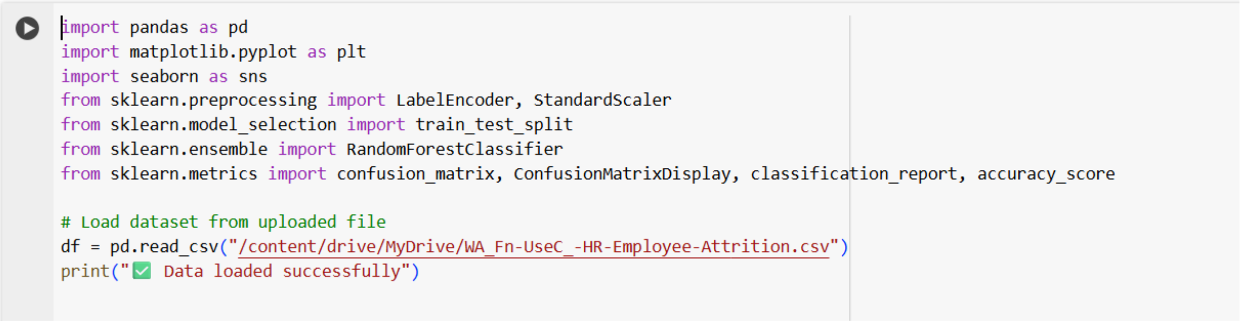
- Dropped irrelevant columns (EmployeeNumber, Over18, etc.)
- Encoded categorical columns using LabelEncoder
- Split data into features (X) and target (y)

Model Building:

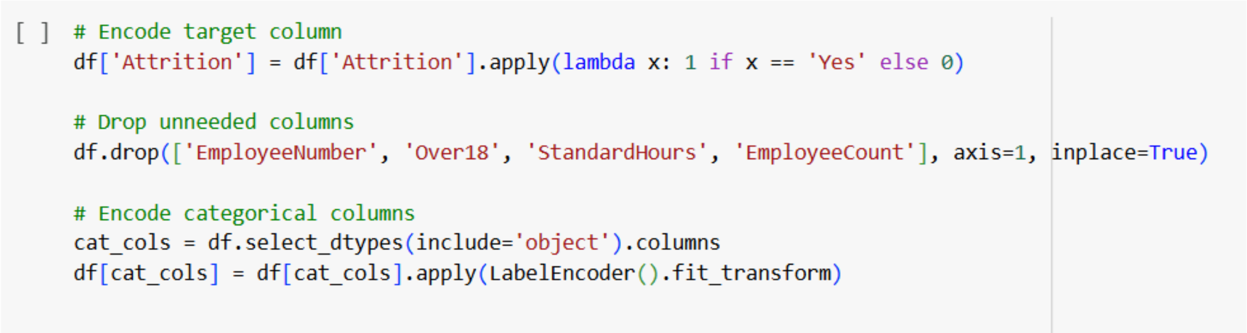
- Train-test split: 80% training, 20% testing
- Trained a RandomForestClassifier
- **Model Evaluation:**
- Used classification report and confusion matrix for evaluation
- Visualized top 10 feature importances

---

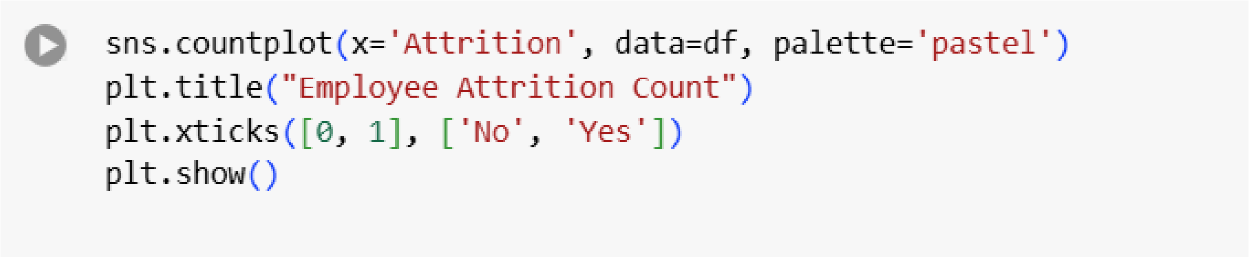
5. CODE



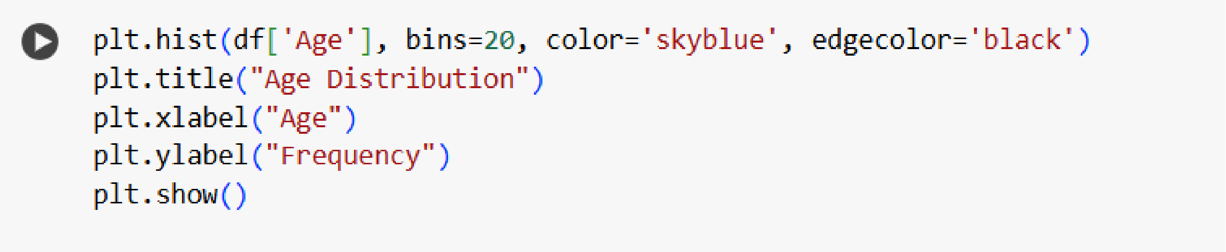
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay, classification_report, accuracy_score

# Load dataset from uploaded file
df = pd.read_csv("/content/drive/MyDrive/WA_Fn-UseC_-HR-Employee-Attrition.csv")
print("✅ Data loaded successfully")
```



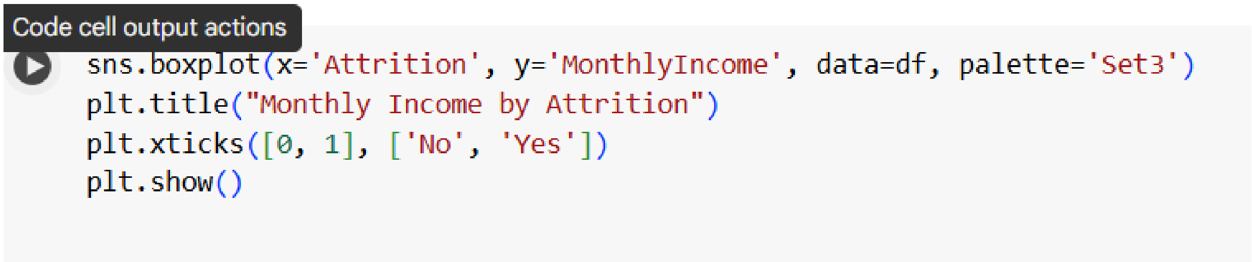
```
# Encode target column
df['Attrition'] = df['Attrition'].apply(lambda x: 1 if x == 'Yes' else 0)

# Drop unneeded columns
df.drop(['EmployeeNumber', 'Over18', 'StandardHours', 'EmployeeCount'], axis=1, inplace=True)

# Encode categorical columns
cat_cols = df.select_dtypes(include='object').columns
df[cat_cols] = df[cat_cols].apply(LabelEncoder().fit_transform)
```



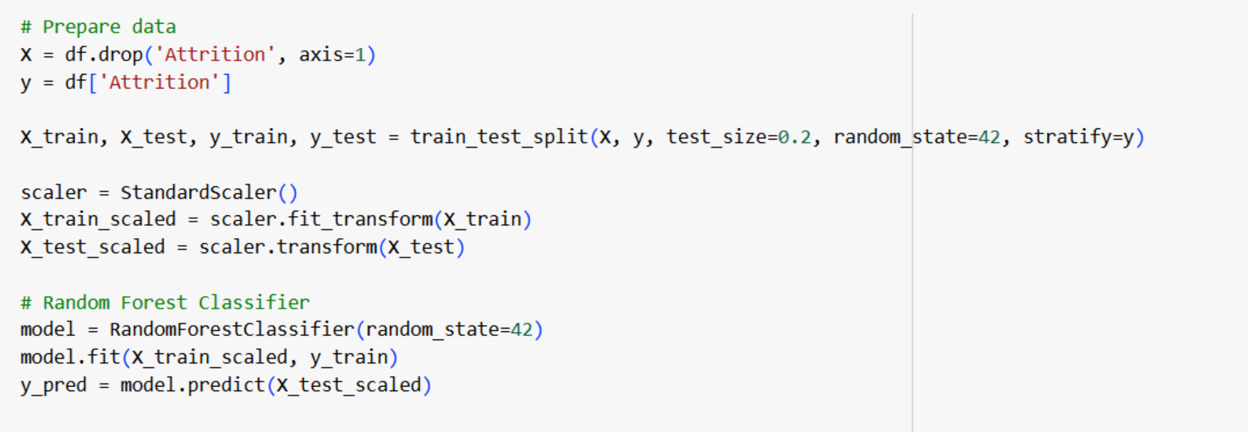
```
sns.countplot(x='Attrition', data=df, palette='pastel')
plt.title("Employee Attrition Count")
plt.xticks([0, 1], ['No', 'Yes'])
plt.show()
```



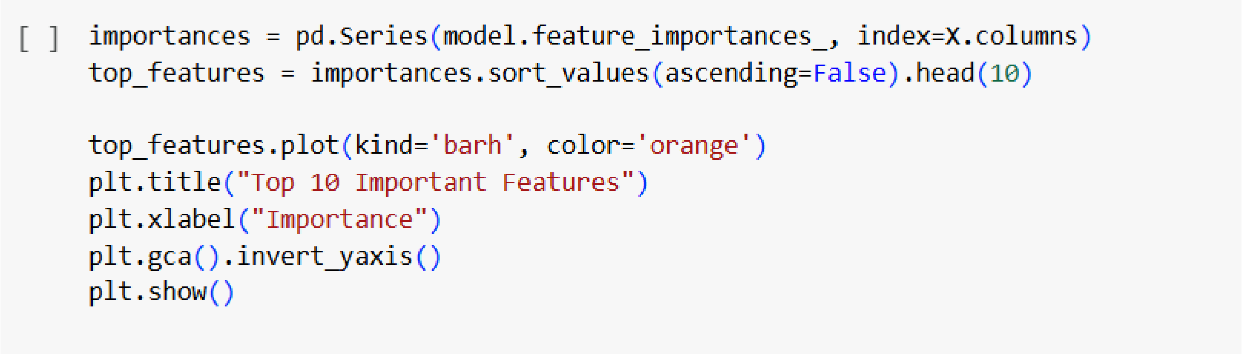
```
plt.hist(df['Age'], bins=20, color='skyblue', edgecolor='black')
plt.title("Age Distribution")
plt.xlabel("Age")
plt.ylabel("Frequency")
plt.show()
```



```
sns.boxplot(x='Attrition', y='MonthlyIncome', data=df, palette='Set3')
plt.title("Monthly Income by Attrition")
plt.xticks([0, 1], ['No', 'Yes'])
plt.show()
```



```
# Prepare data
X = df.drop('Attrition', axis=1)
y = df['Attrition']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Random Forest Classifier
model = RandomForestClassifier(random_state=42)
model.fit(X_train_scaled, y_train)
y_pred = model.predict(X_test_scaled)
```



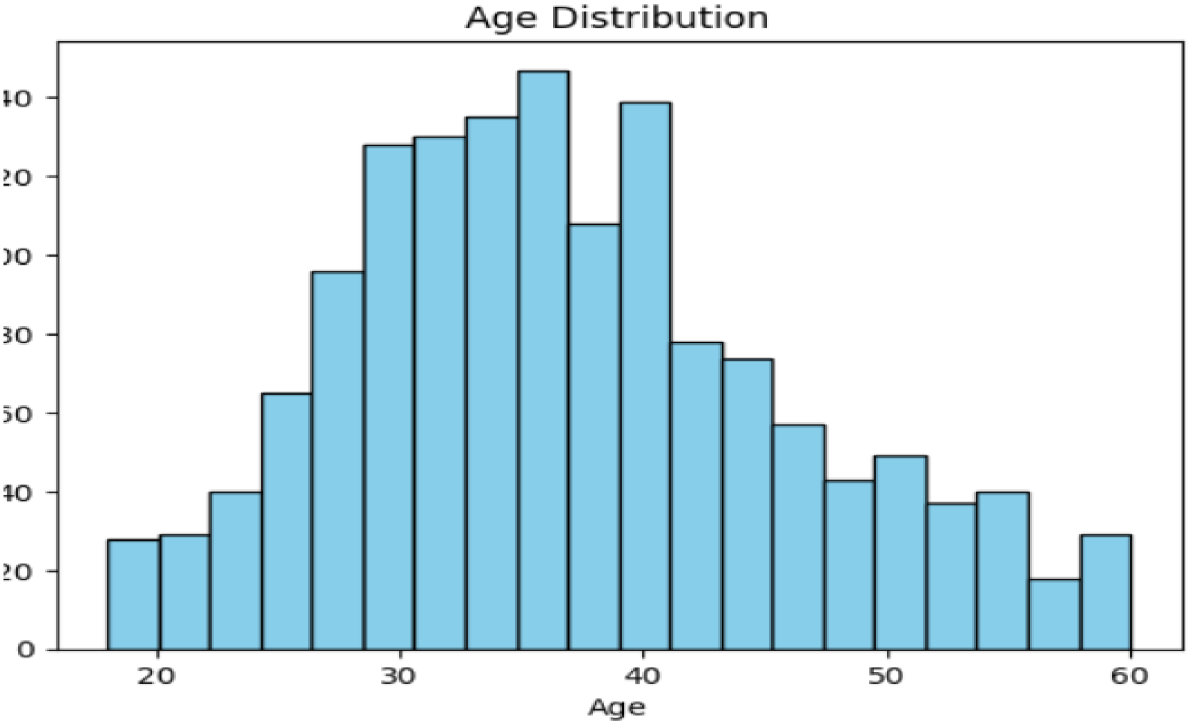
```
importances = pd.Series(model.feature_importances_, index=X.columns)
top_features = importances.sort_values(ascending=False).head(10)

top_features.plot(kind='barh', color='orange')
plt.title("Top 10 Important Features")
plt.xlabel("Importance")
plt.gca().invert_yaxis()
plt.show()
```

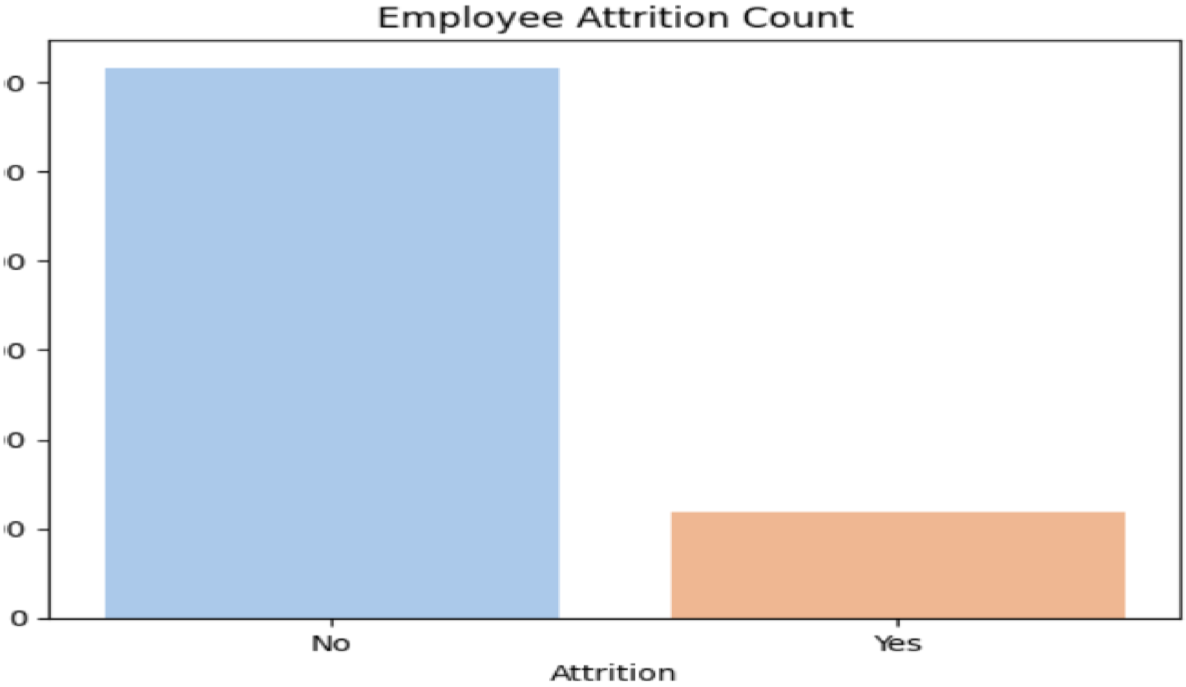
6. OUTPUT.

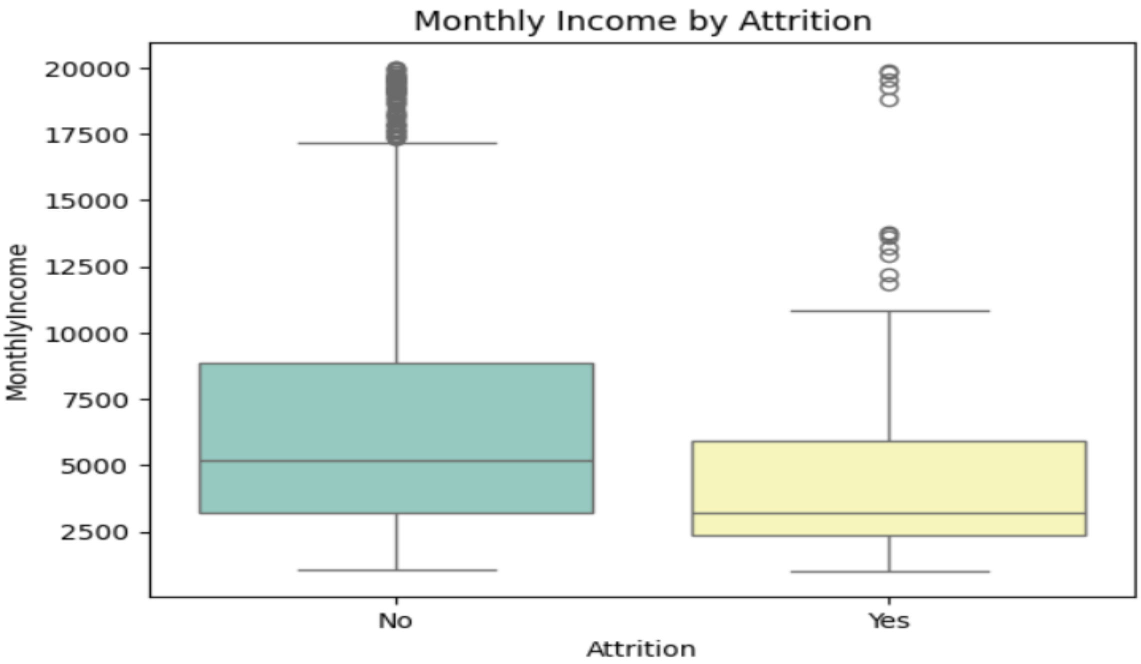
Model Evaluation Output:

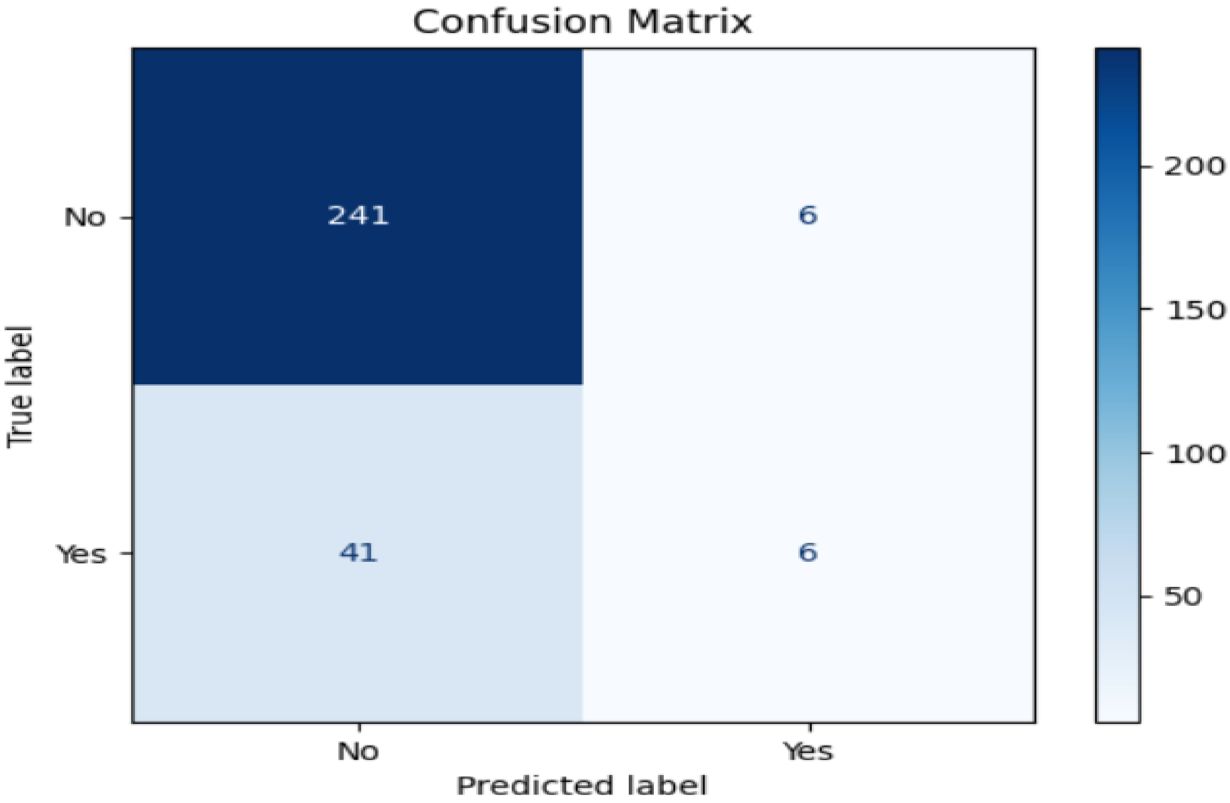
- Confusion Matrix and Classification Report printed in terminal
- Feature importance visualized using matplotlib

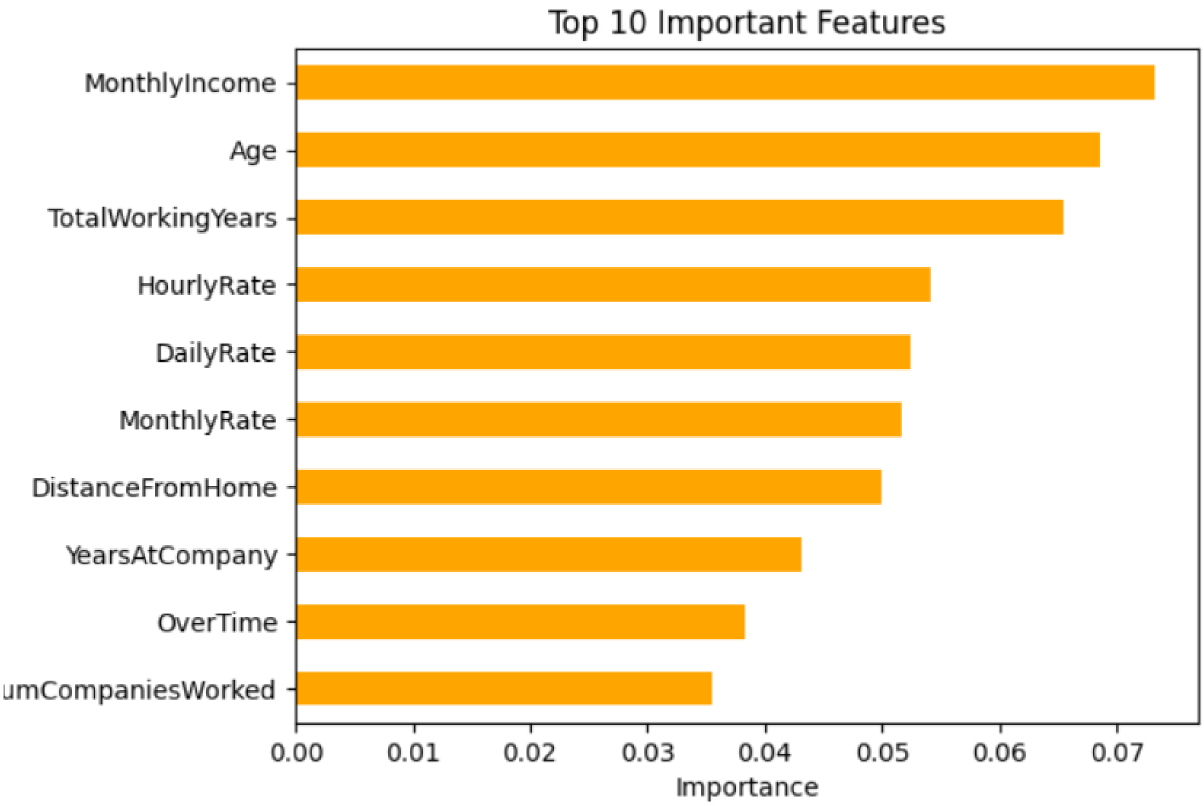
Screenshots:






Monthly Income by Attrition
MonthlyIncome
20000
17500
15000
12500
10000
7500
5000
2500
No
Yes
Attrition


Confusion Matrix
True label
No
Yes
241
6
41
6
No
Yes
Predicted label
200
150
100
50



Key Features Influencing Attrition:

- Job Role
- Monthly Income
- Job Satisfaction
- Years at Company
- Work-Life Balance

10. References

- Dataset: IBM HR Analytics, Kaggle
- Libraries: scikit-learn, pandas, matplotlib, seaborn
- Python documentation and tutoriaLS