IBM HR Analytics (Employee Attrition & Performance) - Exploratory Data Analysis (EDA) Presentation

Project made by Geetanjali Kaushik

Objective

To analyze employee attrition patterns and identify key factors influencing employee turnover using Exploratory Data Analysis.

Dataset Overview

The IBM HR Analytics dataset contains 1470 employee records with 35 features including Age, JobRole, MonthlyIncome, WorkLifeBalance, and Attrition.

Step 1: Import Libraries

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

These libraries are used for data manipulation, numerical computation, and visualization.

Explanation

- `pandas` → load + manipulate dataset (DataFrame)
- `numpy` → numerical operations
- `matplotlib` + `seaborn` → graphs
- `warnings.filterwarnings('ignore')` → hides warnings to keep notebook clean (good for presentation)

Step 2: Load Dataset

Code

```
IBM_HR_data = pd.read_csv(r"D:\...\WA_Fn-UseC_-HR-Employee-
Attrition.csv")
IBM_HR_data.head()
```

Output

- First 5 rows shown, dataset has **35 columns**

Explanation

- Reads the CSV into a DataFrame named `IBM_HR_data`
- `head()` helps confirm:dataset loaded correctly
- columns look correct
- values are readable

Step 3 : Column Names

Code

```
IBM_HR_data.columns
```

Output

- List of all columns (Age, Attrition, Department, MonthlyIncome, OverTime, etc.)

Explanation

- Confirms available features for analysis
- Helps you decide which columns are important for attrition (target)

Step 4: Dataset Info (Rows, Types, Memory)

Code

```
IBM_HR_data.info()
```

Output

- **1470 rows × 35 columns**
- **26 int64**, **9 object**
- **No null values**

Explanation

- Shows:
  - dataset size
  - data types (numeric vs categorical)
  - non-null counts (used to detect missing values)
- Here, dataset is clean in terms of missing values

Step 5: Missing Values Check

Code

```
IBM_HR_data.isnull().sum()
```

Output

- All columns show **0 missing values**

Explanation

- Confirms there are **no null/missing entries**
- So no imputation (fillna) required for this dataset

Step 6: Statistical Summary

Code

```
IBM_HR_data.describe()
```

Output

- Count, mean, std, min/max, quartiles for **numeric columns (26)**

Explanation

- Useful to understand:
  - salary ranges (`MonthlyIncome`)
  - age range (min 18 to max 60)
  - tenure distribution (`YearsAtCompany`)
- Helps spot unrealistic values or outliers

Step 7 : Dataset Shape + Duplicate Removal

Code

```
IBM_HR_data.shape
IBM_HR_data.drop_duplicates(inplace=True)
```

Output

- Shape shown: **(1470, 35)**
- Duplicate removal gives no printed output

Explanation

- `shape` confirms rows/columns
- `drop_duplicates()` removes exact duplicate records (good practice)
- In IBM dataset, duplicates are usually none, but this is still correct cleaning step

---

Step 8 : Attrition Rate (Normalized Count)

Code

```
IBM_HR_data['Attrition'].value_counts(normalize=True)
```

Output

- No = **0.8388 (~83.88%)**
- Yes = **0.1612 (~16.12%)**

Explanation

- Attrition rate is **~16%**
- This is an **imbalanced target variable** (fewer “Yes” cases)
- Important for modeling later (class imbalance handling)

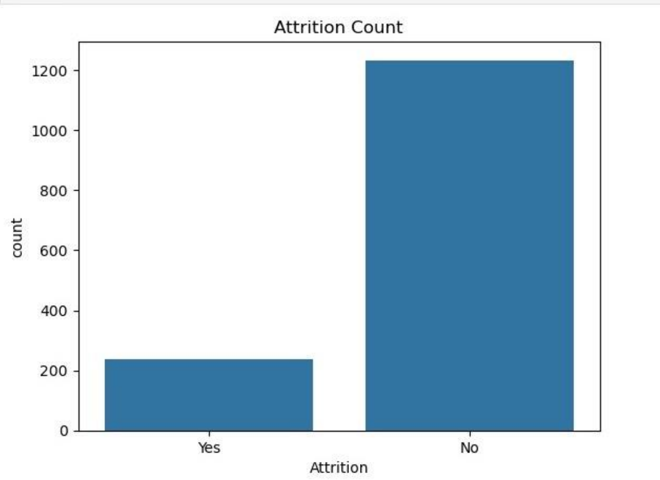
Step 9: Attrition Count Plot

Code

```
sns.countplot(x='Attrition', data=IBM_HR_data)
plt.title("Attrition Count")
plt.show()
```

Output

- Bar chart: “No” much higher than “Yes”



Explanation / Insight

- Confirms imbalance visually
- Business meaning: IBM has **more retained employees**, but **attrition still significant (~16%)**

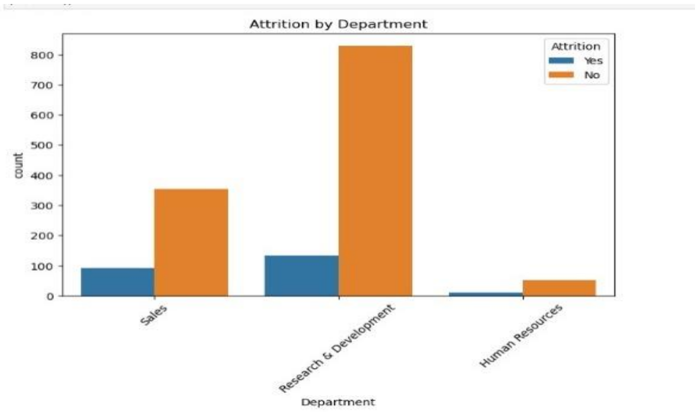
Step 10: Attrition by Department

Code

```
plt.figure(figsize=(8,5))
sns.countplot(data=IBM_HR_data, x='Department', hue='Attrition')
plt.title("Attrition by Department")
plt.xticks(rotation=45)
plt.show()
```

Output

- Department-wise bars split by Attrition



Explanation / Insight

- Compares attrition across departments
- Usually: **Sales** tends to show higher attrition compared to R&D
- HR action: investigate workload, targets, incentives department-wise

---

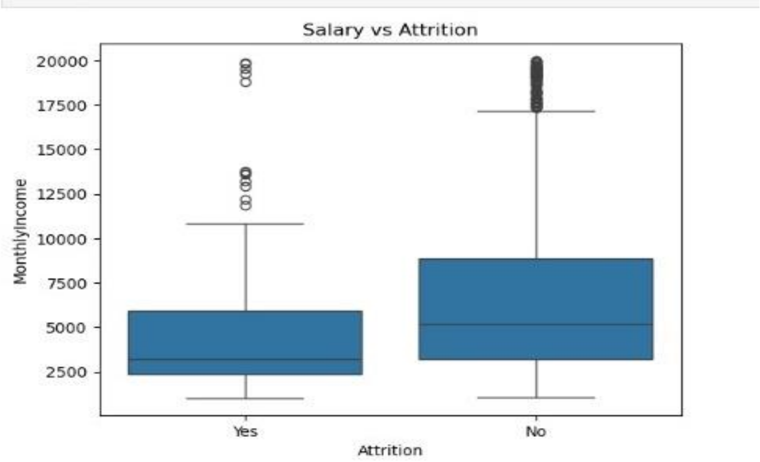
Step 11 : Salary (Monthly Income) vs Attrition

Code

```
sns.boxplot(x='Attrition', y='MonthlyIncome', data=IBM_HR_data)
plt.title("Salary vs Attrition")
plt.show()
```

Output

- Boxplot showing income distribution for Attrition Yes/No



Explanation / Insight

- Boxplot reveals:Attrition “Yes” group often has **lower median income**
- Business meaning: **lower pay band employees** are more likely to leave
- Recommendation: review compensation / growth plans for low-income ranges

---

Step 12 : Age vs Attrition (Count Plot)

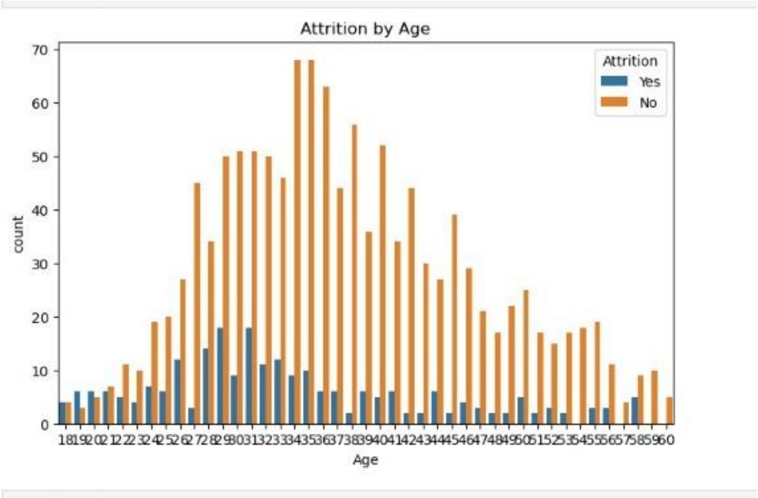
Code

```
plt.figure(figsize=(8,5))

sns.countplot(x='Age', hue='Attrition', data=IBM_HR_data)
plt.title("Attrition by Age")
plt.show()
```

Output

- Many thin bars (age-wise distribution)



Explanation

- Shows which exact ages have more attrition counts
- BUT this plot becomes crowded because Age has many unique values

Better presentation tip (optional for clean PPT): use Age Groups instead:

```
IBM_HR_data['AgeGroup'] = pd.cut(IBM_HR_data['Age'],
                                 bins=[17,30,40,50,60,100],
                                 labels=['18-30','31-40','41-50','51-
60','61+'])
```

```
sns.countplot(x='AgeGroup', hue='Attrition', data=IBM_HR_data)
```

Insight: highest attrition is usually **18–30** group.

Step 13 : Feature Engineering (Attrition Flag + Age Group)

Code

```
IBM_HR_data['AttritionFlag'] = IBM_HR_data['Attrition'].map({'Yes':1,
'No':0})
IBM_HR_data['AgeGroup'] = pd.cut(IBM_HR_data['Age'],
bins=[17,30,40,50,60,100],
                                 labels=['18-30','31-40','41-50','51-
60','61+'])
IBM_HR_data['TenureYears'] = IBM_HR_data['YearsAtCompany']
```

Output

- Table showing Attrition, AttritionFlag, Age, AgeGroup

Explanation

- Converts target into numeric:
    - Yes → 1, No → 0 (useful for statistics & ML)
- Creates AgeGroup for easier comparison
- TenureYears used for retention analysis

Step 14 : Overall Turnover + Group Breakdown Tables

Code

```
overall_turnover = IBM_HR_data['AttritionFlag'].mean()
print(f"Overall turnover (attrition) rate: {overall_turnover:.2%}")
```

Output

- Overall attrition rate: **16.12%**

Explanation

- Mean of 0/1 flag directly gives attrition rate

Step 15: Attrition Rate by Gender, Age Group, Education, Department, Job Role

Code

```
by_gender =
IBM_HR_data.groupby('Gender')['AttritionFlag'].agg(['count','sum','mean
'])
by_age =
IBM_HR_data.groupby('AgeGroup')['AttritionFlag'].agg(['count','sum','me
an'])
by_dept =
IBM_HR_data.groupby('Department')['AttritionFlag'].agg(['count','sum','
mean'])
by_role =
IBM_HR_data.groupby('JobRole')['AttritionFlag'].agg(['count','sum','mea
n'])
```

Output (from your tables)

- Gender: Male ~17.01%, Female ~14.80%
- AgeGroup: 18–30 has **~25.91%** (highest)
- Dept: Sales **~20.63%** (highest)
- JobRole: Sales Representative **~39.76%** (highest)

Explanation / Insight

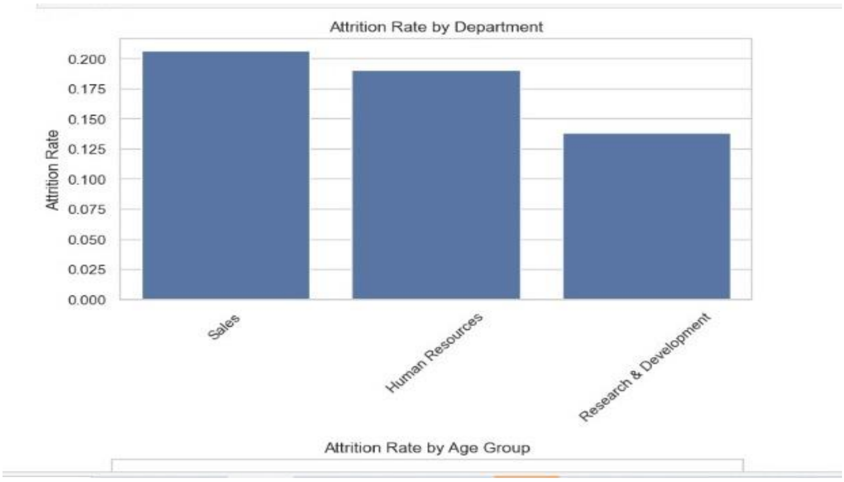
- Biggest attrition risk segments:
    - **Young employees (18–30)**
    - **Sales department**
    - **Sales Representative role**
- HR should prioritize retention actions here first

Step 16 : Visualization Graphs

16A) Attrition Rate by Department

Code

```
sns.barplot(data=by_dept.sort_values('mean', ascending=False),
x='Department', y='mean')
```



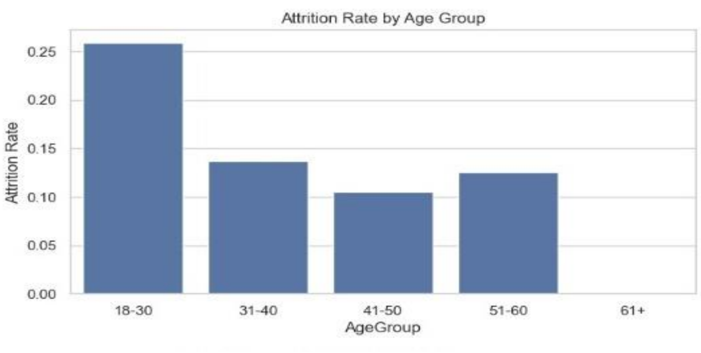
Insight

- Sales shows highest attrition rate → role pressure/targets may be driver

16B) Attrition Rate by AgeGroup

Code

```
sns.barplot(data=by_age.sort_values('mean', ascending=False),
x='AgeGroup', y='mean')
```



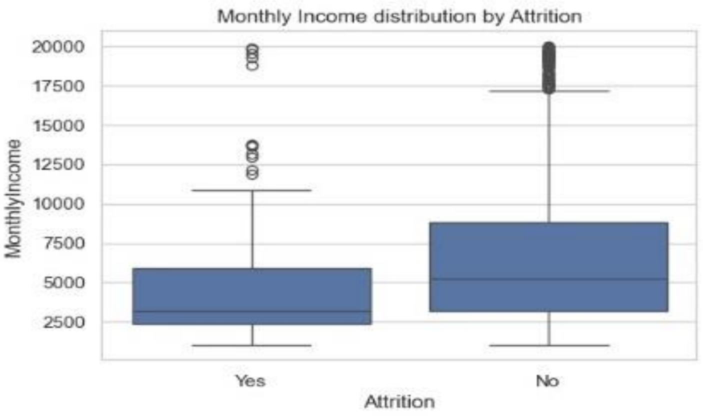
Insight

- 18–30 highest attrition → early career switching / growth expectations

16C) Monthly Income by Attrition

Code

```
sns.boxplot(x='Attrition', y='MonthlyIncome', data=IBM_HR_data)
```



Insight

- Lower income linked with higher attrition → compensation strategy important

---

Key Insights

• Attrition rate is ~16%
• Younger and lower-paid employees are more likely to leave
• Job role and work-life balance significantly impact attrition

Recommendations

• Improve onboarding programs
• Review compensation for high-risk roles
• Enhance work-life balance initiatives

Conclusion

EDA successfully identified major drivers of attrition and provides actionable insights for HR decision-making.

Thank you for review my presentation ☺