**HR Analytics**

Description: Analyze employee data to understand patterns in attrition, department-wise performance, promotion timelines, and salary distribution. The project is useful for HR departments to retain talent and build data-driven employee policies.

## **Project Title: HR Analytics for Employee Retention and Performance**

### **1. Project Structure**

HR-Analytics-Project/  
│  
├── data/  
│ └── employee\_data.csv  
│  
├── notebooks/  
│ └── HR\_Analytics\_Analysis.ipynb  
│  
├── README.md  
└── requirements.txt

### **2. Dataset**

**Dataset Name**: [Employee Attrition and Performance Dataset (Kaggle)](https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset)

**Download and place it as**: data/employee\_data.csv

### **3. Python Requirements (requirements.txt)**

pandas  
matplotlib  
seaborn  
numpy  
jupyter

### **4. Step-by-Step Notebook Outline: HR\_Analytics\_Analysis.ipynb**

#### **Step 1: Import Libraries**

import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns

#### **Step 2: Load the Dataset**

df = pd.read\_csv('../data/employee\_data.csv')  
df.head()

#### **Step 3: Data Cleaning and Preprocessing**

# Check for missing values  
df.isnull().sum()  
  
# Drop irrelevant columns  
df = df.drop(['EmployeeCount', 'Over18', 'StandardHours', 'EmployeeNumber'], axis=1)  
  
# Encode categorical variables if needed  
df['Attrition'] = df['Attrition'].map({'Yes': 1, 'No': 0})

#### **Step 4: EDA – Exploratory Data Analysis**

**Attrition Rate**

sns.countplot(x='Attrition', data=df)  
plt.title("Employee Attrition Count")

**Department-wise Attrition**

sns.countplot(x='Department', hue='Attrition', data=df)  
plt.title("Attrition by Department")

**Promotion Timeline**

sns.histplot(df['YearsSinceLastPromotion'], kde=True)  
plt.title("Years Since Last Promotion")

**Salary Distribution**

sns.boxplot(x='JobLevel', y='MonthlyIncome', data=df)  
plt.title("Salary Distribution by Job Level")

### **5. Insights and Recommendations**

* Departments with the highest attrition.
* Employees with long gaps in promotions are at higher risk.
* Salary disparities between job levels may influence turnover.

### **6. README.md Sample**

# HR Analytics for Employee Retention and Performance  
  
**## Description**This project analyzes employee data to understand attrition, departmental trends, salary insights, and promotion timelines.  
  
**## Dataset**- Source: [Kaggle - IBM HR Analytics](https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset)  
  
**## Goals**- Understand patterns of employee attrition.  
- Analyze department-wise performance.  
- Explore salary and promotion trends.  
  
**## Tools**- Python (Pandas, Seaborn, Matplotlib)  
- Jupyter Notebook  
  
**## Usage**Run `HR\_Analytics\_Analysis.ipynb` from the notebooks folder.

**HR Analytics project code** in a Python script format (hr\_analytics\_analysis.py) that you can directly use or upload to GitHub:

### **Filename: hr\_analytics\_analysis.py**

# HR Analytics Analysis Script  
  
**# Step 1: Import Libraries**import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
import os  
  
**# Step 2: Load the Dataset**DATA\_PATH = 'data/employee\_data.csv'  
  
if not os.path.exists(DATA\_PATH):  
 raise FileNotFoundError(f"Dataset not found at {DATA\_PATH}. Please make sure it's placed correctly.")  
  
df = pd.read\_csv(DATA\_PATH)  
print("Initial Data Loaded:")  
print(df.head())  
  
**# Step 3: Data Cleaning and Preprocessing**print("\nMissing Values:\n", df.isnull().sum())  
  
**# Drop irrelevant columns**columns\_to\_drop = ['EmployeeCount', 'Over18', 'StandardHours', 'EmployeeNumber']  
df.drop(columns=columns\_to\_drop, axis=1, inplace=True)  
  
# Encode Attrition  
df['Attrition'] = df['Attrition'].map({'Yes': 1, 'No': 0})  
  
**# Step 4: Exploratory Data Analysis (EDA)**  
# 4.1 Attrition Count  
plt.figure(figsize=(6, 4))  
sns.countplot(x='Attrition', data=df)  
plt.title("Employee Attrition Count")  
plt.savefig("output/attrition\_count.png")  
plt.clf()  
  
**# 4.2 Department-wise Attrition**plt.figure(figsize=(8, 5))  
sns.countplot(x='Department', hue='Attrition', data=df)  
plt.title("Attrition by Department")  
plt.xticks(rotation=45)  
plt.tight\_layout()  
plt.savefig("output/department\_attrition.png")  
plt.clf()  
  
**# 4.3 Years Since Last Promotion**plt.figure(figsize=(6, 4))  
sns.histplot(df['YearsSinceLastPromotion'], kde=True, bins=20)  
plt.title("Years Since Last Promotion")  
plt.tight\_layout()  
plt.savefig("output/promotion\_gap.png")  
plt.clf()  
  
**# 4.4 Salary Distribution by Job Level**plt.figure(figsize=(8, 5))  
sns.boxplot(x='JobLevel', y='MonthlyIncome', data=df)  
plt.title("Salary Distribution by Job Level")  
plt.tight\_layout()  
plt.savefig("output/salary\_by\_joblevel.png")  
plt.clf()  
  
print("EDA charts saved in 'output' folder.")  
  
**# Step 5: Summary**print("\nSummary Insights:")  
print("- High attrition in certain departments.")  
print("- Long gaps in promotions may influence attrition.")  
print("- Salary distribution varies significantly across job levels.")

### **Folder Structure**

HR-Analytics-Project/  
├── data/  
│ └── employee\_data.csv  
├── output/  
│ └── attrition\_count.png  
│ └── department\_attrition.png  
│ └── promotion\_gap.png  
│ └── salary\_by\_joblevel.png  
├── hr\_analytics\_analysis.py  
├── requirements.txt  
└── README.md

### **Next Steps**

* Place the employee\_data.csv file in the data/ folder.
* Create the output/ folder before running the script to save the charts.
* Run the script using:

python hr\_analytics\_analysis.py

The **step-by-step instructions to upload your HR Analytics project to GitHub**:

## **Step 1: Create a GitHub Repository**

1. Go to <https://github.com>.
2. Click **+** in the top-right corner and select **"New repository"**.
3. Fill in the details:
   * **Repository name**: HR-Analytics-Project
   * **Description**: Analyze employee data to uncover attrition, performance, and salary trends.
   * **Public/Private**: Choose based on your preference.
   * **Initialize with a README**: Optional (you can also add it later).
4. Click **Create repository**.

## **Step 2: Prepare Your Project Folder Locally**

Organize your project folder like this:

HR-Analytics-Project/  
├── data/  
│ └── employee\_data.csv # (not recommended to upload large or private datasets)  
├── output/ # folder for saved plots (optional)  
├── HR\_Analytics\_Analysis.ipynb  
├── hr\_analytics\_analysis.py  
├── README.md  
└── requirements.txt

## **Step 3: Initialize Git and Push Code**

Open **Terminal (Linux/macOS) or Git Bash (Windows)**:

# Navigate to your project directory  
cd path/to/HR-Analytics-Project  
  
**# Initialize Git**git init  
  
**# Link your GitHub repo**git remote add origin https://github.com/your-username/HR-Analytics-Project.git  
  
**# Add and commit files**git add .  
git commit -m "Initial commit with HR Analytics code and notebook"  
  
**# Push to GitHub**git branch -M main  
git push -u origin main

## **Step 4: Add README.md (Optional)**

Here’s a sample:

**# HR Analytics Project**Analyze employee data to uncover trends in attrition, department performance, promotion timelines, and salary distribution.  
  
**## Dataset**- [IBM HR Analytics Dataset on Kaggle](https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset)  
  
**## Files**- `HR\_Analytics\_Analysis.ipynb` – Jupyter notebook with step-by-step analysis.  
- `hr\_analytics\_analysis.py` – Script version of the analysis.  
- `requirements.txt` – Python packages required to run this project.  
  
**## Instructions**Clone the repo, place the dataset in the `data/` folder, and run the notebook or script.