Title Page

Project Title: Healthcare Data Exploration

Name: Raamanjal Singh Gangwar

Roll No: 2024011003001187

Class: Computer Science & Engineering (AI)-C

Introduction

This project aims to analyze a healthcare dataset using Exploratory Data Analysis (EDA) techniques. EDA helps in uncovering patterns, identifying anomalies, and understanding relationships among variables. By visualizing key health indicators such as age, blood pressure, sugar level, and weight, this project offers valuable insights into the dataset. Using Python and its libraries like Pandas, Matplotlib, and Seaborn, we explore the dataset effectively. The analysis lays a strong foundation for further predictive modeling and supports informed decision-making in the healthcare sector.

Methodology

We have used Python programming and its popular libraries such as Pandas, Matplotlib, and Seaborn for data analysis and visualization. The steps followed are:

1-Loading the Dataset: Using Pandas to read CSV files.

2-Previewing the Data: Displaying the first few records to understand data structure.

3-Basic Information: Using info() to understand data types and null values.

4-Descriptive Statistics: Using describe() to explore statistical properties.

5-Visual Analysis:

Distribution plots using Seaborn

Correlation Matrix (Heatmap)

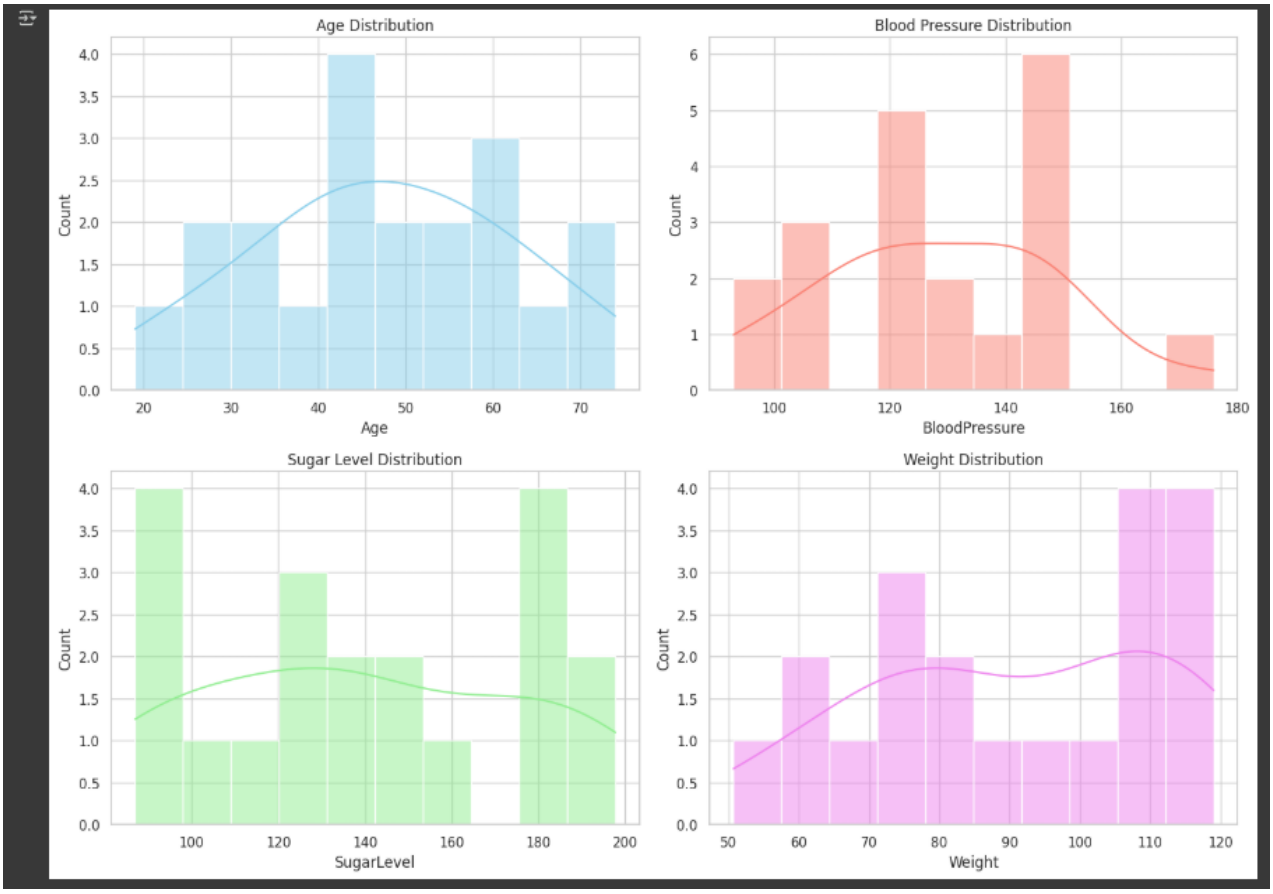
Scatter Plot for multivariable relationships

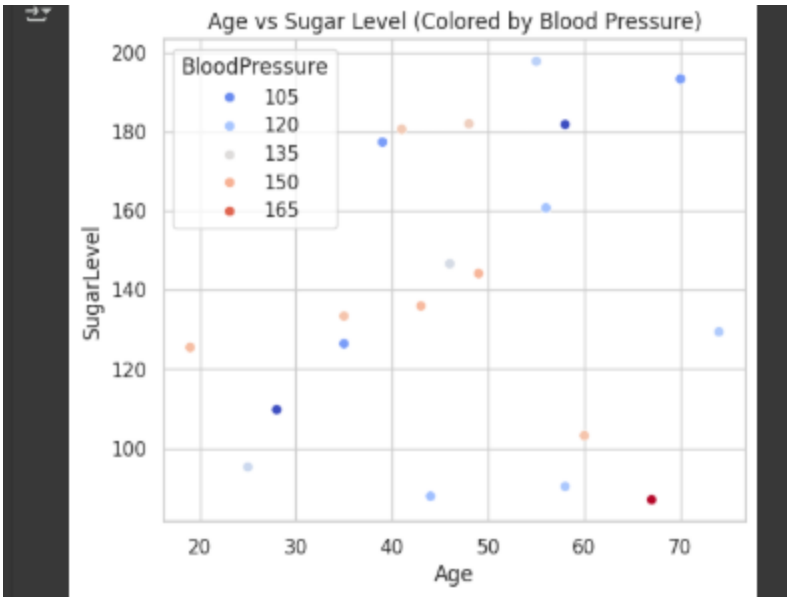
Code Implemented

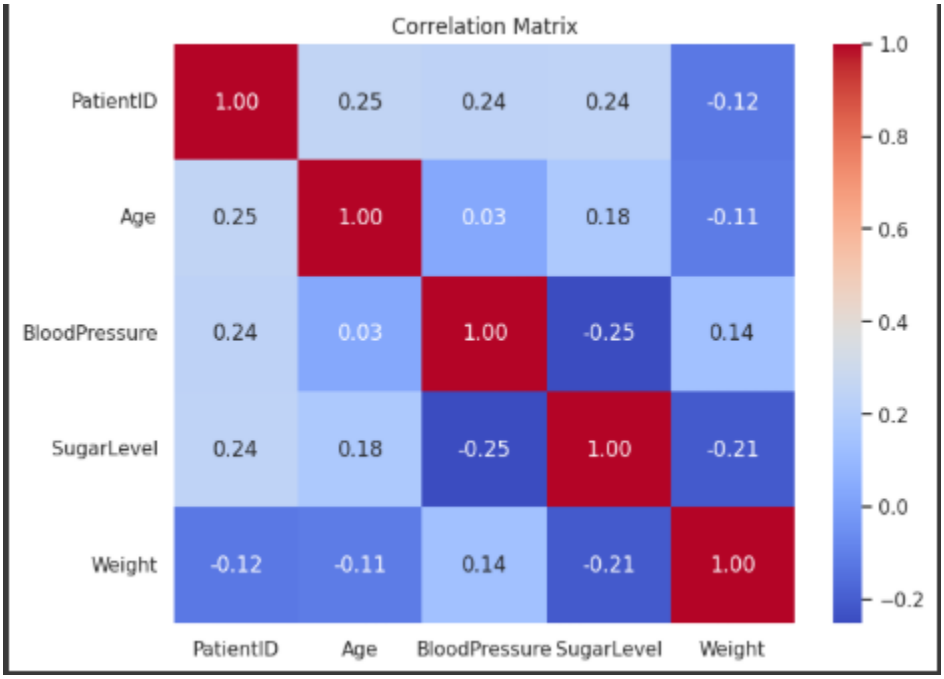
```
# Import libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Set seaborn style
sns.set(style="whitegrid")
# Upload your healthcare_data.csv file
df = pd.read_csv('/content/healthcare_data.csv')
# Preview the dataset after loading
print("=== Preview of Dataset ===")
display(df.head())
# Basic info and summary
print("=== Basic Information ===")
print(df.info())
print()
# Display the first five records of the dataset
print("\n=== First 5 Records ===")
print(df.head())
print()
 # Display summary statistics of numerical columns
print("\n=== Summary Statistics ===")
print(df.describe())
# Distribution plots
plt.figure(figsize=(14, 10))

 # Age distribution
plt.subplot(2, 2, 1)
sns.histplot(df['Age'], bins=10, kde=True, color='skyblue')
plt.title('Age Distribution')

# Blood Pressure distribution
plt.subplot(2, 2, 2)
sns.histplot(df['BloodPressure'], bins=10, kde=True, color='salmon')
plt.title('Blood Pressure Distribution')

# Sugar Level distribution
plt.subplot(2, 2, 3)
sns.histplot(df['SugarLevel'], bins=10, kde=True, color='lightgreen')
plt.title('Sugar Level Distribution')
```

```
# Weight distribution
plt.subplot(2, 2, 4)
sns.histplot(df['Weight'], bins=10, kde=True, color='violet')
plt.title('Weight Distribution')

# Adjust spacing between plots
plt.tight_layout()
plt.show()
# Correlation matrix heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Correlation Matrix")
plt.show()
# Scatter plot: Age vs Sugar Level
plt.figure(figsize=(6, 5))
sns.scatterplot(data=df, x='Age', y='SugarLevel', hue='BloodPressure',
palette='coolwarm')
plt.title("Age vs Sugar Level (Colored by Blood Pressure)")
plt.show()
```

Output


Age Distribution
Count
Age
Blood Pressure Distribution
Count
BloodPressure
Sugar Level Distribution
Count
SugarLevel
Weight Distribution
Count
Weight


Age vs Sugar Level (Colored by Blood Pressure)
BloodPressure
105
120
135
150
165
SugarLevel
200
180
160
140
120
100
20
30
40
50
60
70
Age


Correlation Matrix
PatientID
Age
BloodPressure
SugarLevel
Weight
1.00 0.25 0.24 0.24 -0.12
0.25 1.00 0.03 0.18 -0.11
0.24 0.03 1.00 -0.25 0.14
0.24 0.18 -0.25 1.00 -0.21
-0.12 -0.11 0.14 -0.21 1.00
PatientID
Age
BloodPressure
SugarLevel
Weight
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

Reference

CSV file- Healthcare.csv

Code-
https://colab.research.google.com/drive/1gA50dgvanhhMxFlFAahR584rSOvY56BR?usp=sharing

Screenshots-
https://colab.research.google.com/drive/1gA50dgvanhhMxFlFAahR584rSOvY56BR?usp=sharing