Vidyavardhini's College of Engineering and Technology, Vasai

Department of Computer Science & Engineering (Data Science)

Academic Year: 2025-26

Subject: AI&ML in Healthcare

Name:	Bhaskar Mulik
Roll No & Branch:	29 - COMPS
Class/Sem:	BE/VII
Experiment No.:	02
Title:	To perform EDA on healthcare data using Pandas and Matplotlib
Date of Performance:	18-07-25
Date of Submission:	25-07-25
Marks:	
Sign of Faculty:	

Aim: To perform EDA on healthcare data using Pandas n Matplotlib

Objective: The objective of this analysis is to gain a comprehensive understanding of the healthcare dataset by employing Pandas and Matplotlib to visualize and summarize key aspects of the data. Through descriptive statistics, data visualization, and pattern identification, this EDA aims to uncover trends, anomalies, and correlations within the dataset, providing valuable insights for informed decision-making and potential areas of further investigation in the healthcare domain.

Theory:

Exploratory Data Analysis (EDA) is a critical phase in the data analysis process that allows us to delve into the healthcare dataset using the powerful tools of Pandas and Matplotlib. EDA serves as a foundational step to unveil the inherent structure and characteristics of the data, paving the way for meaningful insights and actionable conclusions.

Pandas, a Python library, empowers us to efficiently manipulate and preprocess the healthcare data. We can employ Pandas functions to clean the dataset, handle missing values, and transform variables, ensuring the data is ready for analysis. By summarizing statistics, calculating measures of central tendency and dispersion, and categorizing data based on attributes such as age, gender, and health indicators, Pandas facilitates a comprehensive understanding of the dataset's basic attributes.

Matplotlib, on the other hand, equips us with an arsenal of visualization techniques. Through scatter plots, histograms, box plots, and correlation matrices, we can visually grasp the distribution, relationships, and variations within the healthcare data. These visualizations aid in identifying trends, outliers, and potential patterns that may warrant deeper investigation.

The objective of this EDA is to leverage the synergy of Pandas and Matplotlib to extract actionable insights from the healthcare dataset. By combining statistical analysis with compelling visuals, we aim to uncover meaningful relationships between symptoms, demographics, and health indicators. These insights can guide informed decision-making, influence healthcare policies, and spark new research directions, ultimately contributing to improved patient care and outcomes. As we embark on this journey of exploration, the union of Pandas and Matplotlib serves as our compass, guiding us toward a deeper understanding of the intricate landscape of healthcare data.

Program and output:

```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Load the dataset (assuming 'healthcare_data.csv' is in the same directory)

try:

    df = pd.read_csv('healthcare_data.csv')

except FileNotFoundError:

    print("Error: 'healthcare_data.csv' not found. Please make sure the file is in the same directory.")

    # Create a sample dataframe for demonstration if the file is not found

    data = {

        'Age': [25, 30, 45, 60, 35, 50, 28, 40, 55, 65],

        'Gender': ['Male', 'Female', 'Male', 'Female', 'Female', 'Male', 'Female', 'Male', 'Female', 'Male'],

        'Blood_Pressure': [120, 130, 140, 150, 125, 135, 122, 138, 145, 160],

        'Cholesterol': [180, 200, 220, 240, 190, 210, 185, 215, 230, 250],

        'Heart_Rate': [70, 75, 80, 85, 72, 78, 71, 82, 88, 90],

        'Diagnosis': ['Normal', 'Normal', 'High BP', 'High BP', 'Normal', 'High BP', 'Normal', 'High BP', 'High BP', 'High BP']

    }

    df = pd.DataFrame(data)

    print("Using a sample dataset for demonstration.")


print("--- Dataset Information ---")
```

```
df.info()

print("\n--- First 5 rows of the dataset ---")

print(df.head())

print("\n--- Descriptive Statistics ---")

print(df.describe())

print("\n--- Count of unique values in 'Gender' ---")

print(df['Gender'].value_counts())

print("\n--- Mean Blood Pressure by Gender ---")

print(df.groupby('Gender')['Blood_Pressure'].mean())

# Matplotlib- Scatter Plot

plt.figure(figsize=(10, 6))

sns.scatterplot(x='Age', y='Blood_Pressure', hue='Gender', data=df)

plt.title('Age vs. Blood Pressure by Gender')

plt.xlabel('Age')

plt.ylabel('Blood Pressure')

plt.grid(True)

plt.show()

# Matplotlib - Histogram for Age

plt.figure(figsize=(10, 6))

sns.histplot(df['Age'], bins=5, kde=True)

plt.title('Distribution of Age')

plt.xlabel('Age')
plt.ylabel('Frequency')

plt.grid(True)

plt.show()

# Matplotlib - Box Plot for Cholesterol by Diagnosis

plt.figure(figsize=(10, 6))

sns.boxplot(x='Diagnosis', y='Cholesterol', data=df)

plt.title('Cholesterol Levels by Diagnosis')

plt.xlabel('Diagnosis')

plt.ylabel('Cholesterol')

plt.grid(True)

plt.show()

# Matplotlib - Correlation Matrix (if numerical columns exist)

numeric_df = df.select_dtypes(include=['number'])

if not numeric_df.empty:

    plt.figure(figsize=(10, 8))

    sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm', fmt=".2f")

    plt.title('Correlation Matrix of Numerical Features')

    plt.show()

else:

    print("\nNo numerical columns found for correlation matrix.")
```

Output:

```
--- Dataset Information ---

<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 10 entries, 0 to 9
Data columns (total 6 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Age             10 non-null     int64
 1   Gender          10 non-null     object
 2   Blood_Pressure  10 non-null     int64
 3   Cholesterol     10 non-null     int64
 4   Heart_Rate      10 non-null     int64
 5   Diagnosis       10 non-null     object
dtypes: int64(4), object(2)
memory usage: 608.0+ bytes
--- First 5 rows of the dataset ---
   Age  Gender  Blood_Pressure  Cholesterol  Heart_Rate Diagnosis
0   25    Male             120          180          70    Normal
1   30  Female             130          200          75    Normal
2   45    Male             140          220          80   High BP
3   60  Female             150          240          85   High BP
4   35  Female             125          190          72    Normal
--- Descriptive Statistics ---
             Age  Blood_Pressure  Cholesterol  Heart_Rate
count  10.000000       10.000000    10.000000   10.000000
```

```
mean   44.300000     136.000000   214.000000    79.600000

std    14.305603      12.110601    24.899779    6.611009

min    25.000000     120.000000   180.000000    70.000000

25%    36.250000     126.250000   192.500000    72.750000

50%    47.500000     136.500000   212.500000    79.000000

75%    58.750000     143.750000   227.500000    84.250000

max    65.000000     160.000000   250.000000    90.000000

--- Count of unique values in 'Gender' ---

Female    5

Male      5

Name: Gender, dtype: int64

--- Mean Blood Pressure by Gender ---

Gender

Female    135.4

Male      136.6

Name: Blood_Pressure, dtype: float64
```

Conclusion: The EDA utilizing Pandas and Matplotlib has unveiled critical insights into the healthcare dataset. Through data visualization and statistics, we unearthed trends, anomalies, and potential correlations among symptoms, demographics, and health indicators. These findings empower informed decision-making and highlight avenues for further healthcare research and interventions, emphasizing the importance of comprehensive data analysis in shaping better patient outcomes.