Experiment No 2

Aim: Perform Exploratory Data Analysis on Healthcare dataset.

Theory:

What is EDA?

Exploratory Data Analysis (EDA) is an approach to analyzing data to summarize its main characteristics, often using visual methods. It is a crucial step in the data science process, allowing data analysts and scientists to understand the structure, patterns, and relationships within the data. EDA helps to identify problems, opportunities, and insights that can inform further analysis, modeling, and decision-making.

Key Aspects of EDA

1. **Data Visualization**: Using plots, charts, and other visualizations to understand the distribution, relationships, and patterns in the data.
2. **Summary Statistics**: Calculating statistical measures, such as means, medians, modes, and standard deviations, to understand the central tendency and variability of the data.
3. **Data Transformation**: Transforming data into a suitable format for analysis, such as handling missing values, encoding categorical variables, and scaling/normalizing data.
4. **Data Quality Check**: Identifying and addressing issues with data quality, such as outliers, duplicates, and inconsistencies.
5. **Feature Engineering**: Creating new features or variables from existing ones to improve the analysis or modeling process.

Why EDA is Important

1. **Understanding Data**: EDA helps to gain a deep understanding of the data, its structure, and its relationships.
2. **Identifying Problems**: EDA can reveal issues with data quality, inconsistencies, and errors.
3. **Informing Modeling**: EDA informs the development of machine learning models by identifying relevant features, relationships, and patterns.
4. **Improving Decision-Making**: EDA provides insights that can inform business decisions, strategic planning, and policy-making.
5. **Reducing Costs**: EDA can help reduce costs by identifying areas where data collection or processing can be optimized.

Types of EDA

1. **Univariate Analysis**: Analyzing individual variables or features to understand their distribution, central tendency, and variability.

2. **Bivariate Analysis**: Analyzing the relationship between two variables or features to identify correlations, patterns, and relationships.
3. **Multivariate Analysis**: Analyzing the relationships between multiple variables or features to identify complex patterns and interactions.
4. **Descriptive Analytics**: Using EDA to describe the basic features of the data, such as summarizing demographic information.
5. **Inferential Analytics**: Using EDA to make inferences about a population based on a sample of data.
6. **Predictive Analytics**: Using EDA to identify patterns and relationships that can be used to make predictions about future events or outcomes.

Some common EDA techniques include:

- Histograms and density plots
- Scatter plots and correlation analysis
- Box plots and outlier detection
- Heatmaps and clustering analysis
- Principal Component Analysis (PCA) and dimensionality reduction
- Correlation matrices and feature selection

By applying these techniques, data analysts and scientists can gain a deeper understanding of their data, identify opportunities for improvement, and inform the development of machine learning models and business strategies.

Code:

1) Importing the required libraries for EDA

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

2) Loading the data into the data frame

```
df=pd.read_csv("/content/healthcare_dataset.csv")
df.head()
df
```

3) Checking the types of data

```
df.dtypes
```

```
Age                    int64
Gender                object
Blood Type            object
Medical Condition     object
Date of Admission     object
Doctor                object
Hospital              object
Insurance Provider    object
Billing Amount       float64
Admission Type        object
Discharge Date        object
Medication            object
Test Results          object
dtype: object
```

4) Dropping the irrelevant columns

```
df.drop(columns=['Name','Room Number',"Date of Admission"],axis=1,inplace=True)
df.columns
```

```
Index(['Age', 'Gender', 'Blood Type', 'Medical Condition', 'Doctor',
       'Hospital', 'Insurance Provider', 'Billing Amount', 'Admission Type',
       'Discharge Date', 'Medication', 'Test Results', 'Billing_amount',
       'age_sta'],
      dtype='object')
```

5) Renaming the columns

```
df.rename(columns={"Blood Type":"BType","Medication":"Meds"},inplace=True)
df.columns
```

```
Index(['Age', 'Gender', 'BType', 'Medical Condition', 'Doctor', 'Hospital',
       'Insurance Provider', 'Billing Amount', 'Admission Type',
       'Discharge Date', 'Meds', 'Test Results', 'Billing_amount', 'age_sta'],
      dtype='object')
```

6) Dropping the duplicate rows

```
[5]  df.drop_duplicates(inplace=True)
```

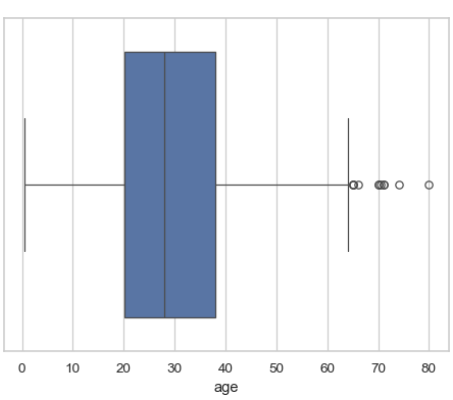
7) Dropping the missing or null values

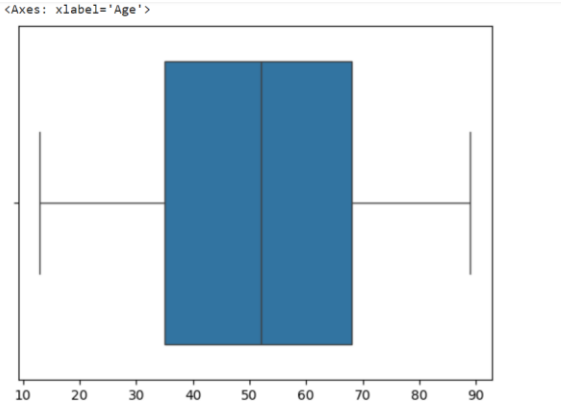
```
print(df.isnull().sum())
```

```
Name                  0
Age                   0
Gender                0
Blood Type            0
Medical Condition     0
Date of Admission     0
Doctor                0
Hospital              0
Insurance Provider    0
Billing Amount        0
Room Number           0
Admission Type        0
Discharge Date        0
Medication            0
Test Results          0
dtype: int64
```

```
df.dropna()
```

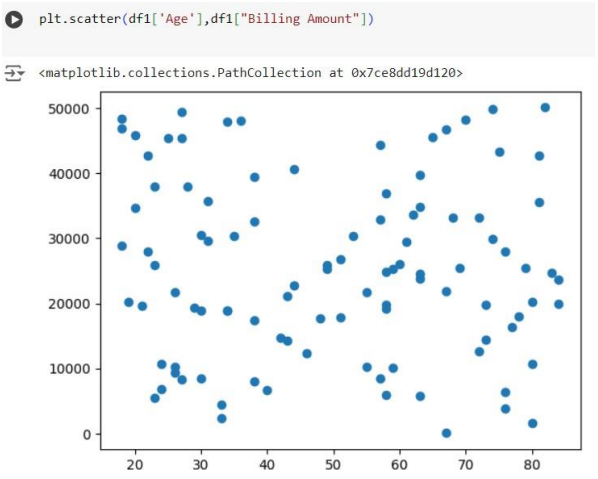
8) Detecting the outliers(Before and after removing the outliers)

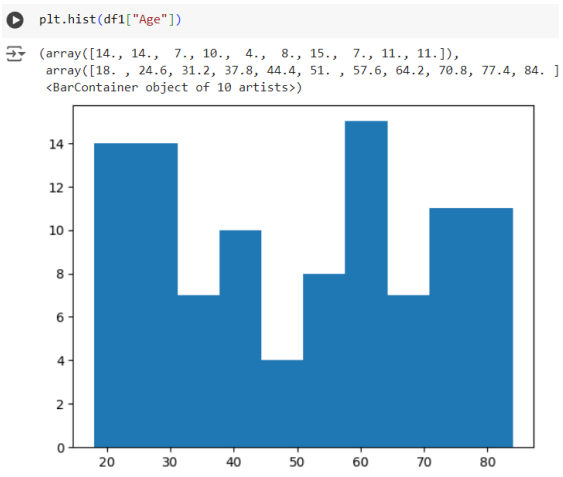




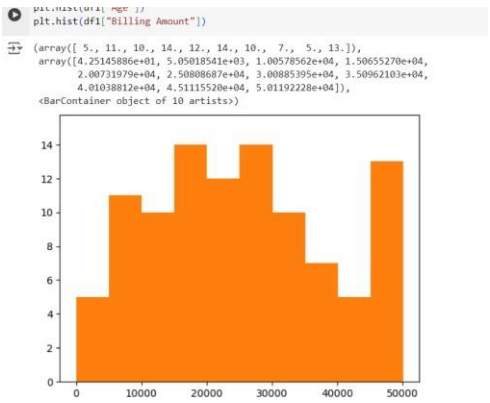
9) Plot different features against one another(scatter), against frequency(histogram)

Scatter plot of Age vs Billing Amount





Histogram of Age



Histogram of Billing Amount

Conclusion: Therefore, various methods and techniques of exploratory data analysis were applied on the dataset.