📊 Exploratory Data Analysis (EDA) Report

This report provides a complete walkthrough of the Exploratory Data Analysis (EDA) process performed on the dataset. It includes basic dataset inspection, cleaning steps, and detailed visual analytics.

Firstly we import some of the popular libraries for operation

Import numpy as np

Import pandas as pd

Import matplotlib.pyplot as plt

Import seaborn as sns

---

1. Dataset Loading

- The dataset was loaded into a Pandas DataFrame.
- Ensures the data is structured for analysis.
- Acts as the foundation for the entire EDA.

---

2. Dataset Overview

head()

- Displays the first 5 rows.
- Helps verify column names and data formatting.
- Useful to understand structure quickly.

tail()

- Shows the last 5 rows.
- Helps check data consistency at the end.
- Useful for spotting trailing nulls or unusual patterns.

info()

- Shows column data types.
- Displays non-null count for each column.
- Useful for identifying missing data and type issues.

describe()

- Provides summary statistics for numerical columns.
- Includes mean, median, standard deviation, min, max.
- Helps identify distribution shape and outliers.
-

shape

- Shows number of rows and columns.
- Helps assess dataset size.
- Useful before and after cleaning.

isnull()

- Identifies missing values across the dataset.
- Useful for planning data cleaning.
- Can be summed to get total number of nulls.

fillna()

- Used to replace missing values.
- Prevents errors during visualization or modeling.
- Can use mean, median, mode, or custom values.

---

3. Univariate Analysis

distplot()

- Shows distribution of a single numerical column.
- Combines histogram with KDE curve.
- Useful for understanding variable spread.

kdeplot()

- Displays smooth density curve.
- Helps understand shape of distribution.
- Can be compared across multiple variables.

histplot()

- Pure histogram showing frequency distribution.
- Helps detect skewness.
- Good for exploring data dispersion.

rugplot()

- Adds small sticks representing data points.
- Shows exact distribution of samples.
- Useful for small datasets.

boxplot()

- Displays quartiles and median.
- Reveals outliers effectively.
- Useful for comparing categories.

---

4. Bivariate & Multivariate Analysis

scatterplot()

- Plots relationship between two numeric variables.
- Helps identify correlation patterns.
- Useful for finding clusters and outliers.

barplot()

- Shows comparison of categories.
- Displays mean values with confidence intervals.
- Useful for categorical vs numerical analysis.

jointplot()

- Combines scatterplot with marginal distributions.
- Shows relationship plus univariate spread.
- Useful for detailed two-variable analysis.

pairplot()

- Creates multi-scatterplots for all numerical variables.
- Helps discover relationships and trends.
- Very effective for high-level multivariate understanding.

lmplot()

- Shows scatterplot with regression line.
- Useful for detecting linear relationships.
- Helps understand predictive patterns.

heatmap()

- Shows correlation values as a colored matrix.
- Helps identify strong/weak relationships.
- Useful for feature selection before modeling.

---

5. Conclusions from EDA

- The dataset's structure and quality were validated using initial inspection commands.
- Missing values were identified and handled properly.
- Visualizations helped uncover trends, correlations, and outliers.
- Heatmap and pairplots assisted in understanding relationships between variables.
