**Vidyavardhini's College of Engineering & Technology**
Department of Computer Engineering
Academic Year: 2025-26

| |
|---|
| Experiment No.4 |
| Perform data exploration |
| Name/Roll-No: Sumit Metkari/32 |
| Branch/Div: Comps/TE-2 |
| Date of Performance: 06-08-25 |
| Date of Submission: 13-08-25 |

**Aim:** To implement data exploration on a given data set

**Objective:-**Develop a program to implement data exploration techniques

**Theory:** Why explore the data? Because before jumping into complex modeling or analysis, we must first understand the nature, structure, and distribution of the dataset. Data exploration helps us answer basic but critical questions such as:

What variables are present and what types are they?
How many records are there?
Are there missing values or extreme values?
What patterns or relationships exist between variables?
Why is data exploration important?
Data exploration serves as the first step in any data analysis or machine learning workflow. Without it we might overlook hidden data issues. We may choose incorrect preprocessing or modeling techniques. Our statistical tests may be invalid because assumptions about data distribution are wrong.

In short, quality insights and model performance depend on how well we know our data. By exploring data, we can detect outliers, understand variable distributions, identify correlations, and choose the right transformations and analysis methods.

The different data exploration steps and functions that can be applied are:

head() – Displays the first few rows to get a quick look at the structure and data types.

tail() – Displays the last few rows, useful for checking data consistency or issues at the end of the file.

shape – Returns the number of rows and columns to understand dataset size.

describe() – Provides summary statistics like mean, standard deviation, min, max, and quartiles.

Scatter plot – Shows the relationship between two variables, helping detect trends, clusters, or anomalies.

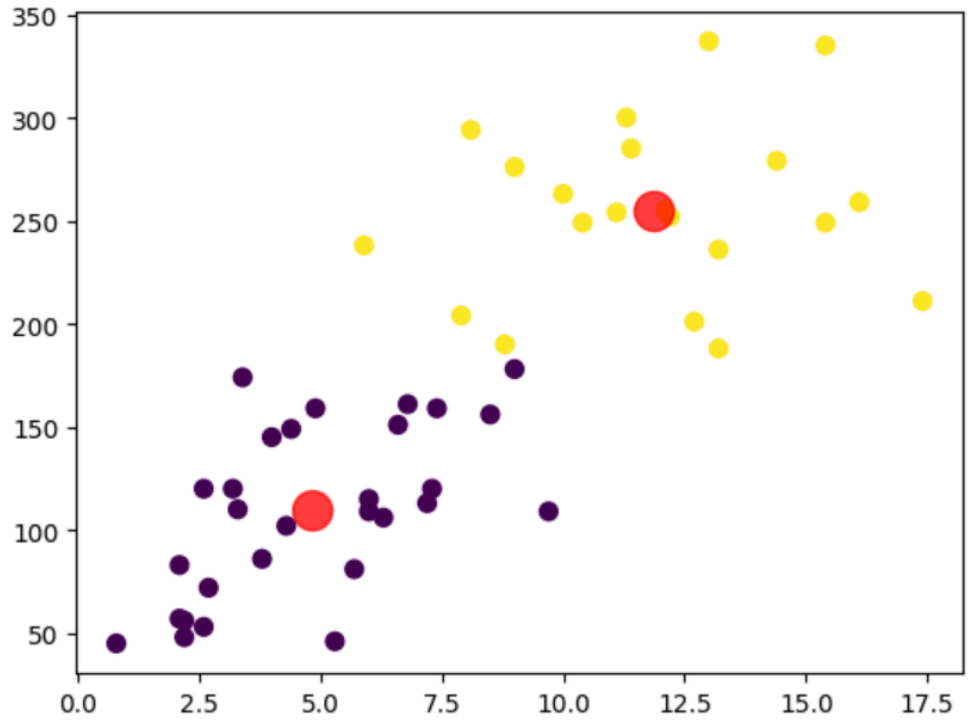
Histogram – Displays the distribution of a single variable to check skewness, modality, and spread.
Quantile-Quantile (Q-Q) plot – Checks if a variable follows a particular theoretical distribution (e.g., normal distribution).

Box plot – Highlights the spread, median, and potential outliers for a variable.

**Code and output:**

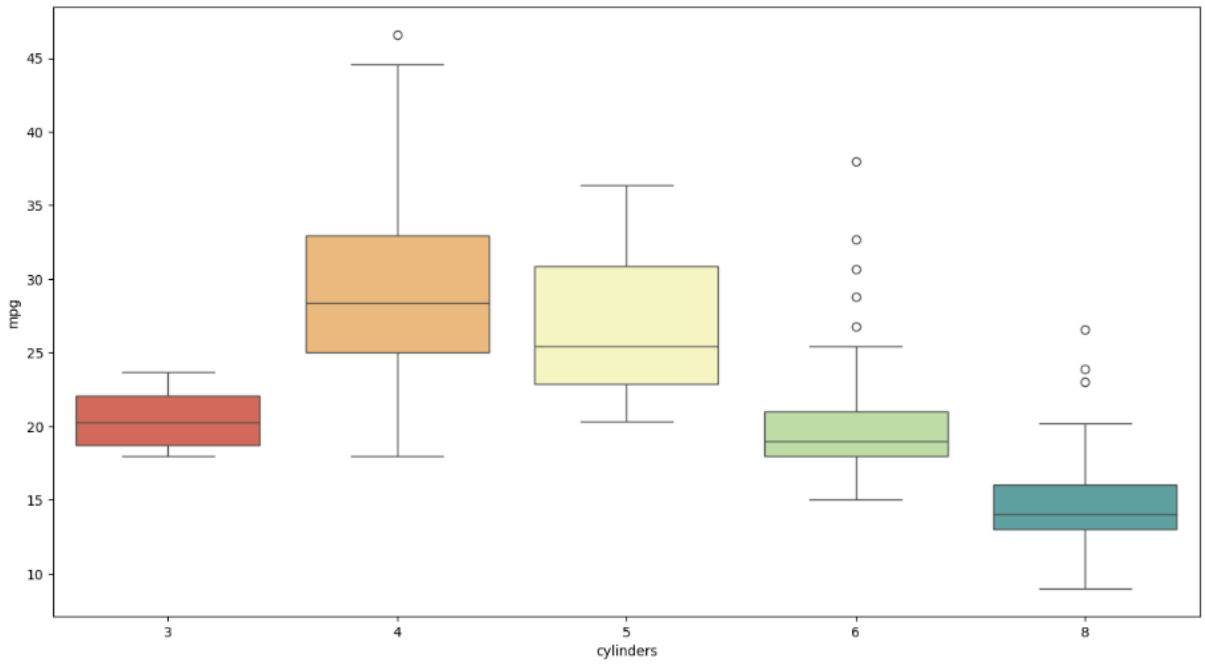
**1)Scatter Plot**

```
plt.scatter(df.iloc[:,0], df.iloc[:,1], c = clusters, s = 50, cmap = "viridis")
plt.scatter(centroids[:,0], centroids[:,1], c = "red", s = 250, alpha=0.75);
```



**2)Boxplot**

```
plt.figure(figsize=(15,8))
sns.boxplot(x="cylinders",y="mpg",data=df,palette="Spectral")
plt.show()
```




**3)Head**

```
df.head()
```

[19]:

| | mpg | cylinders | displacement | horsepower | weight | acceleration | model_year | origin |
|---|---|---|---|---|---|---|---|---|
| **0** | 18.0 | 8 | 307.0 | 130.0 | 3504 | 12.0 | 1970 | usa |
| **1** | 15.0 | 8 | 350.0 | 165.0 | 3693 | 11.5 | 1970 | usa |
| **2** | 18.0 | 8 | 318.0 | 150.0 | 3436 | 11.0 | 1970 | usa |
| **3** | 16.0 | 8 | 304.0 | 150.0 | 3433 | 12.0 | 1970 | usa |
| **4** | 17.0 | 8 | 302.0 | 140.0 | 3449 | 10.5 | 1970 | usa |

**4)Shape**

```
df.shape
```

[13]:

```
(392, 8)
```

**5)Tail**

```
df.tail()
```

[5]:

| | mpg | cylinders | displacement | horsepower | weight | acceleration | model_year | origin |
|---|---|---|---|---|---|---|---|---|
| **393** | 27.0 | 4 | 140.0 | 86.0 | 2790 | 15.6 | 82 | usa |
| **394** | 44.0 | 4 | 97.0 | 52.0 | 2130 | 24.6 | 82 | europe |
| **395** | 32.0 | 4 | 135.0 | 84.0 | 2295 | 11.6 | 82 | usa |
| **396** | 28.0 | 4 | 120.0 | 79.0 | 2625 | 18.6 | 82 | usa |
| **397** | 31.0 | 4 | 119.0 | 82.0 | 2720 | 19.4 | 82 | usa |

**6)Describe**

```
df.describe()
```

[6]:

| | mpg | cylinders | displacement | horsepower | weight | acceleration | model_ |
|---|---|---|---|---|---|---|---|
| **count** | 398.000000 | 398.000000 | 398.000000 | 392.000000 | 398.000000 | 398.000000 | 398.00 |
| **mean** | 23.514573 | 5.454774 | 193.425879 | 104.469388 | 2970.424623 | 15.568090 | 76.01 |
| **std** | 7.815984 | 1.701004 | 104.269838 | 38.491160 | 846.841774 | 2.757689 | 3.69 |
| **min** | 9.000000 | 3.000000 | 68.000000 | 46.000000 | 1613.000000 | 8.000000 | 70.00 |
| **25%** | 17.500000 | 4.000000 | 104.250000 | 75.000000 | 2223.750000 | 13.825000 | 73.00 |
| **50%** | 23.000000 | 4.000000 | 148.500000 | 93.500000 | 2803.500000 | 15.500000 | 76.00 |
| **75%** | 29.000000 | 8.000000 | 262.000000 | 126.000000 | 3608.000000 | 17.175000 | 79.00 |
| **max** | 46.600000 | 8.000000 | 455.000000 | 230.000000 | 5140.000000 | 24.800000 | 82.00 |

**Conclusion**: Data exploration plays a vital role in uncovering underlying patterns, trends, and relationships within a dataset. It informs better decision-making around feature selection, data transformation, and model selection. Without this step, critical issues such as outliers, bias, or class imbalance may remain undetected, potentially resulting in flawed analyses, underperforming models, and inaccurate conclusions.