# TASK 3

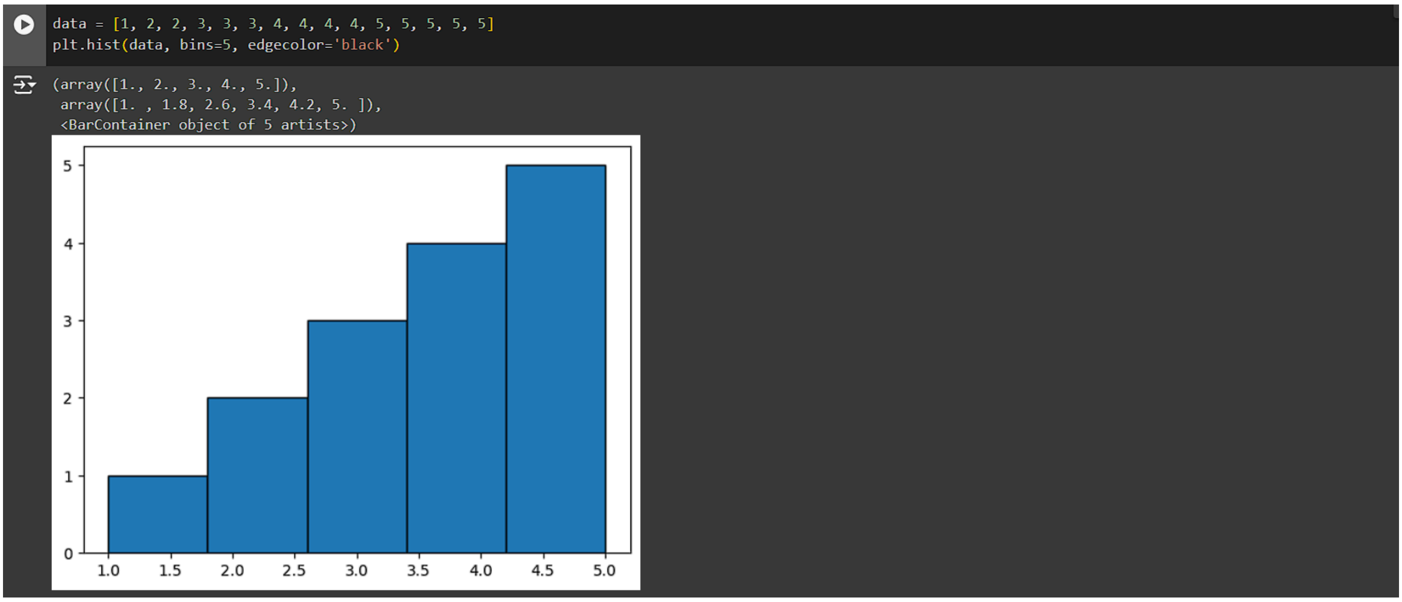
## Exploratory Data Analysis (EDA)

-Krish Batra (2762856)

EDA is the process of analyzing datasets to summarize their main characteristics, often using visual methods. It helps in understanding the data, identifying patterns, and detecting anomalies. Key steps include:

### 1. Data Visualization:

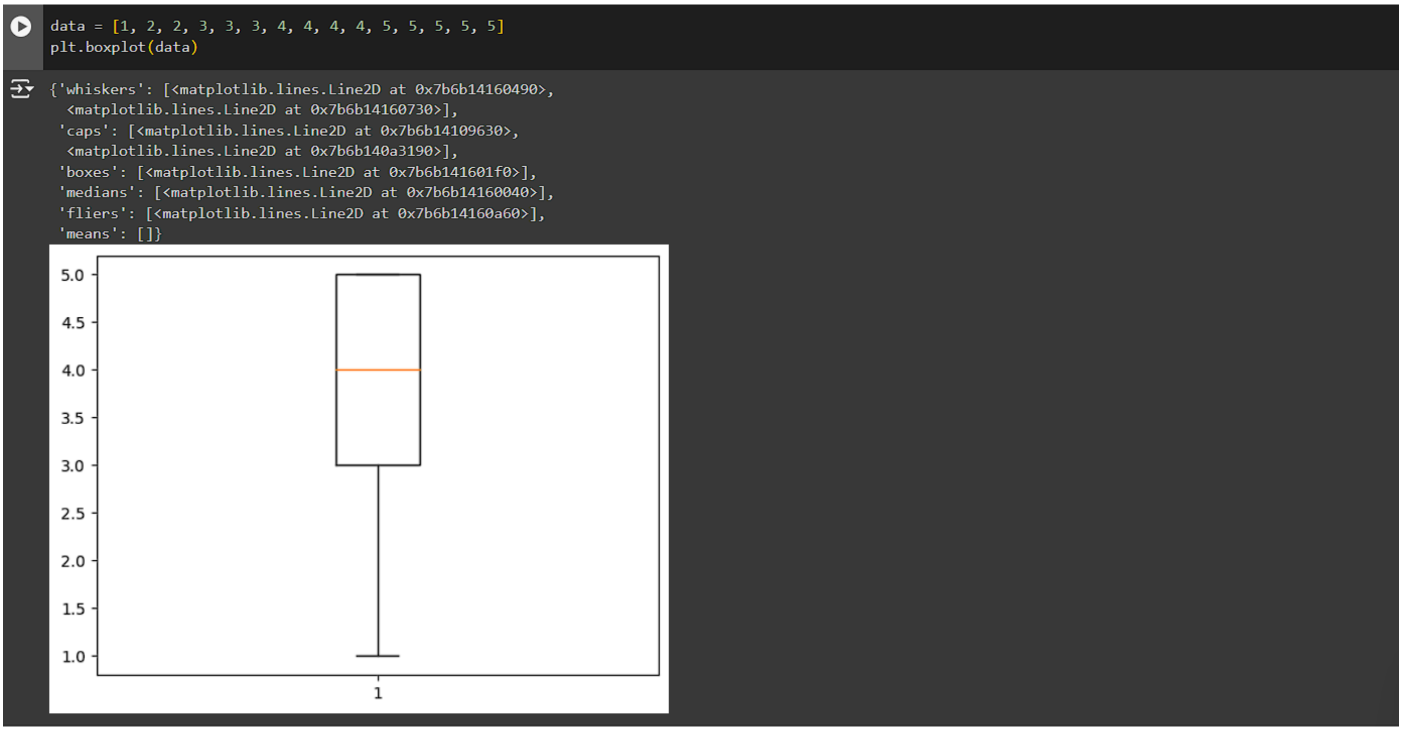
- **Histograms:** For understanding the distribution of a single variable.



```
data = [1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5]
plt.hist(data, bins=5, edgecolor='black')
```

```
(array([1., 2., 3., 4., 5.]),
 array([1. , 1.8, 2.6, 3.4, 4.2, 5. ]),
 <BarContainer object of 5 artists>)
```

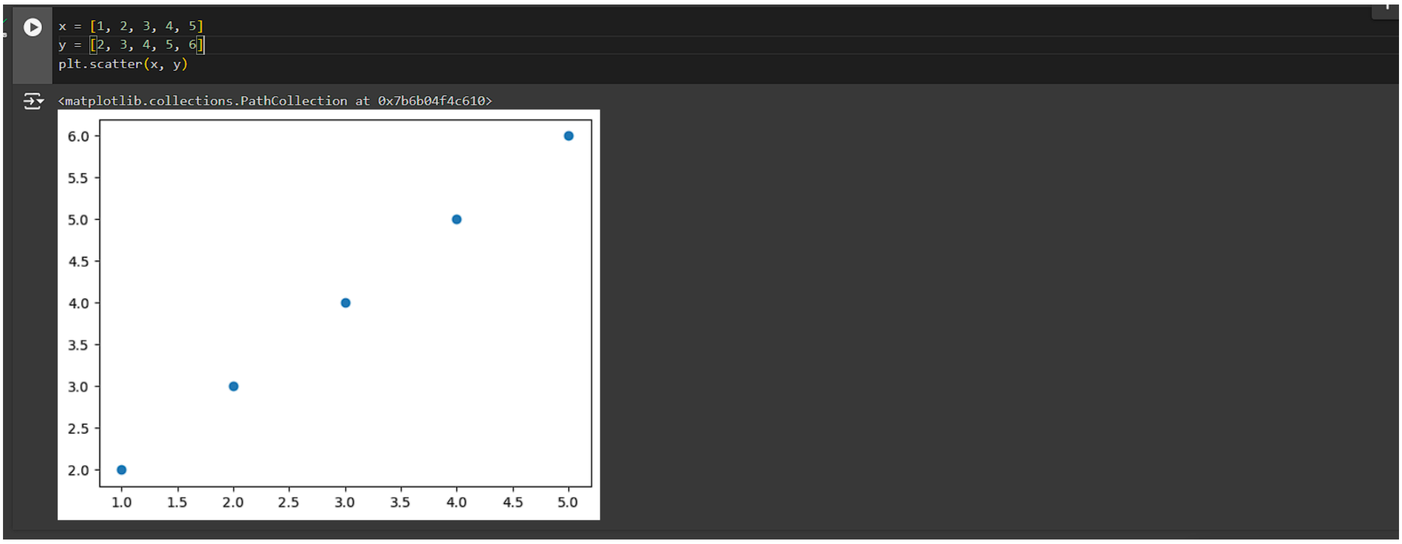
- **Box Plots:** To identify outliers and understand data spread.



```
data = [1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5]
plt.boxplot(data)
```

```
{'whiskers': [<matplotlib.lines.Line2D at 0x7b6b14160490>,
  <matplotlib.lines.Line2D at 0x7b6b14160730>],
 'caps': [<matplotlib.lines.Line2D at 0x7b6b14109630>,
  <matplotlib.lines.Line2D at 0x7b6b140a3190>],
 'boxes': [<matplotlib.lines.Line2D at 0x7b6b141601f0>],
 'medians': [<matplotlib.lines.Line2D at 0x7b6b14160040>],
 'fliers': [<matplotlib.lines.Line2D at 0x7b6b14160a60>],
 'means': []}
```

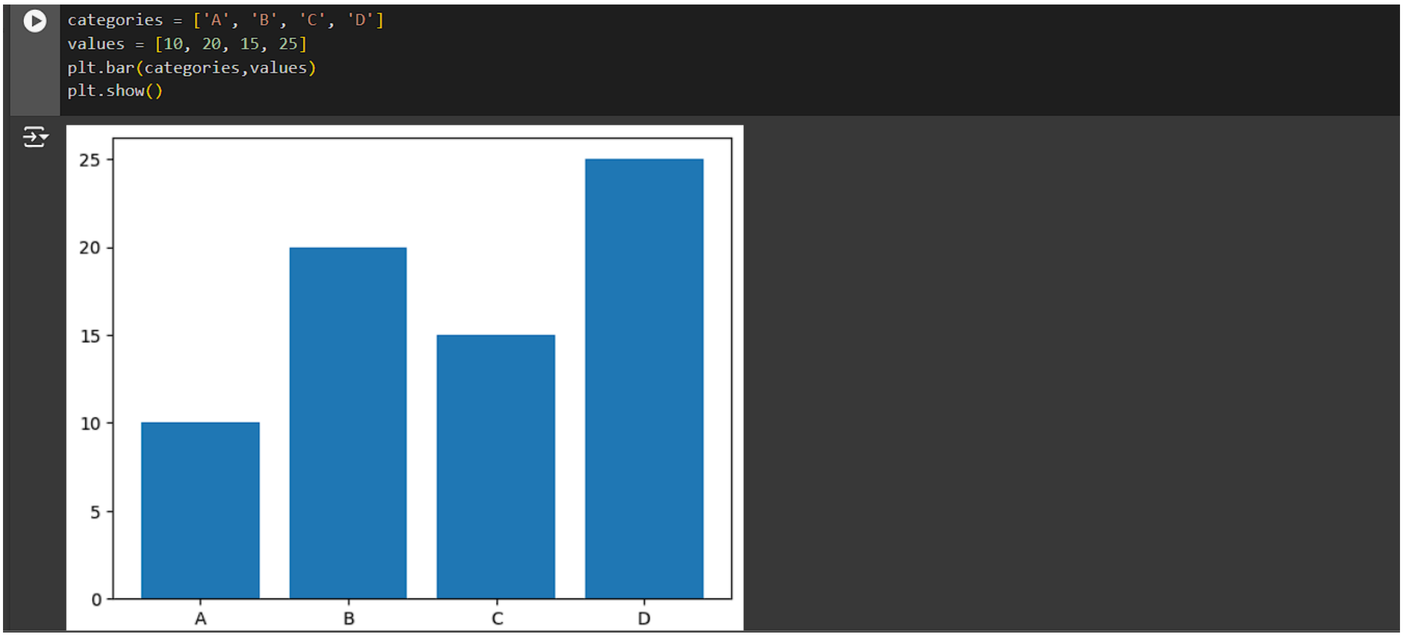
- **Scatter Plots:** To analyze relationships between two numerical variables.



```
x = [1, 2, 3, 4, 5]
y = [2, 3, 4, 5, 6]
plt.scatter(x, y)
```

```
<matplotlib.collections.PathCollection at 0x7b6b04f4c610>
```

- **Bar Charts:** For categorical data frequency.



```
categories = ['A', 'B', 'C', 'D']
values = [10, 20, 15, 25]
plt.bar(categories,values)
plt.show()
```

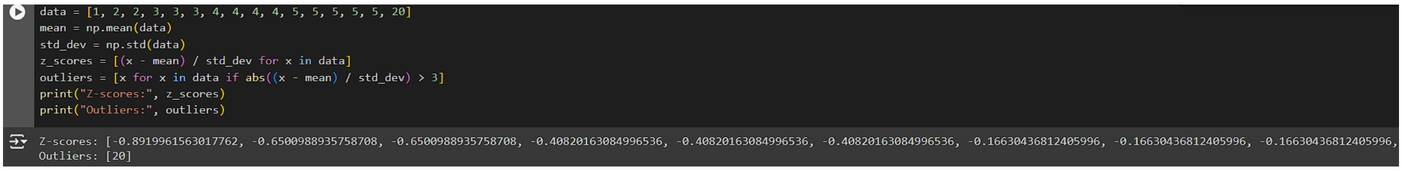
## 2. Summary Statistics:

- Mean, median, mode (measures of central tendency).
- Variance, standard deviation (measures of dispersion).

## 3. Outlier Detection:

- Identifying data points that significantly deviate from the norm using IQR, Z-scores, etc.
- **Z-SCORE**
  1. Z-Score Method:

The Z-score indicates how many standard deviations a data point is from the mean. A Z-score greater than 3 or less than -3 is often considered an outlier.

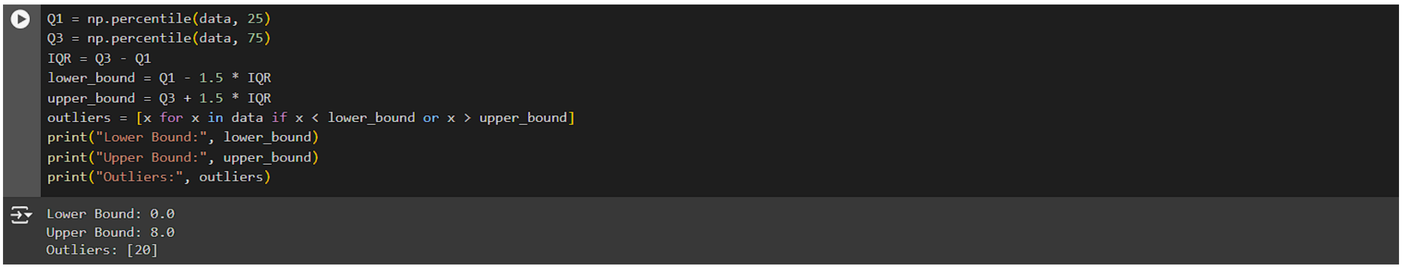


```
data = [1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, 20]
mean = np.mean(data)
std_dev = np.std(data)
z_scores = [(x - mean) / std_dev for x in data]
outliers = [x for x in data if abs((x - mean) / std_dev) > 3]
print("Z-scores:", z_scores)
print("Outliers:", outliers)
```

```
Z-scores: [-0.8919961563017762, -0.6500988935758708, -0.6500988935758708, -0.40820163084996536, -0.40820163084996536, -0.40820163084996536, -0.16630436812405996, -0.16630436812405996, -0.16630436812405996,
Outliers: [20]
```

2. IQR Method:

   The Interquartile Range (IQR) measures the spread of the middle 50% of the data. Outliers are typically defined as points outside the range [Q1−1.5∗IQR,Q3+1.5∗IQR].

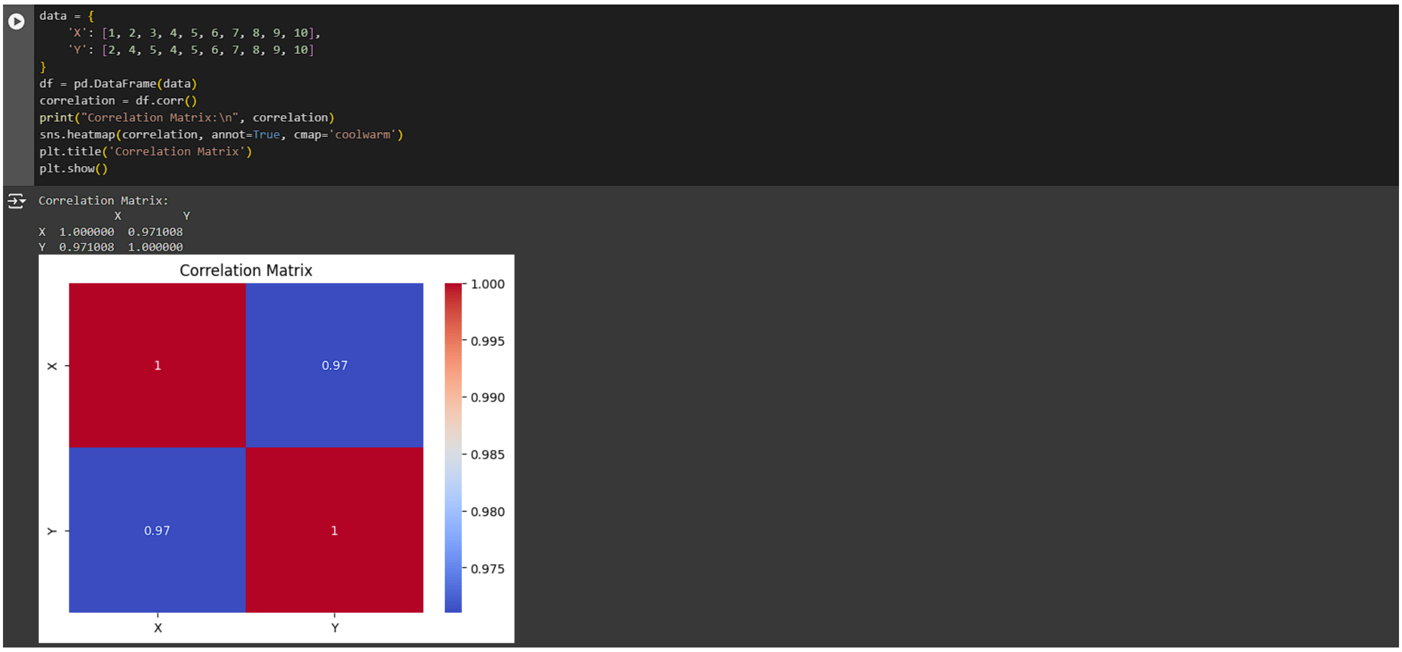


```
Q1 = np.percentile(data, 25)
Q3 = np.percentile(data, 75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
outliers = [x for x in data if x < lower_bound or x > upper_bound]
print("Lower Bound:", lower_bound)
print("Upper Bound:", upper_bound)
print("Outliers:", outliers)
```

```
Lower Bound: 0.0
Upper Bound: 8.0
Outliers: [20]
```

## 4. Correlation Analysis:

- Using correlation matrices and coefficients to understand **relationships** between variables.



```
data = {
    'X': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
    'Y': [2, 4, 5, 4, 5, 6, 7, 8, 9, 10]
}
df = pd.DataFrame(data)
correlation = df.corr()
print("Correlation Matrix:\n", correlation)
sns.heatmap(correlation, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```

```
Correlation Matrix:
           X         Y
X  1.000000  0.971008
Y  0.971008  1.000000
```

## Importance of Statistics in Data Science

Statistics is foundational in data science, providing tools to extract insights from data, make predictions, and inform decisions. Key areas include:

1. **Descriptive Statistics**:
   - Summarizing and describing data features.
   - Using graphical representations and summary statistics.

2. **Inferential Statistics**:
    - Making inferences about populations from samples.
    - Hypothesis testing, confidence intervals, and regression analysis.

## Applications of Statistics in Data Science

Statistics is applied across various domains to solve real-world problems:

1. **Business Analytics and Market Research**:
    - Identifying trends and consumer behavior.
    - Making informed business decisions using regression analysis and hypothesis testing.
2. **Healthcare and Medical Research**:
    - Analyzing clinical trial data and patient outcomes.
    - Conducting epidemiological studies and public health interventions.
3. **Finance and Risk Management**:
    - Evaluating financial risks using time series analysis and Monte Carlo simulations.
    - Optimizing investment strategies and managing portfolios.
4. **Social Sciences and Demographic Studies**:
    - Analyzing population trends and socioeconomic indicators.
    - Assessing the impact of policy decisions.
5. **Environmental Monitoring and Climate Science**:
    - Studying weather patterns and climate change impacts.
    - Assessing environmental policies.
6. **Sports Analytics and Performance Tracking**:
    - Evaluating player performance and team strategies.
    - Enhancing training programs and game tactics.
7. **Transportation and Logistics Optimization**:
    - Analyzing traffic patterns and optimizing supply chains.
    - Improving route planning and inventory management.
8. **Fraud Detection and Cybersecurity**:
    - Detecting fraudulent activities and security threats.
    - Using anomaly detection and clustering analysis.

## Tools for Performing EDA and Statistics

**Python Libraries**:

- Pandas: Data manipulation and analysis.
- Matplotlib and Seaborn: Data visualization.
- Plotly: Interactive visualizations.