Descriptive Statistics in Python, Pandas, and Matplotlib

# Descriptive Statistics in Python

## Overview

Descriptive statistics summarize and describe the main features of a dataset. Python, with its rich ecosystem of libraries, makes it easy to perform descriptive statistical analysis.

## Basic Statistics in Python

### Mean

```python  
data = [1, 2, 3, 4, 5]  
mean\_value = sum(data) / len(data)  
print("Mean:", mean\_value)  
```

### Median

```python  
data = [1, 2, 3, 4, 5]  
data.sort()  
n = len(data)  
median\_value = data[n//2] if n % 2 != 0 else (data[n//2 - 1] + data[n//2]) / 2  
print("Median:", median\_value)  
```

### Mode

```python  
from statistics import mode  
data = [1, 2, 2, 3, 4]  
mode\_value = mode(data)  
print("Mode:", mode\_value)  
```

### Standard Deviation

```python  
from statistics import stdev  
data = [1, 2, 3, 4, 5]  
std\_dev = stdev(data)  
print("Standard Deviation:", std\_dev)  
```

# Pandas

## Overview

Pandas is a powerful and flexible open-source data analysis and manipulation library for Python. It provides data structures like Series and DataFrame for handling data efficiently.

## Basic Usage

### Creating a DataFrame

```python  
import pandas as pd  
  
data = {  
 "Name": ["Alice", "Bob", "Charlie", "David"],  
 "Age": [25, 30, 35, 40],  
 "Score": [85, 90, 95, 100]  
}  
df = pd.DataFrame(data)  
print(df)  
```

### Descriptive Statistics

```python  
print("Mean Age:", df["Age"].mean())  
print("Median Age:", df["Age"].median())  
print("Standard Deviation of Scores:", df["Score"].std())  
```

### Summary Statistics

```python  
print(df.describe())  
```

# Matplotlib

## Overview

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. It is widely used for plotting data.

## Basic Usage

### Line Plot

```python  
import matplotlib.pyplot as plt  
  
x = [1, 2, 3, 4, 5]  
y = [10, 20, 25, 30, 35]  
  
plt.plot(x, y)  
plt.xlabel("X-axis Label")  
plt.ylabel("Y-axis Label")  
plt.title("Line Plot Example")  
plt.show()  
```

### Bar Chart

```python  
categories = ["A", "B", "C", "D"]  
values = [5, 7, 3, 8]  
  
plt.bar(categories, values)  
plt.xlabel("Category")  
plt.ylabel("Value")  
plt.title("Bar Chart Example")  
plt.show()  
```

### Histogram

```python  
data = [1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5]  
  
plt.hist(data, bins=5)  
plt.xlabel("Value")  
plt.ylabel("Frequency")  
plt.title("Histogram Example")  
plt.show()  
```

## Combining Pandas and Matplotlib

### Plotting DataFrame

```python  
df.plot(kind="bar", x="Name", y="Score", title="Scores by Name")  
plt.show()  
```

### Customizing Plots

```python  
ax = df.plot(kind="line", x="Name", y="Age", color="red", title="Ages by Name")  
ax.set\_xlabel("Name")  
ax.set\_ylabel("Age")  
plt.show()  
```