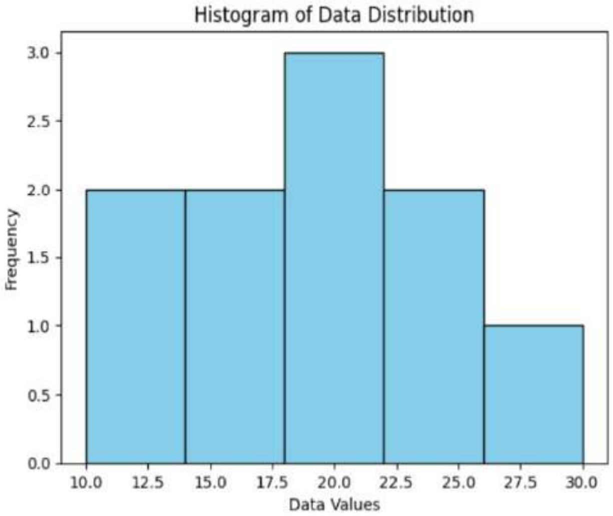
3. Statistical Analysis

```
# Compute Key Statistical Measures

        import numpy as np



        data = np.array([12, 15, 14, 10, 18, 20, 25, 30, 22, 19])

        mean_value = np.mean(data)

        median_value = np.median(data)

        variance_value = np.var(data)

        std_dev_value = np.std(data)


        print("Mean:", mean_value)

        print("Median:", median_value)

        print("Variance:", variance_value)

        print("Standard Deviation:", std_dev_value)

      #Output

        Mean: 18.5

        Median: 17.0

        Variance: 37.25

        Standard Deviation: 6.103
```

```
# Analyzing Data Distribution

        import matplotlib.pyplot as plt


        plt.hist(data, bins=5, color='skyblue', edgecolor='black')

        plt.xlabel('Data Values')

        plt.ylabel('Frequency')

        plt.title('Histogram of Data Distribution')

        plt.show()
```

#Output:



Applying Statistical Functions in Real-World Scenarios

Example: Analyzing Monthly Stock Returns

```
stock_returns = np.array([2.1, 1.5, -0.5, 3.2, 2.8, -1.2, 0.9, 1.7, 2.4, -0.8])


mean_return = np.mean(stock_returns)
std_dev_return = np.std(stock_returns)
risk_factor = std_dev_return / mean_return


print("Average Monthly Return:", mean_return)
print("Standard Deviation of Returns:", std_dev_return)
print("Risk Factor (Volatility):", risk_factor)
```

Output

```
Average Monthly Return: 1.21
Standard Deviation of Returns: 1.42
Risk Factor (Volatility): 1.1735
```