Assignment Code: DS-AG-005

Question 1: What is the difference between descriptive statistics and inferential statistics? Explain with examples.

Answer :-

Descriptive Statistics

Definition:

Descriptive statistics summarize and organize characteristics of a data set. They describe what the data shows, without making predictions or generalizations beyond the data.

Key Features:

- Focuses on raw data you already have
- Uses measures like mean, median, mode, standard deviation, range
- Often visualized using charts, tables, histograms, pie charts

Example:

Suppose you analyze employee performance scores for 100 staff members in an HR dataset.

- Mean score = 78
- Standard deviation = 5.2
- Histogram shows most scores are between 75–85

This is descriptive—it tells you what’s happening in your sample.

Inferential Statistics

Definition:

Inferential statistics use sample data to make predictions or generalizations about a larger population. It involves probability theory and hypothesis testing.

Key Features:

- Draws conclusions beyond the data
- Uses techniques like confidence intervals, regression, hypothesis testing, ANOVA
- Requires assumptions about the population

Example:

You want to know if a new training program improves performance across the company.

- You test it on a sample of 50 employees
- Use a t-test to compare their scores with those who didn't receive training
- If p-value < 0.05, you infer the training likely improves performance for the entire workforce

Question 2: What is sampling in statistics? Explain the differences between random and stratified sampling.

Answer:-

Sampling is the process of selecting a subset (sample) from a larger group (population) to analyze and draw conclusions. Since studying an entire population is often impractical, sampling helps you make informed decisions efficiently.

Example:

If you're analyzing employee satisfaction across a company of 5,000 people, you might survey 300 employees instead of all 5,000. That 300 is your sample.

Difference between random and stratified sampling:

Simple random sampling selects a sample from the entire population where every individual has an equal chance of being chosen, whereas stratified sampling

divides the population into homogeneous subgroups (strata) and then randomly samples from each stratum to ensure all subgroups are represented in proportion to their size in the overall population.

Question 3: Define mean, median, and mode. Explain why these measures of central tendency are important.

Answer:-

Definitions of Mean, Median, and Mode

Mean: The arithmetic mean is found by summing all the numbers in a dataset and dividing by the total count of numbers.

Median: The median is the middle value in a dataset that has been ordered from smallest to largest. If there is an even number of data points, the median is the average of the two middle values.

Mode: The mode is the value that appears most often in a dataset. A dataset can have more than one mode or no mode at all.

Why these measures of central tendency are important?

These measures are important because they provide a single, representative value for a dataset, making complex data easier to summarize, understand, and compare, thereby aiding in decision-making and pattern recognition.

Question 4: Explain skewness and kurtosis. What does a positive skew imply about the data?

Answer:-

Skewness measures a distribution's asymmetry (how uneven it is), while kurtosis measures its tailedness (how heavy or light the tails are compared to a normal distribution) and peak sharpness.

A positive skew implies the data has a longer tail on the right, with the mean being pulled higher than the median due to a few very large values.

Question 5: Implement a Python program to compute the mean, median, and mode of a given list of numbers.

numbers = [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24, 24, 26, 28]

(Include your Python code and output in the code box below.)

Answer:-

In [3]:
```
import statistics

numbers = [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24, 24, 26, 28]

mean = statistics.mean(numbers)
median = statistics.median(numbers)
mode= statistics.mode(numbers)

print("Mean:", mean)
print("Median:", median)
print("Mode:", mode)
```

```
Mean: 19.6
Median: 19
Mode: 12
```

Question 6: Compute the covariance and correlation coefficient between the following two datasets provided as lists in Python:

list_x = [10, 20, 30, 40, 50]

list_y = [15, 25, 35, 45, 60]

(Include your Python code and output in the code box below.)

Answer:-

In [9]:
```
import numpy as np
import statistics

list_x = [10, 20, 30, 40, 50]
list_y = [15, 25, 35, 45, 60]

#coverting list_x, list_y into arrays.
x = np.array(list_x)
y = np.array(list_y)

covariance = statistics.covariance(x, y)
correlation_coefficient = np.corrcoef(x, y)[0, 1]

print("Covariance:", covariance)
print("Correlation Coefficient:", correlation_coefficient)
```

```
Covariance: 275.0
Correlation Coefficient: 0.995893206467704
```

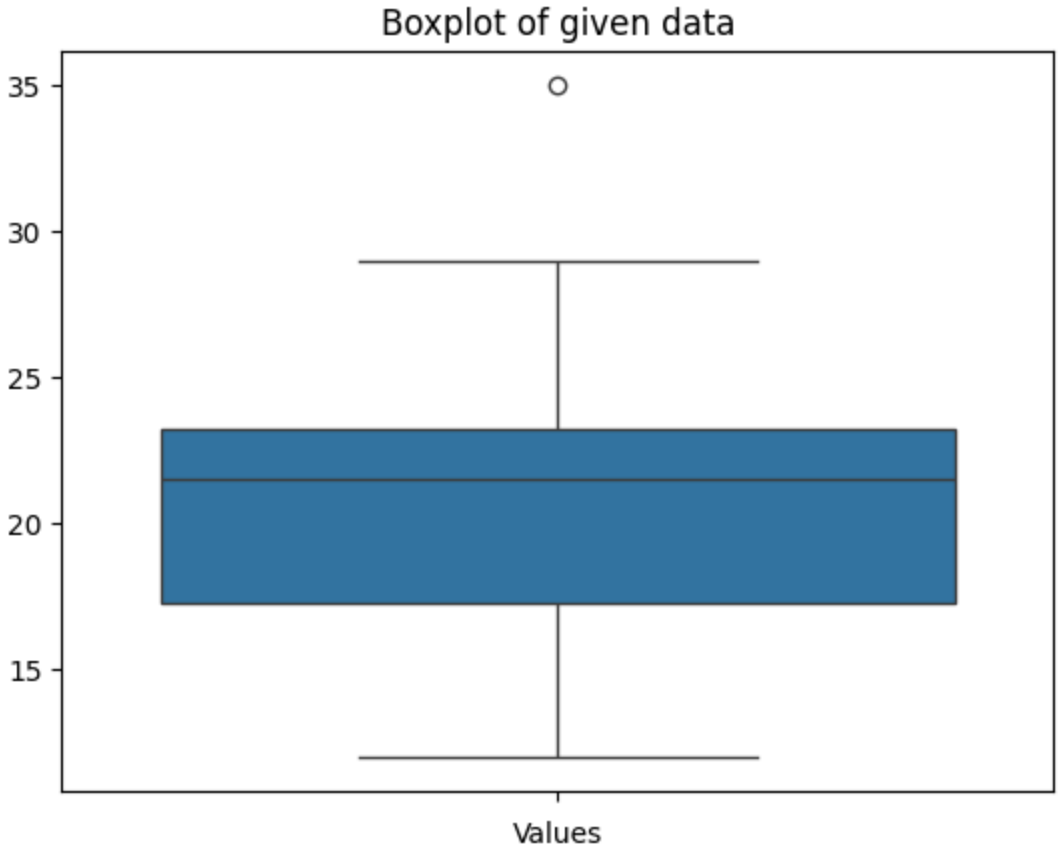
Question 7: Write a Python script to draw a boxplot for the following numeric list and identify its outliers. Explain the result:

data = [12, 14, 14, 15, 18, 19, 19, 21, 22, 22, 23, 23, 24, 26, 29, 35]

(Include your Python code and output in the code box below.)

Answer:-

In [10]:
```
import matplotlib.pyplot as plt
import seaborn as sns

data = [12, 14, 14, 15, 18, 19, 19, 21, 22, 22, 23, 23, 24, 26, 29, 35]

sns.boxplot(data=data)
plt.title("Boxplot of given data")
plt.xlabel("Values")
plt.show()
```



The boxplot will visually show:

- Box: Interquartile range (IQR) from Q1 to Q3
- Line inside box: Median
- Whiskers: Extend to values within 1.5×IQR
- Dots beyond whiskers: Outliers

Interpretation:

- Median: Around 22–23
- IQR: Q1 ≈ 18, Q3 ≈ 24 → IQR = 6
- Outlier threshold:
- Lower bound = Q1 − 1.5×IQR = 18 − 9 = 9
- Upper bound = Q3 + 1.5×IQR = 24 + 9 = 33
- Outlier: 35 is above 33 → outlier

Result:

The boxplot reveals that 35 is an outlier, suggesting one unusually high value compared to the rest of the dataset.

Question 8: You are working as a data analyst in an e-commerce company. The marketing team wants to know if there is a relationship between advertising spend and daily sales.

● Explain how you would use covariance and correlation to explore this relationship.

● Write Python code to compute the correlation between the two lists:

advertising_spend = [200, 250, 300, 400, 500]

daily_sales = [2200, 2450, 2750, 3200, 4000]

(Include your Python code and output in the code box below.)

Answer:-

How to Use Covariance and Correlation

To explore the relationship between advertising spend and daily sales, you can use:

Covariance

- Measures the direction of the relationship.
- Positive covariance → both variables increase together.
- Negative covariance → one increases while the other decreases.
- Limitation: Doesn’t tell you the strength or scale of the relationship.

Correlation Coefficient

- Measures both direction and strength of the relationship.
- Ranges from −1 to +1:
- +1 → perfect positive correlation
- 0 → no correlation
- −1 → perfect negative correlation
- Scale-independent, so it's more interpretable.

In [18]:
```
import numpy as np

# Data
```

```
advertising_spend = [200, 250, 300, 400, 500]
daily_sales = [2200, 2450, 2750, 3200, 4000]

# Covariance
cov_matrix = np.cov(advertising_spend, daily_sales, ddof=0)
covariance = cov_matrix[0, 1]

# Correlation
correlation_matrix = np.corrcoef(advertising_spend, daily_sales)
correlation = correlation_matrix[0, 1]

print("Covariance:", covariance)
print("Correlation:", correlation)
```

```
Covariance: 67900.0
Correlation: 0.9935824101653329
```

Interpretation:

- The large positive covariance means advertising spend and sales increase together.
- The correlation (~0.99) shows a very strong positive linear relationship → more advertising spend strongly drives higher sales.

Question 9:Your team has collected customer satisfaction survey data on a scale of 1-10 and wants to understand its distribution before launching a new product.

● Explain which summary statistics and visualizations (e.g. mean, standard deviation, histogram) you'd use.

● Write Python code to create a histogram using Matplotlib for the survey data:

survey_scores = [7, 8, 5, 9, 6, 7, 8, 9, 10, 4, 7, 6, 9, 8, 7]

(Include your Python code and output in the code box below.)

Answer:-

1. Explanation

To understand the distribution of customer satisfaction scores (1–10 scale) before launching a new product, we should use:

Summary Statistics

i.) Mean (Average): Gives the central tendency of satisfaction.

ii.) Median: Useful if the data is skewed.

iii.) Mode: Shows the most common score.

iv.) Standard Deviation (SD): Measures variability (how spread out the scores are).

v.) Minimum & Maximum: To see the range of responses.

Visualizations

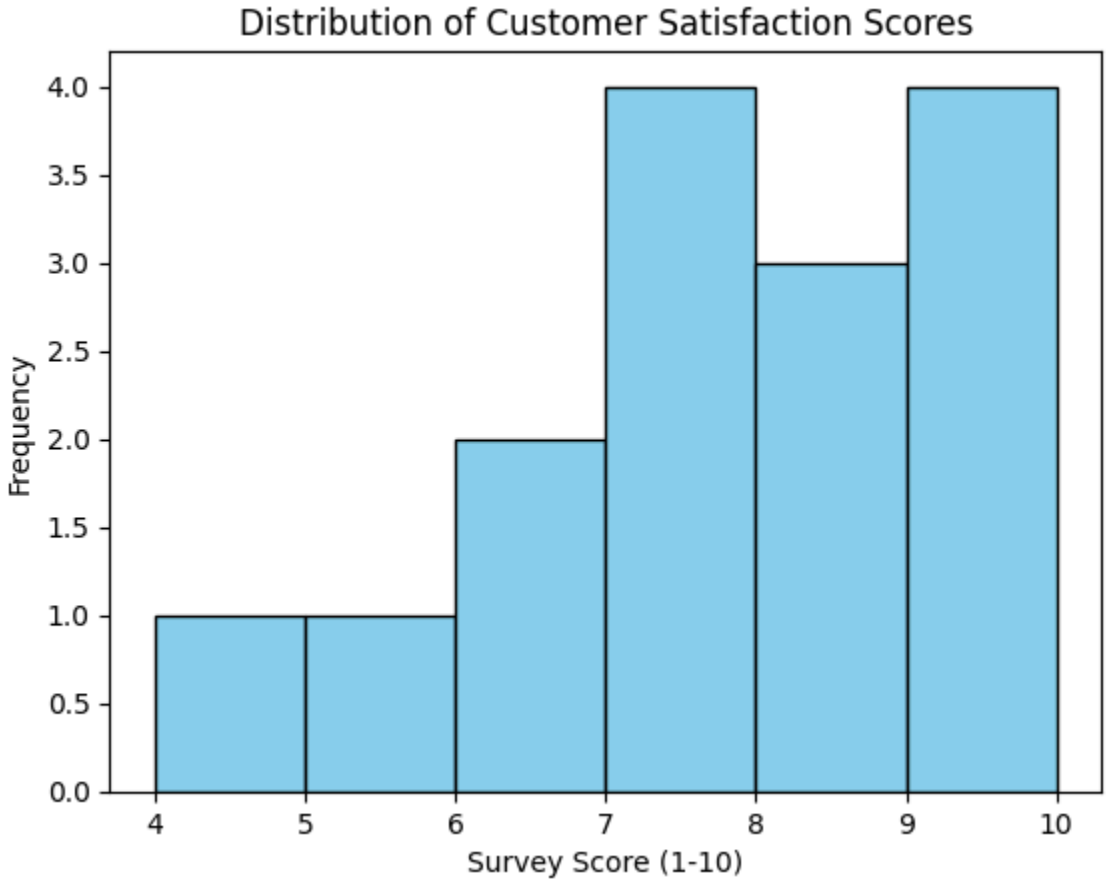
i.) Histogram: Shows the frequency distribution of scores.

ii.) Boxplot (optional): Highlights spread, median, and potential outliers.

iii.) Bar chart of counts: If you want to see exact frequencies for each score.

In [19]:
```
import matplotlib.pyplot as plt
import numpy as np

# Survey data
survey_scores = [7, 8, 5, 9, 6, 7, 8, 9, 10, 4, 7, 6, 9, 8, 7]

# Summary statistics
mean_score = np.mean(survey_scores)
median_score = np.median(survey_scores)
std_dev = np.std(survey_scores)

print("Mean:", mean_score)
print("Median:", median_score)
print("Standard Deviation:", std_dev)

# Histogram
plt.hist(survey_scores, bins=6, color='skyblue', edgecolor='black')
plt.title("Distribution of Customer Satisfaction Scores")
plt.xlabel("Survey Score (1-10)")
plt.ylabel("Frequency")
plt.show()
```

```
Mean: 7.333333333333333
Median: 7.0
Standard Deviation: 1.577621275493231
```



Histogram:

- Most responses cluster around 7–9.
- Few responses at the low end (like 4 or 5).
- Indicates customers are generally satisfied (leaning toward higher scores).