Assignment Code: DS-AG-005

Statistics Basics| **Assignment**

Instructions: Carefully read each question. Use Google Docs, Microsoft Word, or a similar tool to create a document where you type out each question along with its answer. Save the document as a PDF, and then upload it to the LMS. Please do not zip or archive the files before uploading them. Each question carries 20 marks.

Total Marks: 200

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

Answer:

Descriptive statistics:- It describes and summarizes data.

Inferential statistics:- It helps to make predictions, decisions, or conclusions about a large group (population) using a small group (sample).

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

Answer:

Sampling in statistics is the process of selecting a small group (sample) from a large group (population) to study and draw conclusions.
In **Random Sampling**, every individual has an equal chance of being selected, which reduces bias but may not cover all subgroups. In **Stratified Sampling,** the population is divided into groups (strata) based on characteristics, and samples are taken from each group, ensuring better representation of the population.

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

Answer:

Mean is the average of all values, found by dividing the sum of data by the number of observations.
Median is the middle value when the data is arranged in order.
Mode is the value that occurs most frequently in the dataset.
These measures of central tendency are important because they give a single value that represents the whole dataset. They help in understanding the general trend, making comparisons, and simplifying large amounts of data for analysis and decision-making

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

Answer:

Skewness measures the asymmetry of a data distribution. If data is symmetric, skewness is zero. A **positive skew** means the tail is longer on the right side, showing that most values are small but a few very large values exist. A **negative skew** means the tail is on the left, with most values being high but some very low values.

Kurtosis measures the peakedness or flatness of a distribution. High kurtosis shows a sharp peak and more outliers, while low kurtosis shows a flatter curve with fewer outliers.

PRACTICAL

QUESTIONS

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:

Paste your code and output inside the box below:

```
import statistics as stats

# Given list of numbers
numbers = [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24,
24, 26, 28]

# Calculate mean, median, and mode
mean_value = stats.mean(numbers)
median_value = stats.median(numbers)
mode_value = stats.mode(numbers)

# Print results
print("Mean:", mean_value)
print("Median:", median_value)
print("Mode:", mode_value)
```

OUTPUT

```
Mean: 19.8
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:

Paste your code and output inside the box below:

```
import numpy as np

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

# Convert to numpy arrays
x = np.array(list_x)
y = np.array(list_y)

# Covariance matrix
cov_matrix = np.cov(x, y, bias=False)  # bias=False → sample covariance
cov_xy = cov_matrix[0, 1]

# Correlation coefficient
corr_matrix = np.corrcoef(x, y)
corr_xy = corr_matrix[0, 1]

print("Covariance between x and y:", cov_xy)
print("Correlation coefficient between x and y:", corr_xy)


OUTPUT
Covariance between x and y: 225.0
Correlation coefficient between x and y: 0.9938586931957764
```

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:

```
import matplotlib.pyplot as plt
import numpy as np

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

# Draw boxplot
plt.boxplot(data, vert=False, patch_artist=True)
plt.title("Boxplot of Data")
plt.xlabel("Values")
plt.show()

# Calculate Q1, Q3 and IQR
Q1 = np.percentile(data, 25)
Q3 = np.percentile(data, 75)
IQR = Q3 - Q1

# Outlier thresholds
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

# Identify outliers
outliers = [x for x in data if x < lower_bound or x > upper_bound]

print("Q1:", Q1)
print("Q3:", Q3)
print("IQR:", IQR)
print("Lower Bound:", lower_bound)
print("Upper Bound:", upper_bound)
print("Outliers:", outliers)

OUTPUT:-
Q1: 18.25
Q3: 23.75
IQR: 5.5
Lower Bound: 10.0
Upper Bound: 32.0
Outliers: [35]
```

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:

Covariance shows the direction of the relationship between advertising spend and sales (positive, negative, or none).
Correlation shows both the direction and strength of this relationship, with values between –1 and +1.
Thus, correlation is more useful to see how strongly advertising spend affects daily sales.

```
import numpy as np

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

# Convert to numpy arrays
x = np.array(advertising_spend)
y = np.array(daily_sales)

# Covariance
cov_matrix = np.cov(x, y, bias=False)
cov_xy = cov_matrix[0, 1]

# Correlation coefficient
corr_matrix = np.corrcoef(x, y)
corr_xy = corr_matrix[0, 1]

print("Covariance between advertising spend and daily sales:", cov_xy)
print("Correlation coefficient:", corr_xy)
```

OUTPUT:-

```
Covariance between advertising spend and daily sales: 87500.0
Correlation coefficient: 0.9912407071619305
```

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:-

We can use the mean and median to know the average satisfaction, the standard deviation to see how spread out the scores are, and the minimum–maximum for the range. A histogram helps visualize how often each score appears, showing the overall distribution of customer satisfaction.

```
import matplotlib.pyplot as plt
import numpy as np

# Given 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, ddof=1)  # sample standard deviation

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("Histogram of Customer Satisfaction Scores")
plt.xlabel("Satisfaction Score")
plt.ylabel("Frequency")
plt.show()
```

OUTPUT:-

```
Mean: 7.4
Median: 7.0
Standard Deviation: 1.55
```