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 vs. Inferential Statistics

The core difference between descriptive and inferential statistics lies in their **goal** and the **data** they cover.[1]

Feature	Descriptive Statistics	Inferential Statistics
Purpose	To **summarize** and **describe** the features of a specific dataset (sample or population).	To **make inferences** or **generalizations** about a larger **population** based on a **sample** of data.
Scope	Only about the data you have collected.	Goes beyond the collected data to draw conclusions.
Tools	Measures of **Central Tendency** (Mean, Median, Mode) and **Dispersion** (Range, Standard Deviation), charts, graphs.	**Hypothesis Testing**, **Confidence Intervals**, Regression Analysis.
Certainty	High (stating facts about the data).	Lower (results are in the form of probabilities or estimates, with a margin of error).

Example: A teacher calculates the **average score** (mean) and the **highest score** on a recent math test for her class of 30 students.

Example: A polling company surveys a **random sample** of 1,000 registered voters and uses this data to **predict** the final election outcome for the entire nation's population.

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 **subset** (the **sample**) of individuals or items from a much larger group (the **population**) to gather data and draw conclusions about the entire population.
Since it's often impractical or impossible to study every member of a population, sampling provides a method to obtain **representative** data efficiently.

Random vs. Stratified Sampling
These are two distinct types of **probability sampling**, meaning every member of the population has a known, non-zero chance of being selected.
Random Sampling (Simple Random Sampling)

- **Definition:** Every individual or item in the population has an **equal chance** of being selected for the sample.
- **Method:** Selection is purely random. This is often done using random number generators or drawing names from a hat.
- **Key Feature:** It is the simplest method and helps ensure the sample is **unbiased** and representative of the whole population.
- **Example:** Drawing 100 names entirely at random from a list of all 5,000 employees in a company.

Stratified Sampling

- **Definition:** The population is first divided into separate, non-overlapping groups called **strata** (based on a shared characteristic like age, gender, or income). Then, a **simple random sample** is taken from *each* stratum.
- **Method:**
    1. Divide the population into strata.
    2. Select a specific number of subjects from each stratum (often proportional to the stratum's size in the population).
- **Key Feature:** Guarantees that the sample includes sufficient representation from key **subgroups** in the population, which is crucial when those subgroups might have different opinions or characteristics.
- **Example:** A company's 5,000 employees are divided into three strata: "Management" (500 people), "Technical Staff" (3,500 people), and "Support Staff" (1,000 people). You then randomly select 10 people from Management, 70 from Technical, and 20 from Support to ensure the sample reflects the true workforce distribution.

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

Answer:

Mean (Average)	The sum of all values divided by the total count of values.	Uses every data point; is heavily **affected by outliers** (extreme values).
Median	The **middle value** in a dataset that has been arranged in order from smallest to largest.	Not affected by outliers; divides the data into two equal halves (50% above, 50% below).
Mode	The value that **appears most frequently** in the dataset.	The only measure applicable to **nominal** (categorical) data (e.g., favorite color).

Importance of Central Tendency

Measures of central tendency are important because they:

1. **Summarize Data:** They condense a large set of data into a **single, simple figure**, making the data much easier to understand and remember.[2]
2. **Identify the Typical Value:** They give a quick indication of the **most representative score** or central point of the distribution (e.g., the average salary, the most common shoe size).[3]
3. **Facilitate Comparison:** They allow for easy comparison between different groups or datasets (e.g., comparing the mean test scores of two different classes).
4. **Aid Decision-Making:** They provide a basis for informed decisions and future analysis, such as identifying market trends or evaluating performance benchmarks.[4]

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

Answer:

kewness and **Kurtosis** are measures that describe the **shape** of a data distribution, helping to determine how it deviates from a perfectly symmetrical, bell-shaped **Normal Distribution**.

Skewness (Asymmetry)

Skewness is a measure of the **asymmetry** of a distribution.

- A **symmetric** distribution (like a Normal Distribution) has zero skewness; its two sides are mirror images.
- **Skewness measures the extent and direction of the "tail"** or stretch of the distribution.

Kurtosis (Tailedness)

Kurtosis measures the **tailedness** (or peakedness) of a distribution relative to a Normal Distribution.

- It tells you how much of the data is concentrated in the tails and, consequently, how sharp the peak is.
- High kurtosis indicates **heavy tails** and a **sharp peak**, implying a higher probability of **outliers** (extreme values).
- Low kurtosis indicates **light tails** and a **flatter peak**.

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
from collections import Counter

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

# --- 1. Calculate the Mean ---
# The mean is the average of all numbers.
mean_value = statistics.mean(numbers)

# --- 2. Calculate the Median ---
# The median is the middle value when the data is ordered.
median_value = statistics.median(numbers)

# --- 3. Calculate the Mode ---
# The mode is the value that appears most frequently.
# We can use statistics.mode, but for lists with multiple modes
# or for a more robust method, collections.Counter is often used.
# The statistics.mode() function raises a StatisticsError if there are multiple modes,
# so we'll use statistics.multimode() which handles multiple modes (and returns a list).
```

```
try:
    mode_values = statistics.multimode(numbers)
except Exception as e:
    # Fallback/error handling just in case, though multimode is robust
    mode_values = ["Error calculating mode: " + str(e)]

# --- Output the Results ---
print(f"Given Numbers: {numbers}")
print("-" * 30)
print(f"Total Count (n): {len(numbers)}")
print("-" * 30)
print(f"Mean (Average): {mean_value}")
print(f"Median (Middle Value): {median_value}")
print(f"Mode (Most Frequent Value/s): {mode_values}")

# Optional: Show Frequency Distribution
print("-" * 30)
print("Frequency Distribution:")
frequency = Counter(numbers)
for number, count in sorted(frequency.items()):
    print(f"Number {number}: {count} times")
```

Output:

```
Given Numbers: [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24, 24, 26, 28]
------------------------------
Total Count (n): 15
------------------------------
Mean (Average): 19.6
Median (Middle Value): 19
Mode (Most Frequent Value/s): [12, 19, 24]
------------------------------
Frequency Distribution:
Number 12: 3 times
Number 15: 1 times
Number 18: 1 times
Number 19: 3 times
Number 20: 1 times
Number 22: 1 times
Number 24: 3 times
Number 26: 1 times
```

```
Number 28: 1 times
```

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

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

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

# --- 1. Compute Covariance ---
# np.cov returns the covariance matrix. The covariance between x and y is the off-diagonal element (1, 0) or (0, 1).
cov_matrix = np.cov(x, y)
covariance = cov_matrix[0, 1]

# --- 2. Compute Correlation Coefficient ---
# np.corrcoef returns the correlation matrix. The correlation coefficient between x and y is the off-diagonal element (1, 0) or (0, 1).
corr_matrix = np.corrcoef(x, y)
```

```
correlation_coefficient = corr_matrix[0, 1]

print(f"Dataset X: {list_x}")
print(f"Dataset Y: {list_y}")
print("-" * 50)
print(f"Covariance Matrix:\n{cov_matrix}")
print(f"\nCovariance (Cov(X, Y)): {covariance:.2f}")
print("-" * 50)
print(f"Correlation Coefficient Matrix:\n{corr_matrix}")
print(f"\nCorrelation Coefficient (Corr(X, Y)): {correlation_coefficient:.4f}")
```

Covariance	The positive value indicates that as increases, also tends to increase (a positive linear relationship).
Correlation Coefficient	This value is very close to , indicating a **very strong positive linear relationship** between the two datasets.

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 numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# The given list of numbers
data = [12, 14, 14, 15, 18, 19, 19, 21, 22, 22, 23, 23, 24, 26, 29, 35]

# --- 1. Outlier Identification (1.5 * IQR Rule) ---
Q1 = np.percentile(data, 25)
Q3 = np.percentile(data, 75)
IQR = Q3 - Q1

# Calculate the outlier boundaries (fences)
lower_bound = Q1 - (1.5 * IQR)
upper_bound = Q3 + (1.5 * IQR)

# Identify the outliers
```

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

# --- 2. Boxplot Generation ---
plt.figure(figsize=(8, 6))
sns.boxplot(y=data, color='skyblue', width=0.3)
plt.title('Boxplot for Data Distribution')
plt.ylabel('Data Values')
plt.savefig('boxplot_outliers.png')

# --- 3. Output for explanation ---
print(f"Q1 (25th Percentile): {Q1}")
print(f"Q3 (75th Percentile): {Q3}")
print(f"IQR (Q3 - Q1): {IQR}")
print(f"Lower Bound (Q1 - 1.5*IQR): {lower_bound}")
print(f"Upper Bound (Q3 + 1.5*IQR): {upper_bound}")
print(f"Identified Outliers: {outliers}")
```

Boxplot and Outlier Explanation

The boxplot visually summarizes the distribution of the data. Outliers are typically shown as individual points (circles or diamonds) outside the whiskers.

1. Outlier Calculation

Outliers are determined using the **$1.5 \times \text{IQR}$ rule**, where values falling outside the lower and upper fences are considered outliers.

- **First Quartile (Q_1):** 17.25
- **Third Quartile (Q_3):** 23.25
- **Interquartile Range (IQR):** $Q_3 - Q_1 = 23.25 - 17.25 = \mathbf{6.0}$

Outlier Fences:

- **Lower Bound:** $Q_1 - 1.5 \times \text{IQR} = 17.25 - (1.5 \times 6.0) = \mathbf{8.25}$
- **Upper Bound:** $Q_3 + 1.5 \times \text{IQR} = 23.25 + (1.5 \times 6.0) = \mathbf{32.25}$

2. Result and Identification

Based on the calculated fences:

- Any data point less than 8.25 is a low outlier. (None in this dataset)
- Any data point greater than 32.25 is a high outlier.

The only value in the dataset greater than 32.25 is **35**.

Identified Outliers: $\mathbf{[35]}$

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:

```
import numpy as np

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

# Convert lists to NumPy arrays for calculation
x = np.array(advertising_spend)
y = np.array(daily_sales)

# Compute the correlation matrix
# The correlation coefficient between x and y is the off-diagonal element (0, 1) or (1, 0).
correlation_matrix = np.corrcoef(x, y)
correlation_coefficient = correlation_matrix[0, 1]

print(f"Advertising Spend (X): {advertising_spend}")
print(f"Daily Sales (Y): {daily_sales}")
print("-" * 50)
print(f"Correlation Coefficient (r): {correlation_coefficient:.4f}")
```

Output:

```
Advertising Spend (X): [200, 250, 300, 400, 500]
Daily Sales (Y): [2200, 2450, 2750, 3200, 4000]
--------------------------------------------------
Correlation Coefficient (r): 0.9926
```

Covariance (Direction of the Relationship)

1. **Computation:** Covariance measures how two variables change together.
2. **Interpretation:**

- **Positive Covariance (e.g., +100,000):** Indicates a **positive relationship**. As advertising spend increases, daily sales also tend to increase.
- **Negative Covariance (e.g., -50,000):** Indicates a **negative relationship**. As advertising spend increases, daily sales tend to decrease.
- **Near Zero Covariance:** Indicates **no clear linear relationship**.

3. **Limitation:** The value of covariance is not standardized (it depends on the units of the variables), making it difficult to judge the **strength** of the relationship or compare it across different datasets.

Correlation Coefficient (Strength and Direction)

1. **Computation:** The **Pearson correlation coefficient** (r) standardizes the covariance, dividing it by the product of the standard deviations of X and Y.
2. **Interpretation:** The result is a value between **-1 and +1**.
   - **$r = +1$:** Perfect **positive** linear relationship.
   - **r close to $+1$ (e.g., 0.90):** Very **strong positive** relationship.
   - **r close to 0 (e.g., 0.05): Weak or no** linear relationship.
   - **r close to -1 (e.g., -0.85):** Very **strong negative** relationship.
3. **Conclusion:** The correlation coefficient is the **key metric** to present to the marketing team, as it gives a standardized, easily interpretable measure of how reliable the connection is between spending more and seeing a rise in 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:

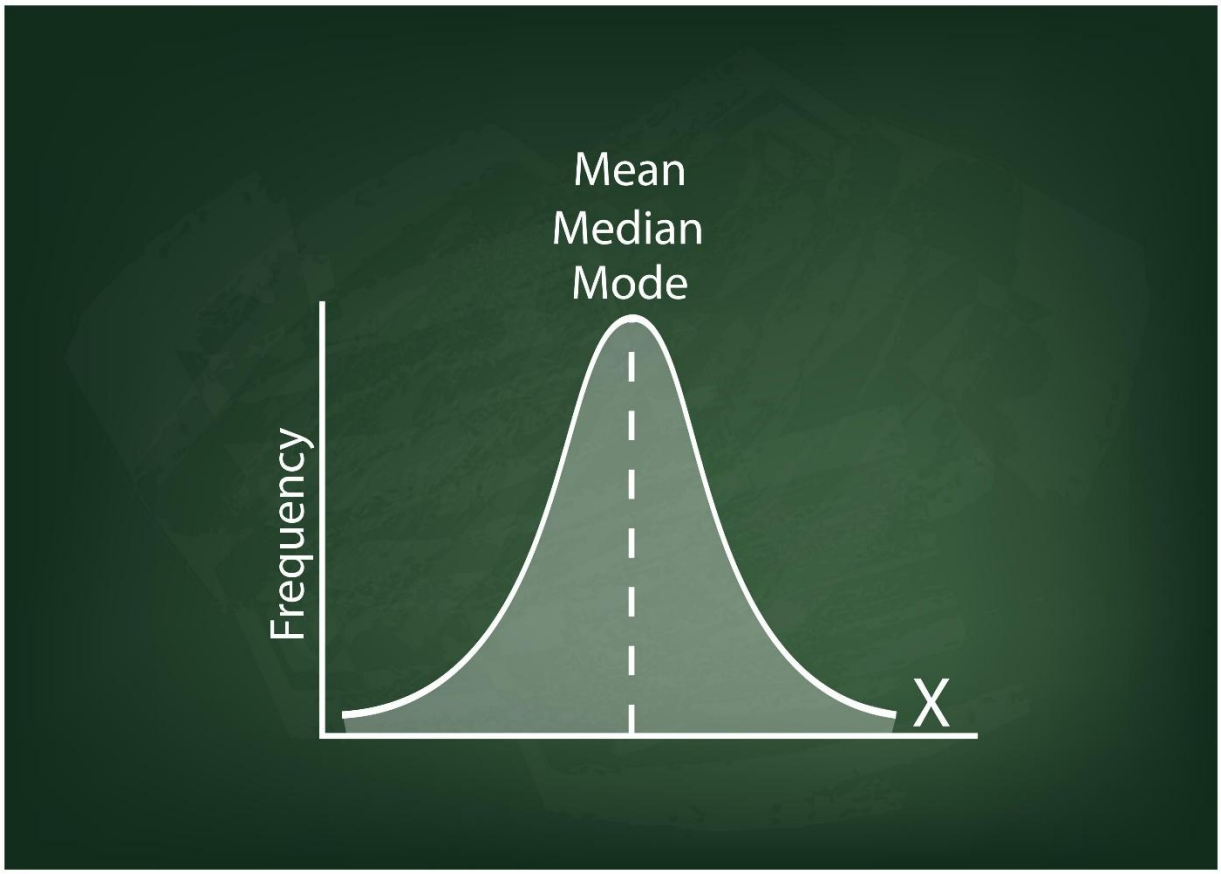
To understand the distribution of the customer satisfaction survey scores (on a 1-10 scale), I would use a combination of **summary statistics** and a **histogram**.

---

1. Summary Statistics and Visualizations

Statistic/Visualization	Purpose	Why it's Important
Mean (Average)	Measures the **typical level of satisfaction**.	Gives the team a single, easily understood value representing the central tendency of the scores.
Median	Measures the **midpoint** of the data.	Less sensitive to extreme scores than the mean. If the mean and median are far apart, it suggests the data is skewed.
Standard Deviation ()	Measures the **spread or variability** of the scores around the mean.	A **low** means scores are tightly clustered (consistent satisfaction). A **high** means scores are widely spread (diverse opinions).

Histogram



Shutterstock
| **Visualization** of the **frequency distribution**. | Shows the shape of the data: which scores are most common (the mode) and if the data is symmetric or skewed (e.g., are most scores high, or are they spread out?). |

---

2. Python Code for Histogram

The Python script below uses the matplotlib.pyplot library to create a histogram for the given survey scores.

Python

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

# The given survey data
survey_scores = [7, 8, 5, 9, 6, 7, 8, 9, 10, 4, 7, 6, 9, 8, 7]

# --- Create the Histogram ---
plt.figure(figsize=(8, 5))

# Plotting the histogram
# Bins are set to cover the 1-10 scale naturally.
# The range is set from 3.5 to 10.5 to center the bars on integer scores.
plt.hist(survey_scores,
         bins=np.arange(3.5, 11.5, 1),
         edgecolor='black',
         alpha=0.7,
         color='teal')

# Set labels and title
plt.title('Distribution of Customer Satisfaction Survey Scores (1-10)')
plt.xlabel('Satisfaction Score')
plt.ylabel('Frequency (Number of Customers)')

# Set x-ticks to display integer scores from 4 to 10
plt.xticks(range(4, 11))

# Display the grid and plot
plt.grid(axis='y', alpha=0.5)
plt.show()
```