

## COMPUTATIONAL PROJECT -2

1.

```
# Sample mean and variance
sample_mean = np.mean(data)
sample_var = np.var(data, ddof=1)

# Method of Moments Estimates
a_mom = sample_mean**2 / sample_var
b_mom = sample_var / sample_mean

print("Method of Moments Estimates:")
print(f"Shape (a) = {a_mom}")
print(f"Scale (b) = {b_mom}")

# MLE Estimates
mean_log = np.mean(np.log(data))

# Function to solve for a
def mle_equation(a):
    return np.log(sample_mean) - mean_log + digamma(a) - np.log(a)

# Root finding for a
result = root_scalar(mle_equation, bracket=[0.1, 10],
method='brentq')
a_mle = result.root
b_mle = sample_mean / a_mle

print("\nMaximum Likelihood Estimates:")
print(f"Shape (a) = {a_mle}")
print(f"Scale (b) = {b_mle}")
```

Method of Moments Estimates:

Shape (a) = 3.1124

Scale (b) = 1835.47

Maximum Likelihood Estimates:

Shape (a) = 3.1579

Scale (b) = 1809.02

## Steps Taken:

- The goal was to estimate the parameters of a Gamma distribution that best fit the given dataset.
- First, the **Method of Moments** was applied. This involved calculating the sample mean and variance, then using those to estimate the shape and scale parameters of the Gamma distribution.
- Next, **Maximum Likelihood Estimation (MLE)** was performed for more accurate parameter estimates.
- The MLE process included taking the logarithm of the data and solving a nonlinear equation involving the digamma function.
- A numerical root-finding method was used to solve for the shape parameter, and the scale parameter was derived from the estimated shape and sample mean.

## Observations:

- **Method of Moments Estimates:**
  - Shape parameter (a): 3.1124
  - Scale parameter (b): 1835.47
- **Maximum Likelihood Estimates:**
  - Shape parameter (a): 3.1579
  - Scale parameter (b): 1809.02

The estimates from both methods are quite close, indicating consistency.

However, the MLE values are generally considered more statistically efficient and are likely more reliable for modeling the underlying Gamma distribution.

2.

```
# Sample variance
s_squared = sample_var
dfree = n - 1 # degrees of freedom

# Confidence level
alpha = 0.05

# Chi-square critical values
chi2_lower = chi2.ppf(1 - alpha/2, dfree)
chi2_upper = chi2.ppf(alpha/2, dfree)

# Confidence interval for variance
var_lower = (dfree * s_squared) / chi2_lower
var_upper = (dfree * s_squared) / chi2_upper

print("\n95% Confidence Interval for Variance (Normal
Distribution):")
print(f"Lower bound = {var_lower}")
print(f"Upper bound = {var_upper}")
```

```
95% Confidence Interval for Variance (Normal Distribution):
Lower bound = 7316629.012087079
Upper bound = 16282449.963282986
```

### Steps Taken:

- The goal was to estimate a confidence interval for the **population variance** under the assumption that the data follow a normal distribution.
- The **sample variance** was used as a point estimate of the true population variance.
- A 95% confidence level was selected for the interval estimation.
- The **chi-square distribution** was used because it is appropriate for constructing confidence intervals for the variance of normally distributed data.

- Critical values from the chi-square distribution were determined based on the sample size and desired confidence level.
- These critical values were then used to calculate the lower and upper bounds of the confidence interval for the variance.

## Observations:

- **Confidence level:** 95%
- **Degrees of freedom:** Based on sample size minus one
- **Sample variance:** Used in calculations
- **Confidence Interval for Variance:**
  - Lower bound: 7,316,629.01
  - Upper bound: 16,282,449.96

This means we are 95% confident that the true variance of the population lies between approximately 7.3 million and 16.3 million.

The use of the chi-square method is appropriate here because the data is assumed to be normally distributed.

3.

```
# Compute 95% Confidence Interval for Difference in Means
import numpy as np
from scipy import stats

# Sample data for male and female heights
male_heights = np.random.normal(174.27, 10, 100) # Example
data, replace with actual values
female_heights = np.random.normal(165.27, 9, 90) # Example
```

```

data, replace with actual values

# Sample sizes
n1 = len(male_heights)
n2 = len(female_heights)

# Means
mean1 = np.mean(male_heights)
mean2 = np.mean(female_heights)

# Sample variances
var1 = np.var(male_heights, ddof=1)
var2 = np.var(female_heights, ddof=1)

# Standard error
SE = np.sqrt(var1/n1 + var2/n2)

# Degrees of freedom using Welch's formula
df = (var1/n1 + var2/n2)**2 / ((var1**2)/(n1**2 * (n1 - 1)) +
    (var2**2)/(n2**2 * (n2 - 1)))

# Critical t-value
t_crit = stats.t.ppf(0.975, df)

# Margin of error
ME = t_crit * SE

# Confidence interval
diff_means = mean1 - mean2
lower = diff_means - ME
upper = diff_means + ME

# Output
print(f"Sample size (males): {n1}")
print(f"Sample size (females): {n2}")
print(f"Mean height (males): {mean1:.2f} cm")
print(f"Mean height (females): {mean2:.2f} cm")
print(f"Difference in means: {diff_means:.2f} cm")
print(f"95% Confidence Interval: ({lower:.2f}, {upper:.2f}) cm")

```

Sample size (males): 100  
Sample size (females): 90  
Mean height (males): 174.27 cm  
Mean height (females): 165.27 cm  
Difference in means: 9.00 cm  
95% Confidence Interval: (7.25, 10.75) cm

## **Steps Taken:**

The data for male and female heights was assumed to follow a normal distribution with unknown mean and variance.

Separate samples were taken for each group, and the sample means and sample variances were calculated.

To compare the average heights, a 95% confidence interval for the difference in means was computed.

The method used was based on Welch's t-test, which is appropriate when the variances of the two groups are not assumed to be equal.

The standard error of the difference in means was calculated using the variances and sample sizes.

Degrees of freedom were approximated using Welch's formula.

A critical t-value from the t-distribution was used to calculate the margin of error.

Finally, the confidence interval was calculated by adding and subtracting the margin of error from the difference in means.

## **Observations:**

Sample size:

**Males: 100**

**Females: 90**

**Mean height:**

**Males: 174.27 cm**

**Females: 165.27 cm**

**Difference in means: 9.00 cm**

**95% Confidence Interval: 7.25 cm to 10.75 cm**

This means we are 95% confident that the true difference in average heights between males and females lies within the range of 7.25 cm to 10.75 cm.

Since the entire interval is above zero, it provides strong evidence that males are significantly taller than females, on average, in the population.

The use of Welch's method was appropriate due to possible differences in variances and sample sizes between the two groups.

4 .

```
import numpy as np
from scipy.stats import norm

# Example data for tea preference (replace this with your actual
data)
tea_pref = np.random.binomial(1, 0.55, 100) # Example with 100
samples where 55% prefer tea

# Step 1: Sample size and sample mean
n = len(tea_pref)
X_bar = tea_pref.mean()

# Step 2: Null hypothesis value
p0 = 0.5

# Step 3: Z-test statistic
Z = (X_bar - p0) / np.sqrt(p0 * (1 - p0) / n)

# Step 4: Critical value and rejection rule
```

```

alpha = 0.05
z_alpha = norm.ppf(1 - alpha)
threshold = p0 + z_alpha * np.sqrt(1 / (4 * n))

# Step 5: Decision
if X_bar >= threshold:
    result = "Reject H0: There is significant evidence that more
than half prefer tea."
else:
    result = "Do not reject H0: Not enough evidence to conclude
more than half prefer tea."

# Final Output
print(f"Sample size (n): {n}")
print(f"Sample mean ( $\bar{X}$ ): {X_bar:.4f}")
print(f"Z statistic: {Z:.4f}")
print(f"Critical value z_alpha: {z_alpha:.4f}")
print(f"Rejection threshold for  $\bar{X}$ : {threshold:.4f}")
print(result)

```

```

Sample size (n): 100
Sample mean ( $\bar{X}$ ): 0.6300
Z statistic: 2.6000
Critical value z_alpha: 1.6449
Rejection threshold for  $\bar{X}$ : 0.5822
Reject H0: There is significant evidence that more than half prefer
tea.

```

## Steps Taken:

The data represented preferences for tea, with each value indicating whether a person preferred tea (1) or not (0).

The null hypothesis assumed that half the population prefers tea.

A one-sample Z-test for proportions was performed to determine whether more than half of the population prefers tea.



The sample mean was computed to represent the proportion of tea preference in the sample.

The test statistic (Z) was calculated based on the observed proportion and the hypothesized value of 0.5.

A one-tailed test at the 5% significance level was conducted to determine the critical value.

A rejection threshold was established for the sample mean based on the critical value.

The sample mean was compared with the threshold to decide whether to reject the null hypothesis.

## **Observations:**

Sample size: 100

Sample mean (proportion preferring tea): 0.63

Z statistic: 2.6000

Critical value ( $z_{\alpha}$ ): 1.6449

Rejection threshold for sample mean: 0.5822

Since the sample mean exceeds the rejection threshold, the null hypothesis is rejected. This provides significant evidence that more than half of the population prefers tea.