COMPUTATIONAL PROJECT -2

Sleeping habits of students

Kagita Meenakshi-MA23BTECH11013

Meka Bhavya Kumari-MA23BTECH11016

Reddy Roshni-MA23BTECH11021

Introduction: The report analyses the sleep patterns and experiences of students. We conducted a survey of 342 students to get a better understanding of sleep habits and various factors affecting the sleep health. The survey investigated various factors affecting student sleep habits, including:

In this project we shall explore the following two factors:

- Students sleep satisfaction

- Have you ever missed a class / assignment / test due to being tired

Students sleep satisfaction:

1. Method of moments and maximum likelihood estimates:

a) Method of moments:

Code:

```
import numpy as np
#data
data = [4, 5, 3, 3, 3, 3, 3, 4, 1, 4, 3, 1, 4, 5, 3, 3, 4, 4, 4, 3, 2, 2, 3, 3, 3, 2, 4, 2, 2, 5, 3,
        3, 5, 3, 4, 3, 4, 1, 1, 4, 4, 3, 4, 3, 4, 3, 4, 4, 1, 1, 3, 3, 5, 3, 1, 2, 1, 4, 1, 3, 2, 2,
        3, 3, 2, 1, 4, 3, 2, 3, 3, 2, 4, 5, 3, 4, 4, 3, 4, 1, 3, 3, 3, 3, 3, 4, 4, 3, 4, 4, 3, 1, 5,
        3, 4, 3, 3, 3, 4, 3, 4, 3, 2, 3, 5, 4, 2, 3, 3, 3, 5, 2, 3, 2, 2, 1, 1, 3, 2, 2, 1, 4, 3, 4,
        1, 2, 4, 3, 3, 3, 2, 5, 1, 3, 1, 4, 3, 4, 4, 2, 4, 3, 5, 4, 2, 3, 2, 1, 3, 1, 3, 4, 4, 3, 1,
        2, 2, 4, 5, 3, 1, 3, 2, 2, 5, 2, 3, 5, 1, 4, 3, 4, 5, 3, 4, 1, 4, 4, 3, 4, 2, 1, 4, 3, 4, 3,
        3, 2, 4, 3, 2, 3, 5, 2, 3, 4, 4, 3, 4, 4, 3, 2, 5, 3, 1, 4, 1, 3, 3, 4, 4, 3, 1, 4, 3, 2, 4,
        3, 2, 5, 4, 5, 3, 2, 5, 3, 4, 4, 2, 2, 4, 3, 3, 4, 3, 1, 3, 3, 4, 2, 3, 3, 3, 3, 4, 3, 3, 2,
        5, 3, 3, 3, 1, 3, 3, 4, 3, 3, 5, 3, 3, 4, 1, 3, 3, 3, 2, 2, 1, 2, 1, 1, 3, 3, 2, 2, 4, 4, 3,
        5, 3, 3, 3, 4, 4, 3, 3, 3, 4, 3, 2, 4, 4, 3, 2, 4, 3, 4, 2, 1, 4, 5, 5, 2, 4, 1, 4, 3, 4, 2,
        3, 4, 3, 5, 4, 4, 3, 3, 2, 2, 4, 3, 4, 3, 4, 2, 2, 2, 4, 4, 4, 3, 1, 2, 1, 2, 2, 4, 3, 4, 4,
        3
       ]

data = np.array(data)
# Sample mean and variance
mean = np.mean(data)
var = np.var(data, ddof=1)
# Method of Moments estimates
a_mom = mean**2 / var
b_mom = var / mean
```

```
# Output
print("Method of Moments Estimates:->")
print(f"Sample Mean: {mean:.4f}")
print(f"Sample Variance: {var:.4f}")
print(f"MoM Estimate of a: {a_mom:.4f}")
print(f"MoM Estimate of b: {b_mom:.4f}")
```

Output:

```
Method of Moments Estimates:->
Sample Mean: 3.0322
Sample Variance: 1.1925
MoM Estimate of a:7.7098
MoM Estimate of b:0.3933
```

b) Maximum likelihood estimates:

Code:

```
import numpy as np
from scipy.special import psi  # digamma function
from scipy.optimize import fsolve
data = [4, 5, 3, 3, 3, 3, 3, 4, 1, 4, 3, 1, 4, 5, 3, 3, 4, 4, 4, 3, 2, 2, 3, 3, 3, 2, 4, 2, 2, 5, 3,
      3, 5, 3, 4, 3, 4, 1, 1, 4, 4, 3, 4, 3, 4, 3, 4, 4, 1, 1, 3, 3, 5, 3, 1, 2, 1, 4, 1, 3, 2, 2,
      3, 3, 2, 1, 4, 3, 2, 3, 3, 2, 4, 5, 3, 4, 4, 3, 4, 1, 3, 3, 3, 3, 3, 4, 4, 3, 4, 4, 3, 1, 5,
      3, 4, 3, 3, 3, 4, 3, 4, 3, 2, 3, 5, 4, 2, 3, 3, 3, 5, 2, 3, 2, 2, 1, 1, 3, 2, 2, 1, 4, 3, 4,
      1, 2, 4, 3, 3, 3, 2, 5, 1, 3, 1, 4, 3, 4, 4, 2, 4, 3, 5, 4, 2, 3, 2, 1, 3, 1, 3, 4, 4, 3, 1,
      2, 2, 4, 5, 3, 1, 3, 2, 2, 5, 2, 3, 5, 1, 4, 3, 4, 5, 3, 4, 1, 4, 4, 3, 4, 2, 1, 4, 3, 4, 3,
      3, 2, 4, 3, 2, 3, 5, 2, 3, 4, 4, 3, 4, 4, 3, 2, 5, 3, 1, 4, 1, 3, 3, 4, 4, 3, 1, 4, 3, 2, 4,
      3, 2, 5, 4, 5, 3, 2, 5, 3, 4, 4, 2, 2, 4, 3, 3, 4, 3, 1, 3, 3, 4, 2, 3, 3, 3, 3, 4, 3, 3, 2,
      5, 3, 3, 3, 1, 3, 3, 4, 3, 3, 5, 3, 3, 4, 1, 3, 3, 3, 2, 2, 1, 2, 1, 1, 3, 3, 2, 2, 4, 4, 3,
      5, 3, 3, 3, 4, 4, 3, 3, 3, 4, 3, 2, 4, 4, 3, 2, 4, 3, 4, 2, 1, 4, 5, 5, 2, 4, 1, 4, 3, 4, 2,
      3, 4, 3, 5, 4, 4, 3, 3, 2, 2, 4, 3, 4, 3, 4, 2, 2, 2, 4, 4, 4, 3, 1, 2, 1, 2, 2, 4, 3, 4, 4,
      3
]
data = np.array(data)
#sample mean,log of sample mean and mean of logs
x_bar = np.mean(data)
log_x_bar = np.log(x_bar)
mean_log_x = np.mean(np.log(data))

#MLE equation: ψ(a) - log(a) = mean_log_x - log(x̄)
rhs = mean_log_x - log_x_bar

#function to solve for a
def mle_equation(a):
   return psi(a) - np.log(a) - rhs
```

```
#Initial guess for a
initial_guess = 1.0

#Use numerical solver to solve for a
a_mle = fsolve(mle_equation, initial_guess)[0]

#Compute b using b = x̄ / a
b_mle = x_bar / a_mle

#output
print("Maximum Likelihood Estimation(Gamma):->")
print(f"Sample mean (x̄): {x_bar:.4f}")
print(f"Mean of log(x): {mean_log_x:.4f}")
print(f"MLE Estimate for shape (a): {a_mle:.4f}")
print(f"MLE Estimate for scale (b): {b_mle:.4f}")
```

Output:

```
Maximum Likelihood Estimation(Gamma):->
Sample mean (x̄): 3.0322
Mean of log(x): 1.0248
MLE Estimate for shape (a): 6.0838
MLE Estimate for scale (b):0.4984
```

Conclusion:

The MLE fit indicates a slightly lower shape and higher rate, meaning the MLE estimates a distribution more concentrated around the mean, which may better reflect the central tendency and skewness of the actual data. Since MLE is generally more robust, we would prefer the MLE estimates for further modeling.

2. 95% confidence interval for variance:

Code:

```
import numpy as np
from scipy.stats import chi2

#data
data = [4, 5, 3, 3, 3, 3, 3, 4, 1, 4, 3, 1, 4, 5, 3, 3, 4, 4, 4, 3, 2, 2, 3, 3, 3, 2, 4, 2, 2, 5, 3,
    3, 5, 3, 4, 3, 4, 1, 1, 4, 4, 3, 4, 3, 4, 3, 4, 4, 1, 1, 3, 3, 5, 3, 1, 2, 1, 4, 1, 3, 2, 2,
    3, 3, 2, 1, 4, 3, 2, 3, 3, 2, 4, 5, 3, 4, 4, 3, 4, 1, 3, 3, 3, 3, 3, 4, 4, 3, 4, 4, 3, 1, 5,
    3, 4, 3, 3, 3, 4, 3, 4, 3, 2, 3, 5, 4, 2, 3, 3, 3, 5, 2, 3, 2, 2, 1, 1, 3, 2, 2, 1, 4, 3, 4,
    1, 2, 4, 3, 3, 3, 2, 5, 1, 3, 1, 4, 3, 4, 4, 2, 4, 3, 5, 4, 2, 3, 2, 1, 3, 1, 3, 4, 4, 3, 1,
    2, 2, 4, 5, 3, 1, 3, 2, 2, 5, 2, 3, 5, 1, 4, 3, 4, 5, 3, 4, 1, 4, 4, 3, 4, 2, 1, 4, 3, 4, 3,
    3, 2, 4, 3, 2, 3, 5, 2, 3, 4, 4, 3, 4, 4, 3, 2, 5, 3, 1, 4, 1, 3, 3, 4, 4, 3, 1, 4, 3, 2, 4,
    3, 2, 5, 4, 5, 3, 2, 5, 3, 4, 4, 2, 2, 4, 3, 3, 4, 3, 1, 3, 3, 4, 2, 3, 3, 3, 3, 4, 3, 3, 2,
    5, 3, 3, 3, 1, 3, 3, 4, 3, 3, 5, 3, 3, 4, 1, 3, 3, 3, 2, 2, 1, 2, 1, 1, 3, 3, 2, 2, 4, 4, 3,
    5, 3, 3, 3, 4, 4, 3, 3, 3, 4, 3, 2, 4, 4, 3, 2, 4, 3, 4, 2, 1, 4, 5, 5, 2, 4, 1, 4, 3, 4, 2,
```

```
    3, 4, 3, 5, 4, 4, 3, 3, 2, 2, 4, 3, 4, 3, 4, 2, 2, 2, 4, 4, 4, 3, 1, 2, 1, 2, 2, 4, 3, 4, 4,
    3
]
data = np.array(data)

#size and sample variance
n = len(data)
s2 = np.var(data, ddof=1)

alpha = 0.05
df = n - 1

chi2_lower = chi2.ppf(alpha / 2, df)     # lower critical value
chi2_upper = chi2.ppf(1 - alpha / 2, df) # upper critical value

#confidence interval for Variance
lower_bound = (df * s2) / chi2_upper
upper_bound = (df * s2) / chi2_lower

#output
print("95% Confidence Interval for Variance:->")
print(f"Sample Variance(s²): {s2:.4f}")
print(f"Confidence Interval: ({lower_bound:.4f}, {upper_bound:.4f})")
```

Output:

```
95% Confidence Interval for Variance:->
Sample Variance(s²):1.1925
Confidence Interval:(1.0320,1.3939)
```

Conclusion:

This range tells us that the variability in sleep satisfaction is moderate. Since the entire interval is above 1, it suggests that students' satisfaction ratings are not tightly clustered around the mean — there's noticeable spread, possibly due to differences in lifestyle, stress, or sleep duration.

3. 95% confidence interval for difference in means:

Code:

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

data1 = [4,5,3,3,3,4,5,3,2,3,4,3,4,4,3,3,1,1,5,2,1,2,3,2,1,2,4,1,3,
     3,4,4,5,3,3,3,4,3,4,3,4,2,2,1,3,2,2,1,3,4,2,4,3,1,4,3,4,4,
     2,4,3,4,3,1,2,3,3,2,5,4,3,4,4,3,4,3,3,4,4,4,3,2,5,3,1,4,3,
     3,4,4,3,4,4,3,5,2,4,3,3,3,4,3,3,2,3,1,3,3,4,3,3,5,3,3,2,2,
     2,3,2,2,3,5,4,3,3,1,1,3,2,3,4,3,2,3,4,2,1,3] #female
```

```
data2 = [3,3,3,4,1,4,1,3,3,4,4,4,3,2,2,3,3,4,2,2,5,3,3,5,1,1,4,3,4,
        4,4,4,3,3,3,1,1,4,3,2,3,4,3,3,3,2,4,5,3,4,3,4,3,3,3,4,3,4,
        3,1,4,3,2,3,5,2,3,3,3,5,3,2,1,4,1,3,3,2,5,1,3,5,2,2,1,3,3,
        4,4,3,1,2,4,5,1,2,2,3,5,1,4,3,5,1,4,4,2,1,4,3,3,2,4,3,2,5,
        2,3,4,3,1,1,3,2,2,5,4,3,5,3,4,2,2,4,3,4,3,1,3,4,2,3,3,3,5,
        3,3,3,4,1,3,3,1,1,1,3,4,4,3,3,3,4,4,3,3,3,2,4,4,3,2,4,4,2,
        4,5,5,2,4,4,4,3,4,5,4,3,2,4,3,4,2,2,4,4,4,3,1,2,2,2,4,4,4,
        3] #male

data1 = np.array(data1)
data2 = np.array(data2)

#Calculate sample means and variances
mean1 = np.mean(data1)
mean2 = np.mean(data2)
var1 = np.var(data1, ddof=1)
var2 = np.var(data2, ddof=1)
n1 = len(data1)
n2 = len(data2)

#standard error
se = np.sqrt(var1 / n1 + var2 / n2)

#degrees of freedom
df = (var1 / n1 + var2 / n2)*2 / ((var12) / (n12 * (n1 - 1)) + (var22) / (n2*2 * (n2 - 1)))


t_crit = t.ppf(0.975, df)  # 95% CI


#Compute confidence interval
diff = mean1 - mean2
lower = diff - t_crit * se
upper = diff + t_crit * se

#output
print("95% Confidence Interval for Difference in Means")
print(f"Mean of data1: {mean1:.4f}")
print(f"Mean of data2: {mean2:.4f}")
print(f"Difference(data1-data2): {diff:.4f}")
print(f"Confidence Interval: ({lower:.4f}, {upper:.4f})")
```

Output:

95% Confidence Interval for Difference in Means
Mean of data1:3.0145

Mean of data2: 3.0441
Difference(data1-data2): -0.0296
Confidence Interval: (-0.2640,0.2047)

Conclusion:
The small difference in mean sleep satisfaction between male and female students (−0.0296) is not statistically significant, as the 95% confidence interval includes zero.

Have you ever missed a class / assignment / test due to being tired?

4. Comprehensively test the hypothesis at a level of significance of 0.05.

Code-

```
 import math
from scipy.stats import norm


# Survey data
x = 222         # Number of "Yes" responses (successes)
n = 342       # Total number of responses
p0 = 0.5       # Null hypothesis proportion


# Sample proportion
p_hat = x / n


# Standard error under H0
se = math.sqrt(p0 * (1 - p0) / n)


# Z-statistic
z = (p_hat - p0) / se


# p-value for one-tailed test (H1: p > 0.5)
p_value = 1 - norm.cdf(z)


# Output results
print(f"Sample Proportion (p̂): {p_hat:.4f}")
print(f"Z-statistic: {z:.4f}")
```

```
print(f"P-value: {p_value:.4f}")


# Conclusion
alpha = 0.05
if p_value < alpha:
    print("Reject the null hypothesis (H0). There is significant evidence that p > 0.5.")
else:
    print("Fail to reject the null hypothesis (H0). Not enough evidence that p > 0.5.")
```

Output:

Z-statistic: 5.5155

P-value: 0.0000

Reject the null hypothesis (H0). There is significant evidence that p > 0.5.

Conclusion:

There is strong statistical evidence that more than half of the students have missed a class/test due to being tired.This highlights the potential academic consequences of poor sleep patterns among college students, emphasizing the need for better sleep management.