

**SCHOOL OF COMPUTER SCIENCES**

**Academic Session 2019/2020, Semester 2**

**CPT 115 (Mathematical Methods for Computer Science)**

**Dr. Pantea Keikhosrokiani**

**ASSIGNMENT 2**

**Title: Data collection, analysis, and interpretation.**

**(Data collection, calculation, descriptive analysis and displaying data, regression analysis and hypotheses testing)**

**Due date: Monday, 15th June 2020**

**Group Members:**

|  |  |  |
| --- | --- | --- |
| **NO.** | **FULL NAME** | **STUDENT’S MATRIC NO.** |
| 1 | TAKI EDDINE TORKI |  |
| 2 | WONG CHONG YANG |  |
| 3 | NURUL FASIHAH BINTI ABD RAZAK | 128372 |
| 4 | WAN ATIFAH NUR ALIA | 128409 |
| 5 | NOR ISLAHIAH BINTI MUHAMMED | 149891 |

**PART I - Data collection**

Please refer “Sheet1” in “File for data collection.xlsx”.

**PART II - Calculation**

a. Create frequency table and calculate relative frequency for your dataset.

| **Calories per day (x)** | **Tally** | **Frequency (f)** | **Relative frequency (rf)** |
| --- | --- | --- | --- |
| 352 | / | 1 | 1/35=0.0285714285714286 |
| 652 | / | 1 | 1/35=0.0285714285714286 |
| 880 | / | 1 | 1/35=0.0285714285714286 |
| 934 | / | 1 | 1/35=0.0285714285714286 |
| 940 | // | 2 | 2/35=0.0571428571428571 |
| 980 | / | 1 | 1/35=0.0285714285714286 |
| 984 | / | 1 | 1/35=0.0285714285714286 |
| 1020 | / | 1 | 1/35=0.0285714285714286 |
| 1044 | / | 1 | 1/35=0.0285714285714286 |
| 1060 | / | 1 | 1/35=0.0285714285714286 |
| 1095 | / | 1 | 1/35=0.0285714285714286 |
| 1170.5 | / | 1 | 1/35=0.0285714285714286 |
| 1196 | / | 1 | 1/35=0.0285714285714286 |
| 1205 | / | 1 | 1/35=0.0285714285714286 |
| 1210 | / | 1 | 1/35=0.0285714285714286 |
| 1221.5 | / | 1 | 1/35=0.0285714285714286 |
| 1249 | / | 1 | 1/35=0.0285714285714286 |
| 1294 | / | 1 | 1/35=0.0285714285714286 |
| 1310 | / | 1 | 1/35=0.0285714285714286 |
| 1333 | / | 1 | 1/35=0.0285714285714286 |
| 1359 | / | 1 | 1/35=0.0285714285714286 |
| 1368 | / | 1 | 1/35=0.0285714285714286 |
| 1381 | // | 2 | 2/35=0.0571428571428571 |
| 1390 | /// | 3 | 3/35=0.0857142857142857 |
| 1417 | / | 1 | 1/35=0.0285714285714286 |
| 1483 | / | 1 | 1/35=0.0285714285714286 |
| 1611 | / | 1 | 1/35=0.0285714285714286 |
| 1627.5 | / | 1 | 1/35=0.0285714285714286 |
| 1651 | / | 1 | 1/35=0.0285714285714286 |
| 1701 | / | 1 | 1/35=0.0285714285714286 |
| 2456 | / | 1 | 1/35=0.0285714285714286 |

Total of frequency = 35

Relative frequency (rf) = Frequency (f) / Total of frequency

b. Calculate the minimum, maximum, mean, and mode values for your collected dataset.

Minimum

= The lowest value in the dataset (x)

= 352

Maximum

= The highest value in the dataset (x)

= 2456

Mean

\* Please refer the next page

Mode:

= The most frequent value in the dataset (x)

= 1390 (frequency: 3)

Mean ( )

= The average value inside the dataset (x)

= Sum of observations / Number of observations

= ∑xf / ∑n or ∑xf / ∑f

= Total of product (xf) / Total frequency

= 43675.5/35

= 1247.87

|  |  |  |  |
| --- | --- | --- | --- |
| **Calories per day (x)** | **Tally** | **Frequency (f)** | **Product (x.f)** |
| 352 | / | 1 | 352x1=352 |
| 652 | / | 1 | 652x1=652 |
| 880 | / | 1 | 880x1=880 |
| 934 | / | 1 | 934x1=934 |
| 940 | // | 2 | 940x2=1880 |
| 980 | / | 1 | 980x1=980 |
| 984 | / | 1 | 984x1=984 |
| 1020 | / | 1 | 1020x1=1020 |
| 1044 | / | 1 | 1044x1=1044 |
| 1060 | / | 1 | 1060x1=1060 |
| 1095 | / | 1 | 1095x1=1095 |
| 1170.5 | / | 1 | 1170.5x1=1170.5 |
| 1196 | / | 1 | 1196x1=1196 |
| 1205 | / | 1 | 1205x1=1205 |
| 1210 | / | 1 | 1210x1=1210 |
| 1221.5 | / | 1 | 1221.5x1=1221.5 |
| 1249 | / | 1 | 1249x1=1249 |
| 1294 | / | 1 | 1294x1=1294 |
| 1310 | / | 1 | 1310x1=1310 |
| 1333 | / | 1 | 1333x1=1333 |
| 1359 | / | 1 | 1359x1=1359 |
| 1368 | / | 1 | 1368x1=1368 |
| 1381 | // | 2 | 1381x2=2762 |
| 1390 | /// | 3 | 1390x3=4170 |
| 1417 | / | 1 | 1417x1=1417 |
| 1483 | / | 1 | 1483x1=1483 |
| 1611 | / | 1 | 1611x1=1611 |
| 1627.5 | / | 1 | 1627.5x1=1627.5 |
| 1651 | / | 1 | 1651x1=1651 |
| 1701 | / | 1 | 1701x1=1701 |
| 2456 | / | 1 | 2456x1=2456 |

Total of frequency (f) = 35

Total of product (xf) = 43675.5

c. Calculate standard deviation and variance for your collected dataset.



Standard deviation ( )

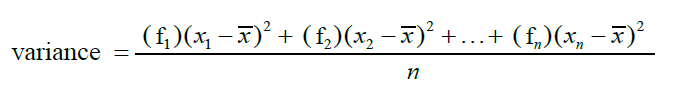
=

=

= 346.85

Variance

= the average of the square of the difference in value of a datum from the mean



=

= Total of [(frequency of the datum)(the square of datum - mean)] / Total of frequency

\* Please refer the next page for calculation table

= 4210592.17 / 35

= 120302.63

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Calories**  **per day**  **(x)** | **Tally** | **Frequency (f)** | **x - mean (1247.87)** | **(x - mean)^2** | **((x - mean)^2)\*f** |
| 352 | / | 1 | -895.87 | 802583.06 | 802583.06 |
| 652 | / | 1 | -595.87 | 355061.06 | 355061.06 |
| 880 | / | 1 | -367.87 | 135328.34 | 135328.34 |
| 934 | / | 1 | -313.87 | 98514.38 | 98514.38 |
| 940 | // | 2 | -307.87 | 94783.94 | 189567.87 |
| 980 | / | 1 | -267.87 | 71754.34 | 71754.34 |
| 984 | / | 1 | -263.87 | 69627.38 | 69627.38 |
| 1020 | / | 1 | -227.87 | 51924.74 | 51924.74 |
| 1044 | / | 1 | -203.87 | 41562.98 | 41562.98 |
| 1060 | / | 1 | -187.87 | 35295.14 | 35295.14 |
| 1095 | / | 1 | -152.87 | 23369.24 | 23369.24 |
| 1170.5 | / | 1 | -77.37 | 5986.12 | 5986.12 |
| 1196 | / | 1 | -51.87 | 2690.50 | 2690.50 |
| 1205 | / | 1 | -42.87 | 1837.84 | 1837.84 |
| 1210 | / | 1 | -37.87 | 1434.14 | 1434.14 |
| 1221.5 | / | 1 | -26.37 | 695.38 | 695.38 |
| 1249 | / | 1 | 1.13 | 1.28 | 1.28 |
| 1294 | / | 1 | 46.13 | 2127.98 | 2127.98 |
| 1310 | / | 1 | 62.13 | 3860.14 | 3860.14 |
| 1333 | / | 1 | 85.13 | 7247.12 | 7247.12 |
| 1359 | / | 1 | 111.13 | 12349.88 | 12349.88 |
| 1368 | / | 1 | 120.13 | 14431.22 | 14431.22 |
| 1381 | // | 2 | 133.13 | 17723.60 | 35447.19 |
| 1390 | /// | 3 | 142.13 | 20200.94 | 60602.81 |
| 1417 | / | 1 | 169.13 | 28604.96 | 28604.96 |
| 1483 | / | 1 | 235.13 | 55286.12 | 55286.12 |
| 1611 | / | 1 | 363.13 | 131863.40 | 131863.40 |
| 1627.5 | / | 1 | 379.63 | 144118.94 | 144118.94 |
| 1651 | / | 1 | 403.13 | 162513.80 | 162513.80 |
| 1701 | / | 1 | 453.13 | 205326.80 | 205326.80 |
| 2456 | / | 1 | 1208.13 | 1459578.10 | 1459578.10 |

Total of frequency (f) = 35

Total of [(frequency of the datum)(the square of datum - mean)] = 4210592.17

**PART III - Descriptive analysis & Displaying Data**

**PSEUDOCODES**

1. Start
2. Import modules numpy as np
3. Import function mode from modules statistics
4. Import modules matplotlib.plyplot as plt
5. Define and initialize array variable for self-reported total calories intake as calories
6. Find the maximum value of the array (calories)
7. Display the maximum value
8. Find the minimum value of the array (calories)
9. Display the minimum value
10. Find the mode value of the array (calories) using statistics module
11. Display the mode value using statistics module
12. Find the frequency of the mode value of the array (calories) using statistics module
13. Display the frequency of the mode value using statistics module
14. Calculate the mean value of the array (calories) using numpy module
15. Display the mean value using numpy module
16. Calculate the variance value of the array (calories) using numpy module
17. Display the variance value using numpy module
18. Calculate the standard deviation value of the array (calories) using numpy module
19. Display the standard deviation value using numpy module
20. Set up boxplot based on the array (calories) using matplotlib.plyplot module
21. Display the boxplot using matplotlib.plyplot module
22. Set up histogram based on the array (calories) using matplotlib.plyplot module
23. Display the histogram using matplotlib.plyplot module
24. End

**FLOWCHART**

Import required modules (numpy, statistics, matplotlib.plyplot)

Define and initialize array variable for self-reported total calories intake (calories)

Find and display the maximum value of the array (calories)

Find and display the minimum value of the array (calories)

Find and display the mode value of the array (calories) using statistics module

Find and display the frequency of the mode value of the array (calories)

using statistics module

Calculate and display the mean value of the array (calories) using numpy module

Calculate and display the variance value of the array (calories) using numpy module

Calculate and display the standard deviation value of the array (calories)

using numpy module

Set up and display boxplot based on the array (calories) using matplotlib.plyplot module

Set up and display histogram based on the array (calories)

using matplotlib.plyplot module

**CODES**

#import the required modules

import numpy as np

from statistics import mode

import matplotlib.pyplot as plt

calories = [1417, 1611, 1483, 1651, 1333, 1368, 1359, #CY

1390, 1381, 652, 1390, 352, 1381, 1390, #Taki

1060, 1205 ,1210 ,1020, 940, 880, 940, #Fasihah

1701,1170.5,1627.5,2456,1221.5,1294,1044, #Islahiah

934, 1095, 1196, 1310, 980, 984, 1249] #Atifah

print("Maximum calories intake:",max(calories))

print("\nMinimum calories intake:",min(calories))

print("\nMode calories intake:", mode(calories))

print("\nFrequency of mode:",calories.count(mode(calories)))

print("\nMean calories intake:", np.mean(calories))

print("\nVariance:",np.var(calories))

print("\nStandard deviation:",np.std(calories))

plt.title('Boxplot for calories intake')

plt.boxplot(calories)

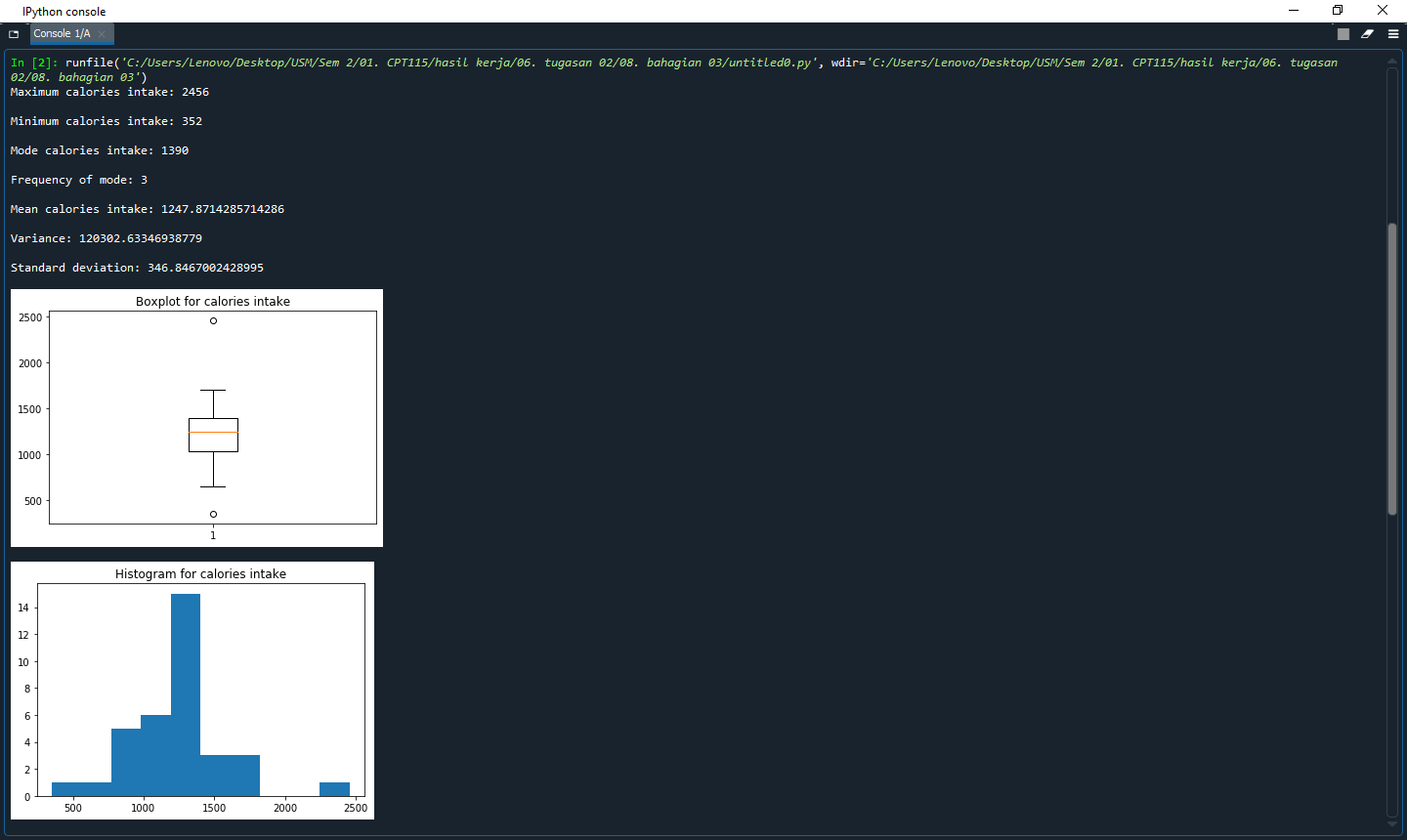
plt.show()

plt.title('Histogram for calories intake')

plt.hist(calories)

plt.show()

**RESULT**



**PART IV - Regression Analysis**

**Pseudo code**

1. Begin
2. Create a function, linear\_func, that returns slope \* x + intercept
3. Create a function, linear\_regression, to plot the graphs
   1. Calculate the slope and y-intercept of the linear regression
   2. Use linear\_func to calculate the coresponding y values of the linear regression
   3. Plot the graph of y against x
   4. Plot the graph of the linear regression
   5. Display the graph
4. Main function
   1. Assign the calorie intakes of all the members into calorie
   2. Assign the weights of all the members into weight
   3. Assign the physical activities of all the members into phy
   4. Plot the graph of weights againt calorie intakes and the linear regression
      1. Set the title of a graph as ‘Correlation between the calorie intakes and the weights of students’
      2. Label the x-axis of the graph as ‘Calorie intake (kcal)
      3. Label the y-axis of the graph as ‘Weight (kg)’
      4. Pass calories and weight to linear\_regression to
   5. Plot the graph of physical activities againt calorie intakes and the linear regression
      1. Set the title of a graph as ‘Correlation between the calorie intakes and the weights of students’
      2. Label the x-axis of the graph as ‘Calorie intake (kcal)
      3. Label the y-axis of the graph as ‘Weight (kg)’
      4. Pass phy and weight to linear\_regression
   6. Plot the graph of physical activities againt calorie intakes and the linear regression
      1. Set the title of a graph as ‘Correlation between the calorie intakes and the weights of students’
      2. Label the x-axis of the graph as ‘Calorie intake (kcal)
      3. Label the y-axis of the graph as ‘Weight (kg)’
      4. Pass phy and calories to linear\_regression
5. End

**Flowchart**

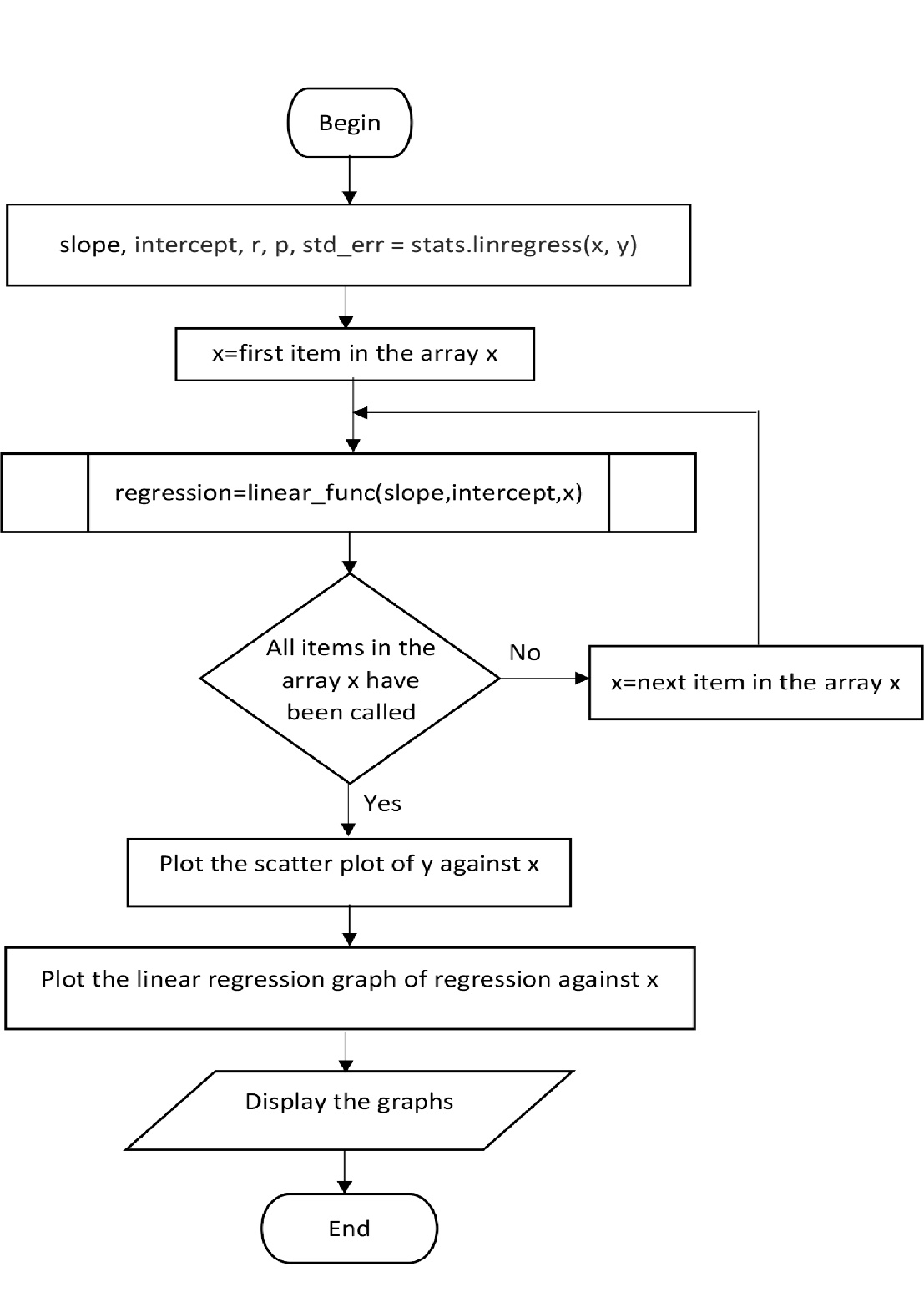
linear\_func(slope, x, intercept):

Begin

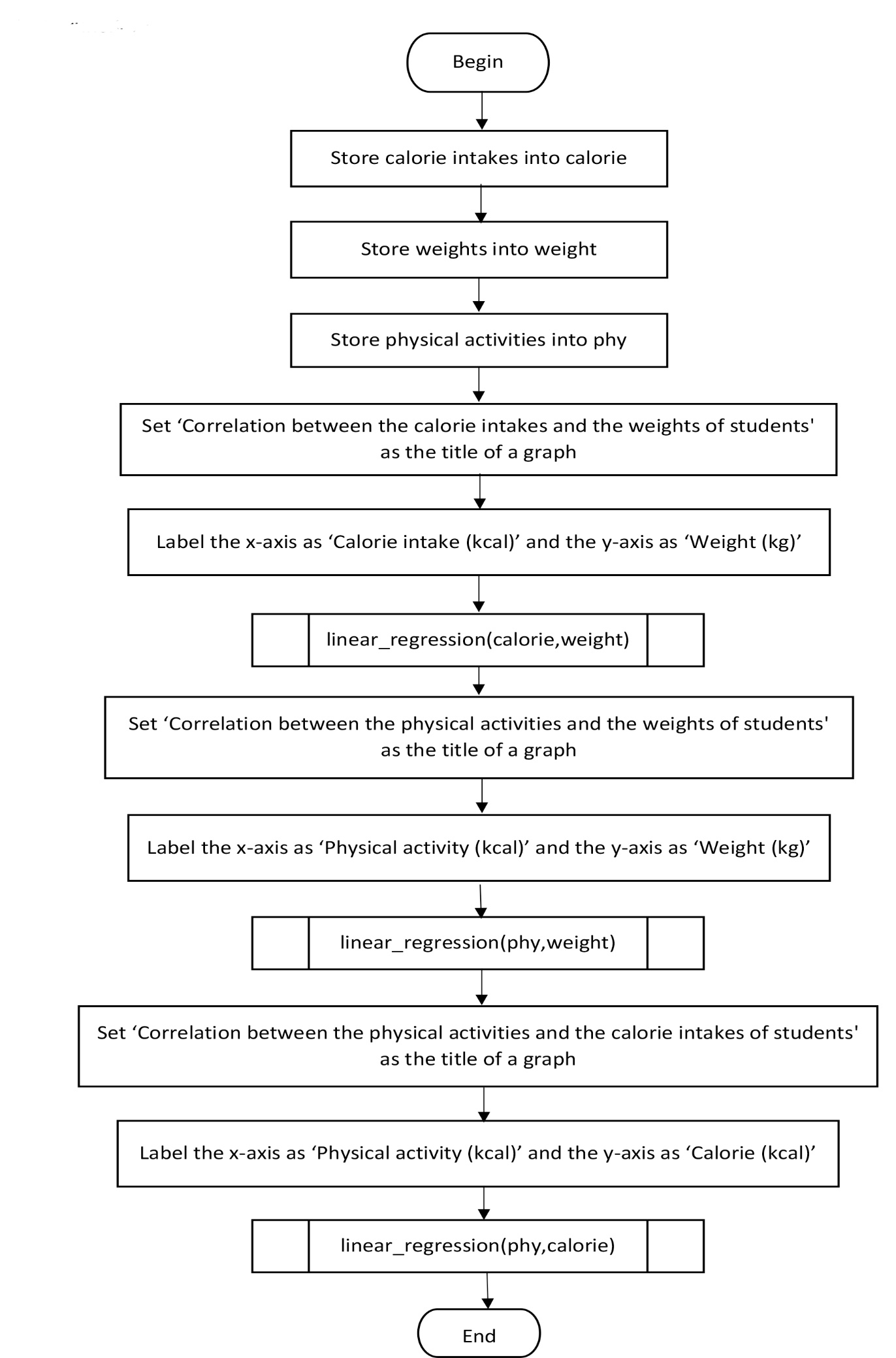
return slope \* x + intercept

End

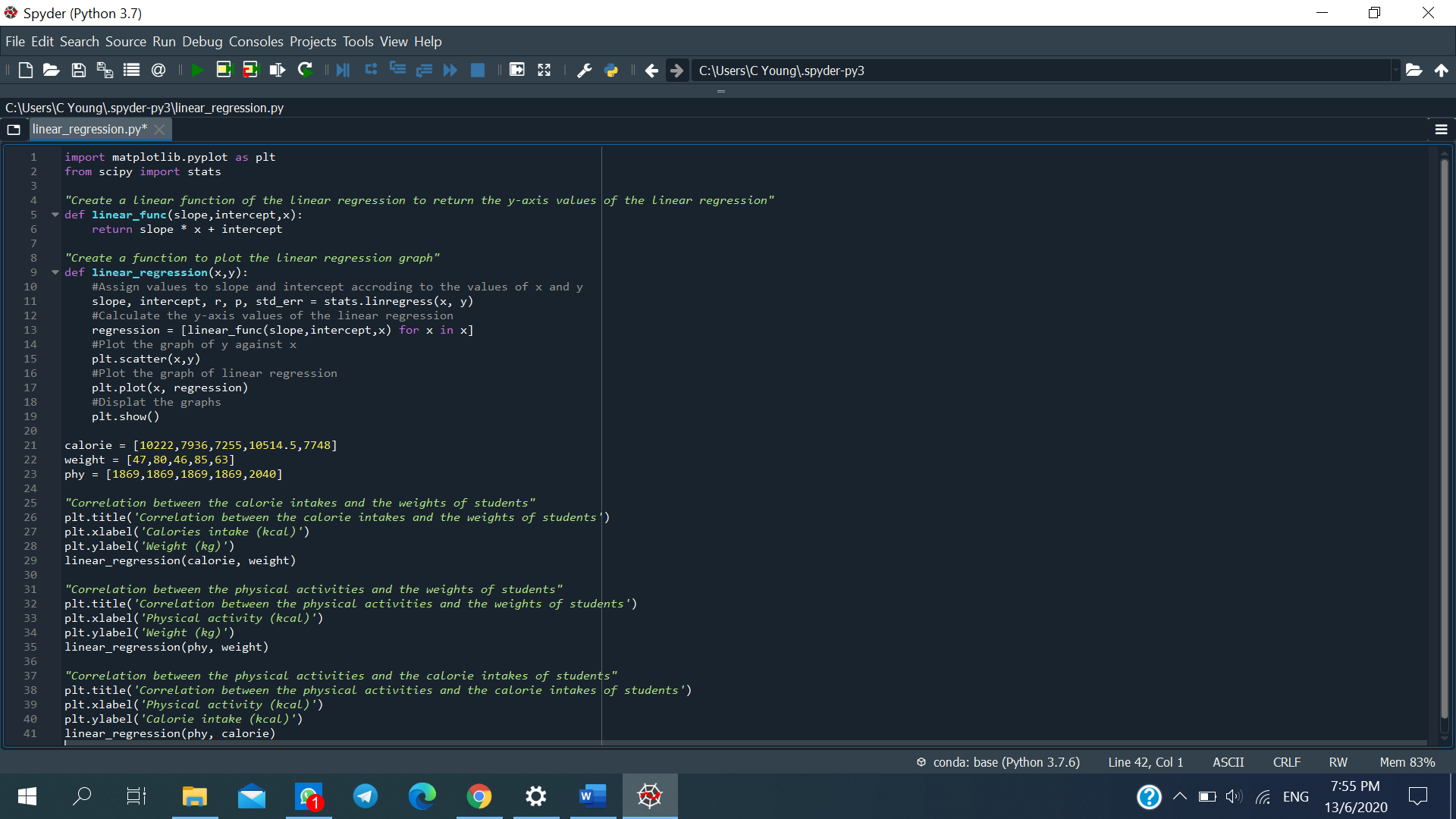
linear\_regression(x, y):



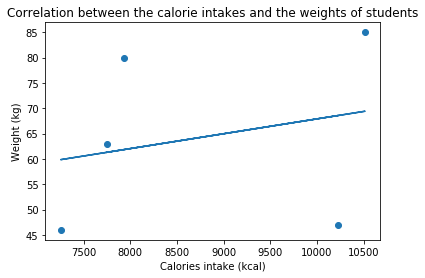
**Main function:**



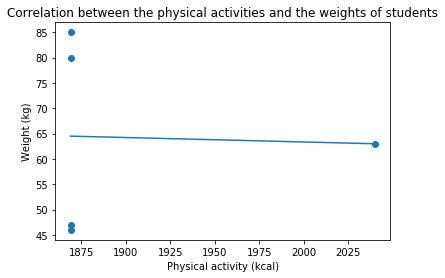
**Python codes**



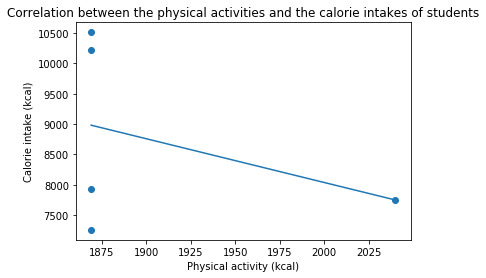
**Outputs**



The linear regression graph has a positive slope. Hence, the graph shows positive correlation between the calorie intakes and the weight of students. As the calorie intakes of the students increase, the weights of the students also increase.



The slope of the linear regression graph is insignificant. There is no correlation between the physical activities and the weights of the students. The physical activities of students do not affect their weights.



The linear regression graph has a negative slope. Hence, the graph shows negative correlation between the physical activities and the calorie intakes of the students. As the physical activities of students increase, their calorie intakes decrease.

**PART V - Hypotheses testing**

1. A normally distributed calories per day intakes from **5** students is known to have a mean of **1247.87** and a standard deviation of **346.85**. For the calories per day intake from **2** students (Wong Chong Yang and Islahiah), the average is **592.47**.



μ0 = Population mean

= Sample mean



Confidence = 99%

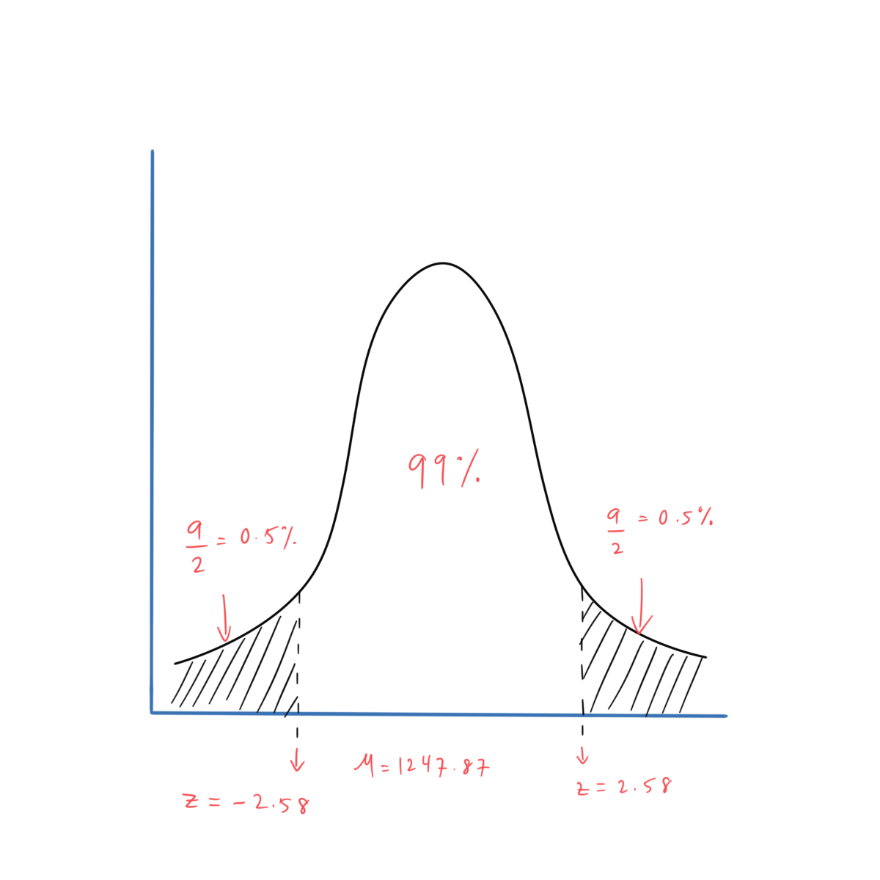
σ = 346.85

= 592.47

μ0 = 1247.87

n = 35

1. **Step 1:**

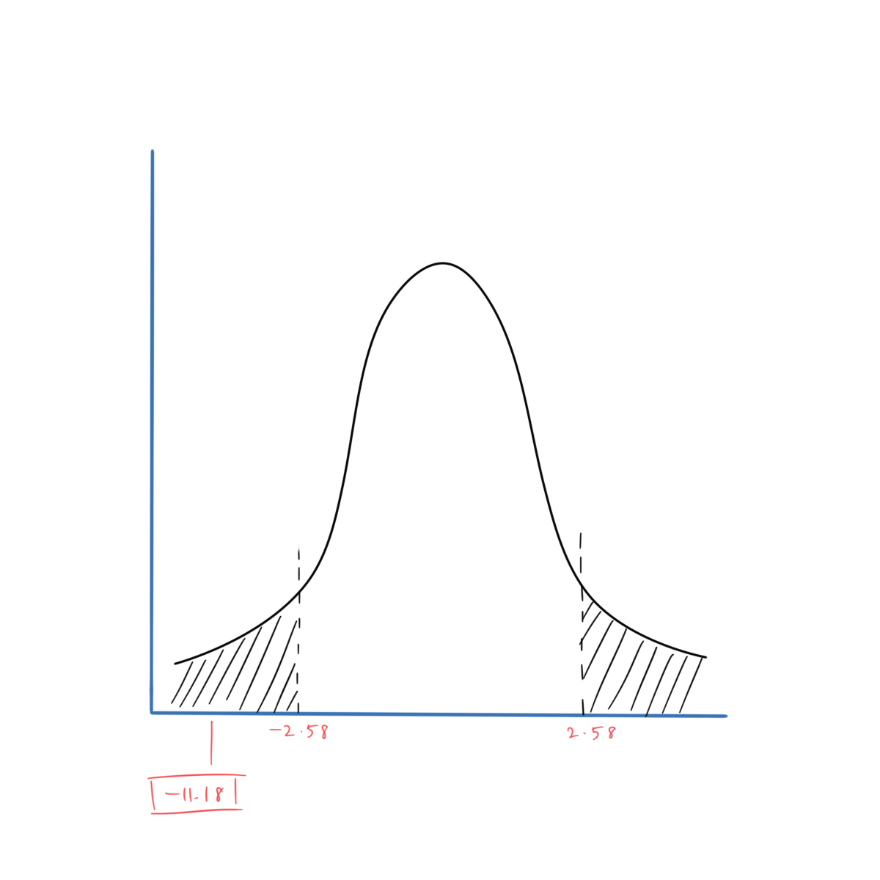
****

**Step 2:**

**Step 3:**

Z test =

=



**Conclusion:**

The test statistics of 1247.87 is not well within the confidence range 99%. Therefore, null hypothesis is rejected.

**PART VI - Conclusion**

The first part of this assignment consists of data collection. The data collection requires all of our group members to collect our data and use the provided excel file as a template to collect our calorie intake for a period of one (1) week during COVID-19 period. By the 2nd of June, every member has submitted their data collections.

For the second part of this assignment, we are required to create a frequency table and calculate relative frequency for the dataset. We are also required to calculate the minimum, maximum, mean and mode values for the collected dataset. And lastly, we are required to calculate standard deviation and variance for the collected dataset. The total frequency acquired is 35. The minimum value is 352, the maximum value is 2456, the mean value is 1247.87, the mode value is 1390 (frequency: 3). The standard deviation is 346.85 and the variance is 120302.63.

The third part of this assignment is the descriptive analysis and displaying data. We have used Python as instructed and have executed the code to find the minimum, maximum, mean, mode, standard deviation and variance values.

The fourth part of this assignment is the regression analysis where we are required to again use Python. The code is executed to find the correlation between calories intake and weight of students, the correlation between physical activities and weight of students and the correlation between and calorie intakes.

The final part of this assignment is the hypotheses testing. Based on all the data collected, it is concluded that the test statistics of 1247.87 is not well within the confidence range 99%. Therefore, null hypothesis is rejected.