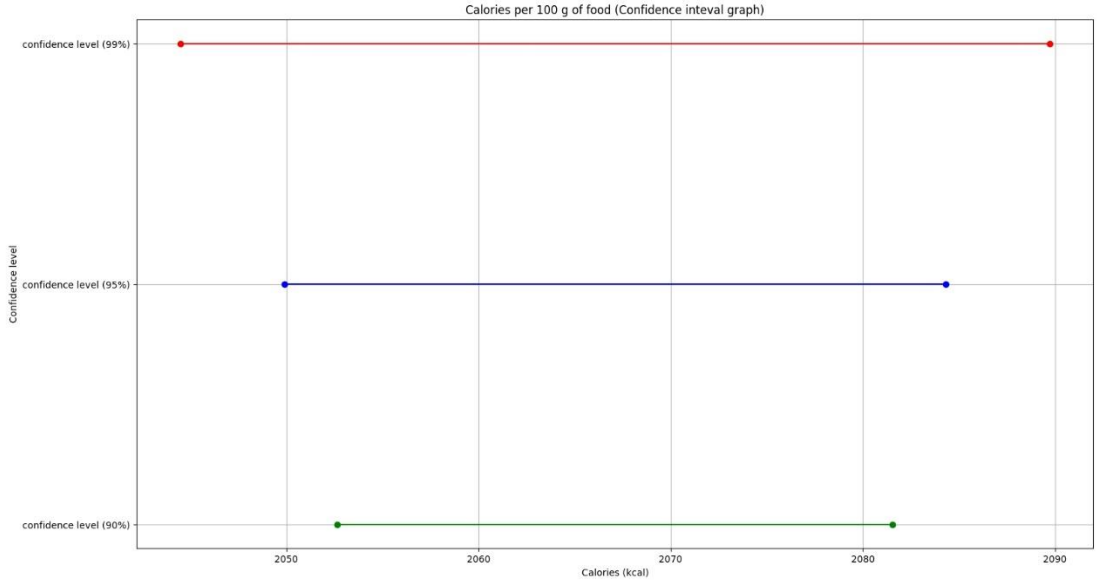
นาย ภัทรพัทธิ์ ชัยอมรเวทย์ รหัสนักศึกษา : **62010684**
คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

Topic : Confidence Interval Graph (CI)

ภาษาที่ใช้ : **python**

Column ที่เลือกใช้ : Calories หน่วย (kcal)



ในแกน x ของกราฟ -> จะบอกถึงจำนวนของ **Calories** ในช่วงระดับความมั่นใจแต่ละช่วง ซึ่งมีหน่วยเป็น **kcal**

ในแกน y ของกราฟ -> จะบอกถึงระดับความมั่นใจที่มาใช้ในการคำนวณ ซึ่งแบ่งเป็น **3** ระดับ ได้แก่ **90% , 95% และ 99%**

[ที่มาของกราฟ] -> ก่อนที่เราจะได้ **Confidence interval graph** มานั้น เราต้องรู้ก่อนว่าการทดลองของเรามีองค์ประกอบอะไรบ้าง เช่น จำนวนประชากร, ค่าเออเร่อ, ค่าเฉลี่ย ฯลฯ ซึ่งระดับความเชื่อมั่นจะคำนวณได้จากสูตรข้างล่างดังนี้

$$\bar{X} - Z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}} < \mu < \bar{X} + Z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}$$

เมื่อ μ เป็นค่าเฉลี่ยของประชากรที่ต้องการประมาณค่า
$\bar{X}$ เป็นค่าเฉลี่ยจากกลุ่มตัวอย่าง
σ เป็นค่าส่วนเบี่ยงเบนมาตรฐานของประชากร (และ σ^2 เป็นค่าความแปรปรวนของประชากร)
s เป็นค่าส่วนเบี่ยงเบนมาตรฐานของกลุ่มตัวอย่าง>
n เป็นขนาดของกลุ่มตัวอย่าง
$Z_{1-\alpha/2}$ เป็นค่าสถิติ Z ที่เปิดได้จากตาราง (บางตารางใช้ค่า $Z_{\alpha/2}$)
$t_{1-\alpha/2}$ เป็นค่าสถิติ t ที่เปิดได้จากตาราง ที่ df=n-1

ในการคำนวณและการหาค่าข้อมูลเชิงสถิตินั้น เราสามารถเขียนโปรแกรมภาษา **Python** เพื่อที่จะนำมาคำนวณหาช่วงระดับความมั่นใจ **(Confidence interval)** ได้ โดยใช้ **lib** ของภาษานี้ ซึ่ง **lib** ที่เรานำมาใช้ได้แก่

1.matplotlib -> ใช้เพื่อวาดกราฟจากไฟล์ข้อมูล และจำแนกได้กราฟได้หลายรูปแบบ

2.pandas -> ใช้เพื่ออ่านข้อมูลที่เป็นตัวเลข **(numeric data)** จากไฟล์ **csv** ได้

3.statistic -> ใช้เพื่อคำนวณหาค่าข้อมูลเชิงสถิติต่างๆ เช่น ค่าเฉลี่ย ส่วนเบี่ยงเบนมาตรฐาน เป็นต้น

4.math -> ใช้เพื่อคำนวณแล้วนำมาใช้กับสูตรระดับความมั่นใจ เช่น หารากที่สอง **(sqrt)**

Code : python

```python
import matplotlib.pyplot as plt
import pandas as pd
import math as M
import statistics
df = pd.read_csv('nutrition.csv')
df = df[['calories']]
y = 'calories'

Calories_mean = df['calories'].mean()
Calories_Std = df['calories'].std()
Calories_max = df['calories'].max()

print('Calories mean is : ' + str('{:.2f}'.format(Calories_mean)))
print('Calories standard deviation of salaries : ' + str('{:.2f}'.format(Calories_Std)))
print('---------------------------------------------------')
print('Formula is (mean - z)*(S.D./sqrt(n)) for lower bound')
print('Formula is (mean + z)*(S.D./sqrt(n)) for upper bound')
lowerBoundlevel90 = (Calories_mean-1.645)*(Calories_Std/M.sqrt(500))
upperBoundlevel90 = (Calories_mean+1.645)*(Calories_Std/M.sqrt(500))
lowerBoundlevel95 = (Calories_mean-1.96)*(Calories_Std/M.sqrt(500))
upperBoundlevel95 = (Calories_mean+1.96)*(Calories_Std/M.sqrt(500))
lowerBoundlevel99 = (Calories_mean-2.576)*(Calories_Std/M.sqrt(500))
upperBoundlevel99 = (Calories_mean+2.576)*(Calories_Std/M.sqrt(500))

print('---------------------------------------------------')
print('The interval of Confidence level at 90% :',str('{:.2f}'.format(lowerBoundlevel90)),'-',str('{:.2f}'.format(upperBoundlevel90)))
print('The interval of Confidence level at 95% :',str('{:.2f}'.format(lowerBoundlevel95)),'-',str('{:.2f}'.format(upperBoundlevel95)))
print('The interval of Confidence level at 99% :',str('{:.2f}'.format(lowerBoundlevel99)),'-',str('{:.2f}'.format(upperBoundlevel99)))
data_dict = {}
data_dict['ConfidenceLevel'] = ['confidence level (90%)','confidence level (95%)','confidence level (99%)']
data_dict['lower'] = [lowerBoundlevel90,lowerBoundlevel95,lowerBoundlevel99]
data_dict['upper'] = [upperBoundlevel90,upperBoundlevel95,upperBoundlevel99]
dataset = pd.DataFrame(data_dict)

plt.plot((lowerBoundlevel90,upperBoundlevel90),(0,0),'ro-',color = 'green')
plt.plot((lowerBoundlevel95,upperBoundlevel95),(1,1),'ro-',color = 'blue')
plt.plot((lowerBoundlevel99,upperBoundlevel99),(2,2),'ro-',color = 'red')
plt.yticks(range(len(dataset)),list(dataset['ConfidenceLevel']))
plt.xlabel('Calories (kcal)')
plt.ylabel('Confidence level')
plt.title('Calories per 100 g of food (Confidence inteval graph)')
plt.xlim(2000,2150)
plt.grid()
plt.show()
```

บรรทัดที่ **1-4 ->** จะเป็นการ **import lib** เพื่อนำมาใช้งานด้านต่างๆ

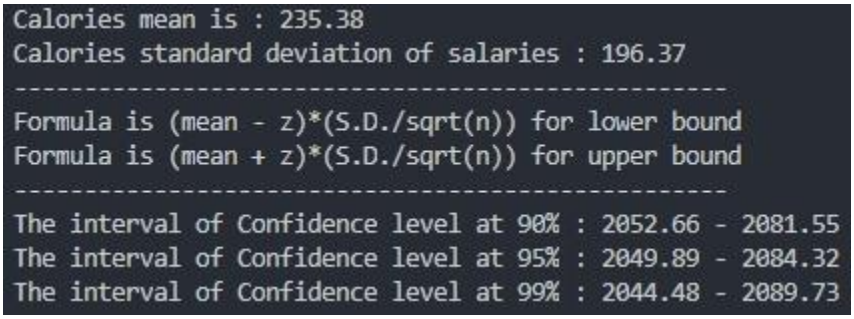
บรรทัดที่ **5-11 ->** จะเป็นการดึงข้อมูลตัวเลขมาจากไฟล์ **csv** ซึ่งข้อมูลจะเป็นประเภทตัวเลข

บรรทัดที่ **13-28 ->** จะเป็นการคำนวณเชิงสถิติ โดยเราใช้ **statistics lib** เพื่อนำมาช่วยคำนวณ **CI** ด้วย

บรรทัด **29-33 ->** จะเป็นการกำหนดขอบล่างและขอบบนให้กับ **CI** แต่ละช่วงว่าถ้าระดับความมั่นใจช่วงนี้ควรมีจะมีขอบเขตเป็นเท่าไหร่ ใช้ค่า **z** เป็นเท่าไหร่ ***ค่า z สามารถเปิดตารางในเว็บดูได้*** โดย **CI 90%** จะใช้ค่า **z = 1.645 / CI 95%** จะใช้ค่า **z = 1.96 / CI 99%** จะใช้ค่า **z = 2.576**

บรรทัดที่ **35-44 ->** จะเป็นการพลอตกราฟจากข้อมูลที่เราคำนวณมาในบรรทัดก่อนหน้า ซึ่งจะใช้ **matplotlib** มาช่วยในการสร้างกราฟ

ผลลัพธ์ของโค้ด



บทวิเคราะห์

จากกราฟ เราคำนวณออกมาแล้วได้ว่า ขอบเขตล่างและขอบเขตบนของแต่ละช่วงแต่ระดับความมั่นใจมีกี่แคลรอลี่ ซึ่งจะสรุปได้ว่าถ้าระดับความมั่นใจอยู่ที่ 90% และเราสุ่มการทดลองไป 500 ครั้ง จะมีการครอบคลุมพารามิเตอร์อยู่ที่ 450 ครั้ง ก็คือ จำนวนแคลรอลี่จากการสุ่มอยู่ในช่วง 2052.66 – 2081.55 และไม่มีการครอบคลุมพารามิเตอร์อยู่ที่ 50 ครั้ง ก็คือ จำนวนแคลรอลี่จากการสุ่มไม่อยู่ในช่วง 2052.66-2081.55 ถ้าระดับความมั่นใจอยู่ที่ 95 % และเราสุ่มการทดลองไป 500 ครั้ง จะมีการคลอบคลุมพารามิเตอร์ จะมีการครอบคลุมพารามิเตอร์อยู่ที่ 475 ครั้ง ก็คือ จำนวนแคลรอลี่จากการสุ่มอยู่ในช่วง 2049.89 – 2084.32 และไม่มการรอบคลุมพารามิเตอร์อยู่ที่ 25 ครั้ง ก็คือ จำนวนแคลรอลี่จากการสุ่มไม่อยู่ในช่วง 2049.89 – 2084.32 ถ้าระดับความมั่นใจอยู่ที่ 99% และเราสุ่มการทดลองไป 500 ครั้ง จะมีการครอบคลุมพารามิเตอร์อยู่ที่ 495 ครั้ง ก็คือ จำนวนแคลรอลี่จากการสุ่มอยู่ในช่วง 2044.48 – 2089.73 และไม่มีการครอบคลุมพารามิเตอร์อยู่ที่ 5 ครั้ง ก็คือ จำนวนแคลรอลี่จากการสุ่มไม่อยู่ในช่วง 2044.48 – 2089.73 ซึ่งค่าที่เราสร้างขึ้นนั้นจะเป็นการทดลองที่เป็นแบบช่วงนั่นเอง