Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Методи оптимізації та планування експерименту

Лабораторна робота №2

**«**ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З

ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ**»**

Виконав:

студент групи ІВ-83

Головенець Р.С.

Перевірив:

ас. Регіда П.Г.

Київ

2020 р.

**Лістинг програми**

**import** random  
**import** math  
**from** numpy **import** linalg  
print(**"Рівняння регресії : \n y=b0+b1\*x1+b2\*x2"**)  
x1min,x1max,x2min,x2max,m,ymax,ymin,=-10,40,15,50,5,240,140  
**while True**:  
 Average,MatrixY,Dispersion=[],[],[]  
 print (**'Матриця Y:'**)  
 **for** i **in** range(0,3):  
 MatrixY.append([random.randint(ymin,ymax)**for** j **in** range(0,m)])  
 Average.append(sum(MatrixY[i])/len(MatrixY[i]))  
 Dispersion.append(sum((k - Average[i]) \*\* 2 **for** k **in** MatrixY[i]) / len(MatrixY[i]))  
 print(MatrixY[i])  
 print(**"Середні значення: \n"**,round(Average[0],3),round(Average[1],3),round(Average[2],3))  
 print(**"Дисперсії: \n"**,round(Dispersion[0],3),round(Dispersion[1],3),round(Dispersion[2],3))  
 deviation = math.sqrt((4 \* m - 4) / (m \* m - 4 \* m))  
 print(**"Основне відхилення : \n"**,round(deviation,3))  
 **if** Dispersion[0] >= Dispersion[1]:  
 F12 = Dispersion[0]/Dispersion[1]  
 **else**:  
 F12 = Dispersion[1] / Dispersion[0]  
 **if** Dispersion[1] >= Dispersion[2]:  
 F23 = Dispersion[1] / Dispersion[2]  
 **else**:  
 F23 = Dispersion[2] / Dispersion[1]  
 **if** Dispersion[0] >= Dispersion[2]:  
 F13 = Dispersion[0] / Dispersion[2]  
 **else**:  
 F13 = Dispersion[2] / Dispersion[0]  
 O12 = (m - 2) / m \* F12  
 O23 = (m - 2) / m \* F23  
 O13 = (m - 2) / m \* F13  
 R12 = abs(O12 - 1) / deviation  
 R23 = abs(O23 - 1) / deviation  
 R13 = abs(O13 - 1) / deviation  
 table099 = {2: 1.73, 6: 2.16, 8: 2.43, 10: 2.62, 12: 2.75, 15: 2.9, 20: 3.08}  
 **if** m **in** (1,2,3):  
 r\_kr=1.73  
 **elif** m **in** (4,5,6):  
 r\_kr=2.16  
 **elif** m **in** (7,8):  
 r\_kr=2.43  
 **elif** m **in** (9,10):  
 r\_kr=2.62  
 **elif** m **in** (11,12,13):  
 r\_kr=2.75  
 **elif** m **in** (14,15,16,17):  
 r\_kr=2.9  
 **else**: r\_kr=3.08  
 print((round(R12,3), **"<"**, round(r\_kr,3)) **if** R12 < r\_kr **else** (round(R12,3),**'>'**,round(r\_kr ,3)))  
 print((round(R23,3),**'<'**,round(r\_kr,3)) **if** R23 < r\_kr **else** (round(R23,3),**'>'**,round(r\_kr,3)))  
 print((round(R13,3),**'<'**,round(r\_kr,3)) **if** R13 < r\_kr **else** (round(R13,3),**'>'**,round(r\_kr,3)))  
 **if** R12 < r\_kr **and** R23 < r\_kr **and** R13 < r\_kr:  
 print(**'Однорідність підтверджується з ймовірністю 0.99'**)  
 MatrixX1X2 = [[-1, -1],  
 [-1, 1],  
 [1, -1]]  
 mx = [sum(i) / len(i) **for** i **in** list(zip(MatrixX1X2[0], MatrixX1X2[1], MatrixX1X2[2]))]  
 my = sum([Average[0], Average[1], Average[2]]) / len([Average[0], Average[1], Average[2]])  
 a1 = sum(i[0] \*\* 2 **for** i **in** MatrixX1X2) / len(MatrixX1X2)  
 a2 = sum(i[0] \* i[1] **for** i **in** MatrixX1X2) / len(MatrixX1X2)  
 a3 = sum(i[1] \*\* 2 **for** i **in** MatrixX1X2) / len(MatrixX1X2)  
 a11 = sum(  
 MatrixX1X2[i][0] \* [Average[0], Average[1], Average[2]][i] **for** i **in** range(len(MatrixX1X2))) / len(  
 MatrixX1X2)  
 a22 = sum(  
 MatrixX1X2[i][1] \* [Average[0], Average[1], Average[2]][i] **for** i **in** range(len(MatrixX1X2))) / len(  
 MatrixX1X2)  
 matrix\_b = [  
 [1, mx[0], mx[1]],  
 [mx[0], a1, a2],  
 [mx[1], a2, a3]  
 ]  
 matrix\_b1 = [  
 [my, mx[0], mx[1]],  
 [a11, a1, a2],  
 [a22, a2, a3]  
 ]  
 matrix\_b2 = [  
 [1, my, mx[1]],  
 [mx[0], a11, a2],  
 [mx[1], a22, a3]  
 ]  
 matrix\_b3 = [  
 [1, mx[0], my],  
 [mx[0], a1, a11],  
 [mx[1], a2, a22]  
 ]  
 b0 = linalg.det(matrix\_b1) / linalg.det(matrix\_b)  
 b1 = linalg.det(matrix\_b2) / linalg.det(matrix\_b)  
 b2 = linalg.det(matrix\_b3) / linalg.det(matrix\_b)  
 print(**'Нормовані рівняння регресії:'**)  
 **for** i **in** range(len(MatrixX1X2)):  
 print(**"y = b0 + b1 \* x1 + b2 \* x2 = b0 + b1 \* "**,MatrixX1X2[i][0],**" + b2 \* "**,MatrixX1X2[i][1],**" = "**,  
 round(b0+b1\*MatrixX1X2[i][0]+b2\*MatrixX1X2[i][1],3))  
  
 x10 = (x1max + x1min) / 2  
 x20 = (x2max + x2min) / 2  
 delta\_x1 = (x1max - x1min) / 2  
 delta\_x2 = (x2max - x2min) / 2  
  
 a0 = b0 - b1 \* (x10 / delta\_x1) - b2 \* (x20 / delta\_x2)  
 a1 = b1 / delta\_x1  
 a2 = b2 / delta\_x2  
  
 print(**'Запишемо натуралізоване рівняння регресії:'**)  
 print(**"y = a0 + a1 \* x1 + a2 \* x2 = a0 + a1 \* "**,x1min,**" + b2 \* "**,x2min,**" = "**,  
 round(a0+a1 \* x1min+a2\*x2min,3))  
 print(**"y = a0 + a1 \* x1 + a2 \* x2 = a0 + a1 \* "**, x1min, **" + b2 \* "**, x2max, **" = "**,  
 round(a0+a1 \* x1min + a2 \* x2max,3))  
 print(**"y = a0 + a1 \* x1 + a2 \* x2 = a0 + a1 \* "**, x1max, **" + b2 \* "**, x2min, **" = "**,  
 round(a0+a1 \* x1max + a2 \* x2min,3))  
 **break  
 else**:  
 print(**"Однорідність не підтвердилась"**)  
 m+=1

**Відповіді на контрольні запитання**

1. Що таке регресійні поліноми і де вони застосовуються?

Апроксимуючі поліноми, за допомогою яких ми можемо описати нашу функцію

.2. Визначення однорідності дисперсії.

Однорідність дисперсії означає, що серед усіх дисперсій нема таких, які б значно перевищували одна одну. 3. Що називається повним факторним експериментом?

3.Експеримент, при реалізації якого визначається значення параметра оптимізації при всіх можливих поєднаннях рівнів варіювання факторів.

**Результат виконання роботи:**

