Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Лабораторна робота №2
З дисципліни «Методи оптимізації та планування»
ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ

ВИКОНАВ:
Студент ІІ курсу ФІОТ
Групи ІО-92
Кушенко Сергій

ПЕРЕВІРИВ:
Регіда П.Г.

Київ 2021 р.

Мета: Провести двофакторний експеримент, перевірити однорідність дисперсії за критерієм Романовського, отримати коефіцієнти рівняння регресії, провести натуралізацію рівняння регресії.

Варіант завдання:

213	-15	30	25	65

Y_{min}=(30-13)*10 = 170
Y_{max}=(20-13)*10 = 70

Лістинг програми:

```
from random import *
from math import *

while True:
    y_MAX = (30 - 13)*10   #170
    y_MIN = (20 - 13)*10   #70

    x1_MIN = -15
    x1_MAX = 30
    x2_MIN = 25
    x2_MAX = 65

    m = 5

    x_arr = [[-1, 1, -1], [-1, -1, 1]]
    y_arr = [[randrange(y_MIN, y_MAX) for i in range(m)]for i in range(3)]

    Y_AVG = [sum(i)/len(i) for i in y_arr]

    sigma = [sum([(y_arr[i][j] - Y_AVG[i]) ** 2 for j in range(m)]) / m for i in
range(3)]
    #ОСНОВНЕ ВІДХИЛЕННЯ
    sig_main = sqrt(2 * (2 * m - 2) / (m * (m - 4)))

    fuv = [max(sigma[int((i + 3) / 2)], sigma[int(i / 2)]) / min(sigma[int((i + 3) /
2)], sigma[int(i / 2)]) for i in range(3)]
    teta = [i*(m-2)/m for i in fuv]
    ruv = [fabs(i-1)/sig_main for i in teta]

    m_X1 = sum(x_arr[0]) / 3
    m_X2 = sum(x_arr[1]) / 3
    M = sum(Y_AVG)/len(Y_AVG)

    a = [(x_arr[0][0]**2 + x_arr[0][1]**2 + x_arr[0][2]**2)/3,
         (x_arr[0][0]*x_arr[1][0] + x_arr[0][1]*x_arr[1][1] +
x_arr[0][2]*x_arr[1][2]) / 3,
          (x_arr[1][0] ** 2 + x_arr[1][1] ** 2 + x_arr[1][2] ** 2) / 3]
    aij = [sum([x_arr[j][i] * Y_AVG[i] for i in range(3)]) / 3 for j in range(2)]

    matr = lambda matrix: matrix[0][0] * matrix[1][1] * matrix[2][2] + \
                               matrix[0][1] * matrix[1][2] * matrix[2][0] + \
                               matrix[1][0] * matrix[2][1] * matrix[0][2] - \
                               matrix[0][2] * matrix[1][1] * matrix[2][0] - \
```

```python
                              matrix[0][1] * matrix[1][0] * matrix[2][2] - \
                              matrix[0][0] * matrix[1][2] * matrix[2][1]

    znamennik = matr([
        [1, m_X1, m_X2],
        [m_X1, a[0], a[1]],
        [m_X2, a[1], a[2]]
    ])

    b0 = matr([
        [M, m_X1, m_X2],
        [aij[0], a[0], a[1]],
        [aij[1], a[1], a[2]]
    ]) / znamennik

    b1 = matr([
        [1, M, m_X2],
        [m_X1, aij[0], a[1]],
        [m_X2, aij[1], a[2]]
    ]) / znamennik

    b2 = matr([
        [1, m_X1, M],
        [m_X1, a[0], aij[0]],
        [m_X2, a[1], aij[1]]
    ]) / znamennik

    delta_X1 = fabs(x1_MAX - x1_MIN) / 2
    delta_X2 = fabs(x2_MAX - x2_MIN) / 2

    x10 = (x1_MAX + x1_MIN) / 2
    x20 = (x2_MAX + x2_MIN) / 2

    a0 = b0 - b1 * x10 / delta_X1 - b2 * x20 / delta_X2
    a1 = b1 / delta_X1
    a2 = b2 / delta_X2

    b11 = b0-b1-b2
    b22 = b0+b1-b2
    b33 = b0-b1+b2

    a11 = a0 + a1*x1_MIN + a2*x2_MIN
    a22 = a0 + a1*x1_MAX + a2*x2_MIN
    a33 = a0 + a1*x1_MIN + a2*x2_MAX
    if ruv[0] < 2 and ruv[1] < 2 and ruv[2] < 2:

        break

print("Y Максимальне = %s, Y Мінімальне = %s" % (y_MAX, y_MIN))
print("Y:", y_arr)
print("Y сер.:", Y_AVG)
print("\nσ²:", [round(i, 2) for i in sigma])
print("\nFuv: ", fuv)
print("Θuv: ", teta)
print("Ruv: ", ruv)
print("\nmx1 = %.2f, mx2 = %.2f" % (m_X1, m_X2))
print("a:   ", [round(i, 1) for i in a])
print("aij: ", [round(i, 2) for i in aij])
print("\nb0 = %.2f, b1 = %.2f, b2 = %.2f" % (b0, b1, b2))
print("Нормоване рівняння регресії виглядає так: y = %.1f + %.1f * x1 + %.1f * x2" %
(b0, b1, b2))
print("\nПеревірка отриманих результатів:")
```

```
print("%.1f - %.1f - %.1f = %.1f" % (b0, b1, b2, b11))
print("%.1f + %.1f - %.1f = %.1f" % (b0, b1, b2, b22))
print("%.1f - %.1f + %.1f = %.1f" % (b0, b1, b2, b33))
print("\na0 = %.2f, a1 = %.2f, a2 = %.2f" % (a0, a1, a2))
print("Натуралізоване рівняння регресії: y = %.1f + %.1f * x1 + %.1f * x2" % (a0, a1,
a2))
print("%.1f + %.1f * %.f + %.1f * %.f = %.1f" % (a0, a1, x1_MIN, a2, x2_MIN, a11))
print("%.1f + %.1f * %.f + %.1f * %.f = %.1f" % (a0, a1, x1_MAX, a2, x2_MIN, a22))
print("%.1f + %.1f * %.f + %.1f * %.f = %.1f" % (a0, a1, x1_MIN, a2, x2_MAX, a33))
```

Результат виконання роботи:

```
C:\Users\HP\Anaconda3\python.exe C:/МОПЕ/Lab2.py
Y Максимальне = 170, Y Мінімальне = 70
Y: [[112, 120, 88, 131, 116], [114, 118, 130, 70, 156], [91, 150, 144, 123, 88]]
Y сер.: [113.4, 117.6, 119.2]

σ²: [201.44, 781.44, 669.36]

Fuv:  [3.8792692613185067, 3.322875297855441, 1.167443528146289]
Ouv:  [2.327561556791104, 1.9937251787132646, 0.7004661168877734]
Ruv:  [0.7421294713250891, 0.5555092626389967, 0.16744453105085375]

mx1 = -0.33, mx2 = -0.33
a:    [1.0, -0.3, 1.0]
aij:  [-38.33, -37.27]

b0 = 118.40, b1 = 2.10, b2 = 2.90
Нормоване рівняння регресії виглядає так: y = 118.4 + 2.1 * x1 + 2.9 * x2

Перевірка отриманих результатів:
118.4 - 2.1 - 2.9 = 113.4
118.4 + 2.1 - 2.9 = 117.6
118.4 - 2.1 + 2.9 = 119.2

a0 = 111.18, a1 = 0.09, a2 = 0.14
Натуралізоване рівняння регресії: y = 111.2 + 0.1 * x1 + 0.1 * x2
111.2 + 0.1 * -15 + 0.1 * 25 = 113.4
111.2 + 0.1 * 30 + 0.1 * 25 = 117.6
111.2 + 0.1 * -15 + 0.1 * 65 = 119.2

Process finished with exit code 0
```

Висновок:

В даній лабораторній роботі було проведено двофакторний експеримент з перевіркою дисперсій на однорідність за критерієм Романовського і отримано коефіцієнти рівняння регресії, крім цього було проведено натуралізацію рівняння регресії. Результати успішно виконаної лабораторної роботи наведені вище у вигляді скріншоту.

Контрольні запитання:

1. **Що таке регресійні поліноми і де вони застосовуються?**
   Регресійні поліноми – це апроксимуючі поліноми, за допомогою яких ми можемо описати функцію. Застосовуються в теорії планування експерименту.

2. **Визначення однорідності дисперсії.**
   Опираючись на вимоги регресивного аналізу достовірне оброблення та використання вихідних даних експериментальних досліджень можливе лише тоді, коли дисперсії вимірювання функцій відгуку в кожній точці експерименту є однаковими. Дана властивість називається *однорідністю дисперсії*.

3. **Що називається повним факторним експериментом?**
   ПФЕ – багатофакторний експеримент в якому використовуються всі можливі комбінації рівні факторів. $N_{ПФЕ} = 2^k$ або 3^k або 5^k.