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 №3

«ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ»

Виконала:
студентка групи ІО-91
Тимошенко Діана
Варіант: 123
Перевірив Регіда П. Г.

Київ
2021 р.

<u>Мета</u>: провести дробовий трьохфакторний експеримент. Скласти матрицю планування, знайти коефіцієнти рівняння регресії, провести 3 статистичні перевірки.

<u>Завдання на лабораторну роботу</u>

1.Скласти матрицю планування для дробового трьохфакторного експерименту. Провести експеримент в усіх точках факторного простору, повторивши N експериментів, де N – кількість експериментів (рядків матриці планування) в усіх точках факторного простору – знайти значення функції відгуку У. Значення функції відгуку знайти у відповідності з варіантом діапазону, зазначеного далі (випадковим чином).

$y_{max} = 200 + x_{cp\ max}$

$y_{min} = 200 + x_{cp\ min}$

$x_{cp\ max} = (x_{1max} + x_{2max} + x_{2max}) / 3$

$x_{cp\ min} = (x_{1min} + x_{2min} + x_{2min}) / 3$

2. Знайти коефіцієнти лінійного рівняння регресії. Записати лінійне рівняння регресії.

3. Провести 3 статистичні перевірки.

4. Написати комп'ютерну програму, яка усе це виконує.

№$_{варианта}$	X_1		X_2		X_3	
	min	max	min	max	min	max
123	-30	0	-25	10	-25	-5

<u>Роздруківка тексту програми</u>:

```
from random import randint
import numpy as np
from math import sqrt

m = 3

cohren_table = {2: 0.7679,
                3: 0.6841,
                4: 0.6287,
                5: 0.5892,
                6: 0.5598}

student_table = {8: 2.306,
                 12: 2.179,
                 16: 2.120,
```

```
                 20: 2.086,
                 24: 2.064}

fisher_table = {8: [5.3, 4.5, 4.1, 3.8],
                12: [4.8, 3.9, 3.5, 3.3],
                16: [4.5, 3.6, 3.2, 3],
                20: [4.4, 3.5, 3.1, 2.9],
                24: [4.3, 3.4, 3, 2.8]}

x1_min, x1_max = -30, 0
x2_min, x2_max = -25, 10
x3_min, x3_max = -25, -5

xcp_min, xcp_max = round((x1_min + x1_min + x1_min) / 3), round((x1_max + x2_max +
x3_max) / 3)
y_min, y_max = 200 + xcp_min, 200 + xcp_max

x_norm = [[1, -1, -1, -1],
          [1, -1, 1, 1],
          [1, 1, -1, 1],
          [1, 1, 1, -1]]

x_natur = [[x1_min, x2_min, x3_min],
           [x1_min, x2_max, x3_max],
           [x1_max, x2_min, x3_max],
           [x1_max, x2_max, x3_min]]
print("Натуралізовані значення факторів", x_natur)

def coef(m):
    y = [[randint(y_min, y_max) for _ in range(m)] for _ in range(4)]
    y_aver = [sum(y[i]) / m for i in range(4)]
    print("Y:", y)
    print("Середні значення функції відгуку за рядками:", y_aver)
    mx1 = sum(x_natur[i][0] for i in range(4)) / 4
    mx2 = sum(x_natur[i][1] for i in range(4)) / 4
    mx3 = sum(x_natur[i][2] for i in range(4)) / 4
    my = sum(y_aver) / 4
    a1 = sum(x_natur[i][0] * y_aver[i] for i in range(4)) / 4
    a2 = sum(x_natur[i][1] * y_aver[i] for i in range(4)) / 4
    a3 = sum(x_natur[i][2] * y_aver[i] for i in range(4)) / 4

    a11 = sum(x_natur[i][0] * x_natur[i][0] for i in range(4)) / 4
    a22 = sum(x_natur[i][1] * x_natur[i][1] for i in range(4)) / 4
    a33 = sum(x_natur[i][2] * x_natur[i][2] for i in range(4)) / 4

    a12 = a21 = sum(x_natur[i][0] * x_natur[i][1] for i in range(4)) / 4
    a13 = a31 = sum(x_natur[i][0] * x_natur[i][2] for i in range(4)) / 4
    a23 = a32 = sum(x_natur[i][1] * x_natur[i][2] for i in range(4)) / 4

    matr_X = [[1, mx1, mx2, mx3],
              [mx1, a11, a21, a31],
              [mx2, a12, a22, a32],
              [mx3, a13, a23, a33]]

    matr_Y = [my, a1, a2, a3]

    b_natur = np.linalg.solve(matr_X, matr_Y)
    print("\nНатуралізоване рівняння регресії: y = {0:.2f} {1:+.2f}*x1 {2:+.2f}*x2
{3:+.2f}*x3".format(*b_natur))

    print("Зробимо перевірку")
    test_y1 = [b_natur[0] + b_natur[1] * x_natur[i][0] + b_natur[2] * x_natur[i][1] +
```

```
b_natur[3] * x_natur[i][2] for i
               in range(4)]

    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_natur,
test_y1[0], x_natur[0][0],

x_natur[0][1], x_natur[0][2]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_natur,
test_y1[1], x_natur[1][0],

x_natur[1][1], x_natur[1][2]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_natur,
test_y1[2], x_natur[2][0],

x_natur[2][1], x_natur[2][2]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_natur,
test_y1[3], x_natur[3][0],

x_natur[3][1], x_natur[3][2]))

    b_norm = [sum(y_aver) / 4,
              sum(y_aver[i] * x_norm[i][1] for i in range(4)) / 4,
              sum(y_aver[i] * x_norm[i][2] for i in range(4)) / 4,
              sum(y_aver[i] * x_norm[i][3] for i in range(4)) / 4]

    print("\nНормоване рівняння регресії: y = {0:.2f} {1:+.2f}*x1 {2:+.2f}*x2
{3:+.2f}*x3".format(*b_norm))

    print("Зробимо перевірку")
    test_y2 = [b_norm[0] + b_norm[1] * x_norm[i][1] + b_norm[2] * x_norm[i][2] +
b_norm[3] * x_norm[i][3] for i in
               range(4)]

    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_norm,
test_y2[0], x_norm[0][1],

x_norm[0][2], x_norm[0][3]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_norm,
test_y2[1], x_norm[1][1],

x_norm[1][2], x_norm[1][3]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_norm,
test_y2[2], x_norm[2][1],

x_norm[2][2], x_norm[2][3]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_norm,
test_y2[3], x_norm[3][1],

x_norm[3][2], x_norm[3][3]))
    cohren(m, y, y_aver, x_norm, b_natur)


# ----------------- Критерій Кохрена -----------------------
def cohren(m, y, y_aver, x_norm, b_natur):
    print("\nКритерій Кохрена")
    dispersion = []
    for i in range(4):
        z = 0
        for j in range(m):
            z += (y[i][j] - y_aver[i]) ** 2
        dispersion.append(z / m)
    print("Дисперсія:", dispersion)
```

```python
    Gp = max(dispersion) / sum(dispersion)
    print("Gp", Gp)
    f1 = m - 1
    Gt = cohren_table[f1]
    if Gp < Gt:
        print("Gp < Gt\n{0:.4f} < {1} => дисперсія однорідна".format(Gp, Gt))
        student(m, dispersion, y_aver, x_norm, b_natur)
    else:
        print("Gp > Gt\n{0:.4f} > {1} => дисперсія неоднорідна => m+=1".format(Gp,
Gt))
        m += 1
        coef(m)


# ---------------------- Критерій Стюдента -------------------------------
def student(m, dispersion, y_aver, x_norm, b_natur):
    print("\nКритерій Стюдента")
    sb = sum(dispersion) / 4
    s_beta = sqrt(sb / (4 * m))
    beta = [sum(y_aver[i] * x_norm[i][j] for i in range(4)) / 4 for j in range(4)]

    t = [abs(beta[i]) / s_beta for i in range(4)]

    f3 = (m - 1) * 4
    t_table = student_table[f3]
    b_impor = []
    for i in range(4):
        if t[i] > t_table:
            b_impor.append(b_natur[i])
        else:
            b_impor.append(0)
    print("Незначні коефіцієнти регресії")
    for i in range(4):
        if b_natur[i] not in b_impor:
            print("b{0} = {1:.2f}".format(i, b_natur[i]))

    y_impor = [b_impor[0] + b_impor[1] * x_natur[i][0] + b_impor[2] * x_natur[i][1] +
b_impor[3] * x_natur[i][2] for i
               in range(4)]

    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_impor,
y_impor[0], x_natur[0][0],

x_natur[0][1], x_natur[0][2]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_impor,
y_impor[1], x_natur[1][0],

x_natur[1][1], x_natur[1][2]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_impor,
y_impor[2], x_natur[2][0],

x_natur[2][1], x_natur[2][2]))
    print("{0:.2f} {1:+.2f}*{5} {2:+.2f}*{6} {3:+.2f}*{7} = {4:.2f}".format(*b_impor,
y_impor[3], x_natur[3][0],

x_natur[3][1], x_natur[3][2]))
    fisher(m, y_aver, b_impor, y_impor, sb)


# ---------------------- Критерій Фішера -------------------------------
def fisher(m, y_aver, b_impor, y_impor, sb):
    print("\nКритерій Фішера")
```

```python
    d = 0
    for i in b_impor:
        if i:
            d += 1
    f3 = (m - 1) * 4
    f4 = 4 - d
    s_ad = sum((y_impor[i] - y_aver[i]) ** 2 for i in range(4)) * m / f4
    Fp = s_ad / sb
    Ft = fisher_table[f3][f4 - 1]
    if Fp < Ft:
      print("Fp < Ft => {0:.2f} < {1}".format(Fp, Ft))
      print("Отримана математична модель при рівні значимості 0.05 адекватна 
експериментальним даним")
    else:
      print("Fp > Ft => {0:.2f} > {1}".format(Fp, Ft))
      print("Рівняння регресії неадекватно оригіналу при рівні значимості 0.05")


if __name__ == '__main__':
      coef(m)
```

Результат виконання програми:

```
"C:\Python 38\python.exe" C:/MOPE_labs/lab3.py
Натуралізовані значення факторів [[-30, -25, -25], [-30, 10, -5], [0, -25, -5], [0, 10, -25]]
Y: [[192, 195, 171], [187, 197, 183], [173, 176, 184], [170, 201, 192]]
Середні значення функції відгуку за рядками: [186.0, 189.0, 177.66666666666666, 187.66666666666666]

Натуралізоване рівняння регресії: y = 181.43 -0.16*x1 +0.19*x2 -0.18*x3
Зробимо перевірку
181.43 -0.16*-30 +0.19*-25 -0.18*-25 = 186.00
181.43 -0.16*-30 +0.19*10 -0.18*-5 = 189.00
181.43 -0.16*0 +0.19*-25 -0.18*-5 = 177.67
181.43 -0.16*0 +0.19*10 -0.18*-25 = 187.67

Нормоване рівняння регресії: y = 185.08 -2.42*x1 +3.25*x2 -1.75*x3
Зробимо перевірку
185.08 -2.42*-1 +3.25*-1 -1.75*-1 = 186.00
185.08 -2.42*-1 +3.25*1 -1.75*1 = 189.00
185.08 -2.42*1 +3.25*-1 -1.75*1 = 177.67
185.08 -2.42*1 +3.25*1 -1.75*-1 = 187.67

Критерій Кохрена
Дисперсія: [114.0, 34.666666666666664, 21.555555555555554, 169.55555555555557]
Gp 0.499018966644866
Gp < Gt
0.4990 < 0.7679 => дисперсія однорідна
```

```
Критерій Стюдента
Незначні коефіцієнти регресії
b1 = -0.16
b2 = 0.19
b3 = -0.18
181.43 +0.00*-30 +0.00*-25 +0.00*-25 = 181.43
181.43 +0.00*-30 +0.00*10 +0.00*-5 = 181.43
181.43 +0.00*0 +0.00*-25 +0.00*-5 = 181.43
181.43 +0.00*0 +0.00*10 +0.00*-25 = 181.43

Критерій Фішера
Fp < Ft => 1.54 < 4.1
Отримана математична модель при рівні значимості 0.05 адекватна експериментальним даним
```

Відповіді на контрольні питання:

1. ***Що називається дробовим факторним експериментом?***

Якщо буде використовуватися лінійна регресія, то можливо зменшити кількість рядків матриці ПФЕ до кількості коефіцієнтів регресійної моделі. Кількість дослідів слід скоротити, використовуючи для планування так звані регулярні дробові репліки від повного факторного експерименту, що містять відповідну кількість дослідів і зберігають основні властивості матриці планування – це означає дробовий факторний експеримент (ДФЕ).

2. ***Для чого потрібно розрахункове значення Кохрена?***

Розрахункове значення Кохрена потрібне для перевірки однорідності дисперсії.

3. ***Для чого перевіряється критерій Стьюдента?***

Критерій Стьюдента перевіряється для перевірки значущості коефіцієнтів регресії.

4. ***Чим визначається критерій Фішера і як його застосовувати?***

За F-критерієм Фішера перевіряється адекватність моделі, він дорівнює відношенню дисперсії адекватності до дисперсії відтворюваності. Знайдене шляхом розрахунку Fр порівнюють з табличним значенням Fт, що визначається при рівні значимості q та кількості ступенів свободи. Якщо Fр < Fт то отримана математична модель з прийнятим рівнем статистичної значимості q адекватна експериментальним даним.