

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

**Лабораторна робота №2**  
з дисципліни  
**«Методи оптимізації та планування експерименту»**

**Виконав:**

Гречаний Є.О.

**Перевірив:**

ас. Регіда П. Г.

**Київ 2020**

## **Тема:** ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ

**Мета:** провести двофакторний експеримент, перевірити однорідність дисперсії за

критерієм Романовського, отримати коефіцієнти рівняння регресії, провести натуруалізацію рівняння регресії.

Завдання:

### **Завдання на лабораторну роботу**

1. Записати лінійне рівняння регресії.
2. Обрати тип двофакторного експерименту і скласти матрицю планування для цього з використанням додаткового нульового фактору ( $x_0=1$ ).
3. Провести експеримент в усіх точках повного факторного простору ( знайти значення функції відгуку  $y$ ). Значення функції відгуку задати випадковим чином у відповідності до варіанту у діапазоні  $y_{\min} \div y_{\max}$

$$y_{\max} = (30 - N_{\text{варіант}}) * 10,$$
$$y_{\min} = (20 - N_{\text{варіант}}) * 10.$$

Варіанти обираються по номеру в списку в журналі викладача.

203	-20	30	-25	10
-----	-----	----	-----	----

### Програмний код

```
import random
import numpy as np

def F(disp):
    return [func(disp[0], disp[1]),
            func(disp[2], disp[0]),
            func(disp[2], disp[1])]

def O(F):
    return [(((m - 2) / m) * F[0]),
            (((m - 2) / m) * F[1]),
            (((m - 2) / m) * F[2])]

def func(a, b):
    if a > b:
        return a / b
    else:
        return b / a

def R(O):
    return [(abs(O[0]) - 1) / 1.7888543819998317),
            ((abs(O[1]) - 1) / 1.7888543819998317),
            ((abs(O[2]) - 1) / 1.7888543819998317)]
```

```

#### base parametrs
m = 5
x1_min, x1_max = -20, 30
x2_min, x2_max = -25, 10
y_max, y_min = 270, 170

y = [[random.randint(y_min, y_max) for j in range(m)] for i in range(3)]
y_ser = [sum(i) / m for i in y]

xn = [[-1, -1], [-1, 1], [1, -1]]
mx = [(xn[0][0] + xn[1][0] + xn[2][0]) / 3,
       ((xn[0][1] + xn[1][1] + xn[2][1]) / 3)]
my = sum(y_ser) / len(y_ser)

vidhilenya = 1.7888543819998317

disp = [*map(np.var, y)]
####

F = F(disp)

O = O(F)

R = R(O)

a_1 = (xn[0][0] ** 2 + xn[1][0] ** 2 + xn[2][0] ** 2) / 3
a_2 = (xn[0][0] * xn[0][1] + xn[1][0] * xn[1][1] + xn[2][0] * xn[2][1]) / 3
a_3 = (xn[0][1] ** 2 + xn[1][1] ** 2 + xn[2][1] ** 2) / 3
a_11 = (xn[0][0] * y_ser[0] + xn[1][0] * y_ser[1] + xn[2][0] * y_ser[2]) / 3
a_22 = (xn[0][1] * y_ser[0] + xn[1][1] * y_ser[1] + xn[2][1] * y_ser[2]) / 3

b_0 = (np.linalg.det([[my, mx[0], mx[1]], [a_11, a_1, a_2], [a_22, a_2, a_3]]) / np.linalg.det(
    [[1, mx[0], mx[1]], [mx[0], a_1, a_2], [mx[1], a_2, a_3]]))
b_1 = (np.linalg.det([[1, my, mx[1]], [mx[0], a_11, a_22], [mx[1], a_22, a_3]]) / np.linalg.det(
    [[1, mx[0], mx[1]], [mx[0], a_1, a_2], [mx[1], a_2, a_3]]))
b_2 = (np.linalg.det([[1, mx[0], my], [mx[0], a_1, a_11], [mx[1], a_2, a_22]]) / np.linalg.det(
    [[1, mx[0], mx[1]], [mx[0], a_1, a_2], [mx[1], a_2, a_3]]))

dif_x1 = abs(x1_max - x1_min) / 2
dif_x2 = abs(x2_max - x2_min) / 2
x10 = (x1_max + x1_min) / 2
x20 = (x2_max + x2_min) / 2
a0 = b_0 - (b_1 * x10 / dif_x1) - (b_2 * x20 / dif_x2)
a1 = b_1 / dif_x1
a2 = b_2 / dif_x2

yn1 = a0 + a1 * x1_min + a2 * x2_min
yn2 = a0 + a1 * x1_max + a2 * x2_min
yn3 = a0 + a1 * x1_min + a2 * x2_max

print("Y")
for i in y:
    for j in i:
        print(j, end=' ')
    print()
print('Середнє значення функції відгуку в рядках: ', *y_ser)
print('Дисперсії по рядках: ', *disp)
print('Основне відхилення: ', vidhilenya)

```

```

print()
for i in range(len(F)):
    print('Fu' + str(i) + ': ', F[i])
print()
for i in range(len(O)):
    print('Ou' + str(i) + ': ', O[i])
print()
for i in range(len(R)):
    print('Ru' + str(i) + ': ', R[i])
print()
for i in R:
    if i > 2:
        print('Недостатньо даних.')
else:
    print('Дисперсія однорідна.')

print()
print(b_0, b_1, b_2)
if [round((b_0 - b_1 - b_2)), round((b_0 + b_1 - b_2)), round((b_0 - b_1 + b_2))] == [*map(round, [yn1, yn2, yn3])]:
    print("Перевірка успішна.")
else:
    print([round((b_0 - b_1 - b_2)), round((b_0 + b_1 - b_2)), round((b_0 - b_1 + b_2))])
    print([*map(round, [yn1, yn2, yn3])])
    print("Перевірка не успішна.")
print('Виконав: студент групи ІО-92 Гречаний Євгеній')
Результати виконання

```

189 236 236 196 195  
 192 184 180 189 243  
 252 213 228 260 236

Середнє значення функції відгуку в рядках: 210.4 197.6 237.8  
 Дисперсії по рядках: 442.64 532.24 281.76  
 Основне відхилення: 1.7888543819998317

Fu0: 1.2024218326405205  
 Fu1: 1.5709823963657013  
 Fu2: 1.8889835320840433

Ou0: 0.7214530995843123  
 Ou1: 0.9425894378194207  
 Ou2: 1.1333901192504259

Ru0: 0.15571245106283557  
 Ru1: 0.03209347991556345  
 Ru2: 0.07456734354268889

Дисперсія однорідна.

217.70000000000002 13.69999999999998 -6.4000000000000036

Перевірка успішна.

Виконав студент групи ІО-92 Гречаний Євгеній