

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 «МНД» на тему  
«: ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ  
ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ»

ВИКОНАВ:  
студент II курсу ФІОТ  
групи ІВ-91  
Прокопчук Д.І.  
Залікова – 9326

ПЕРЕВІРИВ:  
ас. Регіда П. Г.

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

### Завдання:

1. Записати лінійне рівняння регресії.
2. Обрати тип двофакторного експерименту і скласти матрицю планування для нього 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  $y_{\min} \div y_{\max}$

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

Варіант: 128

№ варіанта	X1		X2	
	min	max	min	max
323	-5	15	-25	10

$$y_{\max} = (30 - 323) * 10 = -2930$$

$$y_{\min} = (20 - 323) * 10 = -3030$$

### Лістинг програми:

```
from random import randint
from math import sqrt
from numpy.linalg import det
from prettytable import PrettyTable

variant = 23
x1_min = -5
x1_max = 15
x2_min = -25
x2_max = 10
y_min = (20 - 323) * 10
y_max = (30 - 323) * 10

romanovsky_table = {(2, 3, 4): 1.73, (5, 6, 7): 2.16, (8, 9): 2.43, (10, 11): 2.62,
                    (12, 13): 2.75, (14, 15, 16, 17): 2.9, (18, 19, 20): 3.08}

x1 = [-1, -1, 1]
x2 = [-1, 1, -1]
nx1 = [x1_min if x1[i] == -1 else x1_max for i in range(3)]
nx2 = [x2_min if x2[i] == -1 else x2_max for i in range(3)]

m = 5
y_1 = [randint(y_min, y_max) for _ in range(m)]
y_2 = [randint(y_min, y_max) for _ in range(m)]
y_3 = [randint(y_min, y_max) for _ in range(m)]

y_lists = [y_1, y_2, y_3]
average_y = []
dispersion_y = []
f_uv = []
sigma_uv = []
r_uv = []
deviation = 0
```

```

romanovsky_value = 0

def calculate_dispersion(y_list, avg_y):
    return sum([(i - avg_y) ** 2 for i in y_list]) / m

def romanovsky_criterion():
    global average_y, dispersion_y, f_uv, sigma_uv, r_uv, deviation,
    romanovsky_value

    average_y = [sum(y_1) / m, sum(y_2) / m, sum(y_3) / m]
    dispersion_y = [round(calculate_dispersion(y_lists[i], average_y[i]), 4) for
i in range(3)]
    deviation = sqrt((2 * (2 * m - 2)) / m * (m - 4))
    uv_pairs = [[dispersion_y[0], dispersion_y[1]], [dispersion_y[1],
dispersion_y[2]], [dispersion_y[2], dispersion_y[0]]]
    f_uv = [round(max(uv_pairs[i]) / min(uv_pairs[i]), 4) for i in range(3)]
    sigma_uv = [round(((m - 2) / m * f), 4) for f in f_uv]
    r_uv = [round((abs(sigma - 1) / deviation), 4) for sigma in sigma_uv]

    for key in romanovsky_table.keys():
        if m in key:
            romanovsky_value = romanovsky_table[key]
            break
    return max(r_uv) <= romanovsky_value

while not romanovsky_criterion():
    for i in y_lists:
        i.append((randint(y_min, y_max)))
    m += 1

mx1, mx2, my = sum(x1) / 3, sum(x2) / 3, sum(average_y) / 3
a1 = sum([i ** 2 for i in x1]) / 3
a2 = sum([x1[i] * x2[i] for i in range(3)]) / 3
a3 = sum([i ** 2 for i in x2]) / 3

a11 = sum([x1[i] * average_y[i] for i in range(3)]) / 3
a22 = sum([x2[i] * average_y[i] for i in range(3)]) / 3

determinant = det([[1, mx1, mx2], [mx1, a1, a2], [mx2, a2, a3]])
b0 = det([[my, mx1, mx2], [a11, a1, a2], [a22, a2, a3]]) / determinant
b1 = det([[1, my, mx2], [mx1, a11, a2], [mx2, a22, a3]]) / determinant
b2 = det([[1, mx1, my], [mx1, a1, a11], [mx2, a2, a22]]) / determinant

delta_x1 = abs(x1_max - x1_min) / 2
delta_x2 = abs(x2_max - x2_min) / 2
x_10 = (x1_max + x1_min) / 2
x_20 = (x2_max + x2_min) / 2

a0 = b0 - b1 * (x_10 / delta_x1) - b2 * (x_20 / delta_x2)
a1 = b1 / delta_x1
a2 = b2 / delta_x2

plan_table = PrettyTable()
plan_table.field_names = ['№', 'X1', 'X2', *[f"Y{i}" for i in range(1, m + 1)]]
for i in range(len(y_lists)):
    plan_table.add_row([i + 1, x1[i], x2[i], *y_lists[i]])

romanovsky_matrix = PrettyTable()
romanovsky_matrix.field_names = ['№', 'AVG Y', 'Dispersion Y', 'F_uv', 'σ_uv',
'R_uv']
for i in range(len(y_lists)):
    romanovsky_matrix.add_row([i + 1, average_y[i], dispersion_y[i], f_uv[i],
sigma_uv[i], r_uv[i]])

```

```

ration_checking_table = PrettyTable()
ration_checking_table.field_names = ['№', 'X1', 'X2', 'AVG Y', 'Experimental']
for i in range(len(y_lists)):
    ration_checking_table.add_row([i + 1, x1[i], x2[i], average_y[i], round(b0 +
b1 * x1[i] + b2 * x2[i], 4)])

naturalize_checking_table = PrettyTable()
naturalize_checking_table.field_names = ['№', 'NX1', 'NX2', 'AVG Y',
'Experimental']
for i in range(len(y_lists)):
    naturalize_checking_table.add_row([i + 1, nx1[i], nx2[i], average_y[i],
round(a0 + a1 * nx1[i] + a2 * nx2[i], 4)])

print(plan_table, end="\n\n")
print(romanovsky_matrix, end="\n\n")
print(f"y = {round(b0, 4)} + {round(b1, 4)}*x1 + {round(b2, 4)}*x2")
print(ration_checking_table, end="\n\n")
print(f"y = {round(a0, 4)} + {round(a1, 4)}*nx1 + {round(a2, 4)}*nx2")
print(naturalize_checking_table)
print(f"Відхилення: {deviation}")
print(f"Критерій Романовського: {romanovsky_value}")

```

### Контрольні запитання:

#### 1. Що таке регресійні поліноми і де вони застосовуються?

В теорії планування експерименту найважливішою частиною є оцінка результатів вимірів. При цьому використовують апроксимуючі поліноми, за допомогою яких ми можемо описати нашу функцію. В ТПЕ ці поліноми отримали спеціальну назву - регресійні поліноми, а їх знаходження та аналіз - регресійний аналіз.

#### 2. Визначення однорідності дисперсій.

Обирають так названу «довірчу ймовірність»  $p$  – ймовірність, з якою вимагається підтвердити гіпотезу про однорідність дисперсій. У відповідності до  $p$  і кількості дослідів  $m$  обирають з таблиці критичне значення критерію. Кожне експериментальне значення  $R_{uv}$  критерію Романовського порівнюється з  $R_{кр.}$  (значення критерію Романовського за різних довірчих ймовірностей  $p$ ) і якщо для усіх кожне  $R_{uv} < R_{кр.}$ , то гіпотеза про однорідність дисперсій підтверджується з ймовірністю  $p$ .

#### 3. Що називається повним факторним експериментом?

Для знаходження коефіцієнтів у лінійному рівнянні регресії застосовують повний факторний експеримент (ПФЕ). Якщо в багатфакторному експерименті використані всі можливі комбінації рівнів факторів, то такий експеримент називається повним факторним експериментом

## Результат виконання роботи:

№	X1	X2	Y1	Y2	Y3	Y4	Y5
1	-1	-1	-2944	-3008	-2992	-2999	-2939
2	-1	1	-2979	-2961	-2986	-3008	-3006
3	1	-1	-2971	-2930	-3025	-2965	-2949

№	AVG Y	Dispersion Y	F_uv	σ_uv	R_uv
1	-2976.4	840.24	2.7316	1.639	0.3572
2	-2988.0	307.6	3.2978	1.9787	0.5471
3	-2968.0	1014.4	1.2073	0.7244	0.1541

$$y = -2978.0 + 4.2 \cdot x_1 + -5.8 \cdot x_2$$

№	X1	X2	AVG Y	Experimental
1	-1	-1	-2976.4	-2976.4
2	-1	1	-2988.0	-2988.0
3	1	-1	-2968.0	-2968.0

$$y = -2982.5857 + 0.42 \cdot nx_1 + -0.3314 \cdot nx_2$$

№	NX1	NX2	AVG Y	Experimental
1	-5	-25	-2976.4	-2976.4
2	-5	10	-2988.0	-2988.0
3	15	-25	-2968.0	-2968.0

Відхилення: 1.7888543819998317

Критерій Романовського: 2.16