

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

ЛАБОРАТОРНА РОБОТА № 2
з дисципліни «Методи наукових досліджень»
на тему «ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З
ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ»

ВИКОНАЛА:
студентка 2 курсу
групи ІВ-91
Сайко С. А.
Залікова – 9126

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

Варіанти обираються по номеру в списку в журналі викладача.

№ варіанту	X ₁		X ₂	
	min	max	min	max
124	-20	15	-15	35

Програмний код:

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

var = 24
x1_min = -20
x1_max = 15
x2_min = -15
x2_max = 35
y_min = (20 - 124) * 10
y_max = (30 - 124) * 10

rmnvs_k_table = {(2, 3, 4): 1.72, (5, 6, 7): 2.13, (8, 9): 2.37, (10, 11):
2.54,
                  (12, 13): 2.66, (14, 15, 16, 17): 2.8, (18, 19, 20): 2.96}

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]
avrg_y = []
dsprsn_y = []
f_uv = []
sgm_uv = []
r_uv = []
dvtm = 0
rmnvsk_value = 0

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

def rmnvsk_cr():
    global avrg_y, dsprsn_y, f_uv, sgm_uv, r_uv, dvtm, rmnvsk_value
    avrg_y = [sum(y_1) / m, sum(y_2) / m, sum(y_3) / m]

    dsprsn_y = [round(clclt_disp(y_lists[i], avrg_y[i]), 4) for i in range(3)]

    dvtm = sqrt((2 * (2 * m - 2)) / m * (m - 4))

    uv_pairs = [[dsprsn_y[0], dsprsn_y[1]], [dsprsn_y[1], dsprsn_y[2]],
[dsprsn_y[2], dsprsn_y[0]]]
    f_uv = [round(max(uv_pairs[i]) / min(uv_pairs[i]), 4) for i in range(3)]

    sgm_uv = [round(((m - 2) / m * f), 4) for f in f_uv]

    r_uv = [round((abs(sigma - 1) / dvtm), 4) for sigma in sgm_uv]

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

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

mx1, mx2, my = sum(x1) / 3, sum(x2) / 3, sum(avrg_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] * avrg_y[i] for i in range(3)]) / 3
a22 = sum([x2[i] * avrg_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 = ['№', 'Середній Y', 'Дисперсія Y', 'F_uv',
    'σ_uv', 'R_uv']
for i in range(len(y_lists)):
    romanovsky_matrix.add_row([i + 1, avrg_y[i], dsprsn_y[i], f_uv[i],
    sgm_uv[i], r_uv[i]])

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

naturalize_checking_table = PrettyTable()
naturalize_checking_table.field_names= ['№', 'NX1', 'NX2', 'Середній Y',
    'Експериментальне']
for i in range(len(y_lists)):
    naturalize_checking_table.add_row([i + 1, nx1[i], nx2[i], avrg_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: {dvtn}")
print(f"Критерій Романовського: {rmnvsk_value}")

```

Результат роботи програми:

№	X1	X2	Y1	Y2	Y3	Y4	Y5
1	-1	-1	-1017	-965	-991	-969	-1018
2	-1	1	-955	-1000	-1030	-995	-1035
3	1	-1	-942	-946	-1029	-1036	-1006

№	Середній Y	Дисперсія Y	F_uv	σ_{uv}	R_uv
1	-992.0	512.0	1.6133	0.968	0.0179
2	-1003.0	826.0	1.9653	1.1792	0.1002
3	-991.8	1623.36	3.1706	1.9024	0.5045

$$y = -997.4 + 0.1 \cdot x_1 + -5.5 \cdot x_2$$

№	X1	X2	Середній Y	Експериментальне
1	-1	-1	-992.0	-992.0
2	-1	1	-1003.0	-1003.0
3	1	-1	-991.8	-991.8

$$y = -995.1857 + 0.0057 \cdot nx_1 + -0.22 \cdot nx_2$$

№	NX1	NX2	Середній Y	Експериментальне
1	-20	-15	-992.0	-992.0
2	-20	35	-1003.0	-1003.0
3	15	-15	-991.8	-991.8

Відхилення: 1.7888543819998317

Критерій Романовського: 2.13

Висновок:

Під час виконання даної лабораторної роботи, я провела двофакторний експеримент, перевірила однорідність дисперсії за критерієм Романовського, отримала коефіцієнти рівняння регресії, провела натуралізацію рівняння регресії. Коефіцієнти знайдені правильно, що підтверджують результати, наведені вище. Мета лабораторної роботи досягнута.

Відповіді на контрольні питання:

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.

Однорідність дисперсії означає, що серед усіх дисперсій немає такої, яка б значно перевищувала інші.

3. Що називається повним факторним експериментом?

ПФЕ (Повний факторний експеримент) – називається такий експеримент, при реалізації якого визначається значення параметра оптимізації при всіх можливих поєднаннях рівнів варіювання факторів.