

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 експерименту**» на тему
**«Проведення двофакторного експерименту з використанням
лінійного рівняння регресії»**

Виконала:
студентка II курсу ФІОТ
групи ІО-93
Дяченко Віта
У списку групи №9

Перевірив:
Регіда П. Г.

Київ – 2021

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

Варіант:

Варіант	min	max	min	max
309	-20	15	-35	10

Код програми:

```
import random
import math
n_variant = 309
m = 6
y_max = (30 - n_variant) * 10
y_min = (20 - n_variant) * 10

x1_min, x1_max, x2_min, x2_max = -20, 15, -35, 10

xn = [[-1, -1], [-1, 1], [1, -1]]

# середнє значення y
def average_value_y(c_list):
    average_l_y = []
    for i in range(len(c_list)):
        sum_el = 0
        for j in c_list[i]:
            sum_el += j
        average_l_y.append(sum_el / len(c_list[i]))
    return average_l_y

# дисперсія
def dispersion(c_list):
    d = []
    for i in range(len(c_list)):
        sum_of_y = 0
        for k in c_list[i]:
            sum_of_y += (k - average_value_y(c_list)[i]) ** 2
        d.append(sum_of_y / len(c_list[i]))
    return d

# перевірка для кожної пари комбінацій
def f_uv(u, v):
    if u >= v:
        return u / v
    else:
        return v / u

# визначник
```

```

def determ(x11, x12, x13, x21, x22, x23, x31, x32, x33):
    det = x11 * x22 * x33 + x12 * x23 * x31 + x32 * x21 * x13 - x13 * x22 * x31
    - x32 * x23 * x11 - x12 * x21 * x33
    return det

# генерація y
y = [[random.randint(y_min, y_max) for j in range(6)] for i in range(3)]
average_y = average_value_y(y)

# основне відхилення
sigma_tet = math.sqrt((2 * (2 * m - 2)) / (m * (m - 4)))

fuv = []
teta = []
ruv = []

fuv.append(f_uv(dispersion(y)[0], dispersion(y)[1]))
fuv.append(f_uv(dispersion(y)[2], dispersion(y)[0]))
fuv.append(f_uv(dispersion(y)[2], dispersion(y)[1]))

teta.append((m - 2) / m * fuv[0])
teta.append((m - 2) / m * fuv[1])
teta.append((m - 2) / m * fuv[2])

# експериментальне значення критерію Романовського
ruv.append(abs(teta[0] - 1) / sigma_tet)
ruv.append(abs(teta[1] - 1) / sigma_tet)
ruv.append(abs(teta[2] - 1) / sigma_tet)
# візьмемо значення з таблиці при m=6
r_kr = 2

# перевірка гіпотези про однорідність дисперсій
for i in range(len(ruv)):
    if ruv[i] > r_kr:
        print("Дисперсія - неоднорідна, повторіть експеримент")

# позначення для системи рівнянь з коефіцієнтами для лінійної регресії
mx1 = (xn[0][0] + xn[1][0] + xn[2][0]) / 3
mx2 = (xn[0][1] + xn[1][1] + xn[2][1]) / 3
my = (average_y[0] + average_y[1] + average_y[2]) / 3
# коефіцієнти регресії
a1 = (xn[0][0] ** 2 + xn[1][0] ** 2 + xn[2][0] ** 2) / 3
a2 = (xn[0][0] * xn[0][1] + xn[1][0] * xn[1][1] + xn[2][0] * xn[2][1]) / 3
a3 = (xn[0][1] ** 2 + xn[1][1] ** 2 + xn[2][1] ** 2) / 3
a11 = (xn[0][0] * average_y[0] + xn[1][0] * average_y[1] + xn[2][0] *
average_y[2]) / 3
a22 = (xn[0][1] * average_y[0] + xn[1][1] * average_y[1] + xn[2][1] *
average_y[2]) / 3

# рішення системи з коефіцієнтами регресії методом Крамера
b0 = determ(my, mx1, mx2, a11, a1, a2, a22, a2, a3) / determ(1, mx1, mx2,
mx1, a1, a2, mx2, a2, a3)

```

```

b1 = determ(1, my, mx2, mx1, a11, a2, mx2, a22, a3) / determ(1, mx1, mx2,
mx1, a1, a2, mx2, a2, a3)
b2 = determ(1, mx1, my, mx1, a1, a11, mx2, a2, a22) / determ(1, mx1, mx2,
mx1, a1, a2, mx2, a2, a3)
# лінійна регресія(практичне)
y_pr1 = b0 + b1 * xn[0][0] + b2 * xn[0][1]
y_pr2 = b0 + b1 * xn[1][0] + b2 * xn[1][1]
y_pr3 = b0 + b1 * xn[2][0] + b2 * xn[2][1]

# натуралізація плану
dx1 = abs(x1_max - x1_min) / 2
dx2 = abs(x2_max - x2_min) / 2
x10 = (x1_max + x1_min) / 2
x20 = (x2_max + x2_min) / 2
# обчислення натуралізованих коефіцієнтів
a_0 = b0 - (b1 * x10 / dx1) - (b2 * x20 / dx2)
a_1 = b1 / dx1
a_2 = b2 / dx2
# натуралізоване рівняння регресії
yP1 = a_0 + a_1 * x1_min + a_2 * x2_min
yP2 = a_0 + a_1 * x1_max + a_2 * x2_min
yP3 = a_0 + a_1 * x1_min + a_2 * x2_max

print('Матриця планування для m =', m)
for i in range(3):
    print(y[i])

print('Експериментальні значення критерію Романовського:')
for i in range(3):
    print(ruv[i])

print('Натуралізовані коефіцієнти: \na0 =', round(a_0, 5), 'a1 =', round(a_1,
5), 'a2 =', round(a_2, 5))
print('У практичний ', round(y_pr1, 5), round(y_pr2, 5), round(y_pr3, 5),
'\nУ середній', round(average_y[0], 5), round(average_y[1], 5),
round(average_y[2], 5))
print('У практичний нормалізований', round(yP1, 5), round(yP2, 5), round(yP3,
5))

```

Результат роботи:

```

Матриця планування для m = 6
[-2797, -2803, -2847, -2869, -2889, -2802]
[-2811, -2843, -2810, -2821, -2888, -2805]
[-2867, -2792, -2841, -2887, -2837, -2819]
Експериментальні значення критерію Романовського:
0.028147431973370256
0.0694775717557371
0.18670269097932804
Натуралізовані коефіцієнти:
a0 = -2834.16931 a1 = -0.17143 a2 = 0.10741
У практичний -2834.5 -2829.66667 -2840.5
У середній -2834.5 -2829.66667 -2840.5
У практичний нормалізований -2834.5 -2840.5 -2829.66667

```

Висновок: В даній роботі я провела двофакторний експеримент з перевіркою дисперсій на однорідність за критерієм Романовського, провела натуралізацію рівняння регресії.

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 – багатфакторний експеримент в якому використовуються всі можливі комбінації рівні факторів. $N_{\text{ПФЕ}} = 2^k$ або 3^k або 5^k