

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 № 3

з дисципліни «Методи планування експерименту»

на тему «ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ
З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ.»

ВИКОНАВ:
студент II курсу ФІОТ
групи ІВ-92
Чередник Віталій Юрійович
Номер у списку групи: 26

ПЕРЕВІРИВ:
ас. Регіда П.Г.

Хід роботи

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

Завдання на лабораторну роботу:

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 Y. Значення функції відгуку знайти у відповідності з варіантом діапазону, зазначеного далі (випадковим чином).

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

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

$$\text{де } x_{\text{cp max}} = \frac{x_{1\max} + x_{2\max} + x_{3\max}}{3}, \quad x_{\text{cp min}} = \frac{x_{1\min} + x_{2\min} + x_{3\min}}{3}$$

2. Знайти коефіцієнти лінійного рівняння регресії. Записати лінійне рівняння регресії.
3. Провести 3 статистичні перевірки.
4. Написати комп'ютерну програму, яка усе це виконує.

Варіант завдання

№_варіанта	X ₁		X ₂		X ₃	
	min	max	min	max	min	max
226	10	60	-25	10	10	15

Код програми

```
import random
import numpy as np
from numpy.linalg import solve
```

```
x1min = 10
x1max = 60
x2min = -25
x2max = 10
x3min = 10
x3max = 15
```

```

ymax = 200 + (x1max + x2max + x3max) / 3
ymin = 200 + (x1min + x2min + x3min) / 3

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

y1 = [random.randint(int(ymin), int(ymax)) for i in range(4)]
y2 = [random.randint(int(ymin), int(ymax)) for i in range(4)]
y3 = [random.randint(int(ymin), int(ymax)) for i in range(4)]

Y = [[y1[0], y2[0], y3[0]],
      [y1[1], y2[1], y3[1]],
      [y1[2], y2[2], y3[2]],
      [y1[3], y2[3], y3[3]]]
print("Матриця планування Y (m=3):")
for i in range(4):
    print(Y[i])

x1 = [10, 10, 60, 60]
x2 = [-25, 10, -25, 10]
x3 = [10, 15, 15, 10]

print("----- Перевірка за критерієм Кохрена -----")
Y_average = []
for i in range(len(Y)):
    Y_average.append(np.mean(Y[i], axis=0))
print("Середні значення відгуку за рядками:", Y_average[0], Y_average[1],
      Y_average[2], Y_average[3])

mx1 = np.average(x1)
mx2 = np.average(x2)
mx3 = np.average(x3)
my = np.average(Y_average)

a1 = (x1[0] * Y_average[0] + x1[1] * Y_average[1] + x1[2] * Y_average[2] + x1[3] *
      Y_average[3]) / 4
a2 = (x2[0] * Y_average[0] + x2[1] * Y_average[1] + x2[2] * Y_average[2] + x2[3] *
      Y_average[3]) / 4
a3 = (x3[0] * Y_average[0] + x3[1] * Y_average[1] + x3[2] * Y_average[2] + x3[3] *
      Y_average[3]) / 4

a11 = (x1[0]*x1[0] + x1[1]*x1[1] + x1[2]*x1[2] + x1[3]*x1[3]) / 4
a22 = (x2[0]*x2[0] + x2[1]*x2[1] + x2[2]*x2[2] + x2[3]*x2[3]) / 4
a33 = (x3[0]*x3[0] + x3[1]*x3[1] + x3[2]*x3[2] + x3[3]*x3[3]) / 4

a12 = (x1[0]*x2[0] + x1[1]*x2[1] + x1[2]*x2[2] + x1[3]*x2[3]) / 4
a13 = (x1[0]*x3[0] + x1[1]*x3[1] + x1[2]*x3[2] + x1[3]*x3[3]) / 4
a23 = (x2[0]*x3[0] + x2[1]*x3[1] + x2[2]*x3[2] + x2[3]*x3[3]) / 4

a32, a31, a21 = a23, a13, a12

Deter1 = [[1, mx1, mx2, mx3], [mx1, a11, a12, a13], [mx2, a12, a22, a23], [mx3, a13,
a23, a33]]
Deter2 = [my, a1, a2, a3]
B = [round(i, 4) for i in solve(Deter1, Deter2)]

ypr1 = B[0] + B[1] * x1[0] + B[2] * x2[0] + B[3] * x3[0]
ypr2 = B[0] + B[1] * x1[1] + B[2] * x2[1] + B[3] * x3[1]
ypr3 = B[0] + B[1] * x1[2] + B[2] * x2[2] + B[3] * x3[2]

```

```

ypr4 = B[0] + B[1] * x1[3] + B[2] * x2[3] + B[3] * x3[3]
print("Отримані практичні значення:", ypr1, ypr2, ypr3, ypr4)

dispersions = []
for i in range(len(Y)):
    a = 0
    for k in Y[i]:
        a += (k - np.mean(Y[i], axis=0)) ** 2
    dispersions.append(a / len(Y[i]))
print("Дисперсії:", dispersions)

Gp = max(dispersions) / sum(dispersions)

Gt = 0.7679
if Gp < Gt:
    print("Дисперсія однорідна")
else:
    print("Дисперсія неоднорідна")

print("----- Перевірка значущості коефіцієнтів за критерієм Стюдента -----")
sb = sum(dispersions) / len(dispersions)
sbs = (sb / (4 * 3)) ** 0.5

beta0 = (Y_average[0] * 1 + Y_average[1] * 1 + Y_average[2] * 1 + Y_average[3] * 1) / 4
beta1 = (Y_average[0] * (-1) + Y_average[1] * (-1) + Y_average[2] * 1 + Y_average[3] * 1) / 4
beta2 = (Y_average[0] * (-1) + Y_average[1] * 1 + Y_average[2] * (-1) + Y_average[3] * 1) / 4
beta3 = (Y_average[0] * (-1) + Y_average[1] * 1 + Y_average[2] * 1 + Y_average[3] * (-1)) / 4

T = [abs(beta0)/sbs, abs(beta1)/sbs, abs(beta2)/sbs, abs(beta3)/sbs]
print("Значення t:", T[0], T[1], T[2], T[3])

d = 0
res = [0] * 4
Tf = 2.306
coefs1 = []
coefs2 = []
for i in range(4):
    if T[i] <= Tf:
        coefs2.append(B[i])
        res[i] = 0
    else:
        coefs1.append(B[i])
        res[i] = B[i]
        d += 1
print("Значущі коефіцієнти регресії:", coefs1)
print("Незначущі коефіцієнти регресії:", coefs2)

y_st = []
for i in range(4):
    y_st.append(res[0] + res[1] * x1[i] + res[2] * x2[i] + res[3] * x3[i])
print("Значення з отриманими коефіцієнтами:", y_st)

print("----- Перевірка адекватності за критерієм Фішера -----")
Sad = 3 * sum([(y_st[i] - Y_average[i]) ** 2 for i in range(4)]) / (4 - d)
Fp = Sad / sb
Ft = 4.5

```

```
print("Fp =", Fp)
if (Fp < Ft):
    print("Рівняння регресії адекватне при рівні значимості 0.05")
else:
    print("Рівняння регресії неадекватне при рівні значимості 0.05")
```

Відповіді на контрольні запитання

1. Що називається дробовим факторним експериментом?

Дробовий факторний експеримент – це частина ПФЕ, який мінімізує число дослідів, за рахунок тієї інформації, яка не дуже істотна для побудови лінійної моделі.

2. Для чого потрібно розрахункове значення Кохрена?

Розрахункове значення Кохрена використовується для перевірки однорідності дисперсій.

3. Для чого перевіряється критерій Стьюдента?

Критерій Стьюдента перевіряється для визначення значущості коефіцієнтів рівняння регресії.

4. Чим визначається критерій Фішера і як його застосовувати?

Критерій Фішера використовується при перевірці на однорідність вибірок для порівняння дисперсій.

Результати роботи програми

```
"C:\Program Files\Anaconda\python.exe" C:/Users/Компик/PycharmProjects/MND/lab3.py
Матриця планування Y (m=3):
[200, 206, 211]
[209, 227, 219]
[210, 202, 202]
[210, 209, 226]

----- Перевірка за критерієм Кохрена -----
Середні значення відгуку за рядками: 205.66666666666666 218.33333333333334 204.66666666666666 215.0
Отримані практичне значення: 205.666 218.33350000000002 204.6675 215.00199999999998
Дисперсії: [20.222222222222218, 54.22222222222222, 14.222222222222223, 60.666666666666664]
Дисперсія однорідна

----- Перевірка значущості коефіцієнтів за критерієм Стьюдента -----
Значення t: 119.57851014829426 0.6141922686399955 3.259943579704589 0.33071891388307517
Значущі коефіцієнти регресії: [211.981, 0.3286]
Незначущі коефіцієнти регресії: [-0.0433, 0.2333]
Значення з отриманими коефіцієнтами: [203.766, 215.267, 203.766, 215.267]

----- Перевірка адекватності за критерієм Фішера -----
Fp = 0.5583786160714317
Рівняння регресії адекватне при рівні значимості 0.05

Process finished with exit code 0
```