

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

ВИКОНАЛА:  
студентка II курсу ФІОТ  
групи ІВ-91  
Бузулук М.В.  
Залікова - 9103

ПЕРЕВІРИВ:  
ас. Регіда П. Г.

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

**Варіант завдання:**

№_варіанта	X <sub>1</sub>		X <sub>2</sub>		X <sub>3</sub>	
	min	max	min	max	min	max
103	-20	30	30	80	30	45

**Лістинг програми:**

```
import random as rd
import numpy as np
import prettytable as pt
from numpy.linalg import det

X0 = np.array([1, 1, 1, 1])
X1 = np.array([-1, -1, 1, 1])
X2 = np.array([-1, 1, -1, 1])
X3 = np.array([-1, 1, 1, -1])

X1max = 30
X1min = -20
X2max = 80
X2min = 30
X3max = 45
X3min = 30

m = 4

x1 = np.array([X1min, X1min, X1max, X1max ])
x2 = np.array([X2min, X2max, X2min, X2max])
x3 = np.array([X3min, X3max, X3max, X3min])

Ymax = 200+int((X1max+X2max+X3max)/3)
Ymin = 200+int((X1min+X2min+X3min)/3)

Y1 = np.array([rd.randint(Ymin, Ymax) for i in range(m)])
Y2 = np.array([rd.randint(Ymin, Ymax) for i in range(m)])
Y3 = np.array([rd.randint(Ymin, Ymax) for i in range(m)])
Y4 = np.array([rd.randint(Ymin, Ymax) for i in range(m)])

table = pt.PrettyTable()

table.add_column('X0', X0)
table.add_column('X1', X1)
table.add_column('X2', X2)
table.add_column('X3', X3)
table.add_column('Y1', Y1)
```

```

table.add_column('Y2', Y2)
table.add_column('Y3', Y3)
table.add_column('Y4', Y4)
print(table)

Y1mid = sum(Y1)/len(Y1)
Y2mid = sum(Y2)/len(Y2)
Y3mid = sum(Y3)/len(Y3)
Y4mid = sum(Y4)/len(Y4)

mx1 = sum(x1)/len(x1)
mx2 = sum(x2)/len(x2)
mx3 = sum(x3)/len(x1)

mmidY = (Y1mid + Y2mid + Y3mid + Y4mid)/4

a1 = (x1[0]*Y1mid+x1[1]*Y2mid+x1[2]*Y3mid+x1[3]*Y4mid)/4
a2 = (x2[0]*Y1mid+x2[1]*Y2mid+x2[2]*Y3mid+x2[3]*Y4mid)/4
a3 = (x3[0]*Y1mid+x3[1]*Y2mid+x3[2]*Y3mid+x3[3]*Y4mid)/4

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

main_det = det([[1, mx1, mx2, mx3],
                [mx1, a11, a12, a13],
                [mx2, a12, a22, a23],
                [mx3, a13, a23, a33]])
b0 = det([[mmidY, mx1, mx2, mx3],
          [a1, a11, a12, a13],
          [a2, a12, a22, a23],
          [a3, a13, a23, a33]]) / main_det
b1 = det([[1, mmidY, mx2, mx3],
          [mx1, a1, a12, a13],
          [mx2, a2, a22, a23],
          [mx3, a3, a23, a33]]) / main_det
b2 = det([[1, mx1, mmidY, mx3],
          [mx1, a11, a1, a13],
          [mx2, a12, a2, a23],
          [mx3, a13, a3, a33]]) / main_det
b3 = det([[1, mx1, mx2, mmidY],
          [mx1, a11, a12, a1],
          [mx2, a12, a22, a2],
          [mx3, a13, a23, a3]]) / main_det

```

```

print('Рівняння регресії: y={}+{}*x1+{}*x2+{}*x3'.format(round(b0, 3), round(b1,3)
, round(b2,3), round(b3, 3)))

disp1 = 0
disp2 = 0
disp3 = 0
disp4 = 0
for i in range(len(Y1)):
    disp1 += (Y1[i]- Y1mid)**2
    disp2 += (Y2[i]- Y2mid)**2
    disp3 += (Y3[i]- Y3mid)**2
    disp4 += (Y4[i]- Y4mid)**2

disp1 /= len(Y1)
disp2 /= len(Y2)
disp3 /= len(Y3)
disp4 /= len(Y4)

Gp = max([disp1, disp2, disp3, disp4])/((disp1+disp2+disp3+disp4)
Gt = 0.6841
if Gp<Gt:
    print('Дисперсія однорідна')
else:
    print('Дисперсія неоднорідна')
    exit()

DX1 = abs(X1max - X1min)/2
DX2 = abs(X2max - X2min)/2
X10 = (X1max + X1min)/2
X20 = (X2max + X2min)/2

a0n = b0 - b1*X10/DX1 - b2*X20/DX2
a1n = b1/DX1
a2n = b2/DX2

Sb = (disp1+disp2+disp3+disp4)/4
Sb2 = (Sb/(4*m))**(1/2)
bet0 = 1/4*(Y1mid*X0[0]+Y2mid*X0[1]+Y3mid*X0[2]+Y4mid*X0[3])
bet1 = 1/4*(Y1mid*X1[0]+Y2mid*X1[1]+Y3mid*X1[2]+Y4mid*X1[3])
bet2 = 1/4*(Y1mid*X2[0]+Y2mid*X2[1]+Y3mid*X2[2]+Y4mid*X2[3])
bet3 = 1/4*(Y1mid*X3[0]+Y2mid*X3[1]+Y3mid*X3[2]+Y4mid*X3[3])

t0 = abs(bet0)/Sb2
t1 = abs(bet1)/Sb2
t2 = abs(bet2)/Sb2
t3 = abs(bet3)/Sb2

d = 0
if t0<2.179:

```

```

        d += 1
    else:
        b0 = 0

    if t1<2.179:
        d += 1
    else:
        b1 = 0

    if t2<2.179:
        d += 1
    else:
        b2 = 0

    if t3<2.179:
        d += 1
    else:
        b3 = 0

    print('кількість значимих коефіцієнтів за критерієм Стьюдента:', d)

    y1hat = b0 + b1*X1[0]+b2*X2[0]+b3*X3[0]
    y2hat = b0 + b1*X1[1]+b2*X2[1]+b3*X3[1]
    y3hat = b0 + b1*X1[2]+b2*X2[2]+b3*X3[2]
    y4hat = b0 + b1*X1[3]+b2*X2[3]+b3*X3[3]

    f4 = 4 - d
    f3 = (m-1)*4

    Sad = (m/(4-d))*((y1hat-Y1mid)**2+(y2hat-Y2mid)**2+(y3hat-Y3mid)**2+(y3hat-
    Y3mid)**2)

    Fp = Sad/Sb

    Ft = ([4.8, 3.9, 3.5, 3.3])

    if Fp > Ft[4-d]:
        print('Рівняння регресії НЕадекватно оригіналу при рівні значимості 0.05')
    else:
        print('Рівняння регресії адекватно оригіналу при рівні значимості 0.05')

```

### Результат роботи програми:

X0	X1	X2	X3	Y1	Y2	Y3	Y4
1	-1	-1	-1	247	244	245	249
1	-1	1	1	239	238	225	239
1	1	-1	1	232	249	230	232
1	1	1	-1	218	246	249	249

Рівняння регресії:  $y=224.425+0.012*x1+0.152*x2+0.175*x3$   
Дисперсія однорідна  
кількість значимих коефіцієнтів за критерієм Стюдента: 3  
Рівняння регресії НЕадекватно оригіналу при рівні значимості 0.05

### Контрольні питання:

#### 1. Що називається дробовим факторним експериментом?

ДФЕ – 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?

Критерій Кохрена використовують для порівняння трьох і більше виборок однакового обсягу  $n$

#### 3. Для чого перевіряється критерій Стюдента?

За допомогою критерія Стюдента перевіряється значущість коефіцієнта рівняння регресії. Тобто, якщо виконується нерівність  $t_s < t_{\text{табл}}$ , то приймається нуль-гіпотеза, тобто вважається, що знайдений коефіцієнт  $\beta_s$  є статистично незначущим і його слід виключити з рівняння регресії. Якщо  $t_s > t_{\text{табл}}$  то гіпотеза не підтверджується, тобто  $\beta_s$  – значимий коефіцієнт і він залишається в рівнянні регресії.

#### 4. Чим визначається критерій Фішера і як його застосовувати?

Отримане рівняння регресії необхідно перевірити на адекватність досліджуваному об'єкту. Для цієї мети необхідно оцінити, наскільки відрізняються середні значення  $y$  вихідної величини, отриманої в точках факторного простору, і значення  $y$ , отриманого з рівняння регресії в тих самих точках факторного простору. Для цього використовують дисперсію адекватності. Адекватність моделі перевіряють за F-критерієм Фішера. Якщо  $F_{\text{прак}} < F_{\text{теор}}$ , то отримана математична модель з прийнятим рівнем статистичної значимості  $q$  адекватна експериментальним даним.