

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

Методи оптимізації та планування експерименту

ЛАБОРАТОРНА РОБОТА № 3

**«ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З
ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ»**

ВИКОНАЛА:

студентка II курсу ФІОТ

групи ІО-92

Варіант №211

Карнаухова Анастасія

ПЕРЕВІРИВ:

Доц. Порєв В.М.

ЛАБОРАТОРНА РОБОТА № 3

Тема: ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ

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).

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

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

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

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:

211	10	60	-35	15	10	15
-----	----	----	-----	----	----	----

$$X1_{\min} = 10$$

$$X2_{\min} = -35$$

$$X3_{\min} = 10$$

$$X1_{\max} = 60$$

$$X2_{\max} = 25$$

$$X3_{\max} = 15$$

Код програми:

```
import math
import numpy as np
from scipy.stats import t, f
import random as r
import prettytable as p
from prettytable import PrettyTable

table0 = PrettyTable()
table0.field_names = ("Студент", "Група")
name = "Карнаухова Анастасія"
group = "ІО-92"
table0.add_row([name, group])
print(table0)

m = 3
prob = 0.95
x1_min = 10
x1_max = 60
x2_min = -35
x2_max = 15
x3_min = 10
x3_max = 15
k = 3
x_ranges = [[x1_min, x1_max], [x2_min, x2_max], [x3_min, x3_max]]
x0_norm = [1, 1, 1, 1]
x1_norm = [-1, -1, 1, 1]
x2_norm = [-1, 1, -1, 1]
x3_norm = [-1, 1, 1, -1]
N = len(x1_norm)
xcp_max = (x1_max + x2_max + x3_max) / 3
xcp_min = (x1_min + x2_min + x3_min) / 3
x_norm = [x1_norm, x2_norm, x3_norm]
Y_min = 200 + xcp_min
Y_max = 200 + xcp_max

x_abs = []

for i in range(k):
    temp = []
    for j in x_norm[i]:
        if j == 1:
            temp.append(x_ranges[i][1])
        else:
            temp.append(x_ranges[i][0])
    x_abs.append(temp)
print('Абсолютні значення: ' + str(x_abs))

Y_exp = []
for i in range(N):
    temp = []
    for _ in range(m):
        temp.append(r.randint(math.floor(Y_min), math.floor(Y_max)))
    Y_exp.append(temp)

def y_perevirka_norm(x1, x2, x3):
    return b0 + x1 * b1 + x2 * b2 + x3 * b3
```

```

def y_perevirka_abs(x1, x2, x3):
    return a0 + a1 * x1 + a2 * x2 + a3 * x3

def get_cohren_critical(prob, f1, f2):
    f_crit = f.isf((1 - prob) / f2, f1, (f2 - 1) * f1)
    return f_crit / (f_crit + f2 - 1)

def get_fisher_critical(prob, f3, f4):
    for i in [j * 0.001 for j in range(int(10 / 0.001))]:
        if abs(f.cdf(i, f4, f3) - prob) < 0.0001:
            return i

def get_critical(prob, f3):
    for i in [j * 0.0001 for j in range(int(5 / 0.0001))]:
        if abs(t.cdf(i, f3) - (0.5 + prob / 0.1 * 0.05)) < 0.000005:
            return i

flag = True
while (flag):
    table1 = p.PrettyTable()
    table1.add_column("X0", x0_norm)
    for i in range(k):
        table1.add_column("X{0}".format(i + 1), x_norm[i])
    for i in range(m):
        table1.add_column("Y{0}".format(i + 1), [j[i] for j in Y_exp])
    print("Нормалізована матриця:\n", table1)

    mx_norm_list = [np.mean(i) for i in x_norm]
    y_aver = [np.mean(i) for i in Y_exp]
    my = np.mean(y_aver)
    a1 = np.mean([x_norm[0][i] * y_aver[i] for i in range(N)])
    a2 = np.mean([x_norm[1][i] * y_aver[i] for i in range(N)])
    a3 = np.mean([x_norm[2][i] * y_aver[i] for i in range(N)])
    a11 = np.mean([x_norm[0][i] ** 2 for i in range(N)])
    a22 = np.mean([x_norm[1][i] ** 2 for i in range(N)])
    a33 = np.mean([x_norm[2][i] ** 2 for i in range(N)])
    a12 = np.mean([x_norm[0][i] * x_norm[1][i] for i in range(N)])
    a13 = np.mean([x_norm[0][i] * x_norm[2][i] for i in range(N)])
    a23 = np.mean([x_norm[1][i] * x_norm[2][i] for i in range(N)])
    a21 = a12
    a31 = a13
    a32 = a23

    znam = np.array([[1, mx_norm_list[0], mx_norm_list[1],
mx_norm_list[2]],
                    [mx_norm_list[0], a11, a12, a13],
                    [mx_norm_list[1], a12, a22, a32],
                    [mx_norm_list[2], a13, a23, a33]])

    b0_matr = np.array([[my, mx_norm_list[0], mx_norm_list[1],
mx_norm_list[2]],
                    [a1, a11, a12, a13],
                    [a2, a12, a22, a32],
                    [a3, a13, a23, a33]])

    b1_matr = np.array([[1, my, mx_norm_list[1], mx_norm_list[2]],
                    [mx_norm_list[0], a1, a12, a13],

```

```

        [mx_norm_list[1], a2, a22, a32],
        [mx_norm_list[2], a3, a23, a33]])

b2_matr = np.array([[1, mx_norm_list[0], my, mx_norm_list[2]],
                    [mx_norm_list[0], a11, a1, a13],
                    [mx_norm_list[1], a12, a2, a32],
                    [mx_norm_list[2], a13, a3, a33]])

b3_matr = np.array([[1, mx_norm_list[0], mx_norm_list[1], my],
                    [mx_norm_list[0], a11, a12, a1],
                    [mx_norm_list[1], a12, a22, a2],
                    [mx_norm_list[2], a13, a23, a3]])

znam_value = np.linalg.det(znam)
b0 = np.linalg.det(b0_matr) / znam_value
b1 = np.linalg.det(b1_matr) / znam_value
b2 = np.linalg.det(b2_matr) / znam_value
b3 = np.linalg.det(b3_matr) / znam_value
print("Рівняння регресії для нормованих значень:\ny = {0} + {1}*x1 +
{2}*x2 + {3}*x3".format(b0, b1, b2, b3))
print("Перевірка знайденого рівняння")
print("Р-ня регресії для X11, X21, X31 =",
y_perevirka_norm(x_norm[0][0], x_norm[1][0], x_norm[2][0]))
print("Середнє y1 =", y_aver[0])
print("Р-ня регресії для X12, X22, X32 =",
y_perevirka_norm(x_norm[0][1], x_norm[1][1], x_norm[2][1]))
print("Середнє y2 =", y_aver[1])
print("Р-ня регресії для X13, X23, X33 =",
y_perevirka_norm(x_norm[0][2], x_norm[1][2], x_norm[2][2]))
print("Середнє y3 =", y_aver[2])

delt_x1 = (x1_max - x1_min) / 2
delt_x2 = (x2_max - x2_min) / 2
delt_x3 = (x3_max - x3_min) / 2
x10 = (x1_max + x1_min) / 2
x20 = (x2_max + x2_min) / 2
x30 = (x3_max + x3_min) / 2
a0 = b0 - b1 * (x10 / delt_x1) - b2 * (x20 / delt_x2) - b3 * (x30 /
delt_x3)
a1 = b1 / delt_x1
a2 = b2 / delt_x2
a3 = b3 / delt_x3

print("Рівняння регресії для абсолютних значень:\ny = {0} + {1}*x1 +
{2}*x2 + {3}*x3".format(a0, a1, a2, a3))

print("Перевірка абсолютних значень")
print("Р-ня регресії для X11, X21, X31 =", y_perevirka_abs(x_abs[0][0],
x_abs[1][0], x_abs[2][0]))
print("Середнє y1 =", y_aver[0])
print("Р-ня регресії для X12, X22, X32 =", y_perevirka_abs(x_abs[0][1],
x_abs[1][1], x_abs[2][1]))
print("Середнє y2 =", y_aver[1])
print("Р-ня регресії для X13, X23, X33 =", y_perevirka_abs(x_abs[0][2],
x_abs[1][2], x_abs[2][2]))
print("Середнє y3 =", y_aver[2])
print("Р-ня регресії для X14, X24, X34 =", y_perevirka_abs(x_abs[0][3],
x_abs[1][3], x_abs[2][3]))
print("Середнє y3 =", y_aver[3])

# Кохрен

```

```

y_var = [np.var(Y_exp[i]) for i in range(N)]
flag = False
f1 = m - 1
f2 = N
f3 = f2 * f1
Gp = max(y_var) / sum(y_var)
Gkr = get_cohren_critical(prob, f1, f2)
print('-' * 100)
if (Gkr > Gp):
    print("Gkr = {0} > Gp = {1} ---> Дисперсії однорідні".format(Gkr,
Gp))
    flag = False
else:
    print("Gkr = {0} < Gp = {1} ---> Дисперсії неоднорідні, збільшимо m
і проведемо розрахунки".format(Gkr, Gp))
    Y_exp[0].append(r.randint(math.floor(Y_min), math.floor(Y_max)))
    Y_exp[1].append(r.randint(math.floor(Y_min), math.floor(Y_max)))
    Y_exp[2].append(r.randint(math.floor(Y_min), math.floor(Y_max)))
    Y_exp[3].append(r.randint(math.floor(Y_min), math.floor(Y_max)))
    m += 1

# Стюдент
S2B = sum(y_var) / N
S2b = S2B / (N * m)
Sb = math.sqrt(S2b)
beta0 = sum([y_aver[i] * x0_norm[i] for i in range(N)]) / N
beta1 = sum([y_aver[i] * x1_norm[i] for i in range(N)]) / N
beta2 = sum([y_aver[i] * x2_norm[i] for i in range(N)]) / N
beta3 = sum([y_aver[i] * x3_norm[i] for i in range(N)]) / N
t0 = abs(beta0) / Sb
t1 = abs(beta1) / Sb
t2 = abs(beta2) / Sb
t3 = abs(beta3) / Sb
tkr = get_critical(prob, f3)

d = sum([1 if tkr < i else 0 for i in [t0, t1, t2, t3]])

a0 = a0 if tkr < t0 else 0
a1 = a1 if tkr < t1 else 0
a2 = a2 if tkr < t2 else 0
a3 = a3 if tkr < t3 else 0

y_new = [y_perevirka_abs(x_abs[0][i], x_abs[1][i], x_abs[2][i]) for i in
range(N)]

print("-" * 100)
print("Після перевірки значимості коефіцієнтів: ")
print("Рівняння регресії для абсолютних значень:\ny = {0} + {1}*x1 + {2}*x2
+ {3}*x3".format(a0, a1, a2, a3))
print("Р-ня регресії для X11, X21, X31 =", y_new[0])
print("Р-ня регресії для X12, X22, X32 =", y_new[1])
print("Р-ня регресії для X13, X23, X33 =", y_new[2])
print("Р-ня регресії для X14, X24, X34 =", y_new[3])

```

Результат роботи програми:

```
+-----+-----+
|      Студент      | Група |
+-----+-----+
| Карнаухова Анастасія | IO-92 |
+-----+-----+
Абсолютні значення: [[10, 10, 60, 60], [-35, 15, -35, 15], [10, 15, 15, 10]]
Нормалізована матриця:
+---+---+---+---+---+---+---+
| X0 | X1 | X2 | X3 | Y1 | Y2 | Y3 |
+---+---+---+---+---+---+---+
| 1  | -1 | -1 | -1 | 221 | 209 | 203 |
| 1  | -1 | 1  | 1  | 225 | 206 | 202 |
| 1  | 1  | -1 | 1  | 215 | 210 | 227 |
| 1  | 1  | 1  | -1 | 220 | 225 | 228 |
+---+---+---+---+---+---+---+
Рівняння регресії для нормованих значень:
y = 215.91666666666674 + 4.916666666666671*x1 + 1.75*x2 + -1.75*x3
Перевірка знайденого рівняння
Р-ня регресії для X11, X21, X31 = 211.00000000000006
Середнє y1 = 211.0
Р-ня регресії для X12, X22, X32 = 211.00000000000006
Середнє y2 = 211.0
Р-ня регресії для X13, X23, X33 = 217.33333333333343
Середнє y3 = 217.3333333333334
Рівняння регресії для абсолютних значень:
y = 218.48333333333334 + 0.1966666666666685*x1 + 0.07*x2 + -0.7*x3
Перевірка абсолютних значень
Р-ня регресії для X11, X21, X31 = 211.00000000000009
Середнє y1 = 211.0
Р-ня регресії для X12, X22, X32 = 211.00000000000009
Середнє y2 = 211.0
Р-ня регресії для X13, X23, X33 = 217.33333333333343
Середнє y3 = 217.3333333333334
Рівняння регресії для абсолютних значень:
y = 218.48333333333334 + 0.1966666666666685*x1 + 0.07*x2 + -0.7*x3
Перевірка абсолютних значень
Р-ня регресії для X11, X21, X31 = 211.00000000000009
Середнє y1 = 211.0
Р-ня регресії для X12, X22, X32 = 211.00000000000009
Середнє y2 = 211.0
Р-ня регресії для X13, X23, X33 = 217.33333333333343
Середнє y3 = 217.3333333333334
Р-ня регресії для X14, X24, X34 = 224.33333333333343
Середнє y3 = 224.3333333333334
-----
Gkr = 0.7679205583193613 > Gr = 0.46083418107833163 ---> Дисперсії однорідні
-----
Після перевірки значимості коефіцієнтів:
Рівняння регресії для абсолютних значень:
y = 218.48333333333334 + 0*x1 + 0*x2 + 0*x3
Р-ня регресії для X11, X21, X31 = 218.4833333333334
Р-ня регресії для X12, X22, X32 = 218.4833333333334
Р-ня регресії для X13, X23, X33 = 218.4833333333334
Р-ня регресії для X14, X24, X34 = 218.4833333333334
```

Висновок:

Проробивши лабораторну роботу, було проведено дробовий трьохфакторний експеримент. Складено матрицю планування та знайдено коефіцієнти рівняння регресії. Також проведено 3 статистичні перевірки. У ході виконання лабораторної роботи проблем не виникло. Результати виконання лабораторної висвітлені на роздруківках.

Відповіді на контрольні запитання:

1. Що називається дробовим факторним експериментом?

У деяких випадках немає необхідності проводити повний факторний експеримент (ПФЕ). Тоді слід скоротити кількість дослідів, використовуючи для планування так звані регулярні дробові репліки від повного факторного експерименту, що містять відповідну кількість дослідів і зберігають основні властивості матриці планування – це означає дробовий факторний експеримент (ДФЕ).

2. Для чого потрібно розрахункове значення Кохрена?

Це значення необхідне щоб перевірити однорідність дисперсії, щоб потім було можливо отримати більш точну статистичну оцінку дисперсії функції відгуку.

3. Для чого перевіряється критерій Стьюдента?

Після розрахунків значень коефіцієнтів здійснюється ще одна статистична перевірка – перевірка значущості коефіцієнтів рівняння регресії. Якщо буде встановлено, що якийсь коефіцієнт рівняння регресії незначущий (з обраною ймовірністю), це означає, що відповідний теоретичний коефіцієнт ряду Тейлора дорівнює нулю і необхідно вилучити з рівняння регресії відповідний доданок. Саме тому ця перевірка має і іншу назву – нуль-гіпотеза.

4. Чим визначається критерій Фішера і як його застосовувати?

Критерій Фішера необхідний, щоб перевірити адекватність моделі. Для цієї мети необхідно оцінити, наскільки відрізняються середні значення у вихідної величини, отриманої в точках факторного простору, і значення у, отриманого з рівняння регресії в тих самих точках факторного простору.