

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Методи наукових досліджень
Лабораторна робота № 4
**«Проведення трьохфакторного експерименту при використанні
рівняння регресії з урахуванням ефекту взаємодії.»**

Виконав:
студент групи ІВ-93
Манчук М.В.

Київ
2021 р.

Мета:

Провести повний трьохфакторний експеримент. Знайти рівняння регресії адекватне об'єкту.

Завдання на лабораторну роботу

1. Скласти матрицю планування для повного трьохфакторного експерименту.
2. Провести експеримент, повторивши N раз досліди у всіх точках факторного простору і знайти значення відгуку Y. Знайти значення Y шляхом моделювання випадкових чисел у певному діапазоні відповідно варіанту. Варіанти вибираються за номером в списку в журналі викладача.

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

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

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

3. Знайти коефіцієнти рівняння регресії і записати його.
4. Провести 3 статистичні перевірки – за критеріями Кохрена, Стюдента, Фішера.
5. Зробити висновки по адекватності регресії та значимості окремих коефіцієнтів і записати скореговане рівняння регресії.
6. Написати комп'ютерну програму, яка усе це моделює.

315	-25	75	5	40	15	25
-----	-----	----	---	----	----	----

Код програми:

```
import random
import numpy as np
from numpy.linalg import solve
from scipy.stats import f, t

def lab4(m, n):
    x1_min = -25
    x1_max = 75
    x2_min = 5
    x2_max = 40
    x3_min = 15
    x3_max = 25

    y_max = 200 + (x1_max + x2_max + x3_max) / 3
    y_min = 200 + (x1_min + x2_min + x3_min) / 3

    x_n = [[1, 1, 1, 1, 1, 1, 1, 1],
            [-1, -1, 1, 1, -1, -1, 1, 1],
            [-1, 1, -1, 1, -1, 1, -1, 1],
            [-1, 1, 1, -1, 1, -1, -1, 1]]

    x1x2_norm, x1x3_norm, x2x3_norm, x1x2x3_norm = [0] * n, [0] * n, [0] * n, [0] * n

    for i in range(n):
        x1x2_norm[i] = x_n[1][i] * x_n[2][i]
        x1x3_norm[i] = x_n[1][i] * x_n[3][i]
        x2x3_norm[i] = x_n[2][i] * x_n[3][i]
        x1x2x3_norm[i] = x_n[1][i] * x_n[2][i] * x_n[3][i]

    y_1 = [random.randint(int(y_min), int(y_max)) for _ in range(n)]
    y_2 = [random.randint(int(y_min), int(y_max)) for _ in range(n)]
    y_3 = [random.randint(int(y_min), int(y_max)) for _ in range(n)]

    y_matrix = [[y_1[0], y_2[0], y_3[0]],
                 [y_1[1], y_2[1], y_3[1]],
```

```

        [y_1[2], y_2[2], y_3[2]],
        [y_1[3], y_2[3], y_3[3]],
        [y_1[4], y_2[4], y_3[4]],
        [y_1[5], y_2[5], y_3[5]],
        [y_1[6], y_2[6], y_3[6]],
        [y_1[7], y_2[7], y_3[7]]]

print("Матриця планування y :")
for i in range(n):
    print(y_matrix[i])

x_0 = [1, 1, 1, 1, 1, 1, 1, 1]

x_1 = [-25, -25, 75, 75, -25, -25, 75, 75]

x_2 = [5, 40, 5, 40, 5, 40, 5, 40]

x_3 = [15, 25, 25, 15, 25, 15, 15, 25]

x1x2, x1x3, x2x3, x1x2x3 = [0] * n, [0] * n, [0] * n, [0] * n

for i in range(n):
    x1x2[i] = x_1[i] * x_2[i]
    x1x3[i] = x_1[i] * x_3[i]
    x2x3[i] = x_2[i] * x_3[i]
    x1x2x3[i] = x_1[i] * x_2[i] * x_3[i]

Y_average = []
for i in range(len(y_matrix)):
    Y_average.append(np.mean(y_matrix[i], axis=0))

list_for_b = [x_n[0], x_n[1], x_n[2], x_n[3], x1x2_norm, x1x3_norm, x2x3_norm,
x1x2x3_norm]
list_for_a = list(zip(x_0, x_1, x_2, x_3, x1x2, x1x3, x2x3, x1x2x3))

print("Матриця планування X:")
for i in range(n):
    print(list_for_a[i])

bi = []
for k in range(n):
    S = 0
    for i in range(n):
        S += (list_for_b[k][i] * Y_average[i]) / n
    bi.append(round(S, 3))

ai = [round(i, m) for i in solve(list_for_a, Y_average)]
print("Рівняння регресії: \n" "y = {} + {}*x1 + {}*x2 + {}*x3 + {}*x1x2 + {}*x1x3 +
{}*x2x3 "
      "+ {}*x1x2x3".format(ai[0], ai[1], ai[2], ai[3], ai[4], ai[5], ai[6], ai[7]))
# вивід даних
print("Рівняння регресії для нормованих факторів: \n" "y = {} + {}*x1 + {}*x2 + {}*x3 +
{}*x1x2 + {}*x1x3 +
" {}*x2x3 + {}*x1x2x3\n".format(bi[0], bi[1], bi[2], bi[3], bi[4], bi[5], bi[6],
bi[7]))

print("Перевірка за критерієм Кохрена:")
print("Середні значення відгуку за рядками:",
      "\n", + round(Y_average[0], 3), round(Y_average[0], 3),
      round(Y_average[1], 3), round(Y_average[2], 3), round(Y_average[3], 3),
      round(Y_average[4], 3), round(Y_average[5], 3), round(Y_average[6], 3),
round(Y_average[7], 4))

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

```

```

Gp = max(dispersions) / sum(dispersions)

Gt = 0.5157

if Gp < Gt:
    print("Дисперсія однорідна\n")
else:
    print("Дисперсія неоднорідна\n")

print("Перевірка значущості коефіцієнтів за критерієм Стюдента:")
sb = sum(dispersions) / len(dispersions)
sbs = (sb / (n * m)) ** 0.5

t_list = [abs(bi[i]) / sbs for i in range(0, n)]

d = 0
res = [0] * n
coeff_1 = []
coeff_2 = []
F3 = (m - 1) * n

for i in range(n):
    if t_list[i] < t.ppf(q=0.975, df=F3):
        coeff_2.append(bi[i])
        res[i] = 0
    else:
        coeff_1.append(bi[i])
        res[i] = bi[i]
        d += 1

print("Значущі коефіцієнти регресії:\n", coeff_1)
print("Незначущі коефіцієнти регресії:\n", coeff_2)

y_st = []
for i in range(n):
    y_st.append(res[0] + res[1] * x_n[1][i] + res[2] * x_n[2][i]
                + res[3] * x_n[3][i] + res[4] * x1x2_norm[i]
                + res[5] * x1x3_norm[i] + res[6] * x2x3_norm[i]
                + res[7] * x1x2x3_norm[i])

print("Значення з отриманими коефіцієнтами:\n", y_st)

print("\nПеревірка адекватності за критерієм Фішера:")
Sad = m * sum([(y_st[i] - Y_average[i]) ** 2 for i in range(n)]) / (n - d)
Fp = Sad / sb
F4 = n - d

if Fp < f.ppf(q=0.95, dfn=F4, dfd=F3):
    print("Рівняння регресії адекватне при рівні значимості 0.05")
else:
    print("Рівняння регресії неадекватне при рівні значимості 0.05")

```

lab4(3, 8)

Результат виконання програми:

```
C:\Anaconda3\python.exe C:/PythonProjects/MND/Lab4.py
Матриця планування у :
[242, 243, 219]
[201, 244, 199]
[223, 201, 239]
[221, 222, 220]
[222, 233, 224]
[228, 227, 230]
[207, 229, 227]
[238, 215, 245]
Матриця планування X:
(1, -25, 5, 15, -125, -375, 75, -1875)
(1, -25, 40, 25, -1000, -625, 1000, -25000)
(1, 75, 5, 25, 375, 1875, 125, 9375)
(1, 75, 40, 15, 3000, 1125, 600, 45000)
(1, -25, 5, 25, -125, -625, 125, -3125)
(1, -25, 40, 15, -1000, -375, 600, -15000)
(1, 75, 5, 15, 375, 1125, 75, 5625)
(1, 75, 40, 25, 3000, 1875, 1000, 75000)
Рівняння регресії:
y = 241.071 + -0.234*x1 + -0.089*x2 + -0.61*x3 + -0.005*x1x2 + 0.006*x1x3 + -0.003*x2x3 + 0.0*x1x2x3
Рівняння регресії для нормованих факторів:
y = 224.958 + -1.042*x1 + -0.792*x2 + -1.292*x3 + 3.708*x1x2 + 4.208*x1x3 + 0.792*x2x3 + 2.125*x1x2x3

Перевірка за критерієм Кохрена:
Середні значення відгуку за рядками:
234.667 234.667 214.667 221.0 221.0 226.333 228.333 221.0 232.6667
Дисперсія однорідна
Перевірка за критерієм Кохрена:
Середні значення відгуку за рядками:
234.667 234.667 214.667 221.0 221.0 226.333 228.333 221.0 232.6667
Дисперсія однорідна

Перевірка значущості коефіцієнтів за критерієм Стюдента:
Значущі коефіцієнти регресії:
[224.958]
Незначущі коефіцієнти регресії:
[-1.042, -0.792, -1.292, 3.708, 4.208, 0.792, 2.125]
Значення з отриманими коефіцієнтами:
[224.958, 224.958, 224.958, 224.958, 224.958, 224.958, 224.958, 224.958]

Перевірка адекватності за критерієм Фішера:
Рівняння регресії адекватне при рівні значимості 0.05
```