

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 №4
З дисципліни «Методи наукових досліджень»
За темою:
«Проведення трьохфакторного експерименту
при використанні рівняння регресії з урахуванням ефекту
взаємодії.»

ВИКОНАВ:
Студент II курсу ФІОТ
Групи ІВ-93
Дромашко Артем
Номер у списку – 6
Варіант - 306

ПЕРЕВІРИВ:
асистент
Регіда П.Г.

Київ 2021 р.

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

Завдання:

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, Ст'юдента, 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.

№ варіанта	x_1		x_2		x_3	
	min	max	min	Max	min	max
306	-30	20	-25	10	-30	-15

Програмний код

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

# variant 201
x1min = -30
x1max = 20
x2min = -25
x2max = 10
x3min = -30
x3max = -15

n = 8

# максимальне та мінімальне значення
y_max = 200 + (x1max + x2max + x3max) / 3
y_min = 200 + (x1min + x2min + x3min) / 3

# матриця ПФЕ
xn = [[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] * 8, [0] * 8, [0] * 8, [0] * 8
```

```

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

# заповнення y(генерація)
y1 = [random.randint(int(y_min), int(y_max)) for i in range(8)]
y2 = [random.randint(int(y_min), int(y_max)) for i in range(8)]
y3 = [random.randint(int(y_min), int(y_max)) for i in range(8)]

# матриця планування
y_matrix = [[y1[0], y2[0], y3[0]],
             [y1[1], y2[1], y3[1]],
             [y1[2], y2[2], y3[2]],
             [y1[3], y2[3], y3[3]],
             [y1[4], y2[4], y3[4]],
             [y1[5], y2[5], y3[5]],
             [y1[6], y2[6], y3[6]],
             [y1[7], y2[7], y3[7]]]

# вивід даних за допомогою цикла
print("Матриця планування y : \n")
for i in range(n):
    print(y_matrix[i])

x0 = [1, 1, 1, 1, 1, 1, 1, 1]

# заміна -1 на x1_min, 1 на x1_max
x1 = [10, 10, 50, 50, 10, 10, 50, 50]

# заміна -1 на x2_min, 1 на x2_max
x2 = [20, 60, 20, 60, 20, 60, 20, 60]

# заміна -1 на x3_min, 1 на x3_max
x3 = [20, 25, 25, 20, 25, 20, 20, 25]

# заповнення нулями x1x2, x1x3, x1x2x3
x1x2, x1x3, x2x3, x1x2x3 = [0] * 8, [0] * 8, [0] * 8, [0] * 8
# заповнення x1x2, x1x3, x1x2x3 добутками
for i in range(n):
    x1x2[i] = x1[i] * x2[i]
    x1x3[i] = x1[i] * x3[i]
    x2x3[i] = x2[i] * x3[i]
    x1x2x3[i] = x1[i] * x2[i] * x3[i]

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

# формуємо списки b і a
list_for_b = [xn[0], xn[1], xn[2], xn[3], x1x2_norm, x1x3_norm, x2x3_norm,
               x1x2x3_norm]
list_for_a = list(zip(x0, x1, x2, x3, x1x2, x1x3, x2x3, x1x2x3))

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

# нормовані фактори b_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))
# розрахунок ai(система рівнянь) через функцію solve, вивід рівняння регресії
ai = [round(i, 3) 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".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", +Y_average[0], Y_average[1], Y_average[2], Y_average[3],
    Y_average[4], Y_average[5], Y_average[6], Y_average[7])
# розрахунок дисперсій
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]))
# експериментально
Gp = max(dispersions) / sum(dispersions)
# теоретично
Gt = 0.5157
# перевірка однорідності дисперсій
if Gp < Gt:
    print("Дисперсія однорідна")
    # критерій Стюдента
    print("Перевірка значущості коефіцієнтів за критерієм Стюдента")
    sb = sum(dispersions) / len(dispersions)
    sbs = (sb / (8 * 3)) ** 0.5

    t_list = [abs(bi[i]) / sbs for i in range(0, 8)]

    d = 0
    res = [0] * 8
    coef_1 = []
    coef_2 = []
    # кількість повторень кожної комбінації
    m = 3
    F3 = (m - 1) * n
    # перевірка значущості коефіцієнтів
    for i in range(n):
        if t_list[i] < t.ppf(q=0.975, df=F3):
            coef_2.append(bi[i])
            res[i] = 0
        else:
            coef_1.append(bi[i])
            res[i] = bi[i]
            d += 1

    # вивід
    print("Значущі коефіцієнти регресії:", coef_1)
    print("Незначущі коефіцієнти регресії:", coef_2)

    # значення y з коефіцієнтами регресії

```

```

y_st = []
for i in range(n):
    y_st.append(res[0] + res[1] * xn[1][i] + res[2] * xn[2][i] + res[3] *
xn[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)

# критерій Фішера
print("\nПеревірка адекватності за критерієм Фішера\n")
Sad = m * sum([(y_st[i] - Y_average[i]) ** 2 for i in range(8)]) / (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")
else:
    print("Дисперсія неоднорідна")

```

Результати роботи програми

Матриця планування у :

```
[195, 185, 176]
[200, 199, 175]
[193, 182, 174]
[205, 174, 172]
[180, 186, 199]
[171, 192, 177]
[203, 194, 181]
[187, 200, 179]
```

Матриця планування X:

```
(1, 10, 20, 20, 200, 200, 400, 4000)
(1, 10, 60, 25, 600, 250, 1500, 15000)
(1, 50, 20, 25, 1000, 1250, 500, 25000)
(1, 50, 60, 20, 3000, 1000, 1200, 60000)
(1, 10, 20, 25, 200, 250, 500, 5000)
(1, 10, 60, 20, 600, 200, 1200, 12000)
(1, 50, 20, 20, 1000, 1000, 400, 20000)
(1, 50, 60, 25, 3000, 1250, 1500, 75000)
```

Рівняння регресії:

$y = 174.542 + 1.812 \cdot x_1 + -0.785 \cdot x_2 + 0.558 \cdot x_3 + -0.018 \cdot x_1 x_2 + -0.079 \cdot x_1 x_3 + 0.034 \cdot x_2 x_3 + 0.001 \cdot x_1 x_2 x_3$

Рівняння регресії для нормованих факторів:

$y = 186.625 + 0.375 \cdot x_1 + -0.708 \cdot x_2 + 1.208 \cdot x_3 + -0.125 \cdot x_1 x_2 + -2.375 \cdot x_1 x_3 + 2.875 \cdot x_2 x_3 + 0.792 \cdot x_1 x_2 x_3$

Перевірка за критерієм Кохрена

Середні значення відгуку за рядками:

185.33333333333334 191.33333333333334 183.0 183.66666666666666 188.33333333333334 180.0 192.66666666666666 188.66666666666666

Дисперсія однорідна

Перевірка значущості коефіцієнтів за критерієм Стюдента

Значущі коефіцієнти регресії: [186.625]

Незначущі коефіцієнти регресії: [0.375, -0.708, 1.208, -0.125, -2.375, 2.875, 0.792]

Значення з отриманими коефіцієнтами:

[186.625, 186.625, 186.625, 186.625, 186.625, 186.625, 186.625, 186.625]

Перевірка адекватності за критерієм Фішера

Рівняння регресії адекватне при рівні значимості 0.05

Контрольні запитання:

1. Дробовий факторний експеримент – частина ПФЕ, який мінімізує число дослідів, за рахунок тієї інформації, яка не дуже істотна для побудови моделі
2. Значення Кохрена використовують для перевірки однорідності дисперсії
3. Критерій Стюдента перевіряє значущість коефіцієнтів рівняння
4. Критерій Фішера використовують при перевірці отриманого рівняння регресії досліджуваному об'єкту

Висновок:

В даній лабораторній роботі проведено повний трьохфакторний експеримент. Знайдено рівняння регресії адекватне об'єкту. Кінцевої мети досягнуто.