

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 оптимізації та планування»
Проведення трьохфакторного експерименту
при використанні рівняння регресії з урахуванням ефекту взаємодії

ВИКОНАВ:
Студент II курсу ФІОТ
Групи ІВ-91
Микитенко В.О.
Номер заліковки - 9119

ПЕРЕВІРИВ:
асистент
Регіда П.Г.

Київ 2021 р.

Мета:

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

Варіант завдання:

Варіант	X ₁		X ₂		X ₃	
	min	max	min	max	min	max
118	20	70	25	65	25	35

Лістинг програми:

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

def main(n):
    # варіант 118
    x1min = 20
    x1max = 70
    x2min = 25
    x2max = 65
    x3min = 25
    x3max = 35

    # 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_мін, 1 на x1_макс
x1 = [x1min, x1min, x1max, x1max, x1min, x1min, x1max, x1max]

# заміна -1 на x2_мін, 1 на x2_макс
x2 = [x2min, x2max, x2min, x2max, x2min, x2max, x2min, x2max]

# заміна -1 на x3_мін, 1 на x3_макс
x3 = [x3min, x3max, x3max, x3min, x3max, x3min, x3min, x3max]

# 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")
else:
    print("\nДисперсія неоднорідна, потрібно розпочати експеримент з
початку\n")
    main(n + 1)

# критерій Стьюдента
print("Перевірка значущості коефіцієнтів за критерієм Стьюдента")
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")

main(8)

```

Результат виконання роботи:

"G:\Programming\Python University\Projects\Меторди досліджень\Lab_4\venv\Scripts\python.exe" "G:/I

Матриця планування y :

```

[244, 223, 242]
[249, 224, 240]
[248, 224, 242]
[243, 251, 235]
[231, 239, 253]
[223, 227, 245]
[224, 251, 230]
[248, 241, 241]

```

Матриця планування X:

```

(1, 20, 25, 25, 500, 500, 625, 12500)
(1, 20, 65, 35, 1300, 700, 2275, 45500)
(1, 70, 25, 35, 1750, 2450, 875, 61250)
(1, 70, 65, 25, 4550, 1750, 1625, 113750)
(1, 20, 25, 35, 500, 700, 875, 17500)
(1, 20, 65, 25, 1300, 500, 1625, 32500)
(1, 70, 25, 25, 1750, 1750, 625, 43750)
(1, 70, 65, 35, 4550, 2450, 2275, 159250)

```

Рівняння регресії:

$y = 234.2 + -0.227 \cdot x_1 + -0.427 \cdot x_2 + 0.35 \cdot x_3 + 0.011 \cdot x_1 x_2 + 0.002 \cdot x_1 x_3 + 0.007 \cdot x_2 x_3 + -0.0 \cdot x_1 x_2 x_3$

Рівняння регресії для нормованих факторів:

$y = 238.25 + 1.583 \cdot x_1 + 0.667 \cdot x_2 + 1.75 \cdot x_3 + 2.667 \cdot x_1 x_2 + -0.917 \cdot x_1 x_3 + -0.167 \cdot x_2 x_3 + -0.5 \cdot x_1 x_2 x_3$

Перевірка за критерієм Кохрена

Середні значення відгуку за рядками:

236.33333333333334 237.66666666666666 238.0 243.0 241.0 231.66666666666666 235.0 243.33333333333334

Дисперсія однорідна

Перевірка значущості коефіцієнтів за критерієм Стьюдента

Значущі коефіцієнти регресії: [238.25]

Незначущі коефіцієнти регресії: [1.583, 0.667, 1.75, 2.667, -0.917, -0.167, -0.5]

Значення з отриманими коефіцієнтами:

[238.25, 238.25, 238.25, 238.25, 238.25, 238.25, 238.25, 238.25]

Перевірка адекватності за критерієм Фішера

Рівняння регресії адекватне при рівні значимості 0.05

Висновок:

В даній лабораторній роботі я провів повний трьохфакторний експеримент з трьома статистичними.