

Міністерство освіти і науки України
Національний технічний університет України «КПІ» імені Ігоря Сікорського
Кафедра обчислювальної техніки ФІОТ

ЗВІТ
з лабораторної роботи №2
з навчальної дисципліни «Методи наукових досліджень»

Тема:
ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З
ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ

Виконала:
Студентка 2 курсу кафедри ОТ ФІОТ,
Навчальної групи ІВ-92
Орлова Ю.Д.
Номер у списку групи: 15

Перевірив:
Регіда П.Г.

Київ 2020

Мета: провести двофакторний експеримент, перевірити однорідність дисперсії за критерієм Романовського, отримати коефіцієнти рівняння регресії, провести натуралізацію рівняння регресії.

Завдання на лабораторну роботу:

1. Записати лінійне рівняння регресії.
2. Обрати тип двофакторного експерименту і скласти матрицю планування для нього з використанням додаткового нульового фактору ($x_0=1$).
3. Провести експеримент в усіх точках повного факторного простору (знайти значення функції відгуку y). Значення функції відгуку задати випадковим чином у відповідності до варіанту у діапазоні $u_{\min} \div u_{\max}$.
4. Перевірити однорідності дисперсії за критерієм Романовського
5. Знайти коефіцієнти нормованих рівнянь регресії і виконати перевірку (підставити значення нормованих факторів і коефіцієнтів у рівняння).
6. Провести натуралізацію рівняння регресії й виконати перевірку натуралізованого рівняння.
7. Написати комп'ютерну програму, яка все це виконує.

№ варіанту	x1		x 2	
215	min	max	min	max
	10	50	-20	60

$$y_{\max} = (30 - N_{\text{варіанту}}) * 10 = (30 - 215) * 10 = -1850$$

$$y_{\min} = (20 - N_{\text{варіанту}}) * 10 = (20 - 215) * 10 = -1950$$

Код програми:

```
import random
from math import sqrt

x1_min = 10
x1_max = 50
x2_min = -20
x2_max = 60

variant = 215
y_min = (30 - variant) * 10
y_max = (20 - variant) * 10

m = 5

# planning matrix
x1 = (-1, 1, -1)
```

```

x2 = (-1, -1, 1)
y = [[round(random.uniform(y_min, y_max), 3) for i in range(m)] for j in range(3)]
print('Матриця планування експерименту:\n{0}\n{1}\n{2}\n'.format(y[0], y[1], y[2]))

# the average value of the response functions in the rows
y1_response = round(sum(y[0][i] for i in range(m)) / m, 3)
y2_response = round(sum(y[1][i] for i in range(m)) / m, 3)
y3_response = round(sum(y[2][i] for i in range(m)) / m, 3)

print('Середні значення функції відгуку:\n{0}   {1}   {2}'.format(y1_response,
y2_response, y3_response))

# variance by lines and the main deviation
dispersion1 = round(sum([(y[0][i] - y1_response) ** 2 for i in range(m)]) / m, 3)
dispersion2 = round(sum([(y[1][i] - y1_response) ** 2 for i in range(m)]) / m, 3)
dispersion3 = round(sum([(y[2][i] - y1_response) ** 2 for i in range(m)]) / m, 3)

main_deviation = round(sqrt((2 * (2 * m - 2)) / (m * (m - 4))), 3)

print('\nДисперсія по рядках:\n{0}   {1}   {2}'.format(dispersion1, dispersion1,
dispersion1))
print('\nОсновне відхилення: {0}\n'.format(main_deviation))

# check the variance for homogeneity
f_uv1 = dispersion1 / dispersion2
f_uv2 = dispersion3 / dispersion1
f_uv3 = dispersion3 / dispersion2

theta_uv1 = ((m - 2) / m) * f_uv1
theta_uv2 = ((m - 2) / m) * f_uv2
theta_uv3 = ((m - 2) / m) * f_uv3

r_uv1 = abs(theta_uv1 - 1) / main_deviation
r_uv2 = abs(theta_uv2 - 1) / main_deviation
r_uv3 = abs(theta_uv3 - 1) / main_deviation

r_criterion = 2

if r_uv1 > r_criterion or r_uv2 > r_criterion or r_uv3 > r_criterion:
    print('Дисперсія неоднорідна. Необхідно збільшити значення m')

# calculation of normalized coefficients of the regression equation
mx1 = (x1[0] + x1[1] + x1[2]) / 3
mx2 = (x2[0] + x2[1] + x2[2]) / 3
my = (y1_response + y2_response + y3_response) / 3

a1 = (x1[0] ** 2 + x1[1] ** 2 + x1[2] ** 2) / 3
a2 = (x1[0] * x2[0] + x1[1] * x2[1] + x1[2] * x2[2]) / 3
a3 = (x2[0] ** 2 + x2[1] ** 2 + x2[2] ** 2) / 3

a11 = (x1[0] * y1_response + x1[1] * y2_response + x1[2] * y3_response) / 3
a22 = (x2[0] * y1_response + x2[1] * y2_response + x2[2] * y3_response) / 3

def calculation_of_the_determinant(s11, s12, s13, s21, s22, s23, s31, s32, s33):
    return s11 * s22 * s33 + s12 * s23 * s31 + s13 * s21 * s32 - s13 * s22 * s31 -
s12 * s21 * s33 - s11 * s23 * s32

b0 = round(calculation_of_the_determinant(my, mx1, mx2, a11, a1, a2, a22, a2, a3) /
calculation_of_the_determinant(1, mx1, mx2, mx1, a1, a2, mx2, a2, a3), 3)
b1 = round(calculation_of_the_determinant(1, my, mx2, mx1, a11, a2, mx2, a22, a3) /

```

```

        calculation_of_the_determinant(1, mx1, mx2, mx1, a1, a2, mx2, a2, a3), 3)
b2 = round(calculation_of_the_determinant(1, mx1, my, mx1, a1, a11, mx2, a2, a22) /
        calculation_of_the_determinant(1, mx1, mx2, mx1, a1, a2, mx2, a2, a3), 3)

print('Нормоване рівняння регресії:\ny = {0} + {1} * x1 + {2} * x2\n'.format(b0, b1,
b2))

# naturalization of coefficients
delta_x1 = abs(x1_max - x1_min) / 2
delta_x2 = abs(x2_max - x2_min) / 2
x10 = (x1_max + x1_min) / 2
x20 = (x2_max + x2_min) / 2

a_0 = round(b0 - b1 * x10 / delta_x1 - b2 * x20 / delta_x2, 3)
a_1 = round(b1 / delta_x1, 3)
a_2 = round(b2 / delta_x2, 3)

print('Натуралізоване рівняння регресії:\ny = {0} + {1} * x1 + {2} *
x2\n'.format(a_0, a_1, a_2))
print('Перевірка:\n'
      'a0 + a1 * x1_min + a2 * x2_min = {0}\n'
      'b0 + b1 * x1 + b2 * x2 = {1}\n'
      '\na0 + a1 * x1_max + a2 * x2_min = {2}\n'
      'b0 + b1 * x1 + b2 * x2 = {3}\n'
      'a0 + a1 * x1_min + a2 * x2_max = {4}\n'
      'b0 + b1 * x1 + b2 * x2 = {5}'.format(round(a_0 + a_1 * x1_min + a_2 * x2_min,
3),
                                             round(b0 + b1 * x1[0] + b2 * x2[0], 3),
                                             round(a_0 + a_1 * x1_max + a_2 * x2_min,
3),
                                             round(b0 + b1 * x1[1] + b2 * x2[1], 3),
                                             round(a_0 + a_1 * x1_min + a_2 * x2_max,
3),
                                             round(b0 + b1 * x1[2] + b2 * x2[2], 3)))

```

Результат виконання програми:

Матриця планування експерименту:

```
[-1902.092, -1928.669, -1886.216, -1900.826, -1909.177]  
[-1870.566, -1900.959, -1896.132, -1851.945, -1908.55]  
[-1921.932, -1851.777, -1908.234, -1885.702, -1906.782]
```

Середні значення функції відгуку:

```
-1905.396   -1885.63   -1894.885
```

Дисперсія по рядках:

```
191.12   191.12   191.12
```

Основне відхилення: 1.789

Нормоване рівняння регресії:

$$y = -1890.257 + 9.883 * x_1 + 5.256 * x_2$$

Натуралізоване рівняння регресії:

$$y = -1907.709 + 0.494 * x_1 + 0.131 * x_2$$

Перевірка:

$$a_0 + a_1 * x_{1_min} + a_2 * x_{2_min} = -1905.389$$
$$b_0 + b_1 * x_1 + b_2 * x_2 = -1905.396$$
$$a_0 + a_1 * x_{1_max} + a_2 * x_{2_min} = -1885.629$$
$$b_0 + b_1 * x_1 + b_2 * x_2 = -1885.63$$
$$a_0 + a_1 * x_{1_min} + a_2 * x_{2_max} = -1894.909$$
$$b_0 + b_1 * x_1 + b_2 * x_2 = -1894.884$$

Висновки: під час виконання програми ми провели двофакторний експеримент. Було перевірено однорідність дисперсії за критерієм Романовського, отримано коефіцієнти рівняння регресії та проведено натуралізацію рівняння регресії. Було написано програму з використанням можливостей алгоритмічної мови високого рівня Python, яка це все виконує. Результати роботи програми підтвердили правильність її роботи.

Контрольні запитання

- 1) Що таке регресійні поліноми і де вони застосовуються?

Регресійні поліноми – це апроксимуючі поліноми, за допомогою яких ми можемо описати нашу функцію.

2) Визначення однорідності дисперсії.

Згідно з вимогами регресивного аналізу коректна обробка та використання результатів експериментальних досліджень можливі лише в тому разі, коли дисперсії вимірювання функції відгуку в кожній точці експерименту однакові. Така властивість називається однорідністю дисперсій.

3) Що називається повним факторним експериментом?

Повним факторним експериментом (ПФЕ) називається такий експеримент, при реалізації якого визначається значення параметра оптимізації при всіх можливих поєднаннях рівнів варіювання факторів.