

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

Методи наукових досліджень  
Лабораторна робота №2  
«Проведення двофакторного експерименту з використанням лінійного  
рівняння регресії»

Виконав:  
студент групи ІВ-92  
Сударєв Артем Анатолійович  
Номер у списку групи – 23

Перевірив:  
Регіда П. Г.

Київ 2020 р.

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

Завдання:

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  $y_{\min} \div y_{\max}$

$$y_{\max} = (30 - N_{\text{варіанту}}) * 10,$$
$$y_{\min} = (20 - N_{\text{варіанту}}) * 10.$$

Варіант:

223		-30		0		-15		35
-----	--	-----	--	---	--	-----	--	----

Лістинг програми:

```
import math
import random
import numpy

def naturalization(x_1, x_2):
    return a_0 + a_1 * x_1 + a_2 * x_2

def uniform_dispersion():
    p = 0
    minimum = min(romanovsky_table, key=lambda x: abs(x - m))
    for ruv in (ruv1, ruv2, ruv3):
        if ruv > romanovsky_table[minimum][0]:
            return False
        for rkr in range(len(romanovsky_table[minimum])):
            if ruv < romanovsky_table[minimum][rkr]:
                p = rkr
    return p_list[p]

# Блок даних, заданих за варіантом 223

variantNumber = 23
m = 5
x1 = [-1, 1, -1]
x2 = [-1, -1, 1]
```

```

y_min = (20 - variantNumber) * 10
y_max = (30 - variantNumber) * 10

x1_min, x1_min_normalized = -30, -1
x1_max, x1_max_normalized = 0, 1
x2_min, x2_min_normalized = -15, -1
x2_max, x2_max_normalized = 35, 1

p_list = (0.99, 0.98, 0.95, 0.90)
romanovsky_table = {2: (1.73, 1.72, 1.71, 1.69),
                      6: (2.16, 2.13, 2.10, 2.00),
                      8: (2.43, 4.37, 2.27, 2.17),
                      10: (2.62, 2.54, 2.41, 2.29),
                      12: (2.75, 2.66, 2.52, 2.39),
                      15: (2.9, 2.8, 2.64, 2.49),
                      20: (3.08, 2.96, 2.78, 2.62)}

y_matrix = [[random.randint(y_min, y_max) for _ in range(m)] for _ in
range(3)] # достатньо провести 3 експеримента

y_average_value = [sum(y_matrix[i][j] for j in range(m)) / m for i in
range(3)] # середнє значення функції відгуку

sigma = [sum([(element - y_average_value[i]) ** 2 for element in
y_matrix[i]]) / m for i in range(3)] # Пошук дисперсій

sigma_theta = math.sqrt((2 * (2 * m - 2)) / (m * (m - 4))) # основне
відхилення

fuv1 = sigma[0] / sigma[1]
fuv2 = sigma[2] / sigma[0]
fuv3 = sigma[2] / sigma[1]

theta_uv1 = ((m - 2) / m) * fuv1
theta_uv2 = ((m - 2) / m) * fuv2
theta_uv3 = ((m - 2) / m) * fuv3

ruv1 = abs(theta_uv1 - 1) / sigma_theta
ruv2 = abs(theta_uv2 - 1) / sigma_theta
ruv3 = abs(theta_uv3 - 1) / sigma_theta

# Розрахунок нормованих коефіцієнтів
mx1 = sum(x1) / 3
mx2 = sum(x2) / 3
m_y = sum(y_average_value) / 3
a1 = sum([element ** 2 for element in x1]) / 3
a2 = sum([x1[i] * x2[i] for i in range(3)]) / 3
a3 = sum([element ** 2 for element in x2]) / 3
a11 = sum([x1[i] * y_average_value[i] for i in range(3)]) / 3
a22 = sum([x2[i] * y_average_value[i] for i in range(3)]) / 3

```

```

denominator_determinant = numpy.linalg.det([[1, mx1, mx2],
                                             [mx1, a1, a2],
                                             [mx2, a2, a3]])

b0 = numpy.linalg.det([[m_y, mx1, mx2],
                       [a11, a1, a2],
                       [a22, a2, a3]]) / denominator_determinant
b1 = numpy.linalg.det([[1, m_y, mx2],
                       [mx1, a11, a2],
                       [mx2, a22, a3]]) / denominator_determinant
b2 = numpy.linalg.det([[1, mx1, m_y],
                       [mx1, a1, a11],
                       [mx2, a2, a22]]) / denominator_determinant

# Натуралізація коефіцієнтів
delta_x1 = math.fabs(x1_max - x1_min) / 2
delta_x2 = math.fabs(x2_max - x2_min) / 2
x10 = (x1_max + x1_min) / 2
x20 = (x2_max + x2_min) / 2
a_0 = b0 - b1 * x10 / delta_x1 - b2 * x20 / delta_x2
a_1 = b1 / delta_x1
a_2 = b2 / delta_x2

equation_coefficients = [round(naturalization(x1_min, x2_min), 2),
                          round(naturalization(x1_max, x2_min), 2),
                          round(naturalization(x1_min, x2_max), 2)]

# Вивід результуючих даних
for i in range(3):
    print(f"y{i + 1} = {y_matrix[i]}; середнє значення = {y_average_value[i]}")
print(f"σ²(y1) = {sigma[0]}")
print(f"σ²(y2) = {sigma[1]}")
print(f"σ²(y3) = {sigma[2]}")
print(f"σ(θ) = {sigma_theta}")
print(f"Fuv1 = {fuv1}")
print(f"Fuv2 = {fuv2}")
print(f"Fuv3 = {fuv3}")
print(f"θuv1 = {theta_uv1}")
print(f"θuv2 = {theta_uv2}")
print(f"θuv3 = {theta_uv3}")
print(f"Ruv1 = {ruv1}")
print(f"Ruv2 = {ruv2}")
print(f"Ruv3 = {ruv3}")
print(f"Однорідна дисперсія = {uniform_dispersion()}")
print(f"mx1 = {mx1}")
print(f"mx2 = {mx2}")
print(f"my = {m_y}")
print(f"a1 = {a1}")
print(f"a2 = {a2}")
print(f"a3 = {a3}")

```

```
print(f"a11 = {a11}")
print(f"a22 = {a22}")
print(f"b0 = {b0}")
print(f"b1 = {b1}")
print(f"b2 = {b2}")

print("Натуралізація коефіцієнтів:")
print(f"Δx1 = {delta_x1}")
print(f"Δx2 = {delta_x2}")
print(f"x10 = {x10}")
print(f"x20 = {x20}")
print(f"a0 = {a_0}")
print(f"a1 = {a_1}")
print(f"a2 = {a_2}")

print(f"Натуралізоване рівняння регресії = {equation_coefficients}")
print("Коефіцієнти натуралізованого рівняння розраховано коректно.")
if equation_coefficients == y_average_value \
    else print("Коефіцієнти натуралізованого рівняння розраховано не правильно!")
```

Результати роботи:

```
Run: labwork2 x
D:\Anaconda3\envs\science_investigations_methods\python.exe D:/pr
y1 = [-5, -21, 9, 19, -9]; середнє значення = -1.4
y2 = [-3, 18, 59, 29, 40]; середнє значення = 28.6
y3 = [-21, 10, -25, 45, 62]; середнє значення = 14.2
 $\sigma^2(y1) = 195.84$ 
 $\sigma^2(y2) = 433.03999999999996$ 
 $\sigma^2(y3) = 1205.36000000000001$ 
 $\sigma(\theta) = 1.7888543819998317$ 
Fuv1 = 0.4522445963421393
Fuv2 = 6.154820261437909
Fuv3 = 2.7834842046924075
 $\theta_{uv1} = 0.27134675780528356$ 
 $\theta_{uv2} = 3.692892156862745$ 
 $\theta_{uv3} = 1.6700905228154446$ 
Ruv1 = 0.40732954539325106
Ruv2 = 1.5053724797052812
Ruv3 = 0.37459199002342697
Однорідна дисперсія = 0.9
mx1 = -0.3333333333333333
mx2 = -0.3333333333333333
my = 13.800000000000002
a1 = 1.0
a2 = -0.3333333333333333
a3 = 1.0
a11 = 5.266666666666667
a22 = -4.333333333333335
b0 = 21.400000000000006
b1 = 14.999999999999998
b2 = 7.8
```

```
Натуралізація коефіцієнтів:
 $\Delta x1 = 15.0$ 
 $\Delta x2 = 25.0$ 
 $x10 = -15.0$ 
 $x20 = 10.0$ 
 $a0 = 33.280000000000001$ 
 $a1 = 0.9999999999999999$ 
 $a2 = 0.312$ 
Натуралізоване рівняння регресії = [-1.4, 28.6, 14.2]
Коефіцієнти натуралізованого рівняння розраховано коректно.

Process finished with exit code 0
```

Відповіді на контрольні запитання:

1) Що таке регресійні поліноми і де вони застосовуються?

В теорії планування експерименту найважливішою частиною є оцінка результатів вимірів. При цьому використовують апроксимуючі поліноми, за допомогою яких ми можемо описати нашу функцію. В ТПЕ ці поліноми отримали спеціальну назву – регресійні поліноми, а їх знаходження та аналіз – регресійний аналіз. Найчастіше в якості базисної функції використовується ряд Тейлора, який має скінченну кількість членів.

2) Визначення однорідності дисперсії.

Однорідність дисперсії означає, що серед усіх дисперсій нема таких, які б значно перевищували одна одну. Перевірка однорідності проводиться за допомогою різних статистичних критеріїв.

3) Що називається повним факторним експериментом?

Повний факторний експеримент (ПФЕ) – це такий факторний експеримент, коли використовуються усі можливі комбінації рівнів факторів.