

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки

ЛАБОРАТОРНА РОБОТА № 2
з дисципліни «Методи наукових досліджень»
на тему «ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З
ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ»

ВИКОНАЛА:
студентка 2 курсу
групи ІВ-91
Яременко В.Д.
Залікова – 9132

ПЕРЕВІРИВ:
ас. Регіда П. Г.

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

$$y_{max} = (30 - N_{\text{варіанту}}) \cdot 10,$$

$$y_{min} = (20 - N_{\text{варіанту}}) \cdot 10.$$

Варіанти обираються по номеру в списку в журналі викладача.

№ варіанту	x ₁		x ₂	
	min	max	min	max
130	10	50	20	60

Програмний код:

```
import random, math, numpy as np

# Кількість експериментів, межі y і x, додатковий нульовий фактор
m = 6
y_max = (30 - 130) * 10 # -950
y_min = (20 - 130) * 10 # -1050
x_min = [10, 50]
x_max = [20, 60]
x0 = 1
# Створення матриці ПФЕ
line1 = [x_min[0], x_min[1]]
line2 = [x_min[0], x_max[1]]
line3 = [x_max[0], x_min[1]]
line4 = [x_max[0], x_max[1]]
matrix = [line1, line2, line3, line4]

# Функція знаходження середнього значення
def average(list_y):
    average = 0
    for y in list_y:
        average += y / len(list_y)
    return average

# Перевірка дисперсії на однорідність. Повторюється поки дисперсія не однорідна
def homogeneity():
    def romanovskyi(dispartion):
        def fuv():
            f_uv = []
            f_uv.append(dispartion[0] / dispartion[1])
```

```

        f_uv.append(dispartion[2] / dispartion[0])
        f_uv.append(dispartion[2] / dispartion[1])
        return f_uv
    def thetatau(f_uv):
        theta_uv = []
        theta_uv.append((m - 2 / m) * f_uv[0])
        theta_uv.append((m - 2 / m) * f_uv[1])
        theta_uv.append((m - 2 / m) * f_uv[2])
        return theta_uv
    # Основне відхилення
    sigma = math.sqrt(2 / m * (2 * m - 2) / (m - 4))
    f_uv = fuv()
    theta_uv = thetatau(f_uv)
    r_uv1 = abs(theta_uv[0] - 1) / sigma
    r_uv2 = abs(theta_uv[1] - 1) / sigma
    r_uv3 = abs(theta_uv[2] - 1) / sigma
    if (r_uv1 < 2.16) and (r_uv2 < 2.16) and (r_uv3 < 2.16): return False
    else: return True
def dispartion(list_y):
    average_y = average(list_y)
    dispartion = 0
    for y in list_y:
        dispartion += (y - average_y)**2 / len(list_y)
    return dispartion

# Функції відгуку в точках експерименту
y1 = [random.randint(0, 100) + y_min for _ in range(m)]
y2 = [random.randint(0, 100) + y_min for _ in range(m)]
y3 = [random.randint(0, 100) + y_min for _ in range(m)]

# Дисперсія
dispartion = [dispartion(y1), dispartion(y2), dispartion(y3)]
# Перевірка однорідності за критерієм Романовського
error = romanovskyi(dispartion)

if not error: print("Перевірку закінчено\nДисперсія однорідна\n")
else:
    print("Перевірку закінчено\nДисперсія неоднорідна\nПроводимо ще один
експеримент\n")
    homogeneity()
    exit()
print("Значення факторів у точках експерименту:")
for line in matrix:
    print(line)
print(f"\nФункції відгуку:\nВ першій точці: {y1}\nВ другій точці: {y2}\nВ третій
точці: {y3}")
print(f"\nСередні значення Y:\n{average(y1)}\n{average(y2)}\n{average(y3)}")
print(f"\nДисперсія: {dispartion}")
y = [y1, y2, y3]
normalisation(y)

def normalisation(y):
    normalized_matrix = [[-1, -1], [-1, 1], [1, -1], [1, 1]]
    # Розрахунок нормованих коефіцієнтів рівняння регресії
    mx1 = (normalized_matrix[0][0] + normalized_matrix[1][0] +
normalized_matrix[2][0]) / 3
    mx2 = (normalized_matrix[0][1] + normalized_matrix[1][1] +
normalized_matrix[2][1]) / 3
    my = (average(y[0]) + average(y[1]) + average(y[2])) / 3
    a1 = (normalized_matrix[0][0]**2 + normalized_matrix[1][0]**2 +
normalized_matrix[2][0]**2) / 3
    a2 = (normalized_matrix[0][0] * normalized_matrix[0][1] + normalized_matrix[1][0]

```

```

* normalized_matrix[1][1] + normalized_matrix[2][0] * normalized_matrix[2][1]) / 3
a3 = (normalized_matrix[0][1]**2 + normalized_matrix[1][1]**2 +
normalized_matrix[2][1]**2) / 3

a11 = (normalized_matrix[0][0] * average(y[0]) + normalized_matrix[1][0] *
average(y[1]) + normalized_matrix[2][0] * average(y[2])) / 3
a22 = (normalized_matrix[0][1] * average(y[0]) + normalized_matrix[1][1] *
average(y[1]) + normalized_matrix[2][1] * average(y[2])) / 3

b0_numerator = np.array([[my, mx1, mx2],
                           [a11, a1, a2],
                           [a22, a2, a3]])
b0_denominator = np.array([[1, mx1, mx2],
                             [mx1, a1, a2],
                             [mx2, a2, a3]])

b0 = np.linalg.det(b0_numerator) / np.linalg.det(b0_denominator)

b1_numerator = np.array([[1, my, mx2],
                           [mx1, a11, a2],
                           [mx2, a22, a3]])
b1_denominator = np.array([[1, mx1, mx2],
                             [mx1, a1, a2],
                             [mx2, a2, a3]])

b1 = np.linalg.det(b1_numerator) / np.linalg.det(b1_denominator)

b2_numerator = np.array([[1, mx1, my],
                           [mx1, a1, a11],
                           [mx2, a2, a22]])
b2_denominator = np.array([[1, mx1, mx2],
                             [mx1, a1, a2],
                             [mx2, a2, a3]])

b2 = np.linalg.det(b2_numerator) / np.linalg.det(b2_denominator)

# Натуралізація коефіцієнтів

delta_x1 = abs(x_max[0] - x_min[0]) / 2
delta_x2 = abs(x_max[1] - x_min[1]) / 2

x10 = (x_max[0] + x_min[0]) / 2
x20 = (x_max[1] + x_min[1]) / 2

a0 = b0 - b1 * (x10 / delta_x1) - b2 * (x20 / delta_x2)
a1 = b1 / delta_x1
a2 = b2 / delta_x2

print("\nНормалізована матриця")
for line in normalized_matrix:
    print(line)
print(f"\nb0 = {b0}; b1 = {b1}; b2 = {b2}")
print(f"Нормоване рівняння регресії:\ny = {round(b0, 3)} + {round(b1, 3)}*x1 +
{round(b2, 3)}*x2")
# Перевірка
print("Середні значення нормованих Y:")
for i in range(3):
    print(f"{b0 + b1*normalized_matrix[i][0] + b2*normalized_matrix[i][1]}")
print("Значення збігаються зі значеннями Yj")
print(f"\na0 = {a0}; a1 = {a1}; a2 = {a2}")
print(f"Натуралізоване рівняння регресії:\ny = {round(a0, 3)} + {round(a1, 3)}*x1
+ {round(a2, 3)}*x2")

```

```
# Перевірка
print("\nСередні значення натуралізованих Y:")
for i in range(3):
    print(f"{a0 + a1*matrix[i][0] + a2*matrix[i][1]}")
print("Значення збігаються зі значеннями Yj")
print("Отже, коефіцієнти рівняння регресії розраховані правильно")

# Виклик функції перевірки однорідності дисперсії
homogeneity()
```

Результат роботи програми:

Перевірку закінчено

Дисперсія однорідна

Значення факторів у точках експерименту:

[10, 50]

[10, 60]

[20, 50]

[20, 60]

Функції відгуку:

В першій точці: [-1017, -1001, -1040, -1061, -1005, -1023]

В другій точці: [-1083, -1047, -1082, -1032, -1009, -1034]

В третій точці: [-1056, -1049, -1026, -1043, -1014, -1056]

Середні значення Y:

-1024.5

-1047.8333333333333

-1040.6666666666667

Дисперсія: [427.25, 725.8055555555555, 245.2222222222223]

Нормалізована матриця

[-1, -1]

[-1, 1]

[1, -1]

[1, 1]

b0 = -1044.2500000000007; b1 = -8.083333333333584; b2 = -11.666666666666782

Нормоване рівняння регресії:

y = -1044.25 + -8.083*x1 + -11.667*x2

Середні значення нормованих Y:

-1024.5000000000005

-1047.8333333333334

-1040.6666666666674

Значення збігаються зі значеннями Yj

a0 = -891.6666666666653; a1 = -1.6166666666667169; a2 = -2.333333333333356

Натуралізоване рівняння регресії:

y = -891.667 + -1.617*x1 + -2.333*x2

Середні значення натуралізованих Y:

-1024.5000000000002

-1047.8333333333334

-1040.6666666666674

Значення збігаються зі значеннями Yj

Отже, коефіцієнти рівняння регресії розраховані правильно

Висновок:

Під час виконання даної лабораторної роботи, провела двофакторний експеримент, перевіряв однорідність дисперсії за критерієм Романовського, отримав коефіцієнти рівняння регресії, провів натуралізацію рівняння регресії. Зробивши перевірку я впевнилася в правильності коефіцієнтів
Мета лабораторної роботи досягнута.

Відповіді на контрольні питання:

1. Що таке регресійні поліноми і де вони застосовуються?

Регресійні поліноми – це апроксимуючі поліноми, за допомогою яких ми можемо описати функцію. Застосовуються в теорії планування експерименту.

2. Визначення однорідності дисперсії.

Однорідність дисперсії означає, що серед усіх дисперсій немає такої, яка б значно перевищували інші.

3. Що називається повним факторним експериментом?

ПФЕ (Повний факторний експеримент) – називається такий експеримент, при реалізації якого визначається значення параметра оптимізації при всіх можливих поєднаннях рівнів варіювання факторів.