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 ЕКСПЕРИМЕТНУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ»

ВИКОНАВ:

студент 2 курсу

групи ІВ-91

Степанюк Р. В.

Залікова – 9127

ПЕРЕВІРИВ:

ас. Регіда П. Г.

Київ – 2021

**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 (хо=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

Варіанти обираються по номеру в списку в журналі викладача.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № варіанту | x1 | | x2 | |
| min | max | min | max |
| 125 | -25 | -5 | 25 | 45 |

**Програмний код**

# Методи наукових досліджень

#

# Степанюк Роман Вікторович ІВ-91 ФІОТ

#

# Варіант 125 (x1\_min = -25, x1\_max = -5, x2\_min = 25, x2\_max = 45)

import random, math, numpy as np

# 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(dispertion):

        def fuv():

            f\_uv = []

            f\_uv.append(dispertion[0] / dispertion[1])

            f\_uv.append(dispertion[2] / dispertion[0])

            f\_uv.append(dispertion[2] / dispertion[1])

            return f\_uv

        def thetauv(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 = thetauv(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 dispertion(list\_y):

        average\_y = average(list\_y)

        dispertion = 0

        for y in list\_y:

            dispertion += (y - average\_y)\*\*2 / len(list\_y)

        return dispertion

    # 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

    dispertion = [dispertion(y1), dispertion(y2), dispertion(y3)]

    # Перевірка однорідності за критерієм Романовського

    error = romanovskyi(dispertion)

#################### В И В Е Д Е Н Н Я   Р Е З У Л Ь Т А Т І В   У   К О Н С О Л Ь ######################

    if not error: print("Перевірку закічнено\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: {dispertion}")

    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

###################### В И В Е Д Е Н Н Я   Р Е З У Л Ь Т А Т І В   У   К О Н С О Л Ь #####################

    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")

# Кількість експериментів, межі у і х, додатковий нульовий фактор

m = 6

y\_max = (30 - 125) \* 10 # -950

y\_min = (20 - 125) \* 10 # -1050

x\_min = [-25, 25]

x\_max = [-5, 45]

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]]

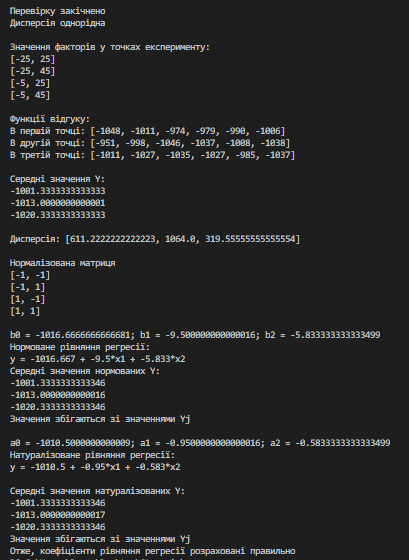
line4 = [x\_max[0], x\_max[1]]

matrix = [line1, line2, line3, line4]

# Виклик функції перевірки однорідності дисперсії

homogeneity()

**Результат роботи програми**



**Висновок:**

Під час виконання даної лабораторної роботи я провів двофакторний експеримент, перевірив однорідність дисперсії за критерієм Романовського, отримав коефіцієнти рівняння регресії, провів натуралізацію рівняння регресії. Зробивши перевірку, впевнився в правильності знайдених коефіцієнтів.  
Отже, мета лабораторної роботи була 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.

1. **Визначення однорідності дисперсії.**

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

1. **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.