Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №2

з дисципліни « Методи оптимізації та планування » на тему

«Проведення двофакторного експерименту

з використанням лінійного рівняння регресії»

Виконала:

студентка II курсу ФІОТ

групи ІВ – 93

Слюсарь Регіна

Номер залікової книжки: ІВ - 9327

Перевірив:

ас. Регіда П.Г.

Київ – 2021

**Мета роботи:** 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 (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 ymin ÷ ymax

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.

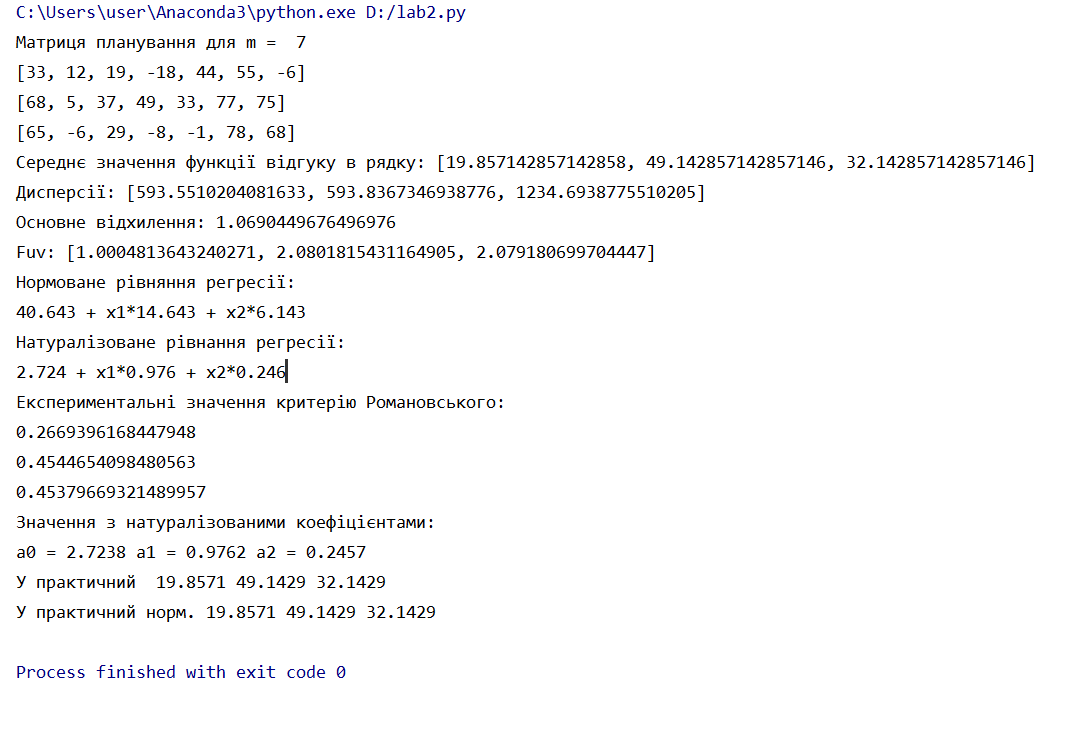
**Варіант завдання:**



**Роздруківка тексту програми:**

import random as rand  
import math  
import numpy as np  
  
m = 7  
y\_max = 80  
y\_min = -20  
  
x1\_min = 10  
x1\_max = 40  
x2\_min = 30  
x2\_max = 80  
xn = [[-1, -1], [1, -1], [-1, 1]]  
  
y = [[rand.randint(y\_min, y\_max) for j in range(7)] for i in range(3)]  
print(**'Матриця планування для m = '**, m)  
for i in range(3):  
 print(y[i])  
  
  
srY = []  
for i in range(len(y)):  
 srY.append(np.mean(y[i], axis=0))  
print(**"Середнє значення функції відгуку в рядку:"**, srY)  
  
  
def fuv(u, v):  
 if u >= v:  
 return u / v  
 else:  
 return v / u  
  
dispersion = []  
for i in range(len(y)):  
 sum = 0  
 for j in y[i]:  
 sum += (j - np.mean(y[i], axis=0)) \*\* 2  
 dispersion.append(sum / len(y[i]))  
print(**"Дисперсії:"**, dispersion)  
  
  
def discriminant(x11, x12, x13, x21, x22, x23, x31, x32, x33):  
 return x11 \* x22 \* x33 + x12 \* x23 \* x31 + x32 \* x21 \* x13 - x13 \* x22 \* x31 - x32 \* x23 \* x11 - x12 \* x21 \* x33  
  
  
sigmaTeta = math.sqrt((2 \* (2 \* m - 2)) / (m \* (m - 4)))  
print(**"Основне відхилення:"**, sigmaTeta)  
  
Fuv = []  
teta = []  
Ruv = []  
  
# F uv  
Fuv.append(max(dispersion[0], dispersion[1])/ min(dispersion[0], dispersion[1]))  
Fuv.append(max(dispersion[2], dispersion[0])/ min(dispersion[2], dispersion[0]))  
Fuv.append(max(dispersion[2], dispersion[1])/ min(dispersion[2], dispersion[1]))  
print(**'Fuv:'**, Fuv)  
# teta  
teta.append(((m - 2) / m) \* Fuv[0])  
teta.append(((m - 2) / m) \* Fuv[1])  
teta.append(((m - 2) / m) \* Fuv[2])  
# R uv  
Ruv.append(abs(teta[0] - 1) / sigmaTeta)  
Ruv.append(abs(teta[1] - 1) / sigmaTeta)  
Ruv.append(abs(teta[2] - 1) / sigmaTeta)  
# koef for 90%  
Rkr = 2  
  
for i in range(len(Ruv)):  
 if Ruv[i] > Rkr:  
 print(**'Неоднорідна дисперсія, повторіть експеримент'**)  
  
mx1 = (xn[0][0] + xn[1][0] + xn[2][0]) / 3  
mx2 = (xn[0][1] + xn[1][1] + xn[2][1]) / 3  
my = (srY[0] + srY[1] + srY[2]) / 3  
  
a1 = (xn[0][0] \*\* 2 + xn[1][0] \*\* 2 + xn[2][0] \*\* 2) / 3  
a2 = (xn[0][0] \* xn[0][1] + xn[1][0] \* xn[1][1] + xn[2][0] \* xn[2][1]) / 3  
a3 = (xn[0][1] \*\* 2 + xn[1][1] \*\* 2 + xn[2][1] \*\* 2) / 3  
  
a11 = (xn[0][0] \* srY[0] + xn[1][0] \* srY[1] + xn[2][0] \* srY[2]) / 3  
a22 = (xn[0][1] \* srY[0] + xn[1][1] \* srY[1] + xn[2][1] \* srY[2]) / 3  
  
b0 = discriminant(my, mx1, mx2, a11, a1, a2, a22, a2, a3) / discriminant(1, mx1, mx2, mx1, a1, a2, mx2, a2, a3)  
b1 = discriminant(1, my, mx2, mx1, a11, a2, mx2, a22, a3) / discriminant(1, mx1, mx2, mx1, a1, a2, mx2, a2, a3)  
b2 = discriminant(1, mx1, my, mx1, a1, a11, mx2, a2, a22) / discriminant(1, mx1, mx2, mx1, a1, a2, mx2, a2, a3)  
b00 = abs(b0)  
b11= abs(b1)  
b22 = abs(b2)  
print(**"Нормоване рівняння регресії: "**)  
print(float(**'{:.3f}'**.format(b00)), sep=**''**, end=**''**)  
  
if b1 < 0:  
 print(**" - "**, end=**''**)  
else:  
 print(**" + "**, end=**''**)  
  
print(**'x1\*'**, float(**'{:.3f}'**.format(b11)), sep=**''**, end=**''**)  
  
if b2 < 0:  
 print(**" - "**, end=**''**)  
else:  
 print(**" + "**, end=**''**)  
  
print(**'x2\*'**, float(**'{:.3f}'**.format(b22)), sep=**''**)  
  
  
y\_pr1 = b0 + b1 \* xn[0][0] + b2 \* xn[0][1]  
y\_pr2 = b0 + b1 \* xn[1][0] + b2 \* xn[1][1]  
y\_pr3 = b0 + b1 \* xn[2][0] + b2 \* xn[2][1]  
  
dx1 = abs(x1\_max - x1\_min) / 2  
dx2 = abs(x2\_max - x2\_min) / 2  
x10 = (x1\_max + x1\_min) / 2  
x20 = (x2\_max + x2\_min) / 2  
  
koef0 = b0 - (b1 \* x10 / dx1) - (b2 \* x20 / dx2)  
koef1 = b1 / dx1  
koef2 = b2 / dx2  
  
print(**'Натуралізоване рівнання регресії:'**)  
print(round(abs(koef0), 3), sep=**''**, end=**''**)  
  
if koef1 < 0:  
 print(**" - "**, end=**''**)  
else:  
 print(**" + "**, end=**''**)  
  
print(**'x1\*'**, round(abs(koef1), 3), sep=**''**, end=**''**)  
  
if koef2 < 0:  
 print(**" - "**, end=**''**)  
else:  
 print(**" + "**, end=**''**)  
  
print(**'x2\*'**, round(abs(koef2), 3), sep=**''**)  
  
  
yP1 = koef0 + koef1 \* x1\_min + koef2 \* x2\_min  
yP2 = koef0 + koef1 \* x1\_max + koef2 \* x2\_min  
yP3 = koef0 + koef1 \* x1\_min + koef2 \* x2\_max  
  
  
print(**'Експериментальні значення критерію Романовського:'**)  
for i in range(3):  
 print(Ruv[i])  
  
print(**'Значення з натуралізованими коефіцієнтами:** \n**a0 ='**, round(koef0, 4), **'a1 ='**, round(koef1, 4), **'a2 ='**, round(koef2, 4), sep=**' '**)  
print(**'У практичний '**, round(y\_pr1, 4), round(y\_pr2, 4), round(y\_pr3, 4))  
print(**'У практичний норм.'**, round(yP1, 4), round(yP2, 4), round(yP3, 4))

**Результати роботи програми:**



**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.

1. **Визначення однорідності дисперсії.**

Обирають так названу «довірчу ймовірність» p – ймовірність, з якою вимагається підтвердити гіпотезу про однорідність дисперсій. У відповідності до p і кількості дослідів m обирають з таблиці критичне значення критерію . Кожне експериментальне значення Ruv критерію Романовського порівнюється з Rкр. (значення критерію Романовського за різних довірчих ймовірностей p) і якщо для усіх кожне Ruv < Rкр., то гіпотеза про однорідність дисперсій підтверджується з ймовірністю p.

1. **Що називається повним факторним експериментом?**

Для знаходження коефіцієнтів у лінійному рівнянні регресії застосовують повний факторний експеримент (ПФЕ). Якщо в багатофакторному експерименті використані всі можливі комбінації рівнів факторів, то такий експеримент називається повним факторним експериментом.

**Висновки:**

Під час виконання даної роботи я змодельовала двофакторний експеримент, перевірила однорідність дисперсії за критерієм Романовського, отримала коефіцієнти рівняння регресії та проведела натуралізацію рівняння регресії.