Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота №3**

*з дисципліни «Методи оптимізації та планування експерименту»*

# *на тему:* «ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ.»

Виконав:

студент 2-го курсу ФІОТ

групи ІВ-81

Бєлов Микита

Номер у списку: 2

Варiант: 102

Перевірив:  
Регіда П.Г.

Київ – 2020

**Варіант:**



**Код програми:**

import numpy as np

array = np.array([[20, -20, 70, 225, 248, 256],

[20, 40, 80, 230, 259, 243],

[70, -20, 80, 241, 244, 232],

[70, 40, 70, 240, 239, 246]])

x1min = 20

x1max = 70

x2min = -20

x2max = 40

x3min = 70

x3max = 80

ymin = 223.3

ymax = 263.3

def average\_y(i):

total = 0

for j in range(3, 6):

total += array[i][j]

return total/3

def average\_x(i):

total = 0

for j in range(0, 4):

total += array[j][i]

return total/4

y1\_average = average\_y(0)

y2\_average = average\_y(1)

y3\_average = average\_y(2)

y4\_average = average\_y(3)

#cредние факторы

mx1 = average\_x(0)

mx2 = average\_x(1)

mx3 = average\_x(2)

my = np.sum(np.array([y1\_average,y2\_average,y3\_average,y4\_average]))/4

a1 = (array[0][0]\*y1\_average + array[1][0] \* y2\_average + array[2][0] \* y3\_average + array[3][0] \* y4\_average)/4

a2 = (array[0][1]\*y1\_average + array[1][1] \* y2\_average + array[2][1] \* y3\_average + array[3][1] \* y4\_average)/4

a3 = (array[0][2]\*y1\_average + array[1][2] \* y2\_average + array[2][2] \* y3\_average + array[3][2] \* y4\_average)/4

a11 = (array[0][0] \* array[0][0] + array[1][0] \* array[1][0] + array[2][0] \* array[2][0] + array[3][0] \* array[3][0])/4

a22 = (array[0][1] \* array[0][1] + array[1][1] \* array[1][1] + array[2][1] \* array[2][1] + array[3][1] \* array[3][1])/4

a33 = (array[0][2] \* array[0][2] + array[1][2] \* array[1][2] + array[2][2] \* array[2][2] + array[3][2] \* array[3][2])/4

a12 = a21 = (array[0][0] \* array[0][1] + array[1][0] \* array[1][1] + array[2][0] \* array[2][1] + array[3][0] \* array[3][1])/4

a13 = a31 = (array[0][0] \* array[0][2] + array[1][0] \* array[1][2] + array[2][0] \* array[2][2] + array[3][0] \* array[3][2])/4

a32 = a23 = (array[0][1] \* array[0][2] + array[1][1] \* array[1][2] + array[2][1] \* array[2][2] + array[3][1] \* array[3][2])/4

b0 = np.linalg.det(np.array([

[my, mx1, mx2, mx3],

[a1, a11, a12, a13],

[a2, a12, a22, a32],

[a3, a13, a23, a33]]))/np.linalg.det(np.array([

[1, mx1, mx2, mx3],

[mx1, a11, a12,a13],

[mx2, a12, a22, a32],

[mx3, a13, a23, a33]]))

b1 = np.linalg.det(np.array([

[1, my, mx2, mx3],

[mx1, a1, a12, a13],

[mx2, a2, a22, a32],

[mx3, a3, a23, a33]]))/np.linalg.det(np.array([

[1, mx1, mx2, mx3],

[mx1, a11, a12,a13],

[mx2, a12, a22, a32],

[mx3, a13, a23, a33]]))

b2 = np.linalg.det(np.array([

[1, mx1, my, mx3],

[mx1, a11, a1, a13],

[mx2, a12, a2, a32],

[mx3, a13, a3, a33]]))/np.linalg.det(np.array([

[1, mx1, mx2, mx3],

[mx1, a11, a12,a13],

[mx2, a12, a22, a32],

[mx3, a13, a23, a33]]))

b3 = np.linalg.det(np.array([

[1, mx1, mx2, my],

[mx1, a11, a12, a1],

[mx2, a12, a22, a2],

[mx3, a13, a23, a3]]))/np.linalg.det(np.array([

[1, mx1, mx2, mx3],

[mx1, a11, a12, a13],

[mx2, a12, a22, a32],

[mx3, a13, a23, a33]]))

print(f"equation\ty = {round(b0,2)} + {round(b1,4)} \* x1 + {round(b2,2)} \* x2 + {round(b3,2)} \* x3")

#Критерий Кохрена

def dispersion(line, average):

total =0

for i in range(3,6):

total += (array[line][i]-average)\*\*2

return total/3

s\_y1 = dispersion(0, y1\_average)

s\_y2 = dispersion(1, y2\_average)

s\_y3 = dispersion(2, y3\_average)

s\_y4 = dispersion(3, y4\_average)

Gp = np.max(np.array([s\_y1, s\_y2, s\_y3, s\_y4]))/np.sum(np.array([s\_y1, s\_y2, s\_y3, s\_y4]))

N = 4

m =3 #кол-во повторений

f1 = 2 #f1 = m-1

f2 = 4 #f2 = N

print(f"Kohren \tGp({round(Gp,4)}) < Gт(0.7679) => the number of repetitions is adequate")

naturalArray = np.array([[1, -1, -1, -1, 15, 18, 16],

[1, -1, 1, 1, 10, 19, 13],

[1, 1, -1, 1, 11, 14, 12],

[1, 1, 1, -1, 16, 19, 16]])

# Стьюдент

S\_B = np.sum(np.array([s\_y1, s\_y2, s\_y3, s\_y4]))/N

S\_bb = S\_B / N / m

S\_b = np.sqrt(S\_bb)

def studentKoeff(column):

total = 0

for i in range(0, 4):

total += eval("naturalArray[i][column] \* y"+str(i+1)+"\_average")

return total/4

b\_0 = studentKoeff(0)

b\_1 = studentKoeff(1)

b\_2 = studentKoeff(2)

b\_3 = studentKoeff(3)

t0 = np.abs(b\_0)/S\_b

t1 = np.abs(b\_1)/S\_b

t2 = np.abs(b\_2)/S\_b

t3 = np.abs(b\_3)/S\_b

tt = 2.306

f3 = f1 \* f2

print(f"Student\todds: t0 = {round(t0,2)} t1={round(t1,2)} t2={round(t2,2)} t3={round(t3,2)} tt={round(tt,2)} \t=> b1 and b2 and b3 can be omitted")

print(f"equation\ty = {round(b0,2)}")

Y1 = b0 + b3 \* array[0][2]

Y2 = b0 + b3 \* array[1][2]

Y3 = b0 + b3 \* array[2][2]

Y4 = b0 + b3 \* array[3][2]

#Фишера

d = 2 #кол-во новых значимых коэффицентов

f4 = N - d

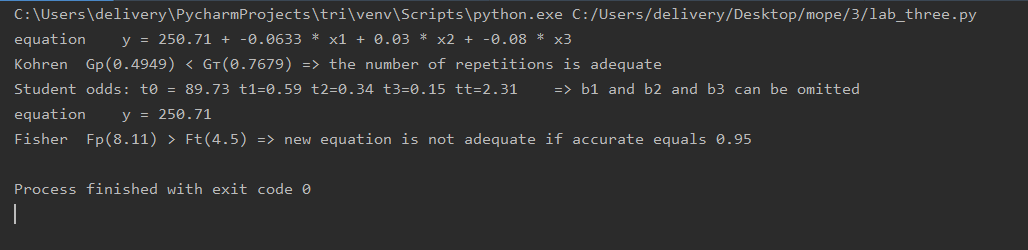
adequateDispersion = (m/(N-d)) \* ((Y1-y1\_average)\*\*2 + (Y2-y2\_average)\*\*2 + (Y3-y3\_average)\*\*2 + (Y4-y4\_average)\*\*2)

Fp = adequateDispersion/S\_bb

Ft = 4.5

print(f"Fisher\tFp({round(Fp,2)}) > Ft({round(Ft,2)}) => new equation is not adequate if accurate equals 0.95")

**Результат роботи:**



**Контрольні питання:**

1. Що називається дробовим факторним експериментом?

Існують випадки, де немає необхідності проводити повний факторний експеримент. Якщо буде використовуватися лінійна регресія, то **можливо зменшити кількість рядків** матриці **ПФЕ** до кількості коефіцієнтів регресійної моделі.

Кількість дослідів слід скоротити, використовуючи для планування так звані регулярні дробові репліки від повного факторного експерименту, які містять відповідну кількість дослідів і зберігають основні властивості матриці планування Це і означає дробовий факторний експеримент (ДФЕ).

1. Для чого потрібно розрахункове **значення Кохрена**?

Його застосовують для перевірки дисперсії на однорідність.

1. Для чого перевіряється критерій Стьюдента?

Критерій Стьюдента перевіряється для перевірки значущості коефіцієнтів регресії.

Якщо виконується нерівність **TS< Tтабл**, то приймається нуль-гіпотеза, тобто вважається, що знайдений коефіцієнт ***?s*** є статистично незначущим і його слід виключити з рівняння регресії.

Якщо **TS > Tтабл** то гіпотеза не підтверджується, тобто ***?s*** – значимий коефіцієнт і він залишається в рівнянні регресії.

1. Чим визначається критерій Фішера і як його застосовувати?

Отримане рівняння регресії необхідно перевірити на адекватність досліджуваному об'єкту.

Для цієї мети необхідно оцінити, **наскільки відрізняються середні значення у вихідної величини**, отриманої в точках факторного простору, і значення у, отриманого з рівняння регресії в тих самих точках факторного простору.

Для цього використовують **дисперсію адекватності**. Адекватність моделі перевіряють за F-критерієм Фішера, що дорівнює **відношенню дисперсії адекватності до дисперсії відтворюваності**.

**Висновки:**

В поданій роботі я відтворив програму на пайтоні, відповів на контрольні питання, розібрався із темою ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ.