Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**Методи оптимізації та планування експерименту**

Лабораторна робота №3:

«ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ

ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ»

Виконав:

студент групи ІВ-82

Сорока А.В.

Номер у списку групи : 20

Перевірив:

Регіда П. Г.

Київ 2020р.

**Лабораторна робота №3**

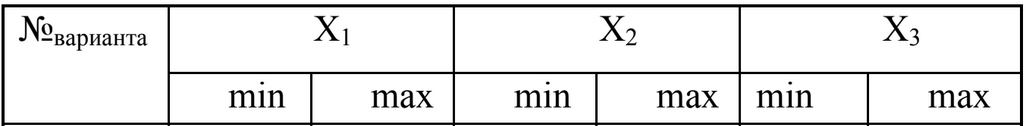
**Тема:** ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ

ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ

**Мета:** провести дробовий трьохфакторний експеримент. Скласти матрицю планування, знайти

коефіцієнти рівняння регресії, провести 3 статистичні перевірки.

**Варіант:**



**Код програми:**

Файл lab3.py

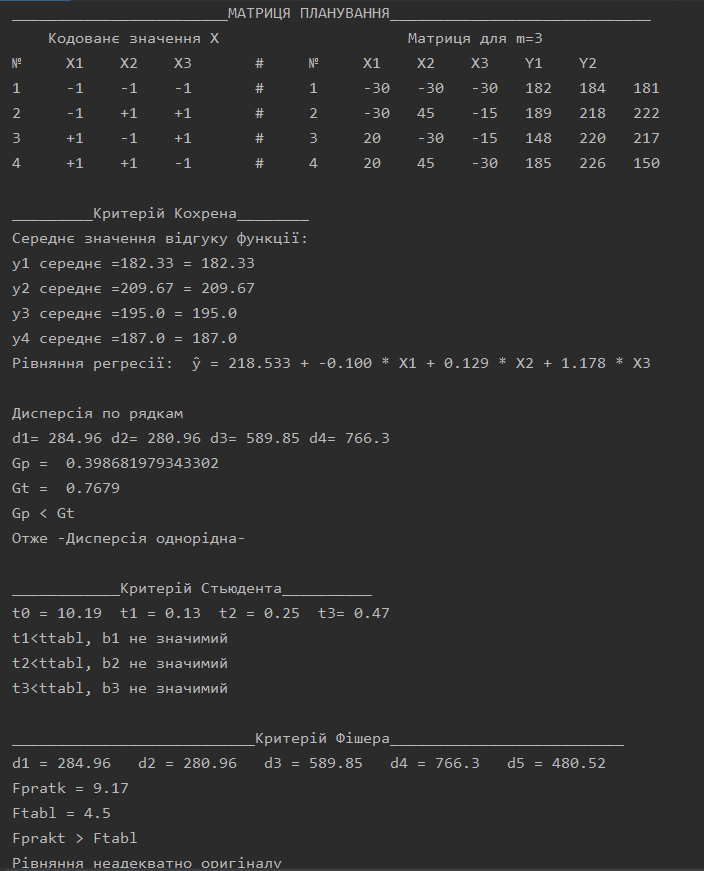
import random  
import numpy as np  
x1min = -30  
x1max = 20  
x2min = -30  
x2max = 45  
x3min = -30  
x3max = -15  
xAvmax = x1max+x2max+x3max/3  
xAvmin = x1min+x2min+x3min/3  
ymax = int(200+xAvmax)  
ymin = int(200+xAvmin)  
  
  
  
print("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_МАТРИЦЯ ПЛАНУВАННЯ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")  
print(" Кодованє значення X Матриця для m=3")  
print("{:<5} {:<5} {:<5} {:<8} {:<5} {:<5} {:<5} {:<5} {:<5} {:<5} {:<5}"  
 .format("№","X1","X2","X3","#","№","X1","X2","X3","Y1","Y2","Y3"))  
X11 = ["-1", "-1", "+1", "+1"]  
X22 = ["-1", "+1", "-1", "+1"]  
X33 = ["-1", "+1", "+1", "-1"]  
X1 = [x1min, x1min, x1max, x1max]  
X2 = [x2min, x2max, x2min, x2max]  
X3 = [x3min, x3max, x3max, x3min]  
Y1 = [random.randrange(138,247, 1) for i in range(4)]  
Y2 = [random.randrange(138,247, 1) for i in range(4)]  
Y3 = [random.randrange(138,247, 1) for i in range(4)]  
for i in range(4):  
 print("{:<5} {:<5} {:<5} {:<8} {:<5} {:<5} {:<5} {:<5} {:<5} {:<5} {:<5} {:<5}"  
 .format(i+1,X11[i],X22[i],X33[i],"#",i+1,X1[i],X2[i],X3[i],Y1[i],Y2[i],Y3[i]))

print("\n\_\_\_\_\_\_\_\_\_Критерій Кохрена\_\_\_\_\_\_\_\_")  
print("Середнє значення відгуку функції: ")  
y1av1 = (Y1[0]+Y2[0]+Y3[0])/3  
y2av2 = (Y1[1]+Y2[1]+Y3[1])/3  
y3av3 = (Y1[2]+Y2[2]+Y3[2])/3  
y4av4 = (Y1[3]+Y2[3]+Y3[3])/3  
  
mx1 = sum(X1)/4  
mx2 = sum(X2)/4  
mx3 = sum(X3)/4  
  
my = (y1av1 + y2av2 + y3av3 + y4av4)/4  
  
a1 = (X1[0]\*y1av1 + X1[1]\*y2av2 + X1[2]\*y3av3 + X1[3]\*y4av4)/4  
a2 = (X2[0]\*y1av1 + X2[1]\*y2av2 + X2[2]\*y3av3 + X2[3]\*y4av4)/4  
a3 = (X3[0]\*y1av1 + X3[1]\*y2av2 + X3[2]\*y3av3 + X3[3]\*y4av4)/4  
  
a11 = (X1[0]\*X1[0] + X1[1]\*X1[1] + X1[2]\*X1[2] + X1[3]\*X1[3])/4  
a22 = (X2[0]\*X2[0] + X2[1]\*X2[1] + X2[2]\*X2[2] + X2[3]\*X2[3])/4  
a33 = (X3[0]\*X3[0] + X3[1]\*X3[1] + X3[2]\*X3[2] + X3[3]\*X3[3])/4  
a12 = a21 = (X1[0]\*X2[0] + X1[1]\*X2[1] + X1[2]\*X2[2] + X1[3]\*X2[3])/4  
a13 = a31 = (X1[0]\*X3[0] + X1[1]\*X3[1] + X1[2]\*X3[2] + X1[3]\*X3[3])/4  
a23 = a32 = (X2[0]\*X3[0] + X2[1]\*X3[1] + X2[2]\*X3[2] + X2[3]\*X3[3])/4  
  
b01 = np.array([[my, mx1, mx2, mx3], [a1, a11, a12, a13], [a2, a12, a22, a32], [a3, a13, a23, a33]])  
b02 = np.array([[1, mx1, mx2, mx3], [mx1, a11, a12, a13], [mx2, a12, a22, a32], [mx3, a13, a23, a33]])  
b0 = np.linalg.det(b01)/np.linalg.det(b02)  
  
b11 = np.array([[1, my, mx2, mx3], [mx1, a1, a12, a13], [mx2, a2, a22, a32], [mx3, a3, a23, a33]])  
b12 = np.array([[1, mx1, mx2, mx3], [mx1, a11, a12, a13], [mx2, a12, a22, a32], [mx3, a13, a23, a33]])  
b1 = np.linalg.det(b11)/np.linalg.det(b12)  
  
b21 = np.array([[1, mx1, my, mx3], [mx1, a11, a1, a13], [mx2, a12, a2, a32], [mx3, a13, a3, a33]])  
b22 = np.array([[1, mx1, mx2, mx3], [mx1, a11, a12, a13], [mx2, a12, a22, a32], [mx3, a13, a23, a33]])  
b2 = np.linalg.det(b21)/np.linalg.det(b22)  
  
b31 = np.array([[1, mx1, mx2, my], [mx1, a11, a12, a1], [mx2, a12, a22, a2], [mx3, a13, a23, a3]])  
b32 = np.array([[1, mx1, mx2, mx3], [mx1, a11, a12, a13], [mx2, a12, a22, a32], [mx3, a13, a23, a33]])  
b3 = np.linalg.det(b31)/np.linalg.det(b32)  
  
  
  
print("y1 середнє ="+str(round(b0 + b1\*X1[0] + b2\*X2[0] + b3\*X3[0],2))+" = "+ str(round(y1av1,2)))  
print("y2 середнє ="+str(round(b0 + b1\*X1[1] + b2\*X2[1] + b3\*X3[1],2))+" = "+ str(round(y2av2,2)))  
print("y3 середнє ="+str(round(b0 + b1\*X1[2] + b2\*X2[2] + b3\*X3[2],2))+" = "+ str(round(y3av3,2)))  
print("y4 середнє ="+str(round(b0 + b1\*X1[3] + b2\*X2[3] + b3\*X3[3],2))+" = "+ str(round(y4av4,2)))  
  
print("Рівняння регресії: ŷ = {:.3f} + {:.3f} \* X1 + {:.3f} \* X2 + {:.3f} \* X3".format(b0, b1, b2, b3))  
  
print("\nДисперсія по рядкам")  
d1 = ((Y1[0] - y1av1)\*\*2 + (Y2[0] - y2av2)\*\*2 + (Y3[0] - y3av3)\*\*2)/3  
d2 = ((Y1[1] - y1av1)\*\*2 + (Y2[1] - y2av2)\*\*2 + (Y3[1] - y3av3)\*\*2)/3  
d3 = ((Y1[2] - y1av1)\*\*2 + (Y2[2] - y2av2)\*\*2 + (Y3[2] - y3av3)\*\*2)/3  
d4 = ((Y1[3] - y1av1)\*\*2 + (Y2[3] - y2av2)\*\*2 + (Y3[3] - y3av3)\*\*2)/3  
print("d1=", round(d1,2),"d2=", round(d2,2),"d3=", round(d3,2),"d4=", round(d4,2))  
  
dcouple = [d1, d2, d3, d4]  
  
m = 3  
Gp = max(dcouple)/sum(dcouple)  
f1 = m-1  
f2 = N = 4  
Gt = 0.7679  
print("Gp = ", Gp)  
print("Gt = ", Gt)  
if Gp < Gt:  
 print("Gp < Gt\nОтже -Дисперсія однорідна-")  
else:  
 print("Дисперсія неоднорідна")

print("\n\_\_\_\_\_\_\_\_\_\_\_\_Критерій Стьюдента\_\_\_\_\_\_\_\_\_\_")  
sb = sum(dcouple)/N  
ssbs = sb/N\*m  
sbs = ssbs\*\*0.5  
  
beta0 = (y1av1\*1 + y2av2\*1 + y3av3\*1 + y4av4\*1)/4  
beta1 = (y1av1\*(-1) + y2av2\*(-1) + y3av3\*1 + y4av4\*1)/4  
beta2 = (y1av1\*(-1) + y2av2\*1 + y3av3\*(-1) + y4av4\*1)/4  
beta3 = (y1av1\*(-1) + y2av2\*1 + y3av3\*1 + y4av4\*(-1))/4  
  
t0 = abs(beta0)/sbs  
t1 = abs(beta1)/sbs  
t2 = abs(beta2)/sbs  
t3 = abs(beta3)/sbs  
  
  
print("t0 =",round(t0,2)," t1 =",round(t1,2)," t2 =",round(t2,2)," t3=",round(t3,2))  
  
f3 = f1\*f2  
ttabl = 2.306  
  
  
if (t0<ttabl):  
 print("t0<ttabl, b0 не значимий")  
 b0=0  
if (t1<ttabl):  
 print("t1<ttabl, b1 не значимий")  
 b1=0  
if (t2<ttabl):  
 print("t2<ttabl, b2 не значимий")  
 b2=0  
if (t3<ttabl):  
 print("t3<ttabl, b3 не значимий")  
 b3=0  
  
yy1 = b0 + b1\*x1min + b2\*x2min + b3\*x3min  
yy2 = b0 + b1\*x1min + b2\*x2max + b3\*x3max  
yy3 = b0 + b1\*x1max + b2\*x2min + b3\*x3max  
yy4 = b0 + b1\*x1max + b2\*x2max + b3\*x3min

print("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Критерій Фішера\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")  
d = 2  
sad = ((yy1 - y1av1)\*\*2 + (yy2 - y2av2)\*\*2 + (yy3 - y3av3)\*\*2 + (yy4 - y4av4)\*\*2)\*(m/(N-d))  
Fp = sad/sb  
print("d1 =", round(d1,2), " d2 =", round(d2,2), " d3 =", round(d3,2), " d4 =", round(d4,2), " d5 =", round(sb,2))  
print("Fpratk =", round(Fp,2))  
print('Ftabl = 4.5')  
Ft=4.5  
if Fp>Ft:  
 print("Fprakt > Ftabl","\nРівняння неадекватно оригіналу")  
else:  
 print("Fprakt < Ftabl","\nРівняння адекватно оригіналу")

**Результат виконання роботи програми:**



**Висновок:**

У ході даної лабораторної роботи був проведений дробовий трьохфакторний експеримент, складена матриця планування, знайдені

коефіцієнти рівняння регресії, проведені перевірки Кохрена, Стьюдента та Фішера. Розроблена тестова програма, результати її роботи наведені вище.

**Контрольні запитання**

1. *Що називається дробовим факторним експериментом?*

У деяких випадках немає необхідності проводити повний факторний експеримент (ПФЕ). 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, що містять відповідну кількість дослідів і зберігають основні властивості матриці планування – це означає дробовий факторний експеримент (ДФЕ).

1. *Для чого потрібно розрахункове значення Кохрена?*

Для перевірки дисперсії на однорідність.

1. *Для чого перевіряється критерій Стьюдента?*

Для перевірки значущості коефіцієнтів регресії. Тобто, Якщо виконується нерівність *ts*< *tтабл*, то приймається нуль-гіпотеза, тобто вважається, що знайдений коефіцієнт *βs* є статистично незначущим і його слід виключити з рівняння регресії. Якщо *ts* > *tтабл* то гіпотеза не підтверджується, тобто *βs* – 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, який дорівнює відношенню дисперсії адекватності до дисперсії відтворюваності.