

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

Лабораторна робота №2

З дисципліни «Методи наукових досліджень»
ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З
ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ

ВИКОНАВ:
Студент II курсу ФІОТ
Групи ІВ-91
Дерачиц Віталій Віталійович
Номер заліковки: 9109
Номер у списку: 9

ПЕРЕВІРИВ:
ас. Регіда П. Г.

Київ 2021 р.

Мета: провести двофакторний експеримент, перевірити однорідність дисперсії за

критерієм Романовського, отримати коефіцієнти рівняння регресії, провести натуралізацію рівняння регресії.

Введемо такі позначення:

N – кількість точок плану (рядків матриці планування)

k – кількість факторів (кількість x)

m – кількість дослідів y за однієї і тієї ж комбінації факторів (test)

s x - нормовані значення факторів ($s = 1, k$)

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 ($x_0=1$).
3. Провести експеримент в усіх точках повного факторного простору (знайти значення функції відгуку y). Значення функції відгуку задати випадковим чином у відповідності до варіанту y діапазоні $y_{\min} \div y_{\max}$
 $y_{\max} = (30 - N_{\text{варіанту}}) * 10$,
 $y_{\min} = (20 - N_{\text{варіанту}}) * 10$.

Варіанти обираються по номеру в списку в журналі викладача.

№ _{варіанта}	x_1		x_2	
	min	max	min	max
109	-20	15	10	60

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.

Программный код

```
import random
from numpy import linalg
import sys

m = 5
main_dev = ((2 * (2 * m - 2)) / (m * (m - 4))) ** (1 / 2)

x1_min = -20
x1_max = 15
x2_min = 10
x2_max = 60

X = [[-1, -1],
      [1, -1],
      [-1, 1]]

y_max = (30 - 109) * 10
y_min = (20 - 109) * 10

array1 = [random.randint(y_min, y_max) for i in range(5)]
array2 = [random.randint(y_min, y_max) for k in range(5)]
array3 = [random.randint(y_min, y_max) for n in range(5)]

average1 = sum(array1) / len(array1)
average2 = sum(array2) / len(array2)
average3 = sum(array3) / len(array3)

dispersion1 = sum([(average1 - array1[i]) ** 2 for i in range(5)]) / 5
dispersion2 = sum([(average2 - array2[k]) ** 2 for k in range(5)]) / 5
dispersion3 = sum([(average3 - array3[n]) ** 2 for n in range(5)]) / 5

dispersions_sum = dispersion1 + dispersion2 + dispersion3
dispersion1_percentage = dispersion1 / dispersions_sum
dispersion2_percentage = dispersion2 / dispersions_sum
dispersion3_percentage = dispersion3 / dispersions_sum

Fuv1 = dispersion1 / dispersion2
Fuv2 = dispersion3 / dispersion1
Fuv3 = dispersion3 / dispersion2

Ouv1 = ((m - 2) / m) * Fuv1
Ouv2 = ((m - 2) / m) * Fuv2
Ouv3 = ((m - 2) / m) * Fuv3

Ruv1 = abs(Ouv1 - 1) / main_dev
Ruv2 = abs(Ouv2 - 1) / main_dev
Ruv3 = abs(Ouv3 - 1) / main_dev

mx1 = (X[0][0] + X[1][0] + X[2][0]) / 3
mx2 = (X[0][1] + X[1][1] + X[2][1]) / 3
my = (average1 + average2 + average3) / 3

a1 = ((X[0][0]) ** 2 + (X[1][0]) ** 2 + (X[2][0]) ** 2) / 3
a2 = (X[0][0] * X[0][1] + X[1][0] * X[1][1] + X[2][0] * X[2][1]) / 3
a3 = ((X[0][1]) ** 2 + (X[1][1]) ** 2 + (X[2][1]) ** 2) / 3

a11 = (X[0][0] * average1 + X[1][0] * average2 + X[2][0] * average3) / 3
a22 = (X[0][1] * average1 + X[1][1] * average2 + X[2][1] * average3) / 3

b0 = (linalg.det([[my, mx1, mx2],
                  [a11, a1, a2],
```

```

        [a22, a2, a3])) / (linalg.det([[1, mx1, mx2],
                                         [mx1, a1, a2],
                                         [mx2, a2, a3]]))

b1 = (linalg.det([[1, my, mx2],
                  [mx1, a11, a2],
                  [mx2, a22, a3]])) / (linalg.det([[1, mx1, mx2],
                                                    [mx1, a1, a2],
                                                    [mx2, a2, a3]]))

b2 = (linalg.det([[1, mx1, my],
                  [mx1, a1, a11],
                  [mx2, a2, a22]])) / (linalg.det([[1, mx1, mx2],
                                                    [mx1, a1, a2],
                                                    [mx2, a2, a3]]))

deltaX1 = abs(x1_max - x1_min) / 2
deltaX2 = abs(x2_max - x2_min) / 2
x10 = (x1_max + x1_min) / 2
x20 = (x2_max + x2_min) / 2

a0 = b0 - b1 * x10 / deltaX1 - b2 * x20 / deltaX2
a1 = b1 / deltaX1
a2 = b2 / deltaX2

print("y_min = " + str(y_min) + " y_max = " + str(y_max))

print("Експеримент 1 Y: ", array1)
print("Експеримент 2 Y: ", array2)
print("Експеримент 3 Y: ", array3)

print("\nСереднє значення 1 Y: ", average1)
print("Середнє значення 2 Y: ", average2)
print("Середнє значення 3 Y: ", average3)

print("\nДисперсія 1: " + str(dispersion1) + " у відсотках:" +
      str(dispersion1_percentage))
print("Дисперсія 2: " + str(dispersion2) + " у відсотках:" +
      str(dispersion2_percentage))
print("Дисперсія 3: " + str(dispersion3) + " у відсотках:" +
      str(dispersion3_percentage))

print("\nFuv1: ", Fuv1)
print("Fuv2: ", Fuv2)
print("Fuv3: ", Fuv3)

print("\nOuv1: ", Ouv1)
print("Ouv2: ", Ouv2)
print("Ouv3: ", Ouv3)

if Ruv1 > 2 or Ruv2 > 2 or Ruv3 > 2:
    print("\nДисперсія неоднорідна!")
    sys.exit()

print("\nRuv1: " + str(Ruv1) + "<Rkr = 2")
print("Ruv2: " + str(Ruv2) + "<Rkr = 2")
print("Ruv3: " + str(Ruv3) + "<Rkr = 2 \nОтже дисперсія однорідна")

print("\nmx1: ", mx1)
print("mx2: ", mx2)
print("my: ", my)

print("\na1 - ", a1)
print("a2 - ", a2)

```

```

print("a3 - ", a3)

print("a11 - ", a11)
print("a22 - ", a22)

print("\nb0 - ", b0)
print("b1 - ", b1)
print("b2 - ", b2)

print("\nНормоване рівняння регресії: y=" + str(b0) + "+" + str(b1) + "*x1 +" +
str(b2) + "*x2")
print("Зробимо перевірку: \n"
      + str(b0) + "+" + (" + str(-1 * b1) + ") + (" + str(-1 * b2) + ") = " +
str(b0 - b1 - b2) + "\n"
      + str(b0) + "+" + (" + str(b1) + ") + (" + str(-1 * b2) + ") = " + str(b0 +
b1 - b2) + "\n"
      + str(b0) + "+" + (" + str(-1 * b1) + ") + (" + str(b2) + ") = " + str(b0 -
b1 + b2) + "\n"
      + "Результат збігається з середніми значеннями y \n")

print("Натуралізоване рівняння регресії \n y = " + str(a0) + "+" + str(a1) +
"*x1 +" + str(a2) + "*x2")
print("Зробимо перевірку по рядках: \n"
      + str(a0) + " + (" + str(a1 * x1_min) + ") + (" + str(a2 * x2_min) + ") =
" + str(a0 + a1 * x1_min + a2 * x2_min) + "\n"
      + str(a0) + " + (" + str(a1 * x1_max) + ") + (" + str(a2 * x2_min) + ") =
" + str(a0 + a1 * x1_max + a2 * x2_min) + "\n"
      + str(a0) + " + (" + str(a1 * x1_min) + ") + (" + str(a2 * x2_max) + ") =
" + str(a0 + a1 * x1_min + a2 * x2_max) + "\n"
      + "Отже коефіцієнти натуралізованого рівняння регресії вірні \n")

```

Результати роботи програми

```
C:\Users\derac\Anaconda3\python.exe C:/Users/derac/PycharmProjects/lab2MND/main.py
y_min = -890 y_max = -790
Експеримент 1 Y: [-857, -816, -862, -795, -793]
Експеримент 2 Y: [-889, -793, -806, -839, -800]
Експеримент 3 Y: [-862, -839, -795, -814, -862]

Середнє значення 1 Y: -824.6
Середнє значення 2 Y: -825.4
Середнє значення 3 Y: -834.4

Дисперсія 1: 879.4400000000002 у відсотках:0.3094092150074587
Дисперсія 2: 1260.24 у відсотках:0.44338427763235666
Дисперсія 3: 702.64 у відсотках:0.24720650736018462

Fuv1: 0.6978353329524536
Fuv2: 0.7989629764395523
Fuv3: 0.5575445946803783

Ouv1: 0.41870119977147213
Ouv2: 0.47937778586373136
Ouv3: 0.33452675680822697

Ruv1: 0.3249559081375147<Rkr = 2
Ruv2: 0.29103666535128714<Rkr = 2
Ruv3: 0.37201085224601343<Rkr = 2
Отже дисперсія однорідна

mx1: -0.3333333333333333
mx2: -0.3333333333333333
my: -828.1333333333333

a1 - -0.022857142857139107
a2 - -0.19599999999999999
a3 - 1.0
a11 - 277.86666666666667
a22 - 271.86666666666667

b0 - -829.90000000000004
b1 - -0.39999999999999999
b2 - -4.8999999999999997

Нормоване рівняння регресії:  $y = -829.90000000000004 + (-0.39999999999999999)x_1 + (-4.8999999999999997)x_2$ 
Зробимо перевірку:
-829.90000000000004 + (0.39999999999999999) + (4.8999999999999997) = -824.60000000000005
-829.90000000000004 + (-0.39999999999999999) + (4.8999999999999997) = -825.40000000000004
-829.90000000000004 + (0.39999999999999999) + (-4.8999999999999997) = -834.40000000000004
Результат збігається з середніми значеннями y

Натуралізоване рівняння регресії
 $y = -823.0971428571432 + (-0.022857142857139107)x_1 + (-0.19599999999999999)x_2$ 
Зробимо перевірку по рядках:
-823.0971428571432 + (0.4571428571428213) + (-1.9599999999999999) = -824.60000000000005
-823.0971428571432 + (-0.34285714285708657) + (-1.9599999999999999) = -825.40000000000003
-823.0971428571432 + (0.4571428571428213) + (-11.759999999999999) = -834.40000000000004
Отже коефіцієнти натуралізованого рівняння регресії вірні
```

Контрольні запитання:

1. 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.
2. Однорідність дисперсії – необхідна умова підтвердження гіпотези про забезпечення нормального закону розподілу вимірюваної величини при обраній кількості повторів m та ймовірності p .
3. Повний факторний експеримент – це експеримент у якому використані всі можливі комбінації рівнів факторів.