

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
експерименту»
на тему: «Проведення двофакторного експерименту з
використанням лінійного рівняння регресії».

Виконав:
студент групи ІО-92
Гладков Даніїл
Залікова книжка № ІО-9204
Номер у списку групи - 02

Перевірив:
ас. Регіда П.Г.

Тема: «Проведення двофакторного експерименту з використанням лінійного рівняння регресії».

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 ($x_0=1$).
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 $y_{\min} \div y_{\max}$
 $y_{\max} = (30 - N_{\text{варіанту}}) * 10$,
 $y_{\min} = (20 - N_{\text{варіанту}}) * 10$.
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:

№ _{варіанта}	x ₁		x ₂	
	min	max	min	max
202	20	70	30	80

$$y_{\max} = (30 - 2) * 10 = 280,$$

$$y_{\min} = (20 - 2) * 10 = 180.$$

Роздруківка тексту програми:

```
from prettytable import PrettyTable as PT
import random as r
import numpy as np
from math import *

class Laba2:
    def __init__(self):
        self.m = 5
        self.RKR = {5:2.00,6:2.00,7:2.17,8:2.17,9:2.29,10:2.29}
        self.X1min, self.X1max, self.X2min, self.X2max = 20, 70, 30, 80
        self.Ymin, self.Ymax = ((20-2)*10), ((30-2)*10)
        self.main1(), self.krit(), self.main2(), self.printing()

    def main1(self):
        def FUV(U,V):
            if (U>=V):
                return (U/V)
            else:
                return (V/U)

        self.Y = [[r.randint(self.Ymin, self.Ymax) for i in range(self.m)] for i in
range(3)]
        self.X1, self.X2 = [-1, 1, -1], [-1, -1, 1],
        if (self.m > 5):
            print("Змінений масив Y-ків через неоднорідність дисперсії")
            for i in range(3):
                print(self.Y[i])
        self.table1 = PT()
        self.table1.field_names = ["X1", "X2", "Y1", "Y2", "Y3", "Y4", "Y5"]
        self.table1.add_rows([
            [self.X1[0], self.X2[0], self.Y[0][0], self.Y[0][1], self.Y[0][2],
self.Y[0][3], self.Y[0][4]],
            [self.X1[1], self.X2[1], self.Y[1][0], self.Y[1][1], self.Y[1][2],
self.Y[1][3], self.Y[1][4]],
            [self.X1[2], self.X2[2], self.Y[2][0], self.Y[2][1], self.Y[2][2],
self.Y[2][3], self.Y[2][4]]])
        self.table2 = PT()
        self.table2.field_names = ["X1min", "X1max", "X2min", "X2max", "Ymin", "Ymax"]
        self.table2.add_rows([[self.X1min, self.X1max, self.X2min, self.X2max,
self.Ymin, self.Ymax]])
        self.Y1_,self.Y2_,self.Y3_=(sum(self.Y[0][j] for j in
range(self.m))/self.m),(sum(self.Y[1][j] for j in
range(self.m))/self.m),(sum(self.Y[2][j] for j in range(self.m))/self.m)
        self.dispY1=((1/self.m)*(sum([pow((j - self.Y1_),2) for j in self.Y[0]])))
        self.dispY2=((1/self.m)*(sum([pow((j - self.Y2_),2) for j in self.Y[1]])))
        self.dispY3=((1/self.m)*(sum([pow((j - self.Y3_),2) for j in self.Y[2]])))
        self.dispTheta=pow(((2*(2*self.m-2))/(self.m*(self.m-4))),0.5)

        self.Fuv1,self.Fuv2,self.Fuv3=FUV(self.dispY1,self.dispY2),FUV(self.dispY3,self.dispY1)
        ,FUV(self.dispY3,self.dispY2)
        self.Tuv1,self.Tuv2,self.Tuv3=((self.m-2)/self.m)*self.Fuv1),(((self.m-
2)/self.m)*self.Fuv2),(((self.m-2)/self.m)*self.Fuv3)
        self.Ruv1,self.Ruv2,self.Ruv3=(fabs(self.Tuv1-
1)/self.dispTheta),(fabs(self.Tuv2-1)/self.dispTheta),(fabs(self.Tuv3-
1)/self.dispTheta)
        self.RUV=[self.Ruv1,self.Ruv2,self.Ruv3]
        for i in range(len(self.RUV)):
            self.krit()
            if self.RUV[i] > self.rkr:
                self.m = self.m + 1
```

```

        print("Дісперсія неоднорідна! Змінимо кількість дослідів на
m={0}".format(self.m))
        self.main1()

    def krit(self):
        self.rkr=2
        self.RKRvalues, self.RKRkeys = list(self.RKR.values()), list(self.RKR.keys())
        for keys in range(len(self.RKRkeys)):
            if (self.RKRkeys[keys] == self.m):
                self.rkr = self.RKRvalues[keys]
                return self.rkr

    def main2(self):

self.mx1, self.mx2, self.my=((self.X1[0]+self.X1[1]+self.X1[2])/3), ((self.X2[0]+self.X2[1]
+self.X2[2])/3), ((self.Y1_+self.Y2_+self.Y3_)/3)

self.a1, self.a2, self.a3=((pow(self.X1[0],2)+pow(self.X1[1],2)+pow(self.X1[2],2))/3),
((self.X1[0]*self.X2[0]+self.X1[1]*self.X2[1]+self.X1[2]*self.X2[2])/3), ((pow(self.X2[0
],2)+pow(self.X2[1],2)+pow(self.X2[2],2))/3)

self.a11, self.a22=((self.X1[0]*self.Y1_)+(self.X1[1]*self.Y2_)+(self.X1[2]*self.Y3_))/
3), (((self.X2[0]*self.Y1_)+(self.X2[1]*self.Y2_)+(self.X2[2]*self.Y3_))/3)
        self.b0 =
((np.linalg.det(np.array([[self.my, self.mx1, self.mx2], [self.a11, self.a1,
self.a2], [self.a22, self.a2,
self.a3]])))/((np.linalg.det(np.array([[1, self.mx1, self.mx2], [self.mx1, self.a1, self.a2]
, [self.mx2, self.a2, self.a3]])))))
        self.b1 = ((np.linalg.det(np.array([[1, self.my
, self.mx2], [self.mx1, self.a1, self.a11], [self.mx2, self.a2, self.a22], [self.mx2, self.a2, self.a3]])))/((np.linalg.det(
np.array([[1, self.mx1, self.mx2], [self.mx1, self.a1, self.a2], [self.mx2, self.a2, self.a3]]))
))))
        self.b2 = ((np.linalg.det(np.array([[1, self.mx1
, self.my], [self.mx1, self.a1, self.a11], [self.mx2, self.a2, self.a22]])))/((np.linalg.det(n
p.array([[1, self.mx1, self.mx2], [self.mx1, self.a1, self.a2], [self.mx2, self.a2, self.a3]]))
))))
        self.deltaX1, self.deltaX2=(fabs(self.X1max-self.X1min)/2), (fabs(self.X2max-
self.X2min)/2)
        self.X10, self.X20=((self.X1max+self.X1min)/2), ((self.X2max+self.X2min)/2)
        self.a0, self.a1, self.a2 = (self.b0-(self.b1*(self.X10/self.deltaX1))-
(self.b2*(self.X20/self.deltaX2))), (self.b1/self.deltaX1), (self.b2/self.deltaX2)

    def printing(self):
        if (self.m == 5):
            print("Матриця планування експерименту (m=5):")
        else:
            print("Початкова матриця планування експерименту (m=5):")
            print(self.table1)
            print("Значення мінімумів та максимумів:")
            print(self.table2)
            print("1) Перевірка однорідності дисперсії за критерієм Романовського:")
            print("1.1) Середнє значення функції відгуку в рядку: Y1сер={0}, Y2сер={1},
Y3сер={2}".format(self.Y1_, self.Y2_, self.Y3_))
            print("1.2) Дисперсія по рядках: D^2{{Y1}}={0:.3f}, D^2{{Y2}}={1:.3f},
D^2{{Y3}}={2:.3f}".format(self.dispY1, self.dispY2, self.dispY3))
            print("1.3) Основне відхилення: D0={0:.3f}".format(self.dispTheta))
            print("1.4) Обчислення Fuv: Fuv1={0:.3f}, Fuv2={1:.3f},
Fuv3={2:.3f}".format(self.Fuv1, self.Fuv2, self.Fuv3))
            print("1.5) Обчислення 0uv: 0uv1={0:.3f}, 0uv2={1:.3f},
0uv3={2:.3f}".format(self.Tuv1, self.Tuv2, self.Tuv3))
            print("1.6) Обчислення Ruv: Ruv1={0:.3f}, Ruv2={1:.3f},
Ruv3={2:.3f}".format(self.Ruv1, self.Ruv2, self.Ruv3))

```

```

        print("1.7) Перевірка: \n Ruv1 = {0:.3f} < Rkp = {1:.3f} \n Ruv2 = {2:.3f} <
Rkp = {3:.3f} \n Ruv3 = {4:.3f} < Rkp = {5:.3f}".format(self.Ruv1, self.rkr, self.Ruv2,
self.rkr, self.Ruv3, self.rkr))
        print("Отже, дисперсія однорідна з ймовірністю 0.9!")
        print("2) Розрахунок нормованих коефіцієнтів рівняння регресії:")
        print("mx1 = {0:.3f}, mx2 = {1:.3f}, my = {2:.3f}".format(self.mx1, self.mx2,
self.my))
        print("a1 = {0:.3f}, a2 = {1:.3f}, a3 = {2:.3f}".format(self.a1, self.a2,
self.a3))
        print("a11 = {0:.3f}, a22 = {1:.3f}".format(self.a11, self.a22))
        print("Знаходження коефіцієнтів регресії методом Крамера:")
        print("b0 = {0:.3f}, b1 = {1:.3f}, b2 = {2:.3f}".format(self.b0, self.b1,
self.b2))
        print("Нормоване рівняння регресії:")
        print("y = b0 + b1*X1 + b2*X2 = {0:.4f} + ({1:.4f})*X1 +
({2:.4f})*X2".format(self.b0, self.b1, self.b2))
        print("Перевірка:")
        print("b0 - b1 - b2 = {0:.3f} - ({1:.3f}) - ({2:.3f}) = {3:.3f} = Y1 середнє =
{4:.3f}".format(self.b0, self.b1, self.b2, (self.b0 - self.b1 - self.b2), self.Y1_))
        print("b0 + b1 - b2 = {0:.3f} + ({1:.3f}) - ({2:.3f}) = {3:.3f} = Y2 середнє =
{4:.3f}".format(self.b0, self.b1, self.b2, (self.b0 + self.b1 - self.b2), self.Y2_))
        print("b0 - b1 + b2 = {0:.3f} - ({1:.3f}) + ({2:.3f}) = {3:.3f} = Y3 середнє =
{4:.3f}".format(self.b0, self.b1, self.b2, (self.b0 - self.b1 + self.b2), self.Y3_))
        print("Результат збігається з середніми значеннями Y!")
        print("3) Натуралізація коефіцієнтів")
        print("ΔX1 = {0:.3f}, ΔX2 = {1:.3f}".format(self.deltaX1, self.deltaX2))
        print("X10 = {0:.3f}, X20 = {1:.3f}".format(self.X10, self.X20))
        print("a0 = {0:.3f}, a1 = {1:.3f}, a2 = {2:.3f}".format(self.a0, self.a1,
self.a2))
        print("Натуралізоване рівняння регресії:")
        print("y = a0 + a1*X1 + a2*X2 = {0:.4f} + ({1:.4f})*X1 +
({2:.4f})*X2".format(self.a0, self.a1, self.a2))
        print("Перевірка по рядках:")
        print("{0:.3f} + ({1:.3f})*{2:.3f} + ({3:.3f})*{4:.3f} = {5:.3f} = Y1 середнє =
{6:.3f}".format(self.a0, self.a1, self.X1min, self.a2, self.X2min, (self.a0 + self.a1 *
self.X1min + self.a2 * self.X2min), self.Y1_))
        print("{0:.3f} + ({1:.3f})*{2:.3f} + ({3:.3f})*{4:.3f} = {5:.3f} = Y2 середнє =
{6:.3f}".format(self.a0, self.a1, self.X1max, self.a2, self.X2min, (self.a0 + self.a1 *
self.X1max + self.a2 * self.X2min), self.Y2_))
        print("{0:.3f} + ({1:.3f})*{2:.3f} + ({3:.3f})*{4:.3f} = {5:.3f} = Y3 середнє =
{6:.3f}".format(self.a0, self.a1, self.X1min, self.a2, self.X2max, (self.a0 + self.a1 *
self.X1min + self.a2 * self.X2max), self.Y3_))
        print("Отже, коефіцієнти натуралізованого рівняння регресії вірні.")
Laba2()

```

Роздруківка результатів виконання програми:

```
D:\Programming\Anaconda\python.exe "C:/Users/Daniil/Desktop/КПИ/4 СЕМЕСТР/МОПЕ (4)/Лабораторні роботи/Laba2МОПЕ/Laba2МОПЕ.py"
Матриця планування експерименту (m=5):
+-----+-----+-----+-----+-----+
| X1 | X2 | Y1 | Y2 | Y3 | Y4 | Y5 |
+-----+-----+-----+-----+
| -1 | -1 | 263 | 269 | 267 | 266 | 203 |
| 1  | -1 | 238 | 235 | 195 | 262 | 211 |
| -1 | 1  | 277 | 238 | 235 | 183 | 269 |
+-----+-----+-----+-----+
Значення мінімумів та максимумів:
+-----+-----+-----+-----+-----+
| X1min | X1max | X2min | X2max | Ymin | Ymax |
+-----+-----+-----+-----+-----+
| 20    | 70    | 30    | 80    | 180   | 280   |
+-----+-----+-----+-----+-----+
1) Перевірка однорідності дисперсії за критерієм Романовського:
1.1) Середнє значення функції відгуку в рядку: Y1сер=253.6, Y2сер=228.2, Y3сер=240.4
1.2) Дисперсія по рядках: D^2{Y1}=643.840, D^2{Y2}=536.560, D^2{Y3}=1097.440
1.3) Основне відхилення: D0=1.789
1.4) Обчислення Fuv: Fuv1=1.200, Fuv2=1.705, Fuv3=2.045
1.5) Обчислення 0uv: 0uv1=0.720, 0uv2=1.023, 0uv3=1.227
1.6) Обчислення Ruv: Ruv1=0.157, Ruv2=0.013, Ruv3=0.127
1.7) Перевірка:
Ruv1 = 0.157 < Rkp = 2.000
Ruv2 = 0.013 < Rkp = 2.000
Ruv3 = 0.127 < Rkp = 2.000
Отже, дисперсія однорідна з ймовірністю 0.9!
2) Розрахунок нормованих коефіцієнтів рівняння регресії:
mx1 = -0.333, mx2 = -0.333, my = 240.733
a1 = -0.508, a2 = -0.264, a3 = 1.000
a11 = -88.600, a22 = -80.467
Знаходження коефіцієнтів регресії методом Крамера:
b0 = 234.300, b1 = -12.700, b2 = -6.600
Нормоване рівняння регресії:
y = b0 + b1*X1 + b2*X2 = 234.3000 + (-12.7000)*X1 + (-6.6000)*X2
Перевірка:
b0 - b1 - b2 = 234.300 - (-12.700) - (-6.600) = 253.600 = Y1 середнє = 253.600
b0 + b1 - b2 = 234.300 + (-12.700) - (-6.600) = 228.200 = Y2 середнє = 228.200
b0 - b1 + b2 = 234.300 - (-12.700) + (-6.600) = 240.400 = Y3 середнє = 240.400
Результат збігається з середніми значеннями Y!
3) Натуралізація коефіцієнтів
ΔX1 = 25.000, ΔX2 = 25.000
X10 = 45.000, X20 = 55.000
a0 = 271.680, a1 = -0.508, a2 = -0.264
Натуралізоване рівняння регресії:
y = a0 + a1*X1 + a2*X2 = 271.6800 + (-0.5080)*X1 + (-0.2640)*X2
Перевірка по рядках:
271.680 + (-0.508)*20.000 + (-0.264)*30.000 = 253.600 = Y1 середнє = 253.600
271.680 + (-0.508)*70.000 + (-0.264)*30.000 = 228.200 = Y2 середнє = 228.200
271.680 + (-0.508)*20.000 + (-0.264)*80.000 = 240.400 = Y3 середнє = 240.400
Отже, коефіцієнти натуралізованого рівняння регресії вірні.

Process finished with exit code 0
```

Висновки:

У ході виконання лабораторної роботи я провів двофакторний експеримент, перевіряв однорідність дисперсії за критерієм Романовського, отримав коефіцієнти рівняння регресії, провів натуралізацію рівняння регресії. Закріпив отримані знання практичним їх використанням при написанні програми, що реалізує завдання лабораторної роботи. Мета лабораторної роботи досягнута.

Відповіді на контрольні запитання:

1) Що таке регресійні поліноми і де вони застосовуються?

Важливою частиною теорії планування експерименту 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. Найчастіше в якості базисної функції використовується ряд Тейлора, який має скінченну кількість членів.

2) Визначення однорідності дисперсій.

Однорідність дисперсій означає, що серед всіх дисперсій немає таких, які б значно перевищували всі інші. Для перевірки однорідності дисперсій у всіх точках спектра плану використовується або критерій Кохрена G, або критерій Фішера F.

Критерій Фішера дозволяє порівнювати дві дисперсії і визначається зі співвідношення $F = S_{\max}^2 / S_{\min}^2$.

Критерій Кохрена заснований на розподілі відносини максимальної дисперсії до суми всіх дисперсій: $G = S_{i\max}^2 / \sum_{i=1}^N S_i^2$.

3) 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. Якщо ми маємо справу з k факторами, кожен з яких може встановлюватися на q рівнях, то для того, щоб здійснити повний факторний експеримент необхідно поставити $n = q^k$ дослідів.