

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 №6
З дисципліни «Методи наукових досліджень»
За темою:
«Проведення трьохфакторного експерименту при
використанні рівняння регресії з квадратичними членами»

ВИКОНАВ:
Студент II курсу ФІОТ
Групи ІВ-93
Ровишин Андрій
Номер у списку – 19
Варіант - 319

ПЕРЕВІРИВ:
асистент
Регіда П.Г.

Київ 2021 р.

Мета: Провести трьохфакторний експеримент і отримати адекватну модель – рівняння регресії, використовуючи рототабельний композиційний план.

Завдання:

1. Ознайомитися з теоретичними відомостями.
2. Вибрати з таблиці варіантів і записати в протокол інтервали значень x_1, x_2, x_3 . Обчислити і записати значення, відповідні кодованим значенням факторів $+1; -1; +l; -l; 0$ для $\bar{x}_1, \bar{x}_2, \bar{x}_3$.
3. Значення функції відгуку знайти за допомогою підстановки в формулу:

$$y_i = f(x_1, x_2, x_3) + \text{random}(10) - 5,$$

де $f(x_1, x_2, x_3)$ вибирається по номеру в списку в журналі викладача.

4. Провести експерименти і аналізуючи значення статистичних перевірок, отримати адекватну модель рівняння регресії. При розрахунках використовувати натуральні значення факторів.
5. Зробити висновки по виконаній роботі.

№ варіанту	x_1		x_2		x_3		$f(x_1, x_2, x_3)$
	min	max	min	max	min	max	
319	-20	30	20	60	-20	-5	$0,5+10,0*x_1+5,2*x_2+6,6*x_3+0,4*x_1*x_1+0,8*x_2*x_2+5,2*x_3*x_3+6,2*x_1*x_2+0,9*x_1*x_3+4,0*x_2*x_3+4,6*x_1*x_2*x_3$

Програмний код

```
from math import fabs
from random import randrange
import numpy as np
from numpy.linalg import solve
from scipy.stats import f, t
from time import perf_counter

counter_1 = 0
counter_2 = 0
counter_3 = 0

for i in range(5):
    m = 3
    n = 15

    # варіант 319
    x1min = -20
    x1max = 30
    x2min = 20
    x2max = 60
    x3min = -20
    x3max = -5

    def function(x1, x2, x3):
        y = 0.5 + 10.0 * x1 + 5.2 * x2 + 6.6 * x3 + 0.4 * x1 * x1 + 0.8 * x2 * x2 + 5.2 * x3 * x3 + 6.2 * x1 * x2 + \
            0.9 * x1 * x3 + 4.0 * x2 * x3 + 4.6 * x1 * x2 * x3 + randrange(0, 10) - 5
        return y

    x01 = (x1max + x1min) / 2
    x02 = (x2max + x2min) / 2
    x03 = (x3max + x3min) / 2
    deltax1 = x1max - x01
    deltax2 = x2max - x02
    deltax3 = x3max - x03
    # матриця ПФЕ
    xn = [[-1, -1, -1, +1, +1, +1, -1, +1, +1, +1],
           [-1, -1, +1, +1, -1, -1, +1, +1, +1, +1],
           [-1, +1, -1, -1, +1, -1, +1, +1, +1, +1],
           [-1, +1, +1, -1, -1, +1, -1, +1, +1, +1],
           [+1, -1, -1, -1, -1, +1, +1, +1, +1, +1],
```

```

[+1, -1, +1, -1, +1, -1, -1, +1, +1, +1],
[+1, +1, -1, +1, -1, -1, -1, +1, +1, +1],
[+1, +1, +1, +1, +1, +1, +1, +1, +1, +1],
[-1.73, 0, 0, 0, 0, 0, 0, 0, 2.9929, 0, 0],
[+1.73, 0, 0, 0, 0, 0, 0, 0, 2.9929, 0, 0],
[0, -1.73, 0, 0, 0, 0, 0, 0, 0, 2.9929, 0],
[0, +1.73, 0, 0, 0, 0, 0, 0, 0, 2.9929, 0],
[0, 0, -1.73, 0, 0, 0, 0, 0, 0, 2.9929],
[0, 0, +1.73, 0, 0, 0, 0, 0, 0, 2.9929],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]]

x1 = [x1min, x1min, x1min, x1min, x1max, x1max, x1max, x1max, -1.73 * deltax1 +
x01, 1.73 * deltax1 + x01, x01, x01,
      x01, x01, x01]
x2 = [x2min, x2min, x2max, x2max, x2min, x2min, x2max, x2max, x02, x02, -1.73 *
deltax2 + x02, 1.73 * deltax2 + x02,
      x02, x02, x02]
x3 = [x3min, x3max, x3min, x3max, x3min, x3max, x3min, x3max, x03, x03, x03, x03, -
1.73 * deltax3 + x03,
      1.73 * deltax3 + x03, x03]
# заповнення нулями x1x2, x1x3, x1x2x3
# заповнення нулями x1kv, x2kv, x3kv
x1x2, x1x3, x2x3, x1x2x3 = [0] * n, [0] * n, [0] * n, [0] * n
x1kv, x2kv, x3kv = [0] * n, [0] * n, [0] * n
for i in range(15):
    x1x2[i] = x1[i] * x2[i]
    x1x3[i] = x1[i] * x3[i]
    x2x3[i] = x2[i] * x3[i]
    x1x2x3[i] = x1[i] * x2[i] * x3[i]
    x1kv[i] = x1[i] ** 2
    x2kv[i] = x2[i] ** 2
    x3kv[i] = x3[i] ** 2
# формуємо список a
list_for_a = list(zip(x1, x2, x3, x1x2, x1x3, x2x3, x1x2x3, x1kv, x2kv, x3kv))

print("Матриця планування з натуралізованими коефіцієнтами X:")
print("      X1      X2      X3      X1X2      X1X3      X2X3
X1X2X3      X1X1"
      "      X2X2      X3X3")
for i in range(n):
    print(end=' ')
    for j in range(len(list_for_a[0])):
        print("{:^12.3f}".format(list_for_a[i][j]), end=' ')
    print("")
# вивід матриці планування
Y = [[function(list_for_a[j][0], list_for_a[j][1], list_for_a[j][2]) for i in
range(m)] for j in range(15)]
print("Матриця планування Y:")
print("      Y1      Y2      Y3")
for i in range(n):
    print(end=' ')
    for j in range(len(Y[0])):
        print("{:^12.3f}".format(Y[i][j]), end=' ')
    print("")
# середні y
Y_average = []
for i in range(len(Y)):
    Y_average.append(np.mean(Y[i], axis=0))
print("Середні значення відгуку за рядками:")
for i in range(n):
    print("{:.3f}".format(Y_average[i]), end=" ")
# розрахунок дисперсій
dispersions = []
for i in range(len(Y)):

```

```

a = 0
for k in Y[i]:
    a += (k - np.mean(Y[i], axis=0)) ** 2
dispersions.append(a / len(Y[i]))

def find_known(num):
    a = 0
    for j in range(n):
        a += Y_average[j] * list_for_a[j][num - 1] / n
    return a

def a(first, second):
    a = 0
    for j in range(n):
        a += list_for_a[j][first - 1] * list_for_a[j][second - 1] / n
    return a

my = sum(Y_average) / n
mx = []
for i in range(10):
    number_lst = []
    for j in range(n):
        number_lst.append(list_for_a[j][i])
    mx.append(sum(number_lst) / len(number_lst))

det1 = [
    [1, mx[0], mx[1], mx[2], mx[3], mx[4], mx[5], mx[6], mx[7], mx[8], mx[9]],
    [mx[0], a(1, 1), a(1, 2), a(1, 3), a(1, 4), a(1, 5), a(1, 6), a(1, 7), a(1, 8),
a(1, 9), a(1, 10)],
    [mx[1], a(2, 1), a(2, 2), a(2, 3), a(2, 4), a(2, 5), a(2, 6), a(2, 7), a(2, 8),
a(2, 9), a(2, 10)],
    [mx[2], a(3, 1), a(3, 2), a(3, 3), a(3, 4), a(3, 5), a(3, 6), a(3, 7), a(3, 8),
a(3, 9), a(3, 10)],
    [mx[3], a(4, 1), a(4, 2), a(4, 3), a(4, 4), a(4, 5), a(4, 6), a(4, 7), a(4, 8),
a(4, 9), a(4, 10)],
    [mx[4], a(5, 1), a(5, 2), a(5, 3), a(5, 4), a(5, 5), a(5, 6), a(5, 7), a(5, 8),
a(5, 9), a(5, 10)],
    [mx[5], a(6, 1), a(6, 2), a(6, 3), a(6, 4), a(6, 5), a(6, 6), a(6, 7), a(6, 8),
a(6, 9), a(6, 10)],
    [mx[6], a(7, 1), a(7, 2), a(7, 3), a(7, 4), a(7, 5), a(7, 6), a(7, 7), a(7, 8),
a(7, 9), a(7, 10)],
    [mx[7], a(8, 1), a(8, 2), a(8, 3), a(8, 4), a(8, 5), a(8, 6), a(8, 7), a(8, 8),
a(8, 9), a(8, 10)],
    [mx[8], a(9, 1), a(9, 2), a(9, 3), a(9, 4), a(9, 5), a(9, 6), a(9, 7), a(9, 8),
a(9, 9), a(9, 10)],
    [mx[9], a(10, 1), a(10, 2), a(10, 3), a(10, 4), a(10, 5), a(10, 6), a(10, 7),
a(10, 8), a(10, 9), a(10, 10)]]

det2 = [my, find_known(1), find_known(2), find_known(3), find_known(4),
find_known(5), find_known(6), find_known(7),
find_known(8), find_known(9), find_known(10)]

beta = solve(det1, det2)
print("\nОтримане рівняння регресії:")
print("{:.3f} + {:.3f} * X1 + {:.3f} * X2 + {:.3f} * X3 + {:.3f} * X1X2 + {:.3f} *
X1X3 + {:.3f} * X2X3"
      + {:.3f} * X1X2X3 + {:.3f} * X11^2 + {:.3f} * X22^2 + {:.3f} * X33^2 = ŷ"
      .format(beta[0], beta[1], beta[2], beta[3], beta[4], beta[5], beta[6],
beta[7], beta[8], beta[9], beta[10]))
y_i = [0] * n
print("Експериментальні значення:")

```

```

    for k in range(n):
        y_i[k] = beta[0] + beta[1] * list_for_a[k][0] + beta[2] * list_for_a[k][1] +
        beta[3] * list_for_a[k][2] + \
            beta[4] * list_for_a[k][3] + beta[5] * list_for_a[k][4] + beta[6] *
list_for_a[k][5] + beta[7] * \
            list_for_a[k][6] + beta[8] * list_for_a[k][7] + beta[9] *
list_for_a[k][8] + beta[10] * list_for_a[k][
            9]

    for i in range(n):
        print("{:.3f}".format(y_i[i]), end=" ")
    print("\n----- Перевірка за критерієм Кохрена -----")
    start_1 = perf_counter()
    Gp = max(dispersions) / sum(dispersions)
    Gt = 0.3346
    print("Gp =", Gp)
    if Gp < Gt:
        print("Дисперсія однорідна")
    else:
        print("Дисперсія неоднорідна")

    counter_1 += perf_counter() - start_1

    print("----- Перевірка значущості коефіцієнтів за критерієм Стюдента")
    sb = sum(dispersions) / len(dispersions)
    sbs = (sb / (n * m)) ** 0.5
    F3 = (m - 1) * n
    start_2 = perf_counter()
    coefs1 = []
    coefs2 = []
    d = 11
    res = [0] * 11
    for j in range(11):
        t_pract = 0
        for i in range(15):
            if j == 0:
                t_pract += Y_average[i] / 15
            else:
                t_pract += Y_average[i] * xn[i][j - 1]
            res[j] = beta[j]
        if fabs(t_pract / sbs) < t.ppf(q=0.975, df=F3):
            coefs2.append(beta[j])
            res[j] = 0
            d -= 1
        else:
            coefs1.append(beta[j])
    counter_2 += perf_counter() - start_2
    print("Значущі коефіцієнти регресії:", [round(i, 3) for i in coefs1])
    print("Незначущі коефіцієнти регресії:", [round(i, 3) for i in coefs2])
    y_st = []
    for i in range(n):
        y_st.append(res[0] + res[1] * x1[i] + res[2] * x2[i] + res[3] * x3[i] + res[4]
* x1x2[i] + res[5] *
            x1x3[i] + res[6] * x2x3[i] + res[7] * x1x2x3[i] + res[8] * x1kv[i]
+ res[9] *
            x2kv[i] + res[10] * x3kv[i])
    print("Значення з отриманими коефіцієнтами:")
    for i in range(n):
        print("{:.3f}".format(y_st[i]), end=" ")

    print("\n----- Перевірка адекватності за критерієм Фішера -----")

```

```

-----")
Sad = m * sum([(y_st[i] - Y_average[i]) ** 2 for i in range(n)]) / (n - d)
Fp = Sad / sb
F4 = n - d
print("Fp =", Fp)
start_3 = 0
if Fp < f.ppf(q=0.95, dfn=F4, dfd=F3):
    print("Рівняння регресії адекватне при рівні значимості 0.05")
else:
    print("Рівняння регресії неадекватне при рівні значимості 0.05")
counter_3 += perf_counter() - start_3

```

Результати роботи програми

Матриця планування з натуралізованими коефіцієнтами X:

X1	X2	X3	X1X2	X1X3	X2X3	X1X2X3	X1X1	X2X2	X3X3
-20.000	20.000	-20.000	-400.000	400.000	-400.000	8000.000	400.000	400.000	400.000
-20.000	20.000	-5.000	-400.000	100.000	-100.000	2000.000	400.000	400.000	25.000
-20.000	60.000	-20.000	-1200.000	400.000	-1200.000	24000.000	400.000	3600.000	400.000
-20.000	60.000	-5.000	-1200.000	100.000	-300.000	6000.000	400.000	3600.000	25.000
30.000	20.000	-20.000	600.000	-600.000	-400.000	-12000.000	900.000	400.000	400.000
30.000	20.000	-5.000	600.000	-150.000	-100.000	-3000.000	900.000	400.000	25.000
30.000	60.000	-20.000	1800.000	-600.000	-1200.000	-36000.000	900.000	3600.000	400.000
30.000	60.000	-5.000	1800.000	-150.000	-300.000	-9000.000	900.000	3600.000	25.000
-38.250	40.000	-12.500	-1530.000	478.125	-500.000	19125.000	1463.062	1600.000	156.250
48.250	40.000	-12.500	1930.000	-603.125	-500.000	-24125.000	2328.062	1600.000	156.250
5.000	5.400	-12.500	27.000	-62.500	-67.500	-337.500	25.000	29.160	156.250
5.000	74.600	-12.500	373.000	-62.500	-932.500	-4662.500	25.000	5565.160	156.250
5.000	40.000	-25.475	200.000	-127.375	-1019.000	-5095.000	25.000	1600.000	648.976
5.000	40.000	0.475	200.000	2.375	19.000	95.000	25.000	1600.000	0.226
5.000	40.000	-12.500	200.000	-62.500	-500.000	-2500.000	25.000	1600.000	156.250

Матриця планування Y:

Y1	Y2	Y3
35410.500	35413.500	35411.500
6889.500	6888.500	6895.500
103623.500	103623.500	103616.500
22300.500	22294.500	22294.500
-50585.500	-50589.500	-50589.500
-9438.500	-9431.500	-9437.500
-153977.500	-153975.500	-153975.500
-27627.500	-27625.500	-27624.500
79342.538	79335.538	79344.538
-97922.587	-97923.587	-97915.587
-872.442	-870.442	-872.442
-17287.602	-17291.602	-17286.602
-21628.599	-21631.599	-21635.599
3307.946	3305.946	3310.946
-10036.750	-10034.750	-10037.750

```
Середні значення відгуку за рядками:
35411.833 6891.167 103621.167 22296.500 -50588.167 -9435.833 -153976.167 -27625.833 79340.871 -97920.587 -871.775 -17288.602 -21631.933 3308.279 -10036.417
Отримане рівняння регресії:
-1.248 + 9.890 * X1 + 5.284 * X2 + 6.573 * X3 + 6.203 * X1X2 + 0.896 * X1X3 + 3.995 * X2X3+ 4.600 * X1X2X3 + 0.399 * X11^2 + 0.799 * X22^2 + 5.194 * X33^2 = ŷ
Експериментальні значення:
35411.645 6891.823 103621.551 22297.729 -50589.042 -9435.864 -153976.469 -27625.291 79339.841 -97920.030 -871.349 -17289.500 -21631.192 3307.066 -10036.413
----- Перевірка за критерієм Кохрена -----
Gr = 0.16067146282973624
Дисперсія однорідна
----- Перевірка значущості коефіцієнтів за критерієм Стьюдента -----
Значущі коефіцієнти регресії: [-1.248, 9.89, 5.284, 6.573, 6.203, 0.896, 3.995, 4.6, 0.399, 0.799, 5.194]
Незначущі коефіцієнти регресії: []
Значення з отриманими коефіцієнтами:
35411.645 6891.823 103621.551 22297.729 -50589.042 -9435.864 -153976.469 -27625.291 79339.841 -97920.030 -871.349 -17289.500 -21631.192 3307.066 -10036.413
----- Перевірка адекватності за критерієм Фішера -----
Fr = 0.929370417626661
Рівняння регресії адекватне при рівні значимості 0.05
Матриця планування з натуралізованими коефіцієнтами X:
X1      X2      X3      X1X2     X1X3     X2X3     X1X2X3    X1X1    X2X2    X3X3
-20.000  20.000  -20.000  -400.000  400.000  -400.000  8000.000  400.000  400.000  400.000
-20.000  20.000   -5.000  -400.000  100.000  -100.000  2000.000  400.000  400.000   25.000
-20.000  60.000  -20.000  -1200.000  400.000  -1200.000  24000.000  400.000  3600.000  400.000
-20.000  60.000   -5.000  -1200.000  100.000  -300.000  6000.000  400.000  3600.000  25.000
 30.000  20.000  -20.000   600.000  -600.000  -400.000 -12000.000  900.000  400.000  400.000
 30.000  20.000   -5.000   600.000  -150.000  -100.000  -3000.000  900.000  400.000   25.000
 30.000  60.000  -20.000  1800.000  -600.000  -1200.000 -36000.000  900.000  3600.000  400.000
 30.000  60.000   -5.000  1800.000  -150.000  -300.000  -9000.000  900.000  3600.000   25.000
-38.250  40.000  -12.500 -1530.000  478.125  -500.000  19125.000  1463.062  1600.000  156.250
 48.250  40.000  -12.500  1930.000  -603.125  -500.000 -24125.000  2328.062  1600.000  156.250
  5.000   5.400  -12.500   27.000  -62.500  -67.500  -337.500   25.000   29.160  156.250
  5.000  74.600  -12.500  373.000  -62.500  -932.500 -4662.500   25.000  5565.160  156.250
  5.000  40.000  -25.475  200.000  -127.375 -1019.000 -5095.000   25.000  1600.000  648.976
  5.000  40.000   0.475  200.000   2.375   19.000   95.000   25.000  1600.000   0.226
  5.000  40.000  -12.500  200.000  -62.500  -500.000 -2500.000   25.000  1600.000  156.250
Матриця планування Y:
Y1      Y2      Y3
35410.500  35413.500  35411.500
 6895.500   6894.500   6888.500
103623.500 103616.500 103616.500
22294.500  22300.500  22302.500
-50591.500 -50587.500 -50584.500
-9430.500  -9435.500  -9430.500
-153982.500 -153982.500 -153976.500
-27628.500  -27630.500  -27622.500
 79337.538  79344.538  79337.538
-97920.587 -97919.587 -97920.587
 -869.442  -873.442  -872.442
-17293.602 -17294.602 -17290.602
-21629.599 -21636.599 -21636.599
 3303.946   3307.946   3305.946
-10035.750 -10038.750 -10036.750
Середні значення відгуку за рядками:
35411.833 6892.833 103618.833 22299.167 -50587.833 -9432.167 -153980.500 -27627.167 79339.871 -97920.254 -871.775 -17292.935 -21634.266 3305.946 -10037.083
Отримане рівняння регресії:
-1.036 + 10.021 * X1 + 5.303 * X2 + 6.468 * X3 + 6.199 * X1X2 + 0.901 * X1X3 + 3.998 * X2X3+ 4.600 * X1X2X3 + 0.400 * X11^2 + 0.798 * X22^2 + 5.189 * X33^2 = ŷ
Експериментальні значення:
35411.219 6891.673 103618.897 22298.684 -50588.412 -9433.291 -153980.401 -27627.614 79340.621 -97919.586 -870.283 -17293.010 -21634.188 3307.286 -10037.093
----- Перевірка за критерієм Кохрена -----
Gr = 0.11685393258426968
Дисперсія однорідна
----- Перевірка значущості коефіцієнтів за критерієм Стьюдента -----
Значущі коефіцієнти регресії: [-1.036, 10.021, 5.303, 6.468, 6.199, 0.901, 3.998, 4.6, 0.4, 0.798, 5.189]
Незначущі коефіцієнти регресії: []
Значення з отриманими коефіцієнтами:
35411.219 6891.673 103618.897 22298.684 -50588.412 -9433.291 -153980.401 -27627.614 79340.621 -97919.586 -870.283 -17293.010 -21634.188 3307.286 -10037.093
----- Перевірка адекватності за критерієм Фішера -----
Fr = 1.002922334055666
Рівняння регресії адекватне при рівні значимості 0.05

Process finished with exit code 0
```

Висновок: під час виконання лабораторної роботи було проведено повний трьохфакторний експеримент і отримати адекватну модель – рівняння регресії, використовуючи рототабельний композиційний план. Лабораторна робота виконана, кінцева мета досягнута.