

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
Факультет інформатики та обчислювальної техніки

Лабораторна робота 2

ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ

Гуменюк Святослав
4 курс
ІК-92

Викладач:
Кир'янов Артемій Юрійович

1 Лістинг програми

```
import random as rn
import statistics as stat
import math
import itertools
import numpy as np

Nvariant = 115
ymax = (30-Nvariant)*10
ymin = (20-Nvariant)*10
xlmin = 10
xlmax = 50
x2min = -20
x2max = 60

def normalized_x(x):
    x0 = [0.5*(max(x[i])+min(x[i])) for i in range(len(x))]
    dx = [x0[i]-min(x[i]) for i in range(len(x))]
    return [[round((x[i][j]-x0[i])/dx[i],2) for j in range(len(x[i]))) \
    for i in range(len(x))]

def F(sigmas):
    return sigmas[0]/sigmas[1] if sigmas[0] >= sigmas[1] else \
    sigmas[1]/sigmas[0]

def romanovskiy_criteria(yy):
    ymean = [stat.mean(yy[i]) for i in range(len(yy))]
    print("y_mean_i_=",*ymean)
    ysigma = [stat.pstdev(yy[i]) for i in range(len(yy))]
    m = len(yy[0])
    sigmaTheta = math.sqrt(2*(2*m-2)/m/(m-4))
    thetas = [(1-2/m)*F(sigmas) for sigmas in \
    list(itertools.combinations(ysigma,2))]
    R = [math.fabs(theta-1)/sigmaTheta for theta in thetas]
    return R

def b(x1,x2,y):
    meanY = [stat.mean(y[i]) for i in range(len(y))]
    mx1 = stat.mean(x1)
    mx2 = stat.mean(x2)
    my = (stat.mean(y[0])+stat.mean(y[1])+stat.mean(y[2]))/3.
    a1 = np.dot(x1,x1)/len(x1)
    a2 = np.dot(x1,x2)/len(x1)
    a3 = np.dot(x2,x2)/len(x2)
```

```

a11 = np.dot(x1,meanY)/len(x1)
a22 = np.dot(x2,meanY)/len(x2)

b0 = np.linalg.det([[my,mx1,mx2], [a11,a1,a2], [a22,a2,a3]])/ \
np.linalg.det([[1,mx1,mx2], [mx1,a1,a2], [mx2,a2,a3]])
b1 = np.linalg.det([[1,my,mx2], [mx1,a11,a2], [mx2,a22,a3]])/ \
np.linalg.det([[1,mx1,mx2], [mx1,a1,a2], [mx2,a2,a3]])
b2 = np.linalg.det([[1,mx1,my], [mx1,a1,a11], [mx2,a2,a22]])/ \
np.linalg.det([[1,mx1,mx2], [mx1,a1,a2], [mx2,a2,a3]])
return [b0,b1,b2]

def a(B):
    deltaX1 = math.fabs(x1max-x1min)/2.
    deltaX2 = math.fabs(x2max-x2min)/2.
    x01 = (x1min+x1max)/2.
    x02 = (x2min+x2max)/2.
    a0 = B[0]-B[1]*x01/deltaX1-B[2]*x02/deltaX2
    a1 = B[1]/deltaX1
    a2 = B[2]/deltaX2
    return [a0,a1,a2]

if __name__ == '__main__':
    rn.seed(228)
    m = 5
    Rcr = 2
    heterogeneous_dispersion = True
    while heterogeneous_dispersion:
        Yy = []
        for i in range(m):
            Yy.append([rn.choice(range(ymin,ymax)) for k in range(3)])
            if all(r is r<Rcr for r in romanovskiy_criteria([*zip(*Yy)])):
                break
        m+=1

    print("Ruv_i=", *romanovskiy_criteria([*zip(*Yy)]))

    X1 = [rn.choice(range(x1min,x1max)) for i in range(3)]
    X2 = [rn.choice(range(x2min,x2max)) for i in range(3)]
    XN = normalized_x([X1,X2])

    print("b_i=",*b(XN[0],XN[1],[*zip(*Yy)]))
    print("a_i=",*a(b(XN[0],XN[1],[*zip(*Yy)])))

    for row in zip(*(XN+Yy)):
        print("_".join(str(row)))

```

2 Результат роботи програми

Нормована матриця планування:

	X_{N1}	X_{N2}	Y_1	Y_2	Y_3	Y_4	Y_5
1	1.0	1.0	-854	-922	-922	-873	-851
2	-1.0	-1.0	-861	-877	-891	-861	-932
3	-0.2	0.08	-866	-922	-878	-892	-941

Критерій Романовського показав наступні значення для R_{uv} :

$$R_{uv1} = 0.16, R_{uv2} = 0.18, R_{uv3} = 0.20$$

Кожне з вищезгаданих значень менше за $R_{cr} = 2$, а отже дисперсія однорідна.

Далі було знайдено нормовані коефіцієнти рівняння:

$$y = -884.39 + 48.12 \cdot x_{n1} - 48.12 \cdot x_{n2}$$

Після чого тривіальною підстановкою нормованих значень в дане рівняння, було перевірено рівність отриманих значень та середніх значень для i -ї точки.

Останнім кроком було знаходження натуралізованих коефіцієнтів:

$$y = -932.52 + 2.41 \cdot x_1 - 1.20 \cdot x_2$$

Після чого перевірка знову ж показала правильність знайдених коефіцієнтів.

3 Висновки

В ході даної лабораторної роботи було проведено двофакторний експеримент з використанням лінійного рівняння регресії. Було перевірено однорідність дисперсії за критерієм Романовського після чого було знайдено нормовані та натуральні коефіцієнти рівняння та перевірено їх правильність за допомогою середніх значень