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

Лабораторна робота 3

ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО 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

x1min = 10
x1max = 50
x2min = -20
x2max = 60
x3min = -20
x3max = 20
xMeanMax = (x1max+x2max+x3max)/3.
xMeanMin = (x1min+x2min+x3min)/3.
ymin = 200+int(xMeanMin)
ymax = 200+int(xMeanMax)

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 cross_sum(a,b):
    sum_ = 0
    for i in range(len(a)):
        sum_+=a[i]*b[i]
    sum_/=len(a)
    return sum_

def check(x,b,y):
    return [[b[0]+b[1]*x[0][i]+b[2]*x[1][i]+b[3]*x[2][i] for i in range(len(x[0]

def B(x1,x2,x3,y):
    meanY = [stat.mean(y[i]) for i in range(len(y))]
    my = stat.mean(meanY)
    mx1 = stat.mean(x1)
    mx2 = stat.mean(x2)
    mx3 = stat.mean(x3)
    a1 = cross_sum(x1,meanY)
    a2 = cross_sum(x2,meanY)
    a3 = cross_sum(x3,meanY)
    a11 = cross_sum(x1,x1)
```

```
    a22 = cross_sum(x2,x2)
    a33 = cross_sum(x3,x3)
    a12 = cross_sum(x1,x2)
    a21 = a12
    a13 = cross_sum(x1,x3)
    a31 = a13
    a23 = cross_sum(x2,x3)
    a32 = a23

    b0 = np.linalg.det([[my,mx1,mx2,mx3], [a1,a11,a12,a13], [a2,a12,a22,a32], \
    [a3,a13,a23,a33]])/np.linalg.det([[1,mx1,mx2,mx3], [mx1,a11,a12,a13], \
    [mx2,a12,a22,a32], [mx3,a13,a23,a33]])
    b1 = np.linalg.det([[1,my,mx2,mx3], [mx1,a1,a12,a13], [mx2,a2,a22,a32], \
    [mx3,a3,a23,a33]])/np.linalg.det([[1,mx1,mx2,mx3], [mx1,a11,a12,a13], \
    [mx2,a12,a22,a32], [mx3,a13,a23,a33]])
    b2 = np.linalg.det([[1,mx1,my,mx3], [mx1,a11,a1,a13], [mx2,a12,a2,a32], \
    [mx3,a13,a3,a33]])/np.linalg.det([[1,mx1,mx2,mx3], [mx1,a11,a12,a13], \
    [mx2,a12,a22,a32], [mx3,a13,a23,a33]])
    b3 = np.linalg.det([[1,mx1,mx2,my], [mx1,a11,a12,a1], [mx2,a12,a22,a2], \
    [mx3,a13,a23,a3]])/np.linalg.det([[1,mx1,mx2,mx3], [mx1,a11,a12,a13], \
    [mx2,a12,a22,a32], [mx3,a13,a23,a33]])
    return [b0,b1,b2,b3]

def A(x1,x2,x3,y):
    meanY = [stat.mean(y[i]) for i in range(len(y))]
    a0 = stat.mean(meanY)
    a1 = np.dot(meanY,x1)/len(x1)
    a2 = np.dot(meanY,x2)/len(x2)
    a3 = np.dot(meanY,x3)/len(x3)
    return [a0,a1,a2,a3]

def kohren_criteria(y):
    s = [stat.pvariance(y[i]) for i in range(len(y))]
    return max(s)/sum(s)

def student_criteria(x,y,text=False):
    tCr = 2.306
    meanY = [stat.mean(y[i]) for i in range(len(y))]
    s = [stat.pvariance(y[i]) for i in range(len(y))]
    sb = stat.mean(s)
    sBetaS = sb/len(y[0])/len(s)
    sBetaS = math.sqrt(sBetaS)
    t0 = abs(cross_sum(meanY,x[0]))/sBetaS
    t1 = abs(cross_sum(meanY,x[1]))/sBetaS
    t2 = abs(cross_sum(meanY,x[2]))/sBetaS
    t3 = abs(cross_sum(meanY,x[3]))/sBetaS
```

```
    if text==True:
        print('Student criteria =', *[t0,t1,t2,t3])
    return [1 if t>tCr else 0 for t in [t0,t1,t2,t3]]

def fisher_criteria(y,B,x1,x2,x3,sign,text=False):
    d = 0
    for i in sign:
        d+=i
    meanY = [stat.mean(y[i]) for i in range(len(y))]
    newY = [B[0]*sign[0]+B[1]*sign[1]*x1[i]+B[2]*sign[2]*x2[i]+ \
    B[3]*sign[3]*x3[i] for i in range(len(x1))]
    diff = [(newY[i]-meanY[i])**2 for i in range(len(newY))]
    try:
        Sad = len(y[0])/(len(y)-d)*sum(diff)
    except ZeroDivisionError:
        return False
    s = [stat.pvariance(y[i]) for i in range(len(y))]
    sb = stat.mean(s)

    if Sad/sb < 4.5 :
        if text==True:
            print('Fisher criteria =',Sad/sb)
        return True
    else:
        return False

if __name__ == '__main__':
    X0 = [1 for k in range(5)]
    X1 = [rn.choice(range(x1min,x1max)) for k in range(5)]
    X2 = [rn.choice(range(x2min,x2max)) for k in range(5)]
    X3 = [rn.choice(range(x3min,x3max)) for k in range(5)]
    XN = normalized_x([X1,X2,X3])

    Seed = 1
    while True:
        rn.seed(Seed)
        m = 3
        Gt = 0.7679
        while True:
            Yy = []
            for i in range(m):
                Yy.append([rn.choice(range(ymin,ymax)) for k in range(5)])
            if kohren_criteria([*zip(*Yy)])<Gt:
                break
            m+=1
        if fisher_criteria([*zip(*Yy)],B(X1,X2,X3,[*zip(*Yy)]),\
```

```
    X1,X2,X3,student_criteria([X0,*XN],[*zip(*Yy)])):
        break
    Seed+=1


print("Kohren criteria =",kohren_criteria([*zip(*Yy)]))
print("b_i =",*B(X1,X2,X3,[*zip(*Yy)]))
print("a_i =",*A(X1,X2,X3,[*zip(*Yy)]))
print(student_criteria([X0,*XN],[*zip(*Yy)],True))
print(fisher_criteria([*zip(*Yy)],B(X1,X2,X3,[*zip(*Yy)]),\
X1,X2,X3,student_criteria([X0,*XN],[*zip(*Yy)]),True))

for row in zip(*(XN+Yy)):
    print(' & '.join(map(str,row)), end='\\\\\n')
```

2 Результат роботи програми

Нормована матриця планування:

	X_{N1}	X_{N2}	X_{N3}	Y_1	Y_2	Y_3
1	-0.62	0.26	0.38	193	237	228
2	-0.05	-0.47	1.0	195	241	203
3	0.43	1.0	-0.46	195	232	228
4	-1.0	-1.0	1.0	213	209	192
5	1.0	-0.5	-1.0	200	206	227

Далі було знайдено нормовані коефіцієнти рівняння:

$$y = 207.45 + 0.12 \cdot x_{n1} + 0.21 \cdot x_{n2} + 0.12 \cdot x_{n3}$$

Критерій Кохрена показав наступне значення:

$$G_p = 0.736$$

Вищезгадане значення менше за $G_T = 0.321$, а отже дисперсія однорідна.

Критерій Стьюдента показав наступні значення:

$$t_0 = 52.12,\ t_1 = 2.26,\ t_2 = 6.59,\ t_3 = 9.27$$

Значення $t_1 < 2.306$, тому коефіцієнт рівняння регресії приймаємо незначними при рівні значимості 0.05

Таким чином рівняння регресії має вигляд:

$$y = 207.45 + 0.21 \cdot x_{n2} + 0.34 \cdot x_{n3}$$

Було перевірка адекватносі моделі за допомогою критерію Фішера:

$$F_p = 0.65$$

Оскільки $F_p < 4.5$, отже рівняння регресії адекватно оригіналу при рівні значимості 0.05

Також було знайдено натуралізовані коефіцієнти рівнянн регресії:

$$y = 213.26 + 2501.0 \cdot x_2 - 784.93 \cdot x_3$$

3 Висновки

В ході даної лабораторної роботи було проведено трьохфакторний експеримент з використанням лінійного рівняння регресії. Було перевірено однорідність десперсії за критерієм Кохрена. Після цього було знайдено натуралізовані та нормалізовані коефіцієнти, та визначено значимість коефіцієнтів за допомогою критерію Стьюдента, який показав, що один з коефіцієнтів є незначним. Адекватность рівняння оригіналу було перевірено за допомогою критерію Фішера, який показав, що рівнняння є адекватним оригіналу.