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 №3

з дисципліни « Методи оптимізації та планування » на тему

ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ

ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ.

Виконав:

студент II курсу ФІОТ

групи ІВ – 93

Гулак Іван

номер в списку - 04

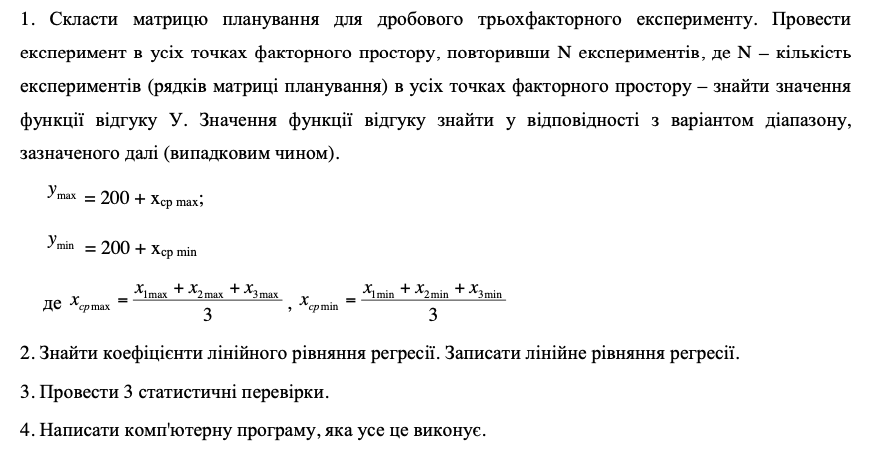
Перевірив:

ас. Регіда П.Г.

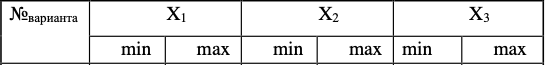
Київ – 2021

**Мета роботи:**  провести дробовий трьохфакторний експеримент. Скласти матрицю планування, знайти коефіцієнти рівняння регресії, провести 3 статистичні перевірки.

**Завдання на лабораторну роботу:**

****

**Варіант завдання:**





**Роздруківка тексту програми:**

import random  
import math  
  
# Лабораторна робота №3  
# Виконав Гулак Іван Володимирович Варіант - 304  
# Вхідні дані  
x1\_min = 15  
x1\_max = 45  
x2\_min = 30  
x2\_max = 80  
x3\_min = 15  
x3\_max = 45  
  
gt = 0.7679  
m = 3  
N = 4  
Tf = 2.306  
Ft = 4.5  
  
y\_max = 200 + (x1\_max + x2\_max + x3\_max) / 3  
y\_min = 200 + (x1\_min + x2\_min + x3\_min) / 3  
  
# Заповнимо матрицю планування для m = 3.  
x1 = [random.randint(x1\_min, x1\_max + 1) for i in range(4)]  
x2 = [random.randint(x2\_min, x2\_max + 1) for i in range(4)]  
x3 = [random.randint(x3\_min, x3\_max + 1) for i in range(4)]  
y1 = [random.randint(int(y\_min), int(y\_max) + 1) for i in range(4)]  
y2 = [random.randint(int(y\_min), int(y\_max) + 1) for i in range(4)]  
y3 = [random.randint(int(y\_min), int(y\_max) + 1) for i in range(4)]  
  
plan\_matrix = [[1, 1, 1, 1],  
 [-1, -1, 1, 1],  
 [-1, 1, -1, 1],  
 [-1, 1, 1, -1]]  
  
# 4) Знайдемо середні значення функції відгуку за рядками:  
averageY = [0, 0, 0, 0]  
for i in range(0, len(x1)):  
 averageY[i] = (y1[i] + y2[i] + y3[i]) / 3  
print(f"average\_Y: {averageY}")  
  
  
# Функції для обчислення mx1, mx2, mx3, my, a1, a2, a3, a11, a22, a33, a12 = a21, a13 = a31, a23 = a32  
def func\_mx(arr, main\_arr):  
 main\_arr.append(sum(arr) / len(arr))  
  
  
def func\_a(arr, main\_arr):  
 main\_arr.append((arr[0] \* averageY[0] + arr[1] \* averageY[1] + arr[2] \* averageY[2] + arr[3] \* averageY[3])/len(arr))  
  
  
def func\_aii(arr, main\_arr):  
 main\_arr.append((arr[0]\*\*2 + arr[1]\*\*2 + arr[2]\*\*2 + arr[3]\*\*2) / len(arr))  
  
  
def func\_aij(arr1, arr2, main\_arr):  
 main\_arr.append((arr1[0] \* arr2[0] + arr1[1] \* arr2[1] + arr1[2] \* arr2[2] + arr1[3] \* arr2[3])/len(arr1))  
  
  
mx = []  
func\_mx(x1, mx)  
func\_mx(x2, mx)  
func\_mx(x3, mx)  
print(f"MX: {mx}")  
  
my = (averageY[0] + averageY[1] + averageY[2] + averageY[3]) / len(averageY)  
print(f"MY: {my}")  
  
a = []  
func\_a(x1, a)  
func\_a(x2, a)  
func\_a(x3, a)  
print(f"A: {a}")  
  
a11 = []  
func\_aii(x1, a11)  
print(f"A11: {a11}")  
  
a22 = []  
func\_aii(x2, a22)  
print(f"A22: {a22}")  
  
a33 = []  
func\_aii(x3, a33)  
print(f"A33: {a33}")  
  
a12 = a21 = []  
func\_aij(x1, x2, a12)  
print(f"A12 = A21: {a12}")  
  
a13 = a31 = []  
func\_aij(x1, x3, a13)  
print(f"A13 = A31: {a13}")  
  
a23 = a32 = []  
func\_aij(x2, x3, a23)  
print(f"A23 = A32: {a23}")  
  
r01 = [1, mx[0], mx[1], mx[2]]  
r02 = [mx[0], a11[0], a12[0], a13[0]]  
r03 = [mx[1], a12[0], a22[0], a32[0]]  
r04 = [mx[2], a13[0], a23[0], a33[0]]  
temp0 = [r01, r02, r03, r04]  
determ = 1\*a11[0]\*a22[0]\*a33[0] + mx[0]\*a12[0]\*a32[0]\*mx[2] + mx[1]\*a13[0]\*mx[1]\*a13[0] + mx[2]\*mx[0]\*a12[0]\*a23[0] - \  
 (mx[2]\*a12[0]\*a12[0]\*mx[2] + a13[0]\*a22[0]\*a13[0]\*1 + a23[0]\*a32[0]\*mx[0]\*mx[0] + a33[0]\*mx[1]\*a11[0]\*mx[1])  
  
  
r11 = [my, mx[0], mx[1], mx[2]]  
r12 = [a[0], a11[0], a12[0], a13[0]]  
r13 = [a[1], a12[0], a22[0], a32[0]]  
r14 = [a[2], a13[0], a23[0], a33[0]]  
temp1 = [r11, r12[0], r13[0], r14[0]]  
determ1 = my\*a11[0]\*a22[0]\*a33[0] + a[0]\*a12[0]\*a32[0]\*mx[2] + a[1]\*a13[0]\*mx[1]\*a13[0] + a[2]\*mx[0]\*a12[0]\*a23[0] - \  
 (a[2]\*a12[0]\*a12[0]\*mx[2] + a13[0]\*a22[0]\*a13[0]\*my + a23[0]\*a32[0]\*a[0]\*mx[0] + a33[0]\*a[1]\*a11[0]\*mx[1])  
  
  
r21 = [1, my, mx[1], mx[2]]  
r22 = [mx[0], a[0], a12[0], a13[0]]  
r23 = [mx[1], a[1], a22[0], a32[0]]  
r24 = [mx[2], a[2], a23[0], a33[0]]  
temp2 = [r21, r22, r23, r24]  
determ2 = 1\*a[0]\*a22[0]\*a33[0] + my\*a12[0]\*a32[0]\*mx[2] + mx[1]\*a13[0]\*mx[1]\*a[2] + mx[2]\*mx[0]\*a[1]\*a23[0] - \  
 (mx[2]\*a[1]\*a12[0]\*mx[2] + a[2]\*a22[0]\*a13[0]\*1 + a23[0]\*a32[0]\*my\*mx[0] + a33[0]\*mx[1]\*a[0]\*mx[1])  
  
  
r31 = [1, mx[0], my, mx[2]]  
r32 = [mx[0], a11[0], a[0], a13[0]]  
r33 = [mx[1], a12[0], a[1], a32[0]]  
r34 = [mx[2], a13[0], a[2], a33[0]]  
temp3 = [r31, r32, r33, r34]  
determ3 = 1\*a11[0]\*a[1]\*a33[0] + mx[0]\*a[0]\*a32[0]\*mx[2] + my\*a13[0]\*mx[1]\*a13[0] + mx[2]\*mx[0]\*a12[0]\*a[2] - \  
 (mx[2]\*a12[0]\*a[0]\*mx[2] + a13[0]\*a[1]\*a13[0]\*1 + a[2]\*a32[0]\*mx[0]\*mx[0] + a33[0]\*mx[1]\*a11[0]\*my)  
  
  
r41 = [1, mx[0], mx[1], my]  
r42 = [mx[0], a11[0], a12[0], a[0]]  
r43 = [mx[1], a12[0], a22[0], a[1]]  
r44 = [mx[2], a13[0], a23[0], a[2]]  
temp4 = [r41, r42, r43, r44]  
determ4 = 1\*a11[0]\*a22[0]\*a[2] + mx[0]\*a12[0]\*a[1]\*mx[2] + mx[1]\*a[0]\*mx[1]\*a13[0] + my\*mx[0]\*a12[0]\*a23[0] - \  
 (mx[2]\*a12[0]\*a12[0]\*my + a13[0]\*a22[0]\*a[0]\*1 + a23[0]\*a[1]\*mx[0]\*mx[0] + a[2]\*mx[1]\*a11[0]\*mx[1])  
  
  
b = [0, 0, 0, 0]  
b[0] = determ1/determ  
b[1] = determ2/determ  
b[2] = determ3/determ  
b[3] = determ4/determ  
  
print("-" \* 50)  
for i in range(4):  
 print(f"b{i}: {b[i]}")  
print("-" \* 50)  
  
# Запишемо отримане рівняння регресії:  
y = []  
for i in range(len(x1)):  
 y.append(b[0] + b[1] \* x1[i] + b[2] \* x2[i] + b[3] \* x3[i])  
 print(f"Y{i+1}: {y[i]}")  
  
  
dispersiya = [0, 0, 0, 0]  
for i in range(0, len(dispersiya)):  
 dispersiya[i] = ((y1[i] - averageY[i])\*\*2 + (y2[i] - averageY[i])\*\*2 + (y3[i] - averageY[i])\*\*2)/3  
print(f"Dispersiya: {dispersiya}")  
  
print("-" \* 80)  
print('Перевірка однорідності дисперсії за критерієм Кохрена:')  
gp = max(dispersiya) / sum(dispersiya)  
if gp < gt:  
 print("Дисперсія однорідна")  
else:  
 print('Дсперсія не однорідна. Потрібно збільшити m')  
  
  
# Далі оцінимо значимість коефіцієнтів регресії згідно критерію Стьюдента  
sb = sum(dispersiya) / len(dispersiya)  
sb2\_odnoridna = sb / (N \* m)  
sb\_odnoridna = math.sqrt(sb2\_odnoridna)  
  
beta = [0, 0, 0, 0]  
beta[0] = (averageY[0] \* 1 + averageY[1] \* 1 + averageY[2] \* 1 + averageY[3] \* 1) / N  
beta[1] = (averageY[0] \* (-1) + averageY[1] \* (-1) + averageY[2] \* 1 + averageY[3] \* 1) / N  
beta[2] = (averageY[0] \* (-1) + averageY[1] \* 1 + averageY[2] \* (-1) + averageY[3] \* 1) / N  
beta[3] = (averageY[0] \* (-1) + averageY[1] \* 1 + averageY[2] \* 1 + averageY[3] \* (-1)) / N  
print(f"Beta: {beta}")  
  
t = []  
for i in range(len(beta)):  
 t.append(abs(beta[i]) / sb\_odnoridna)  
print(f"t0: {t}")  
  
print("-" \* 80)  
print('\nОцінка значимості коефіцієнтів регресії згідно критерію Стьюдента:')  
d = 0 # кількість значимих коефіцієнтів  
temp = [0, 0, 0, 0]  
for i in range(0, N):  
 if t[i] <= Tf:  
 print(f"t[{i}] = {t[i]} <= Tf = {Tf} >= b[{i}] = {b[i]} - не значний коефіцієнт")  
 temp[i] = 0  
 else:  
 print(f"t[{i}] = {t[i]} > Tf = {Tf} >= b[{i}] = {b[i]} - значний коефіцієнт")  
 temp[i] = b[i]  
 d += 1  
  
y\_2 = []  
for i in range(0, N):  
 y\_2.append(temp[0] + temp[1] \* x1[i] + temp[2] \* x2[i] + temp[3] \* x3[i])  
print(y\_2)  
  
  
# Критерій Фішера  
print("-" \* 80)  
print("Критерій Фішера")  
s\_ad = []  
tmp = []  
kof = m / (N - d)  
for i in range(len(y\_2)):  
 tmp.append((y\_2[i] - averageY[i])\*\*2)  
s\_ad.append(kof \* sum(tmp))  
print(f"S\_ad: {s\_ad}")  
  
fp = s\_ad[0] / sb2\_odnoridna  
print(f"Fp: {fp}")  
if fp > Ft:  
 print("рівняння регресії неадекватно оригіналу при рівні значимості 0.05")  
else:  
 print("рівняння регресії адекватно оригіналу при рівні значимості 0.05")  
  
print(f"y = {b[0]} + {b[1]} \* x1 + {b[2]} \* x2 + {b[3]} \* x3")

**Результати роботи програми:**

average\_Y: [238.0, 246.66666666666666, 248.33333333333334, 226.0]

MX: [32.0, 58.25, 36.5]

MY: 239.75

A: [7658.25, 13903.0, 8791.583333333332]

A11: [1055.5]

A22: [3582.75]

A33: [1385.5]

A12 = A21: [1934.75]

A13 = A31: [1159.25]

A23 = A32: [2113.75]

--------------------------------------------------

b0: 247.82759263156913

b1: 5.6856901968098015

b2: -0.14884919461883592

b3: 1.44886707690974

--------------------------------------------------

Y1: 443.5286542457003

Y2: 447.287847623983

Y3: 513.2754031403892

Y4: 491.83954159049193

Dispersiya: [160.66666666666666, 16.88888888888889, 9.555555555555555, 8.666666666666666]

--------------------------------------------------------------------------------

Перевірка однорідності дисперсії за критерієм Кохрена:

Дсперсія не однорідна. Потрібно збільшити m

Beta: [239.75, -2.5833333333333215, -3.4166666666666714, 7.75]

t0: [118.71279690846198, 1.2791437970671897, 1.6917708283791966, 3.8374313912015867]

--------------------------------------------------------------------------------

Оцінка значимості коефіцієнтів регресії згідно критерію Стьюдента:

t[0] = 118.71279690846198 > Tf = 2.306 >= b[0] = 247.82759263156913 - значний коефіцієнт

t[1] = 1.2791437970671897 <= Tf = 2.306 >= b[1] = 5.6856901968098015 - не значний коефіцієнт

t[2] = 1.6917708283791966 <= Tf = 2.306 >= b[2] = -0.14884919461883592 - не значний коефіцієнт

t[3] = 3.8374313912015867 > Tf = 2.306 >= b[3] = 1.44886707690974 - значний коефіцієнт

[308.6800098617782, 294.19133909268083, 313.02661109250744, 286.9470037081321]

--------------------------------------------------------------------------------

Критерій Фішера

S\_ad: [22731.02359722861]

Fp: 5573.099996596346

рівняння регресії неадекватно оригіналу при рівні значимості 0.05

y = 247.82759263156913 + 5.6856901968098015 \* x1 + -0.14884919461883592 \* x2 + 1.44886707690974 \* x3