Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Методи оптимізації та планування експерименту

Лабораторна робота №3

**«**ПРОВЕДЕННЯ ТРЬОХФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ.**»**

Виконав:

студент групи ІВ-83

Головенець Р.С.

Перевірив:

ас. Регіда П.Г.

Київ

2020 р.

**Лістинг програми**

**import** random  
**from** numpy **import** linalg  
**import** scipy.stats  
**import** math  
x1min,x1max,x2min,x2max,x3min,x3max,N,m,key=10,40,15,50,10,30,4,3,0  
Average\_max=(x1max+x2max+x3max)/3  
Average\_min=(x1min+x2min+x3min)/3  
ymin=round(200+Average\_min)  
ymax=round(200+Average\_max)  
print(**"Рівняння регресії : \n y=b0+b1\*x1+b2\*x2+b3\*x3"**)  
X = [[-1.0, -1.0, -1.0],  
 [-1.0, 1.0, 1.0],  
 [1.0, -1.0, 1.0],  
 [1.0, 1.0, -1.0]]  
MatrixX = [[x1min, x2min, x3min],  
 [x1min, x2max, x3max],  
 [x1max, x2min, x3max],  
 [x1max, x2max, x1min]]  
print (**'Матриця X :'**)  
**for** i **in** range(0, 4):  
 print(MatrixX[i])  
**while True**:  
 MatrixY, Average, Dispersion, Beta, t = [], [], [], [], []  
 print (**'Матриця Y с діапазоном('**,ymin,**','**,ymax,**')'**)  
 **for** i **in** range(0, 4):  
 MatrixY.append([random.randint(ymin, ymax) **for** j **in** range(0, m)])  
 Average.append(sum(MatrixY[i]) / len(MatrixY[i]))  
 Dispersion.append(sum((k - Average[i]) \*\* 2 **for** k **in** MatrixY[i]) / len(MatrixY[i]))  
 print(MatrixY[i])  
 print(**"Середні значення Y: \n"**,round(Average[0],3),round(Average[1],3),round(Average[2],3),round(Average[3],3))  
 mx1 = (x1min+x1min+x1max+x1max) / 4  
 mx2 = (x2min+x2max+x2min+x2max) / 4  
 mx3 = (x3min+x3max+x3max+x3min) / 4  
 my = (Average[0]+Average[1]+Average[2]+Average[3])/4  
 a1 = sum([MatrixX[i][0] \* Average[i] **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 a2 = sum([MatrixX[i][1] \* Average[i] **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 a3 = sum([MatrixX[i][2] \* Average[i] **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 a11 = sum([MatrixX[i][0] \*\* 2 **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 a22 = sum([MatrixX[i][1] \*\* 2 **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 a33 = sum([MatrixX[i][2] \*\* 2 **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 a12 = sum([MatrixX[i][0] \* MatrixX[i][1] **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 a13 = a31 = sum([MatrixX[i][0] \* MatrixX[i][2] **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 a23 = a32 = sum([MatrixX[i][1] \* MatrixX[i][2] **for** i **in** range(len(MatrixX))]) / len(MatrixX)  
 det = linalg.det([[1, mx1, mx2, mx3],  
 [mx1, a11, a12, a13],  
 [mx2, a12, a22, a32],  
 [mx3, a13, a23, a33]])  
 det0 = linalg.det([[my, mx1, mx2, mx3],  
 [a1, a11, a12, a13],  
 [a2, a12, a22, a32],  
 [a3, a13, a23, a33]])  
 det1 = linalg.det([[1, my, mx2, mx3],  
 [mx1, a1, a12, a13],  
 [mx2, a2, a22, a32],  
 [mx3, a3, a23, a33]])  
 det2 = linalg.det([[1, mx1, my, mx3],  
 [mx1, a11, a1, a13],  
 [mx2, a12, a2, a32],  
 [mx3, a13, a3, a33]])  
 det3 = linalg.det([[1, mx1, mx2, my],  
 [mx1, a11, a12, a1],  
 [mx2, a12, a22, a2],  
 [mx3, a13, a23, a3]])  
 b0 = det0 / det  
 b1 = det1 / det  
 b2 = det2 / det  
 b3 = det3 / det  
 b = [b0, b1, b2, b3]  
 print(**'Отримане рівняння регресії: \n'**,round(b0,3),**' + '**,round(b1,3),**' \* x1 +'**,round(b2,3),  
 **' \* x2 +'**,round(b3,3),**' \* x3'**)  
 **for** i **in** range(len(MatrixX)):  
 print(**"y = b0 + b1 \* x1 + b2 \* x2 +b3 \* x3= b0 + b1 \* "**, MatrixX[i][0], **" + b2 \* "**, MatrixX[i][1],  
 **"+b3 \* "**, MatrixX[i][2], **" = "**, round(b0 + b1 \* MatrixX[i][0] + b2 \* MatrixX[i][1]+b3\*MatrixX[i][2], 3))  
 print(**"Дисперсії: \n"**,round(Dispersion[0],3),round(Dispersion[1],3),round(Dispersion[2],3),round(Dispersion[3],3))  
 print(**"Перевіримо критерії Кохрена"**)  
 Gp = max(Dispersion) / sum(Dispersion)  
 f1 = m - 1  
 f2 = 4  
 q = 0.05  
 tableGt = {2: 7679,3: 0.6841, 4: 0.6287, 5: 0.5892, 6: 0.5598, 7: 0.5365, 8: 0.5175, 9: 0.5017, 10: 0.4884,  
 range(11, 17): 0.4366,range(17, 37): 0.3720, range(37, 145): 0.3093}  
 Gt = tableGt.get(m)  
 print(**'Gp='**,round(Gp,3),**", Gt="**,Gt)  
 **if** Gp<Gt:  
 print(Gp, **"<="**, Gt)  
 print(**"Дисперсія однорідна"**)  
 **else**:  
 print(Gp, **">="**, Gt)  
 print(**"Дисперсія не однорідна"**)  
 m += 1  
 key=1  
 **if** key==0:  
 print(**"перевіримо критерій Стьюдента"**)  
 S2betaSum = sum(Dispersion) / N  
 S2beta = S2betaSum / (N \* m)  
 Sbeta = math.sqrt(S2beta)  
 MatrixCodeX = [[1.0, -1.0, -1.0, -1.0],  
 [1.0, -1.0, 1.0, 1.0],  
 [1.0, 1.0, -1.0, 1.0],  
 [1.0, 1.0, 1.0, -1.0]]  
 **for** i **in** range(4):  
 Beta.append(round(sum([MatrixCodeX[j][i] \* Average[j] **for** j **in** range(len(MatrixCodeX))])/N,3))  
 t.append(round(abs(Beta[i]/Sbeta),3))  
 print(**"t: "**, t)  
 f3 = f1 \* f2  
 print(**"f3="**, f3)  
 tableS = round(scipy.stats.t.ppf((1 + (1 - q)) / 2, f3),3)  
 print(**"Табличне значення ="**,tableS)  
 **for** i **in** range(4):  
 **if** t[i] < tableS:  
 b[i] = 0  
 print(t[i], **"<"**, tableS)  
 y = []  
 y.append(round(b0 + b1 \* X[0][0] + b2 \* X[0][1] + b3 \* X[0][2]))  
 y.append(round(b0 + b1 \* X[1][0] + b2 \* X[1][1] + b3 \* X[1][2]))  
 y.append(round(b0 + b1 \* X[2][0] + b2 \* X[2][1] + b3 \* X[2][2]))  
 y.append(round(b0 + b1 \* X[3][0] + b2 \* X[3][1] + b3 \* X[3][2]))  
 print(**"y: "**, y)  
 **for** i **in** range(len(y)):  
 print(y[i], **"=="**, round(Average[i],3))  
 print(**"Перевіримо критерій Фішера"**)  
 d = 0  
 **for** i **in** range(len(b)):  
 **if** b[i] != 0:  
 d += 1  
 print(**"d="**, d)  
 f4 = N - d  
 print(**"f4="**,f4)  
 Sum = 0  
 **for** i **in** range(len(y)):  
 Sum += pow((y[i] - Average[i]), 2)  
 Sad = (m / (N - d)) \* Sum  
 Fp = Sad / S2betaSum  
 print(**"Fp="**,round(Fp,3))  
 Ft = round(scipy.stats.f.ppf(1 - q, f4, f3),3)  
 print(**"Ft="**, Ft)  
 **if** Fp > Ft:  
 print(**"Рівняння регресії неадекватно оригіналу при рівні значимості 0.05"**)  
 **break  
 else**:  
 print(**"Рівняння регресії адекватно оригіналу при рівні значимості 0.05"**)  
 **break  
 else**:  
 key=0

**Відповіді на контрольні запитання**

1. Що називається дробовим факторним експериментом?

Скорочена кількість дослідів, використовуючи для планування так звані регулярні дробові репліки від повного факторного експерименту, що містять відповідну кількість дослідів і зберігають основні властивості матриці планування.

2. Для чого потрібно розрахункове значення Кохрена?

Для перевірки однорідності дисперсії

3. Для чого перевіряється критерій Стьюдента?

Для перевірки значущості коефіціентів

4. Чим визначається критерій Фішера і як його застосовувати?

Визначається відношенням дисперсії адекватності до дисперсії відтворюваності.

Знайдене шляхом розрахунку Fp порівнюють з табличним значенням Fт, що визначається при

рівні значимості q та кількості ступенів свободи f4 = N - d і

f3 = f1 \* f2

Якщо Fp< Fт то отримана математична модель з прийнятим рівнем статистичної значимості q

адекватна експериментальним даним.

**Результат виконання роботи:**



