МІНІСТЕРСТЕРСВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛБНОЇ ТЕХНІКИ

Звіт з лабораторних робіт

З курсу «Методи оптимізаціі та планування експерименту»

Лабораторна робота №3

Виконала:

Бугайченко П.І.

Номер заліковки: 8203

Перевірив:

Регіда П.Г.

Київ 2020 р.

**Виконання роботи**

**Варіант 202 (70,20)(80,30)(35, 30)**

import random  
import numpy as np  
  
koh = [0, 9065, 7679, 6841, 6287, 5892, 5598, 5365, 5175, 5017, 4884]}  
student = {8: 2.306, 12: 2.179, 16: 2.120, 20: 2.086, 14: 2.064, 28: 2.048}  
  
x1 = [20, 70]  
x2 = [30, 80]  
x3 = [30, 35]  
x\_min\_aver = (x1[0] + x2[0] + x3[0]) / 3  
x\_max\_aver = (x1[1] + x2[1] + x3[1]) / 3  
y\_min = 200 + x\_min\_aver  
y\_max = 200 + x\_max\_aver  
  
incrementation = True  
  
m = 3  
n = 4  
while incrementation:  
 f1 = m - 1  
 f2 = n  
 f3 = f1 \* f2  
  
 usual\_matrix = [[20, 20, 30], [20, 80, 35], [70, 30, 35], [70, 80, 30]]  
 norm\_matrix = [[1, -1, -1, -1], [1, -1, 1, 1], [1, 1, -1, 1], [1, 1, 1, -1]]  
  
 average\_ys = []  
 mx = []  
 a = []  
 a\_duplicate = []  
  
 dispersion = []  
 b = []  
 kohren\_exp = 0  
 beta = []  
 t\_kof = []  
  
  
 # случайно сгенер. у для матриц  
 for i in range(4):  
 for t in range(m):  
 new\_y = random.randint(y\_min, y\_max)  
 usual\_matrix[i].append(new\_y)  
 norm\_matrix[i].append(new\_y)  
  
  
 # средние игрики  
 for i in range(4):  
 ay = 0  
 for k in range(m):  
 ay += usual\_matrix[i][k + 3]  
 average\_ys.append(ay / m)  
  
  
  
 for i in range(3):  
 t = 0  
 k = 0  
 d = 0  
 for k in range(4):  
 d += usual\_matrix[k][i] \* average\_ys[k]  
 t += usual\_matrix[k][i]  
 k += usual\_matrix[k][i] \*\* 2  
 mx.append(t / n)  
 a\_duplicate.append(k / n)  
 a.append(k / n)  
  
 a12 = (x[1][1] \* x[1][2] + x[2][1] \* x[2][2] + x[3][1] \* x[3][2] + x[4][1] \* x[4][2]) / 4  
 a13 = (x[1][1] \* x[1][3] + x[2][1] \* x[2][3] + x[3][1] \* x[3][3] + x[4][1] \* x[4][3]) / 4  
 a23 = (x[1][2] \* x[1][3] + x[2][2] \* x[2][3] + x[3][2] \* x[3][3] + x[4][2] \* x[4][3]) / 4  
  
 my = sum(average\_ys) / n  
  
  
 arr0 = [[my, mx[0], mx[1], mx[2]], [a[0], a\_duplicate[0], a12, a13], [a[1], a12, a\_duplicate[1], a32], [a[2], a13, a23, a\_duplicate[2]]]  
 arr1 = [[1, my, mx[1], mx[2]], [mx[0], a[0], a12, a13], [mx[1], a[1], a\_duplicate[1], a32], [mx[2], a[2], a23, a\_duplicate[2]]]  
 arr2 = [[1, mx[0], my, mx[2]], [mx[0], a\_duplicate[0], a[0], a13], [mx[1], a12, a[1], a32], [mx[2], a13, a[2], a\_duplicate[2]]]  
 arr3 = [[1, mx[0], mx[1], my], [mx[0], a\_duplicate[0], a12, a[0]], [mx[1], a12, a\_duplicate[1], a[1]], [mx[2], a13, a23, a[2]]]  
 b = [[1, mx[0], mx[1], mx[2]], [mx[0], a\_duplicate[0], a12, a13], [mx[1], a12, a\_duplicate[1], a32], [mx[2], a13, a23, a\_duplicate[2]]]  
  
 detb = np.linalg.det(b)  
  
 b0 = np.linalg.det(arr0) / detb  
 b1 = np.linalg.det(arr1) / detb  
 b2 = np.linalg.det(arr2) / detb  
 b3 = np.linalg.det(arr3) / detb  
 b.append(b0)  
 b.append(b1)  
 b.append(b2)  
 b.append(b3)  
  
  
 for i in range(4):  
 s = 0  
 for t in range(m):  
 s += (usual\_matrix[i][t + 3] - average\_ys[i]) \*\* 2  
 dispersion.append(s / m)  
  
  
  
 kohren\_exp = (dispersion) / sum(dispersion)  
  
 if kohren\_exp < koh[m]/10000:  
 incrementation = False  
 print("Дисперсія однорідна")  
 else:  
 m += 1  
 print("Дисперсія неоднорідна")  
  
 sb = sum(dispersion) / n  
 sb\_beta2 = sb / (n \* m)  
 sb\_beta = sqrt(sb\_beta2)  
  
 for i in range(4):  
 b = 0  
 for k in range(4):  
 b += average\_ys[k] \* norm\_matrix[i][k]  
 beta.append(b / n)  
 t\_kof.append(abs(b) / sb\_beta)  
  
 for i in range(4):  
 if t\_kof[i] < student[f3]:  
 b[i] = 0  
  
print(" m = ", m)  
print("рівняння : y = {} + {} \* x1 + {} \* x2 + {} \* x3".format(b[0], b[1], b[2], b[3]))

Контрольні запитання

1. Що називається дробовим факторним експериментом?

У деяких випадках немає необхідності проводити повний факторний експеримент (ПФЕ). Якщо буде використовуватися лінійна регресія, то можливо зменшити кількість рядків матриці ПФЕ до кількості коефіцієнтів регресійної моделі. Кількість дослідів слід скоротити, використовуючи для планування так звані регулярні дробові репліки від повного факторного експерименту, що містять відповідну кількість дослідів і зберігають основні властивості матриці планування – це означає дробовий факторний експеримент (ДФЕ).

2. Для чого потрібно розрахункове значення Кохрена?

Для перевірки дисперсії на однорідність.

3. Для чого перевіряється критерій Стьюдента?

Для перевірки значущості коефіцієнтів регресії. Тобто, Якщо виконується нерівність ts< tтабл, то приймається нуль-гіпотеза, тобто вважається, що знайдений коефіцієнт βs є статистично незначущим і його слід виключити з рівняння регресії. Якщо ts > tтабл то гіпотеза не підтверджується, тобто βs – значимий коефіцієнт і він залишається в рівнянні регресії.

4. Чим визначається критерій Фішера і як його застосовувати?

Отримане рівняння регресії необхідно перевірити на адекватність досліджуваному об'єкту. Для цієї мети необхідно оцінити, наскільки відрізняються середні значення у вихідної величини, отриманої в точках факторного простору, і значення у, отриманого з рівняння регресії в тих самих точках факторного простору. Для цього використовують дисперсію адекватності. Адекватність моделі перевіряють за F-критерієм Фішера, який дорівнює відношенню дисперсії адекватності до дисперсії відтворюваності.